Advanced Optimal Control of Fixed-Wing UAV for Ground Target Tracking: Analysis and Application

by **Ignacio Javier Torres Herrera**

Thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy
under the supervision of
A/Prof. Ricardo Aguilera and A/Prof. Quang Ha

School of Electrical and Data Engineering
Faculty of Engineering and IT
University of Technology Sydney
June 4, 2025

Certificate of Original Authorship

I, Ignacio Javier Torres Herrera, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Electrical and Data Engineering at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: Production Note:
Signature removed prior to publication.

Date: June 4, 2025

Abstract

This thesis presents a comprehensive control strategy for fixed-wing Unmanned Aerial Vehicles (UAVs) engaged in ground target tracking, addressing the challenges of aerodynamic constraints, unpredictable target movements, and environmental disturbances. It starts by developing a Nonlinear Model Predictive Controller (NMPC) formulated as a non-convex optimization problem, based on a nonlinear three-dimensional target tracking system model. Stability conditions for the nonlinear closed-loop system are established by analyzing a linear controller within a specified terminal region, enabling the use of convex Model Predictive Control techniques within the NMPC framework. This ensures that the UAV's controlled trajectory reaches the terminal region within a fixed prediction horizon, allowing effective tracking of the ground target.

To enhance robustness against disturbances and unknown target trajectories, the control strategy is extended by modeling the target's movement as a first-order dynamic system and deriving a comprehensive three-dimensional Dubins model that accounts for both target movement and exogenous disturbances. A bilinear time-varying disturbance model is introduced, and a Kalman filter-based estimation strategy is employed to estimate both disturbances and the target's movement. This information is integrated into the NMPC to achieve highly accurate tracking.

Further improving performance without relying on explicit disturbance models, with the integration of a phase-based Iterative Learning Control (ILC) with the NMPC. The ILC leverages data collected during each iteration of the tracking task, adjusting control actions based on the phase angle of the UAV's orbit around the target. This hybrid control scheme enhances disturbance rejection and tracking accuracy by iteratively refining control actions, without the need for precise disturbance modeling or observer-based estimation.

Extensive simulations and experimental results validate the effectiveness of the proposed control strategy. The results demonstrate that the UAV successfully tracks moving ground targets with unknown trajectories, maintaining high accuracy even in the presence of significant uncertainties and disturbances. This work contributes a robust and adaptable control framework for fixed-wing UAV ground target tracking, suitable for dynamic envi-

ronments where traditional methods may be insufficient.

Dedication

To Margaret.

Acknowledgements

First, I would like to thank my supervisors, A/Prof Ricardo Aguilera and A/Prof Quang Ha. Their continuous support, guidance, encouragement, and patience were the fuel for this research work.

I also want to thank my university friends, Pablo Poblete, Rodrigo Cuzmar, Lanh Nguyen, Duy Nguyen, and Trung Le, for always being there for me, whether by supporting me with technical problems or simply offering a chat during difficult moments.

I am also grateful to my family. Even though I am on the other side of the world, their unconditional love has been my constant support throughout this journey.

Ignacio Javier Torres Herrera
June 4, 2025
Sydney, Australia

Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Takeoff and Landing
GTT	Ground Target Tracking
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
ILC	Iterative Learning Control
PID	Proportional Integral Derivative
LQR	Linear Quadratic Regulator
CLF	Candidate Lyapunov Function
RMSE	Root Mean Square Error
SMC	Sliding Mode Control
LTV	Linear Time-Variant
LTI	Linear Time-Invariant
KF	Kalman Filter
FPV	First Person View
OSD	On-Screen Display
GPU	Graphics Processing Unit

Chapter 1

Introduction

This introductory chapter gives the central base and scope of this thesis. It starts with a brief hypothesis on employing Nonlinear Model Predictive Control (NMPC) for ground target tracking with fixed-wing Unmanned Aerial Vehicle (UAV) to achieve optimal control performance and stability. The objective section explains the general and specific purposes of the work, including the NMPC, experimental platform development, observer-based disturbances rejection, and the exploration of data-driven approaches for this particular application. Following these objectives, the major contributions of thesis are summarized, which highlights theoretical and practical progress, implementation guidelines and innovative estimates techniques. A list of publications and cooperation reflects the width of the spread and effect of this research, and the observation of the dissertation structure concludes the chapter.

1.1 Research Proposal

Based on the challenges identified in ground target tracking (GTT) with fixed-wing UAVs and the objectives outlined in this research, the hypothesis and objectives are presented.

1.1.1 Hypothesis

- An NMPC strategy that integrates three-dimensional UAV kinematics, observer-based disturbance estimation, and a data-driven compensation mechanism can be implemented in low-cost fixed-wing UAVs to guarantee stability and achieve high tracking accuracy. This approach enables real-time rejection of both modeled and unmodeled disturbances, ultimately driving the tracking error to zero at steady state.

1.1.2 Objectives

The general objective of this thesis is to develop an NMPC strategy for ground target tracking using fixed-wings UAVs with guaranteed stability, capable of reject exogenous disturbances to reach zero tracking error at steady state. For this general objective, the following specific objectives are considered:

1. To design an NMPC approach for ground target tracking, considering 3D kinematics and establish the stability guarantees.
2. To design and develop an experimental UAV platform capable of real-time implementation of non-convex optimization.
3. To develop and implement an observer-based methodology that allows to reject disturbances and incorporate the target's movement into the controller to achieve zero tracking error in steady state.
4. To develop a data-driven strategy with convergence guarantees that is capable of rejecting unmodeled disturbances and incorporate this alongside NMPC for improved target tracking.

1.2 Mayor Contributions of the Thesis

In this section, the mayor contributions of this thesis are outlined.

1. This thesis proposes an NMPC approach for ground target tracking with stability guarantees. Furthermore, the stability domain is analyzed for different parameters and weighting factors on the NMPC.
2. This thesis uses the proposed NMPC and give the guidelines to implement the proposal in a low-cost commercial UAV, with minimum customization of the traditional flight controllers that can be found in the market. Additionally, the parallelization of the non-convex optimization algorithm is covered with a simple yet effective decorator technique using Python and Numba.
3. A bilinear time-variant Kalman filtering technique is implemented to account for exogenous disturbances. A rotating method is implemented on the estimation of the disturbances, seeking to rotate them alongside the position of the UAV. As result, the controller is able to estimate the exogenous disturbances, such as wind, and target movement without needing to measure them.
4. Finally, a data-driven-based compensation technique for unmodeled disturbances is proposed using phase-based Iterative Learning Control (ILC) approach. The proposed strategy relies on input/output data to improve the tracking while the

UAV is loitering around the target. To account for the NMPC loop, a reduced-order dynamic system model analysis is proposed. Results show accurate tracking even when considering a change in the sign of the disturbances values.

1.3 List of Publications

Published Journal Papers

- **I. Torres**, R. Aguilera and Q.P. Ha , "Design and Performance Evaluation of Nonlinear Model Predictive Control for 3D Ground Target Tracking with Fixed-Wing UAVs", IEEE Open Journal of the Industrial Electronics Society.

Submitted Journal Papers

- **I. Torres**, R. Aguilera and Q.P. Ha, "Offset-free Ground Target Tracking with Fixed-Wings UAVs using NMPC", ISA Transactions.
- **I. Torres**, R. Aguilera and Q.P. Ha, "Iterative Learning Control for unmodeled disturbances compensation for Ground Target Tracking using NMPC", Control Engineering Practice.

Published Conference Papers

- **I. Torres**, L. V. Nguyen, H. Trung, R. Aguilera and Q.P. Ha, "UAV Target Tracking using Nonlinear Model Predictive Control", 2022 International Conference on Electrical, Computer and Energy Technologies (ICECET), Prague, Czech Republic, 2022, pp. 1-7, doi: 10.1109/ICECET55527.2022.9873035.
- **I. Torres**, Q. Ha and R. Aguilera, "Implementation Guidelines of a UAV Fixed-Wing for Advanced Real-Time Control Algorithms", 2024 IEEE/SICE International Symposium on System Integration (SII), Ha Long, Vietnam, 2024, pp. 804-809, doi: 10.1109/SII58957.2024.10417433.
- **I. Torres**, R. Aguilera, Q. Ha, "On the Stability of Nonlinear Model Predictive Control for 3D Target Tracking", IFAC-PapersOnLine, Volume 58, Issue 18, 2024, Pages 194-199, ISSN 2405-8963, doi: https://doi.org/10.1016/j.ifacol.2024.09.030.

Collaborations

- L. V. Nguyen, **I. Torres**, H. Trung, M.D. Phung, R. Aguilera and Q.P. Ha, "Stag hunt game-based approach for cooperative UAVs", in 2022 Int. Symp. Automation and Robotics in Construction, 2022, p.p. 367-374.

- Nguyen, L.V., Le, T.,**Torres, I.**, et al. Intelligent path planning for civil infrastructure inspection with multi-rotor aerial vehicles. Constr Robot 8, 19 (2024). https://doi.org/10.1007/s41693-024-00135-9

Awards

- First Place Award - Demonstration Presentation in SEDE Research Showcase Competition. May 2023.

1.4 Thesis Structure

The remainder of the thesis is divided into six chapters, summarized as follows:

- Chapter 2 depicts a comprehensive literature review on the two more popular types of UAVs, namely quadcopters and fixed-wings, considering their applications and the usual applied control techniques for different tasks. Moreover, an in-depth review on MPC and NMPC techniques focused on fixed-wings is presented, alongside the challenges in their applications. Consequently, the attention is focused on the control challenges regarding target tracking using fixed-wings UAVs, the state of the art approaches and the new challenges for this application.

- Chapter 3 presents an NMPC designed for a fixed-wing UAV tasked with tracking a ground target. Initially, a nonlinear three-dimensional model for the target tracking system is developed. The NMPC is then formulated as a non-convex optimization problem. To establish sufficient stability conditions for the nonlinear closed-loop system, a linear controller with bounded constraints within a specified terminal region is analyzed. By ensuring that the controlled trajectory is drawn into this terminal region near the system reference, the use of linear MPC techniques within the NMPC framework is enabled. Consequently, the NMPC closed-loop system is proven to reach the terminal region within a fixed prediction horizon, allowing the UAV to effectively track the ground target. An initialization technique is employed during optimization to mitigate suboptimality and preserve stability. System stability is achieved for three different speed references by adjusting weighting factors. Extensive simulations validate the proposed approach, and experimental results provide insights from field tests, confirming the efficacy of the control strategy. The results demonstrate that the UAV successfully tracks the target reference while remaining within desired operational parameters.

- Chapter 4 introduces a robust NMPC for ground target tracking using fixed-wing UAVs, capable of countering disturbances and tracking a moving ground target with an unknown trajectory while maintaining high accuracy. It starts by presenting the

UAV's kinematics, incorporating exogenous disturbances. The target's movement is modeled as a first-order dynamic system, enabling the derivation of a comprehensive three-dimensional Dubins model for target tracking that accounts for both target movement and disturbances. Using this model, a bilinear time-varying disturbance model is presented. A Kalman filter-based strategy is employed to estimate both disturbances and the target's movement, providing critical information to the NMPC for precise tracking. The effectiveness of the proposed approach is verified through simulations and experimental results.

- Chapter 5 proposes a novel control strategy that combines NMPC with phase-based ILC for ground target tracking using fixed-wing UAVs. Fixed-wing UAVs offer extended flight durations suitable for tasks like target recognition and monitoring but pose significant control challenges due to their reliance on continuous forward motion and inability to hover. Traditional ground target tracking methods often require prior knowledge of the target's trajectory or precise disturbance models, which may not be feasible in dynamic environments with unpredictable targets and complex disturbances. The presented approach leverages phase-based ILC to iteratively refine control actions based on the phase angle of the UAV's orbit around the target, enhancing tracking performance without explicit disturbance modeling or observer-based estimation. The NMPC component optimally handles system dynamics and constraints within a finite prediction horizon, while the ILC component adjusts for disturbances recurring at specific orbital phases. This hybrid control scheme improves disturbance rejection and tracking accuracy, enabling the UAV to effectively track moving ground targets with unknown trajectories. Simulation results demonstrate the effectiveness of the proposed method in scenarios with significant uncertainties and disturbances.
- Chapter 6 summarizes the conclusions of the work and achievements of this research. Future research endeavors are also suggested.

Chapter 2

Literature Review

2.1 Introduction

This chapter is a comprehensive literature review and background for the research on advanced control techniques for UAV-based ground target tracking. The chapter begins with the general context of Unmanned Aerial Vehicles, their history, classification (multi-rotor, fixed-wing, VTOL), and multidimensional applications in industries such as agriculture, surveillance, renewable energy, and military. The review then discusses some control methods, separating low-level dynamic control from higher-level kinematic methods, and discusses the challenges posed in controlling fixed-wing UAVs due to their reliance on aerodynamic lift and inability to hover.

On this basis, the chapter then addresses ground target tracking. It discusses the detection techniques utilized to accurately register ground targets, and contrasts cooperative and non-cooperative tracking environments. The analysis highlights the limitations of simplified models (e.g., 2D Dubins paths) in capturing the richness of real three-dimensional motion.

Large sections of the chapter are devoted to examining MPC and more precisely NMPC as effective methods of managing the complex dynamics and disturbances of tracking. This is explored in depth, and encompasses the limitations of non-convex optimisation that define NMPC and how heuristic algorithms are used in overcoming these limitations.

Besides, the chapter also presents various disturbance rejection approaches. It explains explicit approaches, i.e., wind estimation and actuator error compensation, and data-driven strategies such as ILC, which learns through iteratively adjusting control actions based on the UAV's orbital phase along the target. This chapter also emphasize the need for robust control frameworks that can handle uncertainty and disturbances well without relying on unnecessarily abstract models.

The chapter is closed with a case study of a mining application, where simulation and experimental results verify the efficacy of the NMPC policy in chasing after a moving ground target. The example not only validates the presented methods but also provides the foundation for future work discussions, including flight optimization and stability analysis.

Overall, this chapter establishes the contextual framework by charting the technological context, highlighting key challenges, and inspiring the advanced control strategies derived in the thesis.

2.2 Unmanned Aerial Vehicles

UAVs has burgeoned into a predominant force within the industrial domain, constituting a rapidly expanding technological domain that has undergone substantial advancements in recent years. UAVs have attained a commendable level of reliability, rendering them a valuable instrument across various fields such as agriculture, surveillance, renewable energy systems, defense, exploration of hazardous environments, and many more[1], [2], [3].

In the last decade, UAVs have seen an important decrease in their costs, with a parallel increase in their ease to operate, expanding their applicability across industry, academic, public and private sector [4]. Since then, several different improvements have been made regarding their application. One example is at [5], where it is demonstrated that UAVs have drastically improved their payload capabilities, from simple deliveries ranging up to proven the ability of passengers transportation.

Moreover, application of UAVs in precision agriculture have seen a significant increase in last years, as per [6]. Notably, the latest trends in application involve yield estimation [7], assessment of water stress [8], and wildfire evaluation and prediction [9], just to name a few.

In the urban scope, UAVs have been applied into three main categories: traffic, environment and infrastructure monitoring [10]. Among these, traffic monitoring is the main trend, where different drone-based approaches have been compared to traditional fixed-placed sensors, solving the issue of blind-spots and constant monitoring of moving traffic [11], [12]. However, most of the approaches use a simulation-based validation process instead of considering experimental results.

Furthermore, particular is the case of UAVs being used into renewable energies. A traditional application of drones is the inspection of big photovoltaic arrays. Traditionally this is done using thermal imagery assessment of the actual condition of the PV, detecting specific spots that require attention [13]. Wind turbine blade inspection also found a

mayor aid in drones, where early crack detection can significantly improve the reliability and extend the life of a turbine [14].

The mining sector is another example of the rapid integration of UAVs in the industry. Drones have been integrated into this applications as a cost-effective way to provide potential risk monitoring after mining activity and land reclamation [15].

Military sector has been greatly impacted with the adoption of UAVs. Recent history have shown the potential of UAVs to targeted precision attacks, where even commercial platforms have found their application into the battlefield [16]. Moreover, artificial intelligence applications are presented in [17], where drones can distinguish between military and civilian personnel.

These examples demonstrate the high flexibility in UAVs applications. Moreover, it shows that the interest in this technological assets during the years has done nothing but increase, and there is no sign this is going to reduce in the near future.

2.2.1 Classification of UAVs

In general, UAVs are classified into two major groups: multi-rotor copters and fixed wings, but a third group is also considered, that combines the capabilities of the other two, called VTOLs.

Multi-rotor UAVs

Specifically, multi-rotor UAVs, commonly known as quadcopters or hexacopters depending on the number of motors, are characterized by multiple rotors with propellers that provide lift [18]. Their ease of setup and operation has made them popular among researchers and practitioners. These UAVs are capable of taking off and land in a reduced space. Moreover, they can hover in place, making them suitable to tasks that require stable, stationary positioning [19]. Particular is the case of aerial photography, where a stationary camera is desirable at the moment of capturing the images [20], such as the already covered wind turbine inspections, tunnel construction sites inspection [21], bridge inspection [22], among others. Due to their high maneuverability, drone swarm are also an increasing research topic, where centralized and de-centralized approaches are applied to control a high number of drones at the same time [23]. As seen, multi-rotors UAVs are widely applied into different scenarios, but all these approaches have face the same limitation of battery autonomy that characterize multi-rotors. Due to their aerodynamics, the required energy to remain airborne limit their operation time, being the type of UAVs with less operation time.

Fixed-wing UAVs

Unlike their multi-rotor counterparts, fixed-wing UAVs cannot hover in place; they rely on aerodynamic lift generated by air flowing under and over their wings, necessitating continuous forward motion at cruise speeds to maintain flight [24], [25], [26], [27], which can present a significant control challenge due to their nature. However, their aerodynamic efficiency affords them longer flights duration, making them increasingly suitable for long-range missions, covering wide areas, or monitoring over extended periods [28]. Fixed-wing UAVs are ideal for applications such as environmental monitoring, mapping, and long-distance surveillance [29].

In recent years, there have been improvements related to their aerodynamic control. For instance, authors at [30] propose a variable-structure-based flight controller with multiple algorithm fusion, aimed to maintain stable flight of the UAV, even when operating under disturbances and uncertainties.

Another issue addressed is the turning on fixed-wing UAVs, where the banking angle needs to change gradually. A joint operation scheme is proposed in [31] for energy minimization with bank angle constraint.

In the high level control scope, improvements have been done in the circular formation guidance, as per [32]. Authors introduce a leader/follower scheme that is applied for four UAVs at the same time, controlling their speed and course angle to maintain formation.

Some of the various applications find in the literature for fixed-wing UAVs are related to surveillance and disguised tailing of suspicious mobile targets [33], firefighting aircraft with optimized water-dropping schemes [34], valuable assets inspection located on distant terrains, over-watch for haul truck operation on a mine [35], among others [36], [37], [38].

VTOLs

Hybrid UAVs integrate the advantages of both multi-rotor and fixed-wing UAVs, aiming to overcome the limitations inherent to each type. By combining the VTOL capabilities of multi-rotor UAVs with the aerodynamic efficiency and extended flight endurance of fixed-wing UAVs, Hybrid UAVs offer a versatile platform suitable for a wide range of applications [39], [40].

Despite their advantages, Hybrid UAVs present several challenges that need to be addressed to fully realize their potential, such as complexity in design and control, weight and payload constraints. While fixed-wing flight is more energy-efficient for long-distance travel, the energy demands during vertical takeoff, landing, and mode transitions can offset these gains. Optimizing energy management systems to handle varying flight modes effectively is essential for maximizing overall mission efficiency [41].

2.2.2 Control Strategies for Fixed-Wing UAVs

From a control perspective, fixed-wing UAV applications typically involve one of three main approaches:

Low-level control based on dynamics

The first approach considers the UAV's dynamics, leading to the design of a low-level control law based on the aircraft's dynamic parameters [42], [43], [44]. This method results in a flight controller aimed at maintaining the UAV in safe flight conditions by controlling aerodynamic parameters such as pitch, roll, heading, and speed. Classical control techniques like Proportional-Integral-Derivative (PID) controllers, Linear Quadratic Regulators (LQR), and robust control methods are often employed [45].

Furthermore, approaches such as adaptive control with fault tolerant have shown promising results, as per [46]. Moreover, some other approaches use sliding mode control combined with other control strategies to achieve high tracking of the references [47].

Kinematic control for task execution

The second approach focuses on the kinematics of the UAV, developing control strategies to accomplish specific tasks. In this case, the safe operation of the UAV is managed by an onboard flight controller, while the outer loop handles task-specific control and guidance [48], [49]. This allows for greater flexibility and modularity in mission-specific operations, enabling adaptive task planning and execution without redesigning the low-level control system.

Some kinematic approaches involve the use of trajectory generators and path planning. Then, a kinematic controller approach such as Dubins path is used. In [50], authors first generate a 3D path that consists on a certain number of waypoints, They rely on sequential convex programming techniques to find a feasible collision-free solution. Due to the approximations made, this approach cannot ensure local optimal.

Path-following kinematic controller based on Lyapunov guidance field has also seen a rise in the last years. One example is presented in [51], where authors developed an outer loop controller to follow an specific trajectory for a UAV. Assumptions are made that the flight controller is well-tuned and actuators respond with precision.

2.3 Ground target tracking with fixed wing UAVs

Target tracking utilizing fixed-wing UAVs has emerged as a prominent research area in applications such as monitoring, recognition, agriculture, and even military applications

[52], [53], [54]. This problem is characterized by two distinct branches: detection and control.

2.3.1 Detection techniques

On the detection side, significant advancements have been made to measure the actual position of a ground target based on images and sensor data [55], [56]. Techniques such as computer vision algorithms, machine learning methods, and sensor fusion have been employed to enhance target detection and localization accuracy [57], [58].

More specifically, authors at [59] show a modified You Only Look Once algorithm to detect small targets, even with low quality images. Moreover, computer vision with target position measurement have seen great improvement by the integration of deep learning approaches, as seen in [60].

This quick look at detection techniques give us the foundations of focusing on the control part for ground target tracking, assuming that the position of the target can be accurately measured by different techniques.

2.3.2 Control Laws

On the control side, developments include the design of control laws, overhead tracking, and optimal techniques to keep the UAV in the vicinity of the target [32], [61], [62]. This can be done by keeping an oriented overhead tracking [63] or keeping a safe and constant distance between the target and the aircraft [64].

Most studies simplify the system model by assuming constant speed and altitude, often using a 2D Dubins model. While this reduces complexity, it limits the applicability of such controllers in real-world scenarios where three-dimensional motion and varying speeds are significant factors [65].

2.3.3 Cooperative vs. non-cooperative tracking

In the realm of ground target tracking, two main categories exist:

Cooperative target tracking

In cooperative tracking, the target's trajectory is shared with the UAV beforehand, allowing the aircraft to provide overwatch for the ground vehicle. Since the trajectory is known in advance, controllers can achieve high-accuracy tracking of the reference values [56], [66], [67]. This is usually implemented with an offline trajectory generator, followed by a path-following approach to control the UAV navigation throughout the generated trajectory.

Non-cooperative tracking

Conversely, non-cooperative target tracking poses significant challenges, particularly when the target's motion is unknown or unpredictable. Most models assume that the target is stationary or moving smoothly, simplifying the system model but compromising accuracy in reference tracking or even assuming this information is available [68], [69], [70]. Linear approaches have been developed for trajectory estimation in scenarios with non-aggressive target movements [26], [71], [72], but they often fail to achieve high tracking accuracy for evasive targets.

2.4 MPC Applications

A widely adopted approach for achieving target tracking is to use MPC as the control algorithm [73], [74]. In essence, MPC is a control technique that employs a system model to predict the system behavior over a finite horizon and then computes a control action sequence at each sampling time based on these predictions. This approach is handy for systems with constraints and non-linearities. By utilizing a prediction model, MPC can anticipate and respond to disturbances and optimize control actions to meet desired system performance criteria.

NMPC extends MPC to handle nonlinear systems, but the resulting optimization problem often becomes non-convex, making the task of finding a global solution notably challenging [75], [76]. Heuristic algorithms have emerged as reliable tools for solving non-convex optimization problems, although they often require substantial computational resources.

Advancements in computational tools have enabled the development of real-time NMPC applications for different types of UAVs. In [77], an NMPC was developed for a local path planner with dynamic obstacle avoidance. Similarly, [78] applied an NMPC approach for an attitude controller during the transition from hover to high-speed flight on a VTOL aircraft. Furthermore, [79] proposed an NMPC for a quadcopter UAV landing over an Unmanned Surface Vehicle operating in open water. Some studies present real-time applications of NMPC considering stability analysis; however, this analysis is often limited to the terminal region only [80].

Nevertheless, this literature review shows a lack of examples for the direct application of NMPC for ground target tracking. Most of the works focused on dynamic linearization techniques to cope with the inherit non-convexity that NMPC involves, as per [81].

Despite these advancements, several open problems still require attention, including the stability of three-dimensional nonlinear ground target tracking problem. To address this concern, Lyapunov theory serves as a valuable tool. Typically, this approach involves the identification of a Candidate Lyapunov Function (CLF). However, obtaining a global CLF

for nonlinear systems can be challenging, so stability hinges on the controller's ability to guide the system to a predetermined terminal region, while also ensuring the presence of a local controller that can stabilize the system within that region [82].

2.5 Non-Convex Optimization

Non-convex optimization problems pose greater challenges compared to convex problems due to the absence of properties and guarantees that facilitate tractability. The intricacy arises from the potential existence of multiple local minima, intensifying the complexity of locating the global minimum. Heuristic algorithms, such as genetic algorithms, particle swarm optimization, and Bat Algorithm, emerge as reliable tools but demand high computational proficiency [28], [75], [76], [83].

It is evident that the complexity of the optimization problem varies according to the system. If the system is linear, then the optimization problem becomes a constrained convex problem, which can be solved using convex optimization algorithms, such as gradient descend, interior-point, sequential quadratic programming, Newton's method, just to name a few. These methods are well-established due to the special structure and properties of convex functions. The key advantage of convex optimization is the existence of those efficient algorithms that guarantee convergence to the global minimum. However, those approaches are valid as long as the system runs in a neighborhood of the linearization point. As presented in [35], this is not always the case and the use of linear approaches may underperform when compared to nonlinear approaches.

Nonconvex optimization problems are generally more challenging than convex problems because non-convex optimization lacks the properties and guarantees that make convex optimization tractable. Non-convex optimization problems can have multiple local minima, and the search for the global minimum usually becomes more complex. On this matter, heuristic algorithms are a reliable tool but they demand high computational proficiency. Particularly, the work reported in this thesis uses Bat Algorithm as a non-convex optimization method.

2.6 Disturbance rejection approaches

Fixed-wing UAVs are susceptible to various disturbances, such as wind gusts, asymmetric fuselage, off-centered actuators, and uneven mass distribution, which can significantly affect their positioning.

2.6.1 Explicit approaches

Many approaches that consider disturbances focus on the impact of wind on UAVs, as wind is one of the primary sources of exogenous disturbances for fixed-wing UAVs. Several studies have been published on this topic [84], [85], [86]. Some strategies tackle the wind problem by increasing the UAV's reference speed, thereby reducing the effect of wind on the UAV's kinematics and virtually eliminating the need to estimate wind speed to achieve control objectives. This approach, however, involves the use of high-performance UAVs or the implementation of digital twins when such hardware is not available, which can limit the practicality for more accessible UAV platforms.

Other works incorporate real wind data into the model, obtained from real-time measurements at ground weather stations [48]. However, this method lacks accuracy because the wind profile at ground level differs from that at the UAV's operating altitude. Furthermore, some approaches employ estimators for wind based on onboard airspeed sensors. By utilizing the wind triangle method [87], it is possible to estimate wind influence on different axes based on the wind speed relative to the UAV's heading, thereby estimating deviations from the intended flight path.

Aside from wind, other sources of disturbances can affect UAV positioning, such as asymmetric fuselage design, off-centered actuators, uneven mass distribution, among others. These factors introduce additional uncertainties into the UAV's dynamics, highlighting the need for robust control strategies that can handle various disturbances without relying solely on accurate modeling or estimation.

Kalman filters have been a valuable tool for solving the estimation problem. They rely on the error of the output and the covariance of the state to estimate, correcting the estimation until the difference between the output and the estimated output is minimized. Several works rely on Kalman filters to estimate wind and disturbances, although most approaches focus on path following rather than ground target tracking [88].

2.6.2 Data-driven approaches

Traditional observer-based methods for disturbance estimation require accurate system modeling, which may not be feasible in environments with significant uncertainties or when disturbances are difficult to model explicitly.

To address these challenges, ILC can be integrated with NMPC to enhance tracking performance without relying on explicit disturbance models. ILC leverages data collected from repetitive tasks to refine control actions iteratively [89], [90], [91]. This approach is well suited for iterative tasks that are repetitive over time.

However, ground target tracking is repetitive based on the movement of the UAV around

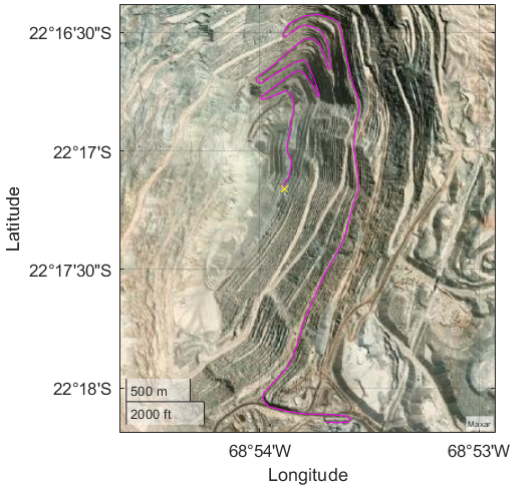


Figure 2.1: Estimated trajectory of a truck in Chuquicamata copper mine.

the target rather than over time. A spatial-based ILC approach can be employed for ground target tracking, where the learning process is synchronized with the phase angle of the UAV's orbit around the target. This method is advantageous over traditional time-based ILC and observer-based methods because it does not depend on precise modeling of disturbances or the target's trajectory. By utilizing phase information, the control scheme can adjust for disturbances that recur at specific phases of the UAV's orbit, effectively improving robustness against environmental factors such as wind gusts or actuator biases.

2.7 Target Tracking - Mine example

From the presented literature review, it is evident that integrating NMPC into online ground target tracking brings challenges yet to be solved. First, the it is necessary to address the applicability of this approach, which involves the use of advanced techniques for non-convex optimization. Moreover, the integration of an experimental prototype with these capabilities involves a careful selection of components, due to the trade-off between computational capabilities and the involved payload limitations for fixed-wing UAVs.

Motivated by this, the obtained results are presented in the conference publication at [35], where NMPC is applied in a real world scenario ground target tracking, alongside a comparison with linear MPC approach. Simulations are conducted using data extracted from satellite images; based on this, a truck trajectory is estimated and a path is generated, considering the loading and unloading locations for the haul truck operating at the mine in CODELCO Chuquicamata mine in northern Chile. This information can be accessed through the mentioned software using the coordinates $22°17'08''S$ $68°53'58''W$. The path is presented in Fig. 2.1.

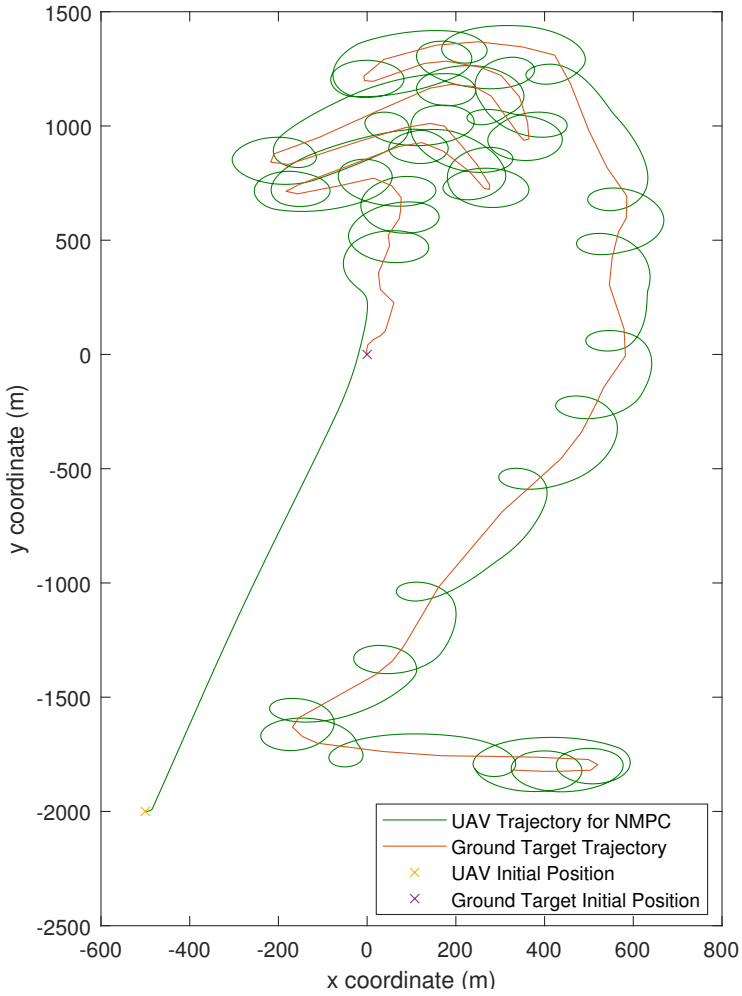
The total road length is 8.6 kilometres. In this case study, it is considered that the truck

is fully loaded and moving uphill at the beginning of the tracking. Additionally, it is considered that the maximum speed is 20 km/h. This is due to the difficult terrain and the slow truck dynamics when breaking and accelerating, among other factors.

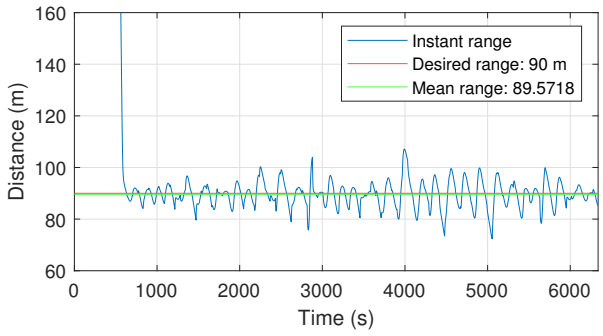
The initial UAV position is located at a point suitable for lift-off and landing for a fixed-wing UAV. If the initial position of the ground target is considered as the origin in a 2-D plane, then new starting point should be allocated around the point $(-500, 2000)$. Results are shown in Fig. 2.2 and it is clear that the UAV under the nonlinear MPC is able to track the target and achieves both range and direction references, which is observed as clockwise turns in Fig. 2.2a and range tracking in Fig. 2.2b. Note that the average value for the range is taken when the controller reaches steady state at the reference distance d_{ss}, i.e., from $t = 630$. This average value is $89.57m$, which has an error of 0.48%, i.e., accurate tracking is achieved during the truck operation.

It is important to note that for this case, there is no result for linear MPC. This is because this approach was not able to find a feasible solution at that distance. Moreover, through simulations, it is determined that the maximum distance from the UAV to the target for the controller to be able to follow the ground target is around 21 times d_{ss}. In other words, for distances higher than 1900 meters in this approach, the linear MPC was not able to control the UAV to track the ground target. This distance might seem sufficient for most applications. However, in the case of large open-pit mine, the feasible initial positions for the UAV tend to be far away from the targets. From Fig. 2.2, it can be seen that the performance of NMPC shows promising results in accurately tracking the ground target. Additionally, persistent over-watch and surveillance are achieved while the truck is in operation. These promising results, which are achieved even with a distant UAV initial position, suggests that this can potentially be used when there is a change in the ground target objective during the UAV operation.

Previous simulation considers the UAV starting point far away from the ground target initial position. However, to obtain results for a linear MPC governing the UAV, a virtual initial position (that is not feasible in practice) is chosen. This virtual position is set at $(130, 230)$. From Fig. 2.3, it is possible to see that both predictive control approaches lead to similar results. Nevertheless, when using NMPC, the UAV takes a shorter path when approaching to d_{ss}, this will cause that the turns made while circumnavigating the ground target tend to be earlier than the case when the UAV is controlled by a linear MPC. In terms of the steady-state range error, both control approaches lead to similar outcomes. However, the use of NMPC results in a slightly more accurate tracking. Table 2.1 summarizes the main results for all case studies, such as optimization time and average range achieve in steady-state. The average range is calculated from the first instance for the UAV's first circumnavigation around the target.



(a) Trajectory for the UAV governed by NMPC and truck in mine.



(b) Range evolution for NMPC and truck in mine.

Figure 2.2: Results for UAV governed by NMPC when tracking a ground target describing a path in Chuquicamata copper mine.

