Autonomous and Semi-autonomous Control of Agile Fixed-Wing Unmanned Aerial Vehicles

Juan Carlos Hernández Ramírez

The Department of Mechanical Engineering
McGill University, Montreal

April 2022

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Doctor of Philosophy.

Abstract

Unmanned aerial vehicles (UAVs) have been increasingly employed for a wide range of civilian applications, including scientific and commercial endeavours. Striving for versatility, novel UAV designs have been proposed which incorporate features from different aircraft categories. The agile fixed-wing UAV is one such platform, which combines the fast and efficient flight of conventional fixed-wing aircraft with the highly maneuverable hovering capabilities of rotorcraft. These aircraft have enormous potential but are challenging to operate, demonstrating the need for control systems capable of harnessing their maneuvering capabilities.

The objective of this thesis is to develop a comprehensive control strategy for the agile fixed-wing UAV, considering both autonomous, and pilot-assist, semi-autonomous operation. The thesis begins by introducing the fixed-wing UAV model, including previously developed aerodynamic and slipstream models needed for controller design and verification. Controller design begins in the next chapter, where the core attitude control system is developed. We propose several geometric control strategies which are then evaluated through simulations and experiments.

The next chapter presents a nonlinear control strategy to enable UAV velocity and position control. This is a unified control architecture, where a single controller is valid for steady and hover flight regimes. Rigorous stability analysis, together with simulations and experimental flights demonstrate the capabilities of the proposed system. The following chapter, we propose an additional control loop to enable path-following functionality, allowing us to prioritize the task of reaching and following spatial paths. As part of this system, a novel moving path frame is developed, specifically tailored for the task of motion control in three-dimensional space.

Finally, a reference generator capable of translating limited pilot inputs into time-parametrized trajectories or spatial paths is developed. This system is then integrated with the control systems to achieve a semi-autonomous control solution. As demonstrated through experimental flights, the proposed system enables a human pilot to safely operate the aircraft in any regime without sacrificing its maneuverability.

Résumé

Les véhicules aériens sans pilote (UAV) sont de plus en plus suggérés pour plusieurs applications civiles, y compris des efforts scientifiques et commerciaux. En quête de versatilité, de nouveaux modéles ont été proposées qui intègrent des caractéristiques de différentes catégories d'aéronefs. L'aéronef agile à voilure fixe est l'une de ces plates-formes, qui combine le vol rapide et efficace des aéronefs à voilure fixe conventionnels avec les capacités de vol stationnaire hautement-maniables des giravions. Ces aéronefs ont un potentiel énorme mais sont très difficiles à piloter, démontrant le besoin de systèmes de contrôle capables d'exploiter leurs agilité.

L'objectif de cette thèse est de développer une stratégie de contrôle complète pour l'aéronef agile à voilure fixe, en tenant compte à la fois du fonctionnement autonome et semi-autonome d'assistance du pilote. La thèse commence par introduire le modèle d'àéronef agile, incluant les modèles aérodynamiques et ce du systéme de propulsion développés précédemment, nécessaires lors de la conception et de la vérification du contrôleur. Le développement du contrôleur commence dans le chapitre suivant, où le système de contrôle d'orientation de base est développé. Nous proposons plusieurs stratégies de contrôle géométrique qui sont ensuite évaluées à travers des simulations et des vols expérimentaux.

Le chapitre suivant présente une stratégie de contrôle non linéaire pour permettre le contrôle de la vélocité et de la position de l'aéronef. Il s'agit d'une architecture de contrôle unifiée, valide pour les manoeuvres conventionnelles et pour les régimes de vol stationnaire. Une analyse de stabilité rigoureuse, ainsi que des simulations et des vols expérimentaux démontrent les capacités du système proposé. Dans le chapitre suivant, nous proposons un système de contrôle de suivi-de-sentier. Ce contrôleur nous permet de prioriser l'objectif d'atteindre et de suivre des courbes tridimensionelles. Pour permettre cette fonctionnalité, un nouveau référentiel, attaché à la courbe, est développé, spécifiquement adapté pour l'application du contrôle de mouvement dans l'espace tridimensionnel.

Enfin, une stratégie pour généer des références est développé, capable de traduire des commandes du pilote en trajectoires paramétrées dans le temps ou en courbes en trois dimensions. Ce système est ensuite intégré aux systèmes de contrôle pour obtenir une solution de contrôle semi-autonome. Nous montrons à travers des vols expérimentaux, que le système proposé permet à un opérateur à distance de piloter l'aéronef en toute sécurité dans n'importe quel régime sans sacrifier l'agilité de la plate-forme.

Acknowledgements

First and foremost, I would like to thank my supervisor, Prof. Meyer Nahon, for guiding me through my graduate studies. I will always be grateful for all the patience, unwavering support, constructive criticism, and for the trust placed in me throughout these years. Each time I needed help, he was always available to meet and discuss, and for that, I am especially thankful. It is precisely through his sincere guidance and advise that I have grown as a researcher, and as a person. I am fortunate to have had him as my mentor.

I also thank my thesis advisory committee, Professors Benoit Boulet and James Forbes, for their continued interest on my research over the years, and for their useful feedback during our meetings which has helped improve this work. I also wish to thank Prof. Inna Sharf, who, while not directly involved in my research, was always available to talk about ideas and how our tests had worked out.

I would like to thank my friends and colleagues in the Aerospace Mechatronics Laboratory. I have been very fortunate to share not only the lab, but my research journey, with such a talented, kind, and fun group of fellow researchers. I wish to especially thank Eitan Bulka, Fares El Tin, Jad Wehbeh, and Jackson Empey; those who've regularly stood with me during the sometimes rewarding, sometimes frustrating, but always exciting task of flight testing. Your help and support has been invaluable, and your refusal to give up has been an inspiration. I could not ask for better partners. I am also thankful to Hunter Song, Mikkel Jorgensen, Romain Chiappinelli, Christian Patience, Joshua Levin, Sahand Rezaei, Walter Jothiraj, Corey Miles, and Zihao Zhou, you have truly made my research experience much richer, and much happier. I am also grateful to Siddharth Kumanduri and Tim Thompson, whose work made my initial flight tests possible.

Next, I would like to thank the Macdonald Campus of McGill University, the West Island Model Aeronautics Club (WIMAC), the Montreal Area Thermal Soarers Club (MATS) for providing a space to perform the experimental tests of this work.

Finally, I would like express my heartfelt gratitude to my family, for all their love and support. To my parents, Rogelio and Rosalba, who always encouraged me to pursue further goals, and whose unwavering support and guidance have allowed me to do so. To my brother, Gonzalo, for his constant encouragement and support, for looking after all of us, and for being the best friend I could ask for. To my wife, Angy, who embarked with me in yet another adventure, for her endless patience, love, and support. I could not have completed this work

without you, thank you for sharing this journey with me. To those that have departed, you are forever in my heart.

This work was made possible with the financial support from the Consejo Nacional de Ciencia y Tecnología (CONACyT), the Fonds de Recherche du Québec Nature et Technologies (FRQNT), the Natural Sciences and Engineering Research Council (NSERC), and McGill's Faculty of Engineering through the McGill Engineering Doctoral Award (MEDA).

Claims of Originality

The main contributions of this thesis are as follows:

- An in-depth study into geometric control techniques applied to the attitude control problem of agile fixed-wing UAV is conducted. Stability analysis is used to establish controller properties which are then evaluated through extensive simulations and experimental flights.
- A novel nonlinear control architecture, for position and velocity control of the agile fixed-wing UAV platform is presented. This unified strategy is valid for all flight regimes. Stability analysis, together with simulations and flight experiments are used to verify the proposed control system.
- A novel path-moving reference frame, specifically tailored to describe curves used for 3D motion is proposed and its properties derived. The applications of this construct apply to a wide range of vehicles and control and path planning applications, not only limited to fixed-wing UAVs.
- A path following controller is developed, using the tacking control formulation together with the proposed reference frame to enable the UAV to reach and follow spatial paths, as opposed to instantaneous position targets. Simulations and experimental flights of the path following solution show the improvement in performance when in outdoor operation.
- A semi-autonomous control strategy is developed to enable pilot-assist functionality. A reference generator is proposed to translate limited inputs from a remote operator into reference trajectories and geometries for the corresponding control systems. The methodology is tested with a human operator in both software-in-the-loop and in experimental flights.

Large parts of this thesis have appeared in the following publications:

- Juan C. Hernandez and Meyer Nahon. Trajectory tracking control of highly maneuverable fixed-wing unmanned aerial vehicles. In *AIAA Scitech 2020 Forum*, pages 2074-2090. AIAA, 2020. DOI: 10.2514/6.2020-2074

- Juan C. Hernandez and Meyer Nahon. Nonlinear vector-projection control for agile fixed-wing unmanned aerial vehicles. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5314-5320. IEEE, 2020.
  DOI: 10.1109/ICRA40945.2020.9196838

- Juan C. Hernandez and Meyer Nahon. A Gravity-Referenced Moving Frame for Vehicle Path Following Applications in 3D. *IEEE Robotics and Automation Letters*, 6(3): 4393-4400, July 2021. DOI: 10.1109/LRA.2021.3068554

- Juan C. Hernandez and Meyer Nahon. Pilot-Assist Landing System for Hover-Capable Fixed-Wing Unmanned Aerial Vehicles in All Flight Regimes. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1179-1186. IEEE, 2021. DOI: 10.1109/ICUAS51884.2021.9476776

- Juan C. Hernandez and Meyer Nahon. Nonlinear Control for Semi-autonomous Operation of Agile Fixed-Wing Unmanned Aerial Vehicles. Under review with *Journal of Intelligent Robotics.*

- Juan C. Hernandez and Meyer Nahon. Aerobatics on Three-Dimensional Paths for Agile Fixed-Wing Unmanned Aerial Vehicles. Submitted to *2022 International Conference on Unmanned Aircraft Systems (ICUAS).*

Notation

Abbreviations

UAV	**U**nmanned **A**erial **V**ehicle
VTOL	**V**ertical **T**ake-**O**ff and **L**anding
GPS	**G**lobal **P**ositioning **S**ystem
IMU	**I**nertial **M**easurement **U**nit
RC	**R**adio **C**ontrol
3D	**T**hree-**D**imensional
DCM	**D**irection **C**osine **M**atrix
SO(3)	**S**pecial **O**rthogonal Group of **3**D rotations
NED	**N**orth-**E**ast-**D**own
AML	**A**erospace **M**echatronics **L**aboratory
DOF	**D**egrees **O**f **F**reedom
MPC	**M**odel **P**redictive **C**ontrol
LQR	**L**inear **Q**uadratic **R**egulator
PD	**P**roportional-**D**erivative
PI	**P**roportional-**I**ntegral
PID	**P**roportional-**I**ntegral-**D**erivative
MEKF	**M**ultiplicative **E**xtended **K**alman **F**ilter
RPM	**R**evolutions **P**er **M**inute
ESC	**E**lectronic **S**peed **C**ontroller
FPV	**F**irst **P**erson **V**iew
LOS	**L**ine **O**f **S**ight
SITL	**S**oftware **I**n **T**he **L**oop
QGC	**QG**round **C**ontrol
PF	**P**ath **F**ollowing

Symbols

t	Time.
$\boldsymbol{\mathcal{F}}_I$	Inertial (NED) frame.
$\boldsymbol{\mathcal{F}}_B$	Body frame.
$\boldsymbol{\mathcal{F}}_R$	Reference frame.
$\boldsymbol{\mathcal{F}}_G$	Moving path (gravity normal) frame.
m	UAV mass.
g	Gravity acceleration.
$\mathbf{J}$	UAV second moment of mass, resolved in $\boldsymbol{\mathcal{F}}_B$.
$\delta_a, \delta_e, \delta_r$	Aileron, elevator, and rudder deflections.
T	Motor thrust.
$\mathbf{C}_{bi}$	DCM from $\boldsymbol{\mathcal{F}}_I$ to $\boldsymbol{\mathcal{F}}_B$.
$\mathbf{C}_{ri}$	DCM from $\boldsymbol{\mathcal{F}}_I$ to $\boldsymbol{\mathcal{F}}_R$.
$\hat{\mathbf{r}}_1$, $\hat{\mathbf{r}}_2$, $\hat{\mathbf{r}}_3$	Basis vectors of $\boldsymbol{\mathcal{F}}_R$ resolved in $\boldsymbol{\mathcal{F}}_I$.
$\hat{\mathbf{k}}_1$, $\hat{\mathbf{k}}_2$, $\hat{\mathbf{k}}_3$	Basis vectors of $\boldsymbol{\mathcal{F}}_I$ resolved in $\boldsymbol{\mathcal{F}}_I$.
$\mathbf{C}_{gi}$	DCM from $\boldsymbol{\mathcal{F}}_I$ to $\boldsymbol{\mathcal{F}}_G$.
$\mathbf{C}_{br}$	Error DCM from $\boldsymbol{\mathcal{F}}_R$ to $\boldsymbol{\mathcal{F}}_B$.
$\boldsymbol{\omega}_b$	Angular velocity between $\boldsymbol{\mathcal{F}}_B$ and $\boldsymbol{\mathcal{F}}_I$, resolved in $\boldsymbol{\mathcal{F}}_B$.
$\boldsymbol{\omega}_r$	Angular velocity between $\boldsymbol{\mathcal{F}}_R$ and $\boldsymbol{\mathcal{F}}_I$, resolved in $\boldsymbol{\mathcal{F}}_R$.
$\boldsymbol{\omega}_g$	Angular velocity between $\boldsymbol{\mathcal{F}}_G$ and $\boldsymbol{\mathcal{F}}_I$, resolved in $\boldsymbol{\mathcal{F}}_G$.
$\mathbf{p}_i$	UAV's centre of mass position resolved in $\boldsymbol{\mathcal{F}}_I$.
$\mathbf{p}_r$	Reference position $\boldsymbol{\mathcal{F}}_R$
$\mathbf{v}_b$	UAV velocity with respect to $\boldsymbol{\mathcal{F}}_B$, resolved in $\boldsymbol{\mathcal{F}}_B$.
$\mathbf{v}_i$	UAV velocity with respect to the $\boldsymbol{\mathcal{F}}_I$, resolved in $\boldsymbol{\mathcal{F}}_I$.
$\mathbf{v}_r$	Reference velocity, resolved in $\boldsymbol{\mathcal{F}}_I$
Ψ_a	Attitude error function, $a = 1, 2, 3$.
$\mathbf{e}_a$	Attitude error innovation term, $a = 1, 2, 3$.
$\mathbf{e}_\omega$	Angular velocity error.
$\mathbf{e}_p$	Position error.
$\mathbf{e}_{pr}$	Path-relative position error.
$\mathbf{e}_v$	Velocity error.
$\mathbf{e}_i$	Composite integrated position error.
$\mathbf{e}_{vi}$	Integrated velocity error.

$\boldsymbol{\mu}(\sigma)$	Curve in 3D space.
σ	Path parametrizing variable.
$\mathbf{T}$	Path tangent vector resolved in $\boldsymbol{\mathcal{F}}_I$.
$\mathbf{H}$	Path horizontal vector resolved in $\boldsymbol{\mathcal{F}}_I$.
$\mathbf{P}$	Path projected vector resolved in $\boldsymbol{\mathcal{F}}_I$.
V_a	Airspeed.
α	Angle of attack.
χ	Course angle.
γ	Climb angle.
C_L, C_D, C_m	Lift, drag, and moment coefficients.
$V_\delta, \hat{V}_\delta$	Effective and estimated airflow over the control surfaces.
S	Wing area.
b	Wing span.
$\bar{c}$	Wing mean aerodynamic chord.
C_{δ_a}, C_{δ_e}, C_{δ_r}	Control coefficients for aileron, elevator, and rudder.
ϕ, θ, ψ	Euler angles: roll, pitch, and yaw.
$\eta, \boldsymbol{\epsilon}$	Principal axis and angle of rotation.
S_T	Throttle, or thrust, pilot input.
S_a, S_e, S_r	Pilot inputs of aileron, elevator, and rudder.
$\mathbf{F}_c$	Intermediate command (position control).
$\mathbf{F}_a$	Aerodynamic forces resolved in $\boldsymbol{\mathcal{F}}_B$.
k_a, $\mathbf{K}_a$	Proportional attitude control gains.
k_ω, $\mathbf{K}_\omega$	Derivative attitude control gains.
k_p, $\mathbf{K}_p$	Proportional position control gains.
k_v, $\mathbf{K}_v$	Derivative position control gains.
k_i, $\mathbf{K}_i$	Integral position control gains.
k_y, k_{ϕ_p}, k_{ϕ_i}	Roll control gains.
η_l, η_u	Upper and lower angle thresholds.
$(\cdot)^T$	Transpose operator.
$(\cdot)^\times$	Skew-symmetric operator.
$\mathcal{P}_a(\cdot)$	Skew-symmetric projection operator.
$\mathrm{tr}(\cdot)$	Trace operator.
$\lVert \cdot \rVert$	Euclidean norm.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Originally developed for military applications, unmanned aerial vehicles (UAVs) have become increasingly prevalent in civilian and commercial applications [1]. This shift in interests has, in turn, prompted and increase in research and development, both in academia and the industry. UAVs are appealing platforms: compared to conventional aircraft with onboard pilots, they are lightweight, low-cost, multipurpose and, perhaps most importantly, safe for a remote operator [2]. UAVs are now used for several commercial and scientific endeavours including mapping, surveillance, experimental data gathering, search and rescue, and transportation and delivery of goods, among other applications [3, 4, 5].

1.1 Motivation

UAVs are roughly classified in two large categories: rotorcraft and fixed-wing aircraft. Rotorcraft are equipped with multiple motors and propellers, allowing them to remain airborne through thrust alone. They are highly maneuverable and capable of hovering in place, which in turn allows them to perform vertical take-offs and landings. These platforms are suited for applications that require precise, low-speed maneuvers or stopping mid flight. Fixed-wing UAVs rely on lift generated through their lifting surfaces to remain airborne, while devoting their available thrust to accelerate. They are much more energetically efficient than rotorcraft, making them the platform of choice for tasks requiring large area coverage, high velocity flight, and endurance.

The agile fixed-wing UAV is a modern class of aircraft designed to bridge the gap between these two categories. While similar in appearance to a conventional fixed-wing aircraft, this platform, like the one shown in Figure 1.1, possess special features that makes it highly maneuverable. Agile fixed-wing UAVs are characterized by a high thrust-to-weight ratio, low aspect ratio wings, and large control surfaces capable of large deflections. Thrust is generated by a powerful motor and propeller combination, and with a thrust-to-weight ratio larger than one, the aircraft can remain airborne and accelerate through thrust alone. The propulsion system serves another important purpose: it generates significant slipstream (or propwash). The design of the aircraft's control surfaces capitalizes on slipstream by using this additional airflow to retain control authority regardless of the vehicle's airspeed.

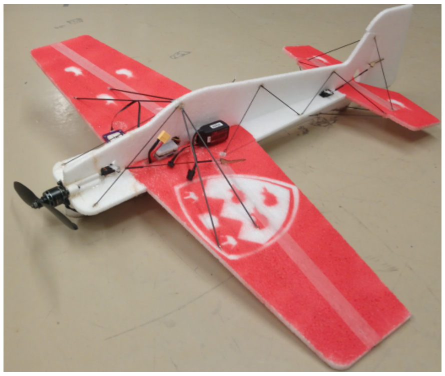


Figure 1.1: An agile fixed-wing UAV.

An agile fixed-wing can be flown in the conventional steady, level flight regime but it is capable of a much wider range of maneuvers [6]. Since the control surfaces retain control authority even at reduced airspeeds, the aircraft can hover in place and perform several aerobatic maneuvers. Figure 1.2 shows an agile fixed-wing platform in two distinct flight regimes. The outstanding maneuvering capabilities of these aircraft have been demonstrated by expert pilots in the radio control (RC) community where impressive aerobatic maneuvers are performed in events and competitions. There is great potential in the agility and maneuverability of these platforms, but only expert pilots can operate them reliably since their configuration is not inherently stable and aerobatic maneuvers require precise control inputs on multiple channels simultaneously. The maneuverability, reliability, and, as a consequence, the range of applications for this class of aircraft is ultimately dependent on pilot skill and experience [7]. There is a clear need for an autonomous control solution that harnesses the versatility and maneuverability of the aircraft. While the long-term goal is fully autonomous flight, this will not happen in the near-term, especially due to the reservations of regulatory agencies [8]. This fact underscores the need for safe, manual remote operation, which can be achieved through a pilot-assisted, semi-autonomous control system specially tailored to these platforms.

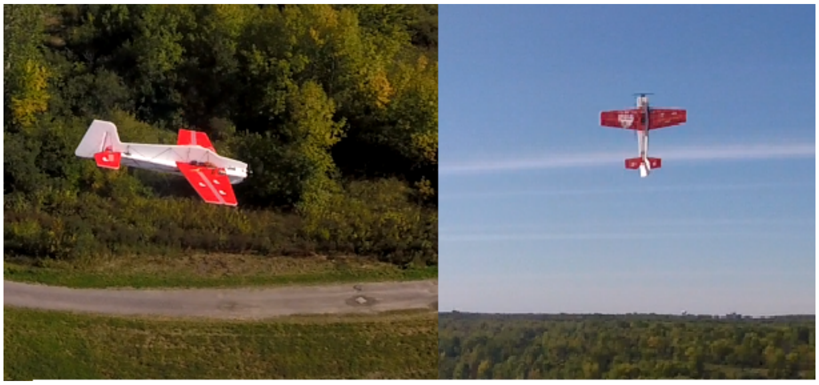


Figure 1.2: An agile fixed-wing UAV in different flight regimes.

1.2 Objectives

This thesis focuses on the development of control strategies for agile fixed-wing UAVs, both in fully autonomous operation, and in semi-autonomous, pilot-assist flight. The control architecture must be valid for the complete range of operation of the aircraft so as to fully exploit the extreme maneuverability of the platform. This includes conventional steady flight, the hover regime, aerobatic maneuvers, and transitions between regimes.

The first objective is the development and verification of an attitude control system that can drive the aircraft to any possible orientation. Since attitude does not naturally evolve in Euclidean space, the design of a global or almost global control system requires special care to avoid singularities, redundancy, and other phenomena. As the core component of the overall UAV control system, it is important that the attitude controller be tested thoroughly.

Once the attitude controller is designed and tested, the focus shifts to the position control system. We consider the problem of commanding an aircraft to follow references in three-dimensional (3D) space without specific information of the flight regime needed to achieve them. This is especially important for the semi-autonomous operation of the aircraft, where pilot input is limited, and the prescribed motions do not easily conform to some specific, precomputed maneuver. As such, the control system needs to be able to operate in different flight regimes, determine if a transition is needed, and remain valid during these transitions.

Trajectory planning for autonomous operation is outside the scope of this work, and has been studied previously [9]. However, several trajectories will need to be designed to test the control system capabilities, both in simulations and experimental flights. In contrast, the generation of trajectories for semi-autonomous operation is a fundamental task, since this component serves as the translator between operator input and control system. This component will determine how the aircraft is piloted. Due to this, there is an additional objective beyond the core control systems: the development of a reference generation strategy that allows the pilot to intuitively prescribe a flight trajectory in each flight regime.

We envision a comprehensive flight control system for the agile fixed-wing UAV platform, capable of operating in all flight regimes and performing aggressive maneuvers, while being versatile enough to accommodate both fully autonomous and pilot-assisted operation, as required, depending on the particular mission.

1.3 Literature review

As the main topic of this work, the literature review focuses on the 3D motion control of UAVs. Agile fixed-wing UAVs are unique platforms which share properties with different UAV categories; hence, the review does not focus solely on fixed-wing aircraft research and explores the contributions in the control design of other relevant UAV classes.

The development of control systems is largely influenced by the characteristics of the target platform. These properties are embedded in the system's dynamic model and understanding them is an important step in identifying the considerations and limitations of the designed control system. To highlight the special characteristics of the agile fixed-wing, this review briefly discusses existing dynamic models of fixed-wing aircraft and their applicability to the platform of interest.

1.3.1 Agile fixed-wing UAV models

Three-dimensional rigid body motion lies at the core of any UAV dynamic model. This topic has been amply studied, and the equations of motion, both translational and rotational, of a rigid body are well established [10, 11]. Of these, the attitude equations warrant special care, since attitude does not evolve in $\mathbb{R}^3$, as position and velocity do. The natural representation of attitude are the direction cosine matrices (DCMs) which belong to the special orthogonal group of three-dimensional rotations (SO(3)). However, several attitude parametrizations exist, and depending on the requirements and scope of each problem, different ones may be used for the purposes of simulation, control design, and implementation [12].

While 3D motion is already a nonlinear problem, the main challenge in obtaining the model of the UAV lies in determining the forces and moments that are applied to the rigid body dynamics. This is particularly complex in the case of fixed-wing aircraft, where forces and moments are generated through aerodynamics. Conventional fixed-wing UAV models [2] are largely inherited from conventional manned aircraft models which have existed for decades. These dynamic models are derived from the Euler angles parametrization which, despite their non global and non unique nature, are well suited for analysis and control design about the steady, level flight regime. This flight condition also justifies the use of simplified models based on stability derivatives, which reduce the aerodynamic analysis complexity [13]. While simplified, these models remain complex, as they are nonlinear and require the use of several look-up tables. For this reason, it common to further simplify them through linearization about specific operating conditions, particularly for control design [14]. Often, several linearized models are required to account for different operational conditions [15].

The particular characteristics of agile fixed-wing UAVs, however, cannot be properly represented with such models. Attitude dynamics require a non-singular representation such as quaternions or DCMs, as is the norm for the more maneuverable rotorcraft platforms [16]. Conventional aerodynamic models are also ill-suited, since they do not account for large angles of attack, the low aspect ratio surfaces, or the large control surfaces [17]. Slipstream, a fundamental characteristic that has shaped the configuration of these platforms, has rarely been studied. In [18], a model based on first principles for the agile fixed-wing UAV was developed. This work serves as the basis of the agile fixed-wing UAV model used throughout this work, and was previously developed and tested by past members of the Aerospace Mechatronics Lab (AML). This model is complemented by a slipstream model [19] and a thruster dynamic model better suited for aerobatic flight [20].

Based on first principles, these models can be applied to very different configurations of aircraft. However, this work focuses on platforms with a specific input configuration, which is that of a conventional fixed-wing aircraft. It is assumed the aircraft has the capability of generating four independent inputs: one body fixed force, and three independent moments. The control allocation problem, how these four forces (and moments) are generated by the available inputs, is platform dependent. While this work focuses on the agile fixed-wing platform shown in Figure 1.1, equipped with one motor and propeller and three independent control surfaces, the results can be applied to any UAV with this input configuration.

1.3.2 UAV control systems

The motion control problem consists of designing a policy for the available inputs such that the aircraft pose, consisting of attitude and position, is driven to some reference. Since 3D motion constitutes a six degree of freedom (DOF) problem and the platform considered in this work has only four inputs, the system is considered underactuated. While only four inputs are available, their configuration is fundamental as it ensures the UAV remains controllable, since the three independent moments guarantee the controllability of the attitude dynamics [21], and position can be controlled by pointing the body fixed force. For this reason, several aerospace platforms share this input configuration, and research on control systems is plentiful.

Control system architecture for UAV platforms can be roughly classified between single, one-step controllers, and cascaded systems. The one-step architecture aims to control position and attitude states simultaneously, often using linear control techniques. Such systems are more common in rotorcraft control, as in [22], where a Linear Quadratic Regulator (LQR) state feedback control technique is used to drive a quadrotor onto a 3D path. Depending on the selected states and outputs, only part of the dynamics may be controlled simultaneously, as in [23], where only the attitude and velocity direction of the quadrotor are controlled through an LQR formulation. In contrast, a complete coupled position and attitude reference may be commanded, as in [24], where a Model Predictive Control (MPC) design is used to track some desired trajectory. As an underactuated system, the UAV is not necessarily capable of following any arbitrary 6 DOF trajectory. Control systems that command such profiles need to account for the dynamic feasibility of the trajectory, or prioritize certain coordinates through some cost function. Optimization based control systems such as LQR and MPC lend themselves well to this. These control systems require the complete UAV dynamics to be linearized about some specific equilibrium point. For this reason, they are popular for multirotor platforms operating near the hover condition, as this avoids the need for obtaining several linear approximations, but they are less common for fixed-wing systems which often transition between different regimes. These schemes are also more rigid, as they cannot be easily adapted to use additional or fewer states, for example, pure attitude control.

By far the most popular control architecture for fixed-wing aircraft is a cascaded system, where an inner loop controls the attitude dynamics and an outer loop achieves the task of translating in space. This architecture lends itself better to the conventional fixed-wing input configuration, since attitude can be directly controlled using the three available moments, with limited interdependence with the translational states [25]. For fixed-wing aircraft, the cascaded control idea serves as the basis for the commonly employed “autopilot assumption”, where underlying attitude, and often airspeed, control systems which converge rapidly to the commanded references are assumed to be in place, allowing the designer to develop position control systems using a simple kinematic model of the aircraft, akin to those used for ground vehicles [26]. This structure is not exclusive to fixed-wing vehicles, as it popular in multirotor control systems, which can be considered differentially flat, allowing for simple, dynamically feasible trajectory generation when using a cascaded system [27].

A separate control scheme has additional advantages. First, it allows for more flexibility in the controller design, both for the inner or outer systems. While linear control techniques can be used, nonlinear control systems have been used for both the position and attitude loops [28] or a combination of both [29]. There is also flexibility in the operation of the aircraft itself, since inner loops can be used directly if, for example, a specific attitude maneuver is needed without a specific spatial position profile [30]. Finally, a cascaded control design may consider the underactuated nature of the system, limiting the available references that can be provided at one time to ensure the prescribed trajectory is feasible.

While attitude and position controllers are commonly used together as part of a more comprehensive system, the inner attitude control system imposes most of the limitations on the allowable motions of the UAV. For this reason, it is useful to first study the state-of-the-art of UAV attitude control techniques.

1.3.2.1 Attitude control systems

Attitude control considers the problem of driving the vehicle's attitude towards a prescribed attitude reference, either constant or time varying. The mathematical description of attitude, its parametrizations, controller design have been amply studied for the more general problem of a rigid body, with particular emphasis on spacecraft systems [31, 32]. Each application is different, and some control techniques are more common with certain platforms.

As with dynamic models, Euler angles are the standard for the attitude control of fixed-wing aircraft, both manned and unmanned. The simplest configuration consists of independent proportional-integral-derivative (PID) controllers for the pitch and roll coordinates [25], allowing for longitudinal and lateral control. Indeed, many commercial UAV autopilot systems are based on this control solution [33, 34]; it is simple to implement, can be swiftly adapted to different platforms via gain tuning, and provides good performance for steady flight maneuvers. The main limitation of this scheme is due to the Euler angle parametrization. Attitude cannot be measured directly, it is estimated from different sensor measurements. While Euler angles can be estimated through Extended Kalman Filter (EKF) techniques, modern autopilot systems commonly employ estimation algorithms based on non-singular attitude parametrizations, such as a Multiplicative Extended Kalman Filter (MEKF) on quaternions [35], or a complementary filter [36]. These schemes generate attitude and angular velocity estimates through measurements provided by an Inertial Measurement Unit (IMU) composed of a gyroscope, an accelerometer, and a magnetometer. From the resulting attitude measurement, commonly a quaternion, Euler angles need to be extracted to be used as feedback in the control law, and they become ill-defined at gymbal lock, which occurs as the pitch approaches 90 degrees. This problem is compounded by the kinematic singularity occurring at this attitude, which prevents the Euler angle rates from being resolved from angular velocity measurements.

Angular velocity control, commonly referred to as rate control, is a control solution where angular velocity references are prescribed instead of a time varying attitude configuration. Rate control can be achieved solely with angular velocity feedback, thus avoiding the problems associated with the attitude parametrization [37]. Rough angular velocity estimates can be obtained with only a gyroscope (although most modern estimation systems use a magnetometer and an accelerometer to improve this estimate), and for this reason, this mode of operation is referred to as “gyro” in the RC community. Rate control is sufficient to drive the UAV position onto a reference trajectory [38], but it is often used as a first step towards a precise attitude control system. This invariably requires attitude feedback which reintroduces any flaws in the parametrization.

The agile fixed-wing UAV is expected to perform large angle maneuvers, which can pass through the gymbal lock configuration. Of particular concern is the fact that one of the key maneuvers the platform is meant to perform, hovering, occurs exactly at gymbal lock. One solution is to define two sets of Euler angles, one for steady “horizontal” flight, and one for hovering “vertical” flight, as in [39], where a dual Euler angle controller is designed for a tailsitter UAV. Deciding when to transition between attitude descriptions can be problematic, since the attitude and angular velocity measurements will be discontinuous. A better solution is choosing an attitude description with no singularities, such as the unit quaternion. Many modern quaternion-based attitude control systems for highly maneuverable UAVs can trace their origins back to the satellite attitude control laws developed in [31, 40]. In [41, 42], nonlinear Proportional-Derivative (PD) control laws on quaternions are studied for the regulation of a quadrotor platform, showing the effectiveness of the parametrization in driving the vehicle to any arbitrary attitude reference. The problem of tracking, for time varying attitude references, has also been studied with quaternions as in [43], where a passivity based filter is additionally developed to avoid using angular velocity measurements. While more popular with rotorcraft, quaternions have been successfully used in hover-capable fixed-wing aircraft, such as tailsitters [44] and agile fixed-wing UAVs themselves [45, 46].

The quaternion representation, while non-singular, is non-unique and as a result suffers from the unwinding phenomenon, where the vehicle may take the longest path to reach a reference attitude. This problem can be resolved with a discontinuous control law, but care must be taken to avoid chattering due to sensor noise [47]. There is much discussion on the true impact of this correction in implementation. However, it is true that using a different representation of attitude can avoid this problem altogether. This is the main reason recent UAV attitude control systems have opted instead for working directly on SO(3). In [30], the attitude tracking control of a quadrotor platform is studied. The proposed control law consist of feed-foward components and a feedback PD control law using a nonlinear attitude error function designed for SO(3). This work serves as a specialized application of the geometric control techniques developed in [48]. A modified SO(3) error function is used in [49] for the problem of a tailsitter vehicle. This work emphasizes how a geometric formulation allows a single controller to handle attitude control in all flight regimes: steady flight, hover, and transitions in between.

The output of most of these attitude control systems are control moments, which need to be mapped onto the available inputs. This mapping is the control allocation problem and it is an added complexity in agile fixed-wing control. Control allocation is of little concern for multirotor vehicles, since the control moments are generated through thrust commands. While the propeller generated thrust is affected by airspeed, this and other aerodynamic effects are negligible for the slower moving rotorcraft. However, agile fixed-wing vehicles generate some, or all of their control moments through aerodynamics, since they depend on control surface deflection. Their control authority is greatly dependent on airflow over the control surfaces, so the control allocation problem is dependent on the translational states. In [49], extensive wind-tunnel testing is performed to derive nonlinear mappings from control surface deflection to moments. These functions are dependent on motor revolutions per minute (RPM), forward speed, and UAV acceleration. The aerobatic control system in [46] also creates a nonlinear mapping, but uses a much simpler function based on blade-element momentum theory to obtain an estimate of slipstream. Experimental results show a simplified model using no airspeed measurements can still achieve good performance. A different approach is explored in [50], where different PID controllers are developed for different airspeed conditions, and a gain scheduling system is used to switch between them depending on airspeed measurements.

1.3.2.2 Position control systems

The traditional control architecture for fixed-wing UAVs consists of nested controllers built on top of the Euler-angle based attitude control system. Thrust and pitch are used to control the longitudinal modes of the aircraft, while roll is used to control the lateral motion [2]. These controllers form the basis of the UAV autopilot system, which are used to steer the UAV through a guidance algorithm developed about a kinematic model of the aircraft, both linear [51] or nonlinear in nature [29, 52]. The tasks of guidance and control are deeply intertwined in fixed-wing aircraft, since platform limitations such as minimum airspeed and maximum angles of attack to avoid stall and limited turning capabilities dependent on maximum bank angles do not allow for any arbitrary trajectory to be prescribed.

Much like rotorcraft, agile fixed-wing UAVs are not bound by these constraints, and conventional control techniques are not sufficient to fully harness their capabilities. The control paradigm of rotorcraft UAVs is markedly different from that of conventional fixed-wing platforms in that it focuses on the stricter problem of trajectory tracking. As detailed in [53, 54], the problem of maneuvering in 3D space can be split between two tasks: the geometric task, which requires the vehicle to converge to and remain within some geometric path, and the dynamic task which additionally requires the vehicle to satisfy some time-parametrized behaviour. It is well understood that platform characteristics impose performance limitations for the stricter dynamic task [55]. This is particularly relevant for conventional fixed-wing aircraft, and explains why guidance algorithms, such as the popular L1 guidance [56], are often used, as they focus on the geometric task through implicit path-following techniques. Since the agile fixed-wing UAV is capable of both conventional steady flight and the more maneuverable regime of rotorcraft, it is useful to review control strategies for both platforms.

Out of the additional capabilities of agile fixed-wing UAV, the hover flight regime has been of special interest to the research community because it enables vertical take-off and landings (VTOL), stopping in place, and slow, precise maneuvering. Initial developments for fixed-wing platforms transitioning into a regime involving large angles of attack were performed in [57, 58], where the indoor perching problem of a small glider vehicle was studied. An approximate optimal feedback control is designed for this task, where only the longitudinal modes of the UAV and a constant heading are considered when generating the reference trajectory . In [45], the transition into hover regime of an agile platform is demonstrated through pure attitude control based on a quaternion parametrization, with thrust kept constant to counter the aircraft's weight. No position feedback is considered, causing the platform to drift in the presence of uncertain thruster dynamics or external disturbances. Evidently, a controlled hover requires position feedback and an additional position control loop. Since the hover flight regime is comparable to the conventional operation of rotorcraft, control design for this condition benefits greatly from the available rotorcraft literature. In [59], a position control system is designed for a VTOL vehicle. The system is a cascaded architecture built about an inner nonlinear, quaternion-based attitude control system. The outer position control system generates a coordinated attitude reference and thrust command, through an intermediate position control law. The reference attitude is extracted to also minimize the total angle of rotation. The ideas from this work are modified and adapted in [60] to operate directly in SO(3), avoiding complications arising from the quaternion parametrization. The flexibility of completing a reference attitude is explored in [61, 30], where the SO(3) reference is prescribed not to minimize the rotation, but to satisfy an additional degree of freedom: the direction of the quadrotor forwards pointing vector, which is particularly useful when a camera is affixed to the vehicle. Hover-capable fixed wing platforms have adapted these thrust pointing algorithms to achieve controlled flight while in hover, as in [62, 63], where tailsitter platforms are considered.

When compared to rotorcraft, an agile fixed-wing UAV is more versatile since it operates in the more efficient steady flight regime. Unfortunately, this versatility adds to the complexity in control system development, since a control system developed for the hover condition might not work in other regimes. It is not uncommon for different control strategies to be used for different flight regimes. In [64], one of the first control systems for tailsitters, separate LQR controllers are developed about the two fundamental flight regimes and an additional switching algorithm is implemented to smooth the transitions between them. Transition between regimes is the main challenge in these hybrid control schemes, and several strategies have been proposed. The work in [65] focuses on designing empirical transitional trajectories and prescribes a complete 6 DOF reference during them. A model-based approach is used in [66], where attitude trajectories are calculated offline through an optimization framework that uses the vehicle dynamics as constraints. Only attitude control is considered, with the position control in both regimes left as an open problem. The alternative is designing unified control schemes that avoid the need for specialized transitional systems, or nonlinear control systems whose region of validity overlaps, allowing for a safe transition. The work on tailsitters presented in [62] focuses on extending the validity of the hover regime controller to be also used for steady flight and transitions. The proposed system requires a nonlinear, sequential convex programming optimization problem to be solved online in order to jointly determine attitude references and thrust. This optimization problem requires precise knowledge of aircraft aerodynamics in all flight regimes. In addition to the inherent uncertainty in the aerodynamic model, the computational costs are problematic, and prevent the system from being completely tested in flight experiments. A different strategy is employed in [63], where the outer position control system generates attitude references from precomputed lookup tables. While less maneuvers are possible, no heavy optimization procedure is performed online, which allows the system to be tested experimentally. The work in [67] was specifically developed for agile fixed-wing UAVs, considering the possibility of aerobatic maneuvers. The unified control strategy uses a cascaded architecture with a quaternion-based attitude control core modified from [45]. The control system requires a complete 6 DOF reference, which allows for aerobatic maneuvers to be performed while following a position trajectory. This system was verified together with a motion-planning system based on a rapidly-exploring random (RRT) algorithmm where a library of optimized trajectories was used to ensure dynamic feasibility [68].

1.3.2.3 UAV path-following

Agile fixed-wing UAVs are lightweight platforms and are susceptible to wind disturbances when in outdoor operation. For high velocity maneuvers, the presence of wind causes nominally feasible, time-parametrized trajectories to be beyond the operating limits of the aircraft. Under these circumstances, the objective of path-following can be used to achieve some geometric objective, even if a specific time-dependent objective cannot be met. Indeed, path-following has been proven to improve performance under windy conditions [69, 70]. Conventional path-following schemes for fixed-wing aircraft are based around the kinematic model assumption and are thus limited to circles and straight lines, trajectories defined by the popular Dubins path model [71]. However, more comprehensive techniques exist that consider the full aircraft dynamics and allow for a wider range of 3D motion, both for fixed-wing UAVs and rotorocraft.

In [52], a backstepping control strategy is used to combine the L1 guidance algorithm with a quaternion based attitude control system. While the selected L1 guidance algorithm restricts the possible spatial curves the UAV can navigate, the underlying nonlinear design raises the possibility of more general 3D paths. Indeed, a more general path-following system is pursued in [72], where a specialized reference frame, the Frenet-Serret frame, is assigned to the spatial curve to derive path dependent errors. Simulation results on an agile fixed-wing model show the effectiveness of the system in driving the UAV onto the path while in the presence of wind. However, the underlying LQR controller limits the available motions of the aircraft. Furthermore, the Frenet-Serret frame limits the possible paths that can be prescribed. Originally used for 2D motion control, the flaws of the Frenet-Serret frame in 3D applications are well known [73], and alternatives have been developed [74]. The alternative parallel transport frame is used in [38] to develop a geometric path-following solution directly on SO(3). While only simple paths are shown for the experimental results, the combination of the selected path descriptor and the non-singular control formulation in theory allow for a more general class of 3D paths the UAV can navigate.

1.3.3 Pilot-assist systems

The research discussed above is but a sample of the plentiful work that has been done in the field of autonomous UAV control. In contrast, works on semi-autonomous control systems, where the presence of a remote human operator is considered, are relatively scarce.

Conventional manual remote operation of fixed-wing UAVs is straightforward: an experienced pilot will command the thrust and control surfaces directly with no underlying stabilizing controller on board. This mode of operation places all the control burden on the pilot, from the bare minimum objective of stabilization, to more complex tasks like navigation and disturbance rejection. While standard fixed-wing aircraft aim to have a stable configuration, agile UAVs are marginally stable by design to enable aggressive maneuvering [75], demanding more skill from the operator for the task of stabilization alone. Compounding the difficulties in nominal operation, pilot feedback is limited to direct visual line of sight (LOS) with the aircraft or, for longer distances, cameras installed on the frame providing a first person view (FPV) [76]. While a remote pilot may be able to approximately determine the aircraft's pose, additional feedback available to pilots in manned aircraft, which might enable the operator to estimate disturbance magnitude and direction, are lost. The role of the vestibular system as a means for pilots to estimate the aircraft's accelerations has for many years been recognized as an important aspect of manned flight [77]. A key aspect of remotely controlled aircraft is that the operator can only recognize and compensate for disturbances after they have visibly altered the expected flight path or intended maneuver. To assist the pilot, research in this area has focused on pilot feedback enhancement, as in [78] or through warning system [79], but solutions through control architectures are limited.

Pilot-assisted flight through semi-autonomous control systems is best exemplified by the stability augmentation systems used for conventional aircraft, such as the one presented in [80] where linear control techniques are used to stabilize the longitudinal and lateral modes of the fixed-wing dynamics linearized about the steady flight condition. These linear nested controllers are the basis for open source autopilots developed for smaller, unmanned vehicles [34, 81]. The latter is especially notable, since it allows for pilot-assisted flight for tailsitters in both steady and hover regimes. This is done through two different controllers which switch as the UAV pitches up or down. The addition of altitude control is presented in [82], where an $\mathcal{H}_\infty$ controller is used to provide robustness against external disturbances.

While these semi-autonomous control systems can be used to control an agile fixed-wing platform, they do not exploit the maneuverability of the aircraft. Research on aggressive, pilot-assisted maneuvers is more prevalent in manned aircraft, as in [83] which studies the enhancement of aerobatic longitudinal maneuvers for fighter aircraft through a control augmentation system composed of nested PID control on pitch and airspeed. These schemes, however, have not been adapted for smaller UAVs.

1.4 Thesis organization

The thesis is organized as follows. Preliminary concepts are discussed in Chapter 2, focusing first on the dynamic model of the agile fixed-wing UAV which will serve as the basis for controller design and analysis. The overall architecture of the proposed system is then briefly discussed in order to introduce its main components and their interaction. These components will be the focus of later chapters, as shown by the component distribution shown in Figure 1.3. Lastly, the simulation and experimental tools used to verify the proposed strategies are presented in this chapter. The core verification tool is the specialized UAV dynamic model developed by previous members of AML. As part of this work, this model has been integrated into the Software in the Loop (SITL) environment, which serves as an intermediate step before experimental flights. This section also introduces the experimental platform used for experimental trials.

The attitude control system is the focus of Chapter 3. Different nonlinear attitude control strategies are developed, which are then evaluated through numerical simulations and experimental flight tests. Results are presented in this chapter as this component is meant to be able to operate on its own when required, and because the correct operation of the complete UAV control system hinges on the attitude control core.

Chapter 4 presents the position control system, built around a nonlinear vector-projection algorithm which generates attitude references and thrust commands given some a virtual acceleration command. This command can be generated through different linear or nonlinear control strategies. In particular, two operating modes are presented, position tracking

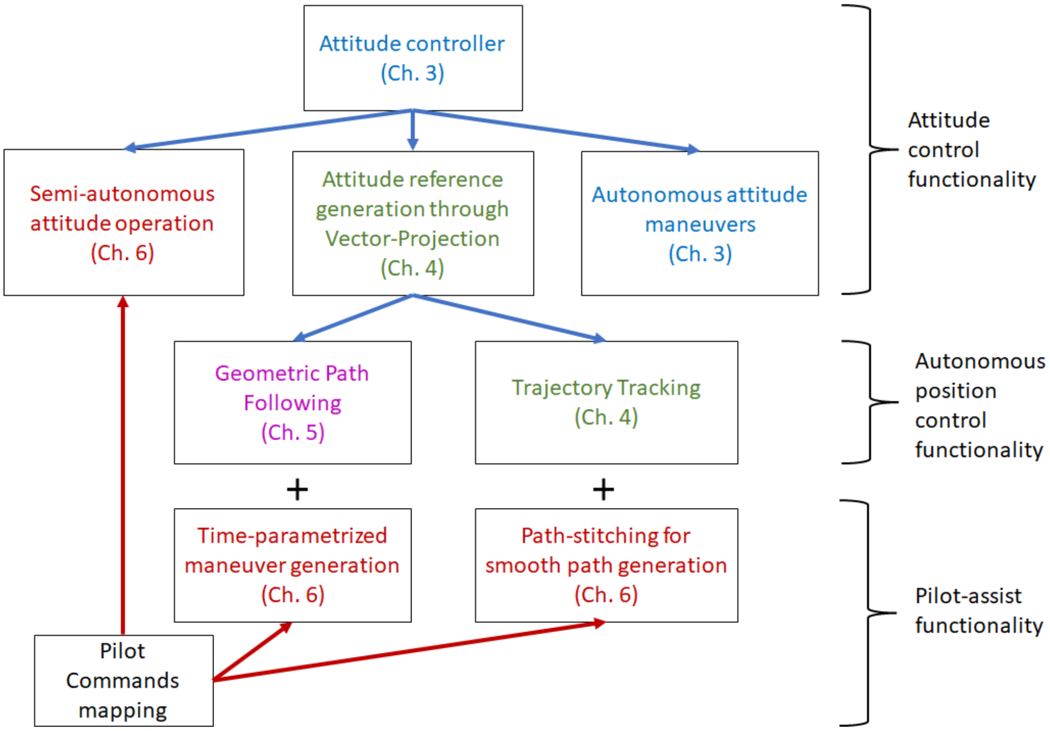

Attitude controller
(Ch. 3)
Semi-autonomous attitude operation
(Ch. 6)
Attitude reference generation through Vector-Projection
(Ch. 4)
Autonomous attitude maneuvers
(Ch. 3)
Attitude control functionality
Geometric Path Following
(Ch. 5)
Trajectory Tracking
(Ch. 4)
Autonomous position control functionality
+
+
Time-parametrized maneuver generation
(Ch. 6)
Path-stitching for smooth path generation
(Ch. 6)
Pilot-assist functionality
Pilot Commands mapping

Figure 1.3: Chapter distribution of developed systems.

control, and velocity tracking control, which can be selected as required. To verify the capabilities of the control system in autonomous operation, representative time-parametrized trajectories are designed, which are then given as references for the UAV to track in simulations and flight experiments.

While the problem of tracking is the focus of the previous chapter, the geometric task is explored in Chapter 5 as an alternative to UAV control. This chapter discusses the representation of reference paths as 3D curves in space, and introduces the necessary differential geometry tools needed to design a path-following solution. The main tool is the development of a novel path frame for agile navigation in 3D which is used to define path-relative errors. A closed-loop reference generation scheme, together with a path evolution policy is then presented to adapt the tracking systems into a path-following solution. As with previous chapters, simulation and experimental results are presented.

The work on the chapters outlined above focuses on the autonomous operation of the agile fixed-wing UAV. Chapter 6 presents the additional components needed to augment these results to achieve a semi-autonomous pilot-assist solution. These results span all three control layers, starting with semi-autonomous attitude control, best suited to achieve specific aerobatic maneuvers or for human operation where position feedback is unreliable. Tracking control is shown to be viable for all operating regimes of the aircraft but its use is emphasized for slow maneuvers requiring high precision, both in steady flight and hover flight. A reference generator using pilot commands as input is developed for different flight conditions. In contrast, the path-following solution is exploited for high velocity, semi-autonomous operation, where the notion of spatial geometry is more intuitive to a human operator than instantaneous position references. A path-stitching algorithm that translates pilot commands into curves in space is then developed for its integration with the path-following controller. Finally, an integration of these systems into a comprehensive semi-autonomous solution is presented, where the control systems changes between tracking and path-following operation smoothly depending on the pilot prescribed velocity reference. Complex experimental flights are then presented to conclude the chapter.

Finally, Chapter 7 presents concluding remarks and a discussion of future research opportunities arising from the results of this work.

Chapter 2

Preliminaries

2.1 Agile fixed-wing UAV model

To evaluate the stability and performance of developed control solutions, a dynamic model of the system of interest is required. There exist several models and simulators of fixed-wing aircraft focused mostly on large, manned platforms, ranging from simplified kinematic models, to dynamic models that do account for aerodynamic forces and moments. In recent years, there has been a greater focus on smaller, unmanned aerial vehicles but most models are still based around the concept of stability derivatives, and are not capable of describing the complete three dimensional motion of agile fixed-wing UAVs [2].

To better capture the behaviour of the agile UAV in all flight regimes, this work uses the agile fixed-wing UAV model developed in [84], but modified such that the attitude propagation occurs directly in the special orthogonal group instead of the proposed quaternion model. This model consists of four main components: a standard rigid body dynamic model, an aerodynamics model based on a component breakdown approach, a thruster model, and a slipstream model. The rigid body model is presented first, together with the necessary concepts for attitude control analysis and design. The other three components, which compose the high-fidelity model, are then briefly discussed.

2.1.1 Rigid body dynamics

To describe the three-dimensional motion of the aircraft, two reference frames are required. As is common in UAVs, the North-East-Down (NED) convention is adopted such that the $\hat{\mathbf{i}}_1$, $\hat{\mathbf{i}}_2$, $\hat{\mathbf{i}}_3$ basis vectors of the inertial reference frame, $\boldsymbol{\mathcal{F}}_i$, align with the North, East, and Down directions, respectively. Similarly, the body reference frame, $\boldsymbol{\mathcal{F}}_b$ is affixed to aircraft centre of mass, and is selected such that the $\hat{\mathbf{b}}_1$ basis vector points out of the aircraft nose, the $\hat{\mathbf{b}}_2$ vector points out of the starboard wing, and the $\hat{\mathbf{b}}_3$ vector points out of the aircraft belly. These reference frames are shown in Figure 2.1.

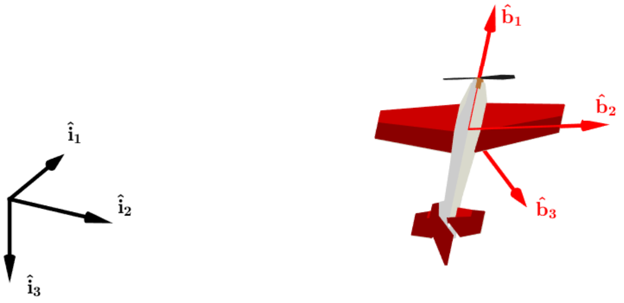


Figure 2.1: Inertial and body reference frames.

With these two frames, the dynamics of a rigid body can be properly stated. They are described by two sets of differential equations which represent, respectively, the rotational and translational motion of the aircraft.

2.1.1.1 Rotational dynamics

The rotational dynamics of a rigid body are given by [10]

$$\dot{\mathbf{C}}_{bi} = -\boldsymbol{\omega}_b^{\times}\mathbf{C}_{bi}, \tag{2.1}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}}_b = (\mathbf{J}\boldsymbol{\omega}_b)^{\times}\boldsymbol{\omega}_b + \mathbf{M}, \tag{2.2}$$

where $\mathbf{C}_{bi}$ is the Direction Cosine Matrix (DCM) from body to inertial frame, a member of the three-dimensional special orthonormal group given by the set

$$\mathrm{SO}(3) = \left\{\mathbf{C} \in \mathbb{R}^{3\times 3} \,|\, \mathbf{C}^T\mathbf{C} = \mathbf{I}_3,\, \det(\mathbf{C}) = 1\right\}, \tag{2.3}$$

where $\mathbf{I}_n \in \mathbb{R}^{n\times n}$ refers to the n-dimensional identity matrix. The term $\boldsymbol{\omega}_b = \boldsymbol{\omega}_b^{bi}$ denotes the angular velocity between the body and inertial frames, resolved in body frame coordinates. The elements of $\boldsymbol{\omega}_b$ are commonly defined as $\boldsymbol{\omega_b} = [p,\, q,\, r]^T$. The matrix $\mathbf{J} \in \mathbb{R}^{3\times 3}$ is the second moment of mass about the aircraft's centre of mass, resolved in the inertial frame, considered constant, and, by necessity, symmetric and positive definite. Finally $\mathbf{M}$ are the net moments acting on the aircraft, mainly due to the effects of the propeller, and aerodynamics. The skew symmetric operator $(\cdot)^{\times} : \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ maps an element of $\mathbb{R}^3$ onto the set of three-dimensional skew symmetric matrices, $\mathfrak{so}(3) = \left\{\mathbf{S} \in \mathbb{R}^{3\times 3} \,|\, \mathbf{S}^T + \mathbf{S} = \mathbf{0}\right\}$, it is defined, for any

$\mathbf{u} \in \mathbb{R}^3$, as

$$\mathbf{u}^\times = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}. \tag{2.4}$$

Equation (2.1) is known as Poisson's equation, and represents the attitude kinematics. It can also be written in terms of the DCM from body to inertial frame, $\mathbf{C}_{ib} = \mathbf{C}_{bi}^T$.

Attitude evolves naturally on SO(3): it is the only representation both unique and without singularities. Other attitude parametrizations suffer from some deficiency, but other properties make their use appealing for control and estimation solutions. While this work focuses on control solutions directly on SO(3), notions of two other parametrizations will be useful for analysis. First is the angle-axis parametrization, the direct consequence of Euler's rotation theorem, and related through Rodrigues's rotation formula,

$$\mathbf{C}(\eta, \hat{\boldsymbol{\epsilon}}) = \cos(\eta)\mathbf{I}_3 + (1 - \cos(\eta))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T - \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times, \tag{2.5}$$

where η is the rotation angle around the unit vector $\hat{\boldsymbol{\epsilon}}$. This parametrization, much like the quaternion, possesses no singularities but is non unique, since $\mathbf{C}(\eta, \hat{\boldsymbol{\epsilon}}) = \mathbf{C}(-\eta, -\hat{\boldsymbol{\epsilon}})$. The control system is not developed using this parametrization but it will useful to verify several properties of the control system. In addition, the angle η is a good measure of total deviation, since it condenses the total angular difference between two reference frames. This angle can be easily calculated through the relation

$$\mathrm{tr}(\mathbf{C}) = 1 + 2\cos(\eta), \tag{2.6}$$

where $\mathrm{tr}(\cdot)$ is the standard trace operator defined for square matrices.

The second parametrization is the 3-2-1 Euler angle sequence which is often used in control solutions for fixed-wing aircraft. This parametrization consists of the sequential rotations around the three primary body axis. A DCM can be constructed with no ambiguity from these three angles as

$$\begin{aligned} \mathbf{C}_{bi} &= \mathbf{C}(\phi, \hat{\mathbf{k}}_1)\mathbf{C}(\theta, \hat{\mathbf{k}}_2)\mathbf{C}(\psi, \hat{\mathbf{k}}_3) \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{aligned} \tag{2.7}$$

where $\hat{\mathbf{k}}_i$ is the unit vector with zeros in all but the i^{th} component, which is equal to one, and ϕ, θ, ψ are the roll, pitch, and yaw angles, respectively. This parametrization is ill suited for the design of control systems for agile platforms, since it is susceptible to gimbal lock and a kinematic singularity as the pitch angle approaches 90 degrees. In this work, Euler angles will only be used to build, when appropriate, a reference attitude profile, and to show results in a more intuitive way, when applicable. To enable the latter, the extraction algorithm presented in [12] will be used, as it is common in several software packages.

The second rotational equation, (2.2), is known as Euler's equation and it represents the attitude dynamics proper. With few exceptions, it is always resolved in the body frame, for two main reasons: the second moment of mass resolved in the body frame is constant, and the angular velocity is always measured, or estimated, directly in body frame coordinates.

2.1.1.2 Translational dynamics

In contrast with the rotational dynamic equation, there is no preference on the reference frame in which the translational equations are presented in the literature. In this work, the following equations are used,

$$\dot{\mathbf{p}}_i = \mathbf{C}_{bi}^T \mathbf{v}_b, \tag{2.8}$$

$$m\dot{\mathbf{v}}_b, = -m\,\boldsymbol{\omega}_b^{\times}\,\mathbf{v}_b + mg\,\mathbf{C}_{bi}\hat{\mathbf{k}}_3 + \mathrm{T}\,\hat{\mathbf{k}}_1 + \mathbf{F}_a, \tag{2.9}$$

where $\mathbf{p}_i$ is the position of the aircraft centre of mass resolved in the inertial frame, or in NED coordinates, $\mathbf{v}_b$ is the aircraft velocity, resolved in body frame coordinates, m is the aircraft mass, considered constant, g is the gravity constant, T is the aircraft thrust force, and $\mathbf{F}_a$ are the aerodynamic forces acting on the aircraft. With the unit vectors $\hat{\mathbf{k}}_i$ as previously defined.

The selection of $\mathbf{p}_i$ is due to sensor measurements, which give the position in a fixed inertial frame. The choice of (2.9) is common in fixed-wing control, since aerodynamic forces are more easily resolved and applied in the body frame. The three elements of $\mathbf{v}_b$ are commonly defined as $\mathbf{v}_b = [u,\, v,\, w]^T$.

2.1.2 High fidelity model

The methodology and model developed in [84] aims to represent through first principles the dynamics of small agile UAVs for their complete flight envelope. It remains one of the most detailed models for this category of aircraft and is capable of capturing behaviour for a ±180 degree range of angle of attack and sideslip.

While several elements of this model can be generalized to represent different platforms, this work uses for the purposes of simulation the specific model matching that of the experimental platform, the *McFoamy* aircraft by West Michigan Park Flyers, shown in Figure 2.2.

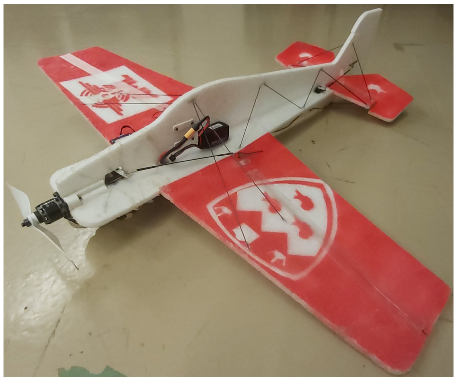


Figure 2.2: McFoamy platform.

Aerodynamic model

The aerodynamics model employs a component breakdown approach which first splits the UAV's main components into segments, and then proceeds to calculate the aerodynamics forces and moments individually before finally adding together all the contributions [18]. The main aircraft components are the wing, the horizontal tail, the rudder, the horizontal fuselage, and the vertical fuselage. These components can in turn be grouped into horizontal and vertical surfaces. This segmentation is illustrated in Figure 2.3.

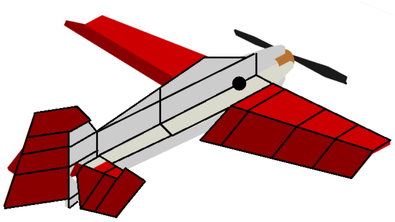


Figure 2.3: Component breakdown approach.

Each segment is capable of producing aerodynamic forces of lift and drag, $\mathbf{F}_{seg}$ and an aerodynamic moment in a specific direction, contained in $\mathbf{M}_{seg}$. Horizontal surfaces produce forces in the direction of the $\hat{\mathbf{b}}_3$ and $\hat{\mathbf{b}}_1$ basis vectors, and a (pitching) moment about the $\hat{\mathbf{b}}_2$ vector. Vertical surfaces can generate forces in the $\hat{\mathbf{b}}_2$ and $\hat{\mathbf{b}}_1$ directions, and a moment about the $\hat{\mathbf{b}}_3$ vector. The aerodynamic forces and moments are all a function of the incoming airflow, concretely, the velocity at the aerodynamic center of each segment, calculated by

$$\mathbf{v}_{seg} = \mathbf{v}_b - \mathbf{v}_w^b + \boldsymbol{\omega}_b^\times \mathbf{r}_{seg} + \mathbf{v}_{s,seg} \tag{2.10}$$

where $\mathbf{r}_{seg}$ is the vector from the aircraft's centre of mass to the segment's aerodynamic center, resolved in body frame coordinates, $\mathbf{v}_w^b$ is the wind velocity resolved in the body frame, and $\mathbf{v}_{s,seg} = [V_s, 0, 0]^T$ is the slipstream velocity at the segment's aerodynamic centre, which only has an axial component. The total aerodynamic forces and moments are then

$$\mathbf{F}_{aero} = \sum \mathbf{F}_{seg}, \tag{2.11}$$

$$\mathbf{M}_{aero} = \sum \left(\mathbf{M}_{seg} + \mathbf{r}_{seg}^\times \mathbf{F}_{seg}\right). \tag{2.12}$$

2.1.2.1 Thruster model

Conventional thruster models only consider axial flow conditions, as this is sufficient for most steady, level flight maneuvers. The model developed in [20] uses propeller geometry together with incoming airflow and propeller rotational speed to predict the forces and moments in the direction of the three body axes. The thuster model is based around blade element momentum theory and accounts for all flow conditions: static, axial, oblique, and reverse flow. In addition, the thruster gyroscopic effects are calculated and added to the thuster moments.

2.1.2.2 Slipstream model

Slipstream, the additional airflow induced over the aircraft by the propeller, is a fundamental characteristic of agile fixed-wing UAVs. This airflow not only adds to the effectiveness of the control surfaces, but allows the aircraft to retain control authority at low airspeed. This property allows the aircraft to remain controllable during hover or near hover maneuvers. Agile fixed-wing are designed so that a large part of the control surfaces are immersed in propeller slipstream, as shown in Figure 2.4.

The slipstream model developed in [19] accounts for both the the acceleration and diffusion phenomena, and it is shown to accurately predict the axial component up to several propeller diameters downstream of the propeller. The swirl component is used to adjust the thruster reaction moment, as it is shown to induce a rolling moment on the aircraft.

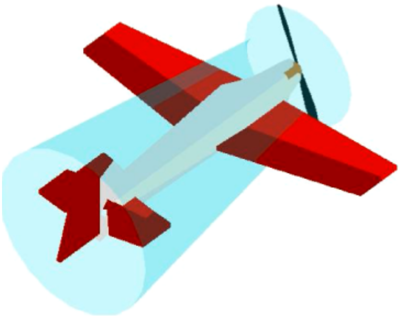


Figure 2.4: Propeller slipstream.

2.1.3 Dynamic model for control development

The high-fidelity model is an important tool for the development and implementation of control architectures since it allows the designer to test the control design on a more comprehensive, realistic system. However, due to its structure and complexity, designing a controller based on this model is a difficult task. It is a common practice, specially so for fixed-wing aircraft, to employ a simplified model for control development, and a more comprehensive one for simulation and verification [2, 85]. For this work, a simple model based on [2, 86] is used. Define the relative velocity with respect to the surrounding air as

$$\mathbf{v}_a = \mathbf{v}_b - \mathbf{v}_w^b = \begin{bmatrix} u_a & v_a & w_a \end{bmatrix}^T . \tag{2.13}$$

Then the airspeed V_a, the angle of attack α and the sideslip angle β are defined as

$$V_a = \|\mathbf{v}_a, \| \tag{2.14}$$

$$\alpha = \operatorname{atan2}(w_a/u_a), \tag{2.15}$$

$$\beta = \operatorname{atan2}(v_a/u_a), \tag{2.16}$$

where, unless specified otherwise, $\|\mathbf{x}\| = \|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^\mathbf{T}\mathbf{x}}$ is the Euclidean norm, defined for any $\mathbf{x} \in \mathbb{R}^n$.

While every component of the UAV generates aerodynamic forces and moments, and indeed this is how the high fidelity model functions, for the simplified model the main interest is in regards to the largest lifting surface, the main wing, and the three control surfaces, the ailerons, elevator, and rudder.

2.1.3.1 Wing Aerodynamics

For the wing, we consider the forces of lift and drag, as well as a pitching moment, which at pre-stall angles of attack can be approximated as

$$L = \frac{1}{2}\rho\, b\, \bar{c}\, V_a^2 C_L, \tag{2.17}$$

$$D = \frac{1}{2}\rho\, b\, \bar{c}\, V_a^2 C_D, \tag{2.18}$$

$$M = \frac{1}{2}\rho\, b\, \bar{c}^2\, V_a^2 C_M, \tag{2.19}$$

where ρ is the air density, b is the wing span, $\bar{c}$ is the mean aerodynamic chord of the wing, and C_L, C_D, and C_M are the dimensionless aerodynamic coefficients of lift, drag, and pitching moment, respectively. These can in turn be approximated as

$$C_L = C_{L\alpha}(\alpha - \alpha_0), \tag{2.20}$$

$$C_D = C_{d0} + \frac{C_L^2}{\pi k_o \mathrm{AR}}, \tag{2.21}$$

$$C_M = -\text{const.}, \tag{2.22}$$

where C_{d0} is the drag coefficient due to skin friction, k_o is Oswald's efficiency factor, $\mathrm{AR} = b^2/S$ is the wing aspect ratio, with S as the surface area, α_0 is the geometric zero-lift angle, equal to zero for symmetric airfoils, the constant value of C_M can be considered zero for thin flat plates since the aerodynamic center and center of pressure are coincident, and, finally, $C_{L\alpha}$ is the lift-curve slope for a finite surface, approximated through a correction of the 2D lift-curve slope to account for the small aspect ratio of the wing as

$$C_{L\alpha} = C_{l\alpha}\frac{\mathrm{AR}}{\mathrm{AR} + 2(\mathrm{AR}+4)/(\mathrm{AR}+2)} \tag{2.23}$$

where $C_{l\alpha} \approx 2\pi$. These equations are a common approximation for steady level flight, which is where these forces are most significant. Being proportional to the square of the velocity, the aerodynamic forces are much diminished and considered negligible compared to the thruster force in the near hover regime, which coincides with high angle of attack.

Finally, the forces in 2.9 are the aerodynamic forces resolved in the body frame through:

$$\mathbf{F}_a = \begin{bmatrix} \cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix} = \mathbf{C}(\alpha, \hat{\mathbf{k}}_2) \begin{bmatrix} -D \\ 0 \\ -L \end{bmatrix} \tag{2.24}$$

2.1.3.2 Control surfaces

Agile fixed-wing UAVs, have large control surfaces capable of large deflections. As such, the control mostly contribute moments around the body principal axes. By design, these surfaces are partially or completely immersed in propeller slisptream which gives them additional control authority. As such, it is important to account for this added airflow by defining the effective airspeed over the control surfaces as

$$V_\delta = \|\mathbf{v}_b - \mathbf{v}_w^b + \mathbf{v}_s\| \tag{2.25}$$

where $\mathbf{v}_s = [v_s \ 0 \ 0]^T$ is the slipstream velocity.

Based on [86], in the previous work of [67] the moments generated by the control surface deflections are calculated as

$$\mathbf{M}_\delta = \frac{1}{2} V_\delta^2 S \begin{bmatrix} b\, C_{l\delta_a} & 0 & 0 \\ 0 & \bar{c}\, C_{m\delta_e} & 0 \\ 0 & 0 & b\, C_{n\delta_r} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_e \\ \delta_r \end{bmatrix} \triangleq \mathbf{G}(V_\delta)\boldsymbol{\delta}, \tag{2.26}$$

where $C_{l\delta_a}$, $C_{m\delta_e}$, and $C_{n\delta_r}$ are the constant control derivative coefficients, determined through experiments in [84], $\boldsymbol{\delta} = [\delta_a \ \delta_e \ \delta_r]^T$ are the control deflections of aileron, elevator, and rudder, respectively. In practice, the effective airspeed over the control surfaces, V_δ, is approximated through $\hat{V_\delta}$, calculated using momentum theory in the absence of wind as

$$\hat{V_\delta} = \sqrt{u^2 + \frac{2\,T}{\rho \pi R^2}}, \tag{2.27}$$

where R_s is the radius of the slipstream far downstream.

Some assumptions are implied through this equations. First, each control surface is directly mapped to a single body moment through the control allocation matrix $\mathbf{G}$, meaning there exists no secondary coupling. While the longitudinal mode associated with the elevator is often considered decoupled, the lateral modes associated with yaw and roll are often modeled with coupling control derivatives. These are often small compared to the primary ones, and indeed the large control surfaces capable of large deflections of the agile fixed-wing results in negligible coupling coefficients.

Second, the control coefficients are assumed constant because the surfaces are immersed in slipstream which only has a component in the first body axis. Slipstream not only increases the total airflow, it also keeps the effective angle of attack over the control surfaces at low values. Indeed, simulations in [84] lead to the conjecture that the total flow in slipstream changes little with respect to the external flow, as an increase in external flow is balanced by a decreased slipstream. This correlates to the the assumption represented in equation (2.27), which assumes the airflow in the directions of the second and third body axis is negligible compared to the axial flow. The final assumption is fundamental for attitude control design and analysis and is given special emphasis.

Assumption 2.1 (Controllability of attitude dynamics). *The control allocation matrix* $\mathbf{G}(V_\delta)$ *remains full rank for all* $t > 0$*, regardless of the operating regime of the UAV. This renders the attitude dynamics fully controllable for all* $t > 0$*.*

It has been established that the attitude dynamics of a rigid body, as in (2.2), are fully controllable if the input moments, in this case $\mathbf{M}_\delta$, are composed of at least three independent control inputs [21, 32]. The control surface deflections, $\boldsymbol{\delta}$ are composed of three independent inputs, hence, controllability hinges on the control allocation matrix being full rank.

This is a common assumption in fixed-wing UAV control, and usually constrains the airspeed and angle of attack between specific values, which define the flight envelope of the aircraft. For the platform considered in this work, this limitation is relaxed due to the dependence of $\mathbf{G}$ on the slipstream velocity, $\mathbf{v}_s$, itself dependent on the angular rotation of the propeller, ω_p. The elements in the allocation matrix will only vanish if the aircraft is still and the motor is inactive, a condition that will never happen in the operation of the UAV.

Finally, the full rank condition, for some configurations of $\mathbf{G}(V_\delta)$ only necessitates the diagonal elements, or primary control coefficients, to be larger that the secondary coupling terms. Since the design of the studied platform allows each control surface to generate a large moment in its assigned direction, they are so dominant that the allocation matrix can be considered diagonal, so the full rank assumption holds.

2.1.3.3 Propeller

The thrust force and reaction force associated with the propeller and motor combination are calculated according to standard definitions for propeller aerodynamic coefficients [86]:

$$T = C_T \rho D_p^4 n^2$$
$$M_p = C_q \rho D_p^5 n^2$$

where D_p is the propeller diameter, n is the propeller rotational speed in revolutions per second, and C_T and C_q are the force and reaction moment coefficients. For the platform of interest, the reaction torque is found to be small enough in all flight modes that it can be safely neglected. The thust force, however, is an important control input, precision is required in order to determine the commanded motor angular velocity. In [67], the thrust equations is first rewritten as

$$T = C_T \rho D^4 \left(\frac{1}{60}\omega_p\right)^2 \triangleq k_t \omega_p^2, \tag{2.28}$$

where ω_p is the propeller (motor) angular velocity in revolutions per minute (RPM). In the literature, several k_t is often considered constant, particularly in quadrotor control derivations. However, for the fixed-wing platform the coefficient is dependent on the axial flow as well as the propeller angular velocity itself. These two elements are often condensed into the nondimensional advance ratio

$$J = \frac{u}{D_p\, n} = \frac{60\, u}{D_p \omega_p}. \tag{2.29}$$

The relationship between J and k_t needs to be obtained experimentally. Figure 2.5 shows the results of the experiments performed in [87] for the McFoamy platform. A curve fit gives the following function

$$k_t = (-1.439 J^2 2 - 2.212 J + 2.245) \times 10^{-7}. \tag{2.30}$$

The net moments in 2.2 are then

$$\mathbf{M} = M\hat{\mathbf{k}}_2 + M_p \hat{\mathbf{k}}_1 + \mathbf{M}_\delta \triangleq \mathbf{M}_a + \mathbf{M}_p + \mathbf{M}_\delta \tag{2.31}$$

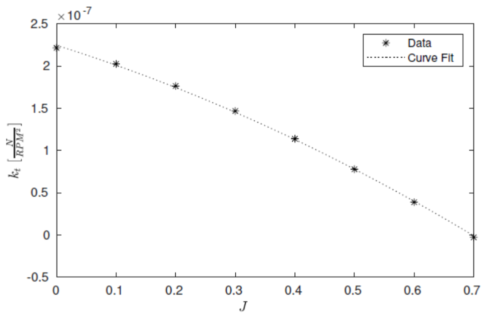


Figure 2.5: Thrust coefficient vs. advance ratio.

2.2 Control system architecture

This work aims to develop a comprehensive control system for the agile fixed-wing UAV platform. This system considers fully autonomous operation as well as pilot-assist, semiautonomous flight. Different operating modes can be used to complete a specific mission, and the control system needs to be flexible enough to enable this functionality while still relying on a unified control theory to avoid switching between disparate control strategies.

The overall control systems assumes that the full pose of the aircraft is available, meaning attitude, angular velocity, position, and velocity. Four independent control inputs are assumed, one body fixed force and three moments. For the specific case of the McFoamy platform this translates to the motor thrust and the three control surfaces of ailerons, elevator and rudder. However, the algorithms developed in this work can be applied to other platforms with different configurations, such as tailsitters and other VTOL capable aircraft.

2.2.1 Cascaded control structure

The control system uses a modular, cascaded structure, where individual control modules can be enabled depending on the desired functionality. A diagram of the control architecture is shown in Figure 2.6. Four main components make up the overall control system, the core attitude controller, an outer position controller, a guidance component for reference generation, and a state machine to manage the different control modes according to pilot input, or mission parameters.

2.2.1.1 Attitude controller

The attitude controller has as its input a reference attitude profile and outputs the required control deflections to drive the UAV attitude to the reference. As expected, this compo-

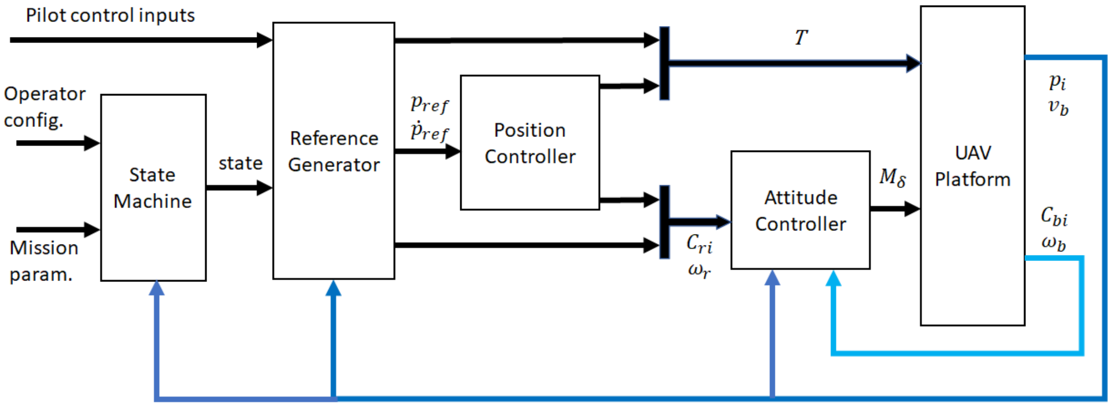


Figure 2.6: Control architecture diagram.

nent takes in measurements of attitude, $\mathbf{C}_{bi}$ and angular velocity, $\boldsymbol{\omega}_b$ but in addition uses translational velocity, $\mathbf{v}_b$ to adjust for changes in the control allocation matrix, $\mathbf{G}$.

The attitude controller is always active, as it is the core control component required for autonomous or semiautonomous flight.

2.2.1.2 Position controller

The position controller is tasked with generating the required attitude reference and thrust command required to drive the UAV position and velocity to the reference position $\mathbf{p}_{ref}$ and $\dot{\mathbf{p}}_{ref}$ and velocity it takes in as inputs. Depending on the nature of the reference, either a time reference trajectory or a geometric path, the position controller enables trajectory tracking and path following functionality, the latter being supported by the maneuver generator.

The position controller is not always active, since pure attitude control, either autonomous or semiautonomous is also considered. In these cases, the thrust and attitude reference is produced by the maneuver generator.

2.2.1.3 Maneuver generator

This component is capable of generating the necessary references for the various control systems to operate. Depending on the system state specified by the state machine, the maneuver generator produces position references, geometric paths, or direct attitude references. Except for very specific maneuvers, position and attitude references are never prescribed at the same time by the maneuver generator; if position position control is enabled, then the attitude reference is internally generated by the position controller. This avoids overparametrizing the reference, since the system only has four independent actuators. Exceptions are reduced attitude commands around the thrust axis, which enable aerobatic maneuvers like inverted flight, knife-edges, and rolling harriers.

Maneuver generation depends on the mode of operation. If autonomous, maneuvers are assembled by joining together 3D geometric trajectories from a library, which includes straight line segments, orbits, helices, etc. These trajectories can be time parametrized, but may also use a path parameter that evolves in closed-loop with the system states, to enable path following functionality. During semiautonomous operation, the reference trajectories are a function of the pilot commands, usually the sticks of a standard four-channel RC transmitter.

2.2.1.4 State machine

The state machine is a simple, if important component needed for implementation. Its main purpose is managing controller status and flags when transitioning between modes of operation. It monitors UAV states and either mission parameters or pilot commands to determine when to switch between tracking and path following functionality. It then initializes trajectories or geometric paths to ensure a smooth transition between both operating modes.

From a controller design perspective, the tasks this component performs are contained independently in other systems, such as the position controller and the trajectory generator. Its existence as a separate module is exclusive to the implementation firmware, as an independent system manager that handles system flags and parameter initialization, as opposed to independent modules, is considered good practice, for a variety of reasons. As such, this component is not explicitly elaborated upon in this work, but the tasks it needs to perform in implementation are indicated where appropriate.

2.2.2 Modes of operation

The proposed agile fixed-wing control system considers two operating schemes, fully autonomous operation, and pilot-assist, semi-autonomous operation. Both of these options can employ a combination of four different control strategies, as needed. These are:

- Attitude control. For autonomous operation, this enables 3D angular maneuvers, such as explicit aerobatic commands. For semi-autonomous operation, this can be used to perform aerobatics on command but is more importantly used as the basis for an attitude stabilization mode. This is an important feature to have, as it can be defaulted to in case of GPS signal loss to safely land the aircraft.

- Position tracking control. Trajectory tracking is the choice for slower, precise maneuvers. As such, it is better suited for hover controls, both autonomous or semi-autonomous, or slower steady maneuvers, such as horizontal landings.

- Velocity control. A modification on position tracking, this control mode disregards the precise position of the UAV and instead prioritizes velocity in the inertial or body frames. At high velocities, this is a more intuitive operation that position tracking, but the lack of position feedback will cause the aircraft to drift in the presence of disturbances.

- Path-following control. Often, precise positioning with respect to time is not necessary, but the UAV motion is still expected to remain contained withing some geometric path. This functionality is achieved through path-following, and can be seen as a relaxed trajectory tracking. For semi-autonomous operation, this functionality can be considered a compromise between position and velocity control: the precise positioning of the aircraft is not controlled, but the velocity reference is not prescribed in open loop, as it is modified to guide the UAV onto a path.

2.3 Simulation and experimental tools

The control strategies developed in this work are tested and verified through a three step process, starting with numerical simulations without regards to controller implementation and culminating with flight tests with an experimental platform. Between these two steps, an intermediate Software in the Loop (SITL) test phase is used.

2.3.1 Numerical simulation

All developed algorithms are first tested in a standard simulation environment using the Simulink package of Matlab. The system consists of the rigid body model and the high fidelity aerodynamic and thruster model, configured with the parameters of the experimental platform, the Park Flyers McFoamy. The various geometric and inertial parameters, as well as the specifications of the propeller and motor are given in Table 2.1, where the aircraft's second moment of mass is given by

$$\mathbf{J} = \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{xz} & 0 & I_z \end{bmatrix}.$$

Attitude propagation is performed directly on SO(3), with a polar decomposition technique for orthogonalization of the DCM [88] to account for numeric errors. As stated previously, full knowledge of the states is assumed. While the high fidelity model drives the simulated system, no knowledge of its internal values is used while testing the control laws, instead, the various equations of simplified model for control development are used when required.

Table 2.1: Aircraft properties

Parameter	Value	Units
mass (m)	0.45	kg
I_x	3.922×10^{-3}	$kg\,m^2$
I_y	1.594×10^{-2}	$kg\,m^2$
I_z	1.934×10^{-2}	$kg\,m^2$
I_{xz}	3.03×10^{-4}	$kg\,m^2$
Wing area (S)	0.143	m^2
Wing span (b)	0.864	m
Maximum aileron deflection ($\bar{\delta}_a$)	55	degrees
Maximum elevator deflection ($\bar{\delta}_e$)	58	degrees
Maximum rudder deflection ($\bar{\delta}_r$)	66	degrees
Maximum motor speed ($\bar{\omega}_m$)	7700	RPM
Propeller radius (R)	0.127	m

2.3.2 Flight experiments

This work uses as an experimental platform the off-the-shelf radio control (RC) aircraft WM Parkflyers McFoamy, upgraded with additional carbon fiber reinforcements and a custom 3D printed motor mount. This is considered an aerobatic airframe and as such is designed to allow for aggressive maneuvering and hover.

Its lightweight EPP foam construction, weighting less that 0.2 kg, results in a take-off weight of 0.45 kg when considering all the necessary equipment. The propulsion system consists of a T-Motor AT2312, 1150 KV Brushless Motor paired with a 10x4.5 Master Airscrew

MR Series propeller; static thrust tests show a maximum thrust of 10 N at the maximum 7700 RPM. These features demonstrate a thrust-to-weight greater that 2 ratio can be achieved. The motor and electronics are powered by a 45C 3S 850mA Lipo battery, with an electronic speed controller (ESC) to precisely adjust the motor's phase voltage to control its rotational speed. A EMAX ES09MD Digital Swash Servo is used to move each of control surfaces. A radio RC receiver is the final component needed for the aircraft to be manually operated.

To enable autonomous and semiautonomous operation, the UAV is equipped with an mRo Pixracer, an autopilot board of the Pixhawk family. This board is built around a 180 MHz ARM Cortex CPU and a 9-axis inertial measurement unit (IMU) composed of a 3-axis accelerometer, 3-axis gyroscope, and 3-axis magenetometer (compass). The Pixracer runs the open source PX4 autopilot firmware. Out of the box, the firmware provides state estimation of the attitude states and, when augmented with a global positioning system (GPS) unit, enables estimation of the full aircraft pose. PX4 includes baseline autopilot control algorithms, but custom control algorithms can be implemented on it, running at a rate of up to 200 Hz. The components of the experimental platform are shown in Figure 2.7. Remote operation, to initialize autonomous maneuvers, or for pilot-assisted flight, is achieved through FrsKy Taranis X9D transmitter. This radio transmitter has the standard four channels typical of fixed-wing operation, and additional channels can be configured through switches to achieve different tasks, such as arming and disarming, and switching flight modes.

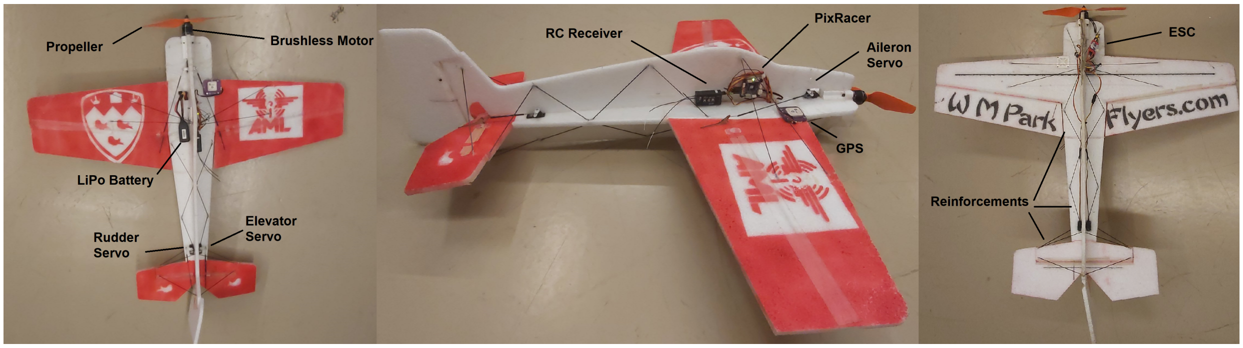


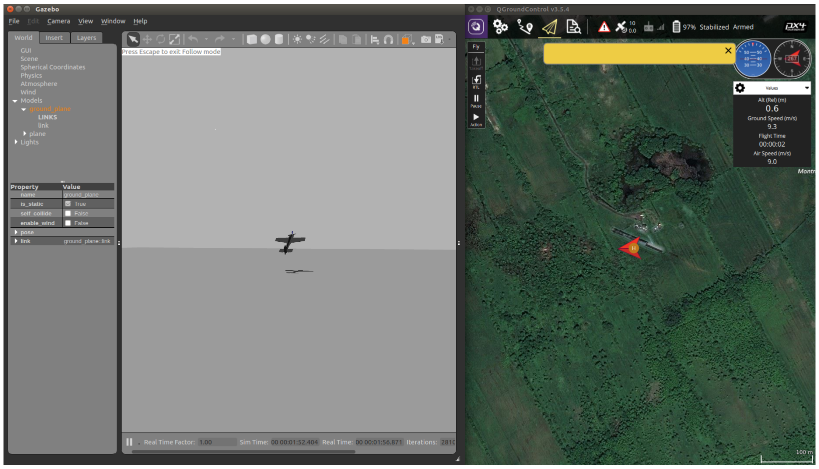
Figure 2.7: Experimental platform components.

Flight experiments were conducted outdoors, at two locations: the West Island Model Aeronautics Club (WIMAC) and the Montreal Area Thermal Soarers Club (MATS).

2.3.3 Software in the Loop

Software in the loop (SITL) is used as an intermediate step between numerical simulations and experimental implementation. The main benefit of a SITL solution is that it allows debugging of the controller code translated from the Simulink simulation to run on the selected firmware. In addition, it further tests the compatibility of the controller with other software routines, in particular, the stock state estimation functions. Finally, it tests the control system running in a realistic, discrete implementation.

PX4 allows for SITL testing through the Gazebo physics engine. The dynamic equations of rigid bodies are propagated by the physics engine itself, but all external forces and moments need to be built as separate functions. Stock PX4-SITL functionality is limited to standard UAVs, and even the stock fixed-wing model employs a basic lift/drag function. To enable its use for the type of aircraft studied in this work, the aerodynamics, thruster, and slipstream models described in the high-fidelity model section were combined and ported into a compliant C++ function, which was then interfaced with a Gazebo linkage representing the McFoamy airframe and its moving parts. The McFoamy CAD model was coupled to the Gazebo linkage to provide visual feedback. Additional feedback is provided through the companion software Gazebo interfaces with, QGround Control (QGC). The software provides standard instrument panel readings, and UAV trajectory when GPS is enabled. QGC can also be used during real flights if an additional telemetry module is added to the aircraft. An example of the SITL simulation environment, together with the added interface on QGround Control, is shown in Figure 2.8.



Property	Value
name	ground_plane
is_static	☑ True
self_collide	□ False
enable_wind	□ False
pose	
link	ground_plane::link

Real Time Factor: 1.00 Sim Time: 00 00:01:52.404 Real Time: 00 00:01:56.871 Iterations: 2810

10 0.0 97% Stabilized Armed

Alt (Rel) (m)
0.6
Ground Speed (m/s)
9.3
Flight Time
00:00:02
Air Speed (m/s)
9.0

Figure 2.8: Gazebo SITL environment with QGC.

This particular SITL implementation also includes functions that simulate real sensor measurements, which the estimator uses to generate state estimates. This is an additional test on the developed control system, which assumed perfect state knowledge.

This simulation environment was used to test the control implementation in a safe manner, rather than risking damage to the physical platform at this precarious stage of development. Further SITL tests were undertaken to further adjust the control gains, as needed. Pilot-assist algorithms, in particular, greatly benefit from this implementation since the RC transmitter used in experiments can connect seamlessly, and the visuals in Gazebo provide visual feedback to a human operator.

Chapter 3

Attitude Control

The attitude controller is the core component of the UAV control system. The separation between position and attitude control obeys the configuration of the system itself. While the agile fixed-wing as a whole is underactuated, the attitude dynamics can be considered fully actuated, and can be driven to any arbitrary configuration. The UAV's position, in turn, can be driven to a reference position through a combination of the body-fixed force magnitude and direction, the latter being prescribed through the UAV attitude.

As such, controlling the UAV's attitude is natural first step, and indeed many position control systems start with the assumption of an underlying autopilot system that controls aircraft attitude. In addition, the capability to control attitude independently is a very useful stand-alone feature; for example, when performing certain aerobatic maneuvers.

3.1 The attitude control problem

Two reference frames are needed to prescribe a reference, or target, attitude. One frame is the already-defined inertial frame, but a new frame, the reference frame $\boldsymbol{\mathcal{F}}_r$, is required. In principle, the definition of its basis vectors, $\hat{\mathbf{r}}_1$, $\hat{\mathbf{r}}_2$, $\hat{\mathbf{r}}_3$ need only satisfy the right-hand rule, since this frame is not necessarily aligned with any specific body. However, it is useful to consider this frame is attached to a virtual aircraft in the same way the body frame basis vectors are aligned with the real UAV. With the reference frame established, a DCM from inertial to reference frame can be defined as $\mathbf{C}_{ri}$ to represent the reference attitude. This reference may be constant but is, in general, a time-varying attitude profile described by a reference kinematic equation,

$$\dot{\mathbf{C}}_{ri} = -\boldsymbol{\omega}_r^{\times}\mathbf{C}_{ri}, \tag{3.1}$$

where $\boldsymbol{\omega}_r$ is the angular velocity of the reference frame with respect to the inertial frame, resolved in reference frame coordinates.

The objectives of attitude regulation and attitude tracking can now be stated.

- **Attitude regulation:** Consider the UAV attitude dynamics given by (2.1) and (2.2). Given a constant attitude $\mathbf{C}_{ri}$, design a control law for $\boldsymbol{\delta}$ such that $\mathbf{C}_{bi} \to \mathbf{C}_{ri}$ and $\boldsymbol{\omega_b} \to \mathbf{0}$ as $t \to \infty$.

- **Attitude tracking:** Consider the UAV attitude dynamics given by (2.1) and (2.2). Given a time varying attitude profile prescribed through the reference kinematic equation (3.1), design a control law for $\boldsymbol{\delta}$ such that $\mathbf{C}_{bi} \to \mathbf{C}_{ri}$ and $\boldsymbol{\omega_b} \to \boldsymbol{\omega}_r^b$ as $t \to \infty$, where $\boldsymbol{\omega}_r^b = \mathbf{C}_{bi}\mathbf{C}_{ri}^T\boldsymbol{\omega}_r$ is the reference angular velocity resolved in the body frame.

Not stated in either objective is the range of initial conditions that guarantee convergence. Depending on the attitude parametrization and control law, the result may be *local* or, if smooth, continuous control laws are used, at most, *almost global* in nature [89]. We focus on developing control laws that are valid for every attitude the UAV might be required to operate at, as opposed to every initial condition. Where applicable, a region of attraction estimate is provided. To enable these estimations, and indeed the overall stability analyses, the following assumption is needed.

Assumption 3.1 (Boundedness of attitude reference)**.** *The reference angular velocity and its first derivative that define the time-varying attitude reference profile 3.1 are assumed uniformly bounded for all time. Explicitly,*

$$\|\boldsymbol{\omega}_r(t)\|_2 \leq \bar{\omega},\ \forall t > 0, \tag{3.2}$$

$$\|\dot{\boldsymbol{\omega}}_r(t)\|_2 \leq \bar{\alpha},\ \forall t > 0. \tag{3.3}$$

Imposing bounds on reference trajectories is a common assumption in control system design and obeys the reasonable expectation that the reference to be followed is a well-behaved signal. It also follows from the physical limitation of the platform; any reference should be a feasible trajectory the aircraft can follow given actuator limitations, structural considerations, etc.

3.1.1 Attitude parametrization

As opposed to a conventional platform, the agile-fixed wing UAV can perform large angle maneuvers, hence it is expected that any attitude reference could be requested. This necessitates that the attitude controller utilize a non-singular attitude parametrization. While several attitude descriptors exist [90], by far the two most popular in UAV control are the quaternion and the DCM itself, often through its Lie algebra [48].

Previous work in AML [67] demonstrated the quaternion parametrization can be used to successfully control the attitude of the agile fixed-wing, and we employed a modified version of this quaternion control law in [91] where a hybrid trajectory-tracking control system was developed. The quaternion parametrization has one noteworthy advantage: the state-estimation algorithms of many modern embedded autopilots, such as the now widespread MEKF [92], provide the attitude estimate in quaternion form. This is true for our implementation, based on Px4. Unwinding results when the control system attempts to drive the attitude to $\mathbf{q}$, when the other, identical attitude, $-\mathbf{q}$ is closer, resulting in a full rotation before reaching the target. This is an undesirable situation, and may occur depending on initial conditions. While the problem can be modified with a sign component that allows the controller to choose the closer element, this constitutes a discontinuous control law which necessitates different analysis tools and can potentially make the system susceptible to measurement noise [32]. As an additional complication, several quaternion conventions exist, which can lead to confusion and errors when integrating systems [93].

Developing the control law directly on SO(3) avoids these problems altogether. The state-estimation advantage of the quaternion is negligible, since a DCM can be easily, and uniquely, recovered from a quaternion through the Rodrigues formula for quaternions,

$$\mathbf{C}(\mathbf{q}) = (q_0^2 - \mathbf{q}_v^T \mathbf{q}_v)\mathbf{I}_3 + 2\mathbf{q}_v\, \mathbf{q}_v^T - 2q_0\mathbf{q}_v^{\times} \tag{3.4}$$

where q_0 and $\mathbf{q}_v$ are the scalar and vector part of the quaternion, respectively. Also, it is easier and less error-prone to change frames with DCM multiplication, essentially a conventional matrix multiplication, than it is to use a dual, quaternion-specific operator. One final advantage of the DCM is the construction of attitude references. When considering the attitude controller as a standalone component, there is little difference between creating arbitrary attitude references in the form of a quaternions or DCMs since a different parametrization is often used to generate the references, such as Euler angles, or the axis-angle description. However, when the attitude reference is generated by an outer control system and additional geometric properties need to be embedded, generating a DCM is the more intuitive option.

For this work, we opt for a geometric control system developed directly on SO(3), but do analyze the benefits of a quaternion parametrization where appropriate.

3.1.2 SO(3) attitude error

The notion of an error, the deviation between two elements, is a fundamental construct in control systems. It is common to use the difference between the target and actual state as the error definition, which the controller then attempts to drive to zero. Indeed, we will define the angular velocity, position, and velocity errors in this manner later.

However, SO(3) is not a vector space but a boundary-less compact manifold [94], and this definition of error would be inadequate. This is made clear by the fact that $(\mathbf{C}_{bi} - \mathbf{C}_{ri}) \notin$ SO(3). For SO(3), matrix multiplication is the binary operator and the error DCM is instead defined as

$$\mathbf{C}_{br} = \mathbf{C}_{bi}\mathbf{C}_{ri}^T, \tag{3.5}$$

which guarantees that $\mathbf{C}_{br} \in$ SO(3). As a relative rotation, the error DCM also has a useful interpretation: it is the DCM from reference to body frame. When both frames align, implying the aircraft has reached the reference attitude, $\mathbf{C}_{br} = \mathbf{I}_3$.

We must now focus on *measuring* the error. For two elements $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$, the error, $\mathbf{e} = \mathbf{x} - \mathbf{y}$ can be measured through the Euclidean norm, $\|\mathbf{e}\|$, a well defined metric in $\mathbb{R}^n$. For the relative attitude problem, obtaining a measure of deviation is not as straightforward. A positive definite error function, $\Psi : \mathrm{SO}(3) \times \mathrm{SO}(3) \to \mathbb{R}$ is needed. Just as different metrics are defined in $\mathbb{R}^n$, different attitude error functions which depend only on the error DCM, $\mathbf{C}_{br}$, have been developed, each with different properties. Before presenting them, it is useful to first define the attitude error kinematics and, as a consequence, the angular velocity error.

3.1.2.1 Attitude error kinematics and angular velocity error

The error kinematics are obtained by differentiating (3.5), substituting the kinematics (2.1) and reference kinematics (3.1). Then

$$\begin{aligned}
\dot{\mathbf{C}}_{br} &= \dot{\mathbf{C}}_{bi}\mathbf{C}_{ri}^T + \mathbf{C}_{bi}\dot{\mathbf{C}}_{ri}^T = (-\boldsymbol{\omega}_b^\times \mathbf{C}_{bi})\mathbf{C}_{ri}^T - \mathbf{C}_{bi}\left(\boldsymbol{\omega}_r^\times \mathbf{C}_{ri}\right)^T \\
&= -\boldsymbol{\omega}_b^\times \mathbf{C}_{br} + \mathbf{C}_{br}\boldsymbol{\omega}_r^\times = -\boldsymbol{\omega}_b^\times \mathbf{C}_{br} + \left(\mathbf{C}_{br}\boldsymbol{\omega}_r^\times \mathbf{C}_{br}^T\right)\mathbf{C}_{br} \\
&= -\boldsymbol{\omega}_b^\times \mathbf{C}_{br} + (\mathbf{C}_{br}\boldsymbol{\omega}_r)^\times \mathbf{C}_{br} = -\left(\boldsymbol{\omega}_b - \mathbf{C}_{br}\boldsymbol{\omega}_r\right)^\times \mathbf{C}_{br},
\end{aligned}$$

where the property $\mathbf{C}\,\mathbf{x}^{\times}\mathbf{C}^{T} = (\mathbf{C}\mathbf{x})^{\times}$ for all $\mathbf{x} \in \mathbb{R}^3$, $\mathbf{C} \in \mathrm{SO}(3)$ was used. This equation gives a natural definition for the angular velocity error,

$$\mathbf{e}_{\omega} \triangleq \boldsymbol{\omega}_b - \mathbf{C}_{br}\boldsymbol{\omega}_r, \tag{3.6}$$

where the DCM transforms the reference angular velocity from reference to body frame, ensuring elements are properly resolved in the same frame before the difference is calculated. Then, the error kinematics can be written as

$$\dot{\mathbf{C}}_{br} = -\mathbf{e}_{\omega}^{\times}\mathbf{C}_{br}, \tag{3.7}$$

which has the same structure as (2.1). For the regulation problem, this equation simplifies to $\dot{\mathbf{C}}_{br} = -\boldsymbol{\omega}_b^{\times}\mathbf{C}_{br}$.

With the error angular velocity defined, the different SO(3) attitude error functions and their properties can now be introduced.

3.1.2.2 Attitude error functions

The choice of error function is a fundamental step in the development of the nonlinear control strategy since a measure of the attitude error is needed for the control law. We consider three possible error functions, with their corresponding innovation terms, or attitude errors.

First is the SO(3) error function developed in [48], which is based on the Frobenius norm,

$$\Psi_1 = \frac{1}{2}\mathrm{tr}\left(\mathbf{I}_3 - \mathbf{C}_{br}\right), \tag{3.8}$$

and its associated innovation term,

$$\mathbf{e}_1 = -\mathcal{P}_a(\mathbf{C}_{br})^{\vee} \tag{3.9}$$

where $\mathrm{tr}(\cdot)$ is the standard trace operator, $\mathcal{P}_a(\cdot)$ is the skew symmetric projection operator defined for square matrices, $\mathbf{A} \in \mathbb{R}^{n\times n}$, as $\mathcal{P}_a(\mathbf{A}) = \frac{1}{2}(\mathbf{A}-\mathbf{A}^T)$ and $(\cdot)^{\vee}$ is the uncross operator defined for skew-symmetric matrices of dimension 3, $\mathbf{N} \in \mathfrak{so}(3)$, such that, for any $\mathbf{x} \in \mathbb{R}^3$, $(\mathbf{x}^{\times})^{\vee} = \mathbf{x}$.

This is the standard SO(3) error function, and is by far the most commonly used in the literature. As we developed the position controller, we made the transition from quaternions to SO(3) for the attitude control component, as it allowed for better integration. This was the error function we originally selected, however, during simulations and experiments, it was found that, for certain maneuvers, the standard quaternion control law outperformed the SO(3) formulation. This motivated research on other attitude error functions, which are presented next.

The second set of attitude error function and innovation term was presented in [95] and is given by

$$\Psi_2 = 2 - \sqrt{1 + \mathrm{tr}(\mathbf{C}_{br})}, \tag{3.10}$$

$$\mathbf{e}_2 = -\frac{1}{\sqrt{1 + \mathrm{tr}(\mathbf{C}_{br})}} \mathcal{P}_a(\mathbf{C}_{br})^{\vee}. \tag{3.11}$$

The third set, developed in [96] for the purpose of observer design, is given by

$$\Psi_3 = \ln(2) - \frac{1}{2}\ln(1 + \mathrm{tr}(\mathbf{C}_{br})), \tag{3.12}$$

$$\mathbf{e}_3 = -\frac{1}{1 + \mathrm{tr}(\mathbf{C}_{br})} \mathcal{P}_a(\mathbf{C}_{br})^{\vee}. \tag{3.13}$$

Any of these functions can be used to construct attitude control laws, but certain properties need to be established first.

Lemma 3.2 (SO(3) error functions)**.** *Consider the sets*

$$\begin{aligned}
\Omega &= \{\mathbf{C}_{br} \in \mathrm{SO}(3) \,|\, \mathrm{tr}(\mathbf{C}_{br}) > -1\}, \\
\Omega_1 &= \{\mathbf{C}_{br} \in \mathrm{SO}(3) \,|\, \Psi_1 < \psi_1\}, \quad \textit{for } 0 < \psi_1 < 2, \\
\Omega_2 &= \{\mathbf{C}_{br} \in \mathrm{SO}(3) \,|\, \Psi_2 < \psi_2\}, \quad \textit{for } \psi_2 = 2, \\
\Omega_3 &= \{\mathbf{C}_{br} \in \mathrm{SO}(3) \,|\, \Psi_3 < \psi_3\}, \quad \textit{for } 0 < \psi_3 < \infty,
\end{aligned}$$

then for each pair of error function Ψ_a and corresponding innovation term, $\mathbf{e}_a$, where $a = \{1, 2, 3\}$, the following statements hold.

(i) The derivative of the error function is given by

$$\dot{\Psi}_a = \mathbf{e}_\omega^T \mathbf{e}_a \tag{3.14}$$

(ii) Each error function Ψ_a is positive definite about $\mathbf{C}_{br} = \mathbf{I}_3$.

(iii) In the set Ω, each Ψ_a has only one critical point, which corresponds to $\mathbf{C}_{br} = \mathbf{I}_3$, and $\mathbf{e}_a = \mathbf{0}$.

(iv) Inside the corresponding set Ω_a, each error function Ψ_a is lower and upper bounded through

$$\underline{b}_a\|\mathbf{e}_a\|^2 \leq \Psi_a \leq \bar{b}_a\|\mathbf{e}_a\|^2, \tag{3.15}$$

for some positive constants $\underline{b}_a$ and $\bar{b}_a$ specific to each error function.

(v) Inside the corresponding set Ω_a, each attitude error function is bounded

$$\|\mathbf{e}_a\| \leq \bar{e}_a \tag{3.16}$$

for some positive $\bar{e}_a$ specific for each error function.

(vi) The derivative of each innovation term is affine with respect to the error angular velocity,

$$\dot{\mathbf{e}}_a = \mathbf{E}_a(\mathbf{C}_{br})\mathbf{e}_\omega, \tag{3.17}$$

furthermore, each matrix $\mathbf{E}_a$ is bounded such that the derivative of each innovation term is upper bounded by the angular velocity error,

$$\|\dot{\mathbf{e}}_a\| \leq p_a\|\mathbf{e}_\omega\| \tag{3.18}$$

for some positive constant p_a specific for each error function.

Proof. The lemma condenses the properties of three different error functions and the proof mostly follows the results presented in [48], [97] [95], and [96]. However, additional analysis was required where previous results were insufficient. The complete proof is given in Appendix A.1. □

There is a relationship between the innovation terms and the angle of the angle-axis parametrization, which is exploited for the proof of Lemma 3.2. Define η, $\hat{\boldsymbol{\epsilon}}$ that satisfy the Rodrigues formula for the error DCM, $\mathbf{C}_{br} = \mathbf{C}(\eta, \hat{\boldsymbol{\epsilon}})$. Then

$$\begin{aligned}
\|\mathbf{e}_1\| &= \|\sin(\eta)\hat{\boldsymbol{\epsilon}}\| = |\sin(\eta)| \\
\|\mathbf{e}_2\| &= \left\|\frac{\sqrt{2}\sin(\eta)}{2\sqrt{1+\cos(\eta)}}\hat{\boldsymbol{\epsilon}}\right\| = \frac{1}{2}\left|\frac{\sin(\eta)}{\cos(\eta/2)}\right| = |\sin(\eta/2)| \\
\|\mathbf{e}_3\| &= \left\|\frac{\sin(\eta)}{2(1+\cos(\eta))}\hat{\boldsymbol{\epsilon}}\right\| = \frac{1}{2}|\tan(\eta/2)|.
\end{aligned}$$

For small angles, $|\eta| \leq \bar{\eta}$, these functions are proportional to each other: $\|\mathbf{e}_1\| \approx 2\|\mathbf{e}_2\| \approx 4\|\mathbf{e}_3\|$, but as η becomes larger, they behave differently. This behavior can be better appreciated in Figure 3.1.

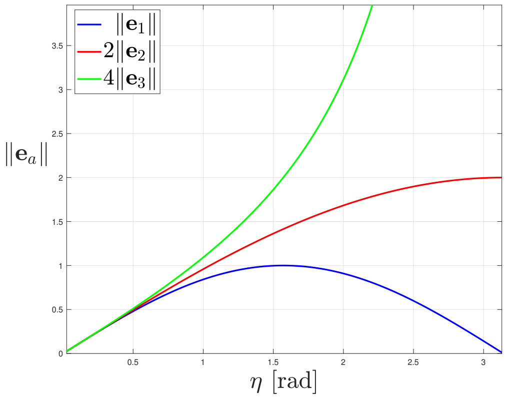


Figure 3.1: Innovation term comparison.

The innovation term $\|\mathbf{e}_1\|$ is most effective at $|\eta| = 90°$ but as $|\eta|$ approaches 180°, its magnitude approaches zero. Depending on the application, this can be a benefit or a hindrance: it prevents a large control action at a large attitude error, but it can also limit the agility of the attitude dynamics. Increasing the gain to improve the behavior results in a larger amplification at small angular errors, which can lead to bad performance, actuator chattering, or instability.

The second innovation term, $\mathbf{e}_2$, aims to correct this shortcoming, and is constructed such that the $\|\mathbf{e}_2\|$ increases with $|\eta|$. The function is nonlinear and tends to 1 as $|\eta|$ tends to 180°. As such, the maximum efficiency of the error function is limited, but increasing the gain to improve the performance at large attitude errors does not deteriorate the control action at smaller angles. Indeed, from Figure 3.1, doubling the gains retains the same behavior at small angles as the previous innovation term, while noticeably increasing the control effectiveness at large angular errors. The fact that the control action is limited by the innovation term can be beneficial, especially when considering actuator saturation.

Meanwhile, innovation term $\mathbf{e}_3$ is very aggressive at large attitude errors. The benefit is that control action at smaller angular errors can be reduced without diminishing the control action at large angular errors. This innovation term, however, is not upper bounded.

A useful analysis is the equivalency of the innovation terms with respect to quaternion innovation terms. As stated previously, the quaternion consists of a scalar and a vector part, and each are closely related to the axis-angle parametrization:

$$\mathbf{q} = \begin{bmatrix} q_0 \\ \mathbf{q}_v \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\eta}{2}\right) \\ \sin\left(\frac{\eta}{2}\right)\hat{\boldsymbol{\epsilon}} \end{bmatrix}.$$

In [40], several quaternion error functions are explored but, by far, the most common continuous innovation term is the vector part of the quaternion itself: $\mathbf{e}_q = \mathbf{q}_v$. If one substitutes the above equation, then

$$\|\mathbf{e}_q\| = \left|\sin\left(\frac{\eta}{2}\right)\right|,$$

which is the same relationship as $\mathbf{e}_2$. This is an important observation: the most common SO(3) innovation term, $\mathbf{e}_1$, is not equivalent to the most common quaternion one; $\mathbf{e}_2$ is. Should a control system be translated from a quaternion to an SO(3) implementation using the most common innovation terms, the behavior will be different. Concretely, performance for large initial angular errors will be greatly diminished. For the sake of completeness, $\mathbf{e}_1$ is analogous to $\mathbf{e}_q = q_0\mathbf{q}_v$ and $\mathbf{e}_3$ is analogous to $\mathbf{e}_q = \mathbf{q}_v/q_0$, two other functions studied in [40]. These equivalences may be shown using simple trigonometric identities.

3.1.3 Attitude error dynamics

As a final step before presenting the attitude control laws, it is necessary to derive the attitude error dynamics. This is done by the following lemma.