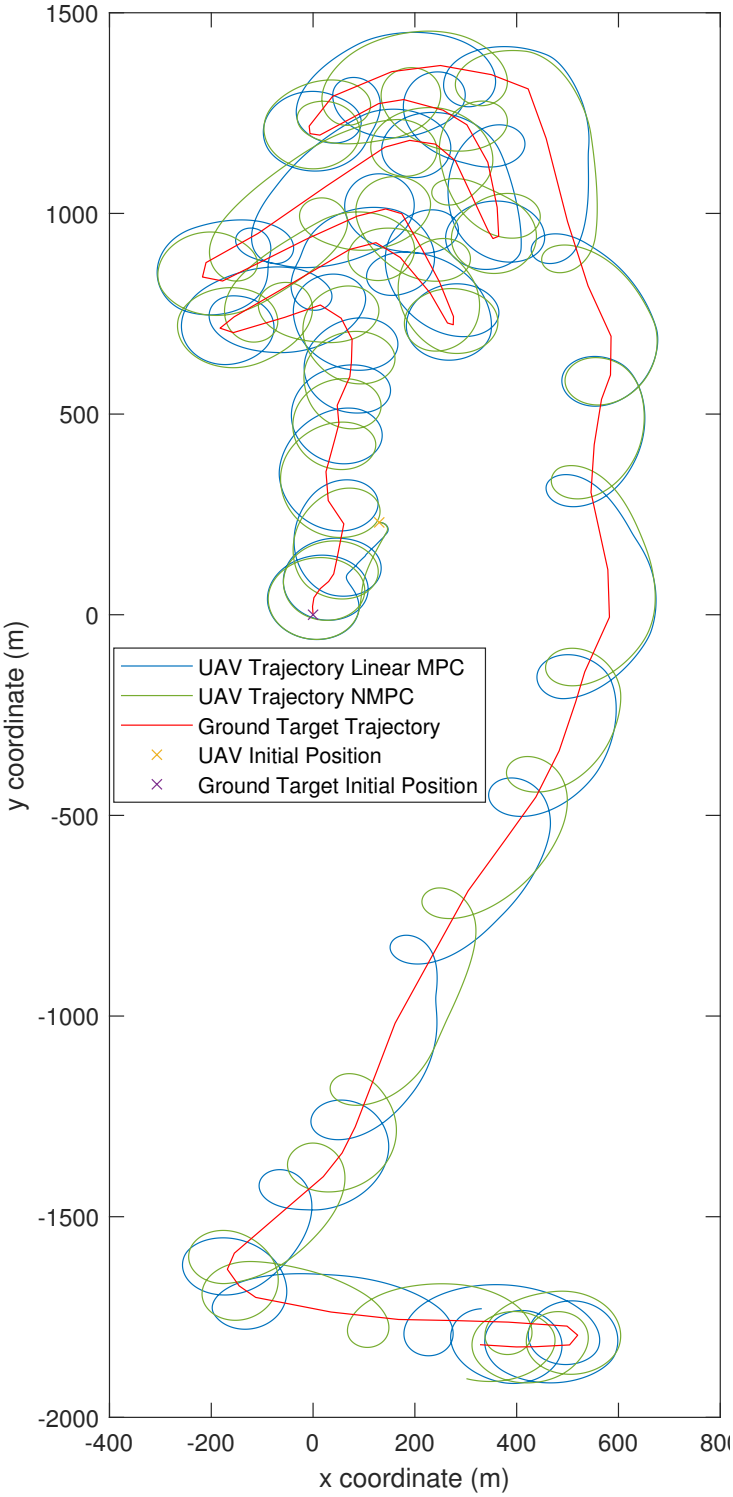


Figure 2.3: Simulation results for MPC and NMPC governing a UAV following a truck in Chuquicamata copper mine

Table 2.1: Simulation results

Path	**Controller**	**Average Optimization Time** (s)	**Average Range** (m)
Linear	MPC	0.0071388	91.77
	NMPC	0.018347	88.7
Sine	MPC	0.005409	92.12
	NMPC	0.017719	87.82
Mine	MPC	0.0050416	91.67
	NMPC	0.016191	89.57

2.8 Discussion

Results show accurate and robust target tracking when using the proposed NMPC strategy. The average maximum error registered throughout all simulations is 2.4%, which can be considered as accurate tracking of the reference range. The three case studies consider a linear path, sine path and a real-world truck trajectory for a copper mine in Chile, CODELCO Chuquicamata.

Notably, in this practical target tracking on a mining site, the linear MPC becomes insufficient to command the UAV. This highlights the benefits of using NMPC, such as lower average range error for all cases, being 0.48% of average range error in particular for the last case study. Furthermore, results of the nonlinear MPC being applied to the UAV with distant initial positions indicate the high reliability that nonlinear controllers have when controlling a system, even when the states are far from the references. On the other hand, NMPC may take more time to optimize the cost function on each iteration, which will impact the total simulation time.

These results motivate research focused on flight optimization, allowing to incorporate the UAV speed into the optimal control problem. Additionally, a natural expansion on the 3D tracking approach will be obtained. Stability of the closed-loop NMPC of the UAV for ground target tracking in the Lyapunov sense for NMPC will be delivered. Moreover, the implementation on a test-bed fixed-wing UAV of the proposal.

Chapter 3

Nonlinear Model Predictive Control for Ground Target Tracking

This study presents the design of a NMPC approach for a fixed-wing UAV to circumnavigate a ground target. First, a nonlinear 3-dimensional target tracking system model is presented. Subsequently, an NMPC is designed and formulated as a non-convex optimal problem. To derive sufficient stability conditions for a nonlinear closed-loop, a linear controller with bounded disturbance is analyzed in a specific terminal region. The controlled trajectory is attracted to the terminal region in the vicinity of the system reference, thereby enabling the use of convex MPC tools for the proposed NMPC. Consequently, the NMPC closed-loop system is proven to reach the terminal region in a fixed prediction horizon, and consequently, the UAV can track the ground target. During the course, an initialization technique is used for optimization to prevent stability compromise by suboptimality. System stability is met for three different speed references with variations in the weighting factors. Extensive simulations are conducted to validate the proposed approach. Experimental results are included, providing insights into the field tests and verifying the control development. The results show that the UAV system is successfully steered to the target reference while effectively remaining within its confines.

3.1 Introduction

UAVs have emerged as a dominant force within the industrial domain and represent a rapidly expanding technology that has undergone significant advancements in recent years. They have achieved a high level of reliability, making them valuable instruments in fields such as agriculture, surveillance, renewable energy systems, defense, exploration of hazardous environments, and more [2], [92].

UAVs are broadly classified into two major groups: multi-rotor and fixed-wing aircraft. The latter presents a particular challenge from a control perspective, as they are unable to maintain a stationary position in the air and must maintain their cruise speed [25], [26], [27].

Regarding fixed-wing UAVs, significant advancements have been made in their control strategies. Broadly, two main approaches exist for this task. The first approach considers the UAV's dynamics, leading to the design of a low-level control law based on the aircraft's dynamic parameters [42], [43]. This method typically results in a flight controller aimed at maintaining the UAV in safe flight conditions.

The second approach focuses on the kinematics of the UAV. Rather than designing a low-level controller, as in the first approach, this one is developed to accomplish specific tasks. In this case, the low-level control is usually handled by an on-board flight controller, while the outer loop manages the task-specific control. The advantage of this approach is that it allows for greater flexibility and modularity in mission-specific operations, enabling task planning and execution to be more adaptive without needing to redesign the low-level control system [48], [49].

Regarding specific tasks, ground target tracking utilizing fixed-wing UAVs has emerged as a prominent research area in applications such as monitoring, recognition, and surveillance in many domains, including agriculture and military applications [52], [53], [54]. In this spectrum, ground target tracking with fixed-wing UAVs has been highlighted for its pivotal role in myriad industrial tasks, accentuating the aircraft's extended operation duration compared to multi-rotor copters [93], [94], [95].

This problem has two well-differentiable branches, detection and control. Several advancements have been made on the detection side to measure the actual position of a ground target based on images [55], [56]. In contrast, the control side has seen the development of control laws, overhead tracking, and other optimal techniques [32], [61], [62]. However, most of these studies are based on a simplified 2D Dubins model, in which the speed and altitude are considered constant, even for nonlinear applications.

MPC uses a mathematical model of the plant to project its future behavior. Based on those prediction, MPC constructs an objective function that evaluates anticipated tra-

jectories and applies penalties to deviations throughout the prediction horizon. Consequently, the MPC problem takes the form of a constrained optimization problem, where a sequence of optimal control actions is obtained at each sampling time. Owing to its forecasting capability, an MPC can anticipate and respond to disturbances and optimize control actions to meet the desired system performance criteria while simultaneously considering nonlinearities and system constraints [96], [97].

A popular application for fixed-wings UAV is ground target tracking. In this scenario, a distance and bearing angle system model is derived to design a UAV control system to automatically approach the ground target from a far distance and then circumnavigating it. There are two main categories in online ground target tracking: cooperative and non-cooperative tracking. In cooperative tracking, the trajectory is previously shared with the UAV, allowing the aircraft to provide over-watch of the ground vehicle. Since the trajectory is known in advance, controllers can achieve high accuracy tracking the reference values [56], [66], [67], and it is usually implemented with an offline trajectory generator, and then using a path following approach to control the UAV navigation throughout the generated trajectory.

Some other approaches have been developed for cooperative tracking, where, instead of the path planning approach, an online controller is used to follow a target, considering full knowledge of the target's movement. Such approaches usually consider linear trajectories, achieving high performance. However, those approaches are not suitable for non-cooperative target tracking with high maneuverability. Usually, the controllers implemented are often regarded to Lyapunov guidance fields and linear approaches, where some optimal approaches have been recently developed [64], [98], [99].

Advancements in MPC for target tracking usually rely on linearized systems that show good performance. Nevertheless, as per [35], there are cases where they fail to effectively steer the UAV towards the target in long-range applications, while nonlinear approaches show effective tracking capabilities, even at great distances from the target. On the other hand, recent developments of computational tools to deal with non-convex optimization problems, make real-time implementation of NMPC for ground target tracking an interesting research topic [100], having the potential to track the ground target from a far distances by fully exploiting the nonlinear dynamic model of the system.

Despite these advancements, several open problems still require attention, including the stability of three dimensional nonlinear ground-target tracking problems.

In this thesis, the formulation and design of an NMPC for a UAV 3D ground target tracking system model is presented. This study considers the formulation of a 5-states 3-inputs 3D ground target tracking system model. Moreover, sufficient conditions to ensure the stability of the proposed NMPC loop in the Lyapunov sense are provided.

To address this, Lyapunov theory serves as a valuable tool in control theory. This approach typically involves identifying a CLF. However, obtaining a global CLF for nonlinear systems is challenging. Consequently, the stability of the system depends on the controller's ability to guide it into a predetermined terminal region. This region serves as a bounded subset of the system states, where a specifically designed local controller can stabilize the system within the defined region [82].

Additionally, considering that weighting factors tuning for NMPC is not a trivial task, a weighting factors variation for NMPC optimization is presented. This provides the weighting factors values that should be used to ensure the stability of the NMPC loop under certain parameters, resulting in a stability domain.

Initially, the nonlinearities are confined within the model using the Lipschitz condition, enabling the utilization of linear MPC tools for terminal region design. A scaling technique is used to find a suitable linear controller with bounded disturbance owing to the Lipschitz condition. This controller stabilizes the system in the terminal region neighborhood of the system reference [101]. This allows one to adapt tools used for convex MPC formulations for the proposed NMPC. Moreover, multi-step NMPC stability is ensured if the last predicted state falls into the terminal region.

From a practical perspective, and considering that the NMPC formulation involves nonconvex optimization, a heuristic algorithm known as the Bat Algorithm, originally presented in [102], to solve the optimization problem is considered. In particular, in the first iteration of the algorithm, the population size is drastically increased alongside movements and change the convergence criteria to obtain a solution that satisfies the stability conditions. During the rest of the UAV operation, the algorithm is initialized with the shifted previous optimal solution. If no better solution is found, the shifted solution is considered. This approach ensures that stability is not compromised by suboptimality.

Stability conditions are used to vary the weighting factors and determine a region of values where the stability of the proposal is assured. Furthermore, simulation results are provided, along with the experimental setup of the hardware and software used and real-world scenario results.

The contributions from this part are summarized as follows:

1. Derivation of a full 3D 5-state 3-inputs ground target tracking nonlinear dynamic model for fixed-wing UAVs applications.

2. The design of an NMPC with ensured stability of the ground target tracking closed-loop system by providing sufficient conditions in the Lyapunov sense.

3. Definition of stability domains for different weighting factors, under different operation conditions.

4. The introduction an initialization technique to solve the optimization problem, ensuring stability is not compromised by possible suboptimality.

5. Experimental verification considering a fixed-wing UAV. To achieve real-time implementation, the proposed NMPC strategy is implemented on a Hobbyking Bixler3 fixed-wing equipped with a companion microcomputer with parallel programming features to optimize the non-convex optimization problem.

The remainder of this chapter is organized as follows: Section 3.4 details the ground target tracking system model, and Section 3.5 describes the NMPC formulation. Section 3.6 presents a stability analysis of the proposed algorithm. Section 3.7 presents the stability domains for different speed references, obtained by varying the weighting factors. Section 3.8 presents the results of the simulation and experimental tests, along with a discussion of the prototype aircraft used in this study. Finally, Section 3.9 concludes this Chapter by summarizing the main contributions of this work and discussing directions for future research.

3.2 Notation

Let $\mathbb{R}$ and $\mathbb{R}_+$ represent the real and non-negative real numbers, respectively. The transpose of a given matrix A is denoted by A'. The maximum and minimum eigenvalue for a matrix A are denoted as $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$, and the induced norm of a matrix A is its largest singular value. The Euclidean norm is denoted as $|\cdot|$ and the ℓ^2-norm is represented by $\|\cdot\|$. Additionally, the weighted squared Euclidean norm is denoted as $\|x\|_P^2 = x'Px$. The state vector x at instant k is denoted as x_k.

3.3 Preliminaries

Consider a discrete-time autonomous system model:

$$x_{k+1} = f(x_k), \qquad f(0) = 0 \tag{3.1}$$

where $x_k \in \mathbb{X} \subset \mathbb{R}^n$ and $\mathbb{X}$ denotes the state constraint set.

Definition 1. *Class-$\mathcal{K}$ Functions: A function $\alpha : \mathbb{R}_+ \to \mathbb{R}_+$ is said to be a class-$\mathcal{K}$ function if it is continuous, strictly increasing and $\alpha(0) = 0$. Function α is said to be a class-$\mathcal{K}_\infty$ function if it is a class-$\mathcal{K}$ function and $\alpha(a) \to \infty$ if $a \to \infty$. A function $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ is said to be a class-$\mathcal{KL}$ function if it is continuous and for each $k \geq 0$ $\beta(\cdot, k)$ is a class-$\mathcal{K}$ function and for each $s \geq 0$ $\beta(s, \cdot)$ is non-increasing and satisfies $\beta(s, k) \to 0$ when $k \to \infty$.*

Definition 2. *(Lyapunov Stability): A function $V : \mathbb{R}^n \to \mathbb{R}$ is said to be a Lyapunov function in $\mathcal{A} \subset \mathbb{R}^n$ for the system* (3.1) *if there exists a compact set $\Omega \subseteq \mathcal{A}$, neighborhood*

of the origin, $x = 0$, some class-$\mathcal{K}_\infty$ functions α_1, α_2 and α_3 such that

$$V(|x|) \geq \alpha_1(|x|), \qquad \forall x \in \mathcal{A}, \tag{3.2a}$$
$$V(|x|) \leq \alpha_2(|x|), \qquad \forall x \in \Omega, \tag{3.2b}$$
$$V(f(x)) - V(x) \leq -\alpha_3(|x|), \qquad \forall x \in \mathcal{A}. \tag{3.2c}$$

If $\mathcal{A} \triangleq \mathbb{R}^n$ then V is said to be a global Lyapunov function.

Theorem 1 ([103])**.** *If the system* (3.1) *admits a Lyapunov function in $\mathcal{A}$, then the origin is asymptotically stable in $\mathcal{A}$.*

Definition 3. *(Lipschitz condition): Let $f : \mathbb{R}^n \to \mathbb{R}^n$, say f is locally Lipschitz on a set $\mathcal{A} \subset \mathbb{R}^n$ if there exists a constant $L \in \mathbb{R}_+$ such that*

$$\|f(x) - f(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathcal{A}. \tag{3.3}$$

3.4 Kinematic system model formulation

The kinematic system model establishes the relationship between the movement of a fixed-wing UAV and a point in the space. In this particular case, the reference point depicts a ground target. Under this assumption, consider that the target's position can be precisely determined, yet its trajectory is unpredictable. The continuous-time kinematic model for the fixed-wing UAV regarding the ground target is represented as follows:

$$\dot{x}_p = \mathcal{V}\cos(\chi)\cos(\psi), \tag{3.4a}$$
$$\dot{y}_p = \mathcal{V}\cos(\chi)\sin(\psi), \tag{3.4b}$$
$$\dot{z}_p = \mathcal{V}\sin(\chi), \tag{3.4c}$$
$$\dot{\chi} = u_\chi, \tag{3.4d}$$
$$\dot{\psi} = u_\psi, \tag{3.4e}$$
$$\dot{\mathcal{V}} = u_\mathcal{V}. \tag{3.4f}$$

In this model, the state vector is defined as $X_v = [x_p, y_p, z_p, \chi, \psi, \mathcal{V}]'$. Here, x_p, y_p, and z_p represent the 3D coordinates of the UAV, $\mathcal{V}$ is the UAV speed, and χ and ψ are the pitch and heading angles of the UAV, respectively [74]. Furthermore, the control inputs u_χ, u_ψ, and $u_\mathcal{V}$ represent the rate of change for the pitch angle χ, heading angle ψ, and speed $\mathcal{V}$. These control inputs are essential for maneuvering the UAV and tracking the ground target effectively. As the UAV must generate enough lift to remain airborne, its speed $\mathcal{V}$ is constrained to the interval $[\mathcal{V}_{\min}, \mathcal{V}_{\max}]$, where $\mathcal{V}_{\max}$ is the max speed that the UAV can achieve, while $\mathcal{V}_{\min}$ is the stall speed, i.e., the minimum speed required to produce lift.

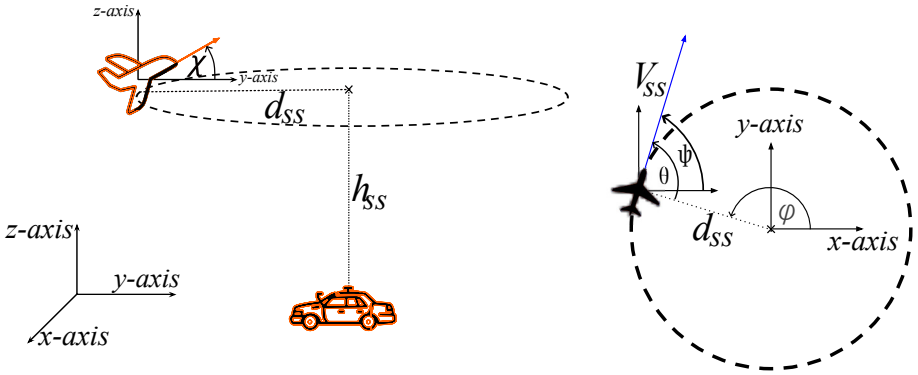


Figure 3.1: Kinematic relation between the UAV and Ground Target.

3.4.1 Two-dimensional Case

In the kinematic system model context, assuming that the pitch angle is fixed at $\chi = 0$ and remains unchanged implies that the altitude $z_p > 0$ will be constant throughout the UAV's motion, as shown in Fig. 3.1. Additionally, considering the UAV speed $\mathcal{V}$ to be constant, the model can be represented using a simplified 2D Dubins model:

$$\dot{x}_p = \mathcal{V}\cos(\psi), \tag{3.5a}$$

$$\dot{y}_p = \mathcal{V}\sin(\psi), \tag{3.5b}$$

$$\dot{\psi} = u_\psi. \tag{3.5c}$$

The challenge is to automatically control the input u_ψ such that the aircraft can orbit around a target in a known position (x_r, y_r) at a distance d_{ss}. In the present work, instant range measurements d between the UAV and the ground target, and the range rate $\dot{d}$ are considered. The range is described based on the Euclidean distance:

$$d = \sqrt{(x_p - x_r)^2 + (y_p - y_r)^2}. \tag{3.6}$$

For the convenience of notation, the ground target is considered to be placed at the origin of the coordinated frame. The proposed control system will take into account the reference angle ϕ, the heading angle ψ, and the bearing angle θ from the reference vector to the heading vector, as depicted in Fig. 3.1. From that, θ can be described as:

$$\theta = \pi - \phi + \psi. \tag{3.7}$$

Based on these premises and the work shown in [104], an expression for $\dot{d}$ can be obtained from (3.6):

$$\dot{d} = \frac{1}{\sqrt{x^2 + y^2}}(x\dot{x} + y\dot{y}) = \cos(\phi)\dot{x} + \sin(\phi)\dot{y}. \tag{3.8}$$

Then, by considering $\dot{x}$ and $\dot{y}$ from the system model (3.5) and the angle relationship in (3.7), from (3.8) it follows that:

$$\dot{d} = -\mathcal{V}\cos\theta. \tag{3.9}$$

This shows that there is a direct relationship between the bearing angle θ and the range rate $\dot{d}$. From (3.7), it is clear that:

$$\dot{\theta} = -\dot{\phi} + \dot{\psi} = -\dot{\phi} + u_{\psi}, \tag{3.10}$$

where:

$$\dot{\phi} = \frac{\cos(\phi)\dot{y} - \sin(\phi)\dot{x}}{\sqrt{x^2 + y^2}} = -\frac{\mathcal{V}}{d}\sin\theta, \tag{3.11}$$

which leads to the transformed system from $(\dot{x}, \dot{y}, \dot{\psi})$ to $(\dot{d}, \dot{\theta})$, i.e.:

$$\dot{d} = -\mathcal{V}\cos\theta, \tag{3.12a}$$

$$\dot{\theta} = \frac{\mathcal{V}}{d}\sin\theta + u_{\psi}. \tag{3.12b}$$

Remark 1. *Equation* (3.12) *exhibits a spurious singularity at* $d = 0$. *In the NMPC implementation, the distance measurement is clipped to a small positive threshold as* d *approaches zero, preventing the controller from ever reaching the singularity. This is also true for the 3D scenario.*

The control objective is not only to lead the UAV to a desired radius reference, d_{ss}, during transient, but also to keep the UAV at this radius during the steady-state. Therefore, it is important to determine what is the required input, u_{ss}, to keep this desired steady-state. Once the radius reference is achieved, then to keep a fixed radius in a desired steady-state, its derivative must become zero. Thus, from (3.12a), it follows that: $\dot{r} = 0 \implies \theta_{ss} = \{\pi/2,\ 3\pi/2\}$. Since this angle needs to be kept constant to keep a fixed radius, so that $\dot{\theta} = 0 \implies u_{ss} = \{-\mathcal{V}/d_{ss},\ \mathcal{V}/d_{ss}\}$. the proposed MPC strategy considers a quadratic cost function comprised of the sum of forecast system state tracking errors at specific sampling instants. Hence, the system model needs to be discretized. Since the system model is non-linear, the representation for next sampling instants are obtained by applying a forward Euler discretization method to (3.12), leading to:

$$d_{k+1} = d_k - T_s\mathcal{V}\cos\theta_k, \tag{3.13a}$$

$$\theta_{k+1} = \theta_k + T_s\left(\frac{\mathcal{V}}{d_k}\sin\theta_k + u_{\psi,k}\right), \tag{3.13b}$$

where T_s is the sampling time used in the discretization. Note that in this approach, the next instant radius d_{k+1} does not directly depend on the control input, $u_{\psi,k}$. However, it does depend on θ_k, which in turn depends on $u_{\psi,k}$. This will cause that the prediction horizon must be at least $N = 2$ to maintain a dependence between the control input, $u_{\psi,k}$, and distance, d_k. To facilitate the MPC formulation, a discrete-time dynamic model of the system tracking error can be derived, which leads to:

$$x_{k+1} = g(x_k) + Bu_k, \tag{3.14}$$

where $g(x_k) = \begin{bmatrix} g_1 & g_2 \end{bmatrix}'$ is given by:

$$g_1(x_k) = x_{1,k} + T_s \mathcal{V} \sin(x_{2k}), \tag{3.15a}$$

$$g_2(x_k) = x_{2,k} + T_s \left(\frac{\mathcal{V} \cos(x_{2,k})}{x_{1,k} + d_{ss}} - \frac{\mathcal{V}}{d_{ss}} \right), \tag{3.15b}$$

and $B = \begin{bmatrix} 0 & T_s \end{bmatrix}'$, $x_k = \begin{bmatrix} x_{1,k} & x_{2,k} \end{bmatrix}' = \begin{bmatrix} d_k - d_{ss} & \theta_k - \theta_{ss} \end{bmatrix}'$, and $u_k = u_{\psi,k} - u_{ss}$, with $u_{ss} = -\mathcal{V}/d_{ss}$ for clockwise tracking, are the tracking error form of the states and input, respectively.

3.4.2 Three-dimensional Case

Considering the Fig. 3.1b, the relationship between the bearing angle, heading angle, and reference angle from the ground target and the UAV is depicted as per (3.7). This implies $\dot{\phi} = \dot{\psi} - \dot{\theta}$. Moreover, from Fig. 3.1a the distance between the UAV and the target tracking is represented as per (3.6). For simplicity, $x_{GT} = y_{GT} = 0$, i.e. the ground target is at the origin of the 3D frame. Furthermore, the derivative of the distance $\dot{d}$ can be depicted as per(3.8). By considering (3.4) and replacing, it leads to

$$\dot{\phi} = \frac{-y_p}{x_p^2 + y_p^2} \dot{x}_p + \frac{x_p}{x_p^2 + y_p^2} \dot{y}_p = \frac{-\mathcal{V} \cos\chi \cos\theta}{d}, \tag{3.16}$$

Considering the rate of change of the altitude $\dot{z}_p$, the rate of change of the speed $\dot{\mathcal{V}}$, the rate of change of the pitch angle $\dot{\chi}$ at (3.4), and by (3.7), (3.8) and (3.16), the 3D target tracking model is depicted as:

$$\dot{d} = -\mathcal{V} \cos\theta \cos\chi, \tag{3.17a}$$

$$\dot{\theta} = \frac{\mathcal{V} \sin\theta \cos\chi}{d} + u_\psi, \tag{3.17b}$$

$$\dot{h} = \mathcal{V} \sin\chi, \tag{3.17c}$$

$$\dot{\chi} = u_\chi, \tag{3.17d}$$

$$\dot{\mathcal{V}} = u_\mathcal{V}, \tag{3.17e}$$

where d is the distance between the UAV and the ground target, θ is the bearing angle, and, for ease of notation, $h \triangleq p_z$. Note that the singularity is prevented using the technique discussed in Remark 1. The control objective is to lead the UAV to a desired state reference, $x_{ss} = \begin{bmatrix} d_{ss} & \theta_{ss} & h_{ss} & \chi_{ss} & \mathcal{V}_{ss} \end{bmatrix}'$, and maintain it during the steady-state. Therefore, is crucial to determine the required steady-state input vector, $u_{ss} = \begin{bmatrix} u_{\psi,ss} & u_{\chi,ss} & u_{\mathcal{V},ss} \end{bmatrix}'$, to maintain the closed-loop system at steady-state. Thus, from (3.17a), it follows that $\dot{d} = 0 \implies \theta_{ss} \in \{\pi/2, -\pi/2\}$ and $\chi_{ss} = 0$, where θ_{ss} determines the UAV direction during tracking. Because the bearing angle must be kept

constant to maintain a fixed distance and altitude, from (3.17b), it implies

$$\dot{\theta}{=}0 \implies u_{\psi,ss}{\in}\left\{\frac{-\mathcal{V}_{ss}}{d_{ss}},\ \frac{\mathcal{V}_{ss}}{d_{ss}}\right\}.$$

Finally, when reaching the reference altitude and speed, from (3.17d) and (3.17e), $\dot{\chi} = 0 \implies u_{\chi,ss} = 0$, and $\dot{\mathcal{V}} = 0 \implies u_{\mathcal{V},ss} = 0$ This steady-state analysis allows us to define the state and input tracking error form as:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \end{bmatrix} = \begin{bmatrix} \Delta r \\ \Delta\theta \\ \Delta h \\ \Delta\chi \\ \Delta\mathcal{V} \end{bmatrix} = \begin{bmatrix} d(t){-}d_{ss} \\ \theta(t){-}\theta_{ss} \\ h(t){-}h_{ss} \\ \chi(t){-}\chi_{ss} \\ \mathcal{V}(t){-}\mathcal{V}_{ss} \end{bmatrix} \tag{3.18}$$

and

$$u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} = \begin{bmatrix} \Delta u_\psi \\ \Delta u_\chi \\ \Delta u_\mathcal{V} \end{bmatrix} = \begin{bmatrix} u_\psi(t){-}u_{\psi,ss} \\ u_\chi(t){-}u_{\chi,ss} \\ u_\mathcal{V}(t){-}u_{\mathcal{V},ss} \end{bmatrix} \tag{3.19}$$

Considering this steady-state analysis, $\theta_{ss} = \pi/2$ and $u_{\psi,ss}{=}{-}\mathcal{V}_{ss}/d_{ss}$. Using the forward Euler method, the following discrete-time tracking error form nonlinear system model is obtained:

$$x_{k+1} = g(x_k) + Bu_k, \tag{3.20}$$

where $g(x_k) = \begin{bmatrix} g_1(x_k) & g_2(x_k) & g_3(x_k) & g_4(x_k) & g_5(x_k) \end{bmatrix}'$ is given by

$$g_1(x_k) = x_{1,k}{+}T_s(x_{5,k}{+}\mathcal{V}_{ss})\cos(x_{4,k})\sin(x_{2,k}), \tag{3.21a}$$

$$g_2(x_k) = x_{2,k}{+}T_s\frac{(x_{5,k}{+}\mathcal{V}_{ss})}{(x_{1,k}{+}d_{ss})}\cos(x_{4,k})\cos(x_{2,k}) - T_s\frac{\mathcal{V}_{ss}}{d_{ss}}, \tag{3.21b}$$

$$g_3(x_k) = x_{3,k}{+}T_s(x_{5,k}{+}\mathcal{V}_{ss})\sin(x_{4,k}), \tag{3.21c}$$

$$g_4(x_k) = x_{4,k}, \tag{3.21d}$$

$$g_5(x_k) = x_{5,k}, \tag{3.21e}$$

with T_s being the sampling time, and B defined as

$$B = \begin{bmatrix} 0 & 0 & 0 \\ T_s & 0 & 0 \\ 0 & 0 & 0 \\ 0 & T_s & 0 \\ 0 & 0 & T_s \end{bmatrix}. \tag{3.22}$$

Remark 2. *Since the NMPC loop relies on the optimization of a cost function, as detailed in Section 3.5, it is essential to express the system's dynamics in terms of the tracking*

error. This formulation ensures that the optimization process minimizes the cost function by driving the system states toward zero, thereby achieving the desired reference values. Furthermore, the discretization is required for a practical implementation of the proposed approach on a companion computer, enabling real-time execution, as will be covered in Section 3.8.3.

Remark 3. *Considering that there are two approaches presented in this section for target tracking, the remainder of this document considers the full 3D kinematics.*

3.4.3 Linearized System Model

Although it is not the goal of this work, linear tools are adopted to establish sufficient conditions to guarantee stability of the NMPC closed-loop system. For this, a linearized model around a desired set-point is obtained, and then to design a linear controller. The linearization point is chosen at $(x_{ss}) = (d_{ss}, \pi/2, h_{ss}, 0, V_{ss})$, following the conditions given in the steady-state analysis. From (3.17), using forward Euler method to discretize and considering tracking error form, the linear system is represented by:

$$x_{k+1} = Ax_k + Bu_k, \tag{3.23}$$

where

$$A = \begin{bmatrix} 1 & T_s\mathcal{V}_{ss} & 0 & 0 & 0 \\ -T_s\mathcal{V}_{ss}/d_{ss}^2 & 1 & 0 & 0 & T_s/d_{ss} \\ 0 & 0 & 1 & T_s\mathcal{V}_{ss} & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Considering (3.23) and (3.20), the following nonlinear system is obtained:

$$x_{k+1} = f(x_k, u_k) = Ax_k + Bu_k + \eta(x_k), \tag{3.24}$$

where $\eta(x_k) = g(x_k) - Ax_k$, represents the difference between the nonlinear system model and its linearized counterpart. It is worth noting that as the state deviation, x_k, approaches zero, so it does the function $\eta(x_k)$, indicating that the nonlinear system and its linear approximation become increasingly similar in this limit.

3.5 NMPC Formulation for UAV Ground Target Tracking

Consider the nonlinear discrete-time system model presented in (3.20), where $x_k \in \mathbb{R}^n$ and $u_k \in \mathbb{R}^m$. The control problem can be expressed via the minimization of the following cost function:

$$V_N(x, \vec{u}) = \sum_{i=0}^{N-1} L(\hat{x}_i, \hat{u}_i) + V_f(\hat{x}_N), \tag{3.25}$$

where $L(\hat{x}_i, \hat{u}_i) = \|\hat{x}_i\|_Q^2 + \|\hat{u}_i\|_R^2$ is the stage cost, with Q and R being positive definite matrices and N is the prediction horizon. Furthermore, $V_f(\hat{x}_N) = \|\hat{x}_N\|_{P_\mu}^2$ denotes the terminal cost and P_μ being positive definite[1]. Vector $\vec{u}$ contains the tentative actions, $\hat{u}_k$, over the prediction horizon, i.e.:

$$\vec{u} = \begin{bmatrix} \hat{u}_0 & \cdots & \hat{u}_{N-1} \end{bmatrix}' \in \mathbb{R}^{m \cdot N}. \tag{3.26}$$

Then, the NMPC can be expressed as the following optimal control problem for the current state $\hat{x}_0$:

$$\mathbb{P}_N(\hat{x}_0) : V_N^{op}(\hat{x}_0, \vec{u}) = \min_{\vec{u}} \{V_N(\hat{x}_0, \vec{u})\}, \tag{3.27a}$$

$$s.t. :$$

$$\hat{x}_{i+1} = f(\hat{x}_i, \hat{u}_i), \tag{3.27b}$$

$$\hat{u}_i \in \mathbb{U}, \tag{3.27c}$$

$$\hat{x}_i \in \mathbb{X}, \tag{3.27d}$$

$$\hat{x}_N \in \mathbb{X}_f \subseteq \mathbb{X}, \tag{3.27e}$$

for all $i \in \{0, \cdots, N-1\}$. In this case, (3.27b) is the system constraint, (3.27c) and (3.27d) are the input and state constraints, respectively. Finally, (3.27e) is the terminal constraint, where the terminal region $\mathbb{X}_f$ can be selected to guarantee the stability of the resulting closed-loop. The optimal input sequence $\vec{u}^{op}$ minimizes cost function (3.25), that is

$$\vec{u}^{op}(\hat{x}_0) \triangleq \arg \left\{ \min_{\vec{u} \in \mathcal{U}(\hat{x}_0)} V_N(\hat{x}_0, \vec{u}) \right\}, \tag{3.28}$$

where $\mathcal{U}(x)$ represents the set of all admissible input sequences that satisfy constraints (3.27c)-(3.27e). Subsequently, the optimal input sequence that minimizes the cost function is expressed as follows:

$$\vec{u}^{op}(\hat{x}_0) = \begin{bmatrix} (\hat{u}_0^{op})' & \cdots & (\hat{u}_{N-1}^{op})' \end{bmatrix}'. \tag{3.29}$$

If $\vec{u}^{op}(\hat{x}_0)$ is applied to system (3.27b), the resulting optimal state sequence is obtained as follows:

$$\vec{x}^{op}(\hat{x}_0) = \begin{bmatrix} \hat{x}_0' & (\hat{x}_1^{op})' & \cdots & (\hat{x}_N^{op})' \end{bmatrix}'. \tag{3.30}$$

The domain of attraction of the cost function $V_N(\hat{x}_0)$ can be defined as:

$$X_N \triangleq \left\{ \hat{x}_0 \in \mathbb{X} : \mathcal{U}(\hat{x}_0) \neq \emptyset \right\}. \tag{3.31}$$

Therefore, X_N contains all $\hat{x}_0 \in \mathbb{X}$ such that there exists a control sequence $\vec{u} \in \mathcal{U}(\hat{x}_0)$ satisfying conditions (3.27c) to (3.27e). Using the receding horizon control [105], the solution of the optimal problem $\mathbb{P}_N(\hat{x}_0)$ in (3.27a) yields the NMPC law $\kappa_N(\cdot) : X_N \rightarrow \mathbb{U}$,

$$\kappa_N(x_0) \triangleq \hat{u}_0^{op}. \tag{3.32}$$

[1]Note that $P_\mu = \mu P$, where $\mu \geq 1$ is a scaling factor. The benefit of using this notation will be clarified in Theorem 2.

Finally, the resulting NMPC loop for the ground target tracking problem is represented by:

$$x_{k+1} = f(x_k, \kappa_N(x_k)). \tag{3.33}$$

3.6 Stability Analysis of the closed loop system

In this section, sufficient conditions for the UAV ground target tracking problem are derived to ensure the stability of the NMPC loop presented in (3.33). While the ultimate goal of this section is to prove the stability of the NMPC closed loop presented in (3.33), linear tools to derive sufficient conditions to ensure the stability of the closed-loop nonlinear 3D target tracking system are used.

3.6.1 Optimal Solution for the Linearized System Model Without Terminal Constraint

Lemma 1. *For the linear system* (3.23)*, if* $\mathbb{P}_N(x)$ *presented at* (3.27a) *does not consider the state and input constraints, i.e.* $\mathbb{U} \triangleq \mathbb{R}^m$ *and* $\mathbb{X} = \mathbb{X}_f \triangleq \mathbb{R}^n$*, then* $V_N(\hat{x}_0, \vec{u})$ *is minimized when:*

$$\vec{u}_{uc}^{op} \triangleq \arg\left\{ \min_{\vec{u} \in \mathbb{R}^{Nm}} V_N(\hat{x}_0, \vec{u}) \right\} \triangleq -H_N^{-1} F_N \hat{x}_0, \tag{3.34}$$

with H_N *and* F_N *defined as per [106].*

3.6.2 Finding a Suitable Local Controller $\kappa_f(x)$

A commonly employed approach for ensuring stability in NMPC is to determine a known controller, denoted as κ_f, that can stabilize the nonlinear system within the terminal region $\mathbb{X}_f$, see e.g. [107]. Motivated by this, using standard tools for linear MPC and the fact that $\eta(x) \to 0$ when $x \to 0$, a feasible local controller by observing the properties of the optimal solution given in (3.34) at one step (i.e., $N = 1$) and $\mu = 1$ is defined. The proposed local controller is defined as follows:

$$\kappa_f(x) = Kx, \tag{3.35}$$

where

$$K = -H_1^{-1} F_1 = -(B'PB + R)^{-1} B'PA, \tag{3.36}$$

with H_1 and F_1 as per [106].

Based on this and considering the terminal cost, it is possible to define the terminal region $\mathbb{X}_f$ in (3.27e) for system (3.17) as follows:

$$\mathbb{X}_f \triangleq \{x' P_\mu x \leq \varphi_x : x \in \mathbb{X}, \kappa_f(x) \in \mathbb{U}\}, \tag{3.37}$$

where $\varphi_x \in \mathbb{R}_+$ is designed to obtain the largest ellipsoid for all $x \in \mathbb{X}_f$, $\kappa_f(x) \in \mathbb{U}$ for system (3.24). Moreover, the origin belongs to $\mathbb{X}$ and $\mathbb{U}$, which means $\mathbb{X}_f \neq \emptyset$.
Finally, considering the nominal system presented in (3.24), the closed loop system is obtained as follows:

$$x_{k+1} = A_K x_k + \eta(x_k), \tag{3.38}$$

where, $A_K = A + BK$.

Moreover, it is evident that $\eta(\cdot)$ is twice differentiable, and as $\eta(0) = 0$, it implies that there exists a constant $\gamma \in \mathbb{R}_+$ such that

$$\|\eta(x)\| \leq \gamma \|x\|, \tag{3.39}$$

on the set $\mathbb{X}_f$. Assuming that $\mathbb{X}_f \subset \mathbb{R}^n$, then, by Definition 3, $\eta(x_k)$ can be bounded using Lipschitz's method. That is $\eta(x)$ is locally Lipschitz on $\mathbb{X}_f$.

Theorem 2. *Consider* $a_1 = \lambda_{\min}(P_\mu)$, $a_2 = \lambda_{\max}(P_\mu)$, *and*

$$\zeta \;=\; \lambda_{\min}(P_\mu^{-1/2}(Q^*)P_\mu^{-1/2}), \tag{3.40}$$

where,

$$Q^* \triangleq Q + K'RK,$$

and $Q_\mu^* \triangleq \mu Q^*$ *with a scaling factor* $\mu \geq 1$. *Let* V_f *in* (3.25) *be designed such that matrix* P_μ *is chosen as the solution to the algebraic Riccati equation:*

$$A_K' P_\mu A_K - P_\mu + Q_\mu^* = 0. \tag{3.41}$$

If the nonlinearities of the fixed-wing UAV model are bounded by (3.39)*, and the following condition*

$$\gamma^2 + 2\gamma \|A_K\|_{P_\mu} \leq \mu\zeta \tag{3.42}$$

is satisfied, then $\kappa_f(x)$ *in* (3.35) *is a local stabilizing controller in* $\mathbb{X}_f$ *for nonlinear system* (3.17).

The proof of Theorem 2 is presented herein.

Proof. By applying Theorem 1, using V_f as a Lyapunov candidate function, $\alpha_1(s) = a_1 s^2$ and $\alpha_2(s) = a_2 s^2$, therefore, inequalities (3.2a) and (3.2b) hold $\forall x \in \mathbb{X}_f$. Moreover, direct calculations provide that:

$$\begin{aligned} V_f(x_{k+1}) - V_f(x_k) + L(x_k, \kappa_f(x_k)) =& 2\eta(x_k)' P_\mu A_K x_k + \eta(x_k)' P_\mu \eta(x_k) \\ &+ x_k'(A_K' P_\mu A_K - P_\mu)x_k + x_k' Q^* x_k \\ =& -x'(Q_\mu^* - Q^*)x + 2\eta(x_k)' P_\mu A_K x_k \\ &+ \eta(x_k)' P_\mu \eta(x_k). \end{aligned} \tag{3.43}$$

Let $x_k = x$. Note that, by (3.39), it is possible to find the Lipschitz constant γ by sweeping $\|x\|_{P_\mu}$ within the terminal region and find its maximum, as follows:

$$\gamma = \max_{0<\|x\|_{P_\mu}\leq\varphi_x} \frac{\|\eta(x)\|_{P_\mu}}{\|x\|_{P_\mu}}, \tag{3.44}$$

The previous expression allows us to bound the last term in (3.43), so that the following inequality holds:

$$\eta(x)' P_\mu \eta(x) \leq \gamma^2 \|x\|_{P_\mu}^2. \tag{3.45}$$

Moreover,

$$\begin{aligned} 2\eta(x)' P_\mu A_K x &= 2(\eta(x)' {P_\mu}^{1/2})({P_\mu}^{1/2} A_K x) \\ &\leq 2\left\| {P_\mu}^{1/2}\eta(x)\right\| \left\| {P_\mu}^{1/2} A_K x\right\| \\ &\leq 2\|\eta(x)\|_{P_\mu} \|A_K x\|_{P_\mu} \\ &\leq 2\gamma \|A_K\|_{P_\mu} \|x\|_{P_\mu}^2. \end{aligned} \tag{3.46}$$

Additionally,

$$\begin{aligned} x'(Q^*)x &= x' {P_\mu}^{1/2} {P_\mu}^{-1/2} (Q^*) {P_\mu}^{-1/2} {P_\mu}^{1/2} x \\ &\geq \lambda_{\min}({P_\mu}^{-1/2}(Q^*){P_\mu}^{-1/2}) \|x\|_{P_\mu}^2 \\ &\geq \zeta \|x\|_{P_\mu}^2. \end{aligned} \tag{3.47}$$

Consequently, (3.43) follows that:

$$V_f(x_{k+1}) - V_f(x_k) + L(x_k, \kappa_f(x_k)) \leq -((\mu-1)\zeta - 2\gamma\|A_K\|_{P_\mu} - \gamma^2)\|x_k\|_{P_\mu}^2. \tag{3.48}$$

Therefore,

$$\begin{aligned} V_f(x_{k+1}) - V_f(x_k) &\leq -((\mu-1)\zeta - 2\gamma\|A_K\|_{P_\mu} - \gamma^2)\|x_k\|_{P_\mu}^2 - L(x_k, \kappa_f(x_k)) \\ &\leq -((\mu-1)\zeta - 2\gamma\|A_K\|_{P_\mu} - \gamma^2)\|x_k\|_{P_\mu}^2 - \zeta\|x_k\|_{P_\mu}^2 \\ &\leq -(\mu\zeta - 2\gamma\|A_K\|_{P_\mu} - \gamma^2)\|x_k\|_{P_\mu}^2. \end{aligned} \tag{3.49}$$

By (3.42), property (3.2c) holds for $\alpha_3(s) = -a_3 s^2$, where

$$a_3 = (\mu\zeta - 2\gamma\|A_K\|_{P_\mu} - \gamma^2).$$

following that

$$\Delta V_f(x) \leq -a_3 \|x\|\|_{P_\mu}^2, \quad \forall x \in \mathbb{X}_f. \tag{3.50}$$

The previous result allows us to establish the following relationship:

$$V_f(x_{k+1}) \leq \delta V_f(x_k), \quad \forall x \in \mathbb{X}_f. \tag{3.51}$$

Considering inequality (3.50),

$$V_f(x_{k+1}) \leq V_f(x_k) - a_3\|x_k\|_{P_\mu}^2 \leq (a_2 - a_3)\|x_k\|_{P_\mu}^2, \tag{3.52}$$

with $a_3 \leq a_2 \implies \delta = 1 - a_3/a_2 \in [0, 1]$. By iterating (3.51), it is possible bound $\|x_k\|^2_{P_\rho}$ as:

$$\|x_k\|^2 \leq \frac{a_2}{a_1}\delta^k \|x_0\|^2. \tag{3.53}$$

This implies that $\lim_{k\to\infty} \|x_k\| = 0$ with $x_0 \in \mathbb{X}_f$. Consequently, from Theorem 1, $\kappa_f(x)$ in (3.35) is a local stabilizing controller in $\mathbb{X}_f$ for the system (3.24). □

Remark 4. *Results from Theorem 2 establish that, provided the set $\mathbb{U}$ where (3.42) is satisfied, the one-step controller in (3.35) is locally stabilizing in $\mathbb{X}_f$, i.e., if $x_k \in \mathbb{X}_f \implies x_{k+1} \in \mathbb{X}_f$, which means that $\mathbb{X}_f$ is, in fact, an invariant set. The aforementioned is used in Theorem 3 to derive the conditions for the stability of the multi-step NMPC.*

3.6.3 Multi-step NMPC Stability Analysis

Given the stabilizing local controller defined in (3.35), is possible to establish sufficient conditions to ensure the stability of the multi-step NMPC closed loop in (3.33).

Theorem 3. *Consider the following constants; ζ as per (3.40) and $\sigma = \lambda_{\min}({P_\mu}^{-1/2} Q {P_\mu}^{-1/2})$. If*

$$\sigma + (\mu - 1)\zeta \geq b_1, \tag{3.54}$$

where $b_1 = 2\gamma\|A_K\|_{P_\mu} + \gamma^2$, and $\hat{x}_N \in \mathbb{X}_f$, whereas P_μ in V_f satisfies (3.41); then, the NMPC loop presented in (3.33) is stable.

Proof. To prove Theorem 3, is imperative to verify the conditions presented in (3.2). According to Theorem 2, the vector norm $\|\cdot\|_{P_\mu}$ is considered. Clearly, condition (3.2a) holds if $\alpha_1(s) = s^2$ for all $x \in \mathbb{X}$. Using $\kappa_f(x)$ defined in (3.35), the following suboptimal feasible input sequence is proposed:

$$\tilde{u} \;=\; \begin{bmatrix} \kappa_f(x)' & \kappa_f(\hat{x}_1)' & \cdots & \kappa_f(\hat{x}_{N-1})' \end{bmatrix}'.$$

Given the insights from Theorem 2:

$$\|\hat{x}_{N-1}\|^2_Q + \|\kappa_f(\hat{x}_{N-1})\|^2_R + \|\hat{x}_N\|^2_{P_\mu} = \|\hat{x}_{N-1}\|^2_{Q^*} + \|\hat{x}_N\|^2_{P_\mu}. \tag{3.55}$$

Considering (3.38), the right hand side of (3.55) can be rewritten as:

$$\|\hat{x}_{N-1}\|^2_{Q^*} + \|\hat{x}_N\|^2_{P_\mu} = \|\hat{x}_{N-1}\|^2_{Q^*} + \|\eta(\hat{x}_{N-1})\|^2_{P_\mu} + \|A_K\hat{x}_{N-1}\|^2_{P_\mu} + 2\eta(\hat{x}_{N-1})' P_\mu A_K \hat{x}_{N-1}. \tag{3.56}$$

Similarly to (3.47), is possible to establish the following inequality:

$$\|\hat{x}_{N-1}\|^2_{Q^*} \leq \xi \|\hat{x}_{N-1}\|^2_{P_\mu}, \tag{3.57}$$

where $\xi = \lambda_{\max}(P_\mu^{-1/2} Q^* P_\mu^{-1/2})$. Additionally, by (3.45) and (3.46):

$$\|\hat{x}_{N-1}\|^2_{Q^*} + \|\hat{x}_N\|^2_{P_\mu} \leq (\xi + \rho)\|\hat{x}_{N-1}\|^2_{P_\mu}, \tag{3.58}$$

where $\rho = \gamma^2 + 2\gamma\|A_K\|_{P_\mu} + \|A_K\|^2_{P_\mu}$. Starting the multi-step cost function (3.55) from $\hat{x}_{N-2}$ and considering (3.58), the following inequality holds:

$$\|\hat{x}_{N-2}\|^2_{Q^*} + \|\hat{x}_{N-1}\|^2_{Q^*} + \|\hat{x}_N\|^2_{P_\mu} \leq \|\hat{x}_{N-2}\|^2_{Q^*} + (\xi + \rho)\|\hat{x}_{N-1}\|^2_{P_\mu}. \tag{3.59}$$

Note that, by considering the nominal system in (3.38), $\hat{x}_{N-1}$ can be expressed in terms of $\hat{x}_{N-2}$, leading to:

$$\|\hat{x}_{N-1}\|^2_{P_\mu} = \|A_K\hat{x}_{N-2}\|^2_{P_\mu} + \|\eta(\hat{x}_{N-2})\|^2_{P_\mu} + 2\eta(\hat{x}_{N-2})' P_\mu A_K \hat{x}_{N-2}. \tag{3.60}$$

Considering (3.45) and (3.46):

$$\|\hat{x}_{N-1}\|^2_{P_\mu} \leq \rho\|\hat{x}_{N-2}\|^2_{P_\mu}. \tag{3.61}$$

Furthermore, considering (3.57), inequality (3.59) can be rewritten as:

$$\|\hat{x}_{N-2}\|^2_{Q^*} + \|\hat{x}_{N-1}\|^2_{Q^*} + \|\hat{x}_N\|^2_{P_\mu} \leq (\xi + \xi\rho + \rho^2)\|\hat{x}_{N-2}\|^2_{P_\mu}. \tag{3.62}$$

By iterating this procedure, and using optimality:

$$V_N^{op}(x, \vec{u}^{op}(x)) \leq V_N(x, \tilde{u}(x)) \leq \left(\sum_{j=0}^{N-2} \xi\rho^j + \rho^{N-1}\right)\|x\|^2_{P_\mu}. \tag{3.63}$$

Therefore, (3.2b) holds for $\alpha_2(s) = c_2 s^2$, where

$$c_2 = \left(\sum_{j=0}^{N-2} \xi\rho^j + \rho^{N-1}\right),$$

for all $x \in \mathbb{X}_f$. Considering the optimal input sequence, presented in (3.29), the optimal value for the cost function in (3.25) is given by:

$$V_N^{op}(x) = V_N(x, \vec{u}^{op}(x)). \tag{3.64}$$

Considering the stabilizing local controller presented in (3.35), a feasible input sequence $\tilde{u}$ is defined as:

$$\tilde{u} = \begin{bmatrix} (\hat{u}_1^{op})' & \cdots & (\hat{u}_{N-1}^{op})' & \kappa_f(\hat{x}_N)' \end{bmatrix}'. \tag{3.65}$$

Note that $\tilde{u}$ is obtained by shifting $\vec{u}^{op}(x)$ and appending $\kappa_f(\hat{x}_N)$. By Constraint (3.27e), $\hat{x}_N \in \mathbb{X}_f$. Moreover, although the input sequence may not be optimal, it is feasible. By considering both optimality and the aforementioned sequence $\tilde{u}$:

$$V_N^{op}(x_{k+1}) \leq V_N(x_{k+1}, \tilde{u}), \tag{3.66}$$

is possible to establish the following inequality by comparing expressions (3.64) and (3.66):

$$\begin{aligned} \Delta V_N(x_k) = V_N^{op}(x_{k+1}) - V_N^{op}(x_k) \leq & V_N(x_{k+1}, \tilde{u}) - V_N^{op}(x_k) \\ = & -L(x_k, \kappa_N(x_k)) - V_f(\hat{x}_N) \\ & + L(\hat{x}_N, \kappa_f(\hat{x}_N)) + V_f(\hat{x}_{N+1}). \end{aligned} \tag{3.67}$$

Additionally, similarly as (3.47)

$$L(x_k, \kappa_N(x_k)) \geq \lambda_{\min}({P_\mu}^{-1/2} Q {P_\mu}^{-1/2}) \|x_k\|^2_{P_\mu}. \tag{3.68}$$

Therefore, from (3.49), it follows that:

$$\Delta V_N(x_k) \leq -(\sigma + (\mu - 1)\zeta - b_1)\|x_k\|^2_{P_\mu}. \tag{3.69}$$

for all x in X_N. By condition (3.54), (3.2c) holds with $\alpha_3(s) = -\ c_3 s^2$, where

$$c_3 = \sigma + (\mu - 1)\zeta - b_1.$$

With the previous analysis, and considering that for an instant $t > 0, x \in \mathbb{X}_f$, it leads to

$$V_N^{opt}(x_{k+1}) \leq V_N^{opt}(x_k) - c_3\|x_k\|^2_{P_\mu}. \tag{3.70}$$

Considering (3.66) and establish the following relationship:

$$V_N^{opt}(x_{k+1}) \leq \delta_n V_N^{opt}(x_k), \tag{3.71}$$

with $\delta_n = 1 - c_3/c_2 \implies \delta_n \in [0, 1)$. By iterating (3.72), the cost function is bounded by:

$$V_N^{opt}(x_k) \leq \delta_n^k V_N^{opt}(x_t). \tag{3.72}$$

This implies that when $k \to \infty \implies V_N^{opt} \to 0, \forall k > t, x \in \mathbb{X}_f$. Furthermore, by the first two conditions in (3.2) and iterating (3.72), is possible to establish that

$$\|x_k\|^2 \leq \frac{c_2}{c_3}\delta_n^k \|x_t\|^2, \tag{3.73}$$

showing that $\lim_{k\to\infty} \|x_k\| = 0$ with $x_t \in \mathbb{X}_f$.

With the three stability conditions in Definition (3.2) verified, the closed-loop for the fixed-wing UAV governed by NMPC presented in (3.33) is ensured to be stable.

□

In essence, Theorem 3 guarantees stability by considering that, from an initial state $x_0 \in X_N$, the system will be steered by the NMPC towards the terminal region $\mathbb{X}_f$. Furthermore, at least the last predicted state must belong to the terminal region, that is $\hat{x}(N) \in \mathbb{X}_f$.

Remark 5. *Note that the linear case is a particular case of the proposal, where $\mu = 1$, and $\eta(\cdot) = 0 \implies \gamma = 0$. From* (3.24)*, it is clear that if the system is purely linear, then $\eta(\cdot) = 0$. Consequently, considering* (3.44)*, $\gamma = 0$. Moreover, from Theorem 2, if $\mu = 1$, then $Q^*_\mu = Q^*$ and $P_\mu = P$. After the system state is steered by the NMPC to the terminal region, the controller* (3.35) *guarantees the stability of the linear system $x_{k+1} = A_K x_k$.*

Remark 6. *Although the reasoning presented in Subsection 3.6.3 does not guarantee optimality, it is important to highlight that by imposing the last element of $\vec{x}_{[1:N]}$ in the terminal region (i.e., $\hat{x}_N \in \mathbb{X}_f$), the convergence of controller* (3.32) *is ensured.*

Table 3.1: System Parameters.

Parameter	**Symbol**	**Value**
Reference distance (m)	d_{ss}	150
Reference bearing angle $(rads)$	θ_{ss}	$\pi/2$
Reference altitude (m)	h_{ss}	50
Reference pitch angle $(rads)$	χ_{ss}	0
UAV reference speed (m/s)	$\mathcal{V}_{ss}$	$\{8, 9, 10\}$
Steady-state yaw input $(rads)$	$u_{\psi,ss}$	$-\mathcal{V}_{ss}/d_{ss}$
Steady-state pitch input $(rads)$	$u_{\chi,ss}$	0
Steady-state speed input $(rads)$	$u_{\mathcal{V},ss}$	0
Scaling factor	μ	1.1
Sampling time (s)	T_s	1
Distance w.f. (range)	q_d	$[0, 1.2^{-2}]$
Bearing angle w.f. (range)	q_θ	$[0, 8]$
Altitude w.f. (range)	q_z	$[0, 1.6^{-2}]$
Pitch angle w.f (fixed)	q_χ	1
Speed w.f. (fixed)	$q_\mathcal{V}$	0.1
Inputs weighting factor matrix	R	$\text{diag}(1, 1, 1)$
States w.f. matrix	Q	$\text{diag}(q_d, q_\theta, q_z, q_\chi, q_\mathcal{V})$

3.7 Stability Domain for Weighting Factors

The stability analysis provided in Section 3.6 shows that ensuring the stability of the proposal highly depends on the system parameters, scaling factor μ and weighting factors, which affect the bounds of the nonlinearities. Hence, weighting factors tuning for the NMPC loop, such that the stability conditions are held, is not an intuitive task. To address this, a weighting factors variation that denotes the values that hold the stability condition presented in (3.54) is provided. For this purpose, the system parameters listed in Table 3.1 are considered.

Furthermore, considering the system in (3.21), it is evident that the states that induce nonlinearities in the system model are d, θ and h. This helps us to determine that changes in q_d, q_θ and q_z are considered, while keeping constant q_χ and $q_\mathcal{V}$. This results in a domain of points where the stability of the NMPC loop for the ground target tracking system is ensured.

From experimental testing, it was found that Bixler 3, in this configuration, stalls at a speed of approximately 25 km/h, or at approximately 7 m/s. Considering this, and to avoid pushing limits on the trust of the aircraft, three different stability domains for three different reference speeds, $\mathcal{V}_{ss} = \{8, 9, 10\}$ are presented. These speeds were achieved

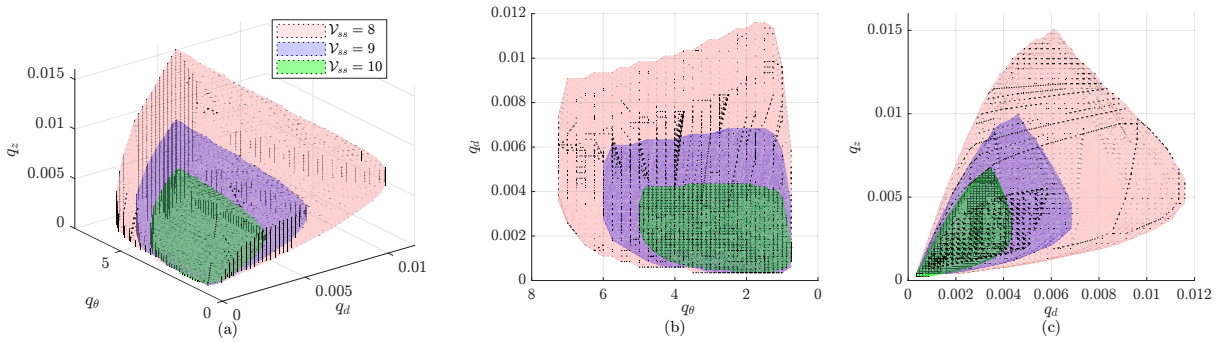


Figure 3.2: Weighting factors variation for different values of $\mathcal{V}_{ss}$. (a) Shows a 3D view, with q_d, q_θ and q_z as axis. (b) 2D q_θ vs. q_d. (c) 2D q_d vs. q_z.

Table 3.2: Stability condition for $\mathcal{V}_{ss} = [8, 9, 10]$, with different weighting factors.

1st	$q_\theta = 4$	$q_d = 0.005$	$q_z = 0.006$
$\mathcal{V}_{ss}$	$\sigma + (\mu - 1)\zeta$	$2\gamma\|A_K\|_{P_\mu} + \gamma^2$	Holds condition?
8	0.1784	0.1689	True
9	0.1816	0.1727	True
10	0.1793	0.1927	False
2nd	$q_\theta = 4$	$q_d = 0.0037$	$q_z = 0.006$
$\mathcal{V}_{ss}$	$\sigma + (\mu - 1)\zeta$	$2\gamma\|A_K\|_{P_\mu} + \gamma^2$	Holds condition?
8	0.1639	0.1401	True
9	0.1753	0.1585	True
10	0.1776	0.1771	True

using approximately 55% of the throttle command and represent the most likely scenario for the experimental velocity. Only changes in the reference speed are considered, because changes in this state affect the stability domain shape the most.

The results are presented in Fig. 3.2, where three distinct regions are shown. A comprehensive view of the 3D environment is shown in Fig. 3.2a. Upon initial inspection, a clear correlation emerges. As the reference speed $\mathcal{V}_{ss}$ decreases, the stability constraints imposed by the weighting factors become less stringent. Furthermore, when the same system parameters are considered, the different stability domains exhibit nested structure. This implies that a set of weighting factors that ensures the stability condition for $\mathcal{V}_{ss} = 10$ guarantees stability at lower values of $\mathcal{V}_{ss}$. Nevertheless, this is not necessarily the case vice-versa. To provide additional insights, Figs. 3.2b and 3.2c are included to offer diverse perspectives. Following condition (3.54), and choosing $q_\theta = 4$, $q_d = 0.005$ and $q_z = 0.006$, the first section of Table 3.2 is obtained, showing that for this particular weighting factors, the stability condition holds for $\mathcal{V}_{ss}$=[8, 9], but not for $\mathcal{V}_{ss}$=10.

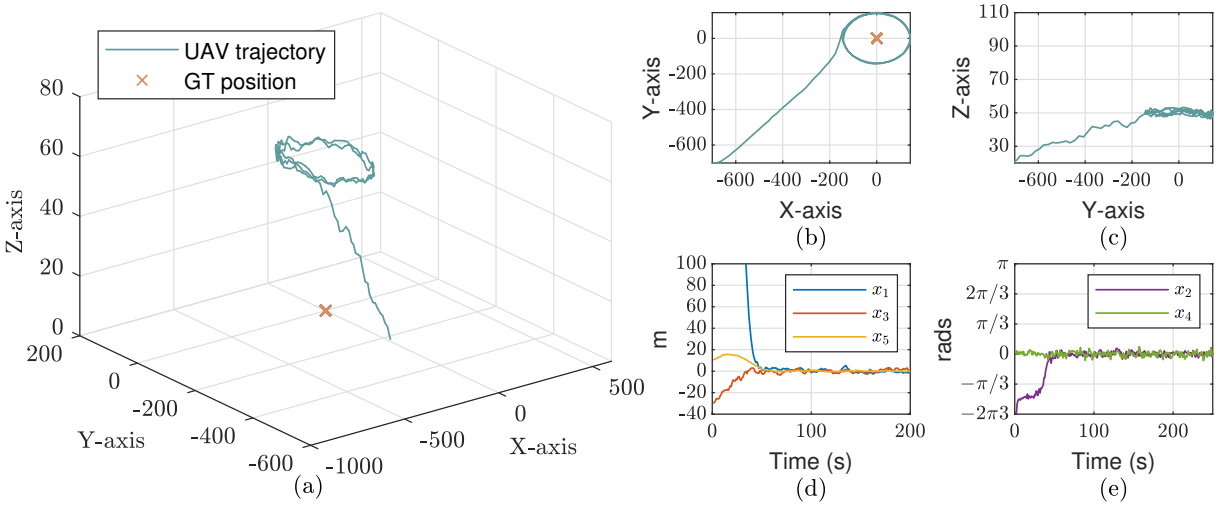


Figure 3.3: Simulation results for a static target scenario. (a) UAV trajectory and ground target position. (b) Top-view of UAV trajectory. (c) Side-view of UAV trajectory. (d) Tracking errors for distance, altitude and speed. (e) Tracking errors for bearing angle and pitch angle.

Choosing q_d=3.7·10^{-3} and maintain the remaining parameters, then the stability condition holds for $\mathcal{V}_{ss} = [8, 9, 10]$, as shown in the second section of Table 3.2.

Remark 7. *In contrast to the case of speed $\mathcal{V}_{ss}$, an increase in the reference values of distance d_{ss} and altitude h_{ss} implies an expansion in the stability domain. Therefore, the stability of the proposed method is not compromised by using greater reference values for d_{ss} and h_{ss}.*

Remark 8. *Because of the fact mentioned in Remark 7, the proposed 3D Dubins model is widely applicable to different fixed-wing UAVs. Because its application hinges mostly on $\mathcal{V}_{ss}$, the stall speed of the UAV should be considered. Then, the stability condition can be tested for different values of $\mathcal{V}_{ss}$ greater than the stall speed, combined with the reference values d_{ss} and h_{ss} and the stability domain of the weighting factors is determined.*

3.8 Results

Simulations were performed to verify the effectiveness of the proposed approach. The parameters are the same as those used in the weighting factor sweep, that is, those presented in Table 3.1, with the exception of $\mathcal{V}_{ss} = 10$. It is worth noting that, to account for an equitable comparison with experimental tests, simulations are subjected to identical parameters as the experiments. The target trajectory is created with real data using a manually controlled vehicle. This manual operation pattern ensures the randomness of the trajectory of the target, thus simulating real-world unpredictability in the velocity and direction.

Table 3.3: Resulting Control Parameters Design.

Parameter	**Symbol**	**Value**
Scaling factor	μ	1.1
Terminal region	$\mathbb{X}_f$	4.6065
Distance weighting factor	q_d	$3.7 \cdot 10^{-3}$
Bearing angle weighting factor	q_θ	4
Altitude weighting factor	q_z	0.006
Pitch angle weighting factor	q_χ	1
Speed weighting factor	$q_\mathcal{V}$	0.1
ζ (from Theorem 2)	ζ	0.2026
σ (from Theorem 3)	σ	0.1574
Lipschitz constant	γ	0.0649

The controller does not know the target direction and acceleration. Nonetheless, its position can be measured at each sampling interval. It is worth noting that despite the target moving at variable speed, the UAV speed is maintained consistently greater than the target speed to ensure effective tracking of the target throughout the simulation.

With the idea of minimizing the deviation for the references, a combination of weighting factors that penalize those deviations is chosen, that is, choosing them in an attempt to impose the maximum values while ensuring stability. Setting the weighting factors to $q_\theta = 4$, $q_d = 3.7 \cdot 10^{-3}$, and $q_z = 0.006$, and considering the previous parameters, leads to

$$A_K = \begin{bmatrix} 1 & 10 & 0 & 0 & 0 \\ -0.0218 & -0.0896 & 0 & 0 & 0.0002 \\ 0 & 0 & 1 & 10 & 0 \\ 0 & 0 & -0.0352 & -0.1456 & 0 \\ 0 & -0.0002 & 0 & 0 & 0.7298 \end{bmatrix},$$

$$P_\mu = \begin{bmatrix} 0.0203 & 0.1867 & 0 & 0 & 0.0001 \\ 0.1867 & 7.4657 & 0 & 0 & 0.0082 \\ 0 & 0 & 0.0215 & 0.1877 & 0 \\ 0 & 0 & 0.0648 & 4.2369 & 0 \\ 0.0001 & 0.0082 & 0 & 0 & 0.4073 \end{bmatrix}.$$

With A_K and P_μ, the terminal region $\mathbb{X}_f$ is calculated as per (3.37), resulting in:

$$\mathbb{X}_f \triangleq \{x' P_\mu x \leq \varphi_x = 4.6065\}. \tag{3.74}$$

As depicted in Section 3.6, the value of φ_x represents the ellipsoid-shaped terminal region, where it is ensured that satisfies $x \in \mathbb{X}_f \subset \mathbb{X}$ and $Kx \in \mathbb{U}$.

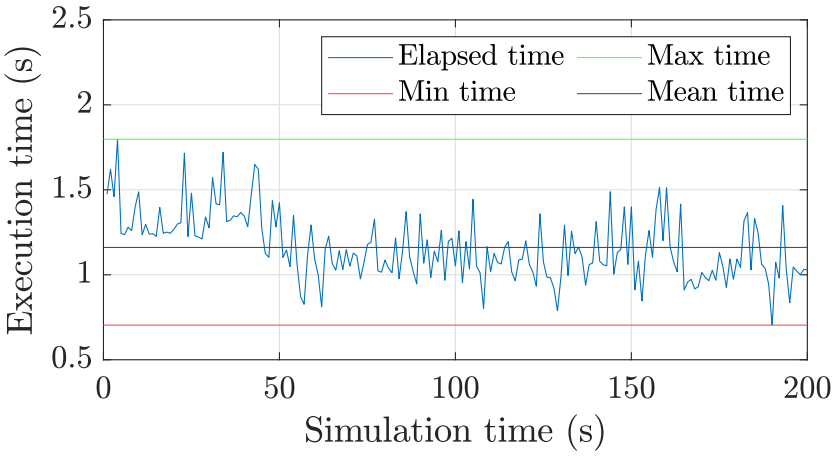


Figure 3.4: Execution time for simulation of stationary target.

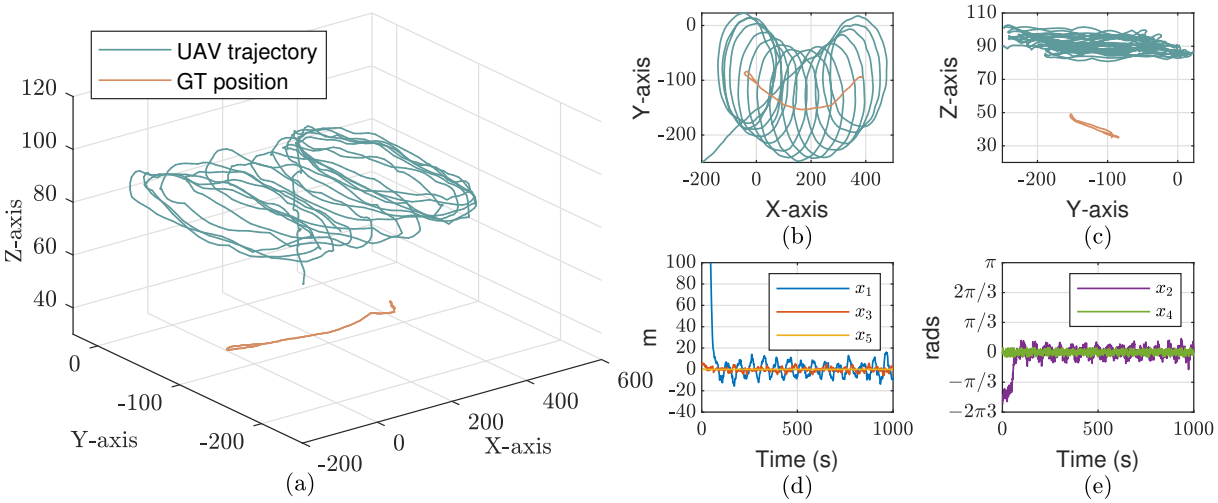


Figure 3.5: Simulation results for a moving target scenario. (a) UAV and ground target trajectories. (b) Top-view of trajectories. (c) Side-view of trajectories. (d) Tracking errors for distance, altitude, and speed. (e) Tracking errors for bearing and pitch angle.

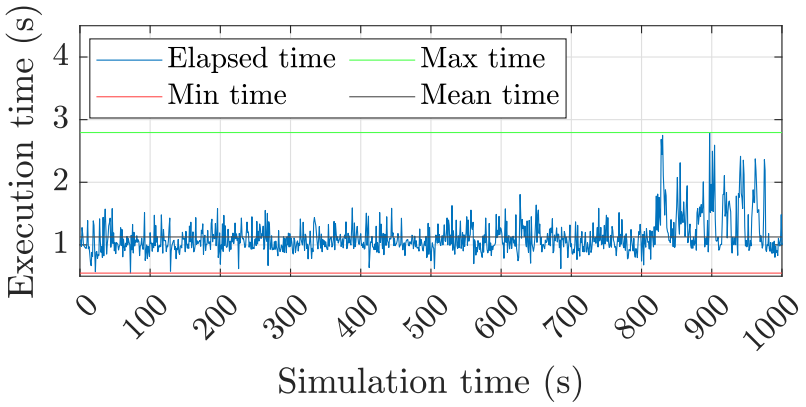


Figure 3.6: Execution time for simulation of moving target.

Considering the parameters presented in Table 3.1 and Theorem 2, the resulting parameter designs are presented in Table 3.3. Notably, the nonlinearities of the UAV are bounded according to (3.39), with the Lipschitz constant $\gamma = 0.0649$. To guarantee the stability of the closed loop, condition (3.54) in Theorem 3 is verified, where

$$\sigma + (\mu - 1)\zeta = 0.1776 > 2\gamma\|A_K\|_{P_\mu} + \gamma^2 = 0.1771,$$

holding the condition and ensuring the stability of the proposal.

Remark 9. *Note that, for this example, $\mu = 1.1$, which satisfies the stability conditions* (3.42) *and* (3.54). *However, using a value of $\mu = 1$ (assuming a linear design), these conditions are not met; thus, stability is not guaranteed.*

Remark 10. *The algorithm used to solve the optimization problem in* (3.27a) *is the Bat Algorithm. It is noteworthy that since this is a heuristic method, obtaining a global optimum cannot be guaranteed. Therefore, to ensure that the closed-loop stability is not compromised, the algorithm is initialized with a stabilizing initial solution as per* (3.65).

Remark 11. *The Bat Algorithm used in this study is configured with a population size of 800 bats. The convergence criterion is based on a fixed number of iterations, which is 30 for this application. According to Remark 10, if no better solution is found than the initialized one, then the shifted optimal solution is considered.*

3.8.1 Simulations results

Simulations are conducted to evaluate the performance of the proposed approach.

Stationary Target

The first case study comprises a stationary target located at the origin of the coordinate frame, whereas the UAV is located at an initial position of $(-700, -700, 20)$. The results of this scenario are shown in Fig. 3.3. Figure 3.3a illustrates the trajectory of the UAV and the fixed ground target position. Additionally, Fig. 3.3b and 3.3c provide different 2D views of the UAV trajectory, emphasizing distance and altitude tracking, respectively.

To address the tracking errors in the states, Fig 3.3d and 3.3e are provided. Figure 3.3d focuses on distance, altitude, and speed errors. The results show an initial significant error for all states that converge to zero at 50 s, persisting for the remainder of the simulation. With respect to the pitch and bearing angle errors, depicted in Fig. 3.3e, a similar convergence pattern is observed. The tracking error of these angles also converged to zero after 50 s, remaining close to their reference values throughout the simulation. This validates that, as stipulated in Theorem 3, the NMPC converges tracking errors to zero.

Table 3.4: Key performance indicators for different $\mathcal{V}_t$.

$\mathcal{V}_t$	**Peak error (m)**	**RMSE (m)**	**Average loitering period (s)**	**Bounds [min, max] (m)**
1	7.65	3.9	61	$[-7.95, 7.65]$
2	16.25	8.03	66	$[-15.1, 16.25]$
3	18.63	11.14	68.7	$[-19.06, 18.63]$
4	27.91	15.52	72	$[-24.61, 27.91]$
5	32.37	19.01	78.7	$[-31.55, 32.37]$
6	46.46	25.35	86	$[-37.54, 46.46]$
7	57.56	31.57	106	$[-43.71, 57.56]$
8	65.57	36.12	144	$[-52.16, 65.57]$
9	76.46	44.54	271	$[-61.2, 76.46]$

Moreover, the execution time for the simulation is presented in Fig. 3.4. Simulations were performed in MATLAB on a computer equipped with an Intel Core(TM) i7-1185G7 CPU running at 3.00 GHz and 16 GB of RAM. The results show that, even with this decent simulation hardware, the algorithm's execution time has a mean of 1.2 seconds, which exceeds the stipulated sampling time T_s. Consequently, the controller cannot be implemented in real-time under these conditions. This is due to the configuration of the bat algorithm, which is running with 800 particles and 40 movements. While this high number of particles improves the solution's quality, it significantly increases the computational burden. This issue is addressed in Section 3.8.3 using parallel programming to overcome the computational challenges, although the current configuration is retained here for comparison purposes.

Moving Target

Results for the moving target with unknown trajectory are presented in Fig. 3.5. This simulation considers the same target movement than the experiments. The target's speed ranges from 0 up to 5.22 meters per second. Furthermore, unmodeled wind with a speed of 2.3 meters per second is introduce in the simulations. This value matches the average wind measured during the experiments while tests were performed. This is to evaluate the accuracy of the proposed mathematical model. Similarly as previous case, Fig. 3.5a depicts the 3-dimensional trajectories for the UAV and the ground target, Fig. 3.5b and Fig. 3.5c present a 2-dimensional top view, and a 2-dimensional side view of the aforementioned trajectories, respectively. These figures present both distance and altitude tracking, indicating a good performance for a moving target. Notably, the clockwise circumnavigation direction of the UAV around target is evident, demonstrating the ability of the UAV to accurately track the moving target. Additionally, from Fig. 3.5c it seems

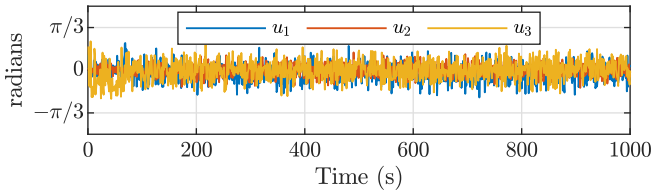


Figure 3.7: Control effort

that the altitude tracking does not performs as expect. Notwithstanding this observation, the high variation on the altitude is due to the fact that high is measured with respect to the target. Considering that the target varies its altitude from 35 to 50 m, it is expected that so it does the UAV altitude.