Proposition 3.3. *Consider the attitude dynamics described by (2.2), the time variant reference attitude profile given by (3.1), and the angular velocity error definition (3.6). Then the attitude error dynamics are given by*

$$\mathbf{J}\dot{\mathbf{e}}_\omega = \left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_1\right)^\times \mathbf{e}_\omega - \boldsymbol{\Delta}_2 + \mathbf{M}, \tag{3.19}$$

where, as in the previous section, $\mathbf{J}$ *is the UAV's second moment of mass resolved in body frame coordinates, and* $\mathbf{M}$ *are the moments acting upon the aircraft, composed of propeller and aerodynamic effects, and where*

$$\boldsymbol{\Delta}_1 = (2\mathbf{J} - tr(\mathbf{J})\mathbf{I}_3)\mathbf{C}_{br}\boldsymbol{\omega}_r, \tag{3.20}$$

$$\boldsymbol{\Delta}_2 = \mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r + (\mathbf{C}_{br}\boldsymbol{\omega}_r)^\times(\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r), \tag{3.21}$$

Proof. See Appendix A.2 □

Note that both $\boldsymbol{\Delta}_1$ and $\boldsymbol{\Delta}_2$ are bounded by virtue of Assumption 3.1,

$$\boldsymbol{\Delta}_1 \leq \left(2\lambda_M(\mathbf{J}) - \sum_a \lambda_a(\mathbf{J})\right)\bar{\omega} \triangleq B_1, \tag{3.22}$$

$$\boldsymbol{\Delta}_2 \leq \lambda_M(\mathbf{J})\left(\bar{\alpha} + \bar{\omega}^2\right) \triangleq B_2, \tag{3.23}$$

where $\lambda(\mathbf{J})$ refers to the eigenvalues of $\mathbf{J}$ and, $\lambda_M(\mathbf{J})$ and $\lambda_m(\mathbf{J})$ refer, respectively, to the maximum and minimum eigenvalues of $\mathbf{J}$.

With all the necessary tools and definitions, the attitude control laws can now be presented.

3.2 Attitude control laws

3.2.1 Proportional Derivative + Feed-forward controller

The first control law we analyze is the tracking control law developed in [30]. This control is a modification to the one developed in [48], itself sharing the structure from passivity control developments for robot manipulators [98]. The result presented here is a generalization for all three attitude error functions, when using the same control law structure.

The following result is needed first.

Lemma 3.4 (Positively invariant sets)**.** *Consider the system given by (2.1) and (2.2), with the net moments (2.31) and with initial conditions satisfying* $\Psi_a(\mathbf{C}_{br}(0) \in \Omega_a$*, and a reference attitude profile (3.1).*

Consider the control law that governs the control surface deflections

$$\boldsymbol{\delta} = \mathbf{G}^{-1}\left(-\mathbf{M}_a - \mathbf{M}_p - k_\omega \mathbf{e}_\omega - k_a \mathbf{e}_a + \boldsymbol{\Delta}_2\right), \tag{3.24}$$

where $k_\omega > 0$, $k_a > 0$, *and* $\boldsymbol{\Delta}_2$ *defined in (3.21),* $\mathbf{M}_a$, $\mathbf{M}_p$, *and* $\mathbf{G}$ *as defined in (2.31, 2.26).*

Then, for each attitude error function, there exists an initial condition for the angular velocity error, $\mathbf{e}_\omega(0)$ *such that* Ω_a *is rendered a positively invariant set.*

Proof. See Appendix A.3. □

The first two elements inside the parenthesis on the right-hand side are meant to compensate the moments arising from aircraft configuration and the propeller, both consider small compared to the control moments. The third and fourth terms constitute the proportional-derivative feed-back component and the term $\mathbf{\Delta}_2$ is a feed-forward component required by the tracking control analysis.

The limitation on the initial condition of the angular velocity can be overcome by increasing the proportional gain, k_a. For the error function Ψ_3, this is more easily circumvented, since ψ_3 can be arbitrarily large, depending on actuator characteristics.

Theorem 3.5 (PD + feed-forward attitude tracking control law). *Consider the system given by (2.1) and (2.2) with initial conditions satisfying $\Psi_a(\mathbf{C}_{br}(0) \in \Omega_a$ and (A.1), and a reference attitude profile satisfying (3.1) and Assumption 3.1.*

The control law (3.24), with $k_\omega > 0$, $k_a > 0$, and $\mathbf{\Delta}_2$ defined in (3.21), drives $\mathbf{e}_\omega \to \mathbf{0}$ and $\mathbf{e}_a \to \mathbf{0}$ exponentially as $t \to \infty$. In turn, $\mathbf{C}_{bi} \to \mathbf{C}_{ri}$ asymptotically (and $\boldsymbol{\omega_b} \to \bar{\boldsymbol{\omega_r}}$), achieving the attitude tracking objective.

Proof. See Appendix A.4 □

3.2.2 Proportional Derivative controller

The control law 3.24 includes the proportional-derivative feedback terms in addition to compensation for the propeller and aerodynamic moments, and the term $\mathbf{\Delta}_2$ (3.21). As stated in the Preliminaries, the former moments are small compared to the moments generated by the control surfaces, so a mismatch from their nominal value can be expected to have a reduced effect. However, the same cannot be said about $\mathbf{\Delta}_2$ which depends on the aircraft's second moment of mass and, more importantly, the reference angular velocity and acceleration. The control moments need to be generated through the control allocation matrix (2.26) which depends on aircraft aerodynamics and is not perfectly known in practice. As such, it is useful to analyze the effects on the control law if the $\mathbf{\Delta}_2$ term is not known precisely.

We opt to analyze the control law without this term altogether, effectively reducing the attitude control system to a PD controller. The effectiveness of model-independent PD control for regulation is well documented, less so, however, is its applicability for the tracking problem. One such work is [31], which analyzes the trade-off between achievable performance and controller complexity for quaternion-based attitude control systems. The following theorem is, to our knowledge, the first equivalent analysis for an SO(3) controller, and it is applicable for all three attitude error functions.

Theorem 3.6 (PD attitude tracking controller)**.** *Consider the system and conditions of Theorem 3.5, but with the simplified control law*

$$\boldsymbol{\delta} = \mathbf{G}^{-1}\left(-\mathbf{M}_a - \mathbf{M}_p - k_\omega \mathbf{e}_\omega - k_a \mathbf{e}_a\right). \tag{3.25}$$

If in addition the reference trajectory satisfies

$$\rho(t) \triangleq \left(\|\dot{\boldsymbol{\omega}}_r\| + \|\boldsymbol{\omega}_r\|^2\right) \in L_2, \tag{3.26}$$

then $\mathbf{e}_a, \mathbf{e}_\omega \to \mathbf{0}$ as $t \to \infty$, and, as a result, $\mathbf{C}_{bi} \to \mathbf{C}_{ri}$ asymptotically, achieving the attitude tracking objective.

Proof. The proof requires notions derived in the proof of the previous theorem and will often refer to results therein. The proof is given in Appendix A.5. □

From the proof, two characteristics of the control system can be established. First, from (A.8), if the desired attitude profile is selected such that the initial error is zero, then, using Schwarz's inequality,

$$V(t) \leq k_\rho \|\rho(t)\|_{L_2} \|\mathbf{x}\|_{L_2}.$$

it can be shown that the first term on the right grows as the reference trajectory becomes more aggressive but, since $\lambda_m(\mathbf{W}_3)$ increases too, the second term diminishes, as seen from (A.10). Both results are intuitive: as the trajectory becomes more aggressive, the tracking performance will decrease, but this can be improved with larger gains, as they determine the value of $\lambda_m(\mathbf{W}_3)$. The first term is also dependent on $\mathbf{J}$; then, as opposed to the control law with feed-forward, for a given set of gains, the tracking performance will depend on the inertial properties of the UAV. This is the main trade-off between performance and controller complexity and prior system knowledge.

The second observation has to do with the additional constraint, (3.26). The control system with feed-forward only requires the angular velocity and acceleration to be upper bounded. In contrast, this control law necessitates a bound on the full behaviour over time of the reference attitude. Depending on the type of operation, this can be difficult to guarantee. However, for the special case of rest-to-rest maneuvers this is trivially satisfied and this is the expected type of operation of the UAV, both when the attitude reference is prescribed directly, or when generated by the position controller.

As a final remark, if the estimates of the aerodynamic and propeller moments, M_a and M_p and the control allocation matrix, $\mathbf{G}$ are considered accurate enough, then the only perturbation to the nominal system is the non-compensated feed-forward term, $\mathbf{\Delta}_2$. If a rest-to-rest maneuver is considered, then, from (3.21), this term vanishes at the end of the maneuver, and Lemma 9.1 from [99] can be used to show the exponential stability of the origin is preserved. The analysis performed in Theorem 3.6 accounts for the more general case when these conditions are not met.

3.2.3 Effective airflow uncertainty

The control moments are generated by deflecting the control surfaces, with the mapping passing through the the control allocation matrix (2.26). This matrix depends on the airflow over the control surfaces, itself a function of aircraft speed, propeller slipstream, and wind. As such, control moment generation through control surface deflection is inherently less precise when compared to other platforms, such as rotorcraft. We focus specifically on the effective airspeed over the control surfaces, V_δ, when an approximation, $\bar{V}_\delta$ is used.

Consider the tracking control law (3.24), using an imperfect effective airspeed approximation. When substituted in the error dynamics (3.19) results in

$$\begin{aligned}\mathbf{J}\dot{\mathbf{e}}_\omega &= \left(\mathbf{J}\mathbf{e}_\omega + \mathbf{\Delta}_1\right)^\times \mathbf{e}_\omega - \mathbf{\Delta}_2 + \mathbf{M}_a + \mathbf{M}_p + \frac{V_\delta^2}{\hat{V}_\delta^2}\left(-\mathbf{M}_a - \mathbf{M}_p + \Delta_2 - k_\omega \mathbf{e}_\omega - k_a \mathbf{e}_a\right), \\ &= \left(\mathbf{J}\mathbf{e}_\omega + \mathbf{\Delta}_1\right)^\times \mathbf{e}_\omega - (1-\rho_V)\mathbf{\Delta}_2 + (1-\rho_V)\left(\mathbf{M}_a + \mathbf{M}_p\right) - \rho_V k_\omega \mathbf{e}_\omega - \rho_V k_a \mathbf{e}_a,\end{aligned}$$

where $\rho_V = \left(V_\delta / \hat{V}_\delta\right)^2$ is the squared ratio between the real and effective airspeed over the control surfaces.

This mismatch has three main effects, on both the feed-forward and the feedback components. First, it introduces the term $(1-\rho_V)\mathbf{\Delta}_2$, the effects of which can be incorporated within the stability analysis of the PD only control law, (3.25). The second consequence is the appearance of the disturbance term $(1-\rho_V)\left(\mathbf{M}_a + \mathbf{M}_p\right)$ which, in reality, is already present, since perfectly matching the aerodynamic torques generated by aircraft configuration is not possible. These moments are comparatively smaller, but using a mismatched airspeed estimate can amplify them enough to affect performance. An alternative to dealing with this matched uncertainty is to add an integral error component, as in [97].

Finally, the imperfect estimate effectively modifies the proportional and derivative gains and the effect on performance can be significant, since it is squared. If the estimate is much greater than the real value, the surfaces will deflect less, and the UAV will lose control authority, diminishing performance. If the approximation is much smaller, the controller will overcompensate, causing the UAV to oscillate. Prior experimental work [84, 87] on the platform suggests the effective airspeed over the control surfaces does not vary significantly during operation, both in steady and aerobatic maneuvers. One alternative is to simplify the airspeed estimate to different constant values depending on UAV speed, propeller angular velocity, and, if available, wind estimates. An even simpler approach is to keep this estimate constant, at a value considered typical during operation, to minimize the magnitude of ρ_V. The analysis in [84] regarding airflow over slipstream immersed surfaces during forward flight provides a starting point. As can be seen from Figure 3.2, at stations located radially between the center and half the propeller radius, the total airflow remains relatively unchanged. This placement is true for the elevator and rudder, and most of the aileron surface.

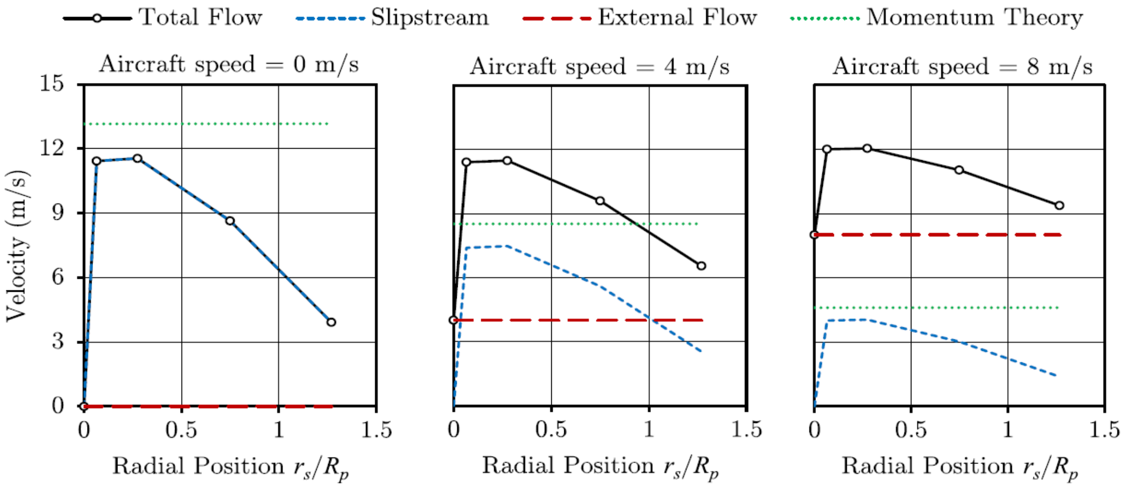


Figure 3.2: Total airflow during forward flight, for a motor speed of 4970 RPM. Radial position is given as a ratio between distance from propeller center and propeller radius. From [84], Section 5.7.

Keeping these calculation constant has two benefits. First, less system knowledge makes the system portable to different platforms. Second, it avoids introducing chattering due to noise-contaminated measurements. The slipstream estimate (2.27) requires velocity measurements and propeller velocity, which are bound to fluctuate when the position control loop is incorporated. If this equation is used, filtering will be required. Using a simplified V_δ estimate can be beneficial, but an integrator term could be required to compensate.

3.3 Attitude control verification

3.3.1 Numerical Simulations

We first employ numerical simulations in Matlab to determine an initial set of gains. While the results of the corresponding theorems serve as guidance, these results are often conservative and allow room for manual adjustment. Numeric simulations are also used to evaluate the different attitude controller configurations, as all conditions within the simulation environment can be specified with precision.

We first evaluate the effect of using different effective airflow approximations, as this will have a direct effect on gain selection.

3.3.1.1 Effective airflow approximation

To study this effect, we choose the vertical loop maneuver, shown in Figure 3.3. Compared to other aerobatic maneuvers showcased later, the vertical loop consists purely of longitudinal motion. This makes it ideal to test the effect of different effective airflow approximations, as sideslip during the maneuver is minimal. The attitude reference for the vertical loop is constructed through the Euler angle parametrization (2.7), where $\psi = \phi = 0$ and $\theta = \frac{2\pi}{t_l}t$, where t_l is the time to complete the maneuver, in seconds. Then, $\mathbf{C}_{ri} = \mathbf{C}(\theta, \hat{\mathbf{k}}_2)$ and, since all motion occurs in a single axis, the matched angular velocity reference is constructed simply by $\omega_r = \dot{\theta}\hat{\mathbf{k}}_2 = \frac{2\pi}{t_l}\hat{\mathbf{k}}_2$.

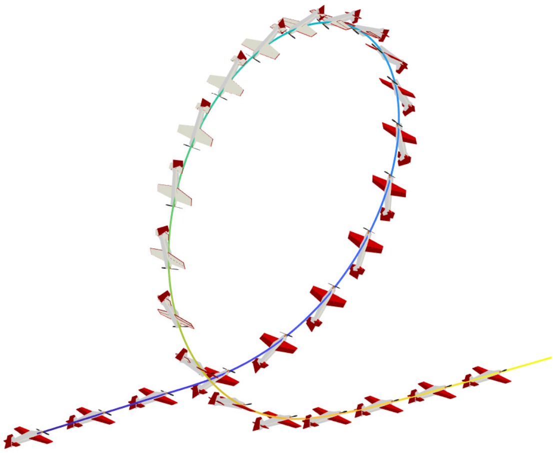


Figure 3.3: 3D view of Vertical loop maneuver, $t_l = 2$ seconds.

In [46], the approximated airflow is constrained to always be above the value corresponding to the thrust required to keep the aircraft in a hover,

$$\hat{V}_\delta = \sqrt{\frac{2\,m\,g}{\rho \pi R^2}} \approx 12\,[m/s],$$

which is meant to avoid excessive control action at low velocities. In practice, this is found to be a conservative lower bound, since the aircraft can still operate with a reduced effective airflow. Still, it serves as a reasonable starting point for a constant approximation. The PD control law (3.25), together with the more agile attitude innovation term (3.11) is used, and controller gains are kept constant at $k_\omega = 0.1569\frac{N\,m\,s}{rad}$ and $k_a = 4.393\frac{N\,m}{rad}$. These gains were tuned by hand, and found to provide the best overall results in the simulation environment. As expected, larger gains provide faster convergence but larger overshoot. The derivative gain, k_ω, is more sensitive to external disturbances, and increasing it will produce oscillations when in the presence of wind. Due to the nature of the first and second attitude innovation terms, the proportional gain, k_a can be increased to improve tracking performance (small errors) without compromising convergence time for attitude setpoints with a large initial angular error. Still, large values, such as 40% more than the selected gain, will result in a high frequency control signal, or actuator chattering.

For this set of simulations, three slipstream approximations are used: a time variant function of aircraft velocity and thrust (2.27), and two constant values, one at $\hat{V} = 12$ [m/s], the hover value, and one at $\hat{V} = 9$ [m/s], a purposefully low value that will cause the control surfaces to deflect more. The time to execute the loop maneuver is set at $t_l = 2\,[s]$.

We first demonstrate the maneuver in ideal conditions, with the UAV operating at the maximum motor RPM, providing maximum control authority, and with no wind. The absolute value of the angle η, calculated from (2.6) is shown in Figure 3.4, where it can be appreciated that the control law, with all the slipstream approximation strategies yields good tracking results.

Plotting the angle η associated with the DCM $\mathbf{C}_{bi}$ (for each of the three simulations) with the angle extracted from $\mathbf{C}_{ri}$ serves an intuitive comparison. However, it should be noted that the attitude controller is not attempting to drive the difference between these two angles to zero, but rather, the innovation term $\mathbf{e}_a$. To compare the performances, it is more useful to analyze the evolution of the innovation term, as in Figure 3.5. To properly compare angular velocity in a single plot, the body angular velocity of each system is resolved in the reference frame, which is shared for all three systems, through $(\boldsymbol{\omega}_b)_r = \mathbf{C}_{br}^T \boldsymbol{\omega}_b$. These angular velocities are shown in Figure 3.6.

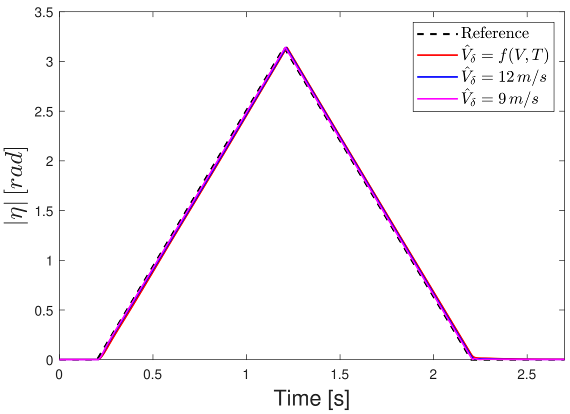


Figure 3.4: Angle η for vertical loop maneuver at max. thrust and no wind.

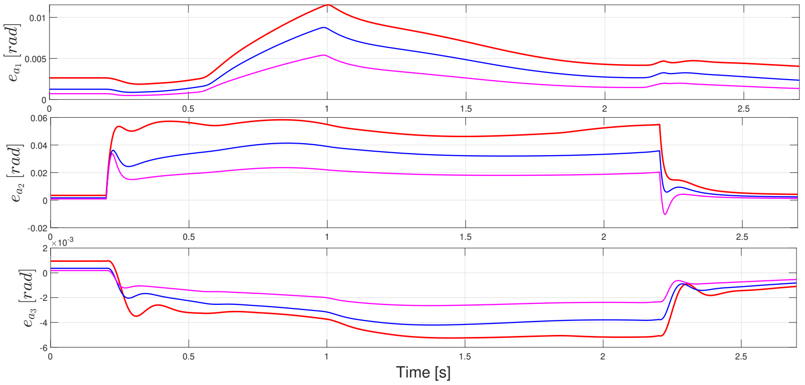


Figure 3.5: Innovation term $\mathbf{e}_a$ for vertical loop maneuver at max. thrust and no wind.

The innovation terms and angular velocities more aptly show the difference in performance between the three strategies. As expected, the smaller approximation of $\hat{V}_\delta = 9$ m/s translates into effective controller gains larger that those of $\hat{V}_\delta = 12$ m/s, resulting in a faster convergence. The approximation corresponding to (2.27) has the worst performance, which can be explained by the calculated $\hat{V}_\delta$, shown in Figure 3.7, which is always larger, thus reducing the effectiveness of the control gains.

A different set of gains can be selected to better operate at the mean value of the approximated airflow, however, if the calculated airflow changes, these gains might no longer be ideal. To show this, another set of simulations is performed, with the same conditions and gains, except the trust is kept at 75% of its maximum value. The approximated airflow using (2.27) is now close to the hover value of 12 m/s. This translates into performance that is very close to the system using this constant value for the approximation, as shown in the innovation terms in Figure 3.8.

The main problem with a variable approximation is a mismatch in the presence of wind.

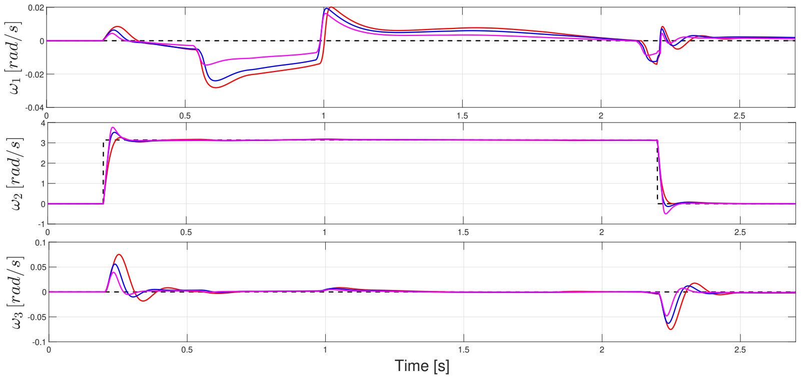


Figure 3.6: Angular velocity $(\boldsymbol{\omega}_b)_r$ for vertical loop maneuver at max. thrust and no wind.

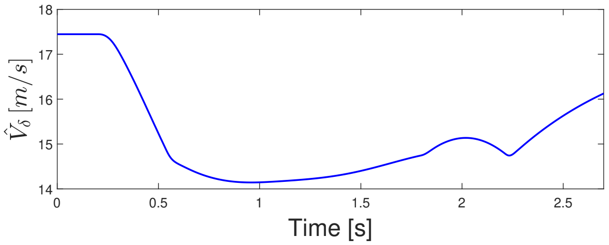


Figure 3.7: Effective airflow approximation, $\hat{V}_\delta$, for vertical loop maneuver, at max. thrust and no wind.

Consider the same conditions and gains as the previous simulation, but with a tailwind of 5 m/s, considered a high wind condition for the platform. When commanded to perform the loop maneuver, the UAV traces the path shown in Figure 3.9, where it can be appreciated that the UAV has little to no translational velocity during most of the vertical segments.

During the initial and final parts of the maneuver, the controller overestimates the true airflow. In contrast, during the loop itself, as the UAV forward velocity tends to zero, the approximation underestimates the airflow as it believes the UAV's airspeed is close to zero, leading to actuator saturation during segments of the maneuver. The corresponding drop in performance is showcased through the larger innovations terms, shown in Figure 3.10.

Even with the differing performances, all approximation strategies show good tracking results. Overall, the lowest, most conservative approximation is consistently better, as it results in the largest effective control gains. A less conservative approximation, such as the one at the hover condition, leads to good results without experiencing excessive saturation. The variable approximation is better when adjusting gains depending on motor velocity, but requires a wind estimate to be more reliable. If this value is not available, using a constant value that avoids oscillations is a good alternative. As a final note, if a constant airflow approximation is used, the problem can be treated as simple gain selection, since for every

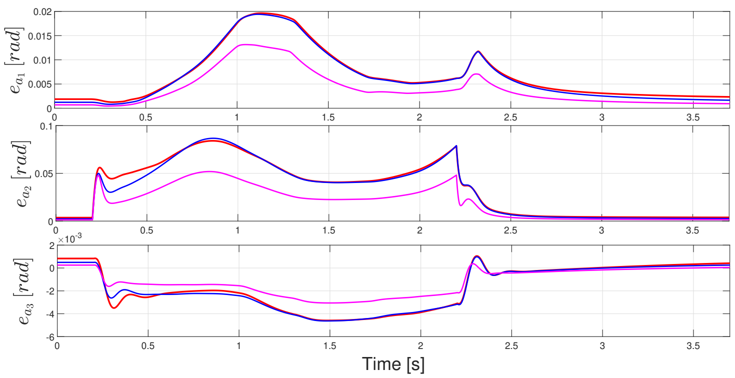


Figure 3.8: Innovation term $\mathbf{e}_a$ for vertical loop maneuver at 75% thrust and no wind.

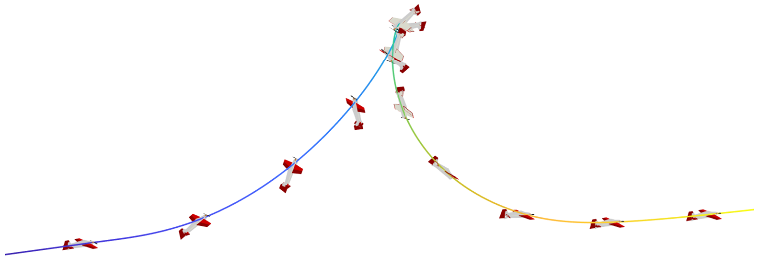


Figure 3.9: 3D view of Vertical loop maneuver, $t_l = 2$ seconds, 5 m/s tailwind.

constant $\hat{V}_\delta$ there exist k_a and k_ω that result in the same effective gains.

3.3.1.2 Control law comparison

We evaluate the difference in performance between the PD plus feed-forward control law (3.24) and the PD control law (3.25) through three different maneuvers. We start with the aggressive turn-around maneuver, a very useful aerobatic maneuver that requires motion around two different axes. We construct this maneuver in two steps, performed over the total duration of the maneuver, t_a. A pitching motion is commanded during the full maneuver, $\theta = \frac{pi}{t_a} t$, while a rolling motion is introduced during the second half of the maneuver to roll the aircraft back into level flight, $\phi = \frac{2pi}{t_a}(t - t_a/2)$. Then $\mathbf{C}_{ri} = \mathbf{C}(\phi, \hat{\mathbf{k}}_1)\mathbf{C}(\theta, \hat{\mathbf{k}}_2)$, which implies $\psi = 0$ throughout. Angular velocity needs to be obtained from the Euler rates through

$$\boldsymbol{\omega}_r = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos(\theta) \\ 0 & -\sin(\phi) & \cos(\phi)\cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{3.27}$$

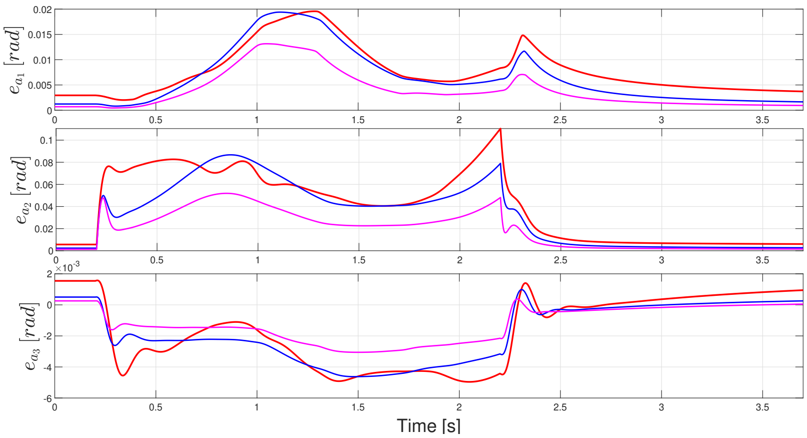


Figure 3.10: Innovation term $\mathbf{e}_a$ for vertical loop maneuver at 75% thrust and 5 m/s tailwind.

where $\dot{\psi} = 0$, $\dot{\theta} = pi/t_a$, and $\dot{\phi} = 2\pi/t_a$ for $t > t_a/2$, $\dot{\phi} = 0$ otherwise. Gains are kept the same as the previous simulations, and the approximated airflow is kept constant at $\hat{V}_\delta = 12$ m/s. Wind is kept at zero, and thrust is set at 75% of its maximum value. The maneuver time is set at $t_a = 2$ seconds to avoid actuator saturation and facilitate the performance comparison. Both systems use the second attitude innovation term (3.11). The 3D view of the maneuver is shown in Figure 3.11.

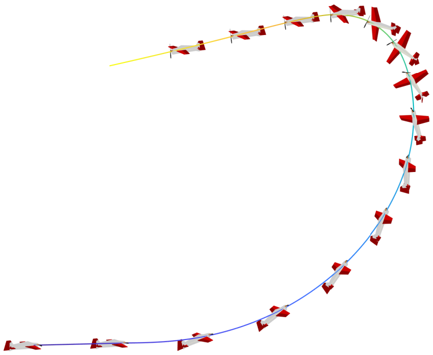


Figure 3.11: The aggressive-turnaround maneuver, $t_a = 2$ seconds.

To compare performance, we plot the innovation terms and the angular velocities; the latter once again resolved in the reference frame, which is shared by both systems. These are shown in Figures 3.12 and 3.13, respectively.

Both control laws result in comparable convergence rates, in particular in the angular velocity states. The differences on the innovation terms, while more noticeable, are nonetheless minor. For the selected control gains, thrust setting, and maneuver parameters, the control surfaces are near saturation; a faster convergence, while possible in theory, is not feasible

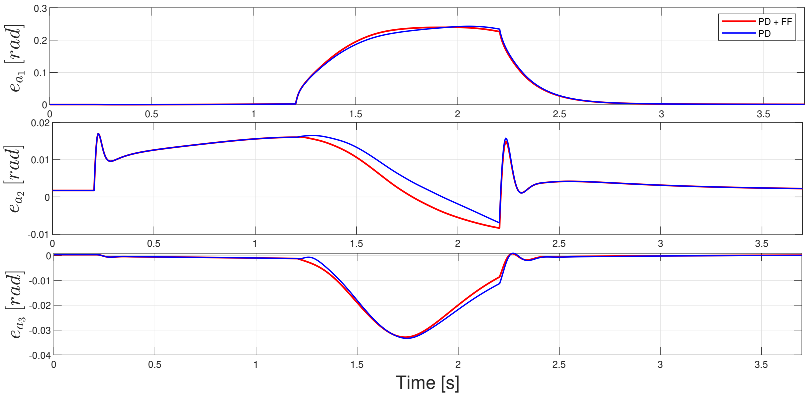


Figure 3.12: Innovation term $\mathbf{e}_a$ for aggressive turnaround.

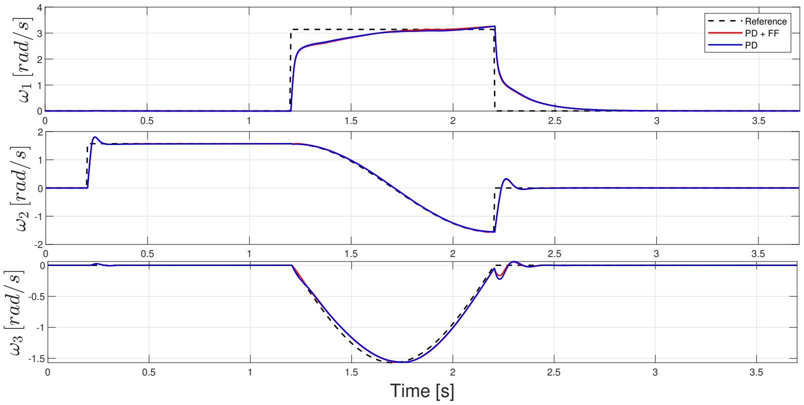


Figure 3.13: Angular velocity $(\boldsymbol{\omega}_b)_r$ for aggressive turnaround maneuver.

due to platform limitations.

While time-varying, the proposed attitude maneuver is discontinuous in $\boldsymbol{\omega}_r$: $\dot{\boldsymbol{\omega}}_r$ is ill defined at at three distinct points. While a continuous maneuver could be constructed, the previous simulation is valuable in showing the effectiveness of the control laws even when the reference attitude profile is discontinuous. We instead consider a smooth trajectory in the next maneuver.

So far, the sample maneuvers have been built through and Euler angle parametrization. while many useful aerobatic maneuvers can be prescribed in this manner, we propose a more complex maneuver. The slanted loop maneuver is achieved through a reference profile constructed through the angle-axis parametrization (2.5), using a constant, but non-trivial rotation axis, and a time-varying angle. To ensure a continuous profile with non-trivial angular acceleration, the angle is prescribed through a fifth order polynomial.

Define $\epsilon_r = [2/3,\, 2/3,\, 1/3]^T$ and $\eta_r = k_3 t^3 + k_4 t^4 + k_5 t^5$, where $k_3 = \frac{20\pi}{t_s^3}$, $k_4 = -\frac{30\pi}{2t_s^4}$, $k_5 = \frac{12\pi}{t_s^5}$, where t_s is the maneuver duration. The reference DCM is then $\mathbf{C}_{ri} = \mathbf{C}(\eta_r, \boldsymbol{\epsilon}_r)$, and the reference angular velocity is constructed through [100]:

$$\boldsymbol{\omega}_r = \left(\sin(\eta_r) - (1 - \cos(\eta_r)\boldsymbol{\epsilon}_r^{\times}\right) \dot{\boldsymbol{\epsilon}}_r + \dot{\eta}_r \epsilon_r, \tag{3.28}$$

for this specific maneuver, $\dot{\boldsymbol{\epsilon}}_r = \mathbf{0}$ and $\dot{\eta}_r = 3k_3t^2 + 4k_4t^3 + 5k_5t^4$. The reference angular acceleration is then $\dot{\boldsymbol{\omega}}_r = \ddot{\eta}_r\boldsymbol{\epsilon}_r$, where $\ddot{\eta}_r = 6k_3t + 12k_4t^2 + 20k_5t^3$.

The simulation keeps the same conditions and gains as the previous one, but changes the maneuver time to $t_s = 2.5$ seconds, to account for the smooth, but slower ramp-up. The 3D view of this maneuver is shown in Figure 3.14, and the innovation term and angular velocity in Figures 3.15 and 3.16, respectively.

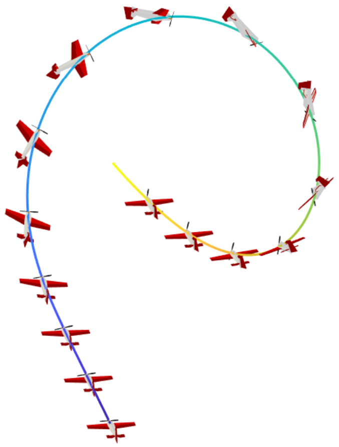


Figure 3.14: The slanted loop maneuver. $t_s = 2$ seconds.

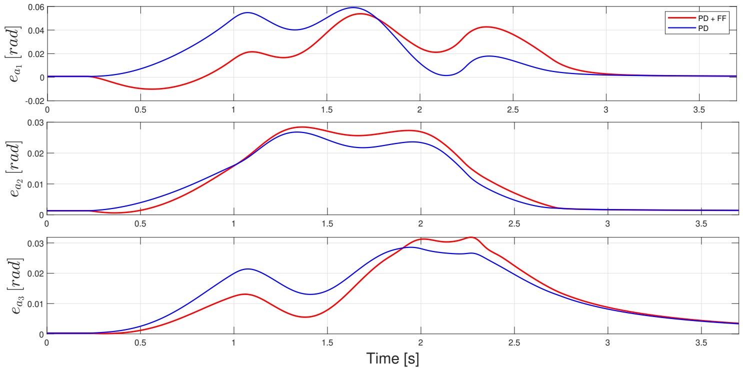


Figure 3.15: Innovation term $\mathbf{e}_a$ for slanted loop maneuver.

Comparing the two control laws, there is a more noticeable improvement on the convergence of the innovation terms, as the error during the full maneuver is lower. Still, tracking

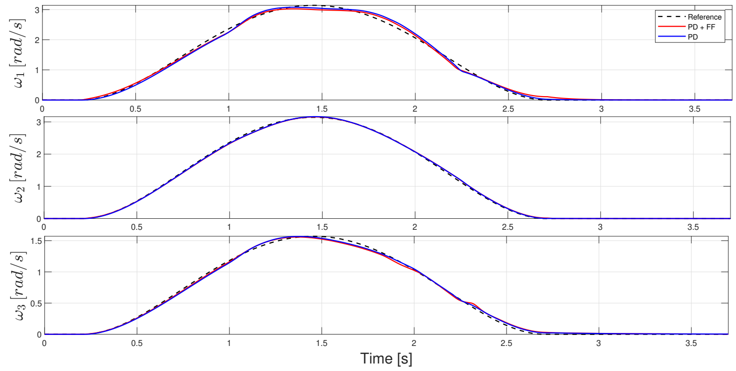


Figure 3.16: Angular velocity $(\boldsymbol{\omega}_b)_r$ for slanted loop maneuver.

is not perfect with the more complex control law, which can be attributed to the mismatch in the feed-forward term due to the effective airflow approximation. Both control laws, while exhibiting some delay, are capable of following the time-varying profile of such a complex maneuver.

As a final comparison between the two control laws, and to further test the controller capabilities, a rolling Harrier maneuver is tested. The rolling Harrier is not rest-to-rest, but evolves perpetually unless commanded to stop. This maneuver is ideal to test the limitations on the PD control system, since the additional condition of Theorem 3.6 is not trivially satisfied. The maneuver is constructed through an Euler angle parametrization, where pitch is held constant at $\theta = 10°$ to compensate for the loss of lift during the maneuver, and roll evolves continuously through $\phi = \omega_{rh} t$, where ω_{rh} is the rate of rotation. Reference angular velocity is prescribed through (3.27), with $\dot{\phi} = \omega_{rh}$.

Simulation conditions are kept the same as in previous simulations, and ω_{rh} is set to 4.5 rad/s. The innovation terms and angular velocity of this maneuver are shown in Figures 3.17 and 3.18, respectively.

While tracking is not perfect, as can be seen from the non-zero innovation terms and the second and third angular velocity components, the error is very small, even for such a fast maneuver. Even though the maneuver evolves *ad infinitum*, the tracking errors do not grow in magnitude for either control law.

Overall, both control laws achieve the attitude control objective with the PD plus feed-forward providing marginally better performance. If computational power allows, using this control law is theoretically the better choice, but any performance improvement will be closely dependent on the accuracy of the system knowledge, particularly on the control allocation mapping. The PD control law offers comparable performance,with the important

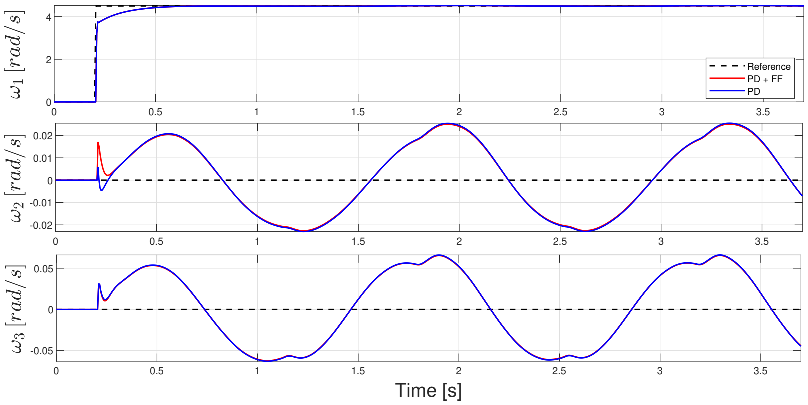


Figure 3.17: Innovation term $\mathbf{e}_a$ for rolling harrier maneuver.

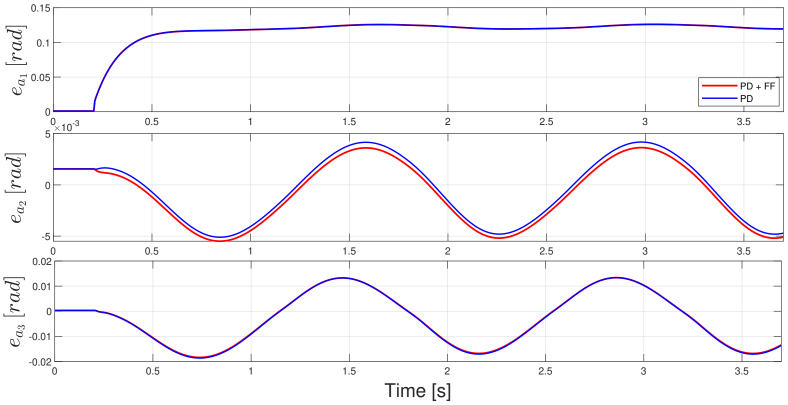


Figure 3.18: Angular velocity $(\boldsymbol{\omega}_b)_r$ for rolling harrier maneuver.

benefit that it requires less system information. As such, it can be considered a good option for the agile UAV platform.

3.3.1.3 Attitude error function comparison

The final maneuver is designed to compare the different innovation terms (3.9, 3.11, 3.13) and corresponding attitude error functions. The gains used in the previous experiment are kept as the baseline, but the proportional gain, k_a, is scaled as shown in Figure 3.1. Since the gains were determined for the second innovation term, the baseline k_a is divided by two when the first innovation term is used, and multiplied by two when the third one is used. The three error definitions were tested with the previously shown maneuvers but no difference in performance was noticed. This is expected: for continuous maneuvers, starting at or close to zero initial attitude error, the three innovation terms, if scaled properly, have equivalent behaviour. This choice will only impact performance when large initial errors are present. To test this, a large, discontinuous roll maneuver is prescribed, where $\theta = 10°$, and roll is instantaneously commanded from zero to $\phi = 178°$, and then back to $\phi = 0°$ after one second. The choice of 178° is meant to avoid the singularities encountered in (3.11) and (3.13), which can be easily accounted for with the former, but not with the latter. Figure 3.19 shows the absolute value of the angle η associated with the body and reference rotation matrices. It can be clearly seen that the innovation term (3.9) is very small when the error is large, and the UAV is not capable of reaching the roll setpoint in time. The other two innovation terms appear to have comparable performance.

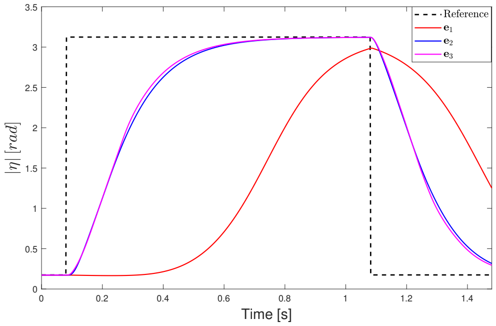


Figure 3.19: Angle η for sudden roll maneuver.

However, when observing the Euler angle parametrization in Figure 3.20, it is revealed that the third innovation term (3.13) causes oscillations in the other two coordinates. The cause can be determined from the control surface deflections, shown in Figure 3.21, which saturate rapidly due to the aggressive nature of the attitude error function.

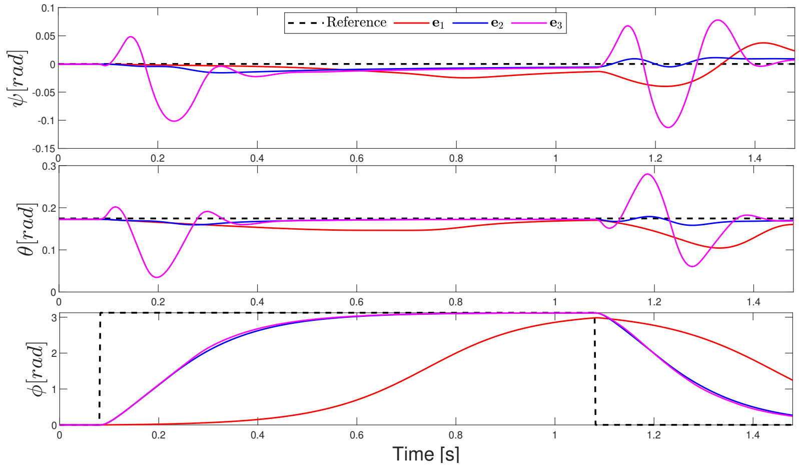


Figure 3.20: Euler angles for sudden roll maneuver.

It should be noted that the innovation term (3.11) also causes the aileron to saturate, thus limiting the performance during the maneuver. However, this saturation occurs due to

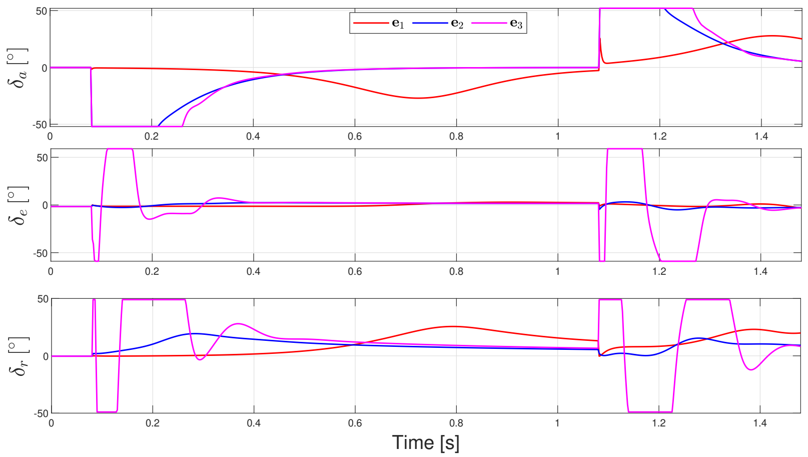


Figure 3.21: Control surface deflections for sudden roll maneuver.

platform limitations and the demanding nature of the maneuver, not because of the attitude error function.

If no sudden, large angle maneuvers are considered for the operation of the UAV, then all three attitude error functions provide similar performance, and the original function (3.9) can be chosen to add some degree of protection should the UAV experience a large disturbance. However, if these maneuvers are considered, the control system will benefit from a more agile error function. Special care must be taken with the more aggressive (3.13), however (3.11) offers similar performance without risking chattering and large oscillations. Due to its simplicity in implementation, and its benefits for large angle maneuvers, which are considered during operation with the position control system, this is the innovation term that is used going forward.

3.3.2 Flight experiments

To verify the functionality of the control system in real operating conditions, and to corroborate the observations gathered from the numeric simulations, several experimental flights are conducted. Gains needed to be adjusted from the simulation values. First, because it is found that different gains for each control moment allow for more flexibility. As such, instead of k_a and k_ω, we substitute these values by diagonal matrices $\mathbf{K}_a$ and $\mathbf{K}_\omega$. The second reason is because the gains used in simulation, which did not account for state estimation, were found to be too large. This is especially true for element associated with the aileron. As the largest control surface in the aircraft, model mismatch in the control allocation impacts this element the most.

In all experiments, gains are kept constant at

$$\mathbf{K}_\omega = \mathrm{diag}(0.0706, 0.7576, 0.7736) \left[\frac{N\,m\,s}{rad}\right]$$
$$\mathbf{K}_a = \mathrm{diag}(0.1656, 1.022, 0.6776) \left[\frac{N\,m}{rad}\right],$$

and, since the aircraft is operating outdoors in varying wind conditions, the airflow approximation is left constant at $\hat{V}_\delta = 6$ m/s, although equivalent gains can be obtained for a different value. Unless otherwise specified, the PD controller (3.25) and the second innovation term (3.11) are used. For ease of operation, the reference maneuver is constructed with an initial rotation around the third inertial axis equal to the vehicle estimated yaw at the time the maneuver is commanded. If the same maneuver is commanded to evaluate different conditions, this initial yaw is kept the same between experiments.

3.3.2.1 The vertical loop maneuver.

A vertical loop maneuver of two seconds of duration is performed in moderate winds (≈ 4 m/s), with the maneuver aligned so that most of the wind is headwind. We use this maneuver to evaluate the drop in performance when the propeller thrust is reduced. The 3D view of the maneuver, for the maximum thrust experiment is shown in Figure 3.22.

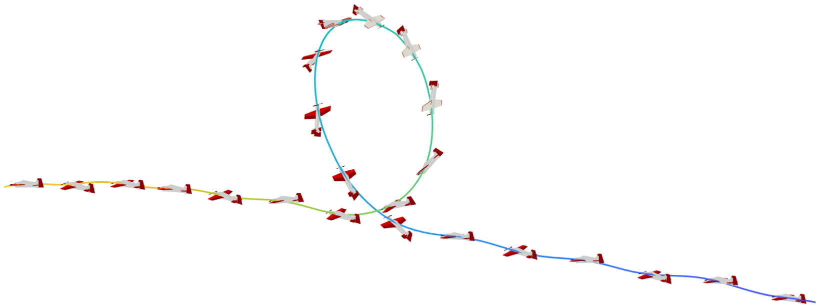


Figure 3.22: Experimental vertical loop maneuver, max. thrust.

The absolute value of the axis-angle parametrization is shown in Figure 3.23, and the corresponding angular velocities are shown in Figure 3.24. As with simulations, all angular velocities are resolved in the reference frame, which is the same for all trials.

As expected, the best tracking performance is achieved at maximum thrust, when the slipstream gives additional authority to the control surfaces. However, the system remains capable of performing the maneuver even at reduced motor speeds, even during portions of the maneuver where wind is coming from the back (tailwind). This test provides a

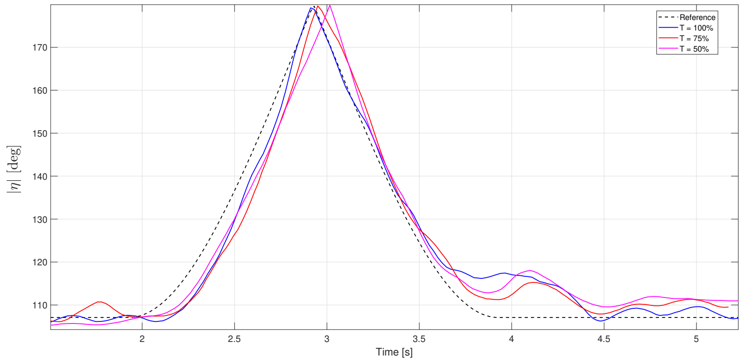


Figure 3.23: Angle η for experimental vertical loop maneuver at different thrust settings.

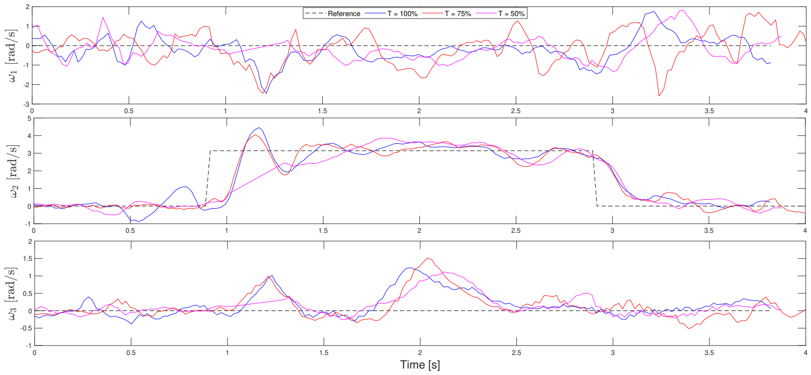


Figure 3.24: Angular velocity for experimental vertical loop maneuver at different thrust settings.

conservative estimate of the minimum thrust necessary to retain control authority, even if performance is diminished. Angular velocity plots show the aircraft experiences oscillations, especially around the first body axis. The control surface associated with this coordinate, the ailerons, is the largest of them, and closest to the propeller. As such, the aircraft is more sensitive to small aileron deflections. Performance can be improved by reducing the control gains on this coordinate.

3.3.2.2 The slanted loop maneuver

Used to evaluate the differences between the two control laws, (3.24) and (3.25), this maneuver was performed under high wind conditions ($\approx$ 7 m/s), and both experiments were conducted with an initial alignment that generates significant crosswind. Wind results in the aircraft drifting, which can be appreciated in the 3D view in Figure 3.25. The aircraft has considerable lateral motion, especially during the loop itself.

The absolute value of the angle η is shown in Figure 3.26, and the corresponding angular

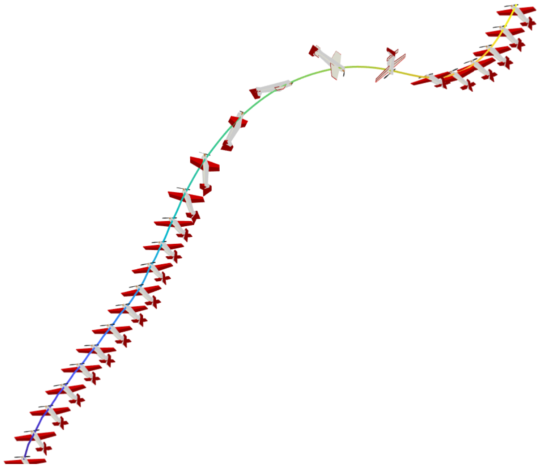


Figure 3.25: Experimental slanted loop maneuver.

velocities in Figure 3.27.

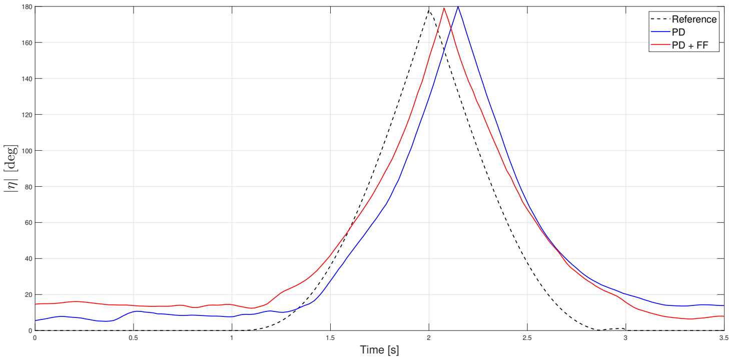


Figure 3.26: Angle η for experimental slanted loop maneuver using different control laws.

The angle plot shows there is an improved performance when employing the more complex PD plus feed-forward control law, but the baseline PD controller still provides comparable results. Of note is the steady-state error before and after the maneuver when the aircraft is tasked to keep a constant attitude. This error is larger than the one encountered during the vertical loop maneuver, and can be attributed to the increase in wind disturbances. This steady-state errors motivates the use of integral action, as in [97].

3.3.2.3 The rolling Harrier maneuver

A rolling Harrier maneuver, with angular rotation of 4.5 rad/s, is tested to verify the UAV does not become unstable when an endlessly evolving maneuver is commanded. The 3D view of the maneuver is shown in Figure 3.28.

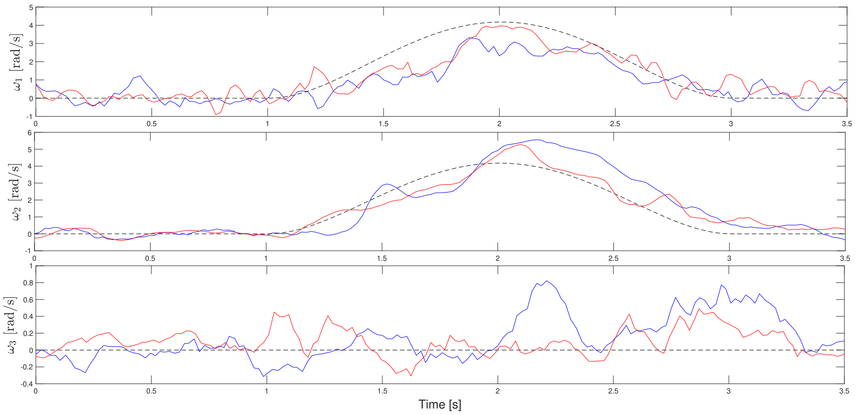


Figure 3.27: Angular velocity for experimental slanted loop maneuver using different control laws.

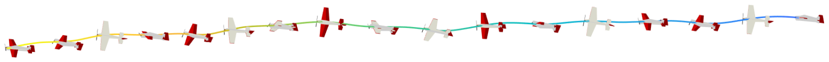


Figure 3.28: Experimental rolling Harrier maneuver.

The condition of gimbal lock is not reached during the rolling Harrier maneuver, so the Euler angle parametrization can be used to provide more descriptive information. The Euler angles are shown in Figure 3.29, and the angular velocity in 3.30.

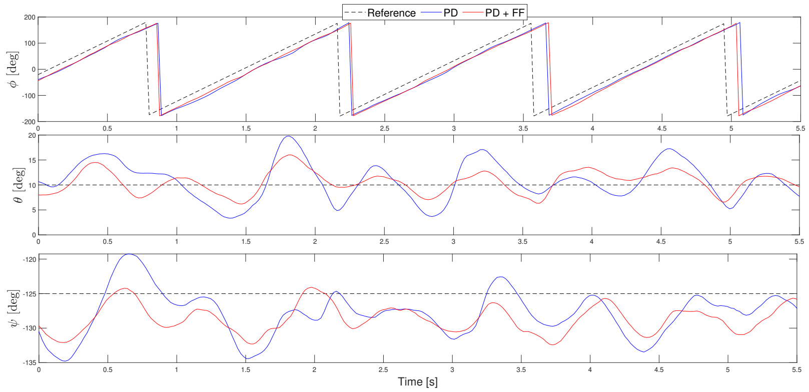


Figure 3.29: Euler angles for experimental slanted loop maneuver using different control laws.

The Euler angles plot shows both controllers are capable of following the reference maneuver, even in the presence of wind. Both controllers demonstrate comparable performance, with the feed-forward component providing better results, particularly in the ‘secondary’ coordinates, also appreciated in the angular velocity plots, where the second and third com-

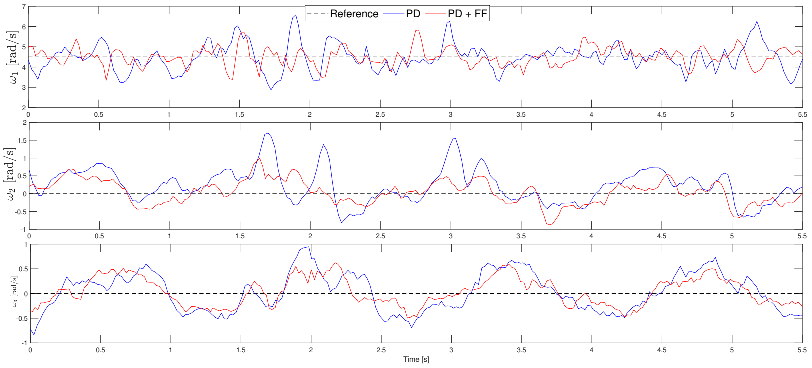


Figure 3.30: Angular velocity for experimental slanted loop maneuver using different control laws.

ponents exhibit less oscillations.

3.3.2.4 Sudden large angle maneuver

Finally, the three attitude error functions are evaluated with this experiment, which shares the parameters of the simulations, a sudden, large rotation around the first body axis, represented by a roll from 0 to 178 degrees. In implementation, it is necessary to shield the computations from numeric overflow. Since the third innovation term increases rapidly, it is constrained to a large, but finite value. The main result is Figure 3.31, where the absolute value of the angle η is shown.

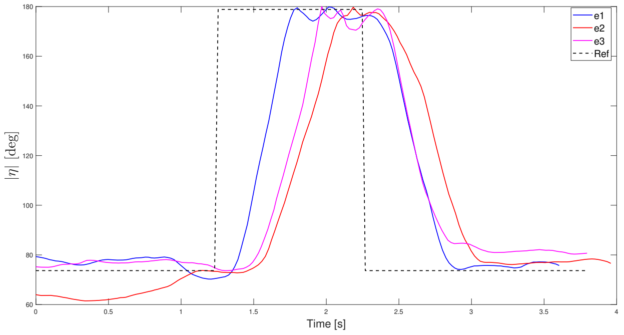


Figure 3.31: Angle η for experimental large roll maneuver, using different innovation terms

As expected, the first innovation term (3.9) results in the worst performance and its slow initial ramp-up can be appreciated. Surprisingly, the most aggressive innovation term (3.13) does not provide the best performance, but this can be attributed to the additional checks that had to be imposed to avoid numeric overflow, which may have the undesired effect of changing its properties. Reinforcing the observations from the simulations, for large angle maneuvers the second innovation term (3.11) provides the best results.

3.3.3 Results discussion

Both simulations and experiments show the proposed attitude control systems enable the agile-fixed wing to reach any target attitude. Several simulations and experiments for the purposes of comparing different strategies were performed, which allow for a more detailed analysis of the different attitude control strategies.

First, we can conclude that the control surfaces do indeed possess an enhanced control authority due to propeller slipstream. The experiments show the control system remains controllable even at reduced velocities, reduced propeller speed, and in the presence of wind. While control authority is diminished and the control surfaces saturate faster, the aircraft is still capable of completing a time-varying attitude maneuver.

Regarding slipstream, or rather, the effective airspeed over the control surfaces, it is found that good performance can be achieved even when a simple estimate is used. Given the difficulty in estimating this feature without additional sensors, using a constant, mean value may provide better results than an inaccurate estimation. Still, a precise estimate can improve performance, as it will adjust for an improved or diminished control authority of the control surfaces, which motivates future research in this topic.

While the more complex PD with feed-forward control law does provide better results, the simpler PD controller has an equivalent performance for most maneuvers. If the inertia matrix is known, and the complete reference angular velocity and acceleration profile are easily obtainable, the former control law should be used. Neither control law is capable of reducing the tracking error to zero, both in simulation and experiments. While foreseeable for the PD controller, the PD plus feed-forward control system is designed for a tracking objective, and, in theory, should drive the error to zero. The shortcomings of this control strategy are closely associated with the imperfect effective airspeed estimate, which prevents the feed-forward terms from exactly cancelling their corresponding terms. As such, the benefits of the PD plus feed-forward control system over the PD controller will largely depend on an accurate estimate. If such an estimate is not available, the benefit of using the more complex control law will diminish.

Finally, in regards to the attitude error functions and their corresponding innovation terms, a straightforward recommendation can be made. Simulations and experiments show little difference for well initialized, smoothly evolving trajectories. However, if large initial attitude errors are considered, the second innovation term provides the best results. The response of the first to large angular errors is significantly slower. In contrast, the third innovation term is too aggressive, and can cause saturation and chattering in the actuators.

Going forwards, the second innovation term, together with the PD plus feed-forward control law and a constant effective airspeed approximation will be used.

Chapter 4

Position Control

In the proposed nested control architecture, the position controller serves as the next loop, tasked with generating attitude references and the thrust commands to enable the aircraft to follow reference trajectories.

Traditionally, fixed-wing control systems aim to control velocity, not position, due to the inherent limitations of conventional platforms. In order to avoid stall conditions, conventional velocity control systems focus on regulating airspeed, rather than a ground-referenced velocity. This is in stark contrast to typical multirotor control literature, where UAV position with respect to a fixed point and the associated inertial velocity are common control objectives. In this work, we aim to exploit the enhanced maneuverability of the agile-fixed wing to meet these more demanding control objectives, while still allowing for more conventional fixed-wing control when necessary. As such, the position control system uses notions of multirotor controllers, but modified and adapted for the characteristics and configuration of the agile fixed-wing UAV.

4.1 The position control problem

To define the position and velocity errors, the definitions for the position, and, in particular, the velocity of the UAV given in Chapter 2 warrant more detail. The physical vector $\underset{\rightarrow}{p}$ describes the relative position between the UAV's centre of mass, c, and the fixed point o, an arbitrary origin. When this vector is resolved in the inertial frame, we obtain the components $\mathbf{p}_i$, which serve as the UAV's position descriptor. Similarly, the physical vector $\underset{\rightarrow}{p_r}$ describes the relative position between some reference point r, which can be considered the centre of mass of some virtual aircraft, and o. When resolved in the inertial frame, we obtain the components $\mathbf{p}_r$, which we define as the reference, or desired position. It is worth noting that the notion of the reference frame, used in the previous chapter, is not at all related to these position definitions. Finally, the physical vector $\underset{\rightarrow}{e_p}$ completes the triad, defined such that $\underset{\rightarrow}{p_r} + \underset{\rightarrow}{e_p} = \underset{\rightarrow}{p}$. These vectors are shown in Figure 4.1. When $\underset{\rightarrow}{e_p}$ is resolved in the inertial frame, we obtain our definition of position error:

$$\mathbf{e}_p = \mathbf{p}_i - \mathbf{p}_r. \tag{4.1}$$

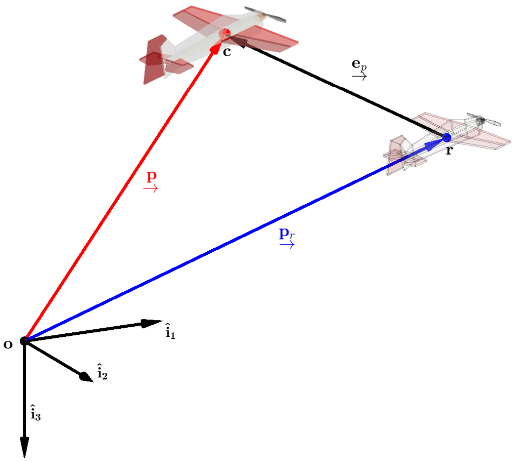

c
$\underset{\rightarrow}{e_p}$
$\underset{\rightarrow}{p}$
r
$\underset{\rightarrow}{p_r}$
o
$\hat{\mathbf{i}}_1$
$\hat{\mathbf{i}}_2$
$\hat{\mathbf{i}}_3$

Figure 4.1: UAV position and reference vectors.

The time derivative of $\underset{\rightarrow}{p}$ with respect to the body frame, resolved in body frame coordinates, gives the components $\mathbf{v}_b$ used in the dynamic equation (2.9). The derivative can also be taken with respect to the inertial frame, using the frame transport theorem [100], and then resolved in the inertial frame, resulting in the components $\mathbf{v}_i$ which we use to describe the UAV's translational velocity in inertial frame coordinates. The dynamic equation (2.9) can be rewritten in terms of $\mathbf{v}_i$ as

$$m\dot{\mathbf{v}}_i = mg\,\hat{\mathbf{k}}_3 + \mathrm{T}\,\mathbf{C}_{br}^T\hat{\mathbf{k}}_1 + (\mathbf{F}_a)_i, \tag{4.2}$$

where $(\mathbf{F}_a)_i$ are the aerodynamic forces resolved in the inertial frame. The reference velocity definition is simpler, as the only frame involved is the inertial frame. We define the reference velocity, $\mathbf{v}_r$, as the components of the derivative of $\underset{\rightarrow}{p_r}$ with respect to the inertial frame, resolved in inertial frame coordinates. The velocity error is then defined as

$$\mathbf{e}_v = \mathbf{v}_i - \mathbf{v}_r \tag{4.3}$$

which can be verified to be the time derivative of the position error (4.1). The control objectives can now be stated with these definitions.

- **Velocity control:** Consider the UAV dynamics given by (4.2), and a reference velocity profile given by $\mathbf{v}_r(t)$. Then, design a control law for the thrust force, T, and a reference attitude $\mathbf{C}_{ri}$, required by the core attitude controller, such that $\mathbf{e}_v \to \mathbf{0}$ as $t \to \infty$.
- **Position tracking:** Consider the UAV dynamics given by (4.2), and a reference inertial position profile given by $\dot{\mathbf{p}}_r(t) = \mathbf{v}_r(t)$. Then, design a control law for the thrust force, T, and a reference attitude $\mathbf{C}_{ri}$, required by the core attitude controller, such that $\mathbf{e}_v \to \mathbf{0}$ and $\mathbf{e}_p \to \mathbf{0}$ as $t \to \infty$.

The main focus is on the position tracking objective, but we consider the less strict velocity control objective as well. This objective is closer to the conventional operation of fixed-wing UAVs and a relaxed requirement can lead to better results under certain conditions, as will be explored in the next chapter.

4.2 Control system

The proposed control system is a unified architecture, in the sense that the same controller is valid throughout the operation of the UAV, as opposed to different, specialized controllers for different flight regimes. This separate structure was first implemented in [101], using a shared quaternion-based attitude controller core, but different position control systems for steady flight and hover. Due to the overlapping attraction regions of both control systems, stability was ensured during transitions, but the switching nature of the controller could cause transient errors even when following smooth trajectories. Both control systems used markedly different control strategies which made proper controller initialization a challenging task. These challenges motivated the design of a unified architecture, to avoid discontinuous switching behaviour and to simplify controller implementation by requiring fewer independent components.

Thrust is the dominant force for agile fixed-wing UAV operation, especially so in near-hover regimes where aerodynamic forces are largely diminished. The aim of the developed control system is to extend the control theory used in other VTOL aircraft during hover to the complete range of operation of the agile fixed-wing. As such, the proposed control system falls under the category of thrust pointing algorithms, such as the ones developed in [59, 97]. This strategy is made clearer by first studying the error dynamics.

4.2.1 Error dynamics

Using the error definition (4.3), the error dynamics can be written as

$$\dot{\mathbf{e}}_v = \dot{\mathbf{v}}_i - \dot{\mathbf{v}}_r = g\,\hat{\mathbf{k}}_3 + \frac{\mathrm{T}}{m}\mathbf{C}_{bi}^T\hat{\mathbf{k}}_1 + \frac{1}{m}(\mathbf{F}_a)_i - \dot{\mathbf{v}}_r. \tag{4.4}$$

The direction of the thrust vector, resolved in the inertial frame, is given by $\hat{\mathbf{b}}_1 = \mathbf{C}_{bi}^T\hat{\mathbf{k}}_1$, which can be driven to some reference $\hat{\mathbf{r}}_1 = \mathbf{C}_{ri}^T\hat{\mathbf{k}}_1$ through the inner attitude controller. Hence, commanding a specific direction of thrust corresponds to prescribing a partial attitude reference through $\hat{\mathbf{r}}_1$.

The error dynamics (4.4) can be rewritten in terms of this partial target attitude, through

$$\dot{\mathbf{e}}_v = g\,\hat{\mathbf{k}}_3 + \frac{\mathrm{T}}{m}\mathbf{C}_{ri}^T\hat{\mathbf{k}}_1 + \frac{1}{m}(\mathbf{F}_a)_i - \dot{\mathbf{v}}_r + \frac{\mathrm{T}}{m}\left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T\hat{\mathbf{k}}_1.$$

In this form, it is clear the second right-hand term can be freely prescribed through a combination of thrust magnitude, T, and its desired direction, $\mathbf{C}_{ri}^T\hat{\mathbf{k}}_1$. This motivates the definition of an intermediate command, $\mathbf{F}_c \in \mathbb{R}^3$, which is prescribed by selecting [59]

$$\hat{\mathbf{r}}_1 = \mathbf{C}_{ri}^T\hat{\mathbf{k}}_1 = \frac{\mathbf{F}_c}{\|\mathbf{F}_c\|}, \tag{4.5}$$

$$\mathrm{T} = m\|\mathbf{F}_c\|. \tag{4.6}$$

Finally, the error dynamics can be written in terms of the intermediate command,

$$\dot{\mathbf{e}}_v = g\,\hat{\mathbf{k}}_3 + \mathbf{F}_c + \frac{1}{m}(\mathbf{F}_a)_i - \dot{\mathbf{v}}_r + \frac{\mathrm{T}}{m}\left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T\hat{\mathbf{k}}_1. \tag{4.7}$$

This structure properly establishes the problem at hand: first, determine the intermediate command, $\mathbf{F}_c$, to satisfy some control law. The intermediate command will determine thrust magnitude, T, and the partial reference attitude, $\hat{\mathbf{r}}_1$, through (4.5, 4.6). Then, complete the attitude reference by setting the other two basis vectors, $\hat{\mathbf{r}}_2$ and $\hat{\mathbf{r}}_3$ through some additional criteria.

We first focus on determining the intermediate command itself, by presenting the position and velocity control laws. We then proceed to completely determine the reference attitude, through the vector-projection algorithm specialized for the agile fixed-wing UAV [91].

4.2.2 Position tracking control law

In multirotor control development, a common simplification is to neglect aerodynamic forces. Doing so enables a simple policy for the intermediate command, such as a proportional-derivative feedback law with a feed-forward component meant to exactly linearize the error dynamics (4.7). Translating this procedure for a winged aircraft was first studied in [62], where the focus was on a tailsitter platform. As determined there, the aerodynamic forces $(\mathbf{F}_a)_i$ depend on the UAV attitude, preventing a closed form solution from being obtained. Their proposed solution is, given an aerodynamic model of the tailsitter, to employ a sequential convex programming algorithm to obtain the reference attitude but the high computational demand prevents its full implementation.

Instead, we propose a different methodology. By using prior attitude measurements, an estimate for the aerodynamic forces $(\hat{\mathbf{F}}_a)_i$ is obtained and used as part of the feed-forward component. Acknowledging the imperfect estimate, an integral term, as in [97], is included, and its effects are included in the stability analysis. Consider the aerodynamic force estimate in the inertial frame,

$$(\hat{\mathbf{F}}_a)_i = \frac{1}{2}\rho b\,\bar{c}\,\hat{V}_a^2\,\mathbf{C}_{bi}^T\mathbf{C}(\hat{\alpha},\hat{\mathbf{k}}_2)\begin{bmatrix}-C_D\\0\\-C_L\end{bmatrix}, \tag{4.8}$$

where the velocity $\hat{V}_a$ and $\hat{\alpha}$ are the approximated airspeed and angle of attack, calculated as

$$\hat{V}_a = \text{sat}_{V_M}\left(\sqrt{u^2+w^2}\right), \tag{4.9}$$

$$\hat{\alpha} = \text{atan2}(w,u), \tag{4.10}$$

where the saturation value, V_M is the maximum velocity the UAV platform can achieve. The lift and drag coefficients, C_L and C_D are approximated through platform dependent polynomial fit,

$$C_L = \begin{cases} P_{Ll}(\alpha) & \alpha \le \alpha_l \\ P_{Lt}(\alpha) & \alpha_l < \alpha \le \alpha_h \\ P_{Lh}(\alpha) & \alpha > \alpha_h \end{cases} \qquad C_D = \begin{cases} P_{Dl}(\alpha) & \alpha \le \alpha_l \\ P_{Dt}(\alpha) & \alpha_l < \alpha \le \alpha_h \\ P_{Dh}(\alpha) & \alpha > \alpha_h \end{cases}$$

where α_l and α_h correspond to the low and high angle of attack regime thresholds, and the three polynomials account for the three angle of attack regimes in [18]: the low alpha regime, the post-stall transition regime where coefficients are interpolated, and the high angle of attack regime where flat plate theory is used.

For the McFoamy experimental platform, these coefficients are given by

$$C_L(\alpha) = \begin{cases} 3.07\alpha, & \alpha \leq 0.271 \\ -0.638\alpha + 1.035, & 0.271 < \alpha \leq 0.482 \\ 0.539\alpha^3 - 2.36\alpha^2 + 2.313\alpha + 0.103, & \alpha > 0.482 \end{cases} \tag{4.11}$$

$$C_D(\alpha) = \begin{cases} 3.23\alpha^2 + 0.0173, & \alpha \leq 0.271 \\ 0.621\alpha + 0.0913, & 0.271 < \alpha \leq 0.482 \\ -0.188\alpha^3 - 0.0264\alpha^2 + 1.42\alpha - 0.2712, & \alpha > 0.482 \end{cases} \tag{4.12}$$

In Figure 4.2, these polynomial approximations are compared with the coefficients obtained in simulation for the main wing segments. It can be appreciated that they capture the stall phenomena at large angles of attack, but with smooth post-stall characteristics.

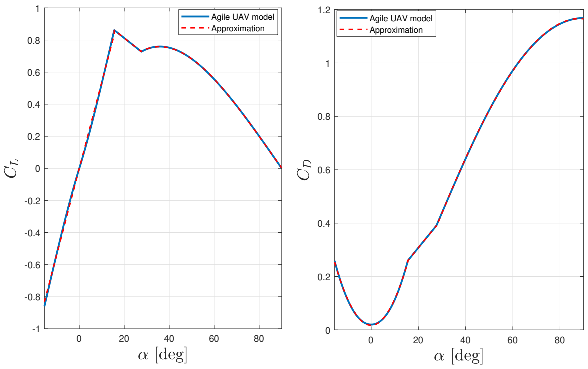


Figure 4.2: Lift and drag coefficient approximations. Solid blue line corresponds to [18], dashed red to the polynomial approximation (4.11), (4.12)

Due to lift and drag coefficients characteristics and the bounded airspeed estimate, the estimated aerodynamic forces can be considered bounded, such that $\|(\hat{\mathbf{F}}_a)_i\| \leq B_a$, for some positive constant B_a.

The intermediate command can now be established through a position tracking control law. Define the intermediate command as

$$\mathbf{F}_c = -k_v\mathbf{e}_v - k_p\mathbf{e}_p - k_i \operatorname{sat}_k(\mathbf{e}_i) - g\hat{\mathbf{k}}_3 + \dot{\mathbf{v}}_r - \frac{1}{m}(\hat{\mathbf{F}}_a)_i, \tag{4.13}$$

where the composite integral error is defined as

$$\mathbf{e}_i = \int_0^t (\mathbf{e}_v(\tau) + c_p\mathbf{e}_p(\tau))\, d\tau, \tag{4.14}$$

where $\text{sat}_k(\cdot)$ is an element-wise saturation function which limits each element to the range $\{-k, k\}$, for $k > 0$, and forpositive gains k_v, k_p, k_v, and c_p.

Before stating the main stability result, the following assumptions are needed.

Assumption 4.1. *Assume the intermediate command satisfies*

$$\|\mathbf{F}_c\| \neq 0, \tag{4.15}$$

and

$$\| - g\,\hat{\mathbf{k}}_3 + \dot{\mathbf{v}}_r - (\hat{\mathbf{F}}_a)_i/m\| < B_p, \tag{4.16}$$

for some positive constant B_p.

While the intermediate command itself has yet to be calculated, these conditions can be satisfied by a well chosen reference profile. The first condition will always be satisfied if a limit on a pure descent is established such that, if $\dot{\mathbf{v}}_r = [0,\ 0,\ \dot{v}_3]$, then $\dot{v}_3 > g$. The second condition obeys a simple maximum reference acceleration, which is a standard requirement in control applications. A conservative bound can be established by using the previously established bound on the aerodynamic force estimate, B_a.

Assumption 4.2. *The aerodynamic force estimate is sufficiently accurate, and some positive constant B_e exists such that*

$$\|(\mathbf{F}_a)_i - (\hat{\mathbf{F}}_a)_i\|_\infty \leq m\,B_e \tag{4.17}$$

The assumption is reasonable if one considers that the aerodynamic forces are, by nature, bounded by the maximum achievable velocity of the UAV, and by other factors such as stall velocity and angle of attack. If wind is considered bounded, then the assumption will hold. In practice, the UAV can only operate within reasonable wind conditions, usually represented as a percentage of the UAV maximum velocity.

We can now state the main result. Due to the conclusions from the previous chapter, we only consider the attitude control combination that yielded the best results, the PD plus feed-forward law, together with the second innovation term, however, the analysis can be adapted to consider other combinations.

Theorem 4.3. *Consider the error dynamics given by (4.7), with initial conditions satisfying*

$$\mathbf{C}_{br} \in \bar{L}_r = \{\mathbf{C}_{br} \in SO(3) \,|\, \Psi(\mathbf{C}_{br}) < \psi < 0.25\}, \tag{4.18}$$

$$\|\mathbf{e}_p(0)\| \leq e_M, \tag{4.19}$$

for some $e_M > 0$, together with the attitude control law (3.24), with innovation term $\mathbf{e_a} = \mathbf{e}_2$ (3.11), and the position control law defined by (4.5), (4.6) and (4.13), (4.14). In addition, for positive constants γ and c_p, the control gains verify

$$k_i\, k > B_e, \tag{4.20}$$

$$\gamma < min\left\{\sqrt{2\lambda_m k_a},\, \frac{2\lambda_m k_\omega}{\lambda_m + 2\lambda_M},\, \frac{4k_a k_\omega \lambda_m^2}{(k_\omega + B_1)^2\lambda_M + 2k_a\lambda_m^2 + 4\lambda_M k_a \lambda_m}\right\}, \tag{4.21}$$

$$c_p < min\left\{\sqrt{k_p},\, \frac{4k_p k_v(1-2\sqrt{\psi})^2}{4k_p k_v(1-2\sqrt{\psi}) + k_v^2(1+2\sqrt{\psi})^2}\right\}, \tag{4.22}$$

and

$$4\lambda_{min}(\mathbf{W}_p)\lambda_{min}(\mathbf{W}_a) > \|\mathbf{W}_{ap}\|_2, \tag{4.23}$$

where

$$\mathbf{W}_a = \begin{bmatrix} \frac{\gamma k_a}{\lambda_M} & -\frac{\gamma(k_\omega + B_1)}{2\lambda_m} \\ -\frac{\gamma(k_\omega + B_1)}{2\lambda_m} & k_\omega - \gamma\left(\frac{1}{2} + \frac{\lambda_M}{\lambda_m}\right) \end{bmatrix}, \tag{4.24}$$

$$\mathbf{W}_p = \begin{bmatrix} c_p k_p(1-2\sqrt{\psi}) & -\frac{1}{2}c_p k_v(1+2\sqrt{\psi}) \\ -\frac{1}{2}c_p k_v(1+2\sqrt{\psi}) & k_v(1-2\sqrt{\psi}) - c_p \end{bmatrix}, \tag{4.25}$$

$$\mathbf{W}_{ap} = \begin{bmatrix} 2c_p(\sqrt{3}kk_i + B_p) & 0 \\ 2(k_p e_M + \sqrt{3}kk_i + B_p) & 0 \end{bmatrix}, \tag{4.26}$$

where $\lambda_m = \lambda_m(\mathbf{J})$ and $\lambda_M = \lambda_M(\mathbf{J})$ are the smallest and largest eigenvalues of $\mathbf{J}$, respectively. Then the errors $\mathbf{e}_p$, $\mathbf{e}_v$, $\mathbf{e}_a$, and $\mathbf{e}_\omega$ converge exponentially to zero, and $\mathbf{e}_i$ is uniformly bounded, verifying the position and attitude tracking objectives.

Proof. See Appendix B.1. □

The combined choice of attitude innovation term (3.11) and the thrust magnitude and direction as functions of the intermediate command (4.6, 4.5), result in a reduction of the allowable initial conditions for attitude error, as shown by (4.18). This value, for the selected innovation term, corresponds to an initial error of $\eta \leq 60°$. This result may appear restrictive, especially when compared to the results in [102], which use a different innovation term and a different mapping between thrust and intermediate command, resulting in an allowed initial angular error of $\eta \leq 90°$.

However, as established in [102], this reduced initial condition is not problematic; the tracking objective will still be verified, albeit in steps. The attitude needs to converge to the more limited initial error condition first. This is also true for the system presented here, and the more aggressive innovation term, depending on the initial condition, can result in η converging faster to the $\eta < 60°$ than the aforementioned controller converging $\eta < 90°$.

To verify this notion, consider the following. Due to the attitude controller, the attitude errors will converge to zero asymptotically regardless of the translational dynamics. This implies there exists some finite time t_c where the UAV attitude will satisfy (4.18). Once this condition is reached, Theorem 4.3 can be invoked. To ensure the system remains stable before this moment, for $t \in [0,\, t_c]$, we must show the position error states remain bounded. This is done through the following Lemma.

Lemma 4.4 (Boundedness of position errors)**.** *Consider the closed-loop control system satisfying the conditions of Theorem 4.3, but with an initial attitude error verifying*

$$\mathbf{C}_{br} \in \bar{L}_l = \{\mathbf{C}_{br} \in SO(3) \,|\, \Psi(\mathbf{C}_{br}) < \psi_l < 2\}.$$

In the time interval $t \in [0,\, t_c]$, after which $\mathbf{C}_{br} \in \bar{L}_r$, the position and velocity errors are bounded.

Proof. See Appendix B.2. □

With this result, the region of attraction of the control system is extended. For a smooth reference trajectory, it is expected that any initial attitude error will be small. However, this result will allow us to prescribe a discontinuous attitude reference without loss of stability in operation, a feature that will be important when determining the complete attitude reference.

4.2.3 Velocity control law

It is useful to possess the capability for velocity control, as this functionality completes the standard autopilot system commonly assumed for the development of guidance algorithms. Indeed, the velocity controller will serve as the basis for the path-following system developed in the next chapter.

Striving for compatibility, the velocity control system is built using the same controller structure as the position tracking control system, such that both functionalities can be enabled or disabled without changing the overall control architecture. As such, the velocity controller is a simple modification to the position tracking controller.

For velocity control, consider instead the following intermediate command

$$\bar{\mathbf{F}}_c = -k_v \mathbf{e}_v - k_p \mathrm{sat}_\nu(\mathbf{e}_{vi}) - g\hat{\mathbf{k}}_3 + \dot{\mathbf{v}}_r - \frac{1}{m}(\hat{\mathbf{F}}_a)_i, \tag{4.27}$$

where

$$\mathbf{e}_{vi} = \int_0^t \mathbf{e}_v(\tau) d\tau., \tag{4.28}$$

with the gains and saturation function as defined for the position controller. Different gains may be selected, but keeping the gains equal between controllers facilitates transition between the two: the integral term of the velocity controller can be substituted by the proportional term of the tracking controller, with careful initialization of the position reference, to switch between the two functionalities while retaining a smooth control law.

Theorem 4.5. *Consider the error dynamics given by (4.7), with initial conditions satisfying (4.18) in closed-loop with the attitude control law (3.24), using the second innovation term (3.11), and the velocity control law (4.27), with the attitude controller gains verifying,*

$$0 < \gamma < min\left\{\sqrt{2\lambda_m k_a}, \frac{2\lambda_m k_\omega}{\lambda_m + 2\lambda_M}, \frac{4k_a k_\omega \lambda_m^2}{(k_\omega + B_1)^2\lambda_M + 2k_a\lambda_m^2 + 4\lambda_M k_a \lambda_m}\right\}, \tag{4.29}$$

and velocity controller gains verifying

$$k_p\nu > B_e \tag{4.30}$$

$$\lambda_{min}(\mathbf{W}_a) > \frac{(2\sqrt{3}k_p\nu + B_p)^2}{4k_v(1 - 2\sqrt{(\psi)}}. \tag{4.31}$$

Then the errors $\mathbf{e}_v$, $\mathbf{e}_a$, and $\mathbf{e}_\omega$ converge exponentially to zero, and $\mathbf{e}_{vi}$ is uniformly bounded, verifying the velocity control objective.

Proof. See Appendix B.3. □

As with the position tracking controller, a large initial angular error outside the sublevel (4.18) is not problematic; the overall stability will be preserved until the attitude subsystem converges. This result is not shown, as it can be considered a special case of Lemma 4.4 where the position error, $\mathbf{e}_p$ is not considered.

The integral term of the velocity control law is almost equivalent to the proportional term in the position controller. The key difference is the lack of true feedback on the UAV position. Under ideal conditions, with no uncertainty or disturbances, both control systems will behave similarly. In practice, the velocity controller on its own will not be able to compensate any drift in position. In this sense, it is a less strict control system.

4.2.4 Vector-projection algorithm

Using either the position tracking control law (4.13) or the velocity control law (4.27) will result in a intermediate command $\mathbf{F}_c$. This command is then generated through a coordinated selection of thrust magnitude and its direction, through (4.5, 4.6).

The control only determines a partial attitude in the form $\hat{\mathbf{r}}_1 \in \mathbb{S}^2$, where $\mathbb{S}^2$ is the two-sphere, defined as $\mathbb{S}^2 = \{\mathbf{x} \in \mathbb{R}^3 | \mathbf{x}^T\mathbf{x} = 1\}$. Aligning the thrust axis is sufficient to achieve the control objective, however, the core attitude controller requires a complete attitude reference. We could disregard the rotation about the thrust axis, and control the reduced attitude directly. Strategies designed on $\mathbb{S}^2$ have been used for quadrotor [103] and satellite control [104], and a modified SO(3) error function is used in [38].

The alternative is to enforce some additional constraint to complete the attitude reference. In [59], a reference quaternion is constructed to minimize the angle of rotation needed to align with the required thrust vector, an idea that is adapted for a controller on SO(3) in [62]. We employed this minimal rotation strategy as the basis for the hover regime controller developed in [101], which produced good results for near-hover maneuvers. However, it could not be used for a more general array of maneuvers because the minimal rotation will not result in a useful flight configuration for a fixed-wing aircraft. For a fixed-wing aircraft, the rotation about the thrust axis has a significant effect on the aerodynamic forces of the aircraft and should be determined carefully.

In [61], a reference DCM is proposed such that the controlled quadrotor satisfies both a thrust direction reference, and, *as much as possible*, a desired direction of its forward facing vector. The vector-projection algorithm proposed in this work is based on this dual objective idea, but the secondary objective is much different. The proposed solution instead completes the attitude reference to achieve a useful fixed-wing UAV configuration while in steady regime. The algorithm is as follows.

- The components of the first reference frame basis vector are calculated through (4.5), explicitly,

$$\hat{\mathbf{r}}_1 = \mathbf{C}_{ri}^T \hat{\mathbf{k}}_1 = \frac{\mathbf{F}_c}{\|\mathbf{F}_c\|}.$$

- The second reference frame basis vector is given by

$$\hat{\mathbf{r}}_2 = \frac{\hat{\mathbf{k}}_3^\times \hat{\mathbf{r}}_1}{\|\hat{\mathbf{k}}_3^\times \hat{\mathbf{r}}_1\|}. \tag{4.32}$$

- The third reference frame basis vector is given by

$$\hat{\mathbf{r}}_3 = \hat{\mathbf{r}}_1^\times \hat{\mathbf{r}}_2. \tag{4.33}$$

- The reference DCM, $\mathbf{C}_{ri}$ is constructed as

$$\mathbf{C}_{ri} = \begin{bmatrix} \hat{\mathbf{r}}_1, & \hat{\mathbf{r}}_2, & \hat{\mathbf{r}}_3 \end{bmatrix}^T \tag{4.34}$$

The algorithm, as its name implies, projects a vector onto a plane. Concretely, it projects the inertial downwards pointing vector, $\underrightarrow{i_3}$, whose components in the inertial frame are $\mathbf{i}_3 = \hat{\mathbf{k}}_3$, onto the plane normal to $\underrightarrow{r_1}$, the desired direction of the thrust axis. This can be seen from

$$\begin{aligned} \underrightarrow{r_3} &= \frac{1}{\left\| \underrightarrow{i_3} \times \underrightarrow{r_1} \right\|} \left(\underrightarrow{r_1} \times \left(\underrightarrow{i_3} \times \underrightarrow{r_1} \right) \right) \\ &= \frac{1}{\left\| \underrightarrow{r_1} \times (\underrightarrow{i_3} \times \underrightarrow{r_1}) \right\|} \left(\underrightarrow{r_1} \times \left(\underrightarrow{i_3} \times \underrightarrow{r_1} \right) \right), \end{aligned}$$

since $\underrightarrow{r_1}$ is unit norm, and will be orthogonal to $\underrightarrow{i_3} \times \underrightarrow{r_1}$. However,

$$\underrightarrow{r_1} \times \left(\underrightarrow{i_3} \times \underrightarrow{r_1} \right) = \left(\underrightarrow{r_1} \cdot \underrightarrow{r_1} \right) \underrightarrow{i_3} - \left(\underrightarrow{r_1} \cdot \underrightarrow{i_3} \right) \underrightarrow{r_1} = \underrightarrow{i_3} - \mathrm{Proj}_{\underrightarrow{r_1}} \underrightarrow{i_3},$$

where $\mathrm{Proj}_{\underrightarrow{r_1}} \underrightarrow{i_3}$ is the projection of $\underrightarrow{i_3}$ onto $\underrightarrow{r_1}$. By subtracting its projection in the direction of $\underrightarrow{r_1}$ from itself, one obtains the projection of $\underrightarrow{i_3}$ onto the plane normal to $\underrightarrow{r_1}$. The vector $\underrightarrow{r_3}$ is simply the normalized projection.

As a consequence of this projection, the second reference vector, $\underrightarrow{r_2}$, will always be contained in the plane normal to $\underrightarrow{i_3}$. Due to the NED inertial frame definition, this means the second reference basis vector will always be contained in the North-East (NE), or horizontal, plane. When the aircraft is aligned with the attitude reference specified by the vector-projection algorithm, it verifies the thrust alignment required to generate the intermediate command, $\mathbf{F}_c$, while enforcing a neutral rotation about the thrust axis. This configuration keeps the aircraft's wing level, as may be appreciated in Figure 4.3.

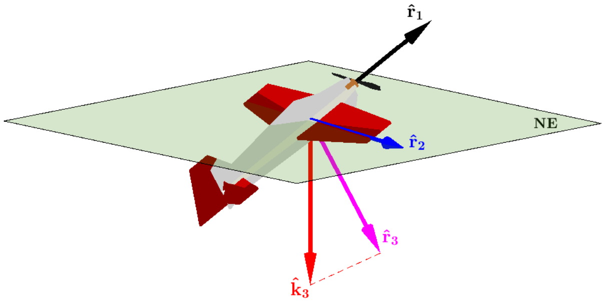


Figure 4.3: Horizontal attitude reference.

In this configuration, when in steady flight regime, the generated lift will help sustain the aircraft, reducing the required thrust force and increasing endurance. When slowing down, as the UAV pitches up, the alignment of the wing will generate drag, helping a transition into thrust-borne flight. While the algorithm commands a null rotation about the thrust axis, in practice it creates a well behaved neutral configuration where an additional rotation can be prescribed independently. In terms of Euler angles, the proposed algorithm effectively decouples the roll coordinate, but this is achieved through a formulation strictly on SO(3).

As will be explored later, there are several options for an independent roll command, including a closed-loop control law. When the reference attitude is augmented with this additional rotation, the control system essentially operates as a set of standard longitudinal and lateral autopilots while in a steady flight regime, but, notably, the same control system is capable of thrust-borne flight and can perform complex aerobatic maneuvers. As such, the vector-projection algorithm serves as an extension for thruster pointing algorithms, that allows the agile-fixed wing to behave like fixed-wing aircraft when necessary, but retaining the high maneuverability enabled by its large thrust-to-weight ratio.

4.2.4.1 Hover singularity

From the definition of the second reference basis vector (4.32), the vector-projection algorithm encounters a singularity when the desired thrust direction, $\underset{\rightarrow}{r_1}$ becomes vertical. Assumption 4.15 will prevent the aircraft from pointing straight down, but the singularity will certainly be encountered when the aircraft transitions into a hover. This obeys a clear physical impediment: the aircraft cannot point its nose upwards while pointing its belly downwards.

The singularity can be avoided by relaxing, or changing, the projection constraint when the aircraft is sufficiently pitched upwards. When the aircraft reaches such a condition, the angle of attack is so large that the wing has effectively stalled, the aircraft is thrust-borne, and the control surfaces retain control authority mainly through propeller slipstream. In this regime, constraining the wing to remain level is not necessary. Instead, we propose to *lock* the wing in position once the aircraft reaches some attitude threshold, as follows.

When the aircraft surpasses a predetermined angle threshold, the projection of the third reference vector onto the NE plane of the last valid step of the vector-projection algorithm, $\hat{\mathbf{r}}_3^*$, is calculated as

$$\hat{\mathbf{h}} = \frac{\left(\hat{\mathbf{k}}_1^T \hat{\mathbf{r}}_3^*\right) \hat{\mathbf{k}}_1 + \left(\hat{\mathbf{k}}_2^T \hat{\mathbf{r}}_3^*\right) \hat{\mathbf{k}}_2}{\left\| \left(\hat{\mathbf{k}}_1^T \hat{\mathbf{r}}_3^*\right) \hat{\mathbf{k}}_1 + \left(\hat{\mathbf{k}}_2^T \hat{\mathbf{r}}_3^*\right) \hat{\mathbf{k}}_2 \right\|} \tag{4.35}$$

In implementation, this is simply the normalized vector created from the first two components of $\hat{\mathbf{r}}_3^*$ while zeroing the third. This direction is, in principle, fixed throughout the operation of the aircraft when at high angles of attack, but it can be modified to rotate the aircraft.

Once $\hat{\mathbf{h}}$ is set, the reference DCM is constructed using the same steps as the vector-projection algorithm but with an alternate second basis vector,

$$\hat{\mathbf{r}}_2 = \frac{\hat{\mathbf{h}}^{\times} \hat{\mathbf{r}}_1}{\|\hat{\mathbf{h}}^{\times} \hat{\mathbf{r}}_1\|}. \tag{4.36}$$

The resulting reference attitude is shown in Figure 4.4, where it can be appreciated that the second reference basis vector is not constrained to the NE plane anymore.

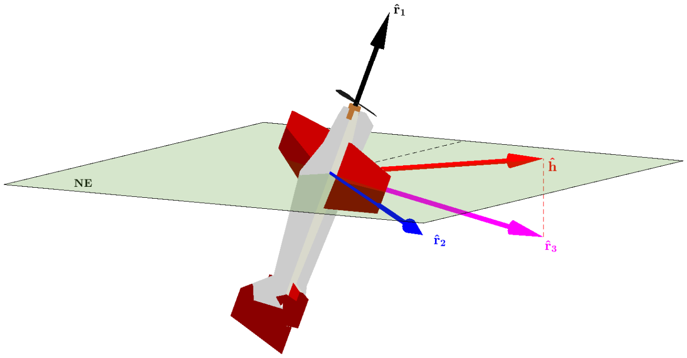


Figure 4.4: Vertical attitude reference.

Defining the fixed direction vector $\hat{\mathbf{h}}$ through (4.35) facilitates a smooth transition from the horizontal to the vertical configuration. It can be verified that, if $\hat{\mathbf{r}}_1$ is kept constant throughout these calculations, then the new $\hat{\mathbf{r}}_3$ will be exactly equal to the previous value, $\hat{\mathbf{r}}_3^*$. In other words, if the intermediate command $\mathbf{F}_c$ is smooth, the reference attitude will be continuous when switching from horizontal to vertical position.

The direction $\hat{\mathbf{h}}$ can be considered, with a slight abuse of notation, as the desired heading of the aircraft. If a smooth transition is not considered, this basis vector could also be constructed through

$$\hat{\mathbf{h}} = \begin{bmatrix}\cos(\psi_r), & \sin(\psi_r), & 0\end{bmatrix}^T, \tag{4.37}$$

where ψ_r is an arbitrary reference heading angle between the North coordinate (first inertial axis), and the projection of the aircraft's belly vector onto the NE plane. In this sense, the alternate reference attitude is equivalent to the solution proposed in [61] for a quadrotor (with a different body axes definition), which is fitting since the UAV behaves like a rotorcraft while in a slow, near-hover regime.

4.2.4.2 Transition considerations

Thanks to the selection of $\hat{\mathbf{h}}$, the transition from horizontal to vertical configuration is guaranteed to be continuous. However, this is not true for the opposite transition. The vector-projection algorithm is stricter in the sense that the base constructor vector, $\hat{\mathbf{k}}_3$ is always fixed, as opposed to $\hat{\mathbf{h}}$ which can be selected to ensure continuity.

Discontinuous operation is, in general, undesirable, but this discontinuity is of less concern. As opposed to hybrid control systems which use distinct control architectures, the proposed control system remains the same in both configurations and thrust magnitude and its direction remain continuous throughout. The only possible discontinuity is in the rotation about the first body axis. Lemma 4.4 shows the system is robust against large initial angular errors, and simulations and experiments show the chosen attitude innovation term, (3.11), allows the attitude control system to rapidly converge to the target attitude from a large initial error, especially in the roll coordinate.

We can use this discontinuous behavior to our advantage. One application is to induce an aggressive turnaround maneuver through the specification of a rapidly changing reference velocity, where the aircraft first enters the vertical configuration, locking the wing in place, and then, as velocity increases in the opposite direction, it will rapidly roll to fly level and non-inverted.

Still, it is desirable to avoid high frequency switching in the control system. To prevent this, we use a simple transition logic with a lower, ξ_l, and upper threshold, ξ_u, where $\xi_l < \xi_u$.

Using two thresholds produces a hysteresis-like effect on the operation, where the direction of change matters. The variable used to compare with the thresholds is the angle between the downwards pointing vector, $\hat{\mathbf{k}}_3$, and the first basis vector of the reference, $\hat{\mathbf{r}}_1$. This angle, $\xi \in [0°, 90°]$ can be calculated as an intermediate step within the vector-projection algorithm, since

$$\xi = \sin^{-1}\left(\|\hat{\mathbf{k}}_3^{\times}\hat{\mathbf{r}}_1\|\right). \tag{4.38}$$

Consider the aircraft is using the horizontal attitude reference. If $\xi > \xi_l$, then the aircraft motion does not risk approaching the geometric singularity and may continue operating as is. If $\xi < \xi_l$, the UAV is close to the geometric singularity and must transition, smoothly, into the vertical configuration, which uses the alternate attitude reference. In this regime, if $\xi < \xi_u$ the aircraft is still considered too pitched up to transition back without risking oscillations, it then remains in the vertical configuration. This is true even if $\xi_l < \xi < \xi_u$; indeed, the region $\xi \in (\xi_l, \xi_u)$ serves as a buffer zone to avoid high frequency switching. If the aircraft lowers its nose enough that $\xi > \xi_u$, then the aircraft may return to the horizontal configuration. This simple logic is better understood from the diagram in the Figure 4.5.

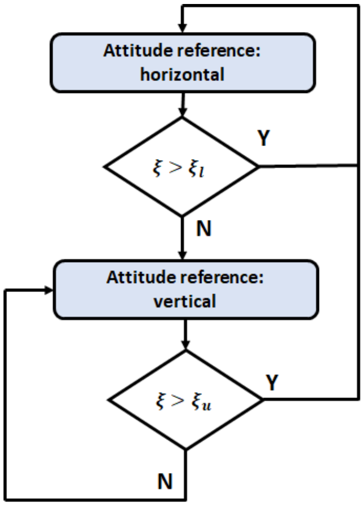


Figure 4.5: Attitude reference transition logic.

With the base attitude determination algorithm established, we proceed to show how an independent rotation about the thrust axis can be exploited to achieve additional objectives.

4.2.5 Rotation about the thrust axis

Although the vector-projection algorithm and its counterpart generate a reference attitude with neutral rotation about the thrust axis, a rolling motion can still be specified independently, by augmenting the reference attitude through a sequential rotation.

Consider some reference rotation about the thrust axis, or reference roll, ϕ_r. Then the augmented reference DCM, $\mathbf{C}_{ri}^{\phi}$ is given by

$$\mathbf{C}_{ri}^{\phi} = \mathbf{C}(\phi_r, \hat{\mathbf{k}}_1)\mathbf{C}_{ri} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi_r) & \sin(\phi_r) \\ 0 & -\sin(\phi_r) & \cos(\phi_r) \end{bmatrix} \begin{bmatrix} \hat{\mathbf{r}}_1^T \\ \hat{\mathbf{r}}_2^T \\ \hat{\mathbf{r}}_3^T \end{bmatrix}. \tag{4.39}$$

Two options for this additional reference roll, ϕ_r are now discussed, in addition to the special treatment given to the vertical attitude reference.

4.2.5.1 Closed loop roll command

The most intuitive application for a roll command is the banking motion employed by a conventional fixed-wing aircraft to control its heading, itself used to steer the aircraft. This serves as the basis for the lateral autopilot of a fixed-wing UAV [2].

While the UAV is capable of lateral motion without the need to bank, owing to its large, slipstream-immersed rudder, doing so is not efficient. Instead, for steady maneuvers, the aircraft should bank to point the lift generated by the main wing to produce lateral acceleration.

Consider the position error resolved in the body frame,

$$(\mathbf{e}_p)_b = \begin{bmatrix} e_{b1} & e_{b2} & e_{b3} \end{bmatrix}^T = \mathbf{C}_{bi}\mathbf{e}_p. \tag{4.40}$$

Then a course command, based on Line-of-Sight (LOS) guidance principles is given by

$$\chi_c = \chi_r + \tan^{-1}(-k_y e_{b2}), \tag{4.41}$$

where $k_y > 0$, and the reference course is

$$\chi_r = \tan^{-1}\left(\frac{v_{r2}}{v_{r1}}\right) = \operatorname{atan2}(v_{r2}, v_{r1}) \tag{4.42}$$

In implementation, it is better to use instead $\tan^{-1}(-k_y e_{b2}) = \operatorname{atan2}(-e_{b2}, d_y)$, where $d_y = 1/k_y$ is a look-ahead constant that determines the convergence characteristics.

In [52], this reference course is differentiated to solve for a reference roll through the coordinated turn equation [2]. We propose instead a Proportional Integral (PI) control law for course, as is common in fixed-wing autopilot design. The closed loop reference roll is given by

$$\phi_r = k_{\phi_p}(\chi - \chi_c) + k_{\phi_i} \int_0^t (\chi(\tau) - \chi_c(\tau))d\tau, \tag{4.43}$$

where the aircraft course is calculated through its conventional definition as

$$\chi = \tan^{-1}\left(\frac{v_{i2}}{v_{i1}}\right) = \text{atan2}(v_{i2}, v_{i1}). \tag{4.44}$$

This closed loop control law will create a roll reference that attempts to reduce the aircraft's lateral error in the body frame. If the aircraft roll is significant, this second body referenced error could also capture a vertical error quantity, or altitude error. Ideally, the aircraft should only roll to compensate for a purely lateral error, not to adjust altitude. A better solution requires the definition of a moving path frame and will be discussed in the next chapter.

4.2.5.2 Aerobatics

The second application for a rotation about the thrust axis is unique to the agile fixed-wing UAV. Since it has excess side-force authority, a roll command can be prescribed independently to perform a series or aerobatic maneuvers while still achieving the control objectives. This notion was originally suggested in [105], where a slow rolling Harrier and a knife-edge are performed as the aircraft navigates a path. Thanks to the roll decoupling property of the vector-projection algorithm, these maneuvers can be easily prescribed without the need to adjust the underlying control structure, or to add an additional lateral controller. Three common aerobatic maneuvers can be enabled straightforwardly:

- A Rolling Harrier can be achieved by setting $\dot{\phi}_r = \omega_{rh}$, where ω_{rh} is the rate of rotation of the rolling motion, and ϕ_r can be obtained through basic numeric integration.
- A knife-edge is commanded by setting $\phi_r = \pm\pi/2$ [rad].
- Inverted flight is achieved through $\phi_r = \pi$ [rad].

While performing these maneuvers, some consideration are needed for the aerodynamic force estimate. Since the aircraft has a symmetric (flat) airfoil, the aerodynamics during inverted flight will be very similar to those when the wing is level. For a knife-edge maneuver, lift will be greatly reduced. In [67], where the aerodynamic estimate is necessary for the thrust controller, the loss in lift is accounted for by reducing the estimate by an empirically obtained factor. In this work, we show that the feedback part of the position control system can compensate even if a zero estimate is given. This is especially useful for the Rolling Harrier, which is not a steady-state maneuver, preventing our proposed aerodynamic estimate from being used. Still, the thrust-based control system should be able to compensate for the oscillating aerodynamic forces of the aircraft during this maneuver.

4.2.5.3 Hover maneuvers

Due to the structure of the vertical attitude reference, there is are two mechanisms to induce a rotation about the thrust axis. Modifying roll is the simplest, as it can be prescribed by augmenting the reference DCM through (4.39), as before. However, this can also be achieved by rotating the constructor vector, $\hat{\mathbf{h}}$.

This latter maneuver is the more intuitive option for a remote operator, since it allows for a piloting scheme similar to that of a multirotor UAV. In conventional quadrotor control, translational references for are specified with respect to aircraft geometry: one stick commands motion from left to right, and the other commands motion forwards and backwards. Vector $\hat{\mathbf{h}}$ doubles as a reference direction for these commands: forward and backward motion is parallel to it, lateral motion is normal to it. Modifying $\hat{\mathbf{h}}$ automatically shifts this reference, but a rotation about the thrust axis created by a sequential rotation will not realign $\hat{\mathbf{h}}$. While a rotation about the thrust axis during a hover could be achieved through (4.39), it is better to rotate the aircraft by increasing or decreasing the initial $\hat{\mathbf{h}}$ vector.

To achieve this, instead of using (4.35) to calculate the initial, $\hat{\mathbf{h}}$, an initial equivalent heading can be obtained through

$$\psi_0 = \sin^{-1}\left(\frac{r^*_{3_2}}{\sqrt{(r^*_{3_1})^2 + (r^*_{3_2})^2}}\right), \tag{4.45}$$

where $r^*_{3_1}$ and $r^*_{3_2}$ are the first and second components of $\hat{\mathbf{r}}^*_3$. Then, if an additional rotation, ψ_r is prescribed, calculate $\hat{\mathbf{h}}$ through

$$\hat{\mathbf{h}} = \Big[\cos(\psi_0 + \psi_r), \quad \sin(\psi_0 + \psi_r), \quad 0\Big]^T. \tag{4.46}$$

This additional rotation, ψ_r, is a free parameter that can be specified by the pilot or an autonomous reference generator.

4.3 Position control verification

We first evaluate controller features through representative simulations, and then analyze the results of experimental flights. This section focuses on the position tracking control law (4.13), since the velocity controller (4.27) can be considered a less strict variant of it. However, the velocity controller serves as the basis of the path-following algorithm presented in the following chapter, as such, its functionality will be tested when testing the path-following algorithm.

4.3.1 Numerical simulations

All Matlab simulations share the same controller gains and parameters, given in Table 4.1. Both during simulations and experiments, it is found that selecting different values for the gains associated with the third inertial coordinate yield better results. This is reflected in the gains $\mathbf{K}_p$, $\mathbf{K}_v$, $\mathbf{K}_i$ being used instead of one shared constant. Parameters are divided between attitude control, position control, closed-loop roll, and transition thresholds. Simulations assume ideal, no wind conditions.

Table 4.1: Controller parameters for simulation.

Param.	Value	Units
k_a	4.393	$\frac{N\,m}{rad}$
k_ω	0.1569	$\frac{N\,m\,s}{rad}$
$\hat{V}_\delta$	12	m/s
$\mathbf{K}_p$	diag(1.08, 1.08, 3.6)	$1/s^2$
$\mathbf{K}_d$	diag(0.672, 0.672, 0.336)	$1/s$
$\mathbf{K}_i$	diag(0.008, 0.008, 0.04)	$1/s^3$
k	10	$m\,s$
k_y	0.2	1
k_{ϕ_p}	4.32	1
k_{ϕ_i}	0.02	s
ξ_u	30	deg
ξ_l	15	deg

Similarly to the attitude control system, larger gains result in faster convergence at the cost of performance. The system is found to be more sensitive in the third coordinate for the derivative gains, a larger value results in noticeable oscillations. This can be attributed to the notion that large, sudden motions in the vertical direction directly affect the angle of attack, and hence, the aerodynamic characteristics of the aircraft. However, this coordinate benefits from larger proportional and integral gains to reduce the steady-state error, especially during steady flight where the lift estimate can be inaccurate, resulting in an effective disturbance on this axis.

The lower angle threshold, ξ_l, is an important parameter, since it will determine when the transition from steady to hover flight occurs. This transition is more demanding than its counterpart, because it requires the UAV to decelerate. In general, a larger value will result in a smoother transition, since the wing will "lock in place" sooner and help the UAV slow down. Since the other threshold, ξ_u, needs to be larger to avoid undesired switching behaviour, the trade-off is determining a value for ξ_l that provides good performance while still allowing the aircraft to quickly transition back to steady flight. In this work, the value of $\xi_l = 15°$ provides good results.

4.3.1.1 Longitudinal and hover maneuvers

The composite maneuver shown in Figure 4.6 allows us to test, and demonstrate, several controller features, specifically, longitudinal steady performance, transitions to-and-from hover, and hovering capabilities. A close-up of the hovering section of the flight, which includes both transitions is shown in Figure 4.7. The inertial position and velocity of the UAV are shown in Figures 4.8 and 4.9, respectively. For the 2D plots, vertical dash and dotted lines indicate changes in the prescribed reference trajectory.