Moreover, Fig. 3.5d and Fig. 3.5e show the tracking errors for all the states throughout the simulation. A closer look at the tracking error of the altitude x_3 in Fig.3.5d, confirms that the error is bounded, with an absolute peak error of 3.5 m. Moreover, from Fig. 3.5d, it is evident that around the first 80 s, the states are moving towards their respective references, showing a fast convergence towards zero error for the distance x_1, altitude x_3, and speed x_5. Considering the moving nature of the target, x_1 never reaches zero error. However, it remains confined, oscillating around zero in steady-state and presenting a maximum deviation from the reference point of 15 m. Furthermore, after the 80 s mark, the root mean squared error (RMSE) for the distance tracking is 6.4 m. A closer look to the altitude tracking shows that the maximum error is 4 m. Notably, after the 80 s mark, the RMSE for the altitude tracking is 1.98 m. From Fig. 3.5e, is possible to draw similarities between x_1 and the bearing angle tracking error x_2. Again, owing to the movement of the target, the error oscillates around zero on steady-state, presenting an error that does not surpass $\pi/8$ radians. In the case of the pitch angle error x_4, after reaching steady-state, it remains close to zero tracking error.

It is evident that the proposed method achieves highly accurate tracking, where the distance error for a moving target remains below 15 m, whereas the bearing angle, altitude, pitch angle and speed errors are negligible. For all states, there is no observable overshoot.

The effort exerted by the controller during the simulation is shown in Fig.3.7. It is clear that the three input tracking errors u_1, u_2, u_3 remain between the bounds imposed by the input constraint $\mathbb{U}$ and around their respective steady-state values $u_{\psi,ss}$, $u_{\chi,ss}$, $u_{\mathcal{V},ss}$, respectively. In addition, the steady-state tracking error is negligible, ensuring the feasibility of the proposed system in a experimental setup.

Drawing parallels from the stationary target simulation, the execution time for this scenario is also presented. The algorithm configuration remains the same, running with 800 particles and 40 movements. In this case, the maximum execution time observed is 1.9 seconds, while the minimum is 0.64 seconds, with a mean execution time of 1.1 seconds.

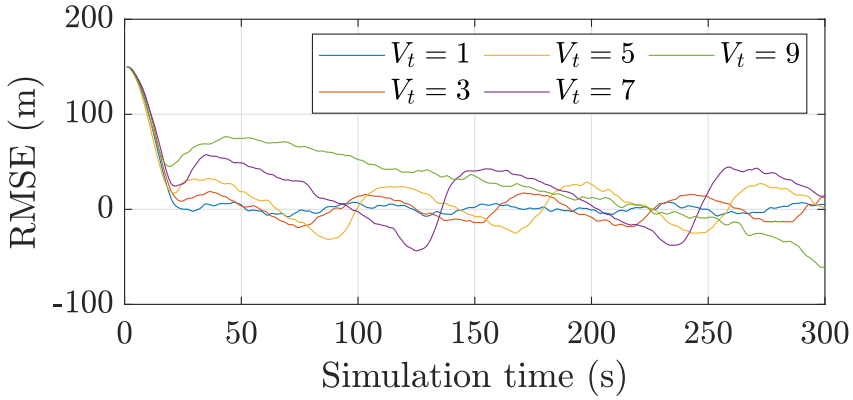


Figure 3.8: Root mean squared error for different values of target speed.

When compared to the stationary target scenario, it is evident that the execution time is not significantly affected by the complexity of the scenario. Instead, it primarily depends on the algorithm's configuration, which explains the similar results obtained in both cases. Again, the execution time average exceeds the sampling time, showing once again that the parallelization of the algorithm is needed to achieve real-time implementation.

Effect of the ground target speed over the model

The system model presented in (3.17) does not account for the ground target speed $\mathcal{V}_t$ in its prediction. Thus, it becomes evident that $\mathcal{V}_t$ has a significant impact on the controller's tracking accuracy. This can be quantified by simulating a ground target moving in a straight line at varying speeds. In this test, the UAV travels at 10 m/s, while the ground target is configured to move at constant speed, ranging from 1 to 9 m/s for each case. The results, included in Table 3.4, present the peak absolute error and the RMSE for each scenario. Furthermore, the instant tracking error is graphically depicted in Fig. 3.8 throughout simulation. It is important to highlight that, while simulations have been conducted for 9 different values of $\mathcal{V}_t$, Fig. 3.8 only shows odd speeds. This is to improve clarity for the reader. Notably, as $\mathcal{V}_t$ increases, the tracking error also increases. Furthermore, the loitering period time is also increased, as the peak value for the error.

The increase in error can be attributed to the fact that, since the ground target speed is not included in the model, prediction errors propagate through the prediction horizon, thereby reducing its performance. Nevertheless, the error presented is bounded for all cases, as presented in the last column of Table 3.4. That is, given that $\mathcal{V}_t \leq 0.9 \cdot \mathcal{V}$, then, it is possible to track the target with an acceptable steady-state error.

Sensitivity Analysis for T_s and N

Considering that the NMPC loop can be implemented with various combinations of sampling time T_s and prediction horizon N, a closed-loop sensitivity analysis against these

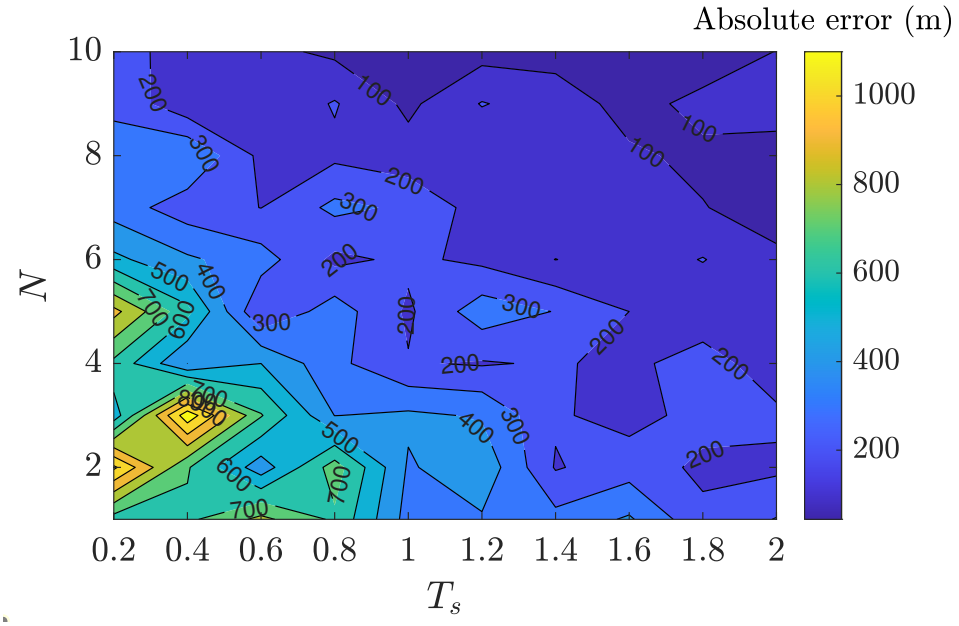


Figure 3.9: Heat map for variation of T_s and N.

parameters is presented. In this analysis, a sweep is performed over both parameters, with T_s ranging from 0.2 to 2 seconds and N ranging from 1 to 10. The results are visualized as a heat map showing the accumulated absolute error for a stationary target loitering scenario, with the same parameters as described in Section 3.8.1. The heat map is presented in Fig. 3.9.

As expected, the results indicate that the performance of the NMPC loop varies depending on the configuration of T_s and N. Notably, higher errors occur with low values of both N and T_s, which can be explained by the controller's limited ability to predict far into the future under this configuration. As N increases, the error decreases, with the best results observed when the prediction horizon is larger. This improvement is evident until a sampling time of approximately 1.8 seconds, after which a slight increase in error can be observed, particularly for $T_s = 2$ seconds. In other words, performance improves as you increase the effective horizon $N \cdot T_s$, but only up to the point where the optimizer can converge to a solution within T_s. A more thorough comparison would keep the ratio of prediction horizon to system timescale, $\mathcal{V}_{ss}/T_s$, roughly constant while varying T_s, and acknowledge that further reductions in T_s requires faster hardware or more efficient solvers to avoid missed deadlines. Unfortunately, that analysis is beyond the scope of this thesis.

However, due to the heuristic nature of the Bat Algorithm used to solve the non-convex optimization problem, increasing the prediction horizon indefinitely is not feasible. This is because a larger prediction horizon N would require more bats in the algorithm, which would significantly increase the computational burden. The heat map also shows small isolated regions of high error, which can be attributed to the inherent uncertainty in

Table 3.5: Comparison results.

Case 1				
Controller	**RMSE (reaching)**	**RMSE (loitering)**	**Absolute peak error**	**Time to reference**
NMPC	169.73	1.8358	3.9476	55
MPC	–	–	–	–
SMC	178.87	3.6142	7.1391	95
Case 2				
Controller	**RMSE (reaching)**	**RMSE (loitering)**	**Absolute peak error**	**Time to reference**
NMPC	41.83	1.63	3.3813	20
MPC	56.12	1.61	3.1526	27
SMC	46.71	3.64	7.4961	70

heuristic algorithms that affect the solution found in each iteration. Despite this, a clear trend in the relationship between T_s and N is evident.

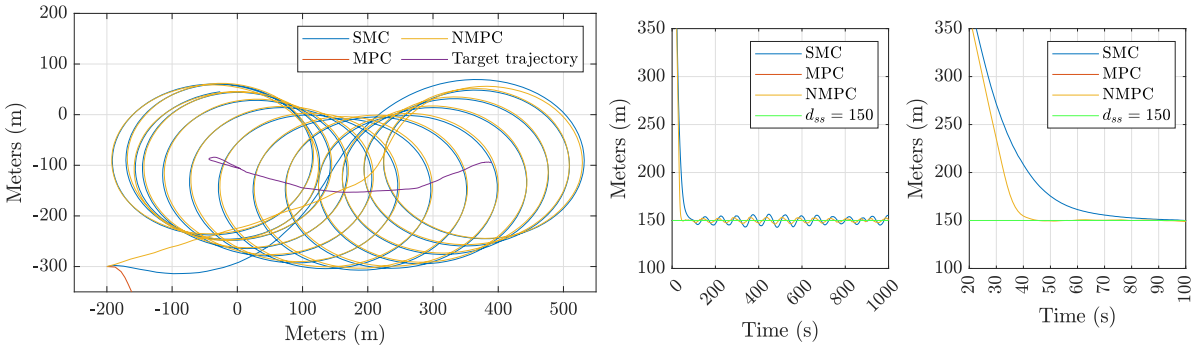
Lastly, the combination of $T_s = 1$ and $N = 10$ results in minimal accumulated absolute error, and this configuration is used for the stability analysis presented in Section 3.6. Furthermore, the experimental platform is pre-configured to communicate with the companion computer at a frequency of 1 Hz. It is for these reasons that this approach is set with $T_s = 1$ and $N = 10$ for the remainder of this work.

3.8.2 Comparison with existing control strategies

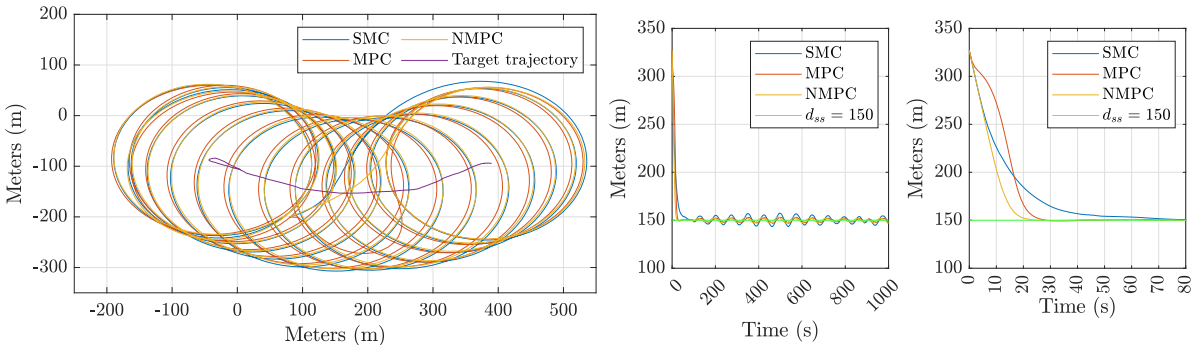
To further evaluate the proposal, a comparison against other existing methods for ground target tracking is presented. In this scenario, the simulation setup from Section 3.8.1 is used, considering the moving target case without wind. The controllers considered for comparison are Sliding Mode Control (SMC), designed as in [32] for single UAV tracking, and linear MPC, designed as in [35].

Results for this comparison are presented in Fig. 3.10 a-b. From left to right, both figures display the trajectories of the UAV governed by different controllers along with the target trajectory, the distance evolution throughout the simulation, and a detailed view of the distance for all cases during the first 100 and 80 seconds, respectively.

The three controllers use the same sampling time and initial conditions. In the case of MPC, the same weighting factors Q and R as for NMPC are used, as presented in the original simulation, and the same prediction horizon $N = 10$. Moreover, for readability, comparisons are focused on the 2D distance, where the largest discrepancies are noted. It



(a)



(b)

Figure 3.10: Comparison results for proposal against SMC and linear MPC. (a) Case 1. (b) Case 2.

is important to highlight that the performance of the controllers under the same simulation but with two cases considering different initial conditions is evaluated.

Results for these two cases are depicted in Table 3.5, where the performance of each controller is measured by RMSE when reaching the target and while loitering, absolute peak error, and time to reach the set-point $d_{ss} = 150$. Particularly, RMSE while reaching the target is computed from the beginning of the simulation up until all controllers reach d_{ss}. Furthermore, RMSE loitering and absolute peak error are calculated from the moment all controllers reach d_{ss} up until the end of the simulation.

Notably, for the first case, only SMC and NMPC were able to complete the test; MPC did not converge. This is because linear MPC relies on a linearized system model to control the UAV system. Since the first case of initial conditions involves the UAV located at a point distant from the linearization point, the model does not represent the real system accurately enough, causing the closed-loop system to diverge in this case, an issue widely covered in [35]. Moreover, during testing, NMPC exhibited faster convergence to the desired steady-state distance d_{ss}. This behavior is reflected also in the RMSE while reaching, which is lower than SMC. Furthermore, NMPC exhibits lower RMSE while loitering and reduced peak error compared to SMC. This enhanced performance is attributed to the inherently optimal nature of NMPC, which often leads to superior

results, as demonstrated in this case.
In the second case, the UAV system is located at a closer distance from the moving target, where linear MPC is able to steer the system to the desired set-points. This issue is not present when using NMPC or SMC since they rely on the nonlinear dynamics. Similar to the first case, is possible to see that NMPC shows faster convergence than both linear MPC and SMC. Consistent with previous observations, NMPC also shows lower RMSE while reaching than the other controllers. In this particular case, linear MPC shows practically identical RMSE while loitering and absolute peak error than NMPC, demonstrating high accuracy for target tracking, but lower time to reference, highlighting the superiority of NMPC. Despite this performance, the great weakness of this approach is its lack of versatility compared to NMPC when affronted to initial conditions far from the linearization point. NMPC greatest feature is achieving consistently faster convergence to the target than other approaches while maintaining acceptable tracking error, highlighting its superior overall performance.

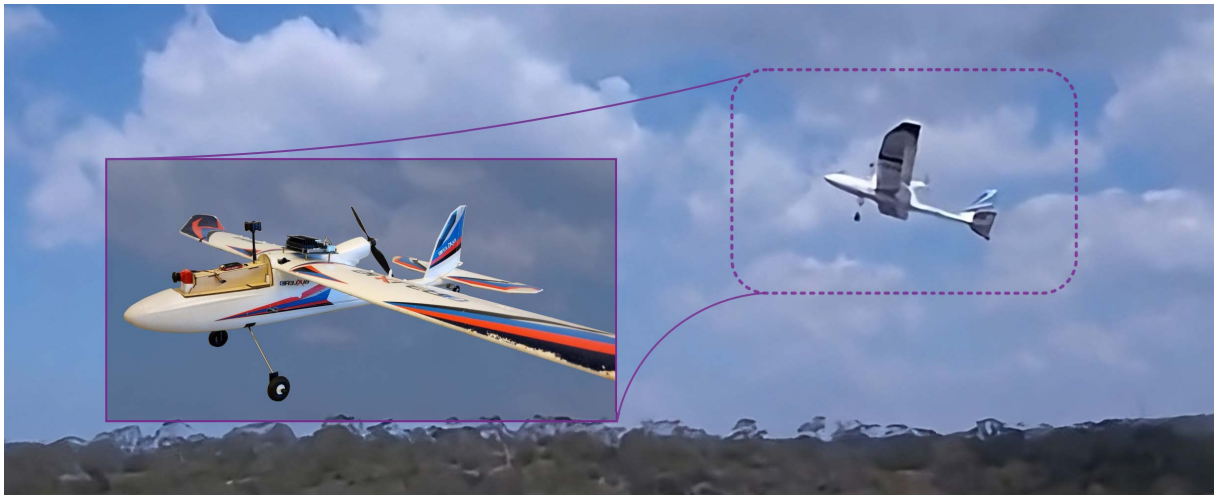
3.8.3 Experimental Results

To comprehensively assess the performance of the proposed method, experimental results are presented. The experimental results were obtained at the Warringah Radio Control Society, located in Belrose, NSW, Australia, with the club coordinates specified as $33°42'45.8"S\ 151°14'23.2"E$.

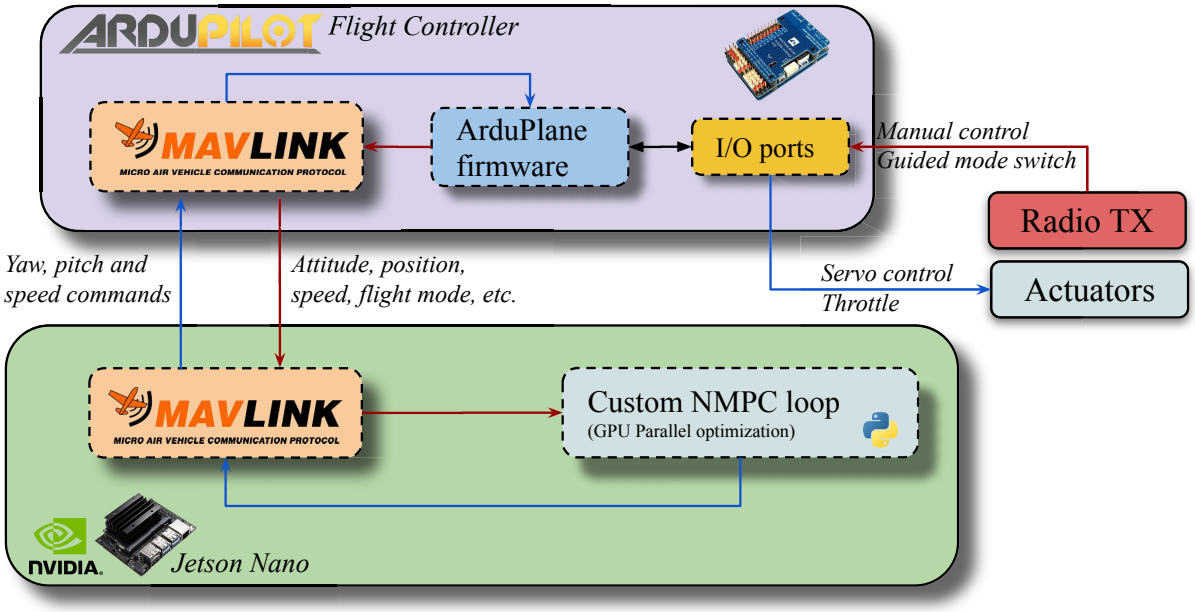
Fixed-wing UAV Experimental Platform

Following the guidelines presented in [99], the experimental setup comprises a Hobbyking Bixler 3 equipped with a Matek H743-Wing V3 flight controller, running ArduPlane, and a Global Navigation Satellite System module Matek M10Q-5883. Real-time data is stored in the UAV using a black-box system. These data are also transmitted to a ground station using a telemetry system, as well as a first person view (FPV) system with On-Screen Display (OSD) for redundancy.

The flight controller manages all aircraft aspects during the flight. A companion computer based on NVIDIA Jetson Nano is also on-board the aircraft. This companion computer is dedicated to solving the nonlinear optimization problem presented at (3.27a). Because the optimization problem is non-convex, is required to use advanced optimization algorithms to solve the optimal control problem at each sampling instant. The NVIDIA Jetson Nano is a microcomputer with significant computational capabilities, particularly its dedicated graphics processing unit (GPU), with 128-CUDA cores, allowing us to efficiently run highly demanding computational algorithms, such as heuristic algorithms, used in the literature to optimize non-convex problems. Considering the CUDA capabilities of Jetson Nano, a custom NMPC loop in Python is used that, using Numba, translates Numpy



(a)



(b)

Figure 3.11: Experimental UAV platform. (a) Hobbyking Bixler 3 on actual mission. (b) Hardware configuration.

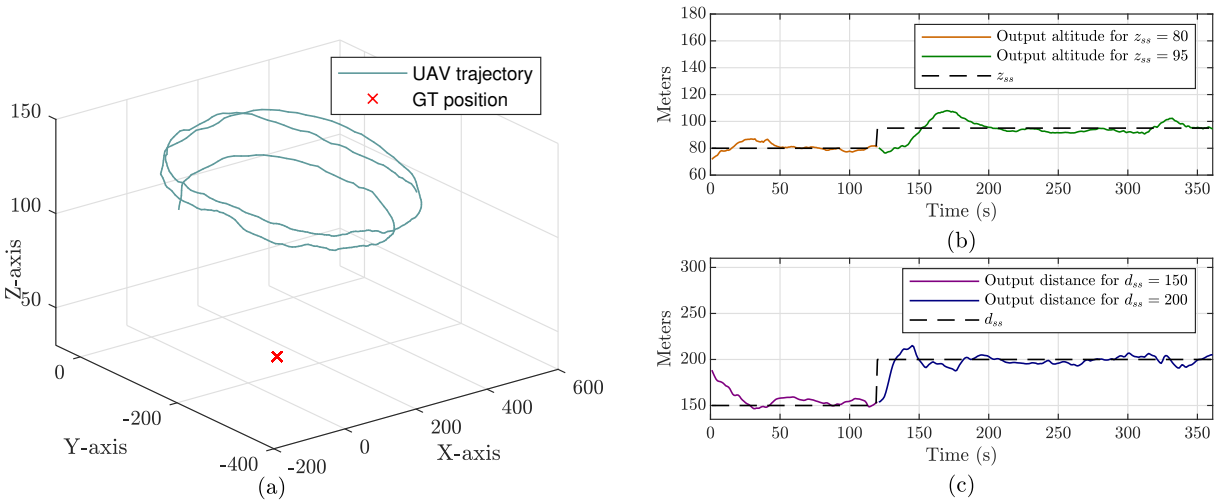


Figure 3.12: Experimental results. (a) UAV and ground target trajectories. (b) Top-view of trajectories. (c) Side-view of trajectories. (d) Tracking errors for distance, altitude and speed. (e) Tracking errors for bearing angle and pitch angle.

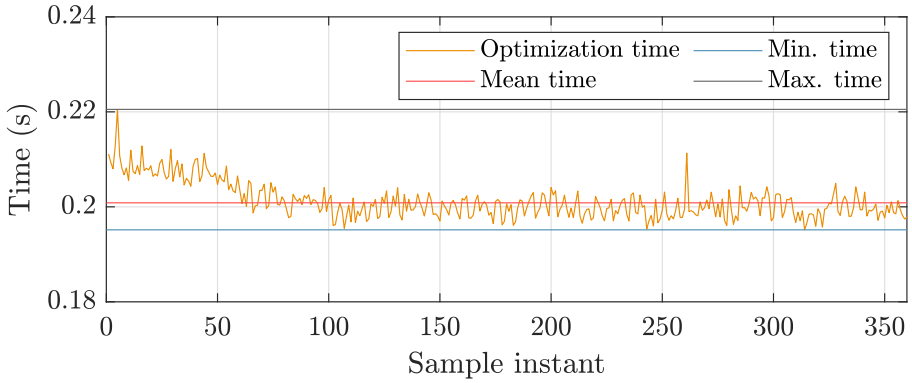


Figure 3.13: Optimization time (GPU based) for step test on experimental platform.

Algorithm 1 Jetson Nano Pseudocode

Require: $MavLinkConnection$ = True
Require: $UAVMode$ = Guided
while True **do**
    $UAVx \leftarrow$getUAVState()
    $Targetpos \leftarrow$getTargetPosition()
    $[u_\psi, u_\chi, u_\mathcal{V}] \leftarrow$ batAlgorithm$(UAVx, Targetpos)$
    commandToUAV$(u_\psi, u_\chi, u_\mathcal{V})$
end while
storeData()

instructions into CUDA code, drastically reducing the execution time of the algorithm. As stated in Remark 10, the solver used is the Bat Algorithm, which shows good performance when optimizing highly non-convex problems. It is important to emphasize that, although the flight controller is running ArduPlane, a highly customizable open-source firmware, the capabilities of the Jetson Nano to execute the non-convex optimization process are considered rather than running the NMPC loop directly on the flight controller. This decision is driven not only by the fact that the flight controller lacks the computational power required to handle demanding non-convex optimization algorithms, such as the Bat Algorithm, but also to maintain the flexibility of the experimental platform. Furthermore, the bat algorithm is well-suited for GPU parallelization using Numba due to its inherently parallel structure. Each bat's exploration can be computed independently, allowing simultaneous calculations that enhance computational efficiency. The aircraft and onboard component connections are illustrated in Fig. 3.11. Furthermore, the pseudo-code is presented at Algorithm 1. Additionally, the actual Python code can be found in GitHub.

Pre-experiments flight

Since the optimal problem presented in (3.27a) is constrained, it is essential to determine the limits of both the actuators and the states. For this particular experimental platform, it is needed to establish the bounds for the heading angle input u_ψ and the pitch angle input u_χ. Additionally, the stall speed $\mathcal{V}_{\min}$ and maximum speed $\mathcal{V}_{\max}$ of the UAV must be defined to ensure safe operation within these limits. For task-specific states, it is important to note that the actual distance d_k is always positive. Furthermore, the actual altitude h_k must remain above a minimum safe altitude, which is determined by the environment in which the experiments are conducted. For these experiments, the minimum safe altitude is set at $h_{\min} > 60$ meters. Lastly, the bearing angle is constrained within the range between$-\pi$ to π. These constraints ensure that the UAV operates safely and remains within its physical limitations throughout the mission. These values are pre-

Table 3.6: Bixler's optimization parameters

Parameter	**Symbol**	**Value**
Heading angle limits	u_ψ	$\pm\pi/4\ rads$
Pitch angle limits	u_χ	$\pm\pi/16$ rads
Stall speed	$\mathcal{V}_{\min}$	$7m/s$
Max speed	$\mathcal{V}_{\max}$	$22m/s$
Actual distance	d_k	$> 0m$
Actual altitude	h_k	$\geq 60m$
Actual bearing angle limits	θ_k	$\pm\pi\ rads$

sented in Table 3.6. Nevertheless, if more in-depth aerodynamic parameters are needed, a good estimation is provided at ecalc.ch. This website is a invaluable tool in fixed wing community, where parameter estimation can be done by user-provided aircraft data.

Step Response

The first test elucidates the response of the system to step input, providing insights into its dynamic behavior. The ground target is considered stationary at position $(165, -175, 30)$ in the 3D frame. The initial reference is introduced at the beginning of the mission; specifically, the initial setting is $d_{ss} = 150$ and $h_{ss} = 80$. The results are shown in Fig. 3.12. After setting the first set of references, is possible to see that the system responds with accurate tracking of the altitude and distance, maintaining small deviations. The deviations are more notorious for the distance. This is because of the action of non-modelled disturbances, such as wind. Nevertheless, the controller is able to remain this deviation negligible. At the 120 s mark, a second set of reference values is introduced for both altitude and distance. Specifically, the new references are $d_{ss} = 200$ and $h_{ss} = 95$. Subsequently, the altitude reference is reached at 150 s, while the distance settles at the 130 s mark. This difference is owing to the different weighting factors, where the penalization of the distance is higher than the altitude, resulting in a promptly correction of the distance, followed by the correction of the altitude. There exist overshoot in both states. However, these overshoots do not surpass 15 m in the case of the distance, and 8 m for the altitude. After the 200 s mark, it is observed that the system remains within minimal error margins. Notwithstanding the disturbances present, the controller demonstrated efficiency in accurately tracking the reference values. Additionally, the optimization time for each sampling instant during the experiment is presented. Figure 3.13 shows the results. It is evident that the optimization time remains practically unaltered, with a mean time of 0.2 seconds. Additionally, the maximum and minimum optimization time are 0.2205 and 0.1952 seconds, respectively.

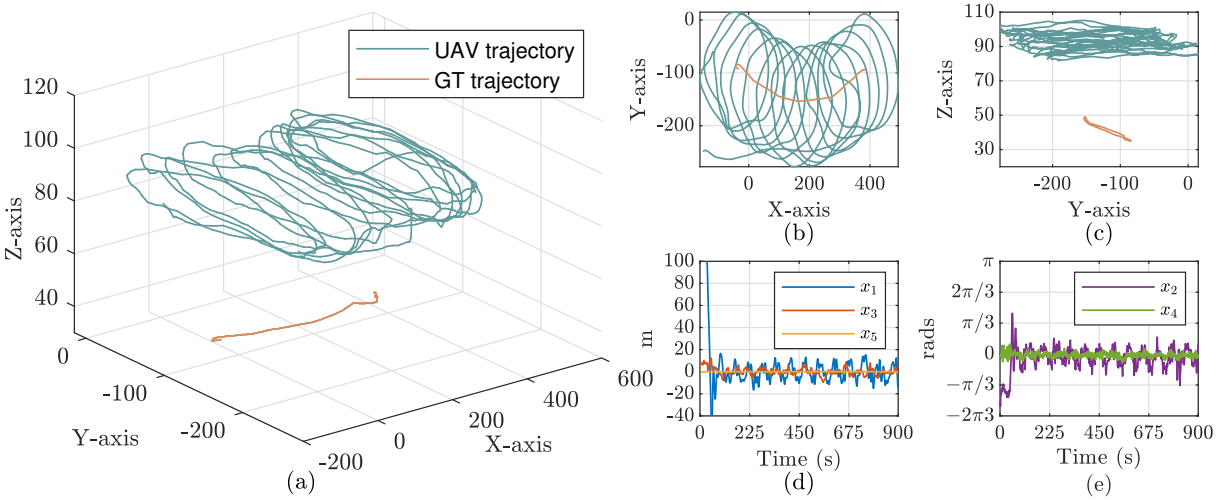


Figure 3.14: Experimental results for a moving target scenario. (a) UAV and ground target trajectories. (b) Top-view of trajectories. (c) Side-view of trajectories. (d) Tracking errors for distance, altitude, and speed. (e) Tracking errors for bearing and pitch angle.

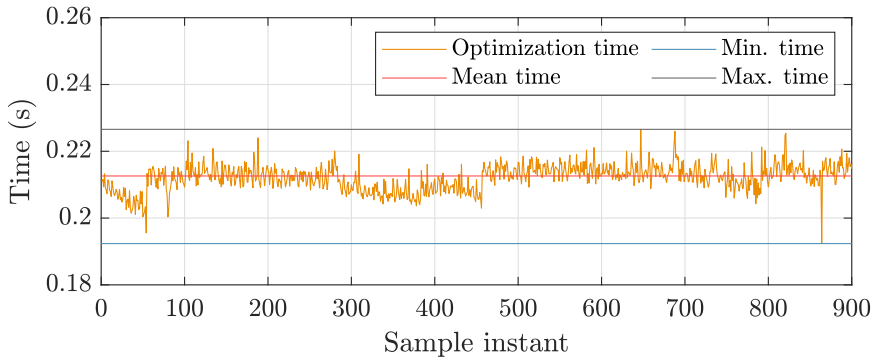


Figure 3.15: Optimization time (GPU based) for moving target on experimental platform.

Moving Target with Unknown Trajectory

Drawing parallels to Subsection 3.8.1, Fig. 3.14a graphically elucidates the 3D trajectories of the UAV and target for the real world scenario. It is important to emphasize that, similarly to the simulation case, the speed of the target ranges from 0 up to 5.22 meters per second, which is within the bounds this proposal can track. A preliminary observation reveals that the circular trajectory and the direction of circumnavigation is accurately achieved. The proposed distance and altitude tracking are evident in Figs. 3.14b and 3.14c, respectively. These illustrate the target trajectory on a 2D plane, emphasizing the circular navigation of the UAV around the target. In contrast, Fig. 3.14c shows the trajectory from a YZ perspective, providing insights into altitude tracking with respect to the target height.

Figures 3.14d and 3.14e present the tracking errors for all states. First, Fig. 3.14d specifically addresses the tracking errors of the distance x_1, altitude x_3 and UAV speed x_5. Upon reaching the reference values, the errors in x_2 and x_3 become negligible. Notably, x_1 exhibits oscillations around the target, where its error is confined between -15 and 15 m. It is worth mentioning that this experimental result shows close similarity with the simulation result presented in Fig. 3.5. This is because the simulation setup is the same as the experimental test, that is, the target movement is identical to the one in this experiment, verifying the accuracy of the presented mathematical model. Once these variables stabilize around the established reference values, they exhibit sustained consistency, barring slight deviations. This dynamic is owing to the unknown nature of the target trajectory. Nevertheless, the NMPC is able to track it effectively for the rest of the experiments. No major divergence is observed for the distance d_k from its reference d_{ss}.

In response, the optimal control effort u_k, obtained by the NMPC can realign the system to bridge these discrepancies. This response is further mirrored by the bearing angle θ_k trajectory. The club's coordinates specified as $33°42'45.8"S$ $151°14'23.2"E$. Finally, it is worth noting that Fig. 3.14 has a hyperlink embedded that will take the reader to an external site, which contains a video recording of the full flight.

Moreover, as previous results,the optimization time at each sampling instant during the experiment is presented. Figure 3.15 shows the results. Compared to previous experimental result, it can be observed that the mean optimization time is around 0.215 seconds. This shows a slight increment compared to the 0.2 seconds from the step experiment. This can be due to a probable temperature spike, making the NVIDIA Jetson GPU to thermal throttling. Nevertheless, the increase in the optimization time is marginal and does not affect the practicality of the proposal. Additionally, the maximum and minimum optimization time is at 0.2266 and 0.1924 seconds, respectively. This results shows that the limits on the optimization time during experimental results are practically the same as the step experiment, suggesting that the complexity of the experiment does not involves

higher optimization time, but rather the configuration on particles and movements of the heuristic solver.

Remark 12. *Although the simulations and experiments share identical controller and system parameters, discrepancies emerge. In particular, altitude tracking in the experimental results outperforms the simulations, which presented oscillations. This difference arises from the fact that the Bat Algorithm used in the experimental results considers more bats and movements in the configuration than the simulation version of the bat algorithm. Parallelization of the algorithm to run on the NVIDIA Jetson Nano GPU allows us to enhance the quality of the solution at each sampling time by increasing the population and movements, thereby improving the likelihood of obtaining superior solutions. This increment does not affect the feasibility of the algorithm for optimizing the problem at each sampling time.*

3.9 Conclusion

In this study, sufficient conditions for ensuring stability in the Lyapunov sense for a fixed-wing UAV tracking a ground target using NMPC are provided. The Lipschitz condition is used to bound the nonlinearities of the system model, defining the design of a terminal region. A local controller is derived using a scaling technique for the matrix Q^*, which allows us to meet the stability conditions.

Based on the stability conditions, a weighting factor sweep is presented, where q_d, q_θ and q_z are modified, resulting in a stability domain for the NMPC loop of a ground target tracking system. An initialization technique is used to solve the optimization problem, which ensures that stability is not compromised by possible suboptimalities originated by heuristic optimization approaches. The simulation and experimental results are presented using a real-world autonomous fixed-wing UAV controlled by the proposed stabilizing NMPC. These results demonstrate the excellent closed-loop performance of the proposed method.

Future research efforts may focus on robustness improvements based on machine learning approaches to mitigate the effects of external disturbances, such as wind and measurement noise.

Chapter 4

Bilinear Time-Variant Disturbance Rejection for a Fixed-Wing UAV Target Tracking using NMPC

This chapter presents a robust Nonlinear Model Predictive Control framework for ground target tracking using fixed-wing UAVs, designed to handle disturbances and achieve accurate real-time tracking of targets with unknown trajectories. Starting with modeling the UAV's kinematics while explicitly incorporating exogenous disturbances into the system dynamics. The target's motion is represented by a first-order dynamic model, which, when combined with the UAV dynamics, yields a full three-dimensional Dubins trajectory formulation centered on the target's current position and heading. To accommodate both external perturbations and evolving target trajectories, a bilinear time-variant state-space representation is introduced. A Kalman filter-based estimation method is then employed to infer both the disturbances acting on the UAV and the target's motion parameters. These estimates are integrated into the NMPC to ensure highly accurate and robust guidance. In comparative studies against established approaches, such as the persistent disturbance estimation, the proposed method achieves improved tracking performance by explicitly transforming the disturbances from a rotational plane, into the 3D plane. These results highlight the practical advantages and overall superiority of the proposed framework for ground target tracking in dynamic, uncertain environments.

4.1 Introduction

In the last decade, there has been a surge in the use of quadcopter UAVs in various applications worldwide [4], [18], [108]. Their relatively simple setup and intuitive handling have made them especially popular within the research community. In contrast, fixed-wing UAVs resemble conventional airplanes, producing lift as airflow passes over their wings during sustained forward motion. Because of their aerodynamics, fixed-wing UAVs often achieve longer flight times compared to their rotary-wing counterparts. However, this comes at the cost of not being able to maintain a fixed position while airborne, as they must continuously move at a speed greater than the minimum required to produce lift, known as the stall speed. This operational nature makes them particularly appealing for extended-range operations, wide-area coverage, and persistent monitoring tasks.