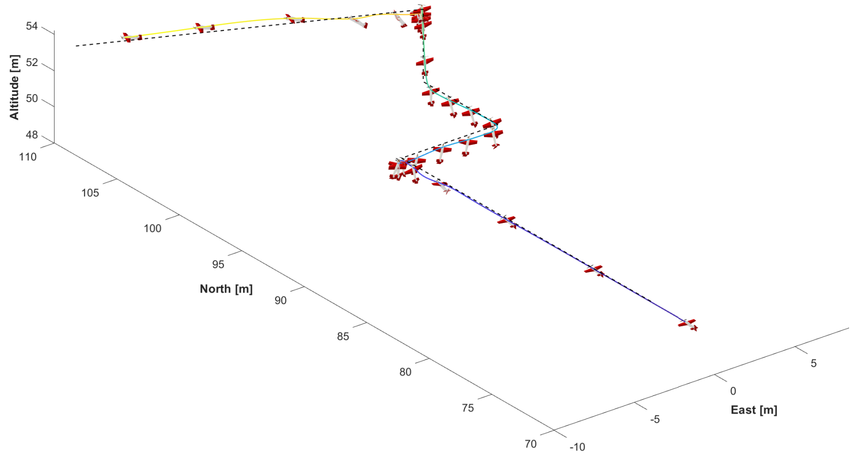


Figure 4.6: 3D view of composite maneuver.

The maneuver is best understood from the velocity plot in Figure 4.9. Initially, a velocity of 10 m/s with zero heading is commanded. Between $t = 3$ and $t = 6\,s$, the UAV is

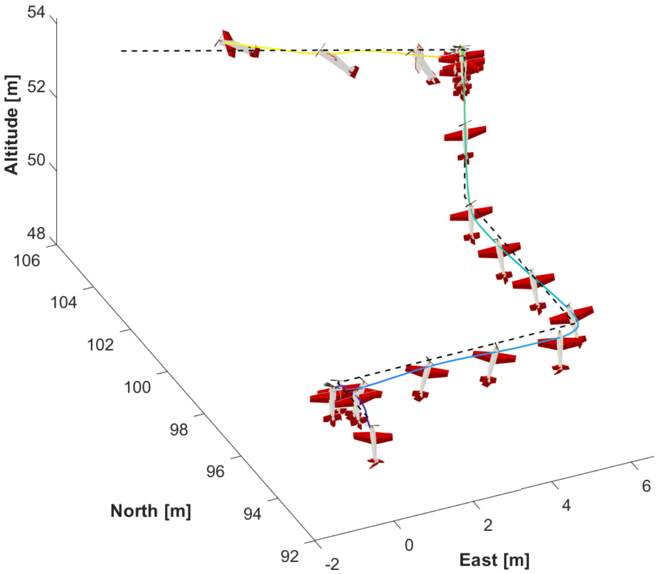


Figure 4.7: 3D close-up of hovering maneuvers.

commanded to slow down with constant deceleration until zero velocity is reached, forcing the aircraft to hover in place. A key aspect of this maneuver is the transition between the two attitude determination algorithms, which occurs at $t = 4.72\,s$. The effects of locking the wing can be seen in the oscillations in the velocity plot, but aircraft motions remains smooth through the transition.

The UAV is commanded to hover in place for 3 seconds, then, at $t = 9\,s$, a lateral velocity reference of $V = 2\,m/s$ is prescribed. This velocity is commanded perpendicular to $\hat{\mathbf{h}}$, and the UAV hovers by tilting sideways, as a multirotor would. This maneuver is notable since, given the choice of body reference frame, it cannot be achieved with a single set of Euler angles: it effectively requires a "fourth gimbal" to be prescribed. At $t = 12\,s$, a velocity of $V = 2\,m/s$, but this time in the direction of $\hat{\mathbf{h}}$ is prescribed. The UAV achieves this motion by tilting forward, in contrast to the previous maneuver.

At $t = 15\,s$, a strictly vertical climb at $V = 2\,m/s$ is commanded. After 2 seconds the velocity abruptly changes to zero, and the aircraft exhibits overshoot in the vertical coordinate, which can be appreciated on all 2D plots. When slowing down in the NE plane, the drag generated by the wing acting as a flat plate helps the UAV slow down, but this mechanism is not available when performing purely vertical maneuvers, hence the overshoot. Better performance can be obtained with a smooth reference, for example a third order velocity profile. Between $t = 18$ and $t = 21\,s$ the reference velocity remains at zero, but a change in heading is achieved by commanding $\dot{\psi}_r = 2$ [rad/s] through (4.46). This rotation

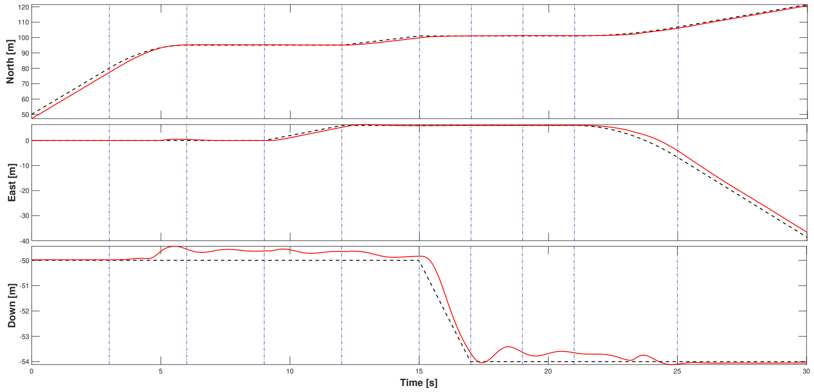


Figure 4.8: Inertial position for composite maneuver. Dashed line corresponds to the reference trajectory.

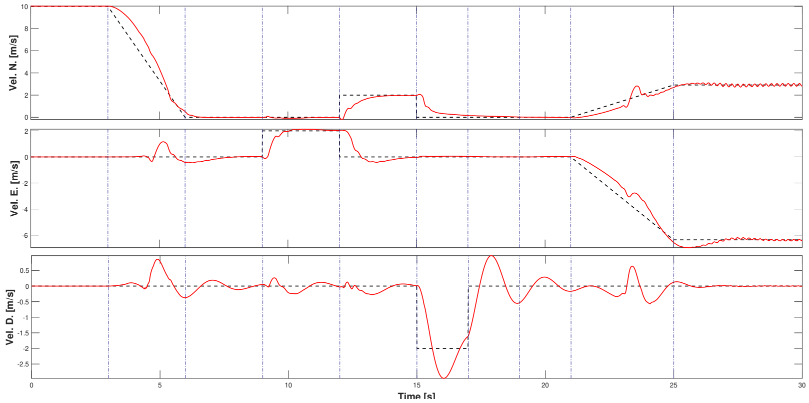


Figure 4.9: Inertial velocity for composite maneuver. Dashed line corresponds to the reference trajectory.

can be appreciated on the close-up figure.

Finally, at $t = 2, s$ the aircraft increases its velocity with constant acceleration until reaching $V = 7\,m/s$. The direction of this motion is in the opposite direction of $\hat{\mathbf{h}}$ and, initially, the UAV achieves this motion by tilting backwards. As the reference velocity increases, and the aircraft levels, it transitions back into the horizontal attitude reference. This transition introduces a discontinuity in the form of a large angular error about the thrust axis, since the reference commands the aircraft to switch from inverted flight. As seen from the 2D position plots, a transient position error is introduced due to this motion, but performance is not greatly impacted.

The simulation demonstrates that a single control structure with a single set of control gains is capable of following such a complex maneuver. There are, however, steady state errors present. During hover, these errors are small, but they increase as the reference velocity increases. Errors remain bounded, and the system remains stable throughout, but the aircraft lags behind when the reference position evolves rapidly.

4.3.1.2 Contracting spiral

The next maneuver consists of an initial straight line at $V = 10\,m/s$, followed by a climbing, contracting spiral maneuver. This maneuver can be generated through

$$\mathbf{p}_r = \begin{bmatrix} (r_0 + \dot{r}(t - t_0))\cos\left(\frac{2\pi}{t_r}(t - t_0) - \frac{\pi}{2}\right) \\ (r_0 + \dot{r}(t - t_0))\sin\left(\frac{2\pi}{t_r}(t - t_0) - \frac{\pi}{2}\right) + r_0 \\ v_z(t - t_0) \end{bmatrix} + \mathbf{p}_{r_0},$$

where $r_0 > 0$ is the initial radius, $\dot{r}$ is the constant radius contraction or expansion rate, t_r is the time needed to complete a rotation, and v_z is an independent climb rate. This profile can be differentiated to obtain smooth velocity and acceleration references. The profile shown in Figure 4.10 is constructed by setting $r_0 = 15\,m$, $\dot{r} = -0.4286\,m/s$, $v_z = -0.5\,m$, and $t_r = 10\,s$. The spiral is configured such that the radius contracts to zero after 35 seconds of initiating the maneuver.

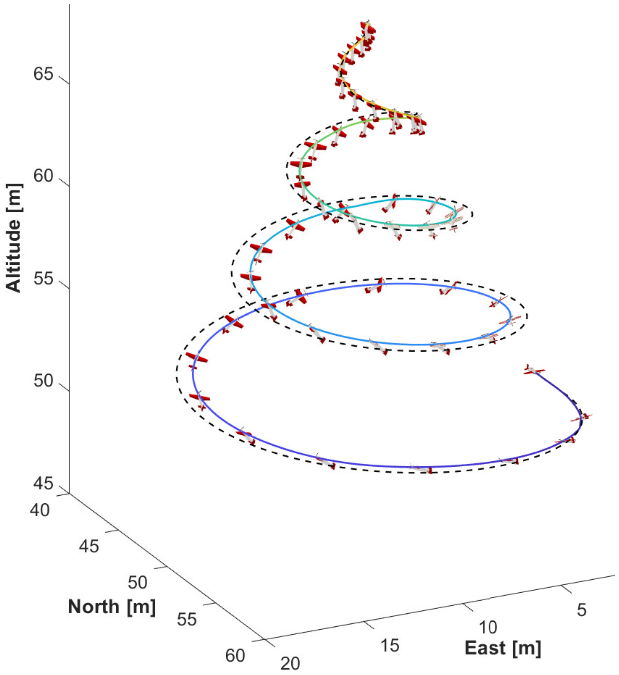


Figure 4.10: 3D view of contracting spiral maneuver.

Two controller features can be appreciated from the 3D view of the maneuver. First, the aircraft banks towards the center of the trajectory, as expected for the normal operation of a fixed-wing. This due to the aircraft lagging behind the time-varying reference, as can be better appreciated in the position plot shown in Figure 4.11, which results in the aircraft's heading being less than the reference heading, inducing a roll command.

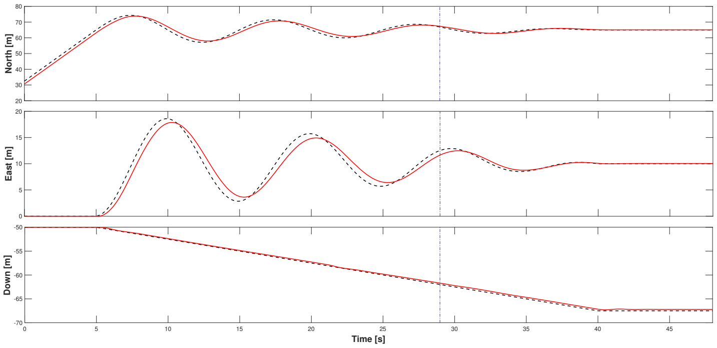


Figure 4.11: Inertial position for contracting spiral maneuver. Dashed line corresponds to the reference trajectory.

The second feature is, again, the controller capability of transitioning from steady to hover regimes, which occurs smoothly. The instant this transition is triggered is shown with a vertical dot and dashed line in the position plot and the error plot, shown in Figure 4.12. Since the maneuver slows down gradually, the transition has little impact on tracking performance.

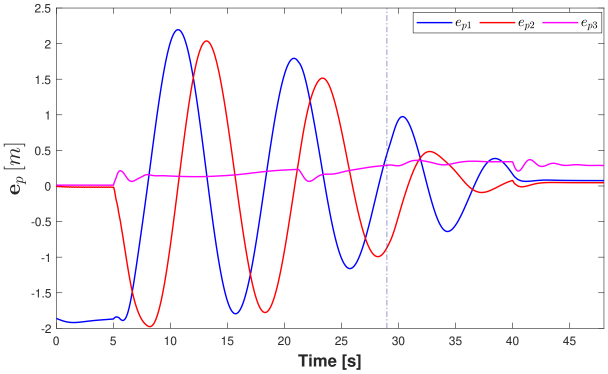


Figure 4.12: Position error for contracting spiral maneuver.

The error plot again shows there exist steady state tracking errors, which are larger when the maneuver evolves faster. Even with these errors, system stability is maintained throughout the maneuver. As the UAV transitions to a hover, the steady state errors tend to zero.

4.3.1.3 Position tracking with aerobatics

The final simulation tests the idea of commanding aerobatic maneuvers while following a trajectory, through an independently prescribed rotation about the thrust axis. The simulation commands the aircraft to transition from a straight line into a circular, constant altitude reference evolving at $V = 9.5\,m/s$. As opposed to the previous maneuver, the aircraft is commanded to rotate about the thrust axis at a rate of 3 rad/s. No aerodynamic force estimate is given during this stage. Figure 4.13 shows a 3D view of the maneuver where, for visual clarity, only the first rotation is shown. The maneuver, however, is performed for several rotations, as can be seen from the position error plot shown in Figure 4.14.

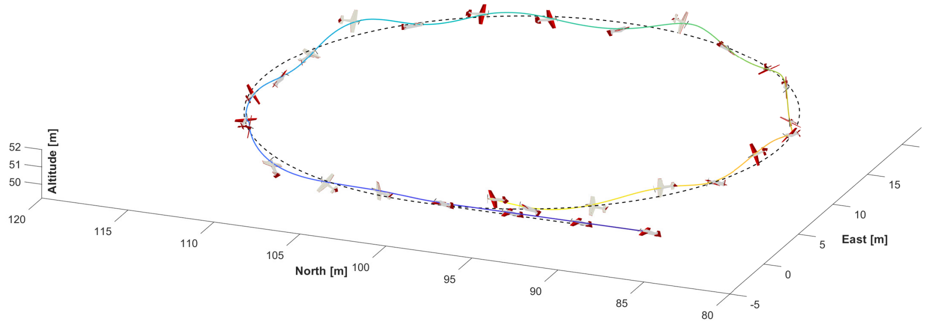


Figure 4.13: 3D view of Rolling Harrier on a circle maneuver.

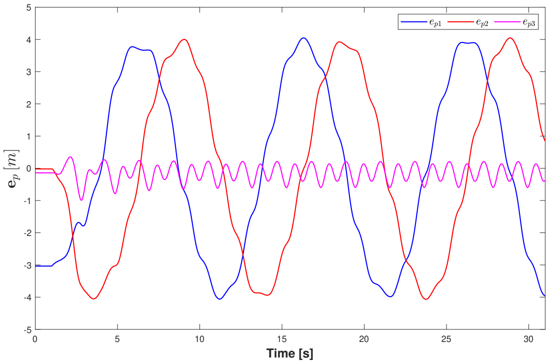


Figure 4.14: Position error for Rolling Harrier on a circle maneuver.

The 3D view of the maneuver shows the aircraft is still able to follow the reference trajectory, even when spinning at a fast rate. However, the unsteady aerodynamics caused by this rotation introduce oscillations in the path traced by the UAV. When compared to the start of the previous experiment, which initializes with the same radius, this maneuver exhibits a larger error. This is a combination of the aircraft being unable to bank to improve performance during a turn, and the rotation adding a significant disturbance to the system.

Still, this simulation indicates a measure of robustness of the proposed control system to disturbances created by aerodynamic effects. Most of the error comes from the aircraft lagging behind the time reference trajectory, but its altitude error remains small and the aircraft is never at risk of becoming unstable.

4.3.2 Flight experiments

Representative experimental flights were conducted over different days, at an outdoor field with varying degrees of wind. Gains and other controller parameters are given in Table 4.2; these are kept constant throughout all maneuvers. The gains used in simulation had to be adjusted for experiments; they initially proved too large and resulted in oscillations. The largest change lies with the proportional gains, whose effect is dominant in the hover regime where UAV inertial velocity is small. Still, the gains remain within the same order of magnitude as those used in simulation and follow the same pattern, where the third coordinate uses a smaller derivative gain but larger proportional and integral components. The threshold ξ_l used in simulation proved to be too conservative in practice, and a lower value of $\xi_l = 10°$ was used which resulted in more agile transitions.

Table 4.2: Controller parameters for experiments.

Param.	Value	Units
$\mathbf{K}_a$	diag((0.1656, 1.022, 0.6776))	$\frac{N\,m}{rad}$
$\mathbf{K}_\omega$	diag(0.0706,0.7576, 0.7736)	$\frac{N\,m\,s}{rad}$
$\hat{V}_\delta$	12	m/s
$\mathbf{K}_p$	diag(0.3982, 0.3982, 0.8479)	$1/s^2$
$\mathbf{K}_d$	diag(0.3853, 0.3853, 0.1422)	$1/s$
$\mathbf{K}_i$	diag(0.003,0.003,0.0015)	$1/s^3$
k	10	$m\,s$
k_y	0.084	1
k_{ϕ_p}	0.756	1
k_{ϕ_i}	0.01	s
ξ_u	20	deg
ξ_l	10	deg

4.3.2.1 Straight line descent

In conventional fixed-wing control, landing requires careful control of airspeed to avoid stalling as the aircraft slows down for the approach. This clashes with a position tracking objective, precluding precise position control from being used in this maneuver. In [106], we proposed that an agile fixed-wing could perform a precise landing maneuver through the developed position tracking controller. We now demonstrate this through a time-parametrized straight line descent. For this maneuver, the reference trajectory commands the aircraft to descend at an angle of 35 degrees, at a reduced speed of 6 [m/s]. The maneuver was performed in mild wind conditions, with mean winds between 2 and 4 [m/s]. Figure 4.15 shows the results of the 3D UAV path, and Figure 4.16 shows the inertial position of the aircraft.

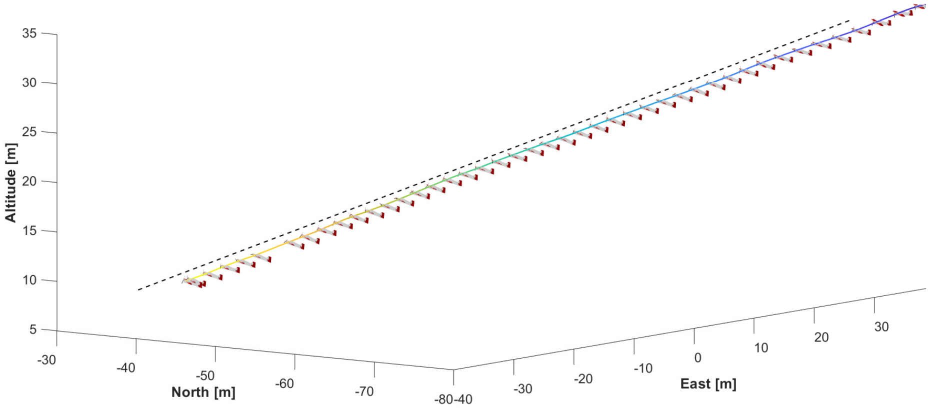


Figure 4.15: 3D view of straight line descent.

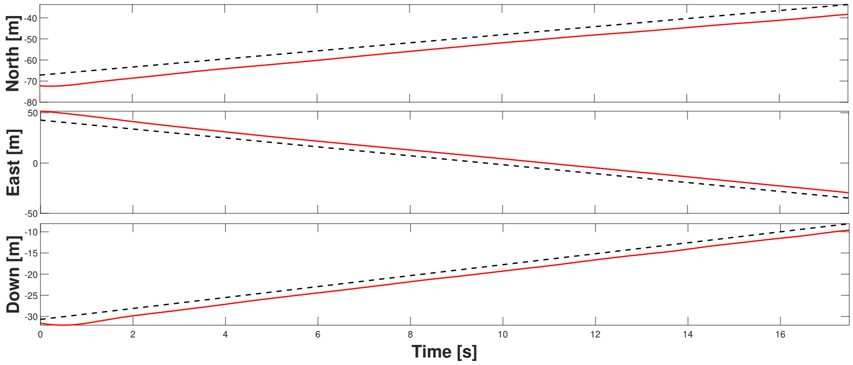


Figure 4.16: Inertial position for straight line descent maneuver. Dashed line corresponds to reference trajectory

With the proposed control system, the UAV is capable of following a time-parametrized path effectively. The UAV follows the trajectory, which is a steep descent, at a velocity where a conventional fixed-wing would stall, and under the effects of wind disturbances. While small, wind velocity is still a significant percentage of the commanded ground speed, but the aircraft is still able to follow the strict time-parametrized trajectory without any airspeed measurements, even if constant steady-state errors are present.

4.3.2.2 Transition into hover

We show the unified control strategy is capable of handling both conventional fixed-wing flight and hover through a maneuver transitioning through both regimes. Indeed, transition to and from hover has been one of the key challenges in hover capable fixed-wing aircraft. Here, we achieve this transition through a simple slow-down maneuver, where the aircraft is commanded to go from eight to zero m/s in two seconds. Once the velocity is zero, the aircraft is commanded to hold position indefinitely. The UAV trajectory through this maneuver is shown in Figure 4.17, and the position plots are shown in Figure 4.18.

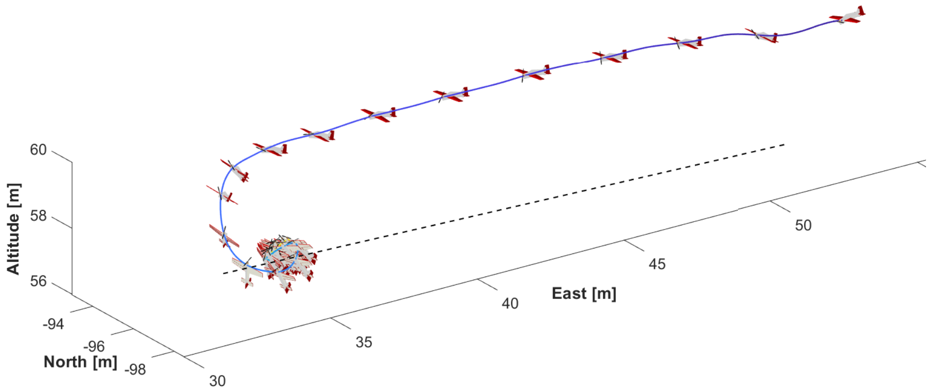


Figure 4.17: 3D view of transition into hover.

The maneuver is conducted under moderate wind conditions with mean winds below 5 m/s. The effects of crosswind can be clearly seen in the steady state portion of the maneuver, with an initial lateral error of about 4 m/s. This initial error, in addition to the smaller altitude error can be attributed to an imperfect aerodynamic force estimate while in the steady regime. This error can be diminished with an increased integral gain, and also through better tuning of the roll control system. Still, the aircraft follows the trajectory closely, and the transition into hover is remarkably smooth. As the difference between the estimated lift decreases due to the reduced ground speed, the altitude error tends to zero.

During the hover, the aircraft is not perfectly vertical, as the position errors force it to point into the wind. Its position oscillates in response to the wind, but it always remains

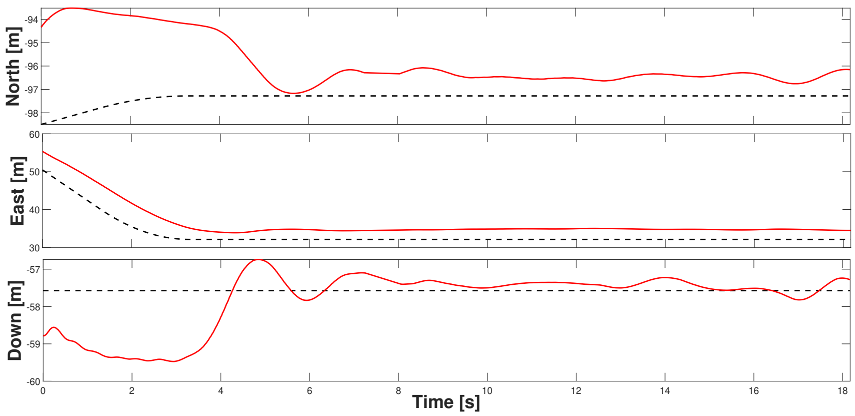


Figure 4.18: Inertial position for transition into hover maneuver. Dashed line corresponds to reference trajectory

stable and close to the target hovering position.

4.3.2.3 Loiter circle

A common steady regime maneuver is the circular loiter, where the UAV holds altitude as it traces a circle. We use this maneuver to test the performance limitations of using a trajectory tracking algorithm for high velocity, steady-state maneuvers under external wind disturbances. We command a wide circle of 25 metres radius, at a constant velocity of 12 m/s. The experiment was conducted under moderate wind conditions, with mean wind of approximately 5 m/s. A top view of the trajectory traced by the UAV is shown in Figure 4.19, and the position plots in Figure 4.20.

The 2D position plots show the control system does a good job at following a time-parametrized target, but when observing the path the aircraft traces, it is clear it does not trace the intended geometric path accurately. The UAV has no measure of wind, and due to the wide circular path, the airspeed is different between segments of the manuever. This can be seen from the third inertial component, which shows the aircraft has a small, but periodic loss in altitude due to a mismatched estimate.

In addition to the wind, the UAV cannot perfectly trace the circular path because it has no knowledge on the intended path; the control system is only tasked with chasing a virtual target with instantaneous position and velocity. While performance can be improved with different gain selection, this result motivates the research presented in the following chapter, which addresses the notion of giving priority to the geometric path, as opposed to an instantaneous position.

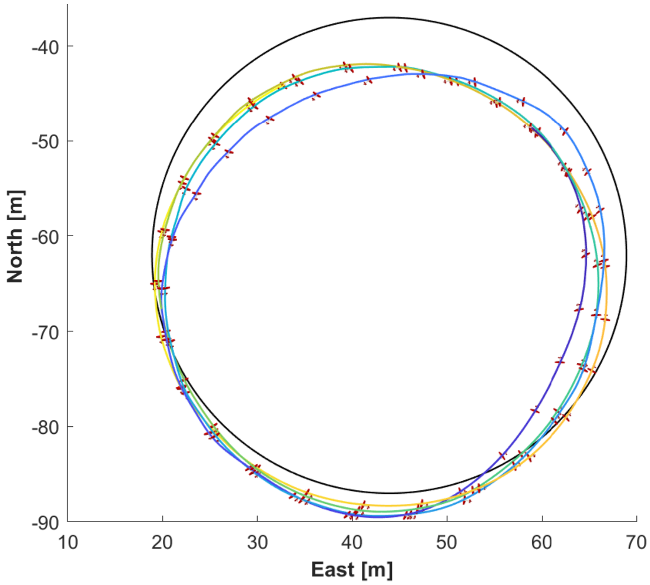


Figure 4.19: Top view of loiter maneuver.

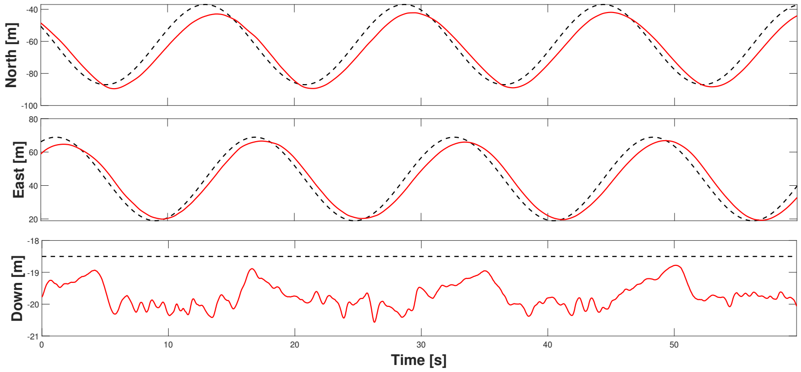


Figure 4.20: Inertial position for loitering maneuver. Dashed line corresponds to reference trajectory

4.3.2.4 Rolling Harrier on a circle

We show the proposed aerobatic capabilities through an experiment that mirrors the simulation of the rolling Harrier while following a circle. A 15 metres radius circle is commanded, at 10 m/s velocity. Simultaneously, a rotation about the thrust axis of $\dot{\phi}_r = 5$ [rad/s] is commanded. Wind conditions are similar to those of the previous experiment. The top and side view of the maneuver are shown in Figure 4.21. Instead of the position comparison, the position error is shown in Figure 4.22.

The diminished performance first observed in the equivalent simulation is more apparent. The UAV lags behind the target position significantly, which causes the circle to be much smaller that the intended one. Still, theses errors remain bounded, and the aircraft exhibits

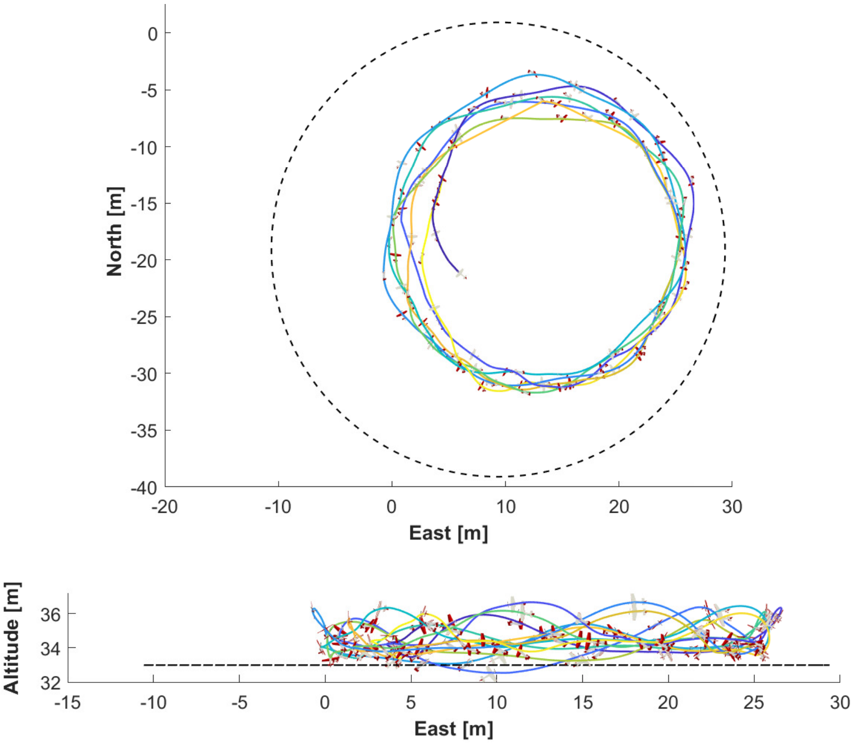


Figure 4.21: Top and side views of rolling Harrier on a circle.

stable operation throughout. The maneuver is exceedingly demanding, yet the aircraft holds altitude remarkably well and is able to trace a circle as intended, albeit with a different in radius.

4.3.3 Results discussion

Simulations and experiments show the proposed control system allows the agile fixed-wing to follow time-parametrized inertial trajectories, ranging from traditional steady flight paths, to maneuvers impossible for conventional aircraft. Remarkably, every maneuver is achieved through a single, unified control architecture.

The presented maneuvers, both simulated and experimental, underscore an additional advantage: the ease of trajectory generation. The control system is shown to perform well with simple, stitched-together trajectories, in contrast to other proposed solutions which require precise, optimized trajectories to perform. The control system requires only position references, and one optional rotation, not a complete, matched attitude and position profile. This is exemplified by the transition to hover, which is easily achieved by making the UAV slow down. A better, optimized trajectory, accounting for system limitations, could yield better results, and would be worth studying.

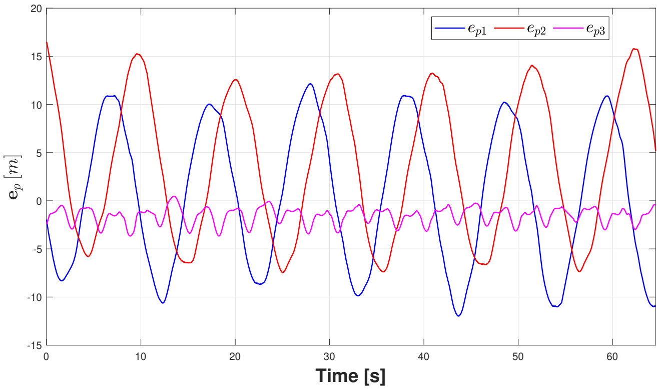


Figure 4.22: Position error for transition rolling Harrier on a circle maneuver.

Finally, the steady-state maneuvers, show the controller struggles with high-velocity trajectories. While it does not risk becoming unstable, the aircraft not tracing the correct path can be unacceptable in some application. This problematic, which obeys well known performance limitations for the tracking problem [55] is addressed in the following chapter.

Chapter 5

Path Following Control

This chapter addresses the problem of following a geometric, 3D path, as opposed to tracking a time-parametrized trajectory. The motivation for this development is twofold. First is the intuitiveness of operation for the envisioned pilot-assist, semi-autonomous system. When piloting a fixed-wing UAV, the *intent* of the remote operator is typically spatial geometry, as opposed to instantaneous position; a notion that will be further developed in the following chapter. Second, the experiments show the limitations of the tracking control system for high velocity references, where the aircraft lags behind the instantaneous position reference.

To illustrate this problem, we present the results of an outdoors flight experiment where a combination of external disturbances, a strict trajectory, and poor initial conditions result in undesirable performance. We command a circle of 15 metre radius, at a constant velocity of 10 m/s. Mean wind velocity is approximately 4 m/s, roughly in the direction of true North. As opposed to the experiment presented in the previous section, where the UAV smoothly transitions into the commanded circle, here, the UAV is flying at its maximum velocity of 14 m/s when the circle maneuver is suddenly commanded. Figure 5.1 shows the top view of this experiment, where various positions of the real UAV and its corresponding time-parametrized reference are given.

As can be seen from the first rotation, the UAV is off the path initially. As the aircraft rotates following the reference, wind pushes it northward, away from the path. The position reference continues evolving, and the UAV, displaced by the wind, eventually catches up to it. This causes the UAV to reduce its velocity, increasing the effect of wind disturbance. As can be seen from initial points of the second and third rotations, the reference then moves ahead of the UAV, which accelerates in an effort to reach it. This behaviour repeats at every rotation, eventually resulting in the aircraft tracing a circular path, but smaller, and shifted,

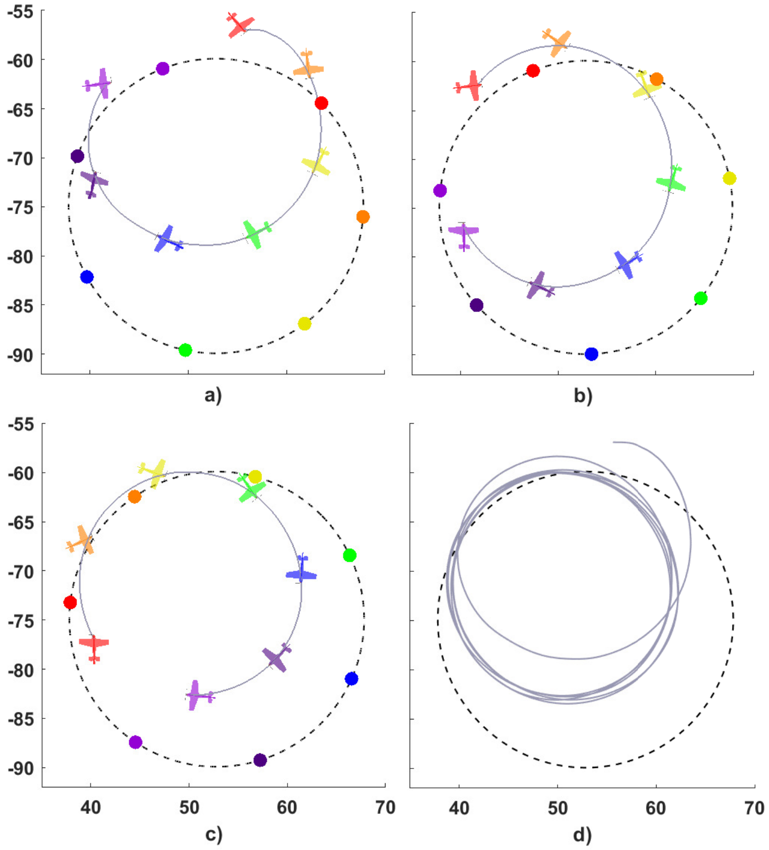


Figure 5.1: Effects of tracking error. Vertical axes are North, horizontal axes are East, in metres. a) First rotation. b) Second rotation. c) Third rotation d) UAV traced path.

from the one traced by the reference.

Staying on a path is something the control system, in its current form, cannot achieve: it is unaware that a path exists to begin with. The controller is only concerned with instantaneous, position references, not shapes in space. A mechanism is needed to convey this path information, and to enable the controller to prioritize being on this path instead of simply reaching an instantaneous position target.

5.1 The geometric task

As presented in [53], the objective of motion control can be separated into two distinct tasks. First is the geometric task, which consists of converging to a desired geometric path, and following it thereafter. In contrast, the dynamic task requires the vehicle to satisfy a stricter time-parametrized assignment, such a desired velocity along the path. The tracking problem attempts to achieve both objectives simultaneously: if $\mathbf{p}_i \to \mathbf{p}_r(t)$ and $\mathbf{v}_i \to \mathbf{v}_r(t)$ then, surely, the vehicle will have converged to the path while moving at the desired velocity along it.

However, if the dynamic convergence is imperfect, the geometric task will be compromised. The path following control paradigm decouples the two objectives, and focuses on achieving the geometric task.

Two modifications are needed to enable path following functionality. First, the strict, time-dependent evolution of the reference must be relaxed, since $\dot{\mathbf{p}}_r(t) = \mathbf{v}_r(t)$ can be considered an independent dynamic system that evolves in open loop. Then, a metric of distance with respect to the path, as opposed to an instantaneous point, needs to be defined. The former will be achieved by redefining the position reference a curve in 3D space and the latter through the definition of a moving path frame attached to this curve.

5.1.1 Curves in space

A spatial curve, $M(\sigma)$, is a collection of points in space, parametrized not by time, but by the parameter σ. Each value of σ has a uniquely assigned point that belongs to $M(\sigma)$. Let $\underset{\rightarrow}{\mu}(\sigma)$ be the physical vector pointing from some arbitrary origin, O, to the point in the $M(\sigma)$ at the corresponding value of σ. For convenience, this arbitrary origin is the same origin as the one previously defined for the definition of $\mathbf{p}_i$. Consider the previously defined inertial frame, $\boldsymbol{\mathcal{F}}_I$. Then, we define the reference path as the components of $\underset{\rightarrow}{\mu}(\sigma)$ resolved in the inertial frame,

$$\mathbf{p}_r(\sigma) = \boldsymbol{\mu}(\sigma). \tag{5.1}$$

With this new definition, the reference is a function of the parameter σ, whose evolution with respect to time, $\dot{\sigma}$ can be assigned independently, relaxing the time-dependence of the reference.

The velocity of $\underset{\rightarrow}{\mu}$ with respect to the inertial frame is given by

$$\frac{d}{dt}\underset{\rightarrow}{\mu}(\sigma) = \dot{\sigma}\frac{d}{d\sigma}\underset{\rightarrow}{\mu}(\sigma) \triangleq \dot{\sigma}\,\underset{\rightarrow}{\mu}'(\sigma) \triangleq \dot{\sigma}\nu(\sigma)\underset{\rightarrow}{T}, \tag{5.2}$$

where $\nu(\sigma) = \left\|\underset{\rightarrow}{\mu}'(\sigma)\right\|$ is the speed of the curve, and

$$\underset{\rightarrow}{T} = \frac{\underset{\rightarrow}{\mu}'(\sigma)}{\left\|\underset{\rightarrow}{\mu}'(\sigma)\right\|} \tag{5.3}$$

is the tangent vector to the curve [73]. These vectors are shown in Figure 5.2. If the speed of the curve is equal to one throughout, the curve is considered unit speed.

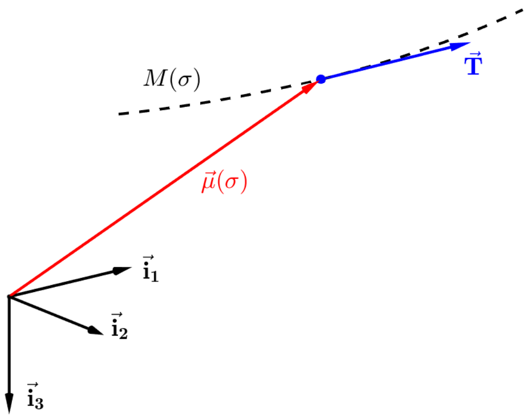


Figure 5.2: Vector definitions for a curve in space.

The components of the tangent vector, resolved in the inertial frame, are given by

$$\mathbf{T} = \mathbf{T}(\sigma) = \frac{\boldsymbol{\mu}'(\sigma)}{\|\boldsymbol{\mu}'(\sigma)\|} = \frac{\boldsymbol{\mu}'(\sigma)}{\nu(\sigma)}, \tag{5.4}$$

where the explicit dependence on σ will be omitted when clear by the context.

5.2 The gravity normal frame

An important differential geometry concept [107], the construct of a moving coordinate frame attached to a curve has often been used in motion control problems. In this work, we use it to derive path-relative position errors, in a general manner that can apply to different paths. Attaching a frame to path implies defining three basis vectors. Since the tangent vector is well defined for any smooth curve, it is always the natural first component.

However, there is no natural way to determine a vector orthogonal to the tangent [108] and, much like the attitude determination problem of the previous chapter, at least one arbitrary assignation is needed to complete the frame. This is a fundamental decision that will simultaneously endow the frame with specific properties and introduce limitations. Since the path relative errors will be determined by these additional orthogonal vectors, this decision will also determine the properties of the path-relative error metric.

In motion and navigation applications, the direction of an error is important. This is especially true for UAVs, platforms with distinct longitudinal and lateral modes, where an altitude error will require different compensation mechanisms than a lateral error. In terms of describing a curve in space, its attitude with respect to an inertial frame is as important as its shape. In view of this, we developed the gravity normal frame [109] to meet two criteria useful for the application of navigation and motion control in 3D.

- The definition of the path frame must be valid for most paths an autonomous vehicle may encounter, regardless of curvature or torsion.
- The resulting path frame must naturally enforce a useful distinction between path-relative errors, sensitive to the path's attitude with respect to the inertial frame.

The applications of the proposed path frame extend beyond fixed-wing UAVs. Indeed, the gravity normal frame is a useful tool that can be used for the general problems of 3D motion control and path planning. As will be made clear through its properties, the proposed frame is especially well suited for navigation tasks, where the orientation of a 3D curve with respect to navigation frame is as important as its shape.

5.2.1 Basis vector definition

Consider a path defined by $\boldsymbol{\mu}(\sigma)$, at least twice differentiable, and whose first derivative with respect to σ, $\boldsymbol{\mu}'(\sigma)$, is contained, $\forall \sigma \in \mathbb{R}$, in the set

$$\mathbb{B}_{\mu} := \{\boldsymbol{\mu}'(\sigma) \in \mathbb{R}^3;\ \|\boldsymbol{\mu}'(\sigma)\| \neq 0;\ \mathbf{k}^{\times}\boldsymbol{\mu}'(\sigma) \neq \mathbf{0}\}. \tag{5.5}$$

The gravity normal reference frame attached to $\boldsymbol{\mu}(\sigma)$ is $\boldsymbol{\mathcal{F}}_G = \{\underrightarrow{T},\ \underrightarrow{H},\ \underrightarrow{P}\}$, where

- The tangent vector, resolved in the inertial frame, $\mathbf{T} = \begin{bmatrix} T_1 & T_2 & T_3 \end{bmatrix}^T$, is calculated through (5.4).
- The horizontal vector, resolved in the inertial frame is given by

$$\mathbf{H} = \frac{1}{\sqrt{T_1^2 + T_2^2}} \begin{bmatrix} -T_2 \\ T_1 \\ 0 \end{bmatrix} \tag{5.6}$$

- The projected vector, resolved in the inertial frame, given by

$$\mathbf{P} = \frac{1}{\sqrt{T_1^2 + T_2^2}} \begin{bmatrix} -T_1 T_3 \\ -T_2 T_3 \\ T_1^2 + T_2^2 \end{bmatrix} \tag{5.7}$$

The gravity normal frame borrows the geometric properties of vector-projection algorithm of the control system, as seen from the resulting basis vectors. The horizontal vector, $\underset{\rightarrow}{H}$, always lies on the horizontal, NE plane, regardless of the direction of the tangent. Meanwhile, the projected vector, $\underset{\rightarrow}{P}$, is the projection of the direction of gravity, $\underset{\rightarrow}{g} = \underset{\rightarrow}{i_3}$, onto the plane normal to the tangent, Π. These vector relationships can be appreciated in Figure 5.3.

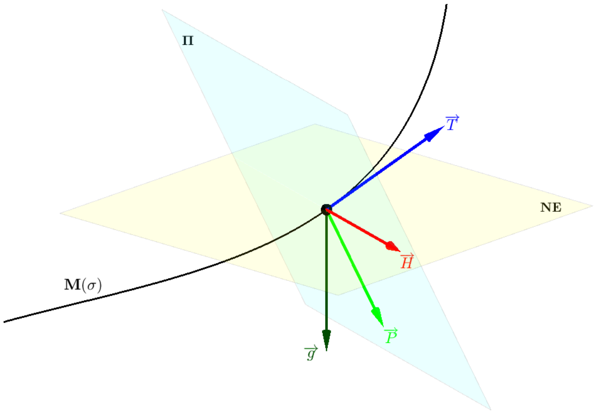


Figure 5.3: Basis vectors of the gravity normal frame.

5.2.2 Frame properties

The gravity normal frame is constructed completely through the tangent vector, it does not require additional derivatives of the path, like the commonly employed Frenet-Serret frame [107]. This dependence introduces two well-known limitations to the Frenet frame: it ill-defined when curvature vanishes, for example, on a straight line path, and its second and third vectors exhibit discontinuous jumps at inflection points, even if the curve is smooth. While these problems can be circumvented in planar curves through a Darboux frame, no solution exists for 3D curves or piece-wise continuous curves in different planes. In contrast, the gravity normal frame remains well defined and continuous throughout. This behaviour can be appreciated on the piece-wise continuous curve shown in Figure 5.4.

The curve in Figure 5.4 is composed of different planar paths stitched together. It can be seen that, while the Frenet frame jumps discontinuously at each change in curvature, the gravity normal frame remains well defined througout, and no additional correction steps are needed. This Figure also shows a third problem with the Frenet frame, that obeys an application consideration, as opposed to a mathematical singularity. The normal, $\underset{\rightarrow}{N}$, and binormal, $\underset{\rightarrow}{B}$, vectors of the Frenet frame do not necessarily capture a distinct lateral and

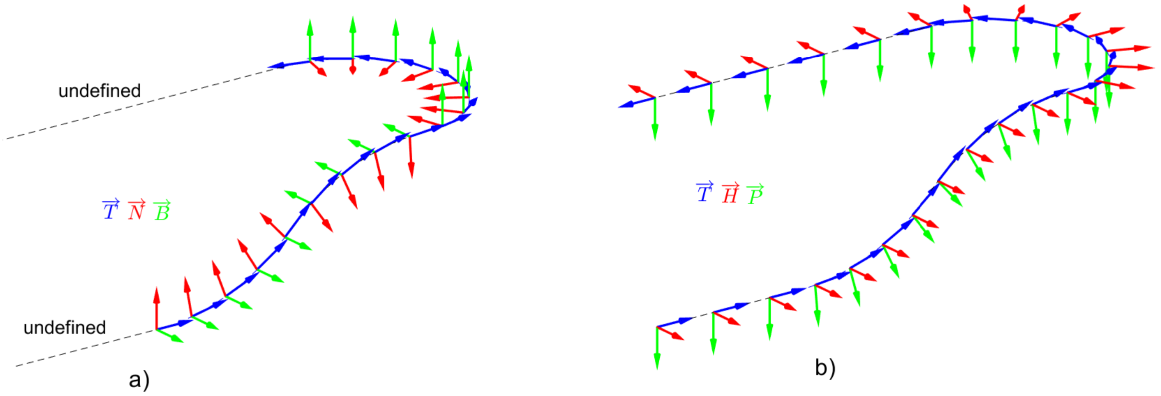


Figure 5.4: Comparison of the a) Frenet and b) Gravity Normal frames on a piece-wise continuous curve.

longitudinal direction, as they flip according to the path. This problem is worse in curves with non trivial torsion, since the normal and binormal vectors will each contain a combined error quantity, that changes at every point. The gravity normal frame does not suffer from this, as will be shown next.

The parallel transport frame, or Bishop frame, is another popular frame in the motion control literature [74, 110]. Its orthogonal vectors evolve smoothly throughout the path regardless of curvature, and can be arbitrarily initialized. A correctly initialized Bishop frame will evolve as the gravity normal frame on the piece-wise curve shown previously. This is because, for planar curves, the gravity normal frame is equivalent to a parallel transport frame with specific initial conditions [109]. However, for curves with non trivial torsion, the Bishop frame will add a "spin" about the tangent vector. This complicates stitching together different curves, as the initial conditions need to be managed carefully to avoid discontinuities. More importantly, it introduces the same implementation problem as the Frenet frame: since the normal vectors spin at every point, errors defined in the normal directions will lack any physical meaning. This behaviour, and its comparison with the gravity normal frame, is shown in Figure 5.5 for a helix, a path with constant torsion.

As seen from Figure 5.5, the Bishop frame evolves smoothly throughout, but, unlike the gravity normal frame, its normal vectors, $\underset{\longrightarrow}{N_1}$ and $\underset{\longrightarrow}{N_2}$ rotate about the tangent as the curve evolves. This is not a problem for the task of describing the curve, but is detrimental for the application of motion control. The argument can be made that both the Frenet and Bishop frames (or hybrid combinations of both [108]) are strictly kinematic constructs and lack meaning in regards to the physical space which autonomous vehicles need to navigate. In

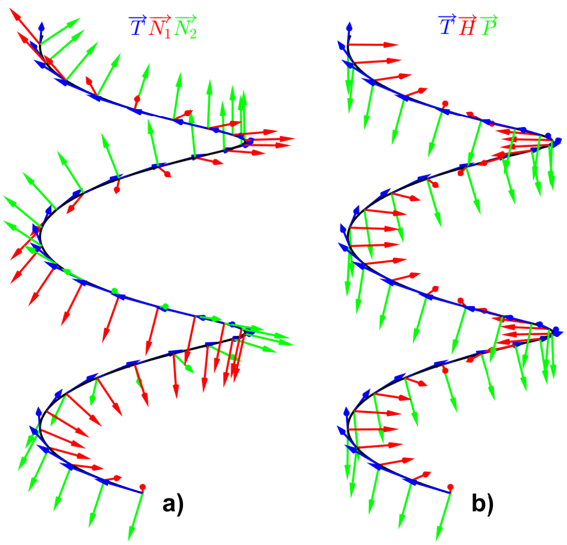


Figure 5.5: Comparison of the a) Bishop and b) Gravity Normal frames on a helix.

contrast, the gravity normal frame accounts for path orientation with respect to the inertial frame, and embeds this information in the resulting basis vectors.

5.2.2.1 Limitations of the gravity normal frame

The trade-off of the gravity normal frame is that it cannot represent every curve in 3D space, only curves that belong in the set $\mathbb{B}_\mu$. As opposed to a restriction on curvature, this obeys the same geometric singularity of the vector-projection algorithm, and disallows any strictly vertical tangent directions at any value of σ. While limiting, we consider it an acceptable trade-off for the purposes of vehicle navigation, especially when compared to limitations on curvature and torsion, since most conventional ground, aerial, and marine vehicles will not be required to perform a maneuver involving a vertical tangent vector.

5.2.3 Path DCM and angular velocity

The basis vectors of the gravity normal frame, resolved in the inertial frame, can be used as to construct the DCM corresponding to the relative attitude between the path frame $\boldsymbol{\mathcal{F}}_G$ and the inertial frame $\boldsymbol{\mathcal{F}}_I$. Define the DCM from inertial to path frame as

$$\mathbf{C}_{gi} = \begin{bmatrix} \mathbf{T} & \mathbf{H} & \mathbf{P} \end{bmatrix}^T . \tag{5.8}$$

This DCM will allow us to express position errors in terms of path frame coordinates.

Proposition 5.1. *The angular velocity of the gravity normal frame, with respect to the inertial frame, resolved in path frame coordinates is given by*

$$\boldsymbol{\omega}_g = \boldsymbol{\omega}_g^{gi} = \dot{\sigma}\boldsymbol{\lambda}(\sigma) = \dot{\sigma}\begin{bmatrix}\lambda_1(\sigma)\\ \lambda_2(\sigma)\\ \lambda_3(\sigma)\end{bmatrix} = \dot{\sigma}\begin{bmatrix}\frac{(\mathbf{k}_3^T\boldsymbol{\mu}')(\mathbf{k}^T(\boldsymbol{\mu}'^{\times}\boldsymbol{\mu}''))}{\|\boldsymbol{\mu}'\|\|\mathbf{k}^{\times}\boldsymbol{\mu}'\|^2}\\ \frac{(\mathbf{k}^{\times}\boldsymbol{\mu}')^T(\boldsymbol{\mu}'^{\times}\boldsymbol{\mu}'')}{\|\boldsymbol{\mu}'\|^2\|\mathbf{k}_3^{\times}\boldsymbol{\mu}'\|}\\ \frac{\mathbf{k}_3^T(\boldsymbol{\mu}'^{\times}\boldsymbol{\mu}'')}{\|\boldsymbol{\mu}'\|\|\mathbf{k}^{\times}\boldsymbol{\mu}'\|}\end{bmatrix} \tag{5.9}$$

Proof. See Appendix C.1 □

5.3 Path following controller

With the moving path frame defined, we can now properly state the path following control objective, and design the control law to achieve it.

5.3.1 Path relative error definition

The path relative-errors, for a given value of σ, is defined as

$$\mathbf{e}_{pr} = \begin{bmatrix}e_s\\ e_c\\ e_h\end{bmatrix}^T = \mathbf{C}_{gi}(\mathbf{p}_i - \boldsymbol{\mu}(\sigma)), \tag{5.10}$$

where e_s is the on-track error, e_c is the cross-track error, and e_h is the altitude errors.

If we use the redefined position reference (5.1), then

$$\mathbf{e}_{pr}(\sigma) = \mathbf{C}_{gi}(\mathbf{p}_i - \mathbf{p}_r(\sigma)) = \mathbf{C}_{gi}\mathbf{e}_p(\sigma),$$

and, more importantly,

$$\begin{aligned} e_s &= \mathbf{T}^T\mathbf{e}_p,\\ e_c &= \mathbf{H}^T\mathbf{e}_p,\\ e_h &= \mathbf{P}^T\mathbf{e}_p, \end{aligned}$$

showing the clear relationship between the three path-relative errors and the basis vectors of the gravity normal frame.

5.3.2 Path following problem statement

Consider the UAV dynamics given by (4.2), and a reference path given by some $\boldsymbol{\mu} \in \mathbb{B}_\mu$. Then, design a control law for the thrust force, T, a reference attitude $\mathbf{C}_{ri}$, required by the core attitude controller, and a path evolution policy, $\dot{\sigma}(t)$ such that $e_c \to \mathbf{0}$ and $e_h \to \mathbf{0}$ as $t \to \infty$.

For the path following problem, no objective is imposed on the convergence of the on-track error, which in this problem formulation represents the UAV lagging behind a virtual target while on the path.

5.3.3 Path following control law

To achieve the path following objective, we propose a control law using notions from [38], but designed to be compatible with the control architecture derived in the previous chapters. In this proposed framework, the path following functionality acts as an additional control loop built on top of the velocity control system (4.27). This feedback system will translate path relative errors into velocity references for the internal velocity control system and into a path evolution policy.

The benefit of this structure is, once again, compatibility. As an additional feature built on top of the velocity controller, path following functionality can be seamlessly activated or deactivated, returning the control system to velocity control functionality. More importantly, since the velocity control system and the position tracking system are built using the same structure, the UAV can easily transition between path following and position tracking functionality, a feature that will be exploited in the following chapter.

The path following control law is given by the following theorem.

Theorem 5.2. *Consider the UAV dynamics (4.7), the velocity controller (4.27), verifying the conditions of Theorem 4.5, and a reference path defined by a unit speed curve $\mu(\sigma) \in \mathbb{B}_\mu$. Define an approach direction*

$$\boldsymbol{\nu}_a = d_l\mathbf{T} - k_c e_c \mathbf{H} - k_h e_h \mathbf{P}, \tag{5.11}$$

with the path-relative errors defined in (5.10), and where k_c, k_h are positive gains and d_l is a strictly positive constant, akin to the look-ahead distance of guidance algorithms.

Consider a reference velocity given by

$$\mathbf{v}_r = \frac{V_c}{d_l}\boldsymbol{\nu}_a, \tag{5.12}$$

where $V_c > V_{min} > 0$ is a commanded desired speed along the path, lower bounded by a minimum operative velocity V_{min}, and a path evolution law calculated as

$$\dot{\sigma} = (V_c\mathbf{T} + k_s\mathbf{e}_{pr})^T\mathbf{T}, \tag{5.13}$$

for $k_s > 0$.

If the gains satisfy

$$\lambda_{min}(\mathbf{K}) > \frac{1}{4k_v(1-\sqrt{\psi})}, \tag{5.14}$$

for

$$\mathbf{K}_p = diag\{k_s,\, V_c\, k_c/d_l,\, V_c\, k_h/d_l\}, \tag{5.15}$$

and

$$4\lambda_{min}(\mathbf{W}_G)\lambda_{min}(\mathbf{W}_a) > \|\mathbf{W}_{Ga}\|_2, \tag{5.16}$$

where

$$\mathbf{W}_G = \begin{bmatrix} \lambda_{min}(\mathbf{K}_p) & -1/2 \\ -1/2 & k_v(1-\sqrt{\psi}) \end{bmatrix}, \qquad \left[\mathbf{W}_{Ga} = \begin{bmatrix} 0 & 0 \\ 2(k_p\sqrt{3}\nu + B_p) & 0 \end{bmatrix}\right], \tag{5.17}$$

then the system will verify Theorem 4.5 and, additionally, the cross-track and altitude errors will converge exponentially to zero.

Proof. See Appendix C.2. □

From the path evolution law (5.13), if, in addition to the cross-track and altitude errors, the on-track error converges to zero, then the velocity on the path will converge to the commanded velocity, V_c, effectively operating as a tracking control system. However, if this is not verified, the vehicle can still converge to the path.

In practice, the path-following control law shapes the externally prescribed command velocity to reach the path. The constant d_l effectively modifies the aggressiveness of the approach direction: at high values, the UAV will prioritize aligning itself to the path, instead of converging to it, resulting in a smoother, but slower, convergence. The value of k_s, meanwhile, will modify the speed at which the path evolves, allowing it to slow down or move ahead faster in closed loop with the UAV position. Small values will prioritize the velocity on the path assignment, to the detriment of the geometric task.

5.3.3.1 Rotation about the thrust axis

Since the path following controller is built around the velocity control system, which uses the vector-projection algorithm for attitude determination, the rotation about the thrust axis remains a free parameter. As before, one option is to prescribe a closed-loop roll command to improve performance. Using the differential geometry tools developed for the path-following solution, one small, but important modification to this command can be implemented to improve lateral performance.

Calculate the course command

$$\chi_c = \chi_r + \tan^{-1}(-k_y e_c), \tag{5.18}$$

where $k_y > 0$, and where e_c is the cross-track error. In implementation, it is better to use instead $\chi_c = \chi_r + \text{atan2}(-e_c, d_y)$, where $d_y = 1/k_y$ is a look-ahead constant that determines the convergence characteristics.

Then, calculate the roll command through the PI control law (4.43), and the obtain the roll augmented reference DCM, $\mathbf{C}_{ri}^{\phi}$, through (4.39).

The difference is only on the lateral error used to calculate the course command, χ_c. Where before this error was resolved in the body frame, as e_{b2}, here the error is relative to the path, through e_c. Using the gravity normal frame, this error is guaranteed to only contain a lateral deviation, which makes it a significantly more effective metric to command the roll. In fact, even if the UAV is operating in position tracking mode, using the cross-track error will give better results.

Alternatively, convergence to the path can be further prioritized by having the reference roll depend only on the cross-track error, through

$$\phi_r = k_{\phi_p} \tan^{-1}(-k_y e_c) + k_{\phi_i} \int_0^t \tan^{-1}(-k_y e_c(\tau)) d\tau. \tag{5.19}$$

Of course, the option to prescribe an arbitrary aerobatic maneuver is retained, but the UAV will instead follow a path while performing this maneuver, as opposed to tracking an instantaneous position target. Since aerobatic maneuvers, such as the rolling Harrier and the knife-edge, significantly modify the aerodynamic forces acting upon the aircraft, the path-following option can lead to better results if the interest is for the UAV to follow a geometry. This notion will be explored in the experiments section.

5.4 Controller verification

5.4.1 Numerical simulations

For the following simulations, the previous controller gains from Table 4.1 are kept, with additional gains for the path following control loop given in Table 5.1. The path following gains are comparatively simpler to determine. Whereas the selection of gains for the attitude and position/velocity control systems heavily depends on aircraft properties, the path following gains obey the behaviour of the closed-loop system. This notion is the motivation for the commonly used assumption of a kinematic model of the aircraft, which assumes the presence of underlying autopilot systems.

Table 5.1: Path following control parameters for simulation.

Param.	Value	Units
d_l	5	m/s
k_c	2	1/s
k_h	1	1/s
k_s	2	1/s

The simulation consists of an initial straight line at constant altitude, followed by a climbing helix of 15 metres radius, with a climb angle $\gamma = 20°$, followed by another straight line at constant altitude. The unit speed helix segment is built through

$$\boldsymbol{\mu}(\sigma) = \begin{bmatrix} r\sin\left(\frac{\sigma}{\sqrt{r^2+c^2}}\right) \\ d\,r\left(1-\cos\left(\frac{\sigma}{\sqrt{r^2+c^2}}\right)\right) \\ -c\frac{\sigma}{\sqrt{r^2+c^2}} \end{bmatrix}, \tag{5.20}$$

where r is the helix radius, $c = r\tan(\gamma)$ where γ is the climb angle, and $d \in \{-1, 1\}$ determines the turn direction.

The commanded velocity is prescribed as $V_c = 10$ m/s. To evaluate convergence to the path, the UAV is initialized 5 metres behind (South) and 10 metres to the right (East) of the initial point on the path. To test the path-following algorithm under ideal conditions, this initial simulation has no wind disturbance. The 3D view of the reference path, and the path traced by the UAV is shown in Figure 5.6. The path-relative errors are shown in Figure 5.7, and the evolution of the path parameter, $\dot{\sigma}$ is shown in Figure 5.8.

The 3D view shows the UAV converges to the path and follows it closely, and this can be

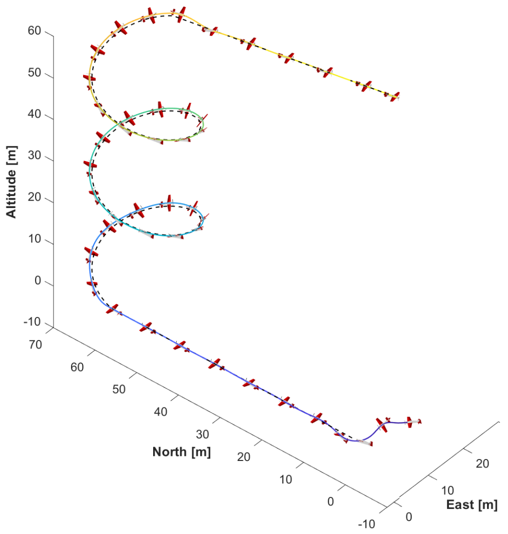


Figure 5.6: 3D view of path following simulation, no wind.

verified through the cross-track and altitude errors, given in Figure 5.7. On this figure, the on-track is close to zero for the straight-line segments, but grows to a constant value during the helix, showing the UAV lags behind the instantaneous position on the curve. However, as seen in Figure 5.8, the path following controller accounts for this lag by reducing the rate at which the path evolves, effectively slowing the maneuver to allow the UAV to remain on the path, which is the main objective. In the straight line segments, $\dot{\sigma}$ converges to the commanded velocity of 10 m/s, showing that, when the on-track error is small, the UAV verifies the dynamic task in addition to the geometric one.

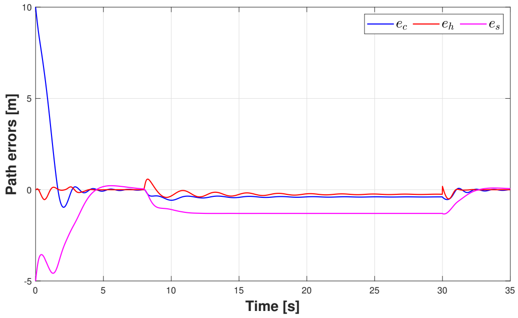


Figure 5.7: Path-relative errors for path following simulation, no wind.

If the gains k_c, k_h, k_s are zeroed, the system reverts to pure velocity control. This, in turn, allows for a simple transition to position tracking functionality. Not shown is an equivalent experiment using the position tracking control system under the same, no wind conditions. In this simulation, the UAV follows the path, but the strict time imposition introduces some additional cross-track and altitude errors. The difference is not significant, and both the

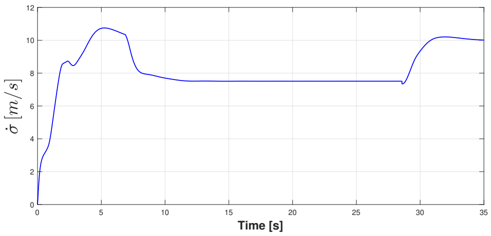


Figure 5.8: Path evolution law for path following simulation, no wind.

path following and trajectory tracking control strategies yield equivalent results. However, this is not the case when wind is considered.

Figure 5.9 shows the results of a simulation using the tracking control system, attempting to track the time-parametrized reference under a mean wind of 5 m/s in East direction.

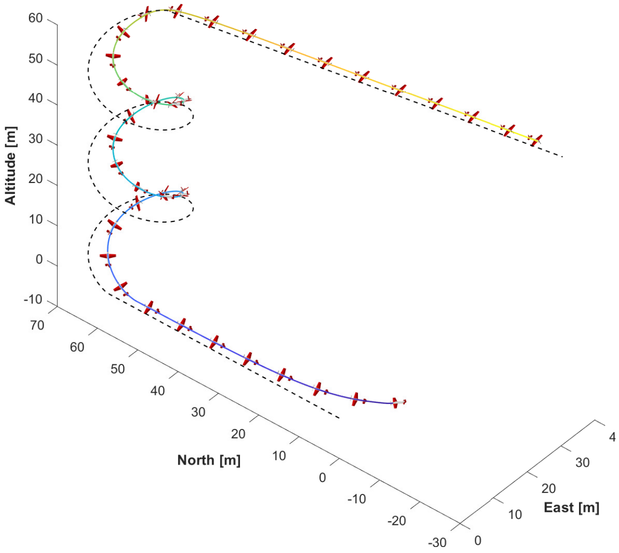


Figure 5.9: 3D view of tracking simulation, 5 m/s wind.

The 3D view makes it clear that the tracking control system struggles to navigate the path when in the presence of wind. While the error in the straight line segments is small, the UAV cannot adequately track the helix segment. Wind pushes the vehicle eastward, causing it to alternate between large and small tracking errors. This causes the UAV to trace a smaller circle, shifted east from the reference path; a comparable behaviour to the one observed in the experiment shown in Figure 5.1.

In contrast, consider a simulation with the same wind conditions, but with path following functionality enabled. The 3D view of the reference path and the path traced by the UAV during the simulation are shown in Figure 5.10.

It is clear from the 3D view that this strategy achieves better results than the tracking

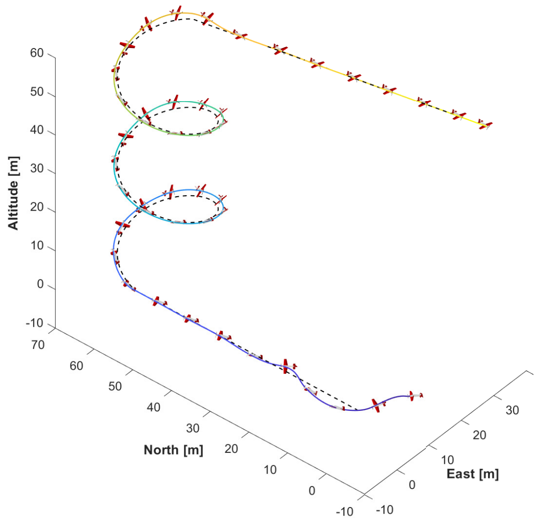


Figure 5.10: 3D view of path following simulation, 5 m/s wind.

control system, at least, in regards to reaching and following the reference path. Figure 5.11 shows the path-relative errors, and it can be appreciated that both the cross-track and altitude errors are small, meaning the UAV is on the geometric path. The biggest change is in the on-track error, which is allowed to increase during sections of the helix segment.

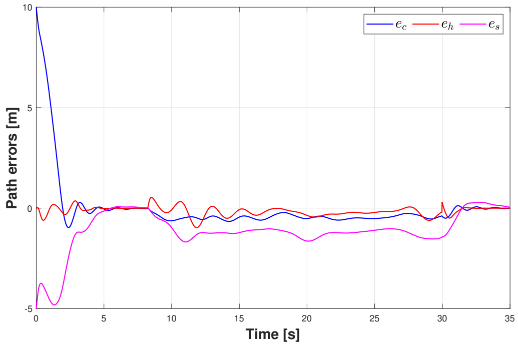


Figure 5.11: Path-relative errors for path following simulation, 5 m/s wind.

The improved performance of the path following control system is better explained by the path evolution law. Figure 5.12 shows $\dot{\sigma}$ throughout the simulation, together with the heading specified by the reference at every time step, calculated as $\chi_r = \text{atan2}(T_2, T_1)$.

For the straight line segments, $\dot{\sigma}$ eventually converges to the commanded velocity $V_c = 10$ m/s, but the path moves slower at first, allowing the UAV to reach the path. This is contrast to the tracking solution, where the virtual target evolves in open loop, regardless of aircraft states. The key section of the maneuver is the helix, where, as before, the path following controller determines the path needs to evolve slower in order for the UAV to stay close to the curve. When the reference heading reaches $\chi_r = pi/2$ [rad], the UAV experiences

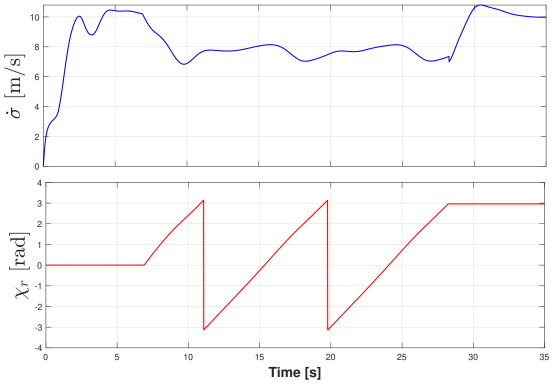


Figure 5.12: Path evolution law and reference heading for path following simulation, 5 m/s wind.

tailwind, which would push it into the path in the tracking control formulation. However, the path evolution compensates by slowing down the velocity reference, allowing the UAV to remain close to the path. The opposite is true when the reference heading creates a headwind condition.

The trade-off in performance during this maneuver is that the dynamic assignment is not met throughout, resulting in the UAV completing the path in more time than the tracking formulation would. However, the geometric objective is not compromised, whereas the tracking controller would alter the path. If the UAV is expected to follow a path, this is an acceptable trade-off.

5.4.2 Flight experiments

Perhaps the most representative experimental result is a comparison with the flight experiment that introduced this chapter.

Immediately after the conclusion of the original experiment which used the trajectory tracking solution shown in Figure 5.1, an equivalent experiment was performed using the path following control system. This was done to ensure, as much as possible, that wind conditions were equivalent between the two flight experiments to allow for a fair comparison. The experiment uses the control parameters given in Table 4.2 (which are the same ones used by the position tracking experiment) together with the path following control parameters shown in Table 5.2. Gains needed to be adjusted from simulation to implementation, but remain in the same order of magnitude. The look-ahead distance, d_l was increased, resulting in a smoother convergence to the path. Since the gains of the velocity tracking controller were reduced from those used in simulation, the reduction of the cross-track and altitude error gains is logical: the UAV cannot converge to the desired velocity as fast as in the ideal simulation, hence the reference velocity should be less aggressive. The biggest reduction lies with the on-track error gain. This gain is responsible for the UAV behaviour once it has reached the path; as it increases, the path following control system behaves more like a position tracking controller. During experiments, where wind is present, it was found that the ideal simulation value was too strict, and the UAV started prioritizing the dynamic task, instead of the geometric one, once it had converged to the path. By reducing this gain, the controller is again able to prioritize following the path, regardless of the instantaneous value of $\mu_(\sigma)$ while in the presence of external disturbances.

Table 5.2: Path following control parameters for experiments.

Param.	Value	Units
d_l	7	m/s
k_c	1	1/s
k_h	0.4	1/s
k_s	0.4	1/s

Once again, the UAV does not enter the circle smoothly, since this maneuver is commanded abruptly when the UAV is flying at its maximum velocity. The top view of this experiment, showing different positions of the real UAV and their corresponding reference points on the path, is given in Figure 5.13. This experiment uses the original roll command 4.43 based on a body referenced position error.

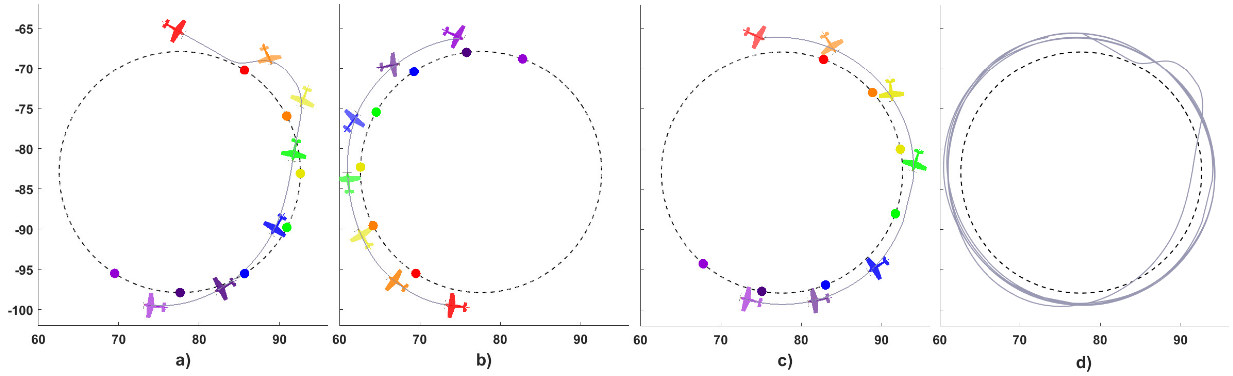


Figure 5.13: Path following on a circle. Vertical axes are North, horizontal axes are East, in metres. a) First rotation. b) Second rotation. c) Third rotation d) UAV Traced path.

This top view clearly shows that the path following strategy achieves better results when operating under equivalent wind conditions. The UAV is never ahead from the instantaneous reference position on the path during this experiment, even when experiencing significant side-slip on the left portion of the circle. It can also be appreciated that the path, as in simulation, slows down from the commanded velocity to achieve better convergence to the path.

The path-relative errors are shown in Figure 5.14. Both the cross-track error and the altitude error are small once the UAV has converged to the path, their magnitude remaining below 1.5 metres. The on-track error is larger, as the aircraft lags behind, however, this does prevent the geometric objective from being satisfied. In fact, the same behaviour first observed in simulation occurs in this experiment, where the on-track error increases and decreases in relation to UAV position on the circle.

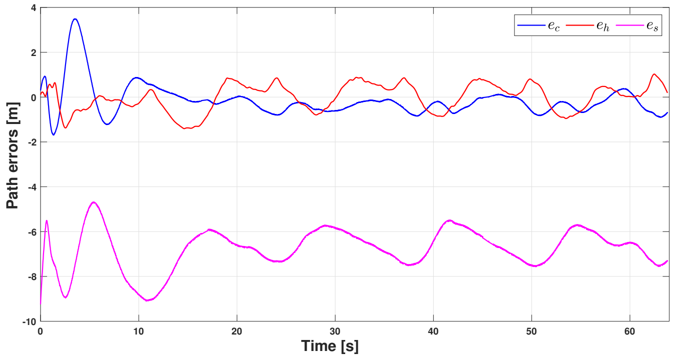


Figure 5.14: Path-relative errors for experimental circle path following.

To show the effectiveness of a better constructed banking command, an additional experiment is conducted. Since it is performed immediately after the previous test, it nominally operates under the same wind conditions. This experiment uses the same controller parameters and reference path as before, but uses the modified closed-loop roll command 5.19 based on the crosstrack error. The top view of this experiment is shown in Figure 5.15, and the path relative errors are shown in Figure 5.16.

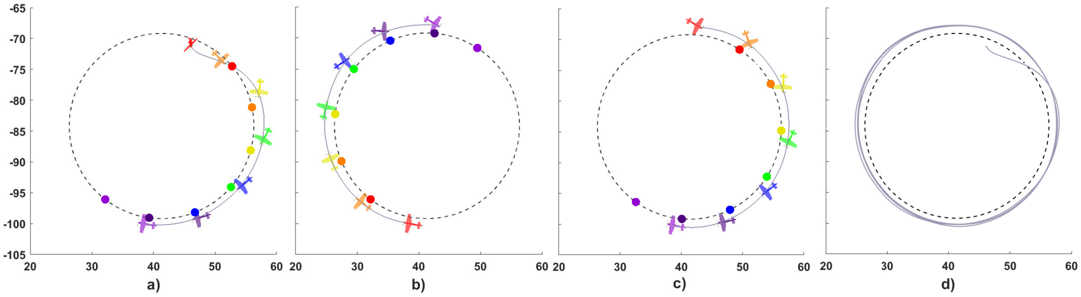


Figure 5.15: Experimental circle path following using modified roll command. Vertical axes are North, horizontal axes are East, in metres. a) First rotation. b) Second rotation. c) Third rotation d) UAV Traced path.

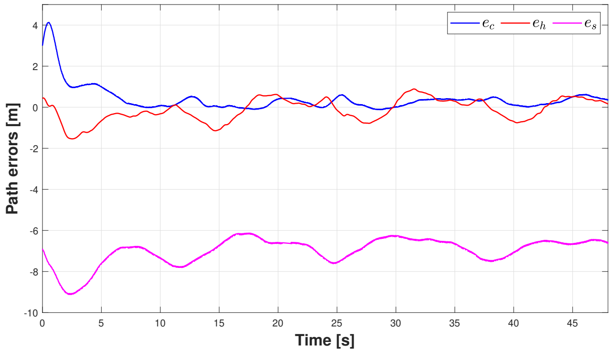


Figure 5.16: Path-relative errors for experimental circle path following.

Results show a the aircraft converges to the intended path more closely than the previous experiment, which used the original roll function. Once the UAV converges to the path, the altitude error remains below one metre, and the cross-track error shows an error of less that 0.5 metres, a remarkable feat considering the light-weight nature of the UAV, its susceptibility to wind, and the lack of an airspeed estimate.

Next, an experimental flight using a 3D curve with non-trivial torsion is presented. We build a descending helix for our reference path through (5.20), setting the radius to $r = 10$ metres, and the descent angle to $\gamma = -20°$. Controller parameters are the same as those from the previous experiments. Mean wind velocity is less than 4 m/s. Figure 5.17 shows the top and 3D view of the reference curve, and the path traced by the UAV. Path-relative errors are shown in Figure 5.18.

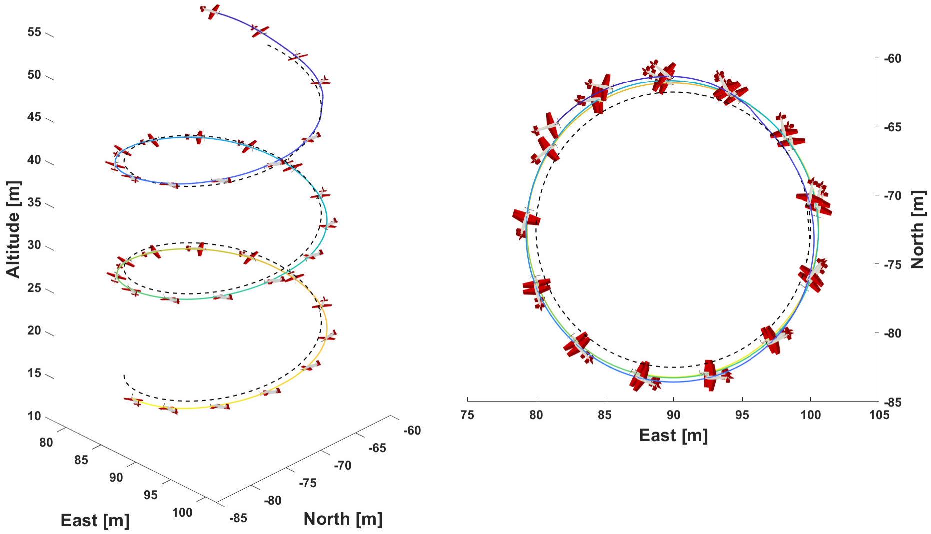


Figure 5.17: Descending helix path following experiment, 3D and top view.

The steep descent makes the maneuver more challenging than the planar circle, yet,

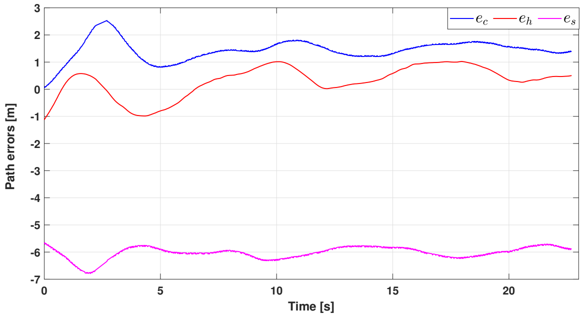


Figure 5.18: Path-relative errors for descending helix path following experiment.

thanks to the path following strategy, the UAV is able to reach the path and stay close to it. As seen from the path-relative errors plot, the cross-track error remains between one and two metres once the UAV has reached the path. The magnitude of the altitude error, is less than one metre throughout the operation.These errors are larger than the constant altitude circle path, which is understandable given the more demanding maneuver.

The on-track error is, as expected, larger, which results in the UAV not evolving at the requested $V_c = 10$ m/s. However, this error does not impede the geometric objective from being acceptably satisfied. In the position tracking formulation, this failure to meet the dynamic task, resulting in the UAV lagging behind the instantaneous trajectory would compromise the geometric task as well.

Finally, two experiments are presented where the UAV is tasked with following a 3D path while performing an arbitrary motion about the thrust axis. First, the UAV is commanded to follow a path consisting of a straight line followed by a climbing helix of 20 metre radius which exits onto a second straight segment, with a commanded velocity of 10 m/s throughout. Simultaneously, the UAV is commanded to perform a rolling Harrier at a constant angular rate of $\dot{\phi}_r = 4[rad/s]$. The top and side views of this maneuver are shown in Figure 5.19, and the cross-track and altitude errors are given in Figure 5.20.

The fast rolling motion causes oscillations about the lateral and vertical coordinates, which translates into oscillating path-relative errors throughout the maneuver. However, the control system manages to steer the UAV onto the commanded path during all segments, and the path relative errors are contained within a 2 metre margin at all times. This is remarkable given the extreme nature of the maneuver, the fact that no aerodynamic estimate is given to the control systems, and due to the presence of significant wind.

A second experiment utilizes the same composite path with the same commanded velocity, but instead demonstrates the UAV can perform a knife-edge maneuver while following the path. This maneuver is achieved by simply commanding a constant $\phi_r = 90°$. Figure 5.21

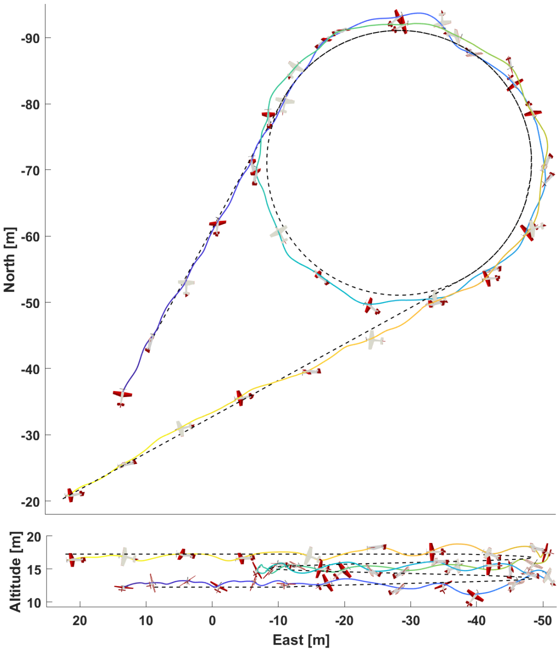


Figure 5.19: Top and side view of experimental rolling Harrier on a geometric path.

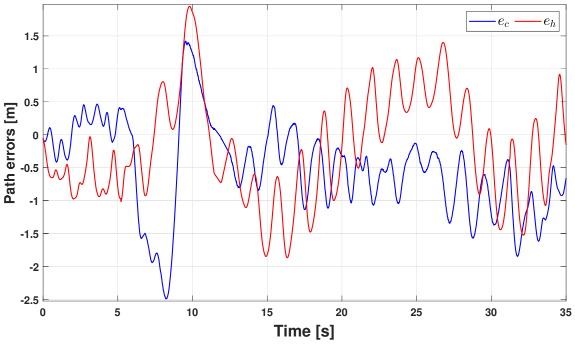


Figure 5.20: Cross-track and altitude errors for experimental Rolling Harrier on a geometric path.

shows the top and side views of this maneuver, and Figure 5.22 shows the corresponding cross-track and altitude errors.

While the rolling Harrier is the more demanding, aggressive maneuver, it is during the knife-edge where the larges drop in performance occurs. This might appear counter intuitive, but it can be explained by the fact that the UAV is most vulnerable to wind during a knife-edge, since it presents its largest surface to the incoming wind, generating significant drag. Still, the control system manages to follow the path, and the cross-track error remains small. The altitude error has increased when compared to the prior maneuver, but this can be

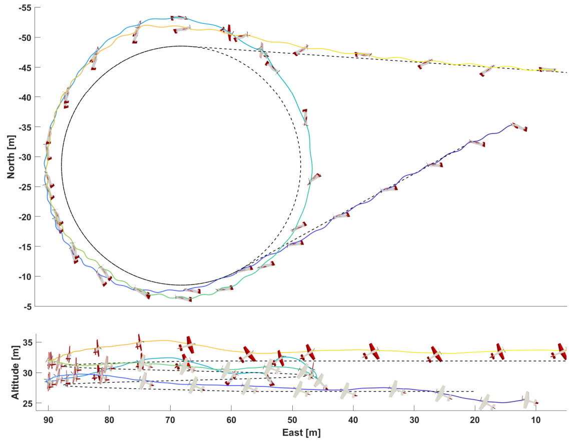


Figure 5.21: Top and side view of experimental knife-edge on a geometric path.

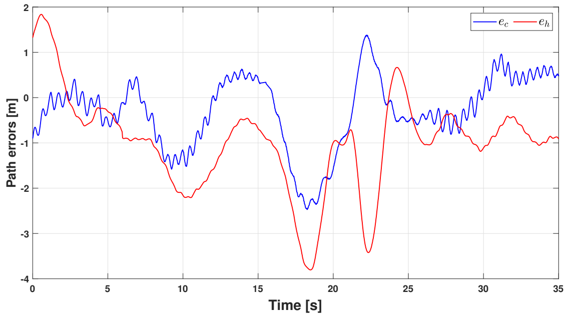


Figure 5.22: Cross-track and altitude errors for experimental knife-edge on a geometric path.

attributed to platform limitations: the motor is now tasked not only with keeping UAV speed, it is now the sole mechanism keeping the UAV airborne and it must additionally compensate for significant drag when the path tangent is perpendicular to the incoming wind.

Both maneuvers show an improvement when compared to their position tracking counterparts, especially when the geometric objective is the main priority. This allows for on-path aerobatic functionality, a difficult task which normally requires extensive system characterization. Indeed, the simplicity in achieving these maneuvers underscores the versatility of the proposed control architecture. The controller was able to successfully perform these complex motions with minimal information: only the arbitrary rotation about the thrust axis and the desired path were prescribed. Unlike other schemes, there is no need for a combined attitude and position reference to be computed *a priori*, the control system is capable of determining the motions on its own.

5.4.3 Results discussion

This chapter focused on two developments: the proposed moving path frame, constructed specifically for the application of vehicle motion control in 3D, and its integration with a path following control strategy compatible with the previously developed agile fixed-wing UAV control framework.

This strategy was developed to improve controller performance, which was diminished in the presence of wind disturbances. Simulations and experimental flight tests clearly demonstrate the benefits of the proposed control strategy. If convergence to a special path is the main objective, using the path following control system will allow the control system to prioritize this geometric goal, without the need for a separate, distinct control architecture. Indeed, this proposed system can be enabled or disabled at any point, which allows for a combination of control strategies as required by mission parameters, or operator input, as will be highlighted in the next chapter.

Chapter 6

Semi-autonomous Control

This chapter focuses on the problem of semi-autonomous control, wherein a human operator issues commands remotely to fly the UAV. The greatest challenge lies in the task of translating pilot inputs into references that can be used by the previously developed control systems. Regarding the control system, the pilot-assist solution requires a combined control strategy, employing all the tools developed in previous chapters. For this reason, this chapter can be regarded as the integration of all the elements of this thesis.

First, the conventional RC pilot inputs are presented, together with a brief discussion of pilot *intent*, concretely, what the pilot is trying to convey to the system, and what the pilot intuitively expects the UAV to do, when providing a specific input. These concepts form the basis of the main focus of this chapter, the reference generator system for semi-autonomous position control. We discuss how the different systems must be integrated to enable this complex functionality.

Finally, we present the simpler semi-autonomous attitude control system as a mostly stand-alone functionality.

6.1 Pilot input and pilot intent

We consider the remote operator utilizes a conventional RC transmitter (RC remote), like the Taranis FrSky X9D shown in Figure 6.1, which uses two sticks capable of two independent motions each. The two sticks map to the four traditional inputs of a fixed-wing aircraft: thrust, ailerons, elevator, and rudder. We define this as S_T, S_a, S_3, and S_r, respectively, their corresponding motion shown in Figure 6.1. Without loss of generality, we consider these as normalized inputs, since they are treated as such by the stock PX4 firmware. Then,

$S_T \in [0,1]$ and S_a, S_e, $S_r \in [-1,1]$, the distinction due to the impossibility of commanding a negative thrust. The mapping between stick motion and these normalized values is in principle linear, but is usually modified [106] to adjust the sensitivity.

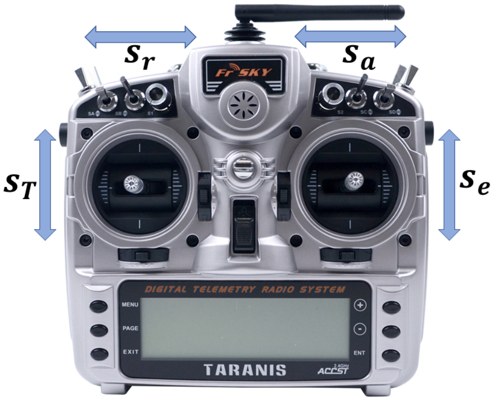


Figure 6.1: Taranis X9D RC transmitter, and stick mapping.

Conventional operation RC operation consists of the pilot prescribing these control inputs in open-loop (with the exception of the ESC for the motor and servomotors for the deflections). The most basic pilot-assist system allows the pilot to specify angular velocity instead of controlling the deflections directly. However, easier and safer operation can be achieved if the UAV levels itself (or to a trim position) when the control sticks are released. In this common mode of operation, S_e and S_a prescribe pitch and roll set points, respectively, effectively a reduced attitude reference. A constant roll will induce a banking turn; hence S_a indirectly specifies a heading rate. For this reason, S_r is mostly a redundant input in this mode, often left connected directly to the rudder. This scheme will be the basis for the semi-autonomous attitude control.

While the semi-autonomous position control problem is different and more complex, considering these basic pilot motions when mapping the four available inputs into references for the underlying control systems is fundamental to ensure an intuitive operation. In open-loop, changing the thrust setpoint will cause the UAV to increase or reduce its speed. An elevator command will cause the UAV to climb or descend. Finally, a roll command will result in a bank, in turn inducing a heading rate. These three quantities correspond to the inputs of the kinematic model [2], commonly assumed in autonomous vehicle control and motion planning, and motivate its use for reference generation. An initial idea is then to generate a velocity reference through

$$\mathbf{v}_r(t) = \begin{bmatrix} V_r \cos(\psi_r)\cos(\gamma_r) \\ V_r \sin(\psi_r)\cos(\gamma_r) \\ -V_r \sin(\gamma_r) \end{bmatrix}, \tag{6.1}$$

where V_r is a reference velocity magnitude, given as a function of S_T, γ_r is a reference climb angle, given as a function of S_e, and ψ_r is a reference heading, which is defined through

$$\dot{\psi}_r = \delta\psi_r, \tag{6.2}$$

where $\delta\psi_r$ is a reference heading rate, prescribed as a function of S_a.

6.1.1 Velocity vs. position references

The kinematic model (6.1) creates a time-parametrized veocity reference, but can be numerically integrated to generate a matching position reference. Translating pilot commands into constant position setpoints or slowly evolving time-parametrized inertial positions is a common feature of multirotor control. In this scheme, the pilot is effectively prescribing the position of a virtual target, which they can accurately envision at slower speeds. This can be extended to some fixed-wing, steady flight maneuvers, such as in [106], where we proposed a pilot-assist landing system.

However, there are two challenges to this strategy at high-velocities. The first corresponds to pilot perception: a remote operator, limited to line-of-sight feedback cannot adequately estimate instantaneous UAV position, but can estimate velocity magnitude and direction. The second, and most relevant to the trajectory generation problem, is that a remote human operator has no notion of a virtual target position in this regime. Concretely, when the pilot commands some UAV motion, the expected UAV behaviour is not time-parametrized, but geometric. For example, a pure roll command means the pilot intends the UAV to trace a circle while holding altitude, not to be at a specific position in a circle at a precise instant.

If a time-parametrized behaviour is enforced in this regime, and the UAV lags behind the virtual target, the remote operator will find the aircraft unresponsive, as he is unaware of the exact position of the virtual reference. This is one of the motivators for developing the path following strategy, which we use for the higher velocity regime as a mechanism to shape pilot-prescribed velocity references to achieve a geometric, spatial objective.

6.2 Reference generation

Three distinct reference generation strategies are proposed, based on the UAV speed prescribed by the pilot. Figure 6.2 summarizes the different features of these flight regimes.

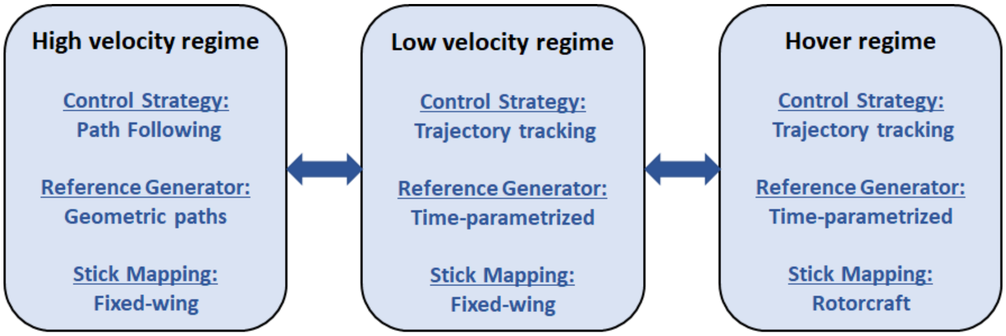


Figure 6.2: Semi-autonomous flight regimes.

At higher velocities, when the UAV behaves like a conventional fixed-wing, pilot input is translated into desired geometric paths the aircraft should navigate. This is then achieved using the path following control system described in Chapter 5.

As the UAV slows down, the semi-autonomous solution changes to the stricter position tracking controller, to allow for more precise maneuvering. This regime is especially useful for precise landings, such as the experimental autonomous landing approach in Figure 4.15. Internally, the reference generator changes from geometric paths to time-parametrized trajectories. However, from the pilot's point of view, there is little to no change in the way the aircraft is controlled, since the input mapping is retained: the UAV still handles like a fixed-wing aircraft. We consider it is within this regime that the transition from horizontal to vertical attitude generation occurs. As such, this can be considered a transitory regime between pure fixed-wing handling and pure rotorcraft handling.

Finally, as the aircraft's reference velocity approaches zero, the semi-autonomous solution changes to the hover regime where the key difference are the pilot input mappings. As the agile fixed-wing hovers, it behaves more like a rotorcraft, and the stick mapping reflects this. Internally, there is no change in the control system itself.

We first determine the operator-prescribed speed, through the simple proportional relation

$$V_r = V_M \, S_T, \tag{6.3}$$

where $V_M > 0$ is the maximum allowed speed during pilot-assisted flight.

6.2.1 High velocity regime

The high velocity regimes occurs when $V_r > V_{hv}$, where V_{hv} is the high velocity threshold. This regime translates pilot input into references, where we consider two types of paths: straight lines and helices.

The reference generator first checks the roll command, S_a, to determine if the user is requesting a straight line or a circular motion. This is done by defining the desired turn radius, r_d, through

$$r_d = \frac{r_m - r_M}{1 - a_{dz}} |S_a| + \frac{r_M - a_{dz} r_m}{1 - a_{dz}}, \tag{6.4}$$

where r_m is the minimum turn radius achieved at the maximum S_a command, and r_M is the maximum turn radius, achieved when $S_a = a_{dz}$, where $a_d z \in (0, 1)$ is a deadzone value for the aileron stick. The value of r_m is limited by platform characteristics, but r_M and a_{dz} obey pilot sensitivity preferences.

If $r_d = 0$, the system considers the pilot has zeroed the aileron stick, and is requesting a straight line path, constructed through

$$\boldsymbol{\mu}(\sigma) = \begin{bmatrix} (\sigma - \sigma_0) \cos\psi_0 \cos\gamma_r \\ (\sigma - \sigma_0) \sin\psi_0 \cos\gamma_r \\ -(\sigma - \sigma_0) \sin\gamma_r \end{bmatrix} + \mathbf{p}_{r0}, \tag{6.5}$$

where γ_r is the reference climb angle, prescribed by the pilot through

$$\gamma_r = \begin{cases} \gamma_{Mc} S_e & S_e \geq 0 \\ \gamma_{Md} S_e & S_e < 0 \end{cases}, \tag{6.6}$$

where γ_{Mc} and γ_{Md} are the maximum climb (positive) and descent (negative) angles. This distinction is made to allow for aggressive climbs but less steep descents, if preferred by the operator. We use the parameters of initial heading, ψ_0, initial reference position, $\mathbf{p}_{r0}$ and initial path parameter, σ_0, to enable the stitching together of different continuous paths. The initial path parameter is particularly important, as it allows the path evolution law (5.13) to be used throughout, without needing to reset its value.

If $r_d > 0$, then the pilot commands a helical path, built through

$$\boldsymbol{\mu}(\sigma) = \begin{bmatrix} r_d \sin\left(\frac{\sigma-\sigma_0}{\sqrt{r_d^2+c_d^2}}\right)\cos(\psi_0) - \mathrm{sgn}(S_a) r_d \left(1-\cos\left(\frac{\sigma-\sigma_0}{\sqrt{r_d^2+c_d^2}}\right)\right)\sin(\psi_0) \\ r_d \sin\left(\frac{\sigma-\sigma_0}{\sqrt{r_d^2+c_d^2}}\right)\sin(\psi_0) + \mathrm{sgn}(S_a) r_d \left(1-\cos\left(\frac{\sigma-\sigma_0}{\sqrt{r_d^2+c_d^2}}\right)\right)\cos(\psi_0) \\ -\frac{c_d(\sigma-\sigma_0)}{\sqrt{r_d^2+c_d^2}} \end{bmatrix} + \mathbf{p}_{r_0}, \quad (6.7)$$

where c_d is a desired climb (or descent) rate defined as

$$c_d = r_d \tan(\gamma_r), \quad (6.8)$$

with γ_r calculated from (6.6).

The above sets of equation guarantee any stitched-together path will be piece-wise continuous. In turn, the gravity normal frame will evolve smoothly throughout any path the user may prescribe. Care must be taken only with the initial values which are transferred when r_d changes. In fact, the above paths correspond to a straight line with zero heading, and a helix starting at the origin with zero heading, after an affine transformation consisting of a ψ_0 rotation about $\hat{\mathbf{k}}_3$ and a displacement of $\mathbf{p}_{r0}$. The last value of σ before the r_d changes is stored as the constant σ_0 for the next segment.

6.2.2 Low velocity regime

The low velocity regime encompasses the range $0 < V_r \leq V_{hv}$. To ensure continuity with the preceding path following strategy, the time parametrized trajectories in this regime are obtained by setting $\mathbf{v}_r(t) = V_r\mathbf{T}$ for the straight line path and helix.

Then, for a straight line,

$$\mathbf{v}_r(t) = V_r \begin{bmatrix} \cos(\psi_0)\cos(\gamma_r) \\ \sin(\psi_0)\cos(\gamma_r) \\ -\sin(\gamma_r) \end{bmatrix}, \quad (6.9)$$

and, for a helix,

$$\mathbf{v}_r(t) = \frac{V_r}{\sqrt{r_d^2+c_d^2}} \begin{bmatrix} r_d \cos\left(\psi_0 + \mathrm{sgn}(S_a)\frac{V_r\, t}{\sqrt{r_d^2+c_d^2}}\right) \\ r_d \sin\left(\psi_0 + \mathrm{sgn}(S_a)\frac{V_r\, t}{\sqrt{r_d^2+c_d^2}}\right) \\ -c_d\, t \end{bmatrix}. \quad (6.10)$$

The matched reference position is then obtained through numeric integration, as

$$\mathbf{p}_r(t) = \int_0^t \mathbf{v}_r(\tau)d\tau + \mathbf{p}_{r0}. \tag{6.11}$$

The input S_r has so far not been considered in either of these two regimes. It cannot be used to control the rudder directly, since it is directed by the closed-loop attitude controller in combination with the other control surfaces. However, this additional available input can be re-purposed for specific tasks. For example, in [106] we proposed a modified trajectory generation scheme during a landing approach, where S_a and S_r are used to prescribe two distinct lateral motions. Consider the alternative velocity reference for a landing approach, constructed from a modified kinematic model,

$$\mathbf{v}_r(t) = \begin{bmatrix} \left(\sqrt{V_r^2 - V_l^2}\cos(\psi_0 + \delta\psi_r t) - V_l \sin(\psi_0 + \delta\psi_r t)\right)\cos(\gamma_r) \\ \left(\sqrt{V_r^2 - V_l^2}\sin(\psi_0 + \delta\psi_r t) + V_l \cos(\psi_0 + \delta\psi_r t)\right)\cos(\gamma_r) \\ -V_r \sin(\gamma_r) \end{bmatrix}, \tag{6.12}$$

where $V_l < V_r$ is an added lateral velocity, prescribed as

$$V_l = V_{l_M}\, S_r, \tag{6.13}$$

where V_{l_M} is a maximum lateral velocity (in practice, this is given as a percentage of the total desired velocity, V_r), and

$$\delta\psi_r = \delta\psi_{max}\, S_a, \tag{6.14}$$

where $\delta\psi_{max}$ is the maximum heading rate, left small for a landing maneuver.

The effect of this alternative trajectory generation is that the aileron command will cause a standard circular motion, while the rudder command will "pull" the aircraft sideways while retaining heading. During experiments, we found that, due to the remote perspective, it is easier to correctly align the UAV heading with the landing path, but that lateral adjustments were needed. This approach allows the pilot to first set the UAV heading, and then correct for lateral deviation without modifying the heading approach. These different lateral motions are better explained in Figure 6.3. Depending on pilot preference, S_a and S_r can be interchanged, and, additionally the added roll can be disabled during a landing maneuver, which allows for maneuverability while ensuring the aircraft touches down safely.

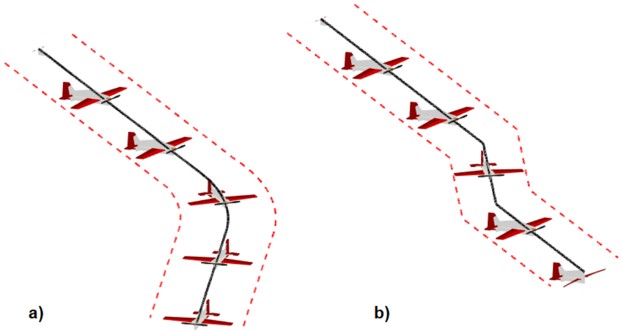


Figure 6.3: Landing approach lateral maneuvers. a) Heading change through S_a. b) Lateral shift through S_r.

6.2.3 Hover regime

When the pilot completely pulls back the throttle stick, S_T, the reference velocity, V_r, becomes zero and, as shown in the position control experiments, the UAV slows down into a hover. In this regime, we allow for an optional change of control input mapping, mirroring the control paradigm of rotorcraft vehicles. To do so, we capitalize on the properties of the vertical attitude determination algorithm, allowing the pilot to rotate the constructor vector through (4.46), specifically

$$\hat{\mathbf{h}} = \begin{bmatrix} \cos(\psi_0 + \delta\psi_r t), & \sin(\psi_0 + \delta\psi_r t), & 0 \end{bmatrix}^T, \tag{6.15}$$

where the heading rate is now specified by the rudder command, through

$$\delta\psi_r = \delta\psi_{max} \, S_r. \tag{6.16}$$

In true multirotor fashion, the rudder, or "yaw" stick will rotate the aircraft about the thrust axis. This then allows the pilot to prescribe motion relative to the UAV's main axes, through the elevator and aileron sticks. The reference velocity can then be constructed through

$$\mathbf{v}_r(t) = \begin{bmatrix} V_h(\cos\psi_r + \zeta) \\ V_h(\sin\psi_r + \zeta) \\ -V_v \end{bmatrix}, \tag{6.17}$$

where $\psi_r = \text{atan2}(h_2, h1) = \psi_0 + \delta\psi_r t$, V_h and ζ are created through a combination of a forward-backward velocity, V_{fb} and a left-right velocity, V_{lr}, prescribed by the pilot,

$$V_{fb} = V_{fb_M} S_e \tag{6.18}$$

$$V_{lr} = V_{lr_M} S_a, \tag{6.19}$$

then $V_h = \sqrt{V_{fb}^2 + V_{lr}^2}$ and $\zeta = \text{atan2}(V_{lr}, V_{fb})$. Finally, V_v is a vertical velocity, which needs to be prescribed through some other input, as the remaining input, S_T, is zeroed to enable hovering maneuvers. Of course, the function of this stick can be modified while in this mode through an additional Boolean channel (switch).

This control mapping results in an intuitive piloting scheme which mirrors that of a standard quadrotor. If a new set of Euler angles were defined about the hover position, S_e would lead to a pitching motion, S_a to a rolling motion, and S_R would cause the aircraft to yaw. This can be seen from Figure 6.4

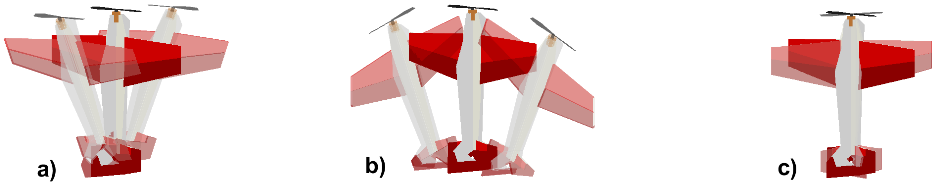


Figure 6.4: Semi-autonomous hover motions. a) Forward-backward motion through S_e. b) Left-right motion through S_a. c) Rotation motion through S_r.

6.2.4 Controller initialization

While the overall control structure is shared between the path following and position tracking strategies, some values need to be properly initialized when transitioning between the two.

When controller gains k_c and k_h are zero, the reference velocity corresponds strictly to the commanded velocity, V_r in the direction of the tangent, T. If the cross-track and altitude errors are assumed small before a transition, then velocity reference for the tracking solution can be initialized as

$$\mathbf{v}_r(t) = V_r \mathbf{T}. \tag{6.20}$$

To initialize position, we consider that both the position tracking law (4.13) and the velocity control law (4.27) use the same gains, k_v and k_p. When transitioning to the position tracking control, it is best to initialize the reference position, to align the "proportional" error terms. This is simply done by setting

$$\mathbf{p}_r(0) = \mathbf{e}_{vi} + \mathbf{p}_i. \tag{6.21}$$

The integral term, $\mathbf{e}_i$, is set to zero when initializing tracking functionality. Initializing this way may cause the UAV reference to jump ahead of the geometric path, but operation will not be affected since the control law remains smooth, and the remote operator is unaware of the precise position of the virtual target.

When transitioning from tracking to path following, should the pilot command the UAV to increase its velocity past the threshold, then the integral velocity error is initialized at

$$\mathbf{e}_{vi} = \mathbf{e}_p. \tag{6.22}$$

6.3 Semi-autonomous attitude control

In situations where position feedback is not available, or in case of failure, it is useful to allow the pilot to still control the UAV through the core attitude control system. As explained at the beginning of this chapter, a stable, pilot-assist attitude control system is a common functionality in RC platforms.

In this scheme, the throttle command, S_T, is passed directly to the motor, and the elevator and aileron commands are used to prescribe pitch and a roll, through

$$\theta_r = \theta_M S_e, \tag{6.23}$$

$$\phi_r = \phi_M S_a, \tag{6.24}$$

where θ_M and ϕ_M are the maximum allowed pitch and roll, respectively. In conventional, Euler angle based attitude control, independent control laws can be constructed for the coordinates of pitch and roll, while yaw is left free to evolve as dictated by the UAV dynamics. As explored in the semi-autonomous position system, letting the UAV rotate through a bank is the most intuitive operation and the Euler angle based control system is ideal in the sense that it does not enforce a rotation about the third axis, allowing for this banked turn.

The challenge presented here, is that θ_r and ϕ_r only determine a reduced attitude but the proposed attitude control system, developed on SO(3), requires a complete attitude reference in the form of a DCM. If such reference is constructed as

$$\mathbf{C}_{ri} = \mathbf{C}(\phi_r, \hat{\mathbf{k}}_1)\mathbf{C}(\theta_r, \hat{\mathbf{k}}_2),$$

it will implicitly command a zero reference yaw, $\psi_r = 0$, which will prevent the UAV from turning.

One alternative is to work directly with the reduced attitude. This is a different problem than that encountered in the attitude determination task in the position control system, since the unspecified rotation is about the third inertial axis $\hat{\mathbf{k}}_3$, not the last body (thrust) axis. This problem, aimed specifically for a fixed-wing UAV platform, has been extensively studied in [111]. One may use θ_r and ϕ_r to construct a reduced attitude reference, $\Gamma_r \in \mathbb{S}^2$ and then use a reduced attitude control law, with the option of adding damping to enforce a coordinated turn with no sideslip.

In this work, we instead opt to keep the base SO(3) control system and complete the attitude reference through the coordinated turn equation assuming no sideslip [2],

$$\dot{\psi}_{coord} = \frac{g}{V_a} \tan(\phi_r), \tag{6.25}$$

where, in the absence of airspeed measurements, the airspeed is approximated as $V_a \approx \|\mathbf{v}_i\|$. We then calculate the reference yaw as

$$\psi_r = \int_0^t \left(\dot{\psi}_{coord}(\tau) + \delta\psi_{add}(\tau) \right) d\tau, \tag{6.26}$$

where the additional heading rate, $\delta\psi_{add}$ is an optional, added rotation rate, prescribed by the rudder command,

$$\delta\psi_{add} = \delta\psi_M S_r, \tag{6.27}$$

where $\delta\psi_M$ is the maximum additional yaw rate. The rotation rate $\dot{\psi}_{coord}$ obeys the coordinated turn equation, and using it will result in a correctly banked turn tracing a circular path. In contrast, $\delta\psi_{add}$, emulates a more aggressive turn maneuver, nominally achieved in open-loop operation through a rudder command. In operation, the pilot can utilize this command independently, to turn the aircraft without banking, or in combination with the roll command to achieve tighter rotations more characteristic of the agile fixed-wing.