A key research area for fixed-wing UAVs is the development of efficient path planning and trajectory-following algorithms. Due to its computational cost, most approaches involve an offline trajectory optimization process. The resulting path is then used as a reference for the UAV, with an attitude controller guiding the UAV to follow that specific trajectory, as seen in [109], [110], [111]. However, an emerging alternative is to focus on real-time tracking of moving ground targets. In such scenarios, the UAV aims to maintain a desired offset from a target, commonly implemented via an online control strategy and a system model based on distance and bearing angle.

Two primary categories of ground target tracking exist: cooperative and non-cooperative. Cooperative tracking assumes prior knowledge of the target's trajectory, allowing the aircraft to achieve high tracking accuracy with well-established controllers and offline trajectory planning [112], [113].

Some other approaches have been developed for cooperative tracking, where, instead of the path planning approach, an online control system is used to loiter around a target, considering full knowledge of the target movement. Such approaches usually consider linear trajectories, achieving high performance. However, those approaches are not suitable for non-cooperative target tracking with high maneuverability. Usually, the control systems implemented are often based on Lyapunov guidance fields and linear approaches, but some optimal approaches have been recently developed [114], [115].

Among optimal controllers, Nonlinear Model Predictive Control approach has gained notoriety due to its ability to handle system nonlinearities, state and input bounds, and its receding horizon policy, as seen in [116], [117], [118].

Despite NMPC being suitable for cooperative target tracking, the non-cooperative case poses challenges that affect particularly this approach. Most of the models that describe unknown trajectory target tracking assume that the target's kinematics are zero, that is, the target is not moving. This has the advantage that the system model is simplified, but

compromises the accuracy in reference tracking, as seen in [70], [119].

Another way to tackle this is to assume smooth patterns and non-aggressive movements of the target. This allows the use of popular approaches for disturbance estimation that work well, such as persistent estimation [120]. However, for targets attempting to evade, or for a rotating system such as this one, these approaches fail to achieve high tracking accuracy. In NMPC, if assuming these simplifications, then the quality of future state predictions is diminished, which ultimately will degrade the controller's performance.

In addition to this matter, there are still other open problems to solve regarding the robustness of the closed-loop system. Since fixed-wings depend on airspeed to produce lift, they are susceptible to disturbances. Many approaches that consider disturbances focus on the wind action over the UAV, and several works have been published regarding this topic [121], [122], [123], as wind is one of the main sources of exogenous disturbances for fixed-wing UAVs.

Other approaches tackle the wind problem by increasing the reference speed, decreasing the effect of the wind on the UAV kinematics and virtually eliminating the need to estimate its value to reach the control references [124]. This consideration involves the use of high-performance UAVs or, when this is not possible, the implementation of digital twins, relegating the implementation on more accessible hardware UAVs to a secondary priority [125].

Some works where the wind action is considered include real wind data, which is obtained using real-time measurements from a ground weather station [48], but lacks accuracy due to the fact that the wind profile is not exactly the same at ground level as it is at the altitude where the UAV is airborne. Furthermore, other approaches present estimators for the wind based on on-board airspeed sensors. By wind triangle [126], it is possible to use the wind speed of the heading to estimate the action of the wind on other axes to determine the deviation from the heading. Some methods that focus on wind estimation rely on airspeed sensors to be able to accurately estimate the wind [127].

Aside from the mentioned wind, there exist other sources of disturbances that would affect the UAV positioning, such as asymmetric fuselage, off-centered actuators, uneven mass distribution, among others [128].

On the other hand, Kalman filters (KF) have been a valuable tool to solve the estimation problem. They rely on the output error and the state covariance for correction, continuing to adjust the estimate until the difference between the actual and estimated outputs is zero. Several works rely on Kalman filters to estimate wind and disturbances [79], [129]. However, most of the approaches are focused on path following, relegating ground target tracking to a second place [87], [130], [131].

To overcome the above limitations, the work at hand presents a robust fixed-wing based 3D ground target tracking system using NMPC. The contributions of this article are summarized as follows:

1. Derivation of the dynamic disturbances system model, including the nonlinear UAV information for ground target tracking. These disturbances consider the relative motion of the target, wind presence, and also possible displacement on the actuators of the UAV.
2. Design of a recursive Kalman filter-based disturbance compensation strategy to estimate them and account for them at the NMPC optimization stage.
3. Unlike other approaches, the use of kinematics enables the proposal to estimate disturbances without the use of an airspeed sensor.
4. Presentation of an observability analysis for the derived bilinear disturbance model.
5. Simulations and experimental results are presented that validate the approach. To achieve real-time implementation, the proposal is implemented on a Hobbyking Bixler 3 UAV, equipped with a flight controller and a companion computer, capable of parallel programming for non-convex optimization.

This Chapter is organized as follows: Section 4.2 shows the system model with disturbances, Section 4.3 depicts the NMPC approach for ground target tracking, Section 4.4 shows the proposed disturbances compensation strategy. Additionally, Section 4.5 presents simulation and experimental validation of the proposal. Finalizing with conclusions in Section 4.6.

4.2 Kinematic system model with exogenous disturbances

Considering the disturbance-free 3D target tracking system model presented in [118] and [100], now is possible to consider the case where an unknown but bounded disturbance affects each dynamic equation, that is, distance, bearing angle, altitude, pitch angle, and speed. Given these considerations, the following system model is presented

$$\dot{d} = -\mathcal{V}\cos(\chi)\cos(\theta) + \delta_r, \tag{4.1a}$$

$$\dot{\theta} = \frac{\mathcal{V}}{d}\cos(\chi)\sin(\theta) + u_\psi + \delta_\theta, \tag{4.1b}$$

$$\dot{h} = \mathcal{V}\sin(\chi) + \delta_z, \tag{4.1c}$$

$$\dot{\chi} = u_\chi + \delta_\chi, \tag{4.1d}$$

$$\dot{\mathcal{V}} = u_\mathcal{V} + \delta_\mathcal{V}, \tag{4.1e}$$

where d is the distance between the UAV and the target, θ is the bearing angle, h is the UAV altitude with respect to the target, χ is the pitch angle, and $\mathcal{V}$ is the UAV speed. Furthermore, δ_r, δ_θ, δ_z, δ_χ, and $\delta_\mathcal{V}$ denote the disturbances in distance, bearing angle, altitude relative to the target, pitch angle, and speed, respectively. Finally, the control inputs u_ψ, u_χ, and $u_\mathcal{V}$ represent the rate of change for the heading angle ψ, pitch angle χ, and speed $\mathcal{V}$, respectively. From Fig. 3.1, is possible to determine that the control objective is to circumnavigate around a target, while keeping a certain speed, distance, altitude and loitering direction. Therefore, the state reference is defined as $x_{ss} = \begin{bmatrix} d_{ss} & \theta_{ss} & h_{ss} & \chi_{ss} & \mathcal{V}_{ss} \end{bmatrix}'$.

4.2.1 Steady-state Analysis

In most practical applications, $\delta \neq \vec{0}$, which means that the steady-state relies on possible disturbances acting over the system. Finding a single fixed equilibrium point for all disturbances values is generally not possible, but to consider their known intervals and determine under which conditions a steady-state exists is possible. Note that the analysis is performed for constant disturbances, otherwise there is no steady-state. Based on this, a parametric study for the system's steady-state condition in the presence of disturbance is presented. First, is important to determine the corresponding steady-state input vector, $u_{ss} = \begin{bmatrix} u_{\psi,ss} & u_{\chi,ss} & u_{\mathcal{V},ss} \end{bmatrix}'$ that satisfies the equilibrium conditions. Consider the disturbed system at (4.1), at steady state, it follows that $\dot{d} = \dot{\theta} = \dot{h} = \dot{\chi} = \dot{\mathcal{V}} = 0$. By imposing this condition on the system model, (4.1d) and (4.1e) follows that $u_{\chi,ss} = -\delta_\chi$ and $u_{\mathcal{V},ss} = -\delta_\mathcal{V}$. Thus, the equilibrium control inputs for χ and $\mathcal{V}$ depend linearly on the corresponding disturbances and must exactly counteract them.

Furthermore, from (4.1c),

$$0 = \mathcal{V}_{ss} \sin(\chi_{ss}) + \delta_z,$$

which implies

$$\sin(\chi_{ss}) = -\frac{\delta_z}{\mathcal{V}_{ss}}, \tag{4.2}$$

following that, if a certain speed $\mathcal{V}_{ss}$ is determined, then

$$\chi_{ss} = \arcsin\left(-\frac{\delta_z}{\mathcal{V}_{ss}}\right).$$

This condition imposes that, for a given set of disturbances, the chosen steady-state speed $\mathcal{V}_{ss}$ must be large enough so that $|\delta_z| \leq \mathcal{V}_{ss}$, otherwise, no steady state angle χ_{ss} is feasible. Moreover, considering distance dynamics at (4.1a),

$$0 = -\mathcal{V}_{ss} \cos(\chi_{ss}) \cos(\theta_{ss}) + \delta_r,$$

and solving for $\cos(\theta_{ss})$

$$\cos(\theta_{ss}) = \frac{\delta_r}{\mathcal{V}_{ss} \cos(\chi_{ss})}, \tag{4.3}$$

with given $\mathcal{V}_{ss}$ and χ_{ss} from (4.2), check if $\mathcal{V}_{ss}\cos(\chi_{ss}) \geq \delta_r$. If not, then no equilibrium point is possible for the disturbance set.

Finally, solving (4.1b) for $u_{\psi,ss}$

$$u_{\psi,ss} = -\delta_\theta - \frac{\mathcal{V}_{ss}\cos(\chi_{ss})\sin(\theta_{ss})}{d_{ss}}. \tag{4.4}$$

If a suitable $u_{\psi,ss}$ exists for the desired distance d_{ss}, and that value does not violate the input constraints, then an equilibrium point exists for the system, considering the disturbance set.

Remark 13. *A particular case for this analysis is when all disturbances are zero, i.e.* $\delta = \vec{0}$. *The previous equilibrium conditions reduce to a simpler disturbance-free scenario, presented in [118]. In this case, the feasibility constraints are satisfied, and the equilibrium point can be explicitly determined. Specifically, if* $\delta = \vec{0}$, *the steady state conditions are* $\dot{d} = 0 \implies \theta_{ss} \in \{\pi/2, -\pi/2\}$ *and* $\chi_{ss} = 0$, *where* θ_{ss} *determines the UAV direction during tracking. Choosing* $\theta_{ss} = \pi/2$, *and noting that the bearing angle must be kept constant to maintain a fixed distance and altitude, from* (4.1b), $\dot{\theta}{=}0 \implies u_{\psi,ss} = -\mathcal{V}_{ss}/d_{ss}$ *Finally, when reaching the reference altitude and speed, from* (4.1d) *and* (4.1e), $\dot{\chi} = 0 \implies u_{\chi,ss} = 0$, *and* $\dot{\mathcal{V}} = 0 \implies u_{\mathcal{V},ss} = 0$.

This steady-state analysis allows us to define the state and input tracking error form for states and inputs as per (3.18) and (3.19), respectively.

4.2.2 Discretized System Model

Considering the system model presented in (4.1), Remark 13, and using forward Euler's discretization method, the nonlinear discrete-time tracking error form of the system is presented as

$$x_{k+1} = f(x_k, u_k) + \omega_k = \mathcal{G}(x_k) + Bu_k + \omega_k, \tag{4.5}$$

where $\mathcal{G}(x_k) = \begin{bmatrix} g_1(x_k) & g_2(x_k) & g_3(x_k) & g_4(x_k) & g_5(x_k) \end{bmatrix}'$ is given by (3.21), B as per (3.22), and ω_k representing the dicrete-time disturbance vector acting over the system, defined as

$$\omega_k = \begin{bmatrix} T_s\delta_{r,k} & T_s\delta_{\theta,k} & T_s\delta_{z,k} & T_s\delta_{\chi,k} & T_s\delta_{\mathcal{V},k} \end{bmatrix}',$$

with T_s representing the sampling time.

4.3 NMPC Formulation for UAV Ground Target Tracking

Consider the system model presented in (4.5). Moreover, consider the cost function $V_N(\cdot)$ at (3.25). Furthermore, the constrained optimal control problem at (3.27a) is formulated

as

$$\mathbb{P}_N(\hat{x}_0): \quad V_N^{op}(\hat{x}_0) = \min_{\vec{u}} V_N(\hat{x}_0, \vec{u}), \tag{4.6a}$$

subject to:

$$\hat{x}_{i+1} = f(\hat{x}_i, \hat{u}_i) + \omega_k, \tag{4.6b}$$

$$\hat{u}_i \in \mathbb{U}, \tag{4.6c}$$

$$\hat{x}_i \in \mathbb{X}, \tag{4.6d}$$

$$\hat{x}_N \in \mathbb{X}_f \subseteq \mathbb{X}, \tag{4.6e}$$

for all $i \in \{0, \ldots, N-1\}$.

In this formulation, (4.6b) describes the system dynamics, accounting for disturbances related to each state. Furthermore, (4.6c) and (4.6d) are the constraints for control inputs and states, respectively. Additionally, the terminal constraint (4.6e) ensures that the final state $\hat{x}_N$ lies within the terminal set $\mathbb{X}_f$, which is chosen to ensure closed-loop stability, as per [118].

The optimal sequence of control inputs, $\vec{u}^{op}$, that minimizes the cost function $V_N(\cdot)$ is expressed as per (3.29). Then, by applying $\vec{u}^{op}$ to the system (4.6b) the optimal state sequence $\vec{x}^{op}(\hat{x}_0)$ is obtained, defined as per (3.30) The domain of attraction for $V_N(\hat{x}_0)$, denoted as X_N, is then defined as per (3.31), which includes all initial states $\hat{x}_0$ for which there exists an input sequence $\vec{u} \in \mathcal{U}(\hat{x}_0)$, satisfying the constraints (4.6c)–(4.6e).

By employing a receding horizon control policy, as in [105], iteratively solving the optimal control problem at (4.6a), and considering the first element of the optimal input sequence $\hat{u}_0^{op}$, yields to

$$\kappa_N(\hat{x}_0) = \hat{u}_0^{op}. \tag{4.7}$$

Finally, implementing NMPC law into system at (4.5) provides the closed-loop NMPC for ground target tracking

$$x_{k+1} = f(x_k, \kappa_N(x_k)) + \omega_k. \tag{4.8}$$

Remark 14. *It is important to highlight that the NMPC for this system is designed to ensure the closed-loop stability, by following [118]. This means that the cost function weighting factors can be designed to ensure closed-loop stability.*

4.4 Proposed Disturbances Compensation Strategy

Traditional disturbance estimation techniques rely on a standard persistent disturbance strategy, as seen in [120]. In most practical applications this approach is enough to accurately estimate exogenous disturbances and account for them in the control side. Such estimation can refine the control inputs and improve overall performance. However, for

the ground target case covered in this work, this approach is not suitable since the external disturbances, such that target movement, wind speed and other exogenous influences over the kinematics of the UAV, are time-varying. Therefore, this work proposes the use of an observer strategy that models the dynamic disturbances in a bilinear form, allowing them to be tracked and accounted for in real time.

4.4.1 Dynamic Disturbances Model

To account for the possible time-varying nature that the disturbances would exhibit in its current representation, a decoupling technique is presented based on the rotating nature of the system's steady-state that exploits the fact that all states are measurable. Considering that the kinematics of the UAV on x and y axis are represented by

$$\dot{x}_p = \mathcal{V}\cos(\chi)\cos(\psi) + \delta_x, \tag{4.9a}$$
$$\dot{y}_p = \mathcal{V}\cos(\chi)\sin(\psi) + \delta_y, \tag{4.9b}$$
$$\dot{\psi} = u_\psi + \delta_\psi, \tag{4.9c}$$

with x_p and y_p representing the 2D position of the UAV, δ_x and δ_y being unknown but bounded disturbances at the respective axis, and $\mathcal{V}, \chi$ and ψ as per (4.1). In this thesis, a first-order kinematic model for the target is adopted, as

$$\dot{x}_t = \mathcal{V}_t\cos(\chi_t)\cos(\psi_t), \tag{4.10a}$$
$$\dot{y}_t = \mathcal{V}_t\cos(\chi_t)\sin(\psi_t), \tag{4.10b}$$
$$\dot{z}_t = \mathcal{V}_t\sin(\chi_t), \tag{4.10c}$$

where $\{x, y, z\}_t$ is the target 3D position. Furthermore, $\mathcal{V}_t$, χ_t, and ψ_t are the (unknown) speed, pitch angle, and heading angle of the target. Although the true target dynamics may be higher-order or driven by acceleration inputs, this first-order assumption gives us a tractable estimation problem for further estimation. In Section 4.4.4 we show that, under mild structural mismatch, such as a target with time-varying acceleration instead of first-order kinematics, the proposed NMPC remains stable and exhibits only a modest increase in tracking error, demonstrating robustness to model uncertainty. From (3.7) and (3.6), and considering (4.9) and (4.10), $\dot{d}$ is expressed as

$$\begin{aligned}\dot{d} =& -\mathcal{V}\cos(\chi)\cos(\theta) + (\delta_x - \mathcal{V}_t\cos(\chi_t)\cos(\psi_t))\cos(\phi) \\ &+ (\delta_y - \mathcal{V}_t\cos(\chi_t)\sin(\psi_t))\sin(\phi).\end{aligned} \tag{4.11}$$

Noting that, by defining the total disturbance in x and y as

$$\begin{aligned}\delta_X &= \delta_x - \mathcal{V}_t\cos(\chi_t)\cos(\psi_t), \\ \delta_Y &= \delta_y - \mathcal{V}_t\cos(\chi_t)\sin(\psi_t),\end{aligned} \tag{4.12}$$

is possible to define the disturbances acting on the distance as

$$\delta_r = \delta_X \cos(\phi) + \delta_Y \sin(\phi), \tag{4.13}$$

which leads to (4.1a). Furthermore, $\phi = \text{atan2}(y_p - y_t, x_p - x_t)$. Then, using the chain rule, $\dot{\phi}$ is expressed as

$$\begin{aligned}\dot{\phi} = &(\mathcal{V}\cos(\chi)\sin(\psi) + \delta_y - \mathcal{V}_t\cos(\chi_t)\sin(\psi_t))\frac{\cos(\phi)}{d} \\ &-(\mathcal{V}\cos(\chi)\cos(\psi) + \delta_x - \mathcal{V}_t\cos(\chi_t)\cos(\psi_t))\frac{\sin(\phi)}{d}.\end{aligned} \tag{4.14}$$

From (3.7) and (4.12), is possible to define

$$\dot{\theta} = \frac{\mathcal{V}}{d}\cos(\chi)\sin(\theta) + u_\psi + \delta_\psi + \delta_X\frac{\sin(\phi)}{d} - \delta_Y\frac{\cos(\phi)}{d}. \tag{4.15}$$

Similarly to d,

$$\delta_\theta = \delta_\psi + \delta_X\frac{\sin(\phi)}{d} - \delta_Y\frac{\cos(\phi)}{d}, \tag{4.16}$$

leading to (4.1b). Finally, considering that the altitude is described with respect to the target, from (4.1c) and (4.10c),

$$\dot{h} = \mathcal{V}\sin(\chi) - \mathcal{V}_t\sin(\chi_t), \tag{4.17}$$

and considering $\delta_z = -\mathcal{V}_t\sin(\chi_t)$, it leads to (4.1c).

For $\dot{\chi}$ and $\dot{\mathcal{V}}$, the disturbances are treated as constant. This assumption aligns with practical implementation considerations, where misalignment in the actuators commonly induces persistent disturbances in the system. Consequently, no spatial dependence is attributed to these disturbances.

Based on this analysis, the disturbance vector is defined as

$$\delta = \begin{bmatrix}\delta_X & \delta_Y & \delta_z & \delta_\psi & \delta_\chi & \delta_\mathcal{V}\end{bmatrix}'. \tag{4.18}$$

Remark 15. *It is important to emphasize that this procedure does not alter the fundamental form of the disturbances. Rather, it transforms them from their original x, y, z coordinate frame into a d, θ, z coordinate system. In this rotated frame, the disturbances unfold in a plane parameterized by ϕ, which explicitly describes their time-varying behavior. As a result, the individual components of the disturbance vector become slowly varying signals, an approximation that holds so long as their amplitude and elevation change little over one rotation. This quasi-constant behavior in the new coordinates is the key principle that underlies the estimation strategy developed in this chapter.*

4.4.2 Bilinear Disturbances System Model

A common assumption for the disturbances is that their rates of change are small enough to assume that their derivatives are approximately zero, i.e. $\dot{\delta} \approx \vec{0}$. The key idea behind this disturbance system model is to use the rotational behavior of the UAV around the target and rotate the disturbances from the target tracking model, where the disturbances are sinusoidal, to an equivalent decoupled disturbances model on the different axes, where the disturbances are expected to change slowly. These disturbances mostly represent deviations on the kinematic model of the UAV; therefore, it will affect the controller's ability to maintain the reference distance d_{ss} from the target. Furthermore, the target's movement, discussed in Section 4.4.1, is incorporated in the disturbance model. From the nonlinear dynamic model at (4.5), defining $\omega_k = \mathcal{C}(\phi_k, d_k)\delta_k$, and using the forward Euler's discretization method, the dynamically disturbed system can then be rewritten as

$$x_{k+1} = \mathcal{G}(x_k) + Bu_k + \mathcal{C}(\phi_k, d_k)\delta_k, \tag{4.19a}$$

$$y_k = x_k + v_k, \tag{4.19b}$$

where $v_k \in \mathbb{R}^p$ is the p-dimensional measurement noise, assumed to be zero-mean Gaussian with covariance $\mathcal{R}_f$, $v_k \sim \mathcal{N}(0, \mathcal{R}_f)$. Furthermore, $\mathcal{C}(\phi_k, d_k) \in \mathbb{R}^{p\times q}$ is a time-varying matrix that rotates the disturbance vector δ_k into the x, y, z coordinate frame, $\mathcal{G}(x_k)$ is a known nonlinear function of the states, while $\delta_k \in \mathbb{R}^q$ is a q-dimensional unknown disturbance vector. Particularly, $\mathcal{C}(\phi_k, d_k) \in \mathbb{R}^{5\times 6}$ is defined as

$$\mathcal{C}(\phi_k, d_k) = \begin{bmatrix} \cos(\phi_k) & \sin(\phi_k) & 0 & 0 & 0 & 0 \\ \dfrac{\sin(\phi_k)}{d_k} & -\dfrac{\cos(\phi_k)}{d_k} & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.20}$$

In the context of this work, the objective is to estimate the disturbances presented in (4.18). Therefore, considering (4.19b) and assumption $\dot{\delta} \approx \vec{0}$, the following model is introduced

$$\delta_{k+1} = \delta_k + w_k \tag{4.21a}$$

$$y_k = \mathcal{C}(\phi_{k-1}, d_{k-1})\delta_k + \mathcal{G}(x_{k-1}) + Bu_{k-1} + v_k, \tag{4.21b}$$

where $w_k \in \mathbb{R}^q$ is the q-dimensional process noise, assumed to be zero-mean Gaussian with covariance $\mathcal{Q}_f$, $w_k \sim \mathcal{N}(0, \mathcal{Q}_f)$. Moreover, δ_k is the disturbance vector at instant k. In this system model, it is assumed that the disturbances remain approximately constant between sampling times, just being affected by the process noise w_k.

Remark 16. *Note that the nonlinearities of the system are captured in vector $\mathcal{G}(\cdot)$, which depends on the system states. Moreover, as stated in Remark 15, the rotational nature of $\mathcal{C}$ with respect to ϕ is explicitly embedded in this model. This allows the optimizer to account for time-varying disturbances and predict their behavior over the prediction horizon, improving the optimization process. In other words, this formulation accounts for actuator displacements, exogenous disturbances and potential measurement errors, and their evolution over time.*

4.4.3 Observability Analysis

Consider the disturbance system in (4.21). Although the system is linear with respect to δ_k, the nonlinearities of the original target tracking system in (4.1) are encapsulated in $\mathcal{G}(\cdot)$. Moreover, the mapping $\mathcal{C}(\cdot)$ relates δ_k to the measurement y_k. For this case, define $C_k = \mathcal{C}(\phi_k, d_k)$ and $A = I_6$, then, the observability matrix for the linear time-variant (LTV) system [132] can be expressed as

$$\mathcal{O}_n = \begin{bmatrix} C_0 \\ C_1 A_0 \\ C_2 A_1 A_0 \\ \vdots \\ C_{n-1} A_{n-1} \cdots A_0 \end{bmatrix} = \begin{bmatrix} \mathcal{C}(\phi_0, d_0) \\ \mathcal{C}(\phi_1, d_1) \\ \mathcal{C}(\phi_2, d_2) \\ \vdots \\ \mathcal{C}(\phi_{n-1}, d_{n-1}) \end{bmatrix}. \tag{4.22}$$

Here, $\mathcal{O}_n \in \mathbb{R}^{np \times q}$ is the observability matrix. Furthermore, in steady state $d_k \rightarrow d_{ss}$, full observability of the system is guaranteed provided that the heading sequence ϕ_k is persistently exciting. Concretely, there must exist an integer $N > 0$ such that the finite-horizon observability matrix

$$\begin{bmatrix} \mathcal{C}(\phi_k, d_{ss}) \\ \mathcal{C}(\phi_{k+1}, d_{ss}) \\ \vdots \\ \mathcal{C}(\phi_{k+N-1}, d_{ss}) \end{bmatrix}$$

has full column rank for every k, ensuring that the LTV disturbance subsystem remains fully observable.

4.4.4 Proposed KF Estimation Strategy

Considering the observability conditions, a state observer can be designed to estimate the disturbances. Based on (4.21), the following discrete-time bilinear time-variant system model is considered:

$$\hat{\delta}_{k+1} = \hat{\delta}_k + \mathcal{K}_k(y_k - \hat{y}_k), \tag{4.23a}$$

$$\hat{y}_k = \mathcal{G}(x_{k-1}) + Bu_{k-1} + \mathcal{C}(\phi_{k-1}, d_{k-1})\hat{\delta}_k, \tag{4.23b}$$

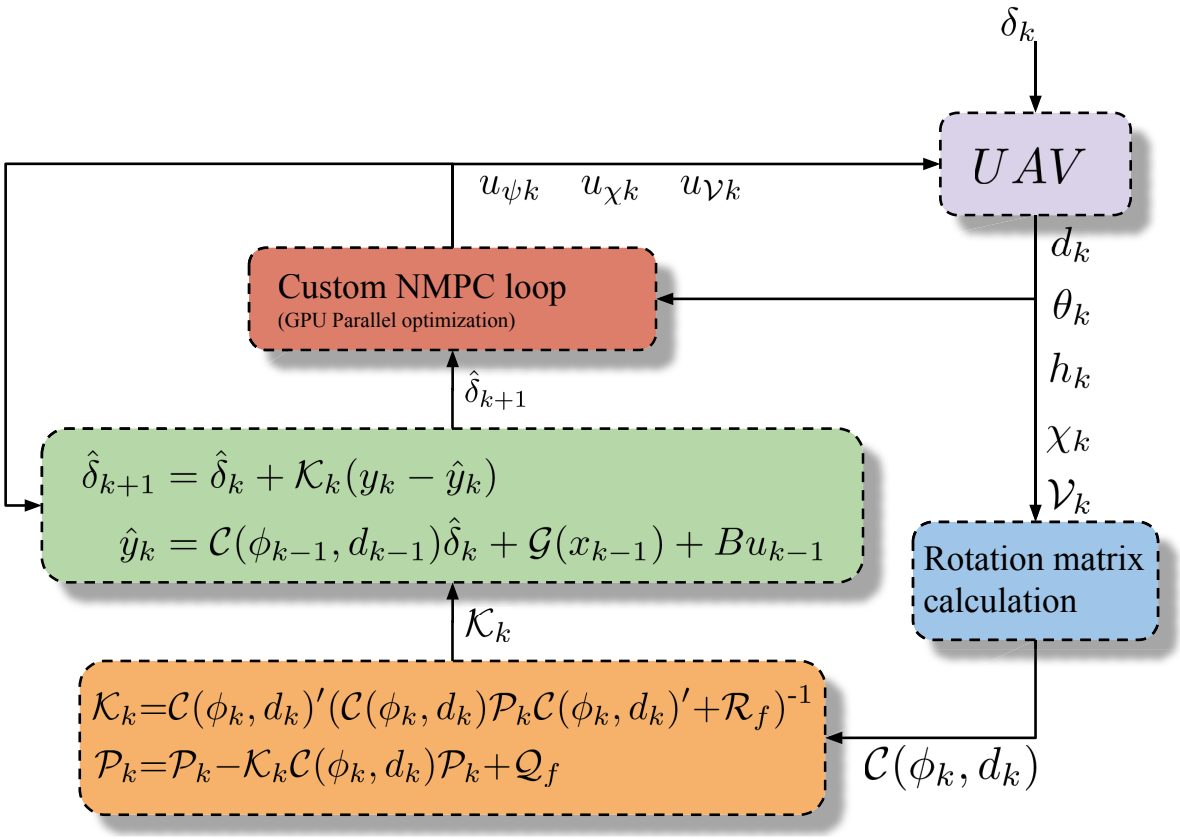


Figure 4.1: Proposed disturbance estimation strategy.

with $\mathcal{K}_k$ being the observer gain matrix. Note that the superscript $\hat{\cdot}$ denotes estimation. Defining the error between the actual disturbance and its estimation at instant k as

$$e_k = \delta_k - \hat{\delta}_k,$$

then, considering the noiseless scenario for (4.21), and from (4.23) the estimation error at instant $k+1$ is expressed as

$$e_{k+1} = I(\delta_k - \hat{\delta}_k) - \mathcal{K}_k\mathcal{C}(\phi_{k-1}, d_{k-1})(\delta_k - \hat{\delta}_k) \tag{4.24}$$

leading to

$$e_{k+1} = (I - \mathcal{K}_k\mathcal{C}(\phi_{k-1}, d_{k-1}))e_k. \tag{4.25}$$

For successful estimation, $e_k \rightarrow 0$ as $k \rightarrow \infty$. Consequently, $\mathcal{K}_k$ is designed such that the disturbance error transition matrix $(I - \mathcal{K}_k\mathcal{C}(\cdot))$ is stable, that is, its eigenvalues have to be inside the unit circle. Given the time-varying nature of $\mathcal{C}(\cdot)$, $\mathcal{K}_k$ has to be computed recursively. Following that, $\mathcal{K}_k$ is obtained by solving the discrete-time algebraic Riccati equation on each sampling time

$$\mathcal{K}_k = \mathcal{C}(\phi_k, d_k)'(\mathcal{C}(\phi_k, d_k)\mathcal{P}_k\mathcal{C}(\phi_k, d_k)' + \mathcal{R}_f)^{-1}, \tag{4.26a}$$

$$\mathcal{P}_k = \mathcal{P}_k - \mathcal{K}_k\mathcal{C}(\phi_k, d_k)\mathcal{P}_k + \mathcal{Q}_f, \tag{4.26b}$$

where $\mathcal{P}_k \in \mathbb{R}^{6\times6}$ is the estimate covariance matrix, $\mathcal{R}_f \in \mathbb{R}^{5\times5}$ is the sensor noise covariance matrix, and $\mathcal{Q}_f \in \mathbb{R}^{6\times6}$ is the process noise covariance matrix. The diagram of the proposal is presented in Fig. 4.1. Furthermore, the pseudo-code of the estimation strategy together with the NMPC is presented in Algorithm 2. Also, it is important to highlight that $\mathcal{R}_f$ represents the variability in the outputs measurement. Furthermore, $\mathcal{Q}_f$ represents the uncertainty of the disturbances model (4.19). Consequently, they can be defined as:

$$\mathcal{Q}_f = \begin{bmatrix} \sigma_{\delta X} & \sigma_{\delta Y} & \sigma_{\delta z} & \sigma_{\delta\psi} & \sigma_{\delta\chi} & \sigma_{\delta\mathcal{V}} \end{bmatrix} \cdot I_6, \tag{4.27a}$$

$$\mathcal{R}_f = \begin{bmatrix} \sigma_r & \sigma_\theta & \sigma_z & \sigma_\chi & \sigma_{\mathcal{V}} \end{bmatrix} \cdot I_5, \tag{4.27b}$$

where the σ_δ components of $\mathcal{Q}_f$ are positive constants that describe the variance of each disturbance in the disturbances system model. Moreover, the σ components of $\mathcal{R}_f$ are the variance of each sensors noise of the system. To achieve close characterization of the sensor noise, $\mathcal{R}_f$ can be determined by measuring a large amount of output data for a constant input to the system. Then, the covariance of that data can be obtained. Additionally, $\mathcal{Q}_f$ can be tuned to achieve the desired observer performance. Small values of $\mathcal{Q}_f$ implies that the system model is accurate, and the observer will rely more on the disturbances model rather than adjusting rapidly due to the output mismatch. On the other hand, a large value on $\mathcal{Q}_f$ implies that the disturbance model has a large uncertainty, which will

Algorithm 2 NMPC with Kalman Filter Observer for 3D Ground Target Tracking

Initialization:
Set $\hat{\delta}_0 \in \mathbb{R}^6$.
Set $\mathcal{P}_0 \in \mathbb{R}^{6\times 6}$.
Define d_{ss}, θ_{ss}, h_{ss}, χ_{ss} and $\mathcal{V}_{ss}$.
Set $\mathcal{Q}_f = \text{diag}(\sigma_{\delta X}, \sigma_{\delta Y}, \sigma_{\delta z}, \sigma_{\delta \psi}, \sigma_{\delta \chi}, \sigma_{\delta \mathcal{V}})$.
Set $\mathcal{R}_f = \text{diag}(\sigma_r, \sigma_\theta, \sigma_z, \sigma_\chi, \sigma_\mathcal{V})$.
Set time step $k \leftarrow 0$.
while mission not completed **do**
    Measure outputs y_k.
    Compute estimated output $\hat{y}_k$
        Compute output error $y_k - \hat{y}_k$
        Update observer gain $\mathcal{K}_k$
        Update disturbance estimate $\hat{\delta}_{k+1}$
        Update error covariance matrix $\mathcal{P}_{k+1}$
    Solve NMPC optimization problem using the estimated disturbance $\hat{\delta}_{k+1}$ to obtain control input u_{k+1}.
    Send control input u_{k+1} to UAV actuators.
    UAV updates its state x_{k+1}
    Increment time step $k \leftarrow k + 1$.
end while

be translated in an observer adjusting rapidly over output mismatches. The drawback of this is that the observer might over-compensate and propagate noise from the sensor to the state estimation.

Remark 17. *Although the disturbance model assumes zero derivatives (i.e., constant disturbances), the proposed observer design, leveraging the time-varying rotating frames, multiple measurements, and adaptive observer gain, continuously updates the disturbance estimates.*

4.5 Results

4.5.1 Simulation

Simulations are conducted to evaluate the performance of both the closed-loop system and the proposed observer. In this simulated environment, multiple types of disturbances in the kinematic model are considered, specifically in the x and y directions.
These disturbances are time-varying, with a low-frequency component to better represent

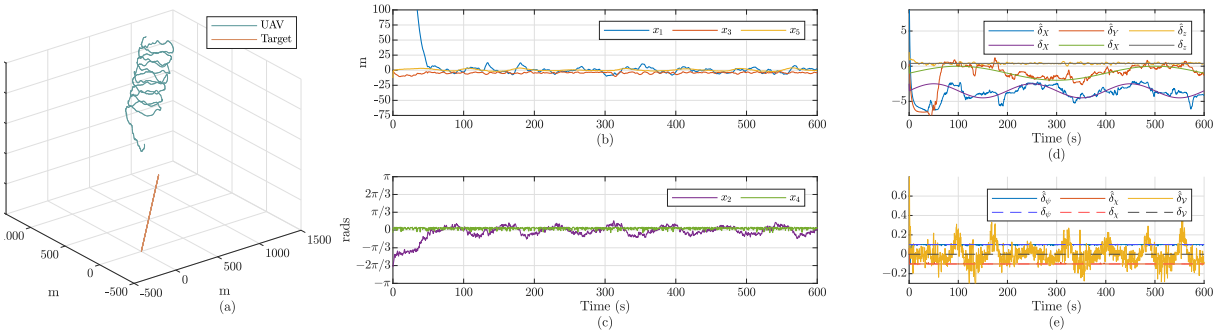


Figure 4.2: Simulation results for target with fixed trajectory. (a) UAV and target trajectories. (b) Distance, altitude and speed tracking errors. (c) Pitch and bearing angle tracking errors. (d) Disturbances estimation.

wind conditions in the field. Furthermore, offsets in the inputs are considered as well, specifically for u_ψ and u_χ. This setup accounts for possible misalignments in the actuators that command the heading and pitch angles, a common scenario. Note that the artificial wind disturbances affect not only the UAV's position but also its speed.

Target with Fixed Trajectory

The first case study involves a target moving at a constant speed and direction, both unknown to the controller. The results can be seen in Fig. 4.2. Figure 4.2(a) shows the UAV and target trajectories, while Figs. 4.2(b) and 4.2(c) present the tracking errors of all states. Moreover, Fig. 4.2(d) and Fig. 4.2(e) illustrates the estimated disturbances, alongside their real values. It can be seen that the UAV accurately tracks the trajectory around the target after it achieves the desired distance, keeping circumnavigation around it. The evolution of the distance error throughout the simulation is observed in Fig. 4.2(b), where, at around the 50-second mark, the UAV reaches the desired distance, starting the circumnavigation, reaching steady state at the same moment. After reaching it, minor deviations are presented. Nevertheless, the highest deviation value is 5 meters, while the RMSE considered from the 100-second mark is 3.2 meters.