6.4 Controller verification

We divide this section between the simpler, semi-autonomous attitude controller and the more complex semi-autonomous position control system. We present SITL results instead of numerical simulations performed in Matlab. The SITL environment is ideal for the pilot-assist solutions, providing a visualization of the UAV for pilot feedback, and allowing for the integration of the RC transmitter used in experiments.

All simulations and experiments use the control gains in Tables 4.2 and 5.2.

6.4.1 Pilot-assist attitude control

For SITL and experimental flights, the reference generator parameters given in Table 6.1 are used.

Table 6.1: Parameters for semi-autonomous attitude control.

Param.	Value	Units
ϕ_M	45	deg
θ_M	45	deg
$\delta\psi_{add}$	1	rad/s

Figure 6.5 shows the top and side view of a SITL simulations, where the semi-autonomous attitude solution is used to perform a maneuver, from take-off to landing.

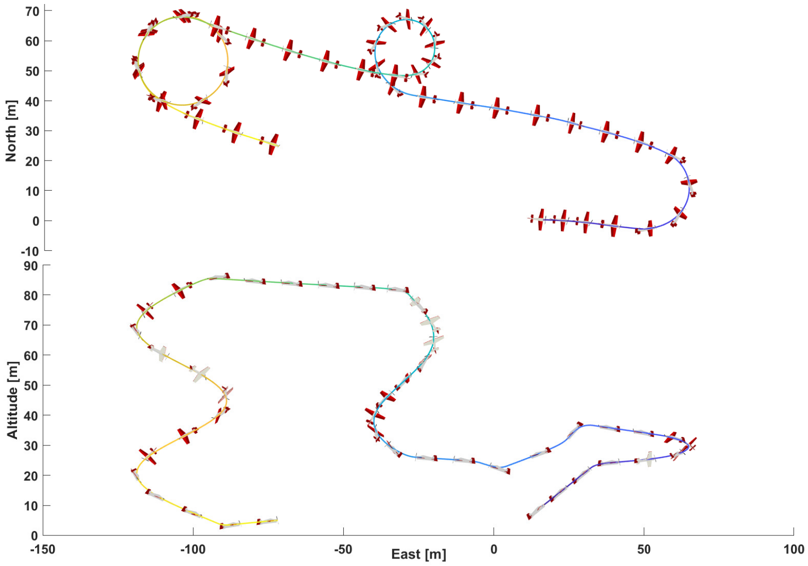


Figure 6.5: Top and side view of SITL pilot-assist attitude.

Attitude control, either autonomous or semi-autonomous, is safer alternative for take-off maneuvers, especially when a take-off platform is used, since thrust is not part of the core attitude controller, and can be prescribed independently. The normalized pilot inputs that trace this maneuver are shown in Figure 6.6.

As seen from the top-view, the coordinated turn equation allows the pilot to trace a helical path when banking, as expected. These banking maneuvers, and the corresponding change in aircraft yaw, can be seen in Figure 6.7, where an Euler angle parametrization of the reference attitude, traced by the operator, and that of the UAV is shown. Since the reference attitude is constructed through Euler angles, these can be immediately correlated with the pilot inputs.

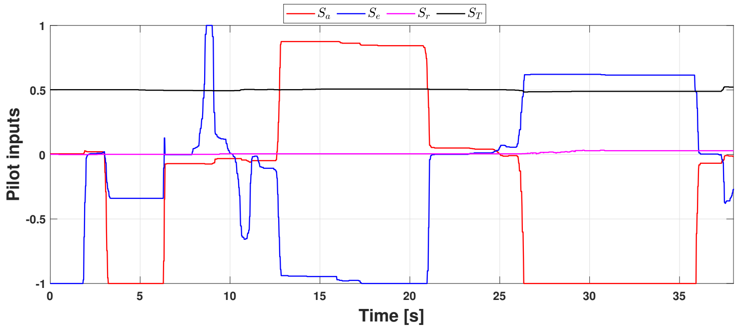


Figure 6.6: Pilot inputs for SITL pilot-assist attitude.

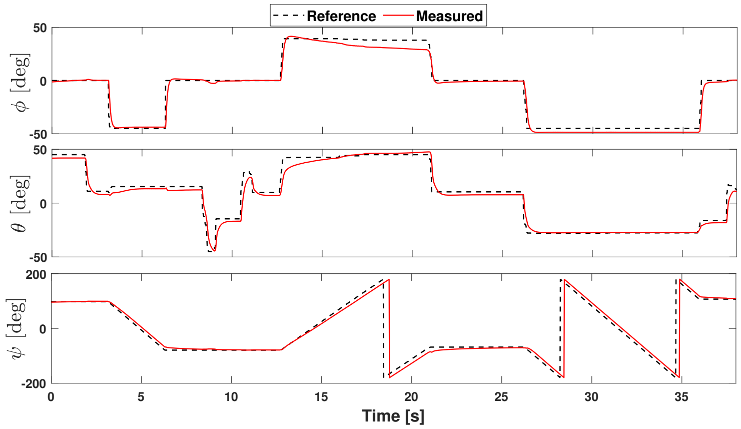


Figure 6.7: Euler angles for SITL pilot-assist attitude.

We now use this semi-autonomous attitude solution in a flight experiment, starting with a take-off maneuver from an angled platform, shown in Figure 6.8. The 3D path of this flight can be seen in Figure 6.9. This flight experiment was performed under high-wind conditions, with mean wind velocity greater that 5 m/s.

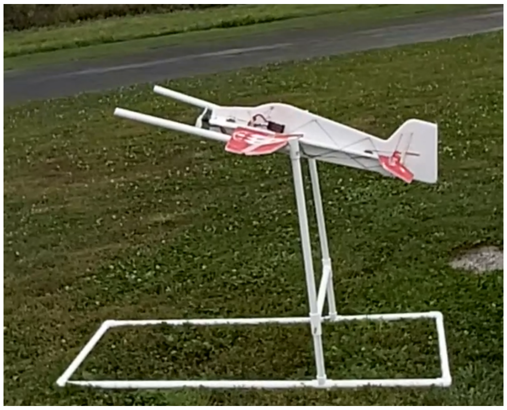


Figure 6.8: Take-off platform for agile fixed-wing UAV.

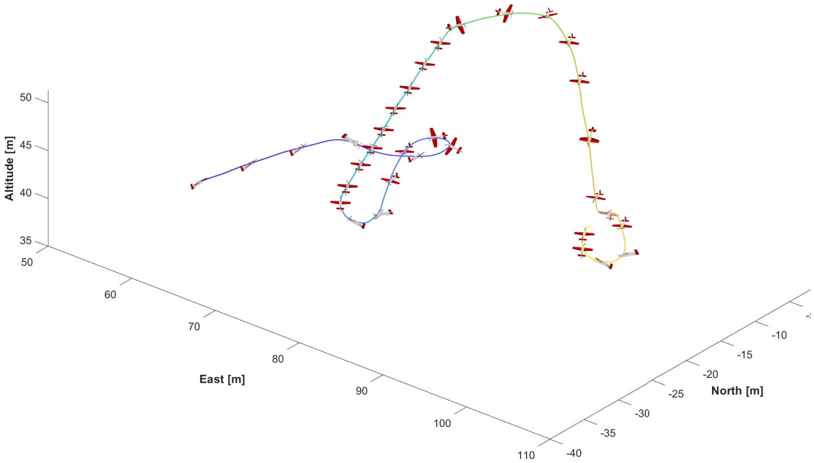


Figure 6.9: 3D view of pilot-assist attitude control experiment.

The effects of the wind first can be observed in the shape of the banking maneuvers, corresponding to the aileron commands shown in Figure 6.10. These are not perfectly circular as in the wind-free SITL simulation. Since there is no airspeed measurement, it is approximated through the ground velocity. An airspeed estimate would yield better results. In addition, the coordinated turn equation assumes no side-slip, yet, in this high wind conditions there is significant side motion during the turns.

The second effect of wind is better observed in the Euler angle decomposition shown in Figure 6.11. The roll coordinate presents noticeable oscillations, indicating the attitude control gains of the first body axis, are too high in windy conditions. This is also a result of wing flexibility, which, as the largest UAV surface, shows flapping behaviour in high winds.

Even in this challenging conditions, the core attitude control system allows the pilot to fly the aircraft safely. The semi-autonomous system allows the operator to focus on the navigation and disturbance rejection tasks, instead of struggling to keep the aircraft stable.

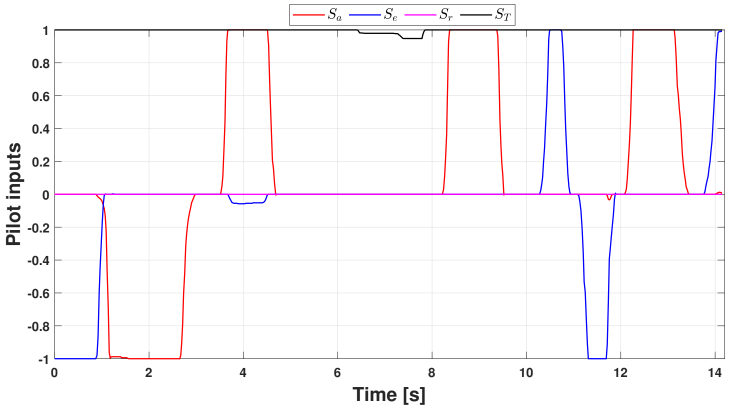


Figure 6.10: Pilot inputs for pilot-assist attitude control experiment.

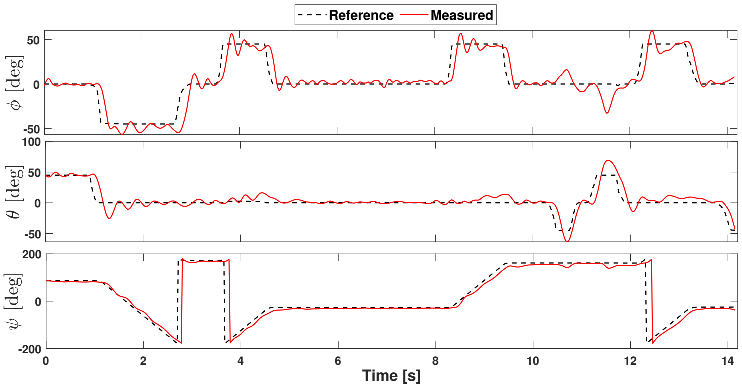


Figure 6.11: Euler angles for pilot-assist attitude control experiment

Since no position feedback is available, the pilot acts as the position control loop in this mode.

6.4.2 Pilot-assist position control

We present an extended experimental test to verify the properties of the integrated system. During this flight, we retain the same control parameters from Tables 4.2 and 5.2 with additional parameters required by the reference generator given in Table 6.2.

The following experiment is performed in moderate winds, with mean wind velocity ≈ 4 m/s. This is a longer flight, almost entirely flown in the pilot-assist position control model. The path traced by the UAV, and the reference paths and trajectories are shown in two views, an isometric 3D view in Figure 6.12 and a top view in Figure 6.13.

The maneuver is best understood together with the pilot inputs shown in Figure 6.14 and the UAV position and its reference, 6.15. The initial section of the maneuver is a pilot-triggered autonomous take-off. Initially, the UAV is at rest on a take-off platform, but at $t = 4$ s, the pilot raises the throttle stick to its maximum value, triggering an autonomous

Table 6.2: Parameters for semi-autonomous position control.

Param.	Value	Units
V_{hv}	5	m/s
V_M	10	m/s
r_m	10	m
r_M	30	m
γ_{Mc}	60	deg
γ_{Md}	30	deg
$\delta\psi_{max}$	0.5	rad/s
V_{fbM}	3	m/s
V_{lrM}	3	m/s

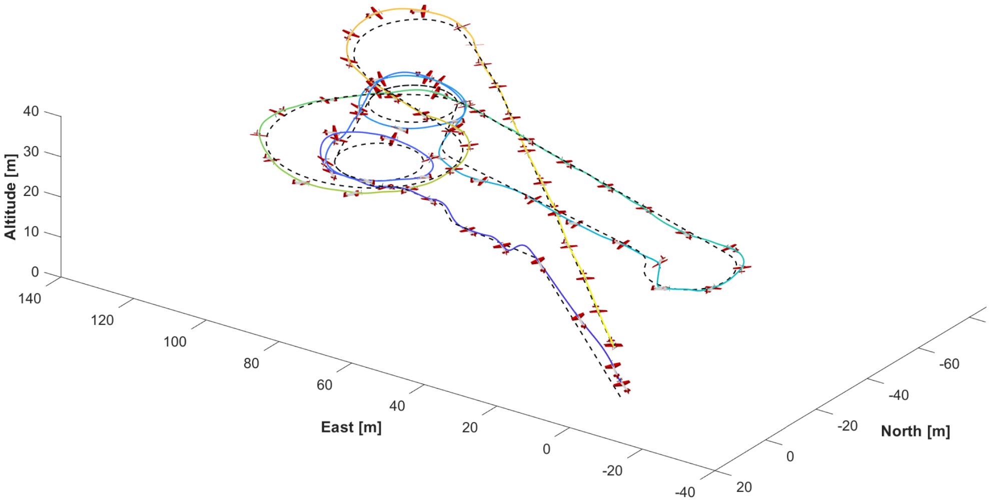


Figure 6.12: 3D view of pilot-assist position control experiment.

take-off maneuver. The system achieves this through pure attitude control, maintaining the initial UAV yaw and climbing at a 45 degree angle. This maneuver lasts until the UAV reaches 30 m altitude, at this instant, around $t = 7$ s, the system cedes control back to the operator.

The transition to position control is not completely smooth, since the UAV is abruptly commanded to stop climbing. This causes an initial overshoot from the trajectory. As the throttle stick is at its maximum value, the UAV is commanded to fly at the maximum velocity, $V_M = 10$ m/s. At this speed, the UAV is in the high velocity regime ($V_r > V_{hv} = 5$ m/s) and the controller internally operates in path following mode.

Two clockwise, minimum radius circles are commanded, followed by a partial circle for repositioning, and a larger radius counter-clockwise circle are commanded through the aileron

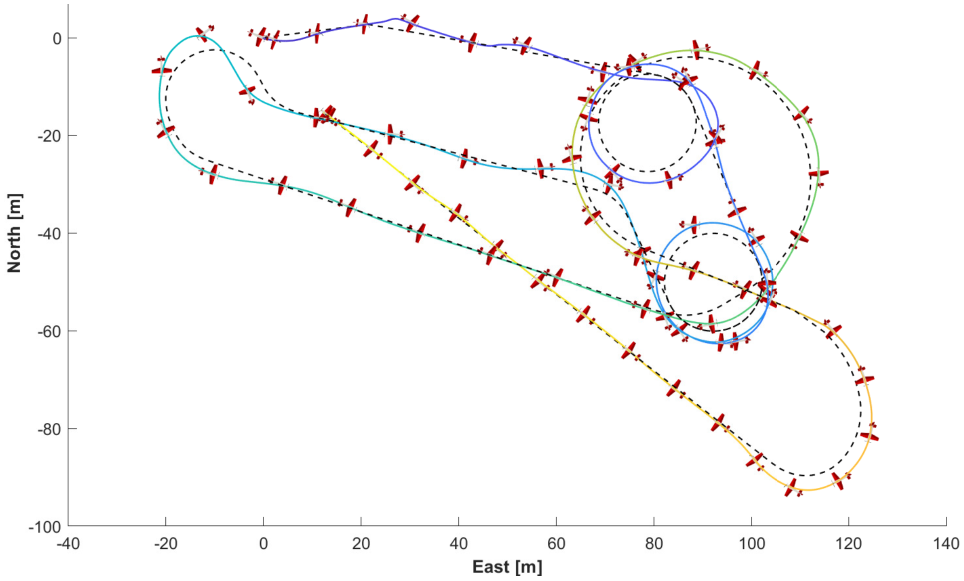


Figure 6.13: Top view of pilot-assist position control experiment.

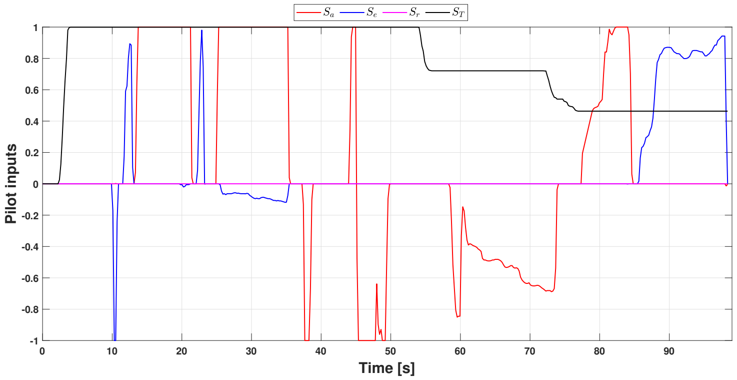


Figure 6.14: Pilot inputs for pilot-assist position control experiment.

stick, S_a. It is worth noting that the pilot slows down the aircraft as it enters the last circle, but the reference speed is still above the defined threshold. Throughout these maneuvers, the path following control system does a good job at keeping the UAV close to the path traced by the pilot.

As the UAV exits this last circle, the pilot slows down further and commands a half rotation to align the UAV for a landing approach. The thrust stick value, S_T, corresponds to a commanded velocity of 4.6 m/s, which is below the high velocity threshold. Then, the control system internally switches to position tracking, to allow for more precise maneuvering. However, as the transition between modes ensures a continuous intermediate command, this change is not noticeable by the remote operator.

The final section of the flight is a slow landing approach, where the operator only modifies the descent angle through S_e, until the UAV lands safely, as seen from the altitude coordinate

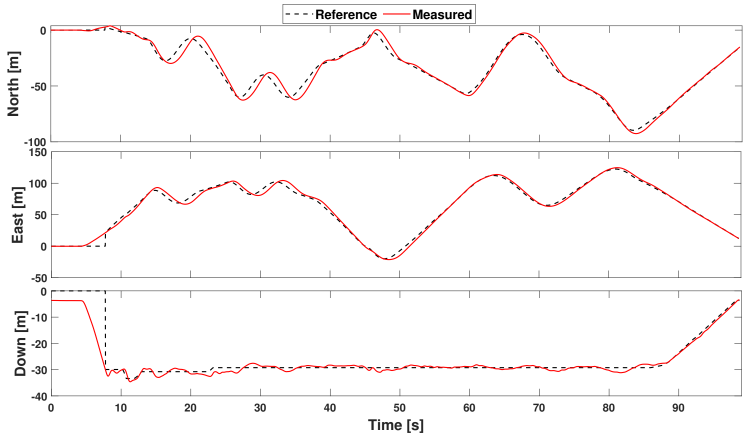


Figure 6.15: UAV position and reference for pilot-assist position control experiment.

returning to the initial value of zero.

6.4.3 Results discussion

The experimental flight shows the proposed strategies can be used to achieve safe, while flexible, operation of the UAV. First, the proposed reference generation strategy is shown to effectively lead to a piece-wise continuous path, which together with the developed gravity normal frame leads to smooth operation of the path-following control strategy.

In addition, the experimental flight showcases the successful transition between a high velocity strategy that prioritizes geometry and a lower velocity scheme that strives for strict time-constrained behaviour, even while operating under windy outdoor conditions.

Convergence to the path is not perfect, and the cross-track error increases when the path changes. The proposed path-stitching strategy is simple for the sake of efficiency, but better results can be obtained if other curves are employed, such as splines. In semi-autonomous operation, the importance of this errors is diminished due to the limited operator perception. During operation, without additional feedback, the remote pilot is not necessarily aware the UAV has deviated slightly from the path. Still, if the proposed path generation strategy is used, it is important to know the error bounds for a particular platform as a guideline for the operator.

While not developed in this work, this experiment also demonstrates the viability of combining autonomous and semi-autonomous functionality to accomplish specific missions. This integration can be improved, for example, by implementing a smooth transition from pure attitude control to semi-autonomous position in order to avoid the initial overshoot.

Finally, it is worth noting that both the semi-autonomous attitude and semi-autonomous position control systems have allowed me, the author, to safely operate the aircraft, including the difficult task of landing of landing under windy conditions even considering that, as inexperienced pilot, I do not have the necessary skill to fly such a challenging platform in fully manual configuration.

Chapter 7

Conclusions

In this final chapter, we summarize the contributions made, and offer suggestions for future work.

7.1 Summary of contributions

The core attitude control system was developed directly on SO(3) both to avoid known issues arising from attitude parametrizations, and due to its compatibility with the proposed position control system. While the core attitude control laws and the innovation terms have been developed previously, attitude control strategies on SO(3) are uncommon in fixed-wing aircraft and this chapter presents a comprehensive study of their applicability to the agile fixed-wing platform.

Through a combination of stability analysis and extensive simulations and experiments, we develop controller strategies that are well suited to different flight conditions and reference trajectory characteristics. We show the proposed attitude control strategy enables the agile fixed-wing UAV to perform both steady and complex aerobatic maneuver, in both autonomous and semi-autonomous operation. While controller design and analysis focuses on agile fixed-wing UAVs, it is applicable to the wider range of vehicles that generate control moments through deflection of airflow-immersed control surfaces.

Next, a nonlinear position control system was developed. The proposed system is built around a nonlinear vector-projection algorithm for the reference attitude determination problem, which effectively extends the results of thrust-borne UAV control systems to the complete range of operation of the agile fixed-wing platform. This results in a unified position control system, capable steady flight, hover and aerobatic maneuvers, and regime transition.

Through rigorous Lyapunov stability analysis we show the viability of the proposed system. Several simulations and experiments then verify its applicability as well as its performance limitations.

The proposed system allows for position tracking and velocity control, and the shared structure enables straightforward transition between the two. In addition, the control system requires only simple reference position, or velocity, trajectories as its input, highly simplifying the trajectory generation problem, as seen from both autonomous operation, or when operated by a human pilot. As opposed to other proposed schemes, there is no need for a pre-computed maneuver library, nor is the system restricted to operate at specific velocities. Finally, the built-in decoupling of roll further showcases the system's flexibility, as shown by the effortlessly achieved aerobatic maneuvers such as the rolling Harrier. This combined attitude and position control system is not limited to the agile-fixed wing UAV; it is readily applicable to other hover-capable platforms with different configurations, such as the popular tailsitter UAV.

Path following functionality was developed next, both as a means to address the perceived limitations of the position tracking system, and to allow for more intuitive operation for a remote human pilot. In developing this additional control loop, a novel moving path frame was developed. This contribution, the gravity normal frame, was proposed to address the limitations of other commonly used frames, and is applicable to the much wider problem of autonomous vehicle motion control and path planning in 3D space, it is not limited to the fixed-wing UAV application used in this work. The frame has several useful properties: it is straight-forward to compute from the curve definition, it is well-defined regardless of path curvature or torsion, it evolves smoothly through piece-wise continuous curves without additional compensation, and path relative errors calculated through it have a clear, and consistent, physical meaning.

In both simulations and experiments, the proposed path following controller is shown to significantly improve the performance of the agile-fixed wing aircraft during high-velocity, steady flight maneuvers in the presence of wind. The developed strategy is designed to be compatible with the previously developed control systems, allowing simple transition between modes of operation without compromising stability.

Finally, the ease of integration of the developed control systems into the semi-autonomous, pilot-assist system demonstrates the benefits of the cascaded, modular strategy. The combined use of position tracking and path following results in intuitive operation in all flight regimes. As demonstrated in SITL and experimental flights with a human operator, the proposed system is capable of performing conventional, steady flight maneuvers, transition into hover regime, operate like a multirotor platform, and achieve precise landings, all through limited pilot commands. Throughout the extensive experimental tests, the overall control system has proven remarkably robust to wind disturbances, even if performance is diminished. This is notable given the extremely low weight of the aircraft, originally designed for indoor use. Yet, with the proposed control strategy, it is able to perform complex aerobatic maneuvers while in windy outdoor conditions.

7.2 Recommendations for future work

The research and experiments carried out in pursuit of the objectives of the thesis identified various avenues of future research:

- Simulations and experiments show that the aerodynamic and slipstream models, while representative of the agile fixed-wing platform, do not match the exact properties of the aircraft. This is mostly observed from the need to adjust controller gains between simulation and experiments, and from the improved performance of the control systems in simulation when compared to experiments.

  Improving these models, in particular in regards to propeller slipstream and wind effects would allow for more accurate simulations, especially when devising robust control strategies.

- In Chapter 3, the control allocation problem was studied, and simple methods were proposed to estimate the effective airflow over the control surfaces. However, the problem is more complex, and better performance can be obtained if a more precise estimate is used. Two recommendations are suggested in this chapter, first, to employ different, but constant, effective airflow estimates which can be selected according to propeller angular velocity. This assumes most of the airflow comes from propeller slipstream. Second, a more accurate value can be obtained if an estimate of wind is available, but this necessitates additional hardware, and estimation algorithms. An additional research avenue is to explore robust SO(3) attitude control laws, which could be used to compensate for an imperfect estimate.

- The position control system proposed in Chapter 4 is composed of two elements, the vector-projection algorithm for attitude determination, and the intermediate command which determines thrust magnitude and direction. This work only proposes as PID plus feed-forward control law for the intermediate command, but many different control techniques could be used without modifying the rest of the control structure. Some examples worth studying include a linear quadratic regulator (LQR), a robust, sliding mode formulation to better address the imperfect aerodynamic estimate, or a control structure that accounts for actuator saturation.

- In relation to the feed-forward terms of the position control system, a more sophisticated methodology could be used to determine this element, such as a disturbance observer [112].

- The path-following control law and corresponding path evolution law presented in Chapter 5 focus on the geometric task: the convergence of the cross-track and altitude errors. Depending on the maneuver, the commanded velocity is not necessarily reached. For this reason, it worthwhile to study other path following strategies that can achieve the dynamic task after the vehicle has converged to the path, such as the ones studied in [53].

- In Chapter 6, the attitude reference for the semi-autonomous attitude control system is completed through the coordinated turn equation. While this gives good results in practice, it over-parametrizes the attitude reference. Instead, a reduced attitude control structure could be studied, such as the one presented in [111], to allow the aircraft to evolve naturally. It is also worthwhile to study the effects of the rudder on lateral motion, given its large control authority, and how a rudder command can be better incorporated into an intuitive pilot-assist attitude control solution.

- One benefit of the developed system is the ease with which maneuvers can be prescribed. However, incorporating dynamic feasibility constraints and other system information in the maneuver design could lead to improved performance, reducing overshoots and transient errors, and is worth considering.

- This work focuses on inertially referenced position and velocity, as an extension of mutirotor control systems. However, airspeed control, as opposed to ground velocity, is the norm in conventional fixed-wing control systems and implementing this functionality could result in better performance, especially during high velocity, steady maneuvers.

Bibliography

[1] B. Canis, *Unmanned aircraft systems (UAS): Commercial outlook for a new industry.* Congressional Research Service Washington, DC, 2015.

[2] R. W. Beard and T. W. McLain, *Small unmanned aircraft.* Princeton University press, 2012.

[3] S. M. Adams and C. J. Friedland, "A survey of unmanned aerial vehicle (UAV) usage for imagery collection in disaster research and management," in *9th International Workshop on Remote Sensing for Disaster Response*, p. 8, 2011.

[4] D. W. Casbeer, R. W. Beard, T. W. McLain, S.-M. Li, and R. K. Mehra, "Forest fire monitoring with multiple small UAVs," in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 3530–3535, IEEE, 2005.

[5] F. Mancini, M. Dubbini, M. Gattelli, F. Stecchi, S. Fabbri, and G. Gabbianelli, "Using unmanned aerial vehicles (UAV) for high-resolution reconstruction of topography: The structure from motion approach on coastal environments," *Remote sensing*, vol. 5, no. 12, pp. 6880–6898, 2013.

[6] J. Hall and T. McLain, "Aerobatic maneuvering of miniature air vehicles using attitude trajectories," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, p. 7257, 2.

[7] C. D. Wickens, J. Hellenberg, and X. Xu, "Pilot maneuver choice and workload in free flight," *Human factors*, vol. 44, no. 2, pp. 171–188, 2002.

[8] D. J. Wing and W. B. Cotton, "Autonomous flight rules-a concept for self-separation in US domestic airspace," *NASA Technical Report*, no. 217174, 2011.

[9] J. Levin, A. Paranjape, and M. Nahon, "Motion planning for a small aerobatic fixed-wing unmanned aerial vehicle," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8464–8470, IEEE, 2018.

[10] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. Springer, 2014.

[11] J. R. Wertz, *Spacecraft attitude determination and control*, vol. 73. Springer Science & Business Media, 2012.

[12] M. D. Shuster *et al.*, "A survey of attitude representations," *Navigation*, vol. 8, no. 9, pp. 439–517, 1993.

[13] J. Russell, *Performance and stability of aircraft*. Butterworth-Heinemann, 1996.

[14] E. L. Duke, R. F. Antoniewicz, and K. D. Krambeer, "Derivation and definition of a linear aircraft model," *NASA Technical Report*, no. 1207, 1988.

[15] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.

[16] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics and Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[17] M. S. Selig, "Real-time flight simulation of highly maneuverable unmanned aerial vehicles," *Journal of Aircraft*, vol. 51, no. 6, pp. 1705–1725, 2014.

[18] W. Khan and M. Nahon, "Real-time modeling of agile fixed-wing UAV aerodynamics," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1188–1195, IEEE, 2015.

[19] W. Khan and M. Nahon, "Development and validation of a propeller slipstream model for unmanned aerial vehicles," *Journal of Aircraft*, vol. 52, no. 6, pp. 1985–1994, 2015.

[20] W. Khan and M. Nahon, "Toward an accurate physics-based UAV thruster model," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1269–1279, 2013.

[21] P. Crouch, "Spacecraft attitude control and stabilization: Applications of geometric control theory to rigid body models," *IEEE Transactions on Automatic Control*, vol. 29, no. 4, pp. 321–331, 1984.

[22] X. Heng, D. Cabecinhas, R. Cunha, C. Silvestre, and X. Qingsong, "A trajectory tracking LQR controller for a quadrotor: Design and experimental evaluation," in *TENCON 2015-2015 IEEE region 10 conference*, pp. 1–7, IEEE, 2015.

[23] D. Yiqun, F. Jun, Y. Bin, Z. Youmin, and A. Jianliang, "Position and heading angle control of an unmanned quadrotor helicopter using LQR method," in *2015 34th Chinese Control Conference (CCC)*, pp. 5566–5571, IEEE, 2015.

[24] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "MPC with nonlinear $\mathcal{H}_\infty$ control for path tracking of a quad-rotor helicopter," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 8564–8569, 2008.

[25] R. Beard, D. Kingston, M. Quigley, D. Snyder, R. Christiansen, W. Johnson, T. McLain, and M. Goodrich, "Autonomous vehicle technologies for small fixed-wing UAVs," *Journal of Aerospace Computing, Information, and Communication*, vol. 2, no. 1, pp. 92–108, 2005.

[26] Y. Kang and J. K. Hedrick, "Linear tracking for a fixed-wing UAV using nonlinear model predictive control," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1202–1210, 2009.

[27] B. Morrell, M. Rigter, G. Merewether, R. Reid, R. Thakker, T. Tzanetos, V. Rajur, and G. Chamitoff, "Differential flatness transformations for aggressive quadrotor flight," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5204–5210, IEEE, 2018.

[28] A. Abdessameud and A. Tayebi, "Global trajectory tracking control of VTOL-UAVs without linear velocity measurements," *Automatica*, vol. 46, no. 6, pp. 1053–1059, 2010.

[29] C. B. Low, "A trajectory tracking control design for fixed-wing unmanned aerial vehicles," in *2010 IEEE International Conference on Control Applications*, pp. 2118–2123, IEEE, 2010.

[30] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV for extreme maneuverability," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6337–6342, 2011.

[31] J.-Y. Wen and K. Kreutz-Delgado, “The attitude control problem,” *IEEE Transactions on Automatic control*, vol. 36, no. 10, pp. 1148–1162, 1991.

[32] N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, “Rigid-body attitude control,” *IEEE Control Systems Magazine*, vol. 31, no. 3, pp. 30–51, 2011.

[33] H. Chao, Y. Cao, and Y. Chen, “Autopilots for small fixed-wing unmanned air vehicles: A survey,” in *2007 International Conference on Mechatronics and Automation*, pp. 3144–3149, IEEE, 2007.

[34] L. Meier, D. Honegger, and M. Pollefeys, “Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms,” in *2015 IEEE international conference on robotics and automation (ICRA)*, pp. 6235–6240, IEEE, 2015.

[35] H. Gui and A. H. de Ruiter, “Quaternion invariant extended Kalman filtering for spacecraft attitude estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 4, pp. 863–878, 2018.

[36] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, “A complementary filter for attitude estimation of a fixed-wing UAV,” in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 340–345, IEEE, 2008.

[37] R. A. Cordeiro, R. Azinheira, and A. Moutinho, “Cascaded incremental backstepping controller for the attitude tracking of fixed-wing aircraft,” in *5th CEAS Conference on Guidance, Navigation and Control*, 2019.

[38] V. Cichella, I. Kaminer, V. Dobrokhodov, E. Xargay, N. Hovakimyan, and A. Pascoal, “Geometric 3D path-following control for a fixed-wing UAV on SO(3),” in *AIAA Guidance, Navigation, and Control Conference*, p. 6415, 2011.

[39] J. Forshaw and V. Lappas, “Transitional control architecture, methodology, and robustness for a twin helicopter rotor tailsitter,” in *AIAA Guidance, Navigation, and Control Conference*, p. 4697, 2012.

[40] O. Egeland and J.-M. Godhavn, “Passivity-based adaptive attitude control of a rigid spacecraft,” *IEEE Transactions on Automatic Control*, vol. 39, no. 4, pp. 842–846, 1994.

[41] A. Tayebi and S. McGilvray, "Attitude stabilization of a four-rotor aerial robot," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 2, pp. 1216–1221, IEEE, 2004.

[42] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in *2013 European Control Conference (ECC)*, pp. 3864–3869, IEEE, 2013.

[43] B. Costic, D. Dawson, M. De Queiroz, and V. Kapila, "Quaternion-based adaptive attitude tracking controller without velocity measurements," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 6, pp. 1214–1222, 2001.

[44] N. B. Knoebel and T. W. McLain, "Adaptive quaternion control of a miniature tailsitter UAV," in *2008 American Control Conference*, pp. 2340–2345, IEEE, 2008.

[45] W. E. Green and P. Y. Oh, "A hybrid MAV for ingress and egress of urban environments," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 253–263, 2009.

[46] E. Bulka and M. Nahon, "Automatic control for aerobatic maneuvering of agile fixed-wing UAVs," *Journal of Intelligent & Robotic Systems*, vol. 93, no. 1-2, pp. 85–100, 2019.

[47] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "On quaternion-based attitude control and the unwinding phenomenon," in *Proceedings of the 2011 American Control Conference*, pp. 299–304, IEEE, 2011.

[48] F. Bullo and A. D. Lewis, *Geometric control of mechanical systems: modeling, analysis, and design for simple mechanical control systems*, vol. 49. Springer, 2019.

[49] S. Verling, B. Weibel, M. Boosfeld, K. Alexis, M. Burri, and R. Siegwart, "Full attitude control of a VTOL tailsitter UAV," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3006–3012, IEEE, 2016.

[50] P. Poksawat, L. Wang, and A. Mohamed, "Gain scheduled attitude control of fixed-wing UAV with automatic controller tuning," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1192–1203, 2017.

[51] Y. Kang and J. Hedrick, "Design of nonlinear model predictive controller for a small fixed-wing unmanned aerial vehicle," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 6685, 2006.

[52] E. Oland and R. Kristiansen, "Quaternion-based backstepping control of a fixed-wing unmanned aerial vehicle," in *2013 IEEE Aerospace Conference*, pp. 1–7, IEEE, 2013.

[53] R. Skjetne, "The maneuvering problem," *NTNU, PhD-thesis*, vol. 1, 2005.

[54] M. Breivik and T. I. Fossen, "Principles of guidance-based path following in 2D and 3D," in *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 627–634, IEEE, 2005.

[55] A. P. Aguiar, J. P. Hespanha, and P. V. Kokotović, "Performance limitations in reference tracking and path following for nonlinear systems," *Automatica*, vol. 44, no. 3, pp. 598–610, 2008.

[56] S. Park, J. Deyst, and J. How, "A new nonlinear guidance logic for trajectory tracking," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 4900, 2004.

[57] R. Cory and R. Tedrake, "Experiments in fixed-wing UAV perching," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, p. 7256, 2008.

[58] J. Moore and R. Tedrake, "Powerline perching with a fixed-wing UAV," in *AIAA Infotech@ Aerospace Conference*, p. 1959, 2009.

[59] A. Roberts and A. Tayebi, "Adaptive position tracking of VTOL UAVs," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 129–142, 2010.

[60] Y. Zou, "Trajectory tracking controller for quadrotors without velocity and angular velocity measurements," *IET Control Theory & Applications*, vol. 11, no. 1, pp. 101–109, 2016.

[61] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *49th IEEE conference on decision and control (CDC)*, pp. 5420–5425, IEEE, 2010.

[62] J. Zhou, X. Lyu, Z. Li, S. Shen, and F. Zhang, "A unified control method for quadrotor tail-sitter UAVs in all flight modes: Hover, transition, and level flight," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4835–4841, IEEE, 2017.

[63] R. Ritz and R. D'Andrea, "A global controller for flying wing tailsitter vehicles," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2731–2738, IEEE, 2017.

[64] R. H. Stone, "Control architecture for a tail-sitter unmanned air vehicle," in *2004 5th Asian Control Conference (IEEE Cat. No. 04EX904)*, vol. 2, pp. 736–744, IEEE, 2004.

[65] J. L. Forshaw, V. J. Lappas, and P. Briggs, "Transitional control architecture and methodology for a twin rotor tailsitter," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 4, pp. 1289–1298, 2014.

[66] S. Verling, T. Stastny, G. Bättig, K. Alexis, and R. Siegwart, "Model-based transition optimization for a VTOL tailsitter," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3939–3944, IEEE, 2017.

[67] E. Bulka and M. Nahon, "Autonomous control of agile fixed-wing UAVs performing aerobatic maneuvers," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 104–113, IEEE, 2017.

[68] J. M. Levin, M. Nahon, and A. A. Paranjape, "Real-time motion planning with a fixed-wing UAV using an agile maneuver space," *Autonomous Robots*, vol. 43, no. 8, pp. 2111–2130, 2019.

[69] R. W. Beard, J. Ferrin, and J. Humpherys, "Fixed wing UAV path following in wind with input constraints," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2103–2117, 2014.

[70] A. Rucco, A. P. Aguiar, F. L. Pereira, and J. B. de Sousa, "A predictive path-following approach for fixed-wing unmanned aerial vehicles in presence of wind disturbances," in *Robot 2015: Second Iberian Robotics Conference*, pp. 623–634, Springer, 2016.

[71] T. McLain, R. W. Beard, and M. Owen, "Implementing Dubins airplane paths on fixed-wing UAVs," in *Handbook of unmanned aerial vehicles*, vol. 2077, pp. 1677 – 1701, Springer, 2014.

[72] K. Tanaka, M. Tanaka, Y. Takahashi, A. Iwase, and H. O. Wang, "3-D flight path tracking control for unmanned aerial vehicles under wind environments," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 12, pp. 11621–11634, 2019.

[73] A. Gray, E. Abbena, and S. Salamon, *Modern differential geometry of curves and surfaces with Mathematica®*. Chapman and Hall/CRC, 2017.

[74] A. J. Hanson and H. Ma, "Parallel transport approach to curve framing," *Indiana University, Techreports-TR425*, vol. 11, pp. 3–7, 1995.

[75] G. Zagainov, "High maneuverability. theory and practice," in *Space Programs and Technologies Conference and Exhibit*, p. 4737, 1993.

[76] J. Y. Chen, "UAV-guided navigation for ground robot tele-operation in a military reconnaissance environment," *Ergonomics*, vol. 53, no. 8, pp. 940–950, 2010.

[77] R. Hosman and H. Stassen, "Pilot's perception in the control of aircraft motions," *Control engineering practice*, vol. 7, no. 11, pp. 1421–1428, 1999.

[78] P. Stegagno, M. Basile, H. H. Bülthoff, and A. Franchi, "A semi-autonomous UAV platform for indoor remote operation with visual and haptic feedback," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3862–3869, IEEE, 2014.

[79] Á. Zarándy, M. Nemeth, Z. Nagy, A. Kiss, L. Santha, and T. Zsedrovits, "A real-time multi-camera vision system for UAV collision warning and navigation," *Journal of Real-Time Image Processing*, vol. 12, no. 4, pp. 709–724, 2016.

[80] P. Chudỳ, A. Tomczyk, and P. Rzucidlo, "Safety enhanced digital flight control system," *Aircraft Engineering and Aerospace Technology*, 2009.

[81] J. A. Mendoza-Mendoza, V. Gonzalez-Villela, G. Sepulveda-Cervantes, M. Mendez-Martinez, and H. Sossa-Azuela, "ArduPilot working environment," in *Advanced Robotic Vehicles Programming*, pp. 19–46, Springer, 2020.

[82] S. Akyurek, U. Kaynak, and C. Kasnakoglu, "Altitude control for small fixed-wing aircraft using $\mathcal{H}_\infty$ loop-shaping method," *IFAC-PapersOnLine*, vol. 49, no. 9, pp. 111–116, 2016.

[83] J. Moynes and J. Gallagher, "Flight control system design for ride qualities of highly maneuverable fighter aircraft," *AGARD Guidance and Control Design Considerations for Low-Altitude and Terminal-Area Flight 21 p(SEE N 78-26049 17-01)*, 1978.

[84] W. Khan, *Dynamics Modeling of Agile Fixed-Wing Unmanned Aerial Vehicles.* PhD thesis, McGill University, Canada, 2016.

[85] A. Noth, S. Bouabdallah, and R. Siegwart, "Dynamic modeling of fixed-wing UAVs," *Swiss Federal Institute of Technology*, vol. 2, 2006.

[86] B. W. McCormick, *Aerodynamics, aeronautics, and flight mechanics.* Wiley, 1995.

[87] E. Bulka, *Control and Obstacle Avoidance for Agile Fixed-Wing Aircraft.* PhD thesis, McGill University, Canada, 2021.

[88] B. C. Hall *et al.*, *Lie groups, Lie algebras, and representations: an elementary introduction*, vol. 10. Springer, 2003.

[89] S. P. Bhat and D. S. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Systems & Control Letters*, vol. 39, no. 1, pp. 63–70, 2000.

[90] J. Stuelpnagel, "On the parametrization of the three-dimensional rotation group," *SIAM review*, vol. 6, no. 4, pp. 422–430, 1964.

[91] J. C. H. Ramírez and M. Nahon, "Nonlinear vector-projection control for agile fixed-wing unmanned aerial vehicles," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5314–5320, IEEE, 2020.

[92] J. K. Hall, N. B. Knoebel, and T. W. McLain, "Quaternion attitude estimation for miniature air vehicles using a multiplicative extended Kalman filter," in *Proceedings of IEEE/ION PLANS 2008*, pp. 1230–1237, 2008.

[93] D. Yazell, "Origins of the unusual space shuttle quaternion definition," in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, p. 43, 2009.

[94] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "On the non-robustness of inconsistent quaternion-based attitude control systems using memoryless path-lifting schemes," in *Proceedings of the 2011 American Control Conference*, pp. 1003–1008, IEEE, 2011.

[95] T. Lee, "Exponential stability of an attitude tracking control system on SO (3) for large-angle rotational maneuvers," *Systems & Control Letters*, vol. 61, no. 1, pp. 231–237, 2012.

[96] D. E. Zlotnik and J. R. Forbes, "Exponential convergence of a nonlinear attitude estimator," *Automatica*, vol. 72, pp. 11–18, 2016.

[97] F. Goodarzi, D. Lee, and T. Lee, "Geometric nonlinear PID control of a quadrotor UAV on SE (3)," in *2013 European Control Conference (ECC)*, pp. 3845–3850, IEEE, 2013.

[98] J.-J. E. Slotine and W. Li, “On the adaptive control of robot manipulators,” *The International Journal of Robotics Research*, vol. 6, no. 3, pp. 49–59, 1987.

[99] H. Khalil, *Nonlinear Systems, third ed.* Prentice Hall, 2001.

[100] P. C. Hughes, *Spacecraft attitude dynamics.* Courier Corporation, 2012.

[101] J. C. Hernandez Ramirez and M. Nahon, “Trajectory tracking control of highly maneuverable fixed-wing unmanned aerial vehicles,” in *AIAA Scitech 2020 Forum*, p. 2074, 2020.

[102] T. Lee, M. Leok, and N. H. McClamroch, “Nonlinear robust tracking control of a quadrotor UAV on SE (3),” *Asian Journal of Control*, vol. 15, no. 2, pp. 391–408, 2013.

[103] K. Gamagedara, M. Bisheban, E. Kaufman, and T. Lee, “Geometric controls of a quadrotor UAV with decoupled yaw control,” in *2019 American Control Conference (ACC)*, pp. 3285–3290, IEEE, 2019.

[104] F. Bullo, R. M. Murray, and A. Sarti, “Control on the sphere and reduced attitude stabilization,” *IFAC Proceedings Volumes*, vol. 28, no. 14, pp. 495–501, 1995.

[105] S. Park, “Autonomous aerobatics on commanded path,” *Aerospace Science and Technology*, vol. 22, no. 1, pp. 64–74, 2012.

[106] J. C. H. Ramírez and M. Nahon, “Pilot-assist landing system for hover-capable fixed-wing unmanned aerial vehicles in all flight regimes,” in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1179–1186, IEEE, 2021.

[107] M. Spivak, *A comprehensive introduction to differential geometry*, vol. 5. Publish or Perish, Incorporated, 1975.

[108] D. Carroll, E. Köse, and I. Sterling, “Improving Frenet’s frame using Bishop’s frame,” *arXiv preprint arXiv:1311.5857*, 2013.

[109] J. C. H. Ramírez and M. Nahon, “A gravity-referenced moving frame for vehicle path following applications in 3D,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4393–4400, 2021.

[110] R. L. Bishop, “There is more than one way to frame a curve,” *The American Mathematical Monthly*, vol. 82, no. 3, pp. 246–251, 1975.

[111] E. M. Coates, D. Reinhardt, and T. I. Fossen, “Reduced-attitude control of fixed-wing unmanned aerial vehicles using geometric methods on the two-sphere,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 5749–5756, 2020.

[112] J. Smith, J. Su, C. Liu, and W.-H. Chen, “Disturbance observer based control with anti-windup applied to a small fixed wing UAV for disturbance rejection,” *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2, pp. 329–346, 2017.

Appendix

A Attitude control proofs

A.1 Proof of Lemma 3.2

Function $\Psi_1(\mathbf{C}_{br})$

From simple derivation,

$$\dot{\Psi}_1 = -\frac{1}{2}\text{tr}(\dot{\mathbf{C}}_{br}) = \frac{1}{2}\text{tr}(\mathbf{e}_\omega^\times \mathbf{C}_{br}) = -\mathbf{e}_\omega^T \mathcal{P}_a(\mathbf{C}_{br})^\vee = \mathbf{e}_\omega^T \mathbf{e}_1,$$

where (3.7) and the property $\text{tr}(\mathbf{x}^\times \mathbf{A}) = -2\mathbf{x}^T \mathcal{P}_a(\mathbf{A})^\vee$, for any $\mathbf{x} \in \mathbb{R}^3$, $\mathbf{A} \in \mathbb{R}^{3\times 3}$ was used. This verifying i) while also verifying that the critical points of Ψ_1 satisfy $\mathbf{e}_1 = \mathbf{0}$.

The trace of any $\mathbf{C} \in \text{SO}(3)$ is bounded: $-1 \leq \text{tr}(\mathbf{C}) \leq 3$. Since the trace is a linear operator, $0 \leq \Psi_1 \leq 2$, where $\Psi_1 = 0$ only at $\mathbf{C}_{br} = \mathbf{I}_3$ which shows ii).

Analyzing the critical points, from (3.9), $\mathbf{e}_1 = \mathbf{0}$ implies $\mathbf{C}_{br} = \mathbf{C}_{br}^T$. Using (2.5), this translates to

$$\cos(\eta)\mathbf{I}_3 + (1-\cos(\phi))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T - \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times = \cos(\eta)\mathbf{I}_3 + (1-\cos(\phi))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T + \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times,$$

which results in $\sin(\eta) = -\sin(\eta) \Rightarrow \sin(\eta) = 0 \Rightarrow \eta = \{0, \pm n\pi\}$, for odd values of n. The first value, $\eta = 0$, corresponds to $\mathbf{C}_{br} = \mathbf{C}_{ri}$, while the second set of values, $\eta = \pm\pi$ corresponds to an attitude of the body 180° opposed to the reference. The second set is not desirable, but this solution is not in $\boldsymbol{\Omega}$: using this value of η in (2.5), and substituting it in the definition (3.8) we obtain

$$\begin{aligned}\Psi_1 &= \frac{1}{2}\text{tr}\left(\mathbf{I}_3 - (\cos(\eta)\mathbf{I}_3 + (1-\cos(\phi))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T - \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times)\right) \\ &= \frac{1}{2}\text{tr}\left(2\mathbf{I}_3 - 2\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T\right) = 3 - 1 = 2,\end{aligned}$$

which is outside the set Ω_1. This verifies iii).

In general, for any value of η, the above equation is

$$\Psi_1 = \frac{1}{2}\mathrm{tr}\left(\mathbf{I}_3 - (\cos(\eta)\mathbf{I}_3 + (1-\cos(\phi))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T - \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times)\right) = 1-\cos(\eta).$$

Similarly, the innovation term can also be written in terms of η and $\hat{\boldsymbol{\epsilon}}$ through (2.5) as

$$\begin{aligned}\mathbf{e}_1 &= \frac{1}{2}(2\sin(\eta))\hat{\boldsymbol{\epsilon}}\\ \|\mathbf{e}_1\|^2 &= \sin^2(\eta).\end{aligned}$$

which shows v) for $\bar{e}_1 = 1$.

Now consider

$$\frac{\|\mathbf{e}_1\|^2}{\Psi_1} = \frac{\sin^2(\eta)}{1-\cos(\eta)},$$

The function on the right is well defined for every value except $\eta = 0$ (and multiples of 2π which represent the same attitude condition), for which both bounds in iv) are trivially satisfied. The limit, however, is well defined, such that for any other value,

$$0 \leq \frac{\sin^2(\eta)}{1-\cos(\eta)} \leq 2.$$

This immediately shows the lower bound is satisfied by $\frac{1}{2}\|\mathbf{e}_1\|^2 \leq \Psi_1$ (hence $\underline{b}_1 = 1/2$). The reciprocal function is not defined $\eta = 0$, but again, the bound is trivially satisfied. Consider a a positive constant $0 < \psi_1 < 2$. Then an upper bound can be determined through

$$\frac{\Psi}{\|\mathbf{e}_1\|^2} = \frac{1-\cos(\eta)}{\sin^2(\eta)} \leq \frac{1}{2-\psi_1},$$

then the upper bound of iv) is verified with $\bar{b}_1 = \frac{1}{2-\psi_1}$.

Finally, by differentiating $\mathbf{e}_1$,

$$\begin{aligned}\dot{\mathbf{e}}_1 &= -\frac{1}{2}\left(\dot{\mathbf{C}}_{br} - \dot{\mathbf{C}}_{br}^T\right)^\vee = \frac{1}{2}\left(\mathbf{e}_\omega^\times\mathbf{C}_{br} + \mathbf{C}_{br}\mathbf{e}_\omega^\times\right)^\vee\\ &= \frac{1}{2}\left(\mathrm{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br}\right)\mathbf{e}_\omega \triangleq \mathbf{E}_1\mathbf{e}_\omega\end{aligned}$$

where the properties $\mathbf{x}^\times\mathbf{A} + \mathbf{A}^T\mathbf{x}^\times = ((\mathrm{tr}(\mathbf{A})\mathbf{I}_3 - \mathbf{A})\,\mathbf{x})^\times$, for all $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{A} \in \mathbb{R}^{3\times3}$, s.t. $\mathbf{A} = \mathbf{A}^T$, and $(\mathbf{u}^\times)^\vee = \mathbf{u}$, for all $\mathbf{u} \in \mathbb{R}^3$, were used. Clearly, the matrix $\mathbf{E}_1$ is bounded, since it depends only on elements of SO(3). A more precise, if conservative, bound can be found using the Frobenius norm

$$\begin{aligned}\|\mathbf{E}_1\|_F^2 &= \mathrm{tr}(\mathbf{E}_1^T\mathbf{E}_1)\\ &= \frac{1}{4}\mathrm{tr}\left(\mathrm{tr}(\mathbf{C}_{br})^2\mathbf{I}_3 - \mathrm{tr}(\mathbf{C}_{br})(\mathbf{C}_{br} + \mathbf{C}_{br}^T) + \mathbf{C}_{br}\mathbf{C}_{br}^T\right)\\ &= \frac{1}{4}\left(3\,\mathrm{tr}(\mathbf{C}_{br})^2 - 2\,\mathrm{tr}(\mathbf{C}_{br})^2 + 3\right)\\ &= \frac{1}{4}\left(\mathrm{tr}(\mathbf{C}_{br})^2 + 3\right)\\ &\leq \frac{1}{4}(3^2 + 3),\end{aligned}$$

then

$$\|\mathbf{E}_1\|_F \leq \sqrt{3}.$$

Since $\|\mathbf{E}_1\|_2 \leq \|\mathbf{E}_1\|_F$, then vi) is satisfied with $p_1 = \sqrt{3}$.

Function $\Psi_2(\mathbf{C}_{br})$

Property i) is verified through simple derivation:

$$\begin{aligned}\dot{\Psi}_2 &= -\frac{1}{2}\frac{1}{\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}\mathrm{tr}(\dot{\mathbf{C}}_{br}) = \frac{1}{2}\frac{1}{\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}\mathrm{tr}(\mathbf{e}_\omega^\times\mathbf{C}_{br})\\ &= -\frac{1}{\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}\mathbf{e}_\omega^T\mathcal{P}_a(\mathbf{C}_{br})^\vee = \mathbf{e}_\omega^T\mathbf{e}_2.\end{aligned}$$

Again, the critical points of Ψ_2 satisfy $\mathbf{e}_2 = \mathbf{0}$.

As shown previously, the term $\mathrm{tr}(\mathbf{C}_{br})$ reaches its maximum value, 3, only when $\mathbf{C}_{br} = \mathbf{I}_3$. Simple substitution verifies that $\Psi_2 = 0$ at this value, and $\Psi_2 > 0$ for any other value, which shows ii).

Critical point analysis is identical to the previous function, with the exception that $\mathbf{e}_2$ is undefined when $\mathrm{tr}(\mathbf{C}_{br}) = -1$. As shown before, this corresponds to a relative rotation of 180° between body and reference frames, or a principal angle of $\eta = n\pi$ for odd values of n. Again, this condition is outside the set Ω:

$$\Psi_2 = 2 - \sqrt{1 + (-1)} = 2,$$

showing iii).

To obtain the bounds in iv), Rodrigues formula (2.5) is used

$$\begin{aligned}\Psi_2 &= 2 - \sqrt{1+1+2\cos(\eta)} = 2 - \sqrt{2(1+\cos(\eta))} = 2 - 2\sqrt{\frac{1+\cos(\eta)}{2}} \\ &= 2 - 2\cos\left(\frac{\eta}{2}\right) = 4\left(\frac{1-\cos(\eta/2)}{2}\right) = 4\sin^2\left(\frac{\eta}{4}\right),\end{aligned}$$

and

$$\mathbf{e}_2 = \frac{\sin(\eta)}{\sqrt{2(1+\cos(\eta))}}\hat{\boldsymbol{\epsilon}},$$

then

$$\begin{aligned}\|\mathbf{e}_2\|^2 &= \frac{\sin^2(\eta)}{2(1+\cos(\eta))} = \frac{1-\cos^2(\eta)}{2(1+\cos(\eta))} = \frac{(1-\cos(\eta))(1+\cos(\eta))}{2(1+\cos(\eta))} \\ &= \frac{1-\cos(\eta)}{2} = \sin^2\left(\frac{\eta}{2}\right) = 1-\cos^2\left(\frac{\eta}{2}\right) = 4\left(\frac{1-\cos(\eta/2)}{2}\right)\left(\frac{1+\cos(\eta/2)}{2}\right) \\ &= 4\,\sin^2\left(\frac{\eta}{4}\right)\cos^2\left(\frac{\eta}{4}\right).\end{aligned}$$

which also shows v), for $\bar{e}_2 = 1$. Consider

$$\frac{\Psi_2}{\|\mathbf{e}_2\|^2} = \cos^2\left(\frac{\eta}{4}\right) \leq 1$$

then $\|\mathbf{e}_2\|^2 \leq \Psi_2$ for any element in Ω, except when $\eta = 0$, for which the inequality is trivial. On the other hand,

$$\frac{\|\mathbf{e}_2\|^2}{\Psi_2} = \frac{1}{\cos^2\left(\frac{\eta}{4}\right)} = \frac{2}{1+\cos\left(\frac{\eta}{2}\right)} \leq 2,$$

then $\Psi_2 \leq 2\|\mathbf{e}_2\|^2$, which holds, again, for every element in Ω except the trivial solution. This shows iv), with $\underline{b}_2 = 1$ and $\bar{b}_2 = 2$.

Finally, the time derivative of the innovation term is

$$\begin{aligned}
\dot{\mathbf{e}}_2 &= -\frac{1}{2}\frac{-\frac{1}{2}\mathrm{tr}(-\mathbf{e}_\omega^\times \mathbf{C}_{br})}{(1+\mathrm{tr}(\mathbf{C}_{br}))^{3/2}}(\mathbf{C}_{br}-\mathbf{C}_{br}^T)^\vee + \frac{1}{2\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}(\mathbf{e}_\omega^\times \mathbf{C}_{br} + \mathbf{C}_{br}^T\mathbf{e}_\omega^\times)^\vee \\
&= -\frac{1}{2}\frac{\mathrm{tr}(\mathbf{e}_\omega^\times \mathbf{C}_{br})}{(1+\mathrm{tr}(\mathbf{C}_{br}))}\mathbf{e}_2 + \frac{1}{2\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}(\mathrm{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br})\mathbf{e}_\omega \\
&= \frac{1}{2}\frac{\mathbf{e}_\omega^T(\mathbf{C}_{br}-\mathbf{C}_{br}^T)^\vee}{(1+\mathrm{tr}(\mathbf{C}_{br}))}\mathbf{e}_2 + \frac{1}{2\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}(\mathrm{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br})\mathbf{e}_\omega \\
&= \mathbf{e}_\omega^T\left(\frac{1}{\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}\mathbf{e}_2\right)\mathbf{e}_2 - \frac{1}{2\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}(\mathrm{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br})\mathbf{e}_\omega \\
&= -\frac{1}{2\sqrt{1+\mathrm{tr}(\mathbf{C}_{br})}}(2\mathbf{e}_2\mathbf{e}_2^T + \mathrm{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br})\mathbf{e}_\omega \\
&\triangleq \mathbf{E}_2\mathbf{e}_\omega.
\end{aligned}$$

Since $\mathbf{E}_2$ depends only on $\mathbf{C}_{br}$, it is, by necessity, bounded. More specifically, using (2.5) it can be shown that

$$\mathbf{E}_2 = \frac{1}{2\sqrt{2(1+\cos(\eta))}}\left((1+\cos(\eta))\mathbf{I}_3 + \sin(\eta)\hat{\boldsymbol{\epsilon}}^\times\right),$$

then

$$\begin{aligned}
\mathbf{E}_2^T\mathbf{E}_2 &= \frac{1}{4}\frac{1}{2(1+\cos(\eta))}\left((1+\cos(\eta))^2\mathbf{I}_3 - \sin^2(\eta)\hat{\boldsymbol{\epsilon}}^\times\hat{\boldsymbol{\epsilon}}^\times\right) \\
&= \frac{1}{4}\frac{1}{2(1+\cos(\eta))}\left((1+\cos(\eta))^2\mathbf{I}_3 + \sin^2(\eta)\mathbf{I}_3 - \sin^2(\eta)\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T\right) \\
&= \frac{1}{4}\frac{1}{2(1+\cos(\eta))}\left(2(1+\cos(\eta))\mathbf{I}_3 - (1+\cos(\eta))(1-\cos(\eta))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T\right) \\
&= \frac{1}{4}\left(\mathbf{I}_3 - \frac{1}{2}(1-\cos(\eta))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T\right)
\end{aligned}$$

It can be verified that the eigenvalues of this matrix are $\{1/4,\ 1/4,\ 1/8(1+\cos(\eta))\}$. The maximum eigenvalue is thus 1/4, hence the induced 2-norm of $\mathbf{E}_2$, $\|\mathbf{E}_2\|_2 = \max\left(\sqrt{\lambda(\mathbf{E}_2^T\mathbf{E}_2)}\right) = 1/2$. Then

$$\|\dot{\mathbf{e}}_2\| \leq \frac{1}{2}\|\mathbf{e}_\omega\|,$$

which shows vi), for $p_2 = 1/2$.

Function $\Psi_3(\mathbf{C}_{br})$

Again, property i) can be shown through derivation,

$$\dot{\Psi}_3 = -\frac{1}{2}\frac{\mathrm{tr}(\dot{\mathbf{C}}_{br})}{1+\mathrm{tr}(\mathbf{C}_{br})} = -\frac{1}{2}\frac{\mathrm{tr}(-\mathbf{e}_\omega^\times \mathbf{C}_{br})}{1+\mathrm{tr}(\mathbf{C}_{br})} = -\frac{\mathbf{e}_\omega^T \mathcal{P}_a(\mathbf{C}_{br})}{1+\mathrm{tr}(\mathbf{C}_{br})} = \mathbf{e}_\omega^T \mathbf{e}_3.$$

This also verifies that the critical points of Ψ_3 verify $\mathbf{e}_3 = \mathbf{0}$.

The function $\Psi_3 = 0$ only when $\mathrm{tr}(\mathbf{C}_{br}) = 3$, the highest possible value for this term. As shown previously, this corresponds to $\mathbf{C}_{br} = \mathbf{I}_3$. Unlike the previous two functions, Ψ_3 is not upper bounded, as it tends to infinity at a relative rotation of $n\pi$ for odd n. regardless, property ii) is verified, since an upper limit is not required for a function to be positive definite.

When $\mathbf{e}_3 = \mathbf{0}$, the critical points, only occur at $\mathbf{C}_{br} = \mathbf{I}$ and $\mathbf{C}_{br} = \mathbf{C}_{br}(\eta = n\pi, \hat{\boldsymbol{\epsilon}}$ for odd values of n. This second condition is not contained in the set Ω, which then shows iii).

Using (2.5), the error function can be written as

$$\Psi_3 = \ln\left(\frac{2}{\sqrt{1+\mathrm{tr}(\mathbf{C}_{br}}}\right) = \ln\left(\frac{2}{\sqrt{2}\sqrt{1+\cos\eta}}\right) = \frac{1}{2}\ln\left(\frac{2}{1+\cos(\eta)}\right),$$

and the innovation term as

$$\mathbf{e}_3 = \frac{\sin(\eta)}{2(1+\cos(\eta))}\hat{\boldsymbol{\epsilon}},$$

then

$$\|\mathbf{e}_3\|_2^2 = \mathbf{e}_3^T\mathbf{e}_3 = \frac{\sin^2(\eta)}{4(1+\cos(\eta))^2}.$$

Note that

$$\begin{aligned}\frac{2}{1+\cos(\eta)} &= \frac{2+2\cos(\eta)}{(1+\cos(\eta))^2} = \frac{(1+2\cos(\eta)+\cos^2(\eta))+(1-\cos^2(\eta))}{(1+\cos(\eta))^2}\\ &= 1+\frac{\sin^2(\eta}{(1+\cos(\eta))^2} = 1+4\|\mathbf{e}_3\|^2,\end{aligned}$$

which results in

$$\Psi_3 = \frac{1}{2}\ln(1+4\|\mathbf{e}_3\|^2),$$

then

$$\|\mathbf{e}_3\|^2 = \frac{e^{2\Psi_3}-1}{4}.$$

The function on the right is strictly increasing with respect to Ψ_3, so for $\Psi_3 = \psi_3 < \infty$,

$$\|\mathbf{e}_3\|^2 \leq \frac{e^{2\psi_3} - 1}{4},$$

which verifies v) for $\bar{e}_3 = \frac{1}{2}\sqrt{e^{2\psi_3} - 1}$.

Consider the function

$$\frac{\Psi_3}{\|\mathbf{e}_3\|^2} = \frac{4\Psi_3}{e^{2\Psi_3} - 1},$$

shown in Figure A.1. The function is well defined everywhere in Ω, and is strictly decreasing. Inside Ω_3, for any $\Psi_3 \leq \psi_3$, then $4\Psi_3/(e^{2\Psi} - 1) \geq 4\psi/(e^{2\psi} - 1)$, and $4\Psi_3/(e^{2\Psi_3} - 1) \leq 2$. Then

$$\begin{aligned} \frac{4\psi_3}{e^{2\psi_3} - 1} \leq \frac{\Psi_3}{\|\mathbf{e}_3\|^2} &\leq 2 \\ \frac{4\psi_3}{e^{2\psi_3} - 1}\|\mathbf{e}_3\|^2 \leq \quad \Psi_3 \quad &\leq 2\|\mathbf{e}_3\|^2, \end{aligned}$$

which shows iv) for $\underline{b}_3 = \frac{4\psi_3}{e^{2\psi 3} - 1}$ and $\bar{b}_3 = 2$.

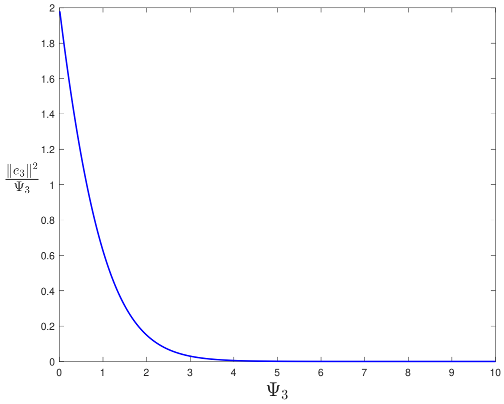


Figure A.1: Upper and lower bounds of Ψ_3.

To show v), the time derivative of $\mathbf{e}_3$ is required. This function is similar to the previous innovation term, and using the same procedure and properties, it can be show that

$$\dot{\mathbf{e}}_3 = \frac{1}{2(1 + \operatorname{tr}(\mathbf{C}_{br}))}(2(1 + \operatorname{tr}(\mathbf{C}_{br}))\mathbf{e}_3\mathbf{e}_3^T + \operatorname{tr}(\mathbf{C}_{br})\mathbf{I}_3 - \mathbf{C}_{br})\mathbf{e}_\omega \triangleq \mathbf{E}_3\mathbf{e}_\omega.$$

Once again, $\mathbf{E}_3$ is bounded, since it depends only on $\mathbf{C}_{br}$, and using (2.5) and the same procedure as before it can be shown that

$$\mathbf{E}_3 = \frac{1}{16(1+\cos(\eta))^2}\left((1+\cos(\eta))^2\mathbf{I}_3 - \sin^2(\eta)\hat{\boldsymbol{\epsilon}}^{\times}\right),$$

then

$$\mathbf{E}_3^T\mathbf{E}_3 = \frac{1}{8(1+\cos(\eta))}\left(\mathbf{I}_3 - \frac{1}{2}(1-\cos(\eta))\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^T\right)$$

The eigenvalues of $\mathbf{E}_3^T\mathbf{E}_3$ are non zero in the set Ω, and are $\{1/8(1+\cos(\eta))^{-1}, 1/8(1+\cos(\eta))^{-1}, 1/16\}$. Since

$$\|\dot{\mathbf{e}}_3\| \leq \|\mathbf{E}_3\|_2\|\mathbf{e}_\omega\| = \max\left(\sqrt{\lambda(\mathbf{E}_2^T\mathbf{E}_2)}\right)\|\mathbf{e}_\omega\|,$$

then vi) is verified for $p_3 = \frac{1}{16}\max\{1,\, e^{2c}\}$. Thus all properties of Lemma 3.2 are verified.

A.2 Proof of Proposition 3.3

The attitude error dynamics can be derived by substituting (2.2) and (3.1) in the angular velocity error definition (3.6).

$$\begin{aligned}
\mathbf{J}\dot{\mathbf{e}}_\omega &= \mathbf{J}\left(\dot{\boldsymbol{\omega}}_b - \frac{d}{dt}(\mathbf{C}_{br}\boldsymbol{\omega}_r)\right) = \mathbf{J}\dot{\boldsymbol{\omega}}_b - \mathbf{J}\left(-\mathbf{e}_\omega^{\times}\mathbf{C}_{br}\boldsymbol{\omega}_r + \mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r\right) \\
&= (\mathbf{J}\boldsymbol{\omega}_b)^{\times}\boldsymbol{\omega}_b + \mathbf{M} + \mathbf{J}\mathbf{e}_\omega^{\times}\mathbf{C}_{br}\boldsymbol{\omega}_r - \mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r \\
&= (\mathbf{J}(\mathbf{e}_\omega + \mathbf{C}_{br}\boldsymbol{\omega}_r))^{\times}(\mathbf{e}_\omega + \mathbf{C}_{br}\boldsymbol{\omega}_r)) + \mathbf{J}\mathbf{e}_\omega^{\times}\mathbf{C}_{br}\boldsymbol{\omega}_r - \mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r + \mathbf{M} \\
&= (\mathbf{J}\mathbf{e}_\omega)^{\times}\mathbf{e}_\omega + (\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}\mathbf{e}_\omega + (\mathbf{J}\mathbf{e}_\omega)^{\times}\mathbf{C}_{br}\boldsymbol{\omega}_r + (\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}(\mathbf{C}_{br}\boldsymbol{\omega}_r) \\
&+ \mathbf{J}\mathbf{e}_\omega^{\times}\mathbf{C}_{br}\boldsymbol{\omega}_r - \mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r + \mathbf{M} \\
&= (\mathbf{J}\mathbf{e}_\omega)^{\times}\mathbf{e}_\omega + \left[(\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times} - (\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}\mathbf{J} - (\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times})\mathbf{J}\right]\mathbf{e}_\omega \\
&- \left(\mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r + (\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}(\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r)\right) + \mathbf{M}
\end{aligned}$$

since

$$(\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}\mathbf{J} + \mathbf{J}(\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times} = ((\mathrm{tr}(\mathbf{J})\mathbf{I}_3 - \mathbf{J})\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}$$

then

$$\mathbf{J}\dot{\mathbf{e}}_\omega = (\mathbf{J}\mathbf{e}_\omega)^{\times}\mathbf{e}_\omega + (2\mathbf{J} - \mathrm{tr}(\mathbf{J})\mathbf{I}_3)\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}\mathbf{e}_\omega - \left(\mathbf{J}\mathbf{C}_{br}\dot{\boldsymbol{\omega}}_r + (\mathbf{C}_{br}\boldsymbol{\omega}_r)^{\times}(\mathbf{J}\mathbf{C}_{br}\boldsymbol{\omega}_r)\right) + \mathbf{M}.$$

Finally, substituting the definitions (3.20) and (3.21) arrives to the main result (3.19), completing the proof.

A.3 Proof of Lemma 3.4

Substituting the control law (3.24) in the error dynamics (3.19) gives

$$\mathbf{J}\dot{\mathbf{e}}_\omega = \left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1}\right)^\times \mathbf{e}_\omega - k_\omega \mathbf{e}_\omega - k_a \mathbf{e}_a.$$

Consider the Lyapunov candidate function

$$\bar{V} = \frac{1}{2}\mathbf{e}_\omega^T \mathbf{J}\mathbf{e}_\omega + k_a \Psi_a.$$

The function is positive definite for all three attitude error functions due to Lemma 3.2. Since the inertia matrix is constant, its time derivative over the system trajectories is

$$\begin{aligned}\dot{\bar{V}} &= \mathbf{e}_\omega^T \mathbf{J}\dot{\mathbf{e}}_\omega + \dot{\Psi}_a = \mathbf{e}_\omega^T \left(\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1}\right)^\times \mathbf{e}_\omega - \mathbf{e}_\omega^T k_\omega \mathbf{e}_\omega - \mathbf{e}_\omega^T k_a \mathbf{e}_a + \mathbf{e}_\omega^T k_a \mathbf{e}_a \\ &= -k_\omega \mathbf{e}_\omega^T \mathbf{e}_\omega = -k_\omega \|\mathbf{e}_\omega\|^2 \leq 0\end{aligned}$$

Since $\dot{\bar{V}}$ is negative semidefinite, then $\bar{V}(t) \leq \bar{V}(0)$ for all $t > 0$. From the definition of $\bar{V}$

$$k_a \Psi_a(t) = \bar{V}(t) - \frac{1}{2}\mathbf{e}_\omega^T \mathbf{J}\mathbf{e}_\omega \leq \bar{V}(t) + \lambda_M(\mathbf{J})\|\mathbf{e}_\omega\|^2 \leq \bar{V}(t)$$

and

$$\bar{V}(0) = \frac{1}{2}\mathbf{e}_\omega^T(0)\mathbf{J}\mathbf{e}_\omega(0) + k_a \Psi(0) \leq \frac{\lambda_M(\mathbf{J})}{2}\|\mathbf{e}_\omega(0)\|^2 + k_a \Psi(0)$$

if the initial condition for the angular error is given by

$$\|\mathbf{e}_\omega(0)\|^2 < \frac{2\,k_a}{\lambda_M(\mathbf{J})}\left(\psi_a - \Psi(0)\right), \tag{A.1}$$

then

$$k_a \Psi_a(t) \leq \bar{V}(t) \leq \bar{V}(0) < \frac{\lambda_M(\mathbf{J})}{2}\frac{2\,k_a}{\lambda_M(\mathbf{J})}\left(\psi_a - \Psi(0)\right) + k_a \Psi(0) = k_a \psi_a$$

which results in $\Psi_a \leq \psi_a$, for all $t > 0$. All trajectories starting in Ω_a stay in the set for all time. This avoids the undefined conditions in the innovation terms $\mathbf{e}_2$ and $\mathbf{e}_3$, and the undesirable equilibrium that occurs at $\Psi_1 = 2$.

A.4 Proof of Theorem 3.5

Consider the Lyapunov candidate function

$$V = \bar{V} + \gamma\, \mathbf{e}_\omega^T \mathbf{e}_a = \frac{1}{2}\mathbf{e}_\omega^T \mathbf{J} \mathbf{e}_\omega + k_a \Psi_a + \gamma\, \mathbf{e}_\omega^T \mathbf{e}_a, \tag{A.2}$$

for some $\gamma > 0$ to be determined later. Owing to Lemma 3.2, V is positive definite and is both upper and lower bounded. Indeed, since

$$\begin{aligned}
\lambda_m(\mathbf{J})\|\mathbf{e}_\omega\|^2 \leq \mathbf{e}_\omega^T \mathbf{J} \mathbf{e}_\omega &\leq \lambda_M(\mathbf{J})\|\mathbf{e}_\omega\|^2, && \text{(Quadratic form)} \\
\underline{b}_a\|\mathbf{e}_a\|^2 \leq \quad \Psi_a \quad &\leq \bar{b}_a\|\mathbf{e}_a\|^2, && \text{(Lemma 3.2 (iv))} \\
-\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| \leq \quad \mathbf{e}_\omega^T \mathbf{e}_a \quad &\leq \|\mathbf{e}_\omega\|\|\mathbf{e}_a\|, && \text{(Cauchy-Schwarz)}
\end{aligned}$$

then

$$\frac{\lambda_m(\mathbf{J})}{2}\|\mathbf{e}_\omega\|^2 + k_a\underline{b}_a\|\mathbf{e}_a\|^2 - \gamma\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| \leq V \leq \frac{\lambda_M(\mathbf{J})}{2}\|\mathbf{e}_\omega\|^2 + k_a\bar{b}_a\|\mathbf{e}_a\|^2 + \gamma\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|$$

or, in matrix form,

$$\mathbf{x}^T \mathbf{W}_1 \mathbf{x} \leq V \leq \mathbf{x}^T \mathbf{W}_2 \mathbf{x} \tag{A.3}$$

where $\mathbf{x} = [\|\mathbf{e}_\omega\|,\, \|\mathbf{e}_\omega\|]^T$ and

$$\mathbf{W}_1 = \begin{bmatrix} k_a\underline{b}_a & -\frac{1}{2}\gamma \\ -\frac{1}{2}\gamma & \frac{1}{2}\lambda_m(\mathbf{J}) \end{bmatrix}, \qquad \mathbf{W}_2 = \begin{bmatrix} k_a\bar{b}_a & \frac{1}{2}\gamma \\ \frac{1}{2}\gamma & \frac{1}{2}\lambda_M(\mathbf{J}) \end{bmatrix}. \tag{A.4}$$

For both $\mathbf{W}_1$ and $\mathbf{W}_2$ to be positive definite, then

$$\begin{aligned}
2k_a\underline{b}_a\lambda_m(\mathbf{J}) - \frac{1}{2}\gamma^2 > 0 \Rightarrow \gamma^2 < 4k_a\underline{b}_a\lambda_m(\mathbf{J}), \\
2k_a\bar{b}_a\lambda_M(\mathbf{J}) - \frac{1}{2}\gamma^2 > 0 \Rightarrow \gamma^2 < 4k_a\bar{b}_a\lambda_M(\mathbf{J}),
\end{aligned}$$

both conditions are satisfied for $\gamma^2 < 4k_a\underline{b}_a\lambda_m(\mathbf{J})$, or $\gamma < 2\sqrt{k_a\underline{b}_a\lambda_m(\mathbf{J})}$.

The time derivative of V, along the system trajectories, is given by

$$\begin{aligned}\dot{V} &= \dot{\bar{V}} + \gamma\,\dot{\mathbf{e}}_\omega^T\mathbf{e}_a + \gamma\,\mathbf{e}_\omega^T\dot{\mathbf{e}}_a\\ &= -k_\omega\mathbf{e}_\omega^T\mathbf{e}_\omega + \gamma\left[\mathbf{J}^{-1}\left((\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1})^\times\mathbf{e}_\omega - k_\omega\mathbf{e}_\omega - k_a\mathbf{e}_a\right)\right]^T\mathbf{e}_a + \gamma\,\mathbf{e}_\omega^T\mathbf{E}_a\mathbf{e}_\omega,\\ &= -k_\omega\mathbf{e}_\omega^T\mathbf{e}_\omega + \gamma\left(\mathbf{J}^{-1}\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1}\right)^\times\mathbf{e}_\omega\right)^T\mathbf{e}_a - \gamma k_\omega\left(\mathbf{J}^{-1}\mathbf{e}_\omega\right)^T\mathbf{e}_a\\ &- \gamma k_a\left(\mathbf{J}^{-1}\mathbf{e}_a\right)^T\mathbf{e}_a + \gamma\,\mathbf{e}_\omega^T\mathbf{E}_a\mathbf{e}_\omega,\end{aligned}$$

where property vi) of Lemma 3.2 was used.

The following equations and inequalites hold:

$$\begin{aligned}-k_\omega\mathbf{e}_\omega^T\mathbf{e}_\omega &= -k_\omega\|\mathbf{e}_\omega\|^2,\\ \mathbf{e}_\omega^T\mathbf{E}_a\mathbf{e}_\omega &\le \|\mathbf{e}_\omega\|\|\mathbf{E}_a\|\|\mathbf{e}_\omega\| \le p_a\|\mathbf{e}_\omega\|^2,\\ -\left(\mathbf{J}^{-1}\mathbf{e}_\omega\right)^T\mathbf{e}_a &\le \|\mathbf{J}^{-1}\mathbf{e}_\omega\|\|\mathbf{e}_a\| \le \|\mathbf{J}^{-1}\|\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| = \frac{1}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|,\\ \left(\mathbf{J}^{-1}\mathbf{e}_a\right)^T\mathbf{e}_a &= \mathbf{e}_a^T\mathbf{J}^{-1}\mathbf{e}_a \ge \lambda_m(\mathbf{J}^{-1})\|\mathbf{e}_a\|^2 = \frac{1}{\lambda_M(\mathbf{J})}\|\mathbf{e}_a\|^2,\end{aligned}$$

where property vi) of Lemma 3.2 was used. Note that the last inequality will flip sign when multiplied by -1, as needed for substitution.

For the last element, using property v) of Lemma 3.2 together with the bound (3.22),

$$\begin{aligned}\left(\mathbf{J}^{-1}\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1}\right)^\times\mathbf{e}_\omega\right)^T\mathbf{e}_a &\le \|\mathbf{J}^{-1}\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_\mathbf{1}\right)^\times\mathbf{e}_\omega\|\|\mathbf{e}_a\|,\\ &\le \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|^2\|\mathbf{e}_a\| + \frac{B_1}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|,\\ &\le \bar{e}_a\frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|^2 + \frac{B_1}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|.\end{aligned}$$

Combining these results,

$$\begin{aligned}\dot{V} &\le -k_\omega\|\mathbf{e}_\omega\|^2 + \gamma\,p_a\|\mathbf{e}_\omega\|^2 + \gamma\,\bar{e}_a\frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|^2\\ &+ \frac{\gamma\,B_1}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| + \frac{\gamma\,k_\omega}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| - \frac{\gamma\,k_a}{\lambda_M(\mathbf{J})}\|\mathbf{e}_a\|^2\\ &= -\left(k_\omega - \gamma\,p_a - \gamma\,\bar{e}_a\frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\right)\|\mathbf{e}_\omega\|^2 - \frac{\gamma\,k_a}{\lambda_M(\mathbf{J})}\|\mathbf{e}_a\|^2 + \frac{\gamma(k_\omega + B_1)}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|,\end{aligned}$$

or, in matrix form,

$$\dot{V} \le -\mathbf{x}^T\mathbf{W}_3\mathbf{x} < 0$$

with $\mathbf{x}$ as defined previously, and

$$\mathbf{W}_3 = \begin{bmatrix} \frac{\gamma\, k_a}{\lambda_M(\mathbf{J})} & -\frac{\gamma(k_\omega + B_1)}{2\lambda_m(\mathbf{J})} \\ -\frac{\gamma(k_\omega + B_1)}{2\lambda_m(\mathbf{J})} & k_\omega - \gamma\, p_a - \gamma\, \bar{e}_a \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})} \end{bmatrix}. \tag{A.5}$$

For $\mathbf{W}_3$ to be positive definite, two conditions need to be met. First

$$\begin{aligned} k_\omega - \gamma\, p_a - \gamma\, \bar{e}_a \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})} &> 0, \\ \gamma\, p_a + \gamma\, \bar{e}_a \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})} &< k_\omega, \\ \gamma &< \frac{k_\omega \lambda_m(\mathbf{J})}{p_a \lambda_m(\mathbf{J}) + \bar{e}_a \lambda_M(\mathbf{J})}, \end{aligned}$$

and

$$\begin{aligned} \left(\frac{\gamma\, k_a}{\lambda_M(\mathbf{J})}\right)\left(\frac{\lambda_m(\mathbf{J})k_\omega - \gamma p_a \lambda_m(\mathbf{J}) - \gamma \bar{e}_a \lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\right) - \left(\frac{\gamma(k_\omega + B_1)}{2\lambda_m(\mathbf{J})}\right)^2 &> 0, \\ \gamma\left(\frac{k_a\, p_a \lambda_m(\mathbf{J}) + \bar{e}_a\, k_a\, \lambda_M(\mathbf{J})}{\lambda_M(\mathbf{J})\lambda_m(\mathbf{J})} + \frac{(k_\omega + B_1)^2}{4\lambda_m^2(\mathbf{J})}\right) &< \frac{k_a\, k_\omega \lambda_m(\mathbf{J})}{\lambda_M(\mathbf{J})\lambda_m(\mathbf{J})}, \\ \gamma\left(\frac{4\lambda_m(\mathbf{J})k_a\, p_a \lambda_m(\mathbf{J}) + 4\lambda_m(\mathbf{J})\bar{e}_a\, k_a\, \lambda_M(\mathbf{J}) + \lambda_M(\mathbf{J})(k_\omega + B_1)^2}{4\lambda_M(\mathbf{J})\lambda_m^2(\mathbf{J})}\right) &< \frac{k_a\, k_\omega}{\lambda_M(\mathbf{J})} \end{aligned}$$

which gives

$$\gamma < \frac{4k_a\, k_\omega \lambda_m^2(\mathbf{J})}{4\lambda_m(\mathbf{J})\left(k_a\, p_a + \bar{e}_a\, k_a\, \lambda_M(\mathbf{J})\right) + \lambda_M(\mathbf{J})(k_\omega + B_1)^2}$$

setting

$$\begin{aligned} \gamma < \min\Bigg\{ & 2\sqrt{k_a \underline{\mathrm{b}}_a \lambda_m(\mathbf{J})}, \frac{k_\omega \lambda_m(\mathbf{J})}{p_a \lambda_m(\mathbf{J}) + \bar{e}_a \lambda_M(\mathbf{J})}, \\ & , \frac{4k_a\, k_\omega \lambda_m^2(\mathbf{J})}{4\lambda_m(\mathbf{J})\left(k_a\, p_a + \bar{e}_a\, k_a\, \lambda_M(\mathbf{J})\right) + \lambda_M(\mathbf{J})(k_\omega + B_1)^2} \Bigg\}. \end{aligned} \tag{A.6}$$

The variable γ is an auxiliary construct to facilitate the proof, it can be chose arbitrarily without impacting the controller. As such, the above inequality can always be satisfied.

The quadratic bounds on V and the final result for $\dot{V}$ concludes, through Theorem 4.10 of [99], that

$$\|\mathbf{x}(t)\|^2 \leq \frac{\lambda_m(\mathbf{W}_1)}{\lambda_m(\mathbf{W}_1)} \exp\left(-\frac{\lambda_m(\mathbf{W}_3)}{\lambda_m(\mathbf{W}_2)} t\right) \|\mathbf{x}(0)\|^2$$

therefore, $\|\mathbf{e}_\omega\|$ and $\|\mathbf{e}_a\|$ converge exponentially to zero. This immediately results in $\boldsymbol{\omega}_b$ converging to $\bar{\boldsymbol{\omega}}_r$, and, through properties (ii) and (iii) of Lemma 3.2, this leads to $\mathbf{C}_{br}$ converging to $\mathbf{I}_3$, thus satisfying the tracking objective, and verifying the Theorem.

A.5 Proof of Theorem 3.6

Consider the same Lyapunov candidate function as in the previous theorem,

$$V = \bar{V} + \gamma\, \mathbf{e}_\omega^T \mathbf{e}_a = \frac{1}{2}\mathbf{e}_\omega^T \mathbf{J} \mathbf{e}_\omega + k_a \Psi_a + \gamma\, \mathbf{e}_\omega^T \mathbf{e}_a,$$

if γ satisfies (A.6), then V is positive definite and lower bounded through (A.3), (A.4).

The time derivative of V over the system trajectories, when substituting the control law 3.25, can be shown to be (same derivation as the previous theorem, with two additional terms)

$$\begin{aligned}\dot{V} \leq &-\left(k_\omega - \gamma\, p_a - \gamma\, \bar{e}_a \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\right)\|\mathbf{e}_\omega\|^2 - \frac{\gamma\, k_a}{\lambda_M(\mathbf{J})}\|\mathbf{e}_a\|^2 + \frac{\gamma(k_\omega + B_1)}{\lambda_m(\mathbf{J})}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\| \\ &+ \gamma \frac{\lambda_M(\mathbf{J})}{\lambda_m(\mathbf{J})}\|\mathbf{e}_a\|\rho(t) + \lambda_M(\mathbf{J})\|\mathbf{e}_\omega\|\rho(t),\end{aligned}$$

or, in matrix form,

$$\dot{V} \leq -\mathbf{x}^T \mathbf{W}_3 \mathbf{x} + \mathbf{z}^T \mathbf{x},$$

where $\mathbf{x} = [\|\mathbf{e}_\omega\|,\ \|\mathbf{e}_\omega\|]^T$, $\mathbf{W}_3$ as in (A.5), and $\mathbf{z} = \lambda_M(\mathbf{J})\rho(t)\left[\gamma/\lambda_m(\mathbf{J}),\quad 1\right]^T$. As in the previous proof, $\mathbf{W}_3$ will be positive definite if γ satisfies (A.6). Then

$$\dot{V} \leq \lambda_m(\mathbf{W}_3)\|\mathbf{x}\|^2 + k_\rho \rho(t)\|\mathbf{x}\|, \tag{A.7}$$

where

$$k_\rho = \lambda_M(\mathbf{J})\sqrt{1 + \frac{\gamma^2}{\lambda_m^2(\mathbf{J})}}.$$

Integrating both sides of (A.7),

$$V(t) - V(0) \leq -\lambda_m(\mathbf{W}_3)\int_0^t \|\mathbf{x}(\tau)\|^2 d\tau + k_\rho \int_0^t \|\mathbf{x}(\tau)\| d\tau \tag{A.8}$$

then, since $V(t)$ is positive definite,

$$V(0) \geq \lambda_m(\mathbf{W}_3)\int_0^t \|\mathbf{x}(\tau)\|^2 d\tau - k_\rho \int_0^t \|\mathbf{x}(\tau)\| d\tau \tag{A.9}$$

rearranging,

$$V(0) + k_\rho \int_0^t \|\mathbf{x}(\tau)\| d\tau \geq \lambda_m(\mathbf{W}_3) \int_0^t \|\mathbf{x}(\tau)\|^2 d\tau,$$

where the right-hand side term is equal to $\|\mathbf{x}\|_{L_2}^2$ and the second left-hand side term is, through the Schwartz inequality,

$$k_\rho \int_0^t \|\mathbf{x}(\tau)\| d\tau \leq \sqrt{\int_0^t \rho^2(\tau) d\tau \int_0^t \|\mathbf{x}(\tau)\|^2 d\tau} = \|\rho(t)\|_{L_2} \|\mathbf{x}\|_{L_2}$$

where the condition 3.26 was used. Then

$$V_0 \geq \lambda_m(\mathbf{W}_3)\|\mathbf{x}\|_{L_2}^2 - k_\rho \|\rho(t)\|_{L_2} \|\mathbf{x}\|_{L_2}.$$

Completing the square on the right,

$$\begin{aligned} \frac{V_0}{\lambda_m(\mathbf{W}_3)} + \frac{k_\rho^2 \|\rho(t)\|_{L_2}^2}{4\lambda_m^2(\mathbf{W}_3)} &\geq \|\mathbf{x}\|_{L_2}^2 - \frac{k_\rho \|\rho(t)\|_{L_2}}{\lambda_m^2(\mathbf{W}_3)} \|\mathbf{x}\|_{L_2} + \frac{k_\rho^2 \|\rho(t)\|_{L_2}^2}{4\lambda_m^2(\mathbf{W}_3)} \\ &= \left(\|\mathbf{x}\|_{L_2} - \frac{k_\rho \|\rho(t)\|_{L_2}}{2\lambda_m(\mathbf{W}_3)} \right)^2 \end{aligned}$$

then

$$\|\mathbf{x}\|_{L_2} \leq \sqrt{\frac{V_0}{\lambda_m(\mathbf{W}_3)} + \frac{k_\rho^2 \|\rho(t)\|_{L_2}^2}{4\lambda_m^2(\mathbf{W}_3)}} + \frac{k_\rho \|\rho(t)\|_{L_2}}{2\lambda_m(\mathbf{W}_3)} \tag{A.10}$$

which shows that $\mathbf{x} \in L_2$. The rest of the proof follows from [31]. Substituting (A.10) in (A.7) shows that V is uniformly bounded along the system trajectories. Then from (3.7) and (3.19), $\dot{\mathbf{x}}$ is uniformly bounded as well, and hence $\mathbf{x}$ is uniformly continuous. Invoking Barbalat's Lemma, it can be concluded that $\mathbf{x} \to \mathbf{0}$ as $t \to \infty$, implying that $\mathbf{e}_a$ and $\mathbf{e}_\omega \to \mathbf{0}$, verifying the theorem.

B Position control proofs

B.1 Proof of Theorem 4.3

Substituting the value of the intermediate command (4.13) in the error dynamics (4.7) results in

$$\dot{\mathbf{e}}_v = -k_v \mathbf{e}_v - k_p \mathbf{e}_p - k_i \operatorname{sat}_k(\mathbf{e}_i) + \boldsymbol{\Delta}_a + \boldsymbol{\Delta}_c$$

where

$$\boldsymbol{\Delta}_a = \frac{1}{m}\left((\mathbf{F}_a)_i - (\hat{\mathbf{F}}_a)_i\right), \tag{B.11}$$

$$\boldsymbol{\Delta}_c = \frac{\mathrm{T}}{m}\left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T \hat{\mathbf{k}}_1. \tag{B.12}$$

The first element appears due to the mismatched estimate of the aerodynamic forces, and the second is a perturbation due to the UAV attitude not converging to the reference attitude instantly.

Investigating the bound on (B.12),

$$\|\boldsymbol{\Delta}_c\| \leq \|\mathbf{F}_c\| \| \left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T \hat{\mathbf{k}}_1\|,$$

where the second right-hand term is the magnitude of the difference between $\hat{\mathbf{b}}_1$ and $\hat{\mathbf{r}}_1$. Since both basis vectors are unit length, it follows that $\| \left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T \hat{\mathbf{k}}_1\| \leq 2$. However, a more useful bound can be found in terms of the innovation term (3.11). Using simple trigonometry,

$$\begin{aligned}\| \left(\mathbf{C}_{bi} - \mathbf{C}_{ri}\right)^T \hat{\mathbf{k}}_1\| &= \sqrt{\|\hat{\mathbf{b}}_1\|^2 + \|\hat{\mathbf{r}}_1\|^2 - 2\|\hat{\mathbf{b}}_1\|\|\hat{\mathbf{b}}_1\|\hat{\mathbf{r}}_1\| \cos(\mu)},\\ &= \sqrt{2(1-\cos(\mu))} = 2|\sin\left(\mu/2\right)|,\end{aligned}$$

where μ is the angle between $\hat{\mathbf{b}}_1$ and $\hat{\mathbf{r}}_1$. Defining η as the principal angle of the angle-axis parametrization of $\mathbf{C}_{br}$, it clear that $|\mu| \leq |\eta|$, since η encodes the complete three dimensional attitude deviation, whereas μ is blind to any rotation around the first axis. Within the set L_a, $|\mu| \leq |\eta|$ implies $|\sin(\mu/2)| \leq |\sin(\eta/2)|$. In Lemma 3.2 it is verified that

$$\|\mathbf{e}_a\| = \left\|\frac{\sqrt{2}\sin(\eta)}{2\sqrt{1+\cos(\eta)}}\hat{\boldsymbol{\epsilon}}\right\| = \frac{1}{2}\left|\frac{\sin(\eta)}{\cos(\eta/2)}\right| = |\sin(\eta/2)|.$$

In addition, from Lemma 3.4, and property iv) of Lemma 3.2, it can be shown that

$$\|\mathbf{e}_a\| < 2\sqrt{\psi}.$$

Combining these results, together with the initial conditions of the theorem leads to

$$\|\boldsymbol{\Delta}_c\| \leq 2\|\mathbf{F}_c\|\|\mathbf{e}_a\| < 2\sqrt{\psi}\|\mathbf{F}_c\|. \tag{B.13}$$

Define the Lyapunov candidate function for the coupled position and attitude system

$$V = V_c + V_p + V_i,$$

where

$$\begin{aligned}
V_c &= \frac{1}{2}\mathbf{e}_\omega^T \mathbf{J}\mathbf{e}_\omega + k_a \Psi_a + \gamma\, \mathbf{e}_\omega^T \mathbf{e}_a, \\
V_p &= \frac{1}{2}\mathbf{e}_v^T \mathbf{e}_v + \frac{1}{2} k_p \mathbf{e}_p^T \mathbf{e}_p + c_p \mathbf{e}_p^T \mathbf{e}_v, \\
V_i &= \int_{\frac{\boldsymbol{\Delta}_a}{k_i}}^{\mathbf{e}_i} \left(k_i \operatorname{sat}_k(\tau) - \boldsymbol{\Delta}_a\right) d\tau,
\end{aligned}$$

The rest of the proof is analogous to the analysis presented in [97], with the necessary modifications to account for the different attitude innovation term and the different intermediate command.

Using the results of Theorem 3.5, the Lyapunov candidate function can be shown to be positive definite and upper and lower bounded, since the integral term is positive definite about $\mathbf{e}_i = \frac{B_e}{k_i}$ thanks to condition (4.20). Define $\mathbf{x}_p = [\|\mathbf{e}_p\|, \|\mathbf{e}_v\|]^T$ and $\mathbf{x}_a = \|[\mathbf{e}_a\|, \|\mathbf{e}_\omega\|]^T$, then due to property iv) of Lemma 3.2,

$$\frac{1}{2}\mathbf{x}_p^T \mathbf{M}_{p1} \mathbf{x}_p + \frac{1}{2}\mathbf{x}_a^T \mathbf{M}_{a1} \mathbf{x}_a + V_i \leq V \leq \frac{1}{2}\mathbf{x}_p^T \mathbf{M}_{p2} \mathbf{x}_p + \frac{1}{2}\mathbf{x}_a^T \mathbf{M}_{a2} \mathbf{x}_a + V_i$$

where

$$\begin{aligned}
\mathbf{M}_{p1} &= \begin{bmatrix} k_p & -c_p \\ -c_p & 1 \end{bmatrix}, \quad & \mathbf{M}_{p2} &= \begin{bmatrix} k_p & c_p \\ c_p & 1 \end{bmatrix}, \\
\mathbf{M}_{a1} &= \begin{bmatrix} 2k_a & -\gamma \\ -\gamma & \lambda_m \end{bmatrix}, \quad & \mathbf{M}_{a2} &= \begin{bmatrix} 4k_a & \gamma \\ \gamma & \lambda_M \end{bmatrix},
\end{aligned}$$

where $\lambda_m = \lambda_m(\mathbf{J})$ and $\lambda_M = \lambda_M(\mathbf{J})$ are the smallest and largest eigenvalues of $\mathbf{J}$, respectively. These four matrices are positive definite due to (4.21) and (4.22).

The time derivative of V along the system trajectories is

$$\begin{aligned}\dot{V} &= k_p\mathbf{e}_p^T\mathbf{e}_v + \mathbf{e}_v^T\left(-k_v\mathbf{e}_v - k_p\mathbf{e}_p - k_i\,\mathrm{sat}_k(\mathbf{e}_i) + \boldsymbol{\Delta}_a + \boldsymbol{\Delta}_c\right)\\ &\quad + c_p\mathbf{e}_v^T\mathbf{e}_v + c_p\mathbf{e}_p^T\left(-k_v\mathbf{e}_v - k_p\mathbf{e}_p - k_i\,\mathrm{sat}_k(\mathbf{e}_i) + \boldsymbol{\Delta}_a + \boldsymbol{\Delta}_c\right)\\ &\quad + \left(k_i\,\mathrm{sat}_k(\mathbf{e}_i) - \boldsymbol{\Delta}_a\right)^T\left(\mathbf{e}_v + c_p\mathbf{e}_p\right) - k_\omega\mathbf{e}_\omega^T\mathbf{e}_\omega + \gamma\dot{\mathbf{e}}_a^T\mathbf{e}_\omega\\ &\quad + \gamma\mathbf{e}_a^T\mathbf{J}^{-1}\left(\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_{r1}\right)^\times\mathbf{e}_\omega - k_p\mathbf{e}_a - k_\omega\mathbf{e}_\omega\right),\\ &= -(k_v - c_p)\|\mathbf{e}_v\|^2 - c_pk_p\|\mathbf{e}_p\|^2 + \boldsymbol{\Delta}_c^T\left(\mathbf{e}_v + c_p\mathbf{e}_p\right) - c_pk_v\mathbf{e}_p^T\mathbf{e}_v\\ &\quad - k_\omega\|\mathbf{e}_\omega\|^2 + \gamma\dot{\mathbf{e}}_a^T\mathbf{e}_\omega + \gamma\mathbf{e}_a^T\mathbf{J}^{-1}\left(\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_{r1}\right)^\times\mathbf{e}_\omega - k_p\mathbf{e}_a - k_\omega\mathbf{e}_\omega\right).\end{aligned}$$

Using the bound (3.22) and property v) of Lemma 3.2, it can be shown that

$$\left(\mathbf{J}^{-1}\left(\mathbf{J}\mathbf{e}_\omega + \boldsymbol{\Delta}_{r1}\right)^\times\mathbf{e}_\omega\right)^T\mathbf{e}_a \le \frac{\lambda_M}{\lambda_m}\|\mathbf{e}_\omega\|^2 + \frac{B_1}{\lambda_m}\|\mathbf{e}_\omega\|\|\mathbf{e}_a\|,$$

which together with property vi) of Lemma 3.2 results in

$$\begin{aligned}\dot{V} &\le -(k_v - c_p)\|\mathbf{e}_v\|^2 - c_pk_p\|\mathbf{e}_p\|^2 + \left(\|\mathbf{e}_v\| + c_p\|\mathbf{e}_p\|\right)\|\boldsymbol{\Delta}_c\| + c_p\,k_v\|\mathbf{e}_p\|\|\mathbf{e}_v\|\\ &\quad - \mathbf{x}_a^T\mathbf{W}_a\mathbf{x}_a.\end{aligned}$$

Using (B.13), together with initial conditions (4.18) and (4.19), and the assumption on the aerodynamic estimate (4.17), the bound on the cross-term of the position and attitude errors can be written as

$$\begin{aligned}\left(\|\mathbf{e}_v\| + c_p\|\mathbf{e}_p\|\right)\|\boldsymbol{\Delta}_c\| &< 2\sqrt{\psi}k_v\|\mathbf{e}_v\|^2 + 2\sqrt{\psi}c_pk_p\|\mathbf{e}_p\|^2\\ &\quad + 2(\sqrt{3}k_i\,k + B_p)\left(\|\mathbf{e}_v\| + c_p\|\mathbf{e}_p\|\right)\|\mathbf{e}_a\|\\ &\quad + 2\sqrt{\psi}c_pk_v\|\mathbf{e}_p\|\|\mathbf{e}_v\| + 2\,k_p\,e_M\|\mathbf{e}_a\|\|\mathbf{e}_v\|.\end{aligned}$$

Then,

$$\begin{aligned}\dot{V} &\le -\mathbf{x}_a^T\mathbf{W}_a\mathbf{x}_a - \mathbf{x}_p^T\mathbf{W}_p\mathbf{x}_p + \mathbf{x}_a^T\mathbf{W}_{ap}\mathbf{x}_p\\ &\le -\begin{bmatrix}\|\mathbf{x}_a\|\|\mathbf{x}_p\|\end{bmatrix}\begin{bmatrix}\lambda_{min}(\mathbf{W}_a) & -\frac{1}{2}\|\mathbf{W}_{ap}\|_2\\ -\frac{1}{2}\|\mathbf{W}_{ap}\|_2 & \lambda_{min}(\mathbf{W}_p)\end{bmatrix}\begin{bmatrix}\|\mathbf{x}_a\|\\ \|\mathbf{x}_p\|\end{bmatrix} \le 0\end{aligned}$$

where (4.21) and (4.22), respectively, ensure $\mathbf{W}_a$ and $\mathbf{W}_p$ are positive definite, and (4.23) results in the final matrix being positive definite. These results imply $\mathbf{e}_a$, $\mathbf{e}_\omega$ $\mathbf{e}_p$, and $\mathbf{e}_v$ converge exponentially to zero, and the integral term is uniformly bounded.

B.2 Proof of Lemma 4.4

The proof follows closely from [102]. Consider the positive definite function

$$V = \frac{1}{2}\mathbf{e}_v^T\mathbf{e}_v + \frac{1}{2}\mathbf{e}_p^T\mathbf{e}_p = \frac{1}{2}\|\mathbf{e}_p\|^2 + \frac{1}{2}\|\mathbf{e}_v\|^2.$$

It then holds that $\|\mathbf{e}_p\| \leq \sqrt{2V}$ and $\|\mathbf{e}_v\| \leq \sqrt{2V}$. The time derivative V along the system trajectories is given by

$$\dot{V} = \mathbf{e}_p^T\mathbf{e}_v + \mathbf{e}_v^T\left(g\,\hat{\mathbf{k}}_3 + \frac{1}{m}\mathbf{f}_a^i - \dot{\mathbf{v}}_r + \frac{\mathrm{T}}{m}\mathbf{C}_{bi}^T\hat{\mathbf{k}}_1\right).$$

Substituting the control law (4.13), it can be verified that

$$\dot{V} \leq \|\mathbf{e}_p\|\|\mathbf{e}_v\| + \|\mathbf{e}_v\|\left(k_p\|\mathbf{e}_p + k_v\|\mathbf{e}_v\| + (2\,B_g + B_e + k\,k_i)\right) \tag{B.14}$$

where $B_g \leq B_p$ is the norm of the feed-forward term without the aerodynamic force estimate:

$$\| - g\,\hat{\mathbf{k}}_3 + \dot{\mathbf{v}}_r\| \leq B_g,$$

this bound is guaranteed to exist due to assumption 4.16. Then,

$$\begin{aligned}\dot{V} &\leq (1 + k_p)\|\mathbf{e}_p\|\|\mathbf{e}_v\| + (2\,B_g + B_e + k\,k_i)\|\mathbf{e}_v\| + k_v\|\mathbf{e}_v\|^2 \\ &\leq 2(1 + k_p)V + 2\,k_vV + \sqrt{2}(2\,B_g + B_e + k\,k_i)\sqrt{V} = \kappa_1V + \kappa_2\sqrt{V},\end{aligned}$$

where $\kappa_1 = 2(1 + k_p + k_v)$ and $\kappa_2 = \sqrt{2}(2\,B_g + B_e + k\,k_i)$. If, within some the interval $t \in [t_a,\,t_b] \in [0,\,t_c]$, $V < 1$, then V might grow depending on the values of κ_1 and κ_2. However, if at some time $t \in [t_a,\,t_b]$, V grows such that $V > 1$, then $V > \sqrt{V}$ and the above inequality becomes

$$\dot{V} \leq (\kappa_1 + \kappa_2)V.$$

Using the Comparison Lemma [99], then

$$V(t) \leq V(t_a)e^{(\kappa_1+\kappa_2)(t-t_a)},$$

implying V is bounded for $t \in [t_a,\,t_b]$. Since an initial $V < 1$ will, at most, result in $V > 1$, then V is bounded for $t \in [0,\,t_c]$ and so are the error states $\|\mathbf{e}_p\|$ and $\|\mathbf{e}_v\|$, completing the proof.

B.3 Proof of Theorem 4.5

The proof is modified from that of Theorem 4.3.

Consider the Lyapunov candidate function

$$V_\nu = V_{att} + \frac{1}{2}\mathbf{e}_v^T\mathbf{e}_v + \int_{\frac{\boldsymbol{\Delta}_a}{k_p}}^{\mathbf{e}_{vi}} (k_p \mathrm{sat}_\nu(\tau) + \boldsymbol{\Delta}_a) d\tau, \tag{B.15}$$

where

$$V_{att} = \frac{1}{2}\mathbf{e}_\omega^T\mathbf{J}\mathbf{e}_\omega + k_a\Psi_a + \gamma\,\mathbf{e}_\omega^T\mathbf{e}_a.$$

Using the same procedure as Theorem 4.3, V is both upper and lower bounded, and its time derivative along the system trajectories can be shown to satisfy

$$\dot{V}_v \leq -k_v(1 - 2\sqrt{\psi})\|\mathbf{e}_v\|^2 + 2(k_p\sqrt{3}\nu + B_p)\|\mathbf{e}_v\|\|\mathbf{e}_a\| + \mathbf{x}_a^T\mathbf{W}_a\mathbf{x}_a,$$

with $\mathbf{x}_a$ as defined in Theorem 4.3. Through the conditions of the Theorem, $\dot{V} \leq 0$ which implies the $\|\mathbf{e}_v\|$, $\|\mathbf{e}_a\|$, $\|\mathbf{e}_\omega\|$ decrease exponentially to zero, while the integral term remains uniformly bounded.

C Path following proofs

C.1 Proof of Proposition 5.1

The rate of change with respect to time of $\mathbf{C}_{gi}$ (5.8) is given by

$$\dot{\mathbf{C}}_{gi} = -\boldsymbol{\omega}_g^\times\mathbf{C}_{gi},$$

then

$$\boldsymbol{\omega}_g^\times = \mathbf{C}_{gi}\dot{\mathbf{C}}_{gi}^T = \dot{\sigma}\begin{bmatrix}\mathbf{T}^T & \mathbf{H}^T & \mathbf{P}^T\end{bmatrix}^T\begin{bmatrix}\mathbf{T}' & \mathbf{H}' & \mathbf{P}'\end{bmatrix},$$

where

$$
\begin{aligned}
\mathbf{T}' &= \frac{\boldsymbol{\mu}''}{\|\boldsymbol{\mu}'\|} - \frac{\boldsymbol{\mu}'^T\boldsymbol{\mu}''}{\|\boldsymbol{\mu}'\|^3}\boldsymbol{\mu}' \\
\mathbf{H}' &= \frac{\mathbf{k}_3^\times\boldsymbol{\mu}''}{\|\mathbf{k}_3^\times\boldsymbol{\mu}'\|} - \frac{(\mathbf{k}_3^\times\boldsymbol{\mu}')^T(\mathbf{k}_3^\times\boldsymbol{\mu}'')}{\|\mathbf{k}_3^\times\boldsymbol{\mu}'\|^3}(\mathbf{k}_3^\times\boldsymbol{\mu}') \\
\mathbf{P}' &= \mathbf{T}^\times\mathbf{H}' + \mathbf{T}'^\times\mathbf{H}.
\end{aligned}
$$

Simple substitution of these values verifies $\boldsymbol{\omega}_g^\times$ is indeed skew symmetric, and that (5.9) holds.

C.2 Proof of Theorem 5.2

Consider the Lyapunov candidate function

$$
V_p = \frac{1}{2}(\mathbf{p}_i - \boldsymbol{\mu}(\sigma))^T(\mathbf{p}_i - \boldsymbol{\mu}(\sigma)) = \frac{1}{2}\|\mathbf{e}_p(\sigma)\|^2
$$

that is positive definite about $\mathbf{p}_i = \boldsymbol{\mu}(\sigma)$. Its time derivative, along the system trajectories, substituting the reference velocity (5.11, 5.12), and path evolution law (5.13)

$$
\begin{aligned}
\dot{V}_p &= (\mathbf{e}_p(\sigma))^T\dot{\mathbf{e}}_p(\sigma) = \mathbf{e}_p^T(\mathbf{v}_i - \dot{\sigma}\boldsymbol{\mu}'(\sigma)) \\
&= \mathbf{e}_p^T\left(V_c\mathbf{T} - \frac{V_c k_c}{d_l}e_c\mathbf{H} - \frac{V_c k_c}{d_l}\mathbf{P}\right) - \left((V_c\mathbf{T} + k_s\mathbf{e}_{pr})^T\mathbf{T}\right)\mathbf{T} + \mathbf{e}_p^T\mathbf{e}_v \\
&= -k_s e_s^2 - \frac{V_c k_c}{d_l}e_c^2 - \frac{V_c k_h}{d_l}e_h^2 + \mathbf{e}_p^T\mathbf{e}_v \\
&\leq -\lambda_{min}(\mathbf{K}_p)\|\mathbf{e}_p\| + \|\mathbf{e}_p\|\|\mathbf{e}_v\|
\end{aligned}
$$

The Lyapunov function for the system can then be taken as $V_G = V_p + V_\nu$, where V_ν is defined in (B.15) and its time derivative can be shown to satisfy, for $\mathbf{x}_G = [\|\mathbf{e}_p\|, \|\mathbf{e}_v\|]^T$,

$$
\dot{V}_G = -\mathbf{e}_G^T\mathbf{W}_G\mathbf{e}_G + \mathbf{e}_G^T\mathbf{W}_{Ga}\mathbf{e}_v - \mathbf{x}_a^T\mathbf{W}_a\mathbf{x}_a
$$

where $\mathbf{W}_G$ will be positive definite through (5.14) and $\dot{V}_G \leq 0$ through (5.16), and the errors will converge exponentially to zero, while the integral error will remain uniformly bounded. Since $\mathbf{C}_{gi}$ is a DCM, then $\|\mathbf{e}_p\| \to 0$ implies $\|\mathbf{e}_{pr}\| \to 0$, verifying the path following objective.