Additionally, the actual disturbances δ_X, δ_Y and δ_z, and their corresponding estimation are presented in Fig. 4.2(d). It can be seen that after the UAV reaches the target and starts the circumnavigation, the estimated disturbances converge to their actual values. At around the 50 seconds mark, the disturbance estimation starts converging, and after the 80 second mark, the disturbances are estimated accurately by the proposed observer. Moreover, Fig. 4.2(e) depicts disturbances δ_ψ, δ_χ and $\delta_\mathcal{V}$, and their estimated values. It is evident that, unlike the previous disturbances, these are accurately estimated from the beginning of the simulation. An interesting observation arises with $\hat{\delta}_\mathcal{V}$, which exhibits an oscillation throughout the simulation. This oscillation is due to the wind effect over the speed, nevertheless, the amplitude of this disturbance is small enough to be neglected

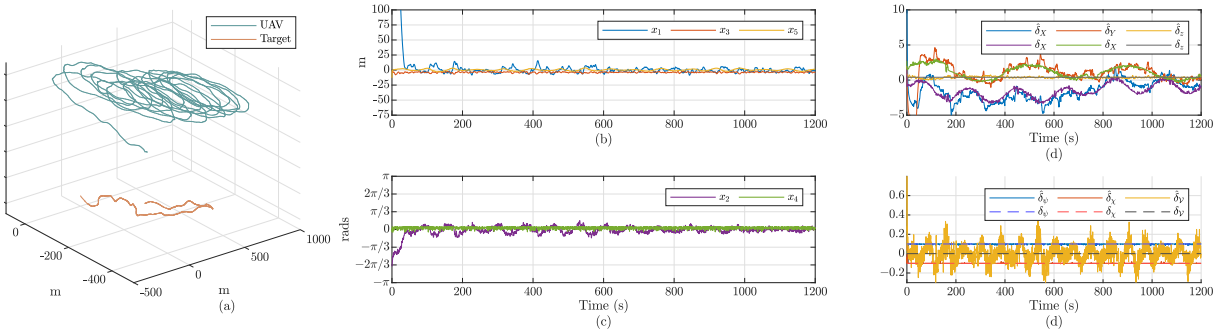


Figure 4.3: Simulation results for target with variable trajectory. (a) UAV and target trajectories. (b) Distance, altitude and speed tracking errors. (c) Pitch and bearing angle tracking errors. (d) Disturbances estimation.

and it oscillates around zero, corresponding to the value of that particular disturbance. This estimation is passed to the NMPC loop, where the values of the disturbances are considered in the cost function at (3.25). This behavior is aligned with the observations at the tracking improvement in Fig. 4.2(b). In this regard, the consideration of the target movement and the disturbances allows the system to virtually reach zero steady state error while loitering the target with unknown movement.

Variable trajectory target

The second case study features a moving target with variable speed and heading. To emulate a real-world scenario, the trajectory was recorded from a real moving vehicle. The disturbances are considered to be the same as the previous case. The results can be seen in Fig. 4.3. The figure depicts the trajectory of the UAV, the state errors and the estimated disturbances compared to their actual values. It can be seen that the UAV tracks the trajectory around the target accurately, with negligible error.

The evolution of the distance error throughout the simulation is observed in Fig. 4.3(b), where, at around 90 seconds mark, the UAV reaches the desired distance, starting the circumnavigation. The small deviations are expected due to the fact that there is no information about the future movement of the target neither the disturbances affecting the UAV. Nevertheless, the highest deviation value is 6 meters, while the RMSE considered from 60 seconds mark is 3.7 meters, highlighting the effectiveness of the proposed method. Additionally, disturbances estimation and their actual values are presented in Fig. 4.2(d) and Fig. 4.2(e). Again, in both Figures, is possible to see that the disturbances are accurately estimated throughout the simulation. It can be seen that, once the UAV reaches the target and begins the circumnavigation, the disturbance estimates $\hat{\delta}_X$ and $\hat{\delta}_Y$ converge, while the rest of the disturbances are estimated from the beginning of the simulation. Is it worth mentioning that the disturbances that the artificial wind generates not only affect the position, but also the speed of the UAV. Nevertheless, the proposal is able to account

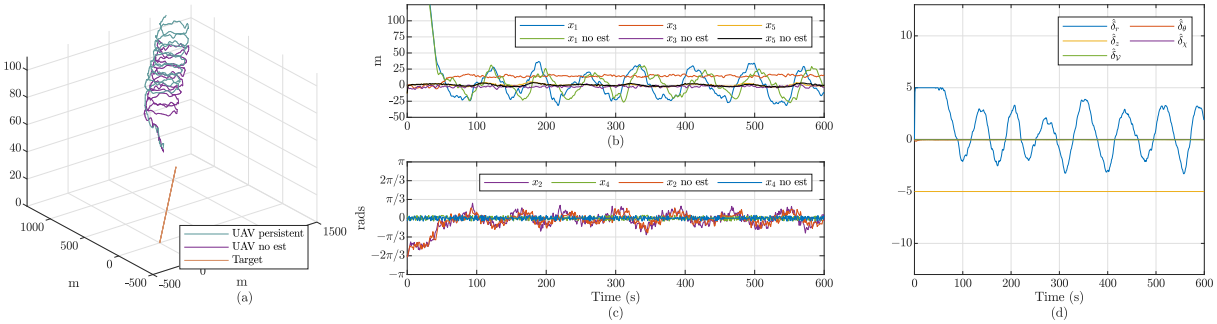


Figure 4.4: Simulation for linear target with and without persistent estimation. (a) UAV and target trajectories. (b) Distance, altitude and speed tracking errors. (c) Pitch and bearing angle tracking errors. (d) Disturbances estimation.

for this effect and counter them, achieving accurate steady state performance.

Alongside these disturbances, small deviations in the actuators are introduced as well. As seen in (4.20), the linear independence of those allow the proposal to estimate them quickly, and account for them shortly after the simulation starts. As shown in Section 4.4.3, all disturbances are estimated once the UAV starts loitering the target.

Comparison with existing methods

To highlight the proposal's advantages, these results are compared with the persistent estimation approach proposed in [120]. Furthermore, another scenario where no disturbances are estimated and the optimizer receives no disturbance-related information is presented. The results of both cases are shown in Fig. 4.4. Under persistent estimation, the closed-loop system experiences oscillations around the reference values, as evidenced by the distance error x_1 peaking at around 25 meters.

The altitude tracking error x_3 displays a constant offset, revealing the reduced accuracy associated with the persistent disturbance approach for this particular system. Figure 4.4(d) further illustrates the oscillatory nature of $\hat{\delta}_r$, underscoring its time-variant behavior, presenting peaks between ± 4 meters. Additionally, the altitude disturbance δ_z saturates at -5 meters, propagating errors through the prediction horizon in the optimization process, degrading the closed-loop performance. In the no-estimation scenario, the system does not rely on any (potentially inaccurate) disturbance estimates.Consequently, the tracking errors are similar or smaller compared to the persistent disturbance approach. Inaccurate disturbance information induces discrepancies that are passed to the optimizer, leading to diminished closed-loop performance. Nonetheless, both cases exhibit strong oscillations in the distance tracking error.

Figure 4.5 depicts the moving target case for both the persistent disturbances approach and without disturbance estimation. Results align with previous linear trajectory case. It is seen that, considering that both approaches present significant oscillations around the

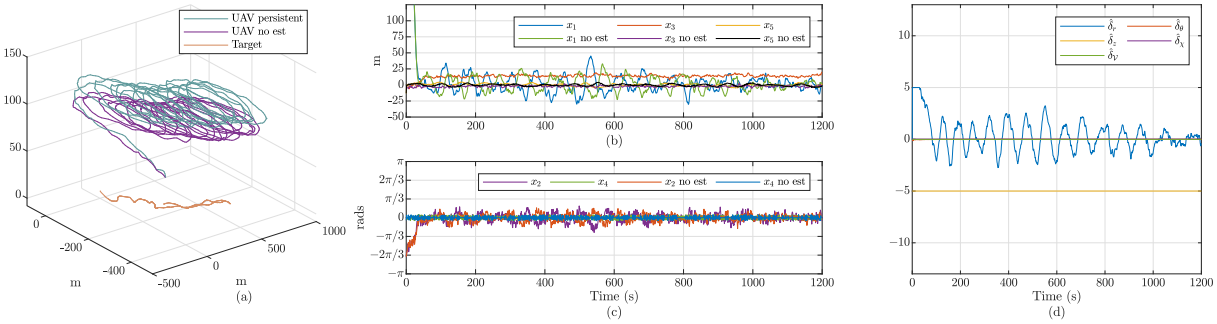


Figure 4.5: Simulation for moving target with and without persistent estimation. (a) UAV and target trajectories. (b) Distance, altitude and speed tracking errors. (c) Pitch and bearing angle tracking errors. (d) Disturbances estimation.

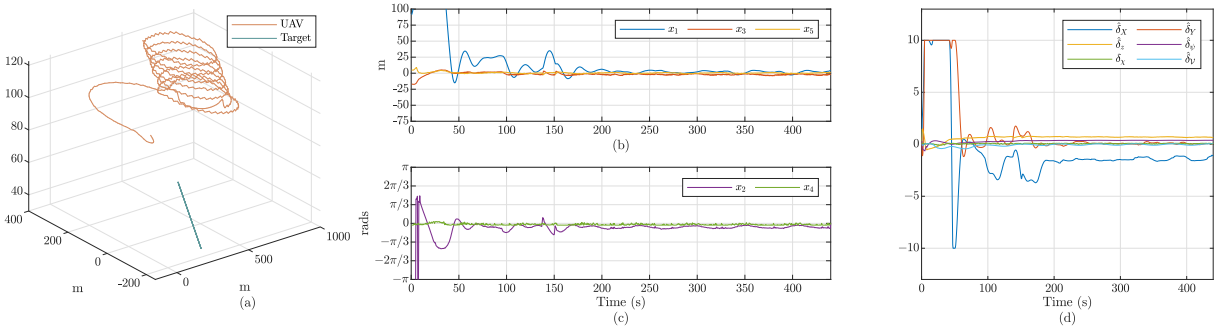


Figure 4.6: Experimental results for target with fixed trajectory. (a) Trajectories for UAV and target. (b) Distance, altitude and speed tracking errors. (c) Bearing and pitch tracking errors. (d) Estimated total disturbances.

reference value, the peak values for distance tracking error are greater in the persistent disturbances case. At around the 550 seconds mark, a spike that reaches 50 meters of distance error is seen. For the rest of the simulation, both cases present oscillations that varies in between ± 25 meters. In the case of the persistent disturbances approach, the estimated disturbance of the distance behaves similarly as previous case, varying its value throughout the simulation around ± 4 meters. Furthermore, it is seen that the altitude disturbance saturates at -5 meters, affecting the tracking of the altitude tracking error.

4.5.2 Experiments

Experimental results are presented to evaluate the proposal's ability to account for external disturbances in a real scenario. The experimental platform is considered following the guidelines at [99]. These experiments were performed at the Warringah Radio Control Society, located in Belrose, NSW, Australia. Similar to the simulations, two test scenarios are presented.

Target with fixed trajectory

Similarly to the simulations, the first scenario considers a target to move with an unknown yet fixed speed and heading. Its starting point is located at $(50, 150, 30)$. The speed of the target is approximately $\mathcal{V}_t = 2.23\ m/s$ and its heading $\psi_t = -1.1\ rads$. Additionally to the parameters setup at the simulation stage, a displacement of the yaw actuator is introduced in the experimental platform. This is to highlight the observer performance to account for possible endogenous disturbances that ultimately will affect the performance of the controller. These calibration issues are one of the main reasons of operational deviations, as stated in [128]. Furthermore, the expected wind during the flight, according to the forecast provided by the Australian Bureau of Meteorology, is around $2-3$ meters per second, with a Northwest direction.

The experimental results for this scenario are depicted in Fig. 4.6. Specifically, Fig. 4.6(a) shows the trajectories of the UAV and the target, where a linear path for the target is observed, while the circumnavigation of the UAV around the target is depicted. Furthermore, Fig. 4.6(b) shows the distance, altitude and speed tracking errors evolution throughout the experiment. During the first 100 seconds, the system runs without the proposed observer. Afterwards, the tracking errors converge to zero error after 50 seconds. It is evident that the proposal significantly improves the tracking of the reference distance. Additionally, Fig. 4.6(c) presents the tracking errors for bearing and pitch angle. These states also converge to the desired value. The bearing angle shows a clear improvement after around 150 seconds, reducing oscillations and settling near its steady-state value. Moreover, the tracking of the bearing angle presents a steady-state error. This can be explained by the fact that the distance error affects the cost function more than the bearing angle error. Notwithstanding with these considerations, the controller is able to reliably remain the UAV at the desired altitude from the target. Additionally, the distance tracking error yields a RMSE of 2.72 m, highlighting its effectiveness. Figure 4.6(d) elucidates the estimated values of the disturbances for this experiment. It is clear that the displacements of the actuators and measurements, namely $\delta_z, \delta_\psi, \delta_\chi$ and $\delta_\mathcal{V}$ are the first to settle at their real values, confirming the ability of the proposal to counter that deviation. Furthermore, the rest of the disturbances are observed only when the UAV starts the loitering, i.e. at around 50 seconds. After that, it takes approximately 50 seconds for the observer to converge the values of the disturbances. Especially interesting is the case of $\hat{d}_X$ and $\hat{d}_Y$ which corresponds to the mixture between the wind influence over the UAV and the target movement. Additionally, the time variant rotation matrix allows us to decouple the components of the wind and estimate them as continuous signals. Noting that the fluctuations are small, both measures remain almost constant after the 180 second mark, with small variations due to the changing wind.

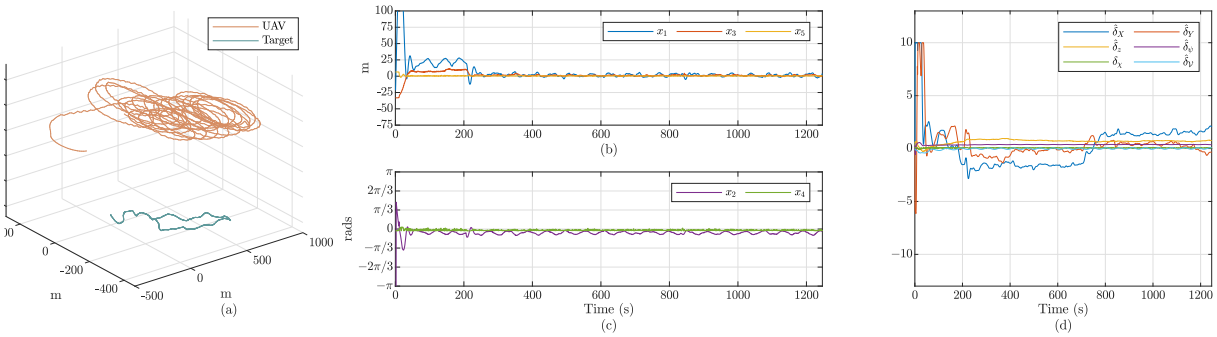


Figure 4.7: Experimental Results for Moving Target. (a) Trajectories for UAV and the ground target. (b) Distance, altitude and speed tracking errors. (c) Bearing angle and pitch angle tracking errors. (d) Estimated disturbances.

Moving target with variable trajectory

Similarly to the second scenario in simulations, the experimental setup is tested while tracking a real vehicle. The movement of the target is not known by the controller and its direction and speed vary throughout the experiment. The target is moving from origin coordinate, following a 3D varying path.

In this case, the moving nature of the target implies that the exogenous disturbances acting on the UAV correspond to disturbances caused by wind presence and the movement of the target all together, which translates into a displacement in the distance of the UAV relative to the target, as shown in (4.1). As the previous case, the 0.2 radians displacement on the yaw input actuator is also considered, alongside with an unknown displacement on the pitch input actuator. Furthermore, although this test was conducted 30 minutes after the previous one, the wind conditions had significantly improved, with speeds not exceeding 1 meter per second. Consequently, the results were more favorable.

The results considering moving target are depicted in Fig. 4.7. Similarly as before, Fig. 4.7(a) shows the trajectories of the UAV alongside the trajectory of the ground target. From the figure, it is clear that the proposal exhibits a clockwise path around the target, which aligns to the reference θ_{ss}.

Moreover, Fig. 4.7(b) show the distance, altitude and speed tracking errors, while Figure 4.7(c) depict the tracking errors for the bearing and pitch angles. Again, the proposal performs accurate tracking even with wind disturbances and target movement, reflected as small oscillations around the zero tracking error for distance. Additionally, the altitude and speed tracking errors are virtually zero, with no oscillations and major deviations. Moreover, it can be seen that for the bearing and pitch angle, the error and oscillations are minimal, showcasing the good performance of the proposal. There are minor spikes present on the distance tracking error, this is due to the fact that, even when the disturbances model assumes their derivatives to be equal to zero, the estimation has to deal

with the changing nature of the target movement and variations in the wind. Then, this causes the estimation of the disturbances need to converge to an updated value. Additionally, for the distance tracking error the root mean-squared error is computed from the 200 second mark, resulting in 5.4 m. This small error highlights the accuracy of the proposed observer.

Finally, Figure 4.7d shows the estimated disturbances throughout the experiments. Similarly as the stationary case, the yaw displacement is the first to settle at 0.22 radians, aligning with the physical displacement implemented in the experimental setup. As expected from previous case, $\hat{\delta}_\psi$ do not show variation once settles. Similar is the case for $\hat{\delta}_z$, $\hat{\delta}_\chi$ and $\hat{\delta}_{\mathcal{V}x}$, where they settled at 0.5 meters, 0.1 rads, and 0 respectively. Furthermore, the rest of the observed disturbances are estimated only when the UAV starts the loitering, i.e. at around 100 seconds. For $\hat{\delta}_X$ and $\hat{\delta}_Y$, more variations can be seen.
A key observation comes from their variable behavior, which is due to the fact that the disturbances system model consider considers both wind and target movement. The previous implies that the value of the estimated disturbances will not be a constant value for a target with variable trajectory. This is reflected then as variations for $\hat{\delta}_X$ and $\hat{\delta}_Y$ that align with the target movement. This is highlighted at around 750 seconds, where the value of $\hat{\delta}_X$ crosses zero and goes from negative to a positive value. At this point, the target changes its x direction to the opposite direction.
Similar is the case for $\hat{\delta}_Y$ but in a lower magnitude. For the total disturbances, their values are bounded from around $\pm 3\ m/s$ for $\hat{\delta}_X$, and $\pm 1\ m/s$ for $\hat{\delta}_Y$.

4.6 Conclusion

In this chapter, a robust UAV ground target tracking system using NMPC is presented. A comprehensive derivation of a three-dimensional relative movement tracking model was provided, encompassing the target's motion as well as both endogenous and exogenous disturbances. To accurately observe disturbances, a rotation-based technique that improves disturbance estimation by transforming disturbances from rotational coordinates into continuous values is presented. This approach yields a disturbance estimation system described by a bilinear time-varying model.
An observability analysis of the system was conducted, establishing conditions under which the disturbances can be estimated. The results align with these theoretical conditions, demonstrating accurate disturbance estimation as long as the UAV performs a loitering maneuver around the ground target.
Simulation and experimental results further validated the effectiveness of the proposed approach. Disturbances were estimated with high accuracy, and tracking errors were significantly reduced. Notably, the estimation process accounted for wind effects, actuator displacements, and the target's movement. Such holistic consideration is particularly

critical in the NMPC framework, where precise prediction of future states is essential for achieving optimal control performance.
The proposed estimation strategy is compared to a traditional persistent disturbance approach, showcasing its performance for this particular system, where the time-varying nature of the disturbances is explicitly considered by using a rotation matrix. This method effectively observes and tracks the disturbances as the UAV loiters the target. Thus, it ensures accurate estimation despite the simplifying constant disturbance assumption. This allows to introduce this data into the optimization process, enhancing the overall closed-loop performance.

Chapter 5

ILC for unmodeled disturbances compensation for Ground Target Tracking using NMPC

In this Chapter, a novel control strategy that combines Nonlinear Model Predictive Control with Iterative Learning Control to enhance the ground target tracking performance of fixed-wing Unmanned Aerial Vehicles in the presence of unmodeled disturbances is proposed. The proposed approach leverages ILC to iteratively refine control actions based on the repetitive nature of the UAV's orbital motion around the target, improving tracking accuracy without requiring explicit disturbance modeling or observer-based estimation. To analyze system behavior under this hybrid control scheme, a reduced-order dynamic model is employed, which allows for a transfer function-based closed-loop stability analysis of the NMPC-ILC framework. The NMPC component generates optimal control actions within the prediction horizon, incorporating system dynamics and constraints, while the ILC component compensates for disturbances recurring at specific orbital phases. In this framework, ILC functions as an input disturbance generator for the optimization process, allowing the NMPC to compute the total control input applied to the system. This approach enhances disturbance rejection and tracking accuracy while ensuring that inputs remain within feasible constraints. The proposed strategy enables UAVs to effectively track moving ground targets with unknown trajectories. Simulation results demonstrate the effectiveness of the method in scenarios with significant uncertainties and disturbances, validating a robust and adaptable control framework for fixed-wing UAV ground target tracking.

5.1 Introduction

Ground Target Tracking has garnered considerable interest in recent years as one of the primary applications where fixed-wing UAVs often excel. This problem has two well differentiable branches: detection and control. In the detection side, several advancements have been made for measuring the actual position of the ground target based on images [55], [56]. On the other hand, the control side has seen the development of control laws, overhead tracking, and other optimal techniques [63], [66]. Early approaches frequently assumed stationary or predictably moving targets, focusing on straightforward guidance laws for overhead tracking [133] or path-planning solutions using artificial potential fields [134]. More recent developments for online target tracking commonly use a two-dimensional Dubins vehicle model [135], with various controllers, including sliding mode control [32], [111], Lyapunov guidance fields [51], and Model Predictive Control [81], being explored.

In the MPC realm, advanced control schemes such as NMPC have proven particularly attractive for GTT due to their ability to handle system constraints, nonlinear dynamics, and predictive behaviors [35]. Nonetheless, many existing MPC-based methods rely heavily on prior knowledge of the target's motion or accurate disturbance models, assumptions that may not hold when the target's trajectory is fully or partially unknown, or when the operating environment is highly uncertain [136].

Additionally, observer-based approaches have been proposed to estimate and compensate for disturbances in UAV applications [137]. These methods, while effective in many cases, can become cumbersome if the system model deviates significantly from reality or if the disturbances cannot be conveniently parameterized. In highly dynamic scenarios, such as those involving abrupt wind gusts or drifting target motion, traditional observer-based disturbance rejection strategies can lead to suboptimal tracking performance and may require extensive tuning or redesign of the observers [138].

In contrast, data-based methods have recently gained notoriety for their ability to address discrepancies between the system's output and the reference without relying on detailed disturbance models [139]. Typically, these approaches use input/output information to estimate the presence of disturbances or to inject additional inputs that correct output errors [140]. Among these methods, Iterative Learning Control stands out as a data-driven technique for incrementally improving control performance by leveraging data from previous iterations of a task [89], [90], [91]. By systematically refining control inputs over successive trials, ILC can achieve robust tracking without the need for explicit disturbance modeling [141].

Standard time-based ILC, however, usually assumes that the disturbances or reference trajectories repeat consistently in time, and also that similar initial conditions apply at

the start of each iteration [142]. This suggests that purely time-based ILC approaches may struggle in tasks where recurring disturbances are tied to particular orbital or spatial positions rather than discrete time instances [143]. In a fixed-wing UAV ground target tracking scenario, the repetitive task can be viewed as the UAV circumnavigating the target, so the repetition is tied to the UAV's position with respect to the target. To address this issue, an angle-parametrization-based ILC framework can be employed in the implementation stage, aligning the learning process with the phase angle of the UAV's orbit. The idea behind this is to allow the controller to detect and compensate for disturbances that recur at specific orbital phases, offering a more direct approach to handling wind gusts, actuator biases, or other unmodeled effects that repeat on a per-orbit basis.

Motivated by these considerations, this chapter proposes a novel strategy that integrates NMPC and phase-based ILC to tackle the ground target tracking problem for fixed-wing UAVs in the presence of unknown disturbances. In the presented approach, the NMPC component provides predictive, optimal control commands within a predefined horizon, ensuring adherence to system constraints while accommodating nonlinear dynamics.

Concurrently, the ILC strategy iteratively refines these commands by learning from orbital phase-specific deviations in the UAV's distance to the target observed in previous orbits, thereby improving tracking accuracy without the need for explicit disturbance modeling or observer-based compensation.

The key contributions of this work are:

- A based ILC scheme that refines control actions based on the UAV's orbital phase, effectively compensating for recurring disturbances without explicit disturbance models.
- The analysis based on a reduce-order dynamic model representation of the closed-loop system that captures the dynamics of the optimal NMPC. This allows to analyze the convergence of the proposed ILC when working along the NMPC.
- A convergence analysis of the proposed framework that exploits the behavior of the closed-loop NMPC system in a neighborhood of the reference to effectively implement ILC and ensure error contraction throughout learning iterations.
- The integration of this ILC scheme with an NMPC, by injecting the generated ILC input into the optimization stage as a known input disturbance, forming a hybrid control architecture that leverages the advantages of NMPC and data-driven learning.
- A thorough evaluation of the proposed approach through digital twin results, demonstrating robust performance in scenarios characterized by significant model uncertainties and unpredictable target trajectories.

The remainder of this chapter is organized as follows: Section 5.2 introduces the kinematic ground target tracking model and discusses the relevant assumptions. Section 5.3 presents the fundamentals of the NMPC formulation, followed by the principles of the proposed phase-based ILC in Section 5.6. The integration of the two methods is provided in Section 5.6, and Section 5.7 provides simulation results and analyses. Finally, concluding remarks and potential avenues for future work are given in Section 5.8.

5.2 Kinematic Ground Target Tracking Model

From the works at [118] and [100], the undisturbed three-dimensional target tracking model at (3.17) is considered. For this model, the steady state presented at Remark 13 is valid. The previous allows to use the tracking error forms for states and inputs as per (3.18) and (3.19), respectively. Consequently, the discrete time system model at (3.20) is considered.

5.3 NMPC Scheme for ground target tracking using UAV

Since the approach presented in this Chapter utilizes the system model described in (3.17), the NMPC formulation outlined in Section 3.5 is adopted. For brevity, the detailed derivation is omitted.

5.4 Stability Analysis

This section briefly outlines the key theoretical results that ensure closed-loop stability under the proposed NMPC framework. Specifically, two theorems originally presented and proven in Section 3.6 are used, which collectively guarantee that the ground target tracking system under NMPC is stable. The main steps involve analyzing the linearized system and deriving an unconstrained optimal control law, constructing a local stabilizing controller associated to a defined terminal region, and ensuring that the multi-step NMPC scheme guarantees closed loop stability.

Both Theorems rely on the existence of a well-defined terminal region $\mathbb{X}_f$, which is invariant under some local control law. This ensures the NMPC can hand off the system to a locally stabilizing controller at the end of the prediction horizon to remain within its confines.

Proper selection of V_f, is crucial for achieving multi-step stability. Using $\mu \geq 1$ as a scaling factor in the terminal cost helps guarantee that the cost function decreases over

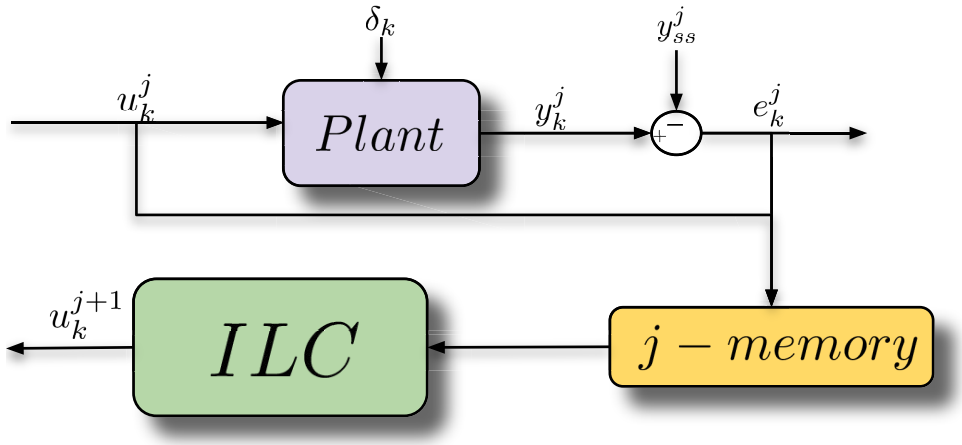


Figure 5.1: ILC basic control scheme.

time.

While increasing the prediction horizon N can improve performance and enlarge the domain of attraction, these results hold as long as N is sufficiently large to ensure feasibility. In many practical scenarios, one designs N to balance computational load and closed-loop performance, as demonstrated in Section 3.8.1.

Remark 18. *Note that the the stability analysis presented at Section 5.4 is valid for the closed-loop system considering NMPC acting over the ground target tracking system. This property is vital to ensure convergence of the ILC approach for ground target tracking, as it is demonstrated in Section 5.5.3.*

5.5 Iterative Learning Control for ground target tracking

ILC improves system performance in repetitive tasks by learning from past errors, refining the input-output relationship from iteration j to $j+1$. After each iteration, the error between the desired and actual output is stored in memory, along with the input applied to the system.

This method is particularly effective when the system dynamics are complex or difficult to model accurately, but the tasks are repetitive enough to allow for learning from previous iterations. The basic ILC control scheme can be seen in Fig. 5.1. There are two classical assumptions for ILC implementation:

Assumption 1. *The disturbance vector δ_k acting over the system is bounded.*

Assumption 2. *The initial conditions of the system are reset at the beginning of each learning iteration j; that is, $x^j_0 = x^{j+1}_0$ for all $j \in \mathbb{Z}_+$.*

During the first iteration ($j = 1$), the control input $u_k^{(j)}$ and the error $e_k^{(j)}$, defined as the deviation between the system output $y_k^{(j)}$ and the reference $y_{ss_k}^{(j)}$, are stored over ℓ time instants as the task is performed. This process generates the vectors

$$u^{(j)} = [u_1^{(j)}, u_2^{(j)}, \cdots, u_\ell^{(j)}]$$

$$e^{(j)} = [e_1^{(j)}, e_2^{(j)}, \cdots, e_\ell^{(j)}],$$

representing the inputs and errors from the previous iteration, respectively.

Upon completing this initial learning iteration, the input signals for the next iteration are calculated by

$$u_{\text{ILC}}^{(j+1)} = u_{\text{ILC}}^{(j)} + L(z) \cdot e^{(j)}, \tag{5.1}$$

where $L(z)$ is the learning rate yet to be designed. This process results in the input vector for next iteration u_{ILC}^{j+1}, defined as

$$u_{\text{ILC}}^{(j+1)} = [u_1^{(j+1)}, u_2^{(j+1)}, \cdots, u_\ell^{(j+1)}].$$

For ground target tracking, it is desired to mitigate overshoot, oscillations, and rapid changes in the error signal that might arise. As per [144], PD-ILC is more suitable for systems with those characteristics. In this regard, the PD-ILC learning rate in z-domain $L(z)$ is defined as

$$L(z) = L_{KP} + \frac{L_{KD}(z-1)}{T_s}, \tag{5.2}$$

where L_{KP} and L_{KD} are the proportional and derivative learning rates, respectively.

Equation (5.1) computes a corrective adjustment vector that reduces the error at each instant k in the next iteration. Essentially, as more iterations of the repetitive task are performed, the accuracy of the reference tracking improves.

In this context, ILC to mitigate disturbances affecting distance tracking performance in ground target tracking is applied. To achieve this, a reduced-order dynamic model that captures the closed-loop dynamics of the original system controlled via NMPC is proposed.

5.5.1 Proposed reduced-order dynamic model

As discussed in Section 5.4, during the transient phase, NMPC guides the system toward a neighborhood of the reference. From the steady-state analysis in Section 5.2, it is established that as the system approaches this neighborhood, the tracking errors Δh, $\Delta\chi$, and $\Delta\mathcal{V}$ tend to zero. Additionally, the nonlinear terms diminish, i.e., $\eta(\cdot) \approx 0$.

In this regime, NMPC behaves similarly to a linear MPC, as the effect of nonlinearities becomes negligible. As shown in [82], when the system operates near the equilibrium

point, the MPC's receding horizon optimization tends to the LQR infinite horizon cost. Therefore, at this point, the NMPC law aligns with the LQR solution.

Specifically, by considering $\mathbb{X}_f$ as the neighborhood of the reference where ILC is active, ensuring that the matched LQR solution does not violate the imposed constraints in (3.27c).

Remark 19. *Note that while the reduced-order model assumes $\Delta h \approx \Delta\chi \approx \Delta\mathcal{V} \approx 0$, Equation (3.24) indicates that only $\Delta\mathcal{V} \approx 0$ is strictly necessary. This is because Δh and $\Delta\chi$ are decoupled from the other system states. Moreover, the inputs Δu_χ and $\Delta u_\mathcal{V}$ influence only the decoupled states.*

For this particular system, the 3×5 gain matrix K for the local controller in (3.35) is given by

$$K = \begin{bmatrix} K_{11} & \cdots & K_{15} \\ K_{21} & \ddots & K_{25} \\ K_{31} & \cdots & K_{35} \end{bmatrix}. \tag{5.3}$$

Based on the previous analysis, it is possible to express Δu_ψ as the equivalent input

$$\Delta u_\psi = -K_{11}\Delta d - K_{12}\Delta\theta, \tag{5.4}$$

with Δd and $\Delta\theta$ as per (3.18). This leads to the reduced-order system model

$$\overline{x}_{k+1} = A_2\overline{x}_k + B_2 u_\psi, \tag{5.5}$$

where $\overline{x} = \begin{bmatrix} \Delta d & \Delta\theta \end{bmatrix}'$. Furthermore,

$$A_2 = \begin{bmatrix} 1 & T_s\mathcal{V}_{ss} \\ \dfrac{-T_s\mathcal{V}_{ss}}{d_{ss}^2} & 1 \end{bmatrix},$$
$$B_2 = \begin{bmatrix} 0 & T_s \end{bmatrix}'.$$

Note that this reduced model aligns with (3.23). By combining this reduced model with the simplified actuation in (5.4), it is possible to generate transfer functions for the simplified closed-loop dynamics.

5.5.2 Transfer function-based closed-loop stability analysis

Based on the reduced system (5.5), the transfer function relating Δu_ψ to $\Delta\theta$, denoted as $G_{u\theta}(z)$, are defined as

$$G_{u\theta}(z) = \frac{T_s(z-1)}{(z-1)^2 + \left(\dfrac{T_s\mathcal{V}_{ss}}{d_{ss}}\right)^2}. \tag{5.6}$$

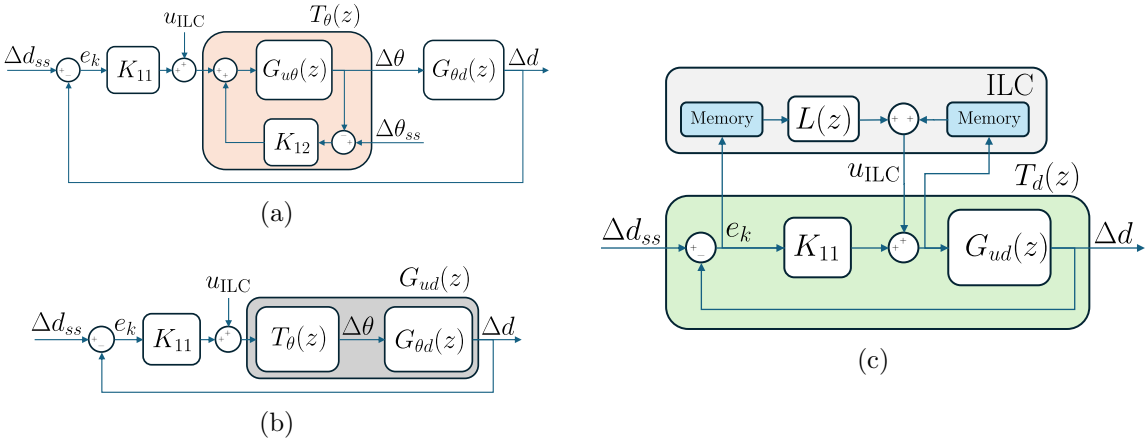


Figure 5.2: Reduced order closed-loop system. (a) Original system. (b) Closed-loop reduction for $\Delta\theta$. (c) Reduced model with ILC compensator and Δd reduction.

Similarly, the transfer function $G_{\theta d}(z)$ is given by

$$G_{\theta d}(z) = \frac{T_s \mathcal{V}_{ss}}{z-1}, \tag{5.7}$$

leading to the system presented in Fig. 5.2(a). Since the reference θ tracking error satisfies $\Delta\theta_{ss} = 0$, the closed-loop transfer function for θ is denoted as

$$T_\theta(z) = \frac{K_{12} T_s d_{ss}^2 (z-1)}{T_s^2 \mathcal{V}_{ss}^2 + d_{ss}^2 (z-1)\left((z-1) + K_{12} T_s\right)}, \tag{5.8}$$

leading to the system shown in Fig. 5.2(b).

Furthermore, defining

$$\begin{aligned} G_{ud}(z) &= T_\theta(z) G_{\theta d}(z) \\ &= \frac{K_{12} \mathcal{V}_{ss} T_s^2 d_{ss}^2}{T_s^2 \mathcal{V}_{ss}^2 + d_{ss}^2 (z-1)\left((z-1) + K_{12} T_s\right)} \end{aligned} \tag{5.9}$$

yields to the system depicted in Fig. 5.2(c). Analogous to $\Delta\theta_{ss}$, considering that the reference $\Delta d_{ss} = 0$, it enables the definition of the closed-loop system $T_d(z)$. Additionally, Fig. 5.2(c) illustrates the ILC applied to the system. Note that the ILC ultimately reacts to the dynamics of $T_d(z)$ and corrects the distance tracking error. Thus, ensuring the stability of the closed-loop transfer functions $T_\theta(z)$ and $T_d(z)$ is crucial.

Lemma 2. *The NMPC is designed to ensure the existence of a local controller κ_f that stabilizes the system in terminal region $\mathbb{X}_f$. Consequently, this local controller is derived from the solution of the Riccati equation. Therefore, from the insights presented in Section 5.5.1, K_1 and K_2 stabilize the system depicted in Fig. 5.2(c).*

5.5.3 Convergence Analysis

For notational simplicity, denote the distance tracking error Δd at iteration j and sampling instant k as $e_k^{(j)}$.From an ILC perspective, the z-domain transfer function relating the

distance tracking error to the ILC input is given by

$$T_{\mathrm{ILC}}(z) = \frac{E(z)}{U_{\mathrm{ILC}}(z)} = \frac{-G_{ud(z)}}{1 + K_{11}G_{ud}(z)}, \tag{5.10}$$

where $E(z)$ and $U_{\mathrm{ILC}}(z)$ are the z-transform of e_k and $u_{\mathrm{ILC},k}$, respectively.

Given that $T_{\mathrm{ILC}}(z)$ is stable, and following a standard ILC procedure from [145], T_{iter} is defined as the iteration-domain operator that maps the error from iteration j to $j+1$ as

$$T_{\mathrm{iter}}(z) = T_{\mathrm{ILC}}(z)[1 + L(z)T_{\mathrm{ILC}}(z)]T_{\mathrm{ILC}}^{-1}(z), \tag{5.11}$$

so that

$$E^{(j+1)}(z) = T_{\mathrm{iter}}(z)E^{(j)}(z). \tag{5.12}$$

Convergence follows from the strict contraction of $E^{(j)}$. Thus, if $\rho(T_{\mathrm{iter}}) < 1$, then $E^{(j)} \to 0$ as $j \to \infty$.

Remark 20. *Although the analysis in this section employs a reduced-order dynamic model to represent the closed-loop dynamics of the original system in* (3.33)*, it does not alter the original NMPC closed loop. This reduction facilitates the analysis of the proposed ILC's convergence when operating alongside NMPC.*

5.6 ILC Implementation

The implementation of ILC for ground target tracking considers the use of a re-parametrization of the input u_{ILC}^{j} based on the angle before considering in iteration $j+1$. Moreover, the combination of the proposed ILC with NMPC is made by considering u_{ILC}^{j} as a known disturbance in the optimization problem.

5.6.1 ϕ-based parametrization

The convergence analysis presented in Section 5.5.3 assumes that the initial conditions of the system are the same between j iterations. To ensure that this condition holds when loitering the target, the generated ILC control input for next iteration is parameterized based on the angle ϕ between the ground target and the UAV, rather than on discrete time steps k. Specifically, after computing the input vector, the input is defined as a function of ϕ_k as $u_{\mathrm{ILC}}^{(j+1)}(\phi_k)$. Although the proposed ILC scheme is ultimately implemented by parametrizing the time index k in terms of a phase variable ϕ_k, the convergence analysis presented in Section 5.5.3 remains valid. The phase-based parametrization simply re-labels the time axis but does not alter the mapping between the error signals from iteration j to $j+1$. In other words, the iteration operator T_{iter} itself is invariant to this time-index transformation, and thus the derived condition $\rho(T_{\mathrm{iter}}) < 1$ guaranteeing convergence is unaffected.

Algorithm 3 ILC Algorithm for 3D Ground Target Tracking

Set initial learning iteration $j \leftarrow 1$.
Initialize ILC control input $u_{\text{ILC}}^{(1)}(\phi_n) \leftarrow 0$ for all ϕ_n.
Initialize state vector $x_0^{(1)}$ with initial conditions.
Define reference distance d_{ss}.
Initialize learning gains L_{KP} and L_{KD}.
repeat
    Initialize empty functions for $u_{\text{ILC}}^{(j)}(\phi_n)$ and $e^{(j)}(\phi_n)$ for all ϕ_n.
    Set time $k \leftarrow 0$.
    while orbit not completed **do**
        Measure current state $x_k^{(j)}$.
        Determine Phase Angle ϕ_k.
        Retrieve ILC Correction $u_{\text{ILC}}^{(j)}(\phi_k)$.
        $\delta u \leftarrow -u_{\text{ILC}}(\phi_k)$.
        Solve ILC optimization problem.
        Send $u_{\psi,k}^{(j)}$ to UAV actuators.
        UAV updates state to $x_k^{(j)}$
        $e^{(j)}(k) \leftarrow d_k^{(j)} - d_{ss}$.
        Increment time $k \leftarrow k + 1$.
    end while
    for each discrete phase angle ϕ_n **do**
        $u_{\text{ILC}}^{(j+1)}(\phi_k) \leftarrow u_{\text{ILC}}^{(j)}(\phi_k) + L(z)e^{(j)}(\phi_k)$.
    end for
    Increment iteration index $j \leftarrow j + 1$.
until

5.6.2 Known input disturbance

Traditionally, an ILC implementation is made by adding the ILC correction input into the system. In this particular system, that would be that, at iteration $j + 1$, ILC would modify the NMPC input by adding the ILC corrections.

The simplicity of this approach makes it straightforward to implement, which is based on adding to the NMPC generated input the ILC correction u_{ILC} after the optimization process involved at solving (3.27). The disadvantage of this approach is that the optimizer has no control over the resulting total input applied to the system, which could potentially lead to exceed constraints, loss of optimality, among other issues.

To consider the ILC correction into the optimization problem, considering Euler's forward method for discretization, (3.17b) is expressed in discrete-time to account for an induced

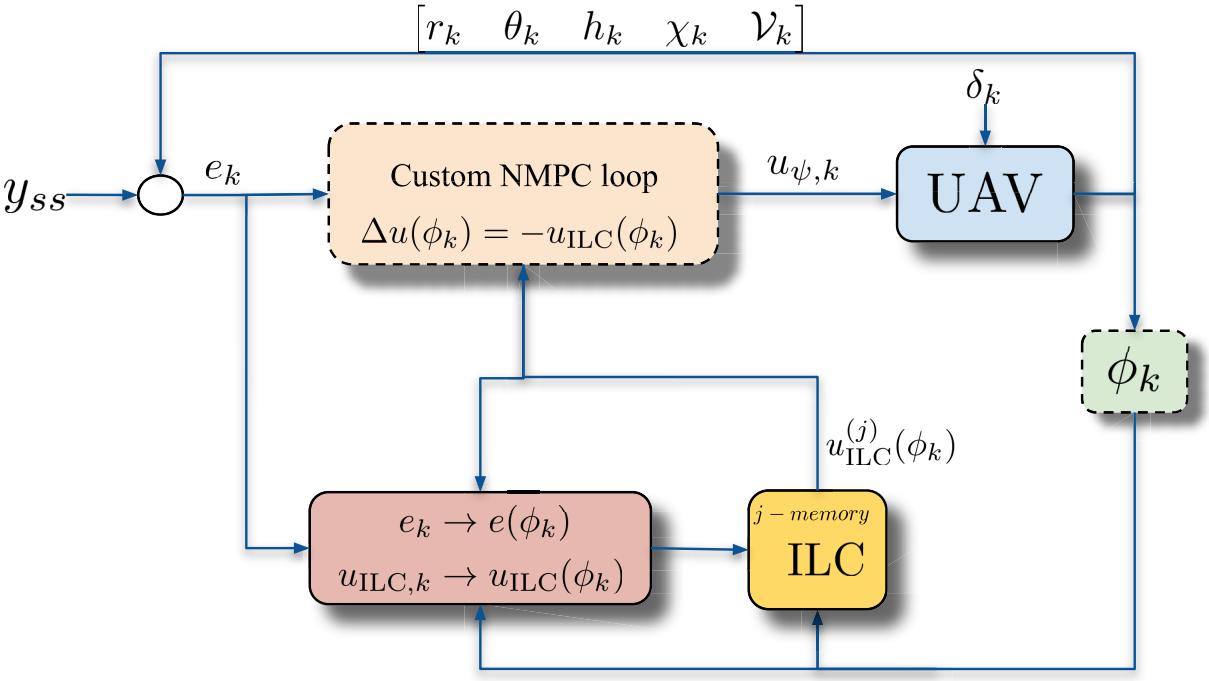


Figure 5.3: Proposed control scheme.

input disturbance δu as

$$\theta_{k+1} = \theta_k + T_s\left(\frac{\mathcal{V}_k \sin\theta_k \cos\chi_k}{r_k} + u_{\psi,k} + \delta u\right). \tag{5.13}$$

For the presented approach, that known displacement is defined as

$$\delta u = -u_{\mathrm{ILC}}(\phi_k). \tag{5.14}$$

After computing the corresponding $u_{\mathrm{ILC}}(\phi_k)$ that will be applied to correct $e^{(j)}(\phi_k)$, rather than directly adding it to the input from the NMPC $u_{\psi,k}$, this is fed into the optimization problem as a known input disturbance in the cost function. Furthermore, since $u_{\mathrm{ILC}}(\phi_k)$ is computed for the whole trajectory of the UAV around the target, it is possible to consider future values of this disturbance during optimization problem along the prediction horizon N.

The key idea of doing this is not only to avoid constraint violation if the input is applied directly, but also to let the NMPC see the true commanded input applied to the system and its evolution over N.

By doing this, any hard or soft constraint regarded to the optimization problem can then remain intact. Moreover, this approach does not break optimality and enhances the NMPC loop by incorporating learning capabilities. This formulation is presented in Fig. 5.3.

In contrast, the ILC correction passed to the optimizer $\Delta u(\phi_k)$ is parameterized explicitly by the phase angle ϕ_k, which captures the spatial relationship between the UAV and the

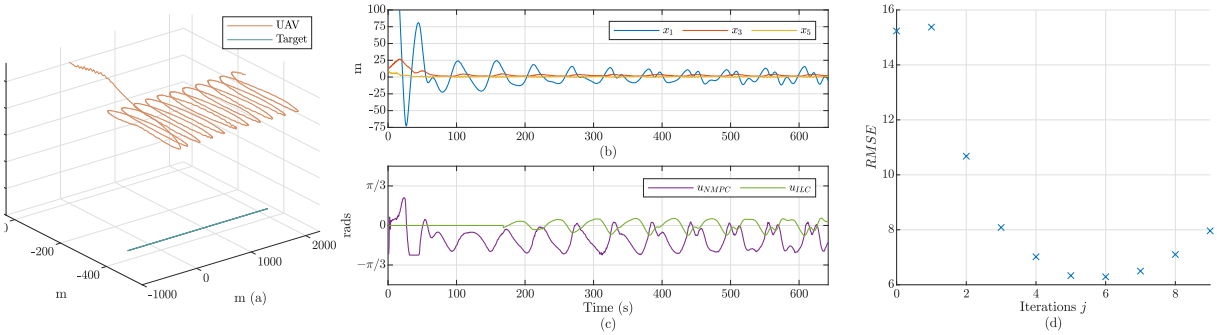


Figure 5.4: Digital-Twin Results P-ILC for Linear Target. (a) Trajectories for UAV and the ground target. (b) Distance, altitude and speed tracking errors. (c) u_{NMPC} and u_{ILC}. (d) RMSE for each iteration j.

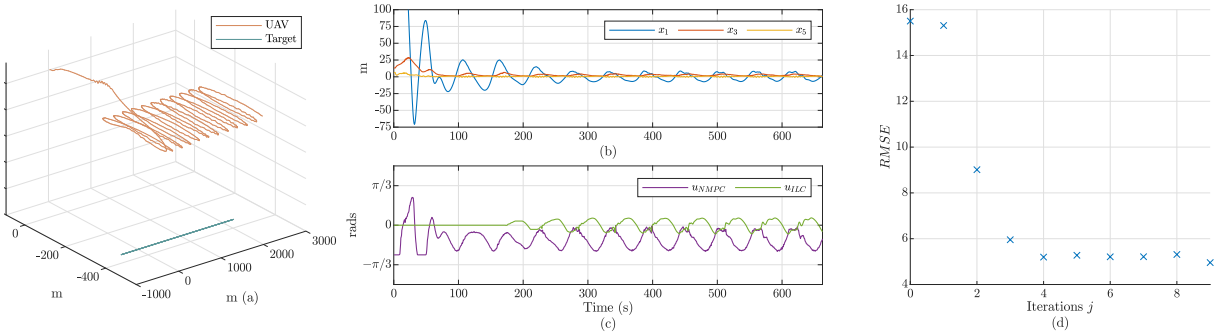


Figure 5.5: Digital-Twin Results NMPC with PD-ILC for Linear Target. (a) Trajectories for UAV and the ground target. (b) Distance, altitude and speed tracking errors. (c) u_{NMPC} and u_{ILC}. (d) RMSE for each iteration j.

ground target. By indexing the ILC updates with respect to ϕ_k, it is ensured that the learning process aligns with the UAV's position relative to the target across iterations. The proposed ILC pseudo-code is presented in Algorithm 3.

5.7 Results

To validate the effectiveness of the proposed approach, results under various disturbance conditions are presented. The controller parameters are listed in Table 5.1 and are chosen to tune the NMPC for stability, following [118]. he weighting factors are set as $Q = \text{diag}(0.037, 4, 0.006, 1, 0.1)$ and $R = \text{diag}(1, 1, 1)$. This results in the local controller (3.35), with K given by

$$K = \begin{bmatrix} 0.0329 & 1.6553 & 0 & 0 & 0.0065 \\ 0 & 0 & 0.0485 & 1.6028 & 0 \\ 0 & 0.0015 & 0 & 0 & 0.2923 \end{bmatrix}. \tag{5.15}$$

From this, $K_{11} = 0.0329$ and $K_{12} = 1.6553$. Then, the poles of $T_\theta(z)$ are equal to $z_{\theta 1} = 0.99645$ and $z_{\theta 2} = 0.1759$, both inside the unit circle. Furthermore, the poles

Table 5.1: System Parameters.

Parameter	**Symbol**	**Value**
Reference distance (m)	d_{ss}	150
Reference bearing angle $(rads)$	θ_{ss}	$\pi/2$
Reference altitude (m)	h_{ss}	85
Reference pitch angle $(rads)$	χ_{ss}	0
UAV reference speed (m/s)	$\mathcal{V}_{ss}$	16
Steady-state yaw input $(rads)$	$u_{\psi,ss}$	$-\mathcal{V}_{ss}/d_{ss}$
Steady-state pitch input $(rads)$	$u_{\chi,ss}$	0
Steady-state speed input	$u_{\mathcal{V},ss}$	0
Sampling time (s)	T_s	0.5
P-Learning rate	L_{KP}	0.00575
D-Learning rate	L_{KD}	0.0033

of T_r are complex conjugates, both located at z_{r12}=0.6257. These results confirm the stability of both closed-loop transfer functions. Finally, for $T_{\text{iter}}(z)$, their poles are conjugates located in $z_{\text{iter12}} = 0.6267$. Thus, since

$$\rho(T_{\text{iter}}(z)) = 0.6267 < 1,$$

the ILC is guaranteed to converge under this configuration.

Two different case studies are considered: the first considers a target with constant movement, and the second involves a real scenario of a vehicle moving along roads at the Warringah Radio Control Society in Belrose, NSW, Australia.

Moreover, for comparison purposes, results are taken using a purely proportional ILC (P-ILC) and a proportional-derivative ILC (PD-ILC).

5.7.1 Case Study 1: Target with Fixed Trajectory

In this case study, both P-ILC and PD-ILC approaches are applied to a target moving along a fixed trajectory. The following comparison highlights the performance differences between the two controllers.

Trajectory Tracking

Figure 5.4(a) shows that under P-ILC, the UAV follows the target's path and maintains the desired relative altitude; moreover, Figure 5.5(a) demonstrates that the PD-ILC enables the UAV to track the target's trajectory, maintaining the desired direction and altitude with fewer deviations. No major differences are present between both approaches for this part.

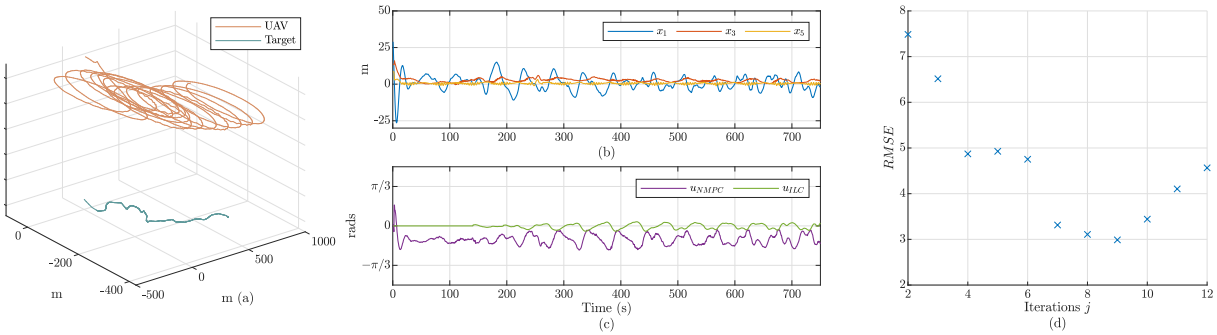


Figure 5.6: Digital-Twin Results NMPC with P-ILC for Variable Trajectory. (a) Trajectories for UAV and the ground target. (b) Distance, altitude and speed tracking errors. (c) u_{NMPC} and u_{ILC}. (d) RMSE for each iteration j.

The tracking errors for distance x_1, altitude x_3, and speed x_5 are compared in Figures 5.4(b) and 5.5(b). For the P-ILC, the error in x_1 is significantly reduced by approximately the sixth iteration, after which the controller overcompensates, leading to oscillations. On the other hand, the PD-ILC not only reduces the x_1 error more rapidly (achieving a substantial reduction within just three iterations) but also markedly diminishes the oscillatory behavior. Although zero tracking error is not fully achieved, the maximum deviation reduces from about 25 m (P-ILC) to roughly 5 m at steady-state for PD-ILC, highlighting the faster compensation for the controller when considering the derivative term.

Moreover, Figs. 5.4(c) and 5.4(d) illustrate the evolution of the control inputs and the RMSE for the P-ILC. The oscillatory behavior in both the control inputs and the RMSE. Specifically, Fig. 5.4(d) shows that the RMSE begins constantly rising after iteration $j = 7$, confirming that the proportional correction introduces sustained oscillations. In contrast, Figs. 5.5(c) and 5.5(d) show that the PD-ILC quickly stabilizes the control inputs after iteration 2 and maintains the RMSE at approximately 5 m from iteration 4 onward. These findings clearly emphasize the advantage of incorporating a derivative term to mitigate oscillations and ensure faster convergence to a steady operating regime.

5.7.2 Case Study 2: Target with Variable Trajectory

The second case study examines the performance of both controllers when the target follows a variable and unknown trajectory. The results are compared below.

The trajectories for the UAV and target on both cases are depicted in Figures 5.6(a) and 5.7(a). With both approaches, it is possible to see that the loitering direction is achieved. Moreover, no major oscillations in the trajectory are present.

The tracking error in x_1 is depicted in Figures 5.6(b) and 5.7(b). With P-ILC, initial iterations reduce the x_1 error, but after 600 s, significant oscillations in the trajectory

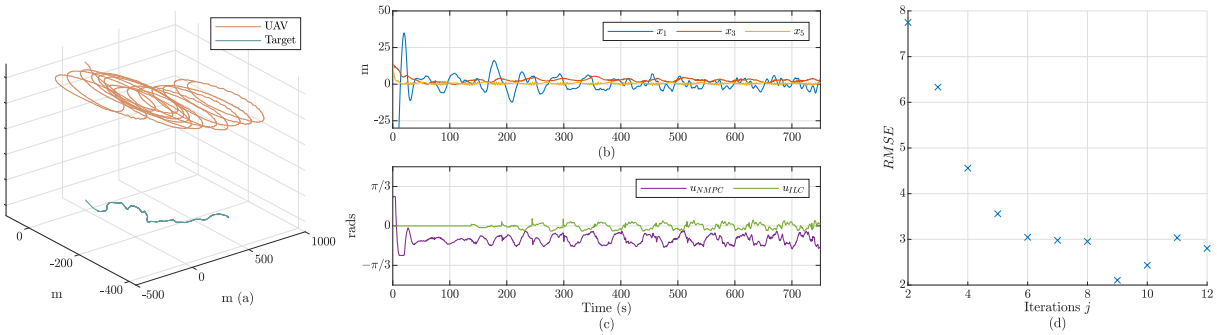


Figure 5.7: Digital-Twin Results NMPC with PD-ILC for Variable Trajectory. (a) Trajectories for UAV and the ground target. (b) Distance, altitude and speed tracking errors. (c) u_{NMPC} and u_{ILC}. (d) RMSE for each iteration j.

become evident as the controller struggles with the target's abrupt maneuvers. Similarly to previous case, the P-ILC approach induces oscillations into the system at the mentioned time mark. Conversely, Fig. 5.7(b) indicates that the PD-ILC maintains accurate tracking even when the target's path changes unexpectedly, preserving accurate tracking and reducing the error for x_1.

Figures 5.6(c) and 5.6(d) for the P-ILC show that sustained oscillations in the control inputs lead to a steady increase in RMSE over the iterations. In comparison, from Fig. 5.7(c), the PD-ILC results in smoother control actions. Furthermore, from Fig. 5.7(d) and a corresponding RMSE that, despite a slight increase between iterations 9 and 11, (from 2 m to 3 m), quickly decreases bellow 4.2 m and stabilizes. On the other hand, Fig. 5.6(d) shows that the P-ILC constantly increases the RMSE after iteration 9, showing the same issue as seen in the fixed trajectory case. This behavior underscores the PD-ILC's enhanced capability to adapt to variable target maneuvers and maintain robust performance.

In summary, the comparison of the two case studies shows that while the P-ILC can reduce tracking errors initially, it suffers from persistent oscillations especially under dynamic target movements. The PD-ILC, on the other hand, achieves faster convergence and significantly smoother performance in both fixed and variable trajectory scenarios.

5.8 Conclusion

This chapter presented an ILC-based nonlinear model predictive control approach for controlling a UAV during loitering maneuvers around a ground target. First, an NMPC scheme was developed to handle the 3D ground-target tracking problem. Next, the repetitive behavior arising from the UAV's circumnavigation of the target is identified, and an ILC strategy to compensate for distance errors is considered. A reduced-order dynamic model was employed to analyze the system's behavior when using both ILC and NMPC.

This enabled a transfer function-based closed-loop stability analysis of the system. Under this framework, the convergence of the ILC was studied, ensuring that the tracking error decreases over successive iterations.

The inner NMPC loop utilizes a conventional 3D tracking model, requiring no explicit disturbance information. Meanwhile, the outer ILC loop leverages historical input-output data to iteratively estimate and compensate for a known input disturbance δ_u. This data-driven technique enables the system to handle unknown disturbances without requiring explicit disturbance modeling.

By integrating u_{ILC} directly into the NMPC optimization problem, the proposed framework ensures that input and state constraints are satisfied, thereby preserving both optimality and feasibility. Additionally, the convergence of the proposed approach was analyzed under system parameters, guaranteeing closed-loop stability and ILC convergence.

Finally, the proposed PD-ILC approach was compared with a purely P-ILC strategy. Results confirmed that the proposed method rapidly achieved zero steady-state error within only a few iterations, demonstrating its effectiveness and robustness in practical UAV loitering and tracking scenarios.

Chapter 6

Conclusions and Future Work

This thesis provides a comprehensive analysis and application of advanced optimal control techniques for ground target tracking using fixed-wing UAVs. The work is structured to build progressively: beginning with a basic two-dimensional target tracking scenario using Nonlinear Model Predictive Control and advancing to the development of a data-driven approach for rejecting unmodeled disturbances. Each chapter builds on its predecessor, contributing to a coherent narrative that enhances the robustness and applicability of the proposed methods.

In this concluding chapter, the key contributions of the thesis in the field of ground target tracking are summarized. The limitations encountered during the research are discussed and offer guidelines for future work. Finally, the thesis conclusion is presented.

6.1 Research Contributions

The significant contributions of this work are depicted as follows:

6.1.1 NMPC-based 3D Tracking and Stability Analysis

The NMPC approach for ground target tracking has proven to be stable under certain conditions and applicable in real-time with an experimental UAV platform. Key contributions presented in Chapter 3 include:

- Developed a complete 3D nonlinear kinematic model for ground target tracking, characterized by five states and three inputs, specifically tailored for fixed-wing UAV applications.
- Designed an NMPC strategy that ensures closed-loop stability in ground target tracking by establishing sufficient conditions in the Lyapunov framework.

- Defined stability domains corresponding to various weighting factors across different operational conditions.
- Introduced an initialization technique for solving the optimization problem, maintaining stability even in cases of suboptimality.
- Experimental validation the proposed approach using a fixed-wing UAV.

6.1.2 Bilinear Kalman Filter for Disturbance Estimation

In Chapter 4, the introduction of a bilinear Kalman filter represents a significant step forward in target tracking accuracy. This approach explicitly accounts for exogenous disturbances and the target's movement, offering the following advantages:

- Developed a dynamic disturbance model for ground target tracking that incorporates nonlinear UAV behavior, accounting for target motion, wind effects, and potential actuator displacements.
- Leveraged kinematic principles to enable disturbance estimation without the need for an airspeed sensor, setting the proposed approach apart from others.
- Achieving offset-free tracking for distance.

6.1.3 Data-Driven Compensation with Iterative Learning Control

Chapter 5 explores data-driven compensation techniques as an alternative to explicit model-based approaches. Instead of relying on a detailed system model, this method leverages input/output data to estimate corrections, significantly reducing error. The core contributions of this approach include:

- Developed a phase-based ILC scheme that refines control actions based on the UAV's orbital phase, effectively compensating for recurring disturbances without relying on explicit disturbance models.
- Conducted a convergence analysis that exploits the closed-loop NMPC system's behavior near the reference, ensuring effective ILC implementation and guaranteeing error contraction across iterations.
- Integrated the ILC scheme with the NMPC controller by incorporating the ILC-generated input as a known disturbance during the optimization stage, creating a hybrid control architecture that combines NMPC with data-driven learning.

6.2 Future Research

The following topics are proposed to further improve the tracking of ground targets using fixed-wing UAVs:

- **Long Prediction Horizons:** In non-convex optimization, extending the prediction horizon makes finding the optimal solution more challenging, especially when the system model exhibits strong nonlinearities. One suggestion is to implement machine learning techniques to mimic the behavior of the nonlinear system. A well-trained neural network may replicate the system's nonlinearities and identify patterns when increasing the prediction horizon, ultimately reducing the computation time for the non-convex optimizer.
- **Physics-Informed Neural Networks:** Using a reduced form of the nonlinear system to initialize the controller could help neural networks reduce their training time and provide accurate estimates of the nonlinearities or disturbances induced in the system model during operation. By explicitly incorporating the system model into the neural network, it is possible to narrow the search space to models that better fit the real-world application.
- **Energy Consumption Optimization:** It may be desirable to implement techniques that minimize the UAV's speed when tracking slow-moving targets, thereby optimizing power consumption and increasing tracking duration. This integration of kinematics and power consumption considerations can be implemented within a single cost function, allowing the controller to be tuned to achieve the desired outcome.

6.3 Thesis Conclusions

The NMPC approach for ground target tracking has proven to be stable under certain conditions. Advances in compact hardware capabilities, such as the NVIDIA Jetson, enable non-convex optimization to be applied in real-world scenarios. In this particular case, this allows the use of a companion computer alongside the flight controller, facilitating the implementation of complex algorithms in real time.

In Chapter 3, the superiority of this approach is demonstrated. Furthermore, sufficient stability conditions are provided, and a stability domain is established for different speeds at various weighting factors, guiding the implementation of NMPC for other target speeds. The kinematic approach allows flexibility in implementation, relegating aerodynamics to the traditional flight controller. Additionally, comparisons and sensitivity analyses for the sampling time and prediction horizon are provided, showing that the implemented configuration yields accurate results for target tracking.

In Chapter 4, a bilinear Kalman filter approach is used to improve target tracking accuracy by explicitly considering exogenous disturbances and the target's movement. This new formulation also allows the combination of disturbances into a single term, enhancing the convergence of the observer. The bilinear nature of the approach does not diminish its capacity to accurately estimate disturbances. The nonlinearities of the controller are considered in the time-variant system; thus, this method does not lose information by linearizing the system or making simplifying assumptions.

Finally, Chapter 5 explores the capabilities of a data-driven compensation technique based on ILC. Rather than using an explicit model, the proposal relies in input/output data to estimate and compute corrections in the input that reduce the error. This iterative process is encoded in an phase-based ILC, which provides the controller with spatial context rather than correcting based solely on time, a strategy that may result in over-corrections or applying inputs where they are not needed.

Validation of these approaches is shown through simulations and experimental results, demonstrating the capabilities of the proposed methods. Moreover, the proposed technique achieves highly accurate tracking with virtually zero tracking error.

In conclusion, the hypothesis for this thesis is proven true through the implementation of the proposal, requiring minimal customization of components and utilizing hardware readily available on the market. This strategy outperforms traditional approaches for target tracking, maintaining minimal tracking error even when faced with highly maneuvering targets. This approach represents a viable alternative for governing a fixed-wing UAV without compromising flexibility and modularity.

Bibliography

[1] P. Zhu et al., "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7380–7399, 2022. DOI: 10.1109/TPAMI.2021.3119563.

[2] P. McEnroe, S. Wang, and M. Liyanage, "A survey on the convergence of edge computing and ai for uavs: Opportunities and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 15 435–15 459, 2022. DOI: 10.1109/JIOT.2022.3176400.

[3] S. Ponte, G. Ariante, U. Papa, and G. Del Core, "An embedded platform for positioning and obstacle detection for small unmanned aerial vehicles," *Electronics*, vol. 9, no. 7, 2020, ISSN: 2079-9292. DOI: 10.3390/electronics9071175. [Online]. Available: https://www.mdpi.com/2079-9292/9/7/1175.

[4] J. Doornbos, K. E. Bennin, Ö. Babur, and J. Valente, "Drone technologies: A tertiary systematic literature review on a decade of improvements," *IEEE Access*, vol. 12, pp. 23 220–23 239, 2024. DOI: 10.1109/ACCESS.2024.3364676.

[5] R. Kellermann, T. Biehle, and L. Fischer, "Drones for parcel and passenger transportation: A literature review," *Transportation Research Interdisciplinary Perspectives*, vol. 4, p. 100 088, 2020, ISSN: 2590-1982. DOI: https://doi.org/10.1016/j.trip.2019.100088. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590198219300879.

[6] A. Rejeb, A. Abdollahi, K. Rejeb, and H. Treiblmaier, "Drones in agriculture: A review and bibliometric analysis," *Computers and Electronics in Agriculture*, vol. 198, p. 107 017, 2022, ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2022.107017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169922003349.

[7] Y. Inoue, "Satellite- and drone-based remote sensing of crops and soils for smart farming – a review," *Soil Science and Plant Nutrition*, vol. 66, no. 6, pp. 798–810, 2020. DOI: 10.1080/00380768.2020.1738899. eprint: https://doi.org/10.1080/00380768.2020.1738899. [Online]. Available: https://doi.org/10.1080/00380768.2020.1738899.

[8] L. Zhang, H. Zhang, Y. Niu, and W. Han, "Mapping maize water stress based on uav multispectral remote sensing," *Remote Sensing*, vol. 11, no. 6, 2019, ISSN:

2072-4292. DOI: 10.3390/rs11060605. [Online]. Available: https://www.mdpi.com/2072-4292/11/6/605.

[9] S. Partheepan, F. Sanati, and J. Hassan, "Autonomous unmanned aerial vehicles in bushfire management: Challenges and opportunities," *Drones*, vol. 7, no. 1, 2023, ISSN: 2504-446X. DOI: 10.3390/drones7010047. [Online]. Available: https://www.mdpi.com/2504-446X/7/1/47.

[10] A. Gohari, A. B. Ahmad, R. B. A. Rahim, A. S. M. Supa'at, S. Abd Razak, and M. S. M. Gismalla, "Involvement of surveillance drones in smart cities: A systematic review," *IEEE Access*, vol. 10, pp. 56 611–56 628, 2022. DOI: 10.1109/ACCESS.2022.3177904.

[11] J. J. Roldan, P. Garcia-Aunon, E. Pena-Tapia, and A. Barrientos, "Swarmcity project: Can an aerial swarm monitor traffic in a smart city?" In *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2019, pp. 862–867. DOI: 10.1109/PERCOMW.2019.8730677.

[12] P. Garcia-Aunon, J. J. Roldán, and A. Barrientos, "Monitoring traffic in future cities with aerial swarms: Developing and optimizing a behavior-based surveillance algorithm," *Cognitive Systems Research*, vol. 54, pp. 273–286, 2019, ISSN: 1389-0417. DOI: https://doi.org/10.1016/j.cogsys.2018.10.031. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389041718303279.

[13] O. Olayiwola, M. Elsden, and M. Dhimish, "Robotics, artificial intelligence, and drones in solar photovoltaic energy applications—safe autonomy perspective," *Safety*, vol. 10, no. 1, 2024, ISSN: 2313-576X. DOI: 10.3390/safety10010032. [Online]. Available: https://www.mdpi.com/2313-576X/10/1/32.

[14] M. Memari, P. Shakya, M. Shekaramiz, A. C. Seibi, and M. A. S. Masoum, "Review on the advancements in wind turbine blade inspection: Integrating drone and deep learning technologies for enhanced defect detection," *IEEE Access*, vol. 12, pp. 33 236–33 282, 2024. DOI: 10.1109/ACCESS.2024.3371493.

[15] H. Ren, Y. Zhao, W. Xiao, and Z. Hu, "A review of uav monitoring in mining areas: Current status and future perspectives," *International Journal of Coal Science & Technology*, vol. 6, no. 3, pp. 320–333, Sep. 2019, ISSN: 2198-7823. DOI: 10.1007/s40789-019-00264-5. [Online]. Available: https://doi.org/10.1007/s40789-019-00264-5.

[16] K. Chávez and O. Swed, "Emulating underdogs: Tactical drones in the russia-ukraine war," *Contemporary Security Policy*, vol. 44, no. 4, pp. 592–605, 2023. DOI: 10.1080/13523260.2023.2257964. eprint: https://doi.org/10.1080/13523260.2023.2257964. [Online]. Available: https://doi.org/10.1080/13523260.2023.2257964.

[17] M. Ayamga, S. Akaba, and A. A. Nyaaba, "Multifaceted applicability of drones: A review," *Technological Forecasting and Social Change*, vol. 167, p. 120 677, 2021, ISSN: 0040-1625. DOI: https://doi.org/10.1016/j.techfore.2021.120677. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0040162521001098.

[18] T. Tengis and A. Batmunkh, "State feedback control simulation of quadcopter model," in *2016 11th International Forum on Strategic Technology (IFOST)*, 2016, pp. 553–557. DOI: 10.1109/IFOST.2016.7884178.

[19] O. Dukkanci, J. F. Campbell, and B. Y. Kara, "Facility location decisions for drone delivery: A literature review," *European Journal of Operational Research*, vol. 316, no. 2, pp. 397–418, 2024, ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2023.10.036. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221723008123.

[20] L. Abualigah, A. Diabat, P. Sumari, and A. H. Gandomi, "Applications, deployments, and integration of internet of drones (iod): A review," *IEEE Sensors Journal*, vol. 21, no. 22, pp. 25 532–25 546, 2021. DOI: 10.1109/JSEN.2021.3114266.

[21] Z. Xu, B. Chen, X. Zhan, Y. Xiu, C. Suzuki, and K. Shimada, "A vision-based autonomous uav inspection framework for unknown tunnel construction sites with dynamic obstacles," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4983–4990, 2023. DOI: 10.1109/LRA.2023.3290415.

[22] L. Nguyen, T. Le, and Q. Ha, "Grey wolf optimization-based path planning for unmanned aerial vehicles in bridge inspection," in *2024 IEEE/SICE International Symposium on System Integration (SII)*, 2024, pp. 810–815. DOI: 10.1109/SII58957.2024.10417594.

[23] S. Oğuz et al., "An open-source uav platform for swarm robotics research: Using cooperative sensor fusion for inter-robot tracking," *IEEE Access*, vol. 12, pp. 43 378–43 395, 2024. DOI: 10.1109/ACCESS.2024.3378607.

[24] L. V. Nguyen, I. Torres, T. H. Le, M. Phung, R. Aguilera, and Q. Ha, "Stag hunt game-based approach for cooperative UAVs," in *2022 Int. Symp. Automation and Robotics in Construction, Bogota, Colombia, 12-15 JUL 2022, accepted*, 2022.

[25] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (uavs): Practical aspects, applications, open challenges, security issues, and future trends," *Intell. Serv. Robot.*, vol. 16, no. 1, pp. 109–137, Jan. 2023, ISSN: 1861-2776. DOI: 10.1007/s11370-022-00452-4. [Online]. Available: https://doi.org/10.1007/s11370-022-00452-4.

[26] Y. Zhi, L. Liu, B. Guan, B. Wang, Z. Cheng, and H. Fan, "Distributed robust adaptive formation control of fixed-wing uavs with unknown uncertainties and disturbances," *Aerospace Science and Technology*, vol. 126, p. 107 600, 2022, ISSN: 1270-

9638. DOI: https://doi.org/10.1016/j.ast.2022.107600. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963822002747.

[27] E. Tal, G. Ryou, and S. Karaman, "Aerobatic trajectory generation for a vtol fixed-wing aircraft using differential flatness," *IEEE Transactions on Robotics*, pp. 1–15, 2023. DOI: 10.1109/TRO.2023.3301312.

[28] V. Roberge, M. Tarbouchi, and G. Labonte, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time uav path planning," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, 2013. DOI: 10.1109/TII.2012.2198665.

[29] S. Hu, X. Yuan, W. Ni, X. Wang, and A. Jamalipour, "Visual-based moving target tracking with solar-powered fixed-wing uav: A new learning-based approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 9115–9129, 2024. DOI: 10.1109/TITS.2024.3386105.

[30] C. Bao, Y. Guo, L. Luo, and G. Su, "Design of a fixed-wing uav controller based on adaptive backstepping sliding mode control method," *IEEE Access*, vol. 9, pp. 157 825–157 841, 2021. DOI: 10.1109/ACCESS.2021.3130296.

[31] X. Ji and T. Wang, "Energy minimization for fixed-wing uav assisted full-duplex relaying with bank angle constraint," *IEEE Wireless Communications Letters*, vol. 12, no. 7, pp. 1199–1203, 2023. DOI: 10.1109/LWC.2023.3266434.

[32] S. Kim, H. Cho, and D. Jung, "Circular formation guidance of fixed-wing uavs using mesh network," *IEEE Access*, vol. 10, pp. 115 295–115 306, 2022. DOI: 10.1109/ACCESS.2022.3218673.

[33] S. Hu, W. Ni, X. Wang, and A. Jamalipour, "Disguised tailing and video surveillance with solar-powered fixed-wing unmanned aerial vehicle," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 5, pp. 5507–5518, 2022. DOI: 10.1109/TVT.2022.3157705.

[34] X. Wang, H. Liu, Y. Tian, Z. Chen, and Z. Cai, "A fast optimization method of water-dropping scheme for fixed-wing firefighting aircraft," *IEEE Access*, vol. 9, pp. 120 815–120 832, 2021. DOI: 10.1109/ACCESS.2021.3106538.

[35] I. Torres, L. V. Nguyen, T. Le, R. P. Aguilera, and Q. Ha, "Uav target tracking using nonlinear model predictive control," in *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 2022, pp. 1–7. DOI: 10.1109/ICECET55527.2022.9873035.

[36] C. Yan, C. Wang, X. Xiang, Z. Lan, and Y. Jiang, "Deep reinforcement learning of collision-free flocking policies for multiple fixed-wing uavs using local situation maps," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1260–1270, 2022. DOI: 10.1109/TII.2021.3094207.

[37] M. Tan, A. Tang, D. Ding, L. Xie, and C. Huang, "Autonomous air combat maneuvering decision method of ucav based on lshade-tso-mpc under enemy tra-

jectory prediction," *Electronics*, vol. 11, no. 20, 2022, ISSN: 2079-9292. DOI: 10.3390/electronics11203383. [Online]. Available: https://www.mdpi.com/2079-9292/11/20/3383.

[38] X. Du, L. Song, Y. Lv, and S. Qiu, "A lightweight military target detection algorithm based on improved yolov5," *Electronics*, vol. 11, no. 20, 2022, ISSN: 2079-9292. DOI: 10.3390/electronics11203263. [Online]. Available: https://www.mdpi.com/2079-9292/11/20/3263.

[39] Y. Zhou, H. Zhao, and Y. Liu, "An evaluative review of the vtol technologies for unmanned and manned aerial vehicles," *Computer Communications*, vol. 149, pp. 356–369, 2020, ISSN: 0140-3664. DOI: https://doi.org/10.1016/j.comcom.2019.10.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S014036641930996X.

[40] G. J. Ducard and M. Allenspach, "Review of designs and flight control techniques of hybrid and convertible vtol uavs," *Aerospace Science and Technology*, vol. 118, p. 107 035, 2021, ISSN: 1270-9638. DOI: https://doi.org/10.1016/j.ast.2021.107035. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963821005459.

[41] L. Pugi, A. Mela, A. Reatti, A. Casazza, R. Fiorenzani, and G. Mattei, "A fixed wing uav with vtol capabilities: Design, control and energy management," *International Journal of Modelling, Identification and Control*, vol. 41, no. 3, pp. 206–221, 2022. DOI: 10.1504/IJMIC.2022.127521. eprint: https://www.inderscienceonline.com/doi/pdf/10.1504/IJMIC.2022.127521. [Online]. Available: https://www.inderscienceonline.com/doi/abs/10.1504/IJMIC.2022.127521.

[42] H. Long, P. Zhang, T. Guo, and J. Zhao, "Fixed-time adaptive fuzzy fault-tolerant control of flapping wing mavs with wing damage," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 5, pp. 6594–6607, 2024. DOI: 10.1109/TAES.2024.3411591.

[43] J.-M. Fahmi and C. A. Woolsey, "Port-hamiltonian flight control of a fixed-wing aircraft," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 1, pp. 408–415, 2022. DOI: 10.1109/TCST.2021.3059928.

[44] F. Liu, Z. Wei, and G. Zhang, "An off-board vision system for relative attitude measurement of aircraft," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 4225–4233, 2022.

[45] A. Israr, E. H. Alkhammash, and M. Hadjouni, "Guidance, navigation, and control for fixed-wing uav," *Mathematical Problems in Engineering*, vol. 2021, no. 1, p. 4 355 253, 2021. DOI: https://doi.org/10.1155/2021/4355253. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2021/4355253. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2021/4355253.

[46] B. Zhang, X. Sun, M. Lv, S. Liu, and L. Li, "Distributed adaptive fixed-time fault-tolerant control for multiple 6-dof uavs with full-state constraints guarantee," *IEEE Systems Journal*, vol. 16, no. 3, pp. 4792–4803, 2022. DOI: 10.1109/JSYST.2021.3128973.

[47] Y. A. Admas, H. M. Mitiku, A. O. Salau, C. O. Omeje, and S. L. Braide, "Control of a fixed wing unmanned aerial vehicle using a higher-order sliding mode controller and non-linear pid controller," *Scientific Reports*, vol. 14, no. 1, p. 23 139, Oct. 2024, ISSN: 2045-2322. DOI: 10.1038/s41598-024-73901-y. [Online]. Available: https://doi.org/10.1038/s41598-024-73901-y.

[48] Z. Zhang et al., "Small fixed-wing unmanned aerial vehicle path following under low altitude wind shear disturbance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 10, pp. 13 991–14 003, 2024. DOI: 10.1109/TITS.2024.3391869.

[49] Y. Wang, M. Shan, and D. Wang, "Motion capability analysis for multiple fixed-wing uav formations with speed and heading rate constraints," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 2, pp. 977–989, 2020. DOI: 10.1109/TCNS.2019.2929658.

[50] P. Zhang, Y. Mei, H. Wang, W. Wang, and J. Liu, "Collision-free trajectory planning for uavs based on sequential convex programming," *Aerospace Science and Technology*, vol. 152, p. 109 404, 2024, ISSN: 1270-9638. DOI: https://doi.org/10.1016/j.ast.2024.109404. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963824005352.

[51] I. Lugo-Cárdenas, S. Salazar, and R. Lozano, "Lyapunov based 3d path following kinematic controller for a fixed wing uav**research supported in part by the mexican national council for science and technology (conacyt) under grant 218857," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 15 946–15 951, 2017, 20th IFAC World Congress, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2017.08.1747. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896317323637.

[52] A. Bouguettaya, H. Zarzour, A. Kechida, and A. M. Taberkit, "Vehicle detection from uav imagery with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6047–6067, 2022. DOI: 10.1109/TNNLS.2021.3080276.

[53] B. Li and Y. Wu, "Path planning for uav ground target tracking via deep reinforcement learning," *IEEE Access*, vol. 8, pp. 29 064–29 074, 2020. DOI: 10.1109/ACCESS.2020.2971780.

[54] J. Ding, L. Wen, C. Zhong, and O. Loffeld, "Video sar moving target indication using deep neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 10, pp. 7194–7204, 2020. DOI: 10.1109/TGRS.2020.2980419.

[55] H. Yan et al., "A new method of video SAR ground moving target detection and tracking based on the interframe amplitude temporal curve," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–17, 2023.

[56] B. Ge, D. An, L. Chen, W. Wang, D. Feng, and Z. Zhou, "Ground moving target detection and trajectory reconstruction methods for multichannel airborne circular SAR," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 4, pp. 2900–2915, 2022.

[57] C. Cao, Y. Zhu, L. Zhao, D. Sun, Z. Liu, and Y. Lian, "An accurate positioning method based on time-division strategy for indoor moving target," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–12, 2023. DOI: 10.1109/TIM.2023.3284938.

[58] Y. Ren, M. Zheng, L. Zhang, H. Fan, and Y. Xie, "An adaptive false target suppression and radial velocity estimation method of moving targets based on image-domain for high-resolution and wide-swath sar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–18, 2024. DOI: 10.1109/TGRS.2024.3397006.

[59] J. Zhang et al., "Mbab-yolo: A modified lightweight architecture for real-time small target detection," *IEEE Access*, vol. 11, pp. 78 384–78 401, 2023. DOI: 10.1109/ACCESS.2023.3286031.

[60] J. Zhang, Y. Meng, and Z. Chen, "A small target detection method based on deep learning with considerate feature and effectively expanded sample size," *IEEE Access*, vol. 9, pp. 96 559–96 572, 2021. DOI: 10.1109/ACCESS.2021.3095405.

[61] B. Qin, D. Zhang, S. Tang, and Y. Xu, "Two-layer formation-containment fault-tolerant control of fixed-wing UAV swarm for dynamic target tracking," *Journal of Systems Engineering and Electronics*, vol. 34, no. 6, pp. 1375–1396, 2023.

[62] Z. Min, Z. Chenming, and H. Kun, "Fixed-wing UAV guidance law for ground target over-flight tracking," *Journal of Systems Engineering and Electronics*, vol. 30, no. 2, pp. 384–392, 2019. DOI: 10.21629/JSEE.2019.02.16.

[63] J. Chen, W. Zhang, S. Wang, Y. Li, and K. Wang, "Ground target guidance method for oriented overhead tracking of fixed-wing UAV," in *2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, 2019, pp. 1–5.

[64] C. Lin, J. Shi, W. Zhang, and Y. Lyu, "Standoff tracking of a ground target based on coordinated turning guidance law," *ISA Transactions*, vol. 119, pp. 118–134, 2022, ISSN: 0019-0578.

[65] X. Song and S. Hu, "2d path planning with dubins-path-based a* algorithm for a fixed-wing uav," in *2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE)*, 2017, pp. 69–73. DOI: 10.1109/CCSSE.2017.8087897.

[66] W. Yang, Z. Shi, and Y. Zhong, "Robust adaptive three-dimensional trajectory tracking control scheme design for small fixed-wing UAVs," *ISA Transactions*, vol. 141, pp. 377–391, 2023, ISSN: 0019-0578.

[67] S. Zhao, F. Yi, Q. Wang, J. Zheng, and X. Wang, "Cooperative tracking of fixed-wing UAVs with arbitrary convergence time: Theory and experiment," *IEEE Transactions on Industrial Informatics*, pp. 1–10, 2024.

[68] S. Park, "Guidance law for standoff tracking of a moving object," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 11, pp. 2948–2955, 2017. DOI: 10.2514/1.G002707. eprint: https://doi.org/10.2514/1.G002707. [Online]. Available: https://doi.org/10.2514/1.G002707.

[69] A. A. Pothen and A. Ratnoo, "Curvature-constrained lyapunov vector field for standoff target tracking," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 10, pp. 2729–2736, 2017. DOI: 10.2514/1.G002281. eprint: https://doi.org/10.2514/1.G002281. [Online]. Available: https://doi.org/10.2514/1.G002281.

[70] Z. Xu, R. Wei, X. Zhao, and S. Wang, "Coordinated standoff target tracking guidance method for uavs," *IEEE Access*, vol. 6, pp. 59 853–59 859, 2018. DOI: 10.1109/ACCESS.2018.2875787.

[71] H. Oh, S. Kim, H.-s. Shin, and A. Tsourdos, "Coordinated standoff tracking of moving target groups using multiple uavs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 1501–1514, 2015. DOI: 10.1109/TAES.2015.140044.

[72] 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, Dec. 2017, ISSN: 1573-0409. DOI: 10.1007/s10846-017-0534-5. [Online]. Available: https://doi.org/10.1007/s10846-017-0534-5.

[73] H. Huang, A. V. Savkin, and W. Ni, "Decentralized navigation of a uav team for collaborative covert eavesdropping on a group of mobile ground nodes," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3932–3941, 2022. DOI: 10.1109/TASE.2021.3139590.

[74] C. Hu, Z. Zhang, Y. Tao, and N. Wang, "Decentralized real-time estimation and tracking for unknown ground moving target using UAVs," *IEEE Access*, vol. 7, pp. 1808–1817, 2019. DOI: 10.1109/ACCESS.2018.2885776.

[75] S. Delalić, A. Alihodžić, M. Tuba, E. Selmanović, and D. Hasić, "Discrete bat algorithm for event planning optimization," in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 1085–1090. DOI: 10.23919/MIPRO48935.2020.9245276.

[76] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, 1942–1948 vol.4. DOI: 10.1109/ICNN.1995.488968.

[77] B. Lindqvist, S. S. Mansouri, A.-a. Agha-mohammadi, and G. Nikolakopoulos, "Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6001–6008, 2020.

[78] L. Bauersfeld, L. Spannagl, G. J. J. Ducard, and C. H. Onder, "MPC flight control for a tilt-rotor VTOL aircraft," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 4, pp. 2395–2409, 2021.

[79] P. M. Gupta, É. Pairet, T. Nascimento, and M. Saska, "Landing a UAV in harsh winds and turbulent open waters," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 744–751, 2023.

[80] D. Huo, L. Dai, R. Chai, R. Xue, and Y. Xia, "Collision-free model predictive trajectory tracking control for UAVs in obstacle environment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 3, pp. 2920–2932, 2023.

[81] L. A. A. Pereira, A. H. D. Nunes, A. M. C. Rezende, V. M. Gonçalves, G. V. Raffo, and L. C. A. Pimenta, "Collision-free vector field guidance and mpc for a fixed-wing uav," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 176–182. DOI: 10.1109/ICRA48506.2021.9560850.

[82] J. Rawlings, D. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, 2017, ISBN: 9780975937730. [Online]. Available: https://books.google.com.au/books?id=MrJctAEACAAJ.

[83] C. Shang, D. Zhang, and Y. Yang, "A gradient-based method for multilevel thresholding," *Expert Systems with Applications*, vol. 175, p. 114 845, 2021, ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2021.114845. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421002864.

[84] S. Sun, K. Dong, C. Guo, and D. Tan, "A wind estimation based on unscented kalman filter for standoff target tracking using a fixed-wing uav," *International Journal of Aeronautical and Space Sciences*, vol. 22, no. 2, pp. 366–375, Apr. 2021, ISSN: 2093-2480. DOI: 10.1007/s42405-020-00290-7. [Online]. Available: https://doi.org/10.1007/s42405-020-00290-7.

[85] N. P. Nguyen, H. Oh, Y. Kim, J. Moon, J. Yang, and W.-H. Chen, "Finite-time disturbance observer-based modified super-twisting algorithm for systems with mismatched disturbances: Application to fixed-wing uavs under wind disturbances," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 15, pp. 7317–7343, 2021. DOI: https://doi.org/10.1002/rnc.5678. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.5678. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.5678.

[86] C. Liu, O. McAree, and W.-H. Chen, "Path-following control for small fixed-wing unmanned aerial vehicles under wind disturbances," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 15, pp. 1682–1698, 2013. DOI: https://doi.org/10.1002/rnc.2938. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.2938. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.2938.

[87] J. Yang, C. Liu, M. Coombes, Y. Yan, and W.-H. Chen, "Optimal path following for small fixed-wing uavs under wind disturbances," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 996–1008, 2021. DOI: 10.1109/TCST.2020.2980727.

[88] Y. Wang, H. Wang, J. Wu, Y. Liu, and Y. Lun, "Uav standoff tracking for narrow-area target in complex environment," *IEEE Systems Journal*, vol. 16, no. 3, pp. 4583–4594, 2022. DOI: 10.1109/JSYST.2021.3093631.

[89] I. Zaitceva and B. Andrievsky, "Methods of intelligent control in mechatronics and robotic engineering: A survey," *Electronics*, vol. 11, no. 15, 2022, ISSN: 2079-9292. DOI: 10.3390/electronics11152443. [Online]. Available: https://www.mdpi.com/2079-9292/11/15/2443.

[90] D. J. Oh, S. G. Baek, K.-T. Nam, and J. C. Koo, "Tracking and synchronization with inversion-based ilc for a multi-actuator-driven wafer inspection cartridge transport robot system," *Electronics*, vol. 10, no. 23, 2021, ISSN: 2079-9292. DOI: 10.3390/electronics10232904. [Online]. Available: https://www.mdpi.com/2079-9292/10/23/2904.

[91] L. V. Nguyen, M. D. Phung, and Q. P. Ha, "Iterative learning sliding mode control for uav trajectory tracking," *Electronics*, vol. 10, no. 20, 2021, ISSN: 2079-9292. DOI: 10.3390/electronics10202474. [Online]. Available: https://www.mdpi.com/2079-9292/10/20/2474.

[92] M. Z. Butt, N. Nasir, and R. B. A. Rashid, "A review of perception sensors, techniques, and hardware architectures for autonomous low-altitude uavs in non-cooperative local obstacle avoidance," *Robotics and Autonomous Systems*, vol. 173, p. 104 629, 2024, ISSN: 0921-8890.

[93] S. Cao, X. Wang, R. Zhang, Y. Peng, and H. Yu, "Aerobatic maneuvering flight control of fixed-wing UAVs: An se(3) approach using dual quaternion," *IEEE Transactions on Industrial Electronics*, pp. 1–11, 2024.

[94] J. Jia, X. Chen, W. Wang, K. Wu, and M. Xie, "Distributed observer-based finite-time control of moving target tracking for UAV formation," *ISA Transactions*, vol. 140, pp. 1–17, 2023, ISSN: 0019-0578.

[95] X. Lu, Z. Li, and J. Xu, "Design and control of a hand-launched fixed-wing unmanned aerial vehicle," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 3, pp. 3006–3016, 2023.

[96] D.-F. He, H. Huang, and Q.-X. Chen, "Stabilizing model predictive control of time-varying non-linear systems using linear matrix inequalities," *IMA Journal of Mathematical Control and Information*, vol. 33, no. 1, pp. 21–35, 2016. DOI: 10.1093/imamci/dnu022.

[97] C. A. T. Tran and Q. Ha, *A Quadratic Constraint Approach to Model Predictive Control of Interconnected Systems*, 1st. Springer Publishing Company, Incorporated, 2018, ISBN: 9789811084072.

[98] Y. Yu, X. Wang, Z. Sun, and L. Shen, "Optimal control of nonlinear systems with unsymmetrical input constraints and its application to the UAV circumnavigation problem," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 5, pp. 2815–2827, 2023.

[99] I. Torres, Q. Ha, and R. Aguilera, "Implementation guidelines of a UAV fixed-wing for advanced real-time control algorithms," in *2024 IEEE/SICE International Symposium on System Integration (SII)*, 2024, pp. 804–809.

[100] I. J. Torres, R. P. Aguilera, and Q. P. Ha, "On the stability of nonlinear model predictive control for 3d target tracking," *IFAC-PapersOnLine*, vol. 58, no. 18, pp. 194–199, 2024, 8th IFAC Conference on Nonlinear Model Predictive Control NMPC 2024, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2024.09.030.

[101] N. T. Nguyen, I. Prodan, and L. Lefèvre, "Stability guarantees for translational thrust-propelled vehicles dynamics through NMPC designs," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, pp. 207–219, 2021.

[102] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 65–74.

[103] D. Limon et al., "Input-to-state stability: A unifying framework for robust model predictive control," in *Nonlinear Model Predictive Control: Towards New Challenging Applications*, L. Magni, D. M. Raimondo, and F. Allgöwer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–26, ISBN: 978-3-642-01094-1.

[104] A. Hashemi, Y. Cao, D. Casbeer, and G. Yin, "Uav circumnavigation of an unknown target without location information using noisy range-based measurements," in *2014 American Control Conference*, 2014, pp. 4587–4592. DOI: 10.1109/ACC.2014.6858643.

[105] J. Zhang, Y. Zeng, and R. Zhang, "Receding horizon optimization for energy-efficient UAV communication," *IEEE Wireless Communications Letters*, vol. 9, no. 4, pp. 490–494, 2020.

[106] R. P. Aguilera, G. Urrutia, R. A. Delgado, D. Dolz, and J. C. Agüero, "Quadratic model predictive control including input cardinality constraints," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 3068–3075, 2017.

[107] S. Subramanian, Y. Abdelsalam, S. Lucia, and S. Engell, "Robust tube-enhanced multi-stage NMPC with stability guarantees," *IEEE Control Systems Letters*, vol. 6, pp. 1112–1117, 2022.

[108] N. Cheng et al., "Ai for uav-assisted iot applications: A comprehensive review," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14 438–14 461, 2023. DOI: 10.1109/JIOT.2023.3268316.

[109] P. Chen, G. Zhang, J. Li, Z. Chang, and Q. Yan, "Path-following control of small fixed-wing uavs under wind disturbance," *Drones*, vol. 7, no. 4, 2023, ISSN: 2504-446X. DOI: 10.3390/drones7040253. [Online]. Available: https://www.mdpi.com/2504-446X/7/4/253.

[110] X. Yu, J. Yang, and S. Li, "Finite-time path following control for small-scale fixed-wing uavs under wind disturbances," *Journal of the Franklin Institute*, vol. 357, no. 12, pp. 7879–7903, 2020, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2020.06.011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S001600322030435X.

[111] X.-H. Nian, W.-X. Zhou, S.-L. Li, and H.-Y. Wu, "2-D path following for fixed wing UAV using global fast terminal sliding mode control," *ISA Transactions*, vol. 136, pp. 162–172, 2023, ISSN: 0019-0578.

[112] T. Oliveira and P. Encarnação, "Ground target tracking for unmanned aerial vehicles," in *AIAA Guidance, Navigation, and Control Conference*. DOI: 10.2514/6.2010-8082. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2010-8082. [Online]. Available: https://arc.aiaa.org/doi/abs/10.2514/6.2010-8082.

[113] J. Kim, "Tracking a ground target utilizing doppler-only measurements of a single passive sonar sensor assisted by straight road constraints," *IEEE Access*, vol. 10, pp. 74 198–74 206, 2022. DOI: 10.1109/ACCESS.2022.3191030.

[114] F. Che, Y. Niu, J. Li, and L. Wu, "Cooperative standoff tracking of moving targets using modified lyapunov vector field guidance," *Applied Sciences*, vol. 10, no. 11, 2020, ISSN: 2076-3417. DOI: 10.3390/app10113709. [Online]. Available: https://www.mdpi.com/2076-3417/10/11/3709.

[115] D. Shin, Y. Song, J. Oh, and H. Oh, "Nonlinear disturbance observer-based standoff target tracking for small fixed-wing uavs," *International Journal of Aeronautical and Space Sciences*, vol. 22, no. 1, pp. 108–119, Feb. 2021, ISSN: 2093-2480. DOI: 10.1007/s42405-020-00275-6. [Online]. Available: https://doi.org/10.1007/s42405-020-00275-6.

[116] H. Oh and S. Kim, "Persistent standoff tracking guidance using constrained particle filter for multiple uavs," *Aerospace Science and Technology*, vol. 84, pp. 257–264,

2019, ISSN: 1270-9638. DOI: https://doi.org/10.1016/j.ast.2018.10.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963817323246.

[117] S. Kim, H. Oh, and A. Tsourdos, "Nonlinear model predictive coordinated standoff tracking of a moving ground vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 2, pp. 557–566, 2013. DOI: 10.2514/1.56254. eprint: https://doi.org/10.2514/1.56254. [Online]. Available: https://doi.org/10.2514/1.56254.

[118] I. J. Torres, R. P. Aguilera, and Q. P. Ha, "Design and performance evaluation of nonlinear model predictive control for 3d ground target tracking with fixed-wing uavs," *IEEE Open Journal of the Industrial Electronics Society*, pp. 1–19, 2024. DOI: 10.1109/OJIES.2024.3519665.

[119] S. Sun, H. Wang, J. Liu, and Y. He, "Fast lyapunov vector field guidance for standoff target tracking based on offline search," *IEEE Access*, vol. 7, pp. 124 797–124 808, 2019. DOI: 10.1109/ACCESS.2019.2932998.

[120] M. Morari and U. Maeder, "Nonlinear offset-free model predictive control," *Automatica*, vol. 48, no. 9, pp. 2059–2067, 2012, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2012.06.038. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109812002932.

[121] Z. Zheng, J. Li, Z. Guan, and Z. Zuo, "Constrained moving path following control for uav with robust control barrier function," *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 7, pp. 1557–1570, 2023. DOI: 10.1109/JAS.2023.123573.

[122] M. Basescu, B. Yeh, L. Scheuer, K. Wolfe, and J. Moore, "Precision post-stall landing using nmpc with learned aerodynamics," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 3031–3038, 2023. DOI: 10.1109/LRA.2023.3264738.

[123] X. Fang, J. Jiang, and W.-H. Chen, "Model predictive control with wind preview for aircraft forced landing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 4, pp. 3995–4004, 2023. DOI: 10.1109/TAES.2023.3235321.

[124] X. Ye, Y. Zeng, Q. Zeng, and Y. Zou, "Airspeed-aided state estimation algorithm of small fixed-wing uavs in gnss-denied environments," *Sensors*, vol. 22, no. 9, 2022, ISSN: 1424-8220. DOI: 10.3390/s22093156. [Online]. Available: https://www.mdpi.com/1424-8220/22/9/3156.

[125] Y. Chen, C. Wang, W. Zeng, and Y. Wu, "Horizontal nonlinear path following guidance law for a small uav with parameter optimized by nmpc," *IEEE Access*, vol. 9, pp. 127 102–127 116, 2021. DOI: 10.1109/ACCESS.2021.3111101.

[126] B. Arain and F. Kendoul, "Real-time wind speed estimation and compensation for improved flight," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 2, pp. 1599–1606, 2014. DOI: 10.1109/TAES.2014.120236.

[127] A. Rautenberg, M. S. Graf, N. Wildmann, A. Platis, and J. Bange, "Reviewing wind measurement approaches for fixed-wing unmanned aircraft," *Atmosphere*,

vol. 9, no. 11, 2018, ISSN: 2073-4433. DOI: 10.3390/atmos9110422. [Online]. Available: https://www.mdpi.com/2073-4433/9/11/422.

[128] B. Meng, K. Zhang, and B. Jiang, "Fixed-time optimal fault-tolerant formation control with prescribed performance for fixed-wing uavs under dual faults," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 9, pp. 875–887, 2023. DOI: 10.1109/TSIPN.2023.3341406.

[129] K. T. Borup, B. N. Stovner, T. I. Fossen, and T. A. Johansen, "Kalman filters for air data system bias correction for a fixed-wing uav," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2164–2176, 2020. DOI: 10.1109/TCST.2019.2931672.

[130] T. Souanef, "$\mathcal{L}_1$ adaptive path-following of small fixed-wing unmanned aerial vehicles in wind," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 4, pp. 3708–3716, 2022. DOI: 10.1109/TAES.2022.3153758.

[131] H. Chen, Y. Cong, X. Wang, X. Xu, and L. Shen, "Coordinated path-following control of fixed-wing unmanned aerial vehicles," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 4, pp. 2540–2554, 2022. DOI: 10.1109/TSMC.2021.3049681.

[132] M. Witczak, V. Puig, D. Rotondo, and P. Witczak, "A necessary and sufficient condition for total observability of discrete-time linear time-varying systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 729–734, 2017, 20th IFAC World Congress, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2017.08.232. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S240589631730544X.

[133] X. Wang, "A cooperative guidance law for UAVs target tracking," *WSEAS TRANSACTIONS on SYSTEMS archive*, vol. 19, pp. 324–335, 2021.

[134] H. M. Jayaweera and S. Hanoun, "A dynamic artificial potential field (d-apf) UAV path planning technique for following ground moving targets," *IEEE Access*, vol. 8, pp. 192 760–192 776, 2020. DOI: 10.1109/ACCESS.2020.3032929.

[135] W. Wu, J. Xu, C. Gong, and N. Cui, "Adaptive path following control for miniature unmanned aerial vehicle confined to three-dimensional dubins path: From take-off to landing," *ISA Transactions*, vol. 143, pp. 156–167, 2023, ISSN: 0019-0578.

[136] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, "Review on model predictive control: An engineering perspective," *The International Journal of Advanced Manufacturing Technology*, vol. 117, no. 5, pp. 1327–1349, Nov. 2021, ISSN: 1433-3015. DOI: 10.1007/s00170-021-07682-3. [Online]. Available: https://doi.org/10.1007/s00170-021-07682-3.

[137] Z. Huang and M. Chen, "Coordinated disturbance observer-based flight control of fixed-wing uav," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 8, pp. 3545–3549, 2022. DOI: 10.1109/TCSII.2022.3165366.

[138] Z. Huang, M. Chen, and P. Shi, "Disturbance utilization-based tracking control for the fixed-wing uav with disturbance estimation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 70, no. 3, pp. 1337–1349, 2023. DOI: 10.1109/TCSI.2022.3229169.

[139] P. Yu, Y. Su, L. Ruan, and T.-C. Tsao, "Compensating aerodynamics of over-actuated multi-rotor aerial platform with data-driven iterative learning control," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6187–6194, 2023. DOI: 10.1109/LRA.2023.3304539.

[140] Z. Liu, X. Yu, W. Lin, and J. J. Rodríguez-Andina, "Iterative learning observer-based high-precision motion control for repetitive motion tasks of linear motor-driven systems," *IEEE Open Journal of the Industrial Electronics Society*, vol. 5, pp. 54–66, 2024. DOI: 10.1109/OJIES.2024.3359951.

[141] D. X. Ba, N. T. Thien, and J. Bae, "A novel iterative second-order neural-network learning control approach for robotic manipulators," *IEEE Access*, vol. 11, pp. 58 318–58 332, 2023. DOI: 10.1109/ACCESS.2023.3280979.

[142] L. Aarnoudse, A. Pavlov, and T. Oomen, "A design framework for nonlinear iterative learning control and repetitive control: Applied to three mechatronic case studies," *Control Engineering Practice*, vol. 149, p. 105 976, 2024, ISSN: 0967-0661. DOI: https://doi.org/10.1016/j.conengprac.2024.105976. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066124001369.

[143] D. Huang, Y. He, W. Yu, N. Qin, Q. Wang, and P. Sun, "Spatial adaptive iterative learning tracking control for high-speed trains considering passing through neutral sections," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 10, pp. 6157–6168, 2023. DOI: 10.1109/TSMC.2023.3279782.

[144] A. Norouzi and C. R. Koch, "Integration of pd-type iterative learning control with adaptive sliding mode control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6213–6218, 2020, 21st IFAC World Congress, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2020.12.1717. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896320323223.

[145] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006. DOI: 10.1109/MCS.2006.1636313.

Appendix A

Appendix

A.1 Bat Algorithm

The Bat Algorithm, introduced by Yang in [102], presents an intriguing heuristic optimization approach inspired by the echolocation capabilities of bats. This method is inspired the bats' ability to use sound pulses to navigate and discern prey and obstacles even in the absence of light. The algorithm formulates echolocation rules as optimization strategies, where bats fly randomly with specific positions $\mathbf{x}_i$ and velocities $\mathbf{v}_i$, emitting sound pulses with adjustable frequency f_{min}, wavelength λ, and loudness A_0 to search for the optimal solution. The Bat Algorithm comprises three main stages:

Initialization

Bats are initially placed at random positions within the search space, and they explore the area by moving randomly to locate potential solutions. Each bat's position $x_{i,j}$ for dimension j is determined using the equation:

$$x_{i,j} = x_{min,j} + rand \cdot (x_{min,j} - x_{max,j}), \tag{A.1}$$

where $i = 1, 2, \cdots, n$, $j = 1, 2, \cdots, d$, and $x_{min,j}$ and $x_{max,j}$ represent the lower and upper boundaries for dimension j, respectively.

Movement of the bats

The position $\mathbf{x}_i$ and velocity $\mathbf{v}_i$ of each bat are influenced by the frequency f_i, which varies between f_{max} and f_{min}. The update equations for the positions and velocities at

time step t are as follows:

$$f_i = f_{min} + (f_{max} - f_{min})\beta, \tag{A.2a}$$

$$\mathbf{v}_i^t = \mathbf{v}^{t-1} i + (\mathbf{x}_i^t - \mathbf{x}) f_i, \tag{A.2b}$$

$$\mathbf{x}_i^t = \mathbf{x}^{t-1} i + \mathbf{v}_i^t, \tag{A.2c}$$

where $\beta \in [0, 1]$ is a random number, and $\mathbf{x}$ represents the current global best solution. Additionally, a new solution is generated locally for each bat by applying a random walk with the equation:

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \epsilon A^t, \tag{A.3}$$

where $\epsilon \in [-1, 1]$ is a random number, and A^t is the average loudness of all bats at time t.

Loudness and pulse emission

The loudness of the bats decreases as they find their prey, while the pulse emission rate increases. This dynamic is updated throughout the iterations with the equations:

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)], \tag{A.4}$$

where α and γ are constants. When $0 < \alpha < 1$ and $\gamma > 0$, $A_i^t \to 0$ and $r_i^t \to r_i^0$ as $t \to \infty$. The loudness and emission rates are only updated when new solutions outperform the previous ones, signifying that the bats are converging towards the optimal solution. For simplicity, the original Bat Algorithm uses $\alpha = \gamma = 0.9$, but these values can be tuned for specific problems.

A.2 HobbyKing Bixler3

A.3 Experimental Platform and General Recommendations

A.3.1 Hardware

In order to create a dependable autonomous aircraft, it is imperative to select hardware components that exhibit reliability and versatility carefully. This selection is crucial for facilitating seamless communication among the different components, ensuring the continuity and efficiency of information exchange.

Fuselage and propulsion

The fuselage is the central housing for all aircraft components, making durability and lightweight construction essential, while the propulsion system generates thrust and lift.

The propulsion system includes a 20A ESC paired with a 2620-1400KK brushless motor, and an 8-inch propeller with a pitch of 40 degrees. This equipment is typically part of standard BNF (Bind and Fly) kits. In this instance, it pertains to the HobbyKing Bixler 3 BNF Kit. The aircraft becomes flight-ready by incorporating a Radio Receiver and battery pack, albeit lacking autonomous capabilities. In Section A.3.1, the supplementary hardware essential for achieving autonomous and intelligent flight is presented.

Flight Controller and peripherals

The flight controller is the first component of additional hardware to provide automation into the UAV. Furthermore, it serves as the central microprocessor that governs the operation of the aircraft. In addition to providing control signals for the motor and control surfaces, the flight controller is responsible for integrating and processing data from various sensors. The flight controller facilitates data logging, real-time telemetry, and signal interpretation. In this study, a the Matek H743-WING V3 as the specific flight controller for the considered fixed-wing UAV.

Global Navigation Satellite System (GNSS)

By utilizing signals from a constellation of satellites, GNSS enables precise determination of the UAV's geographical coordinates, altitude, and ground speed. In addition to navigation, the GNSS module contributes to flight safety by enabling features such as Return-to-Launch (RTL) in the event of communication loss or critical situation. For this research, the Matek M10Q-5883 GNSS and Compass module into the fixed-wing UAV are used.

Radio Receiver

The radio receiver is responsible for receiving commands from the radio controller and transmitting them to the aircraft's flight controller. It interfaces between the pilot's inputs and the UAV's control system. This work incorporates the TBS Crossfire Nano Receiver into the fixed-wing UAV to ensure reliable and long-range operation. This receiver has demonstrated impressive performance, making it suitable for long-range surveillance, target tracking, mapping, or search and rescue missions. It should be noted that this component necessitates connection to a UART port on the flight controller, as opposed to a USART, a distinction that often goes overlooked.

Telemetry

Telemetry enables the remote monitoring and control of various parameters, enhancing situational awareness and operational efficiency. These systems collect data from the UAV that is transmitted wirelessly to a ground control station, where it can be visualized

Figure A.1: Example of OSD for a mission.

and analyzed in real-time. In this proposal, the Holybro SiK Telemetry Radio V3 as the telemetry system for the fixed-wing UAV system is considered.

First Person View (Optional)

A camera and video transmitter are incorporated into the aircraft as an optional component. This addition serves multiple purposes, providing an in-flight perspective from the UAV and leveraging the flight controller's OSD capability. Furthermore, the flight controller's OSD feature overlays real-time information onto the video feed, enhancing the usability and effectiveness of the visual data. This redundancy complements the telemetry system, providing an additional means of receiving real-time information during flight operations. Figure A.1 presents an example of the OSD for a typical mission.

On-board microcomputer

The NVIDIA®Jetson Nano is a compact and powerful micro-computer with significant computational capabilities, particularly its multi-core dedicated Graphic Processing Unit (GPU). The Jetson Nano is specifically designed for classification, object detection, segmentation, and speech-processing applications. This work utilizes the Jetson Nano to efficiently execute computationally demanding algorithm.

By running the algorithms on the GPU, faster execution times is achieved, which enhances the overall performance of the UAV system. The manufacturer provides technical

Table A.1: NVIDIA®Jetson Nano technical specifications

Parameter	Specification
GPU	128-core Maxwell
CPU	Quad-core ARM A57 @ 1.43 GHz
Memory	4 GB 64-bit LPDDR4 25.6 GBs
Video Encode	4x 1080p @ 30 | 9x 720p @ 30 (H.264/H.265)
Video Decode	8x 1080p @ 30 | 18x 720p @ 30 (H.264/H.265)
Camera	2x MIPI CSI-2 DPHY lanes
Connectivity	Gigabit Ethernet, M.2 Key E
Display	HDMI and display port
USB	4x USB 3.0, USB 2.0 Micro-B
Others	GPIO, I2C, I2S, SPI, UART
Dimensions	69 mm x 45 mm, weight 61g with heat-sink

Figure A.2: Experimental UAV Setup.

specifications for the Jetson Nano, which are presented in Table A.1. The GPU implementation of the algorithm is coded in Python3 and employs Numba as a just-in-time (JIT) compiler. Numba translates Python and NumPy code into NVIDIA's Compute Unified Device Architecture (CUDA) code, executed efficiently on the GPU. The experimental setup for testing can be seen in Fig. A.2. When adding extra hardware to fixed-wing UAVs, it's generally advised to mount the equipment without disturbing the aircraft's center of mass. In this approach, the Bixler 3's center of mass is determined through empirical testing, and then positioned the NVIDIA Jetson Nano directly above it. This placement helps ensure that the microcomputer's addition has minimal impact on the aircraft's aerodynamics. This way, the NVIDIA Jetson Nano was placed on top of the wings.

A.3.2 Software

The software running on the aircraft encompasses the firmware on the flight controller and the control algorithm. The firmware running on the flight controller is based on ArduPilot, and it is specific for fixed-wing aircrafts, providing the necessary code and algorithms to manage the UAV's operations effectively. Furthermore, this firmware enables

communication with the NVIDIA Jetson Nano through the MAVLink protocol.

MAVLink communication

MAVLink is a communication protocol between the flight controller and external devices. To implement MAVLink communication in Python, the pymavlink library for Python3 can be utilized. This library streamlines communication, ensuring a robust and dependable link between the flight controller and the NVIDIA Jetson. The communication between the devices is typically accomplished through UART ports.

Parallel programming of a Bat Inspired Heuristic Algorithm running on NVIDIA Jetson Nano

In Table A.1, it is evident that the core strength of the NVIDIA Jetson Nano lies in its GPU, which is equipped with a multi-core NVIDIA Maxwell architecture and boasts 128 CUDA cores. The optimization problem defined in (3.27a) is non-convex, making gradient-based algorithms prone to settling at local minima. To more effectively navigate this problem, the proposes rely on the computational prowess of the GPU for executing non-convex optimization algorithms. While heuristic algorithms have been highlighted in the literature as potent tools for addressing non-convex optimization challenges, they are notably resource-intensive. A prime example is the Bat Algorithm. With its intrinsic parallel processing abilities, the GPU can significantly expedite the Bat Algorithm, leading to swifter execution and more refined search mechanisms. The Bat Algorithm, introduced by Yang in [102], presents an intriguing heuristic optimization approach inspired by the echolocation capabilities of bats. This method is inspired the bats' ability to use sound pulses to navigate and discern prey and obstacles even in the absence of light. The algorithm formulates echolocation rules as optimization strategies, where bats fly randomly with specific positions $\mathbf{x}_i$ and velocities $\mathbf{v}_i$, emitting sound pulses with adjustable frequency f_{min}, wavelength λ, and loudness A_0 to search for the optimal solution.

The key for parallelization lies in the use of Numba. Numba translates Python functions into optimized machine code using LLVM[1] compiler library. For paralellizing an existing Python code, it is needed to apply one of the Numba decorators in Python and Numba will compile it and run it on the GPU.

[1]Originally, LLVM stood for Low Level Virtual Machine; however, LLVM project evolved and, since 2011, LLVM is no longer an acronym.