Drone(s) trajectory optimization for mapping missions

Dora Novak

université PARIS-SACLAY

Drone(s) trajectory optimization for mapping missions

Optimisation de trajectoire de drone(s) pour des missions de cartographie

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n°580 Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat: Automatique
Graduate school: Informatique et Sciences du Numérique. Référent: Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Laboratoire des Signaux et Systèmes** (Université Paris-Saclay, CNRS, CentraleSupélec), sous la direction de **Sihem TEBBANI** (Professeur, CentraleSupélec/L2S).

Thèse soutenue à CentraleSupélec, Paris-Saclay, le 24 octobre 2024, par

Dora NOVAK

Composition du Jury

Membres du jury avec voix délibérative

Sorin OLARU Professeur, CentraleSupélec - Université Paris-Saclay	Président
Nicolas LANGLOIS Professeur, ESIGELEC	Rapporteur & Examinateur
Ionela PRODAN Maître de conférences, HDR, Grenoble INP	Rapporteure & Examinatrice
Pedro CASTILLO Chargé de recherche, CNRS/Université de Technologie de Compiègne	Examinateur
Mohamed DJEMAI Professeur des universités, Université Polytechnique Hauts-de-France	Examinateur

Thèse de doctorat
NNT : 2024UPASG069

"Fais de ta vie un rêve, et d'un rêve, une réalité."

Antoine de Saint-Exupéry

For my grandma, za baku Vinku.

Acknowledgements

I would firstly want to thank all the members of the jury. I would like to thank Nicolas Langlois and Ionela Prodan for agreeing to review my manuscript, as well as for your time and precious suggestions. Sorin Olaru, Mohamed Djemai, and Pedro Castillo - thank you for expressing your interest in my work and valuable discussions during the defense. I want to thank Matko Orsag for hosting me at the LARICS lab in Croatia, for sharing your experience, and for facilitating the experimental validation of my work. I would also like to acknowledge Jurica and the rest of the LARICS team, for the warm welcome and knowledge sharing.

Merci Sihem pour ton encadrement. Merci de m'avoir appris comment valoriser mes travaux et présenter mes résultat dans la manière fluide, intéressante et cohérente. Merci d'avoir été là pour moi dans les situations compliqués et de m'avoir amené au bout de ce voyage transformatif de 3 ans.

I would not have been able to complete this journey with success without my dear family and friends. Thank you for all your love and support, sharing both moments of difficulty and happiness.

To my parents, who have shown their love and faith in me through all my studies. Thank you for all your support in fulfilling my dreams. Thank you for making stressful moments easier for me, with your problem-solving approach and always finding solutions together.

Hvala mama, za svu tvoju ljubav, brigu i podršku. Hvala ti što si mi uvijek ukazivala na ono bitno u životu.

Hvala tata što si uvijek tu za mene. Hvala ti za sva rješenja koja ponudiš tako da se problem ubrzo učini kao da nije ni postojao! Hvala ti što si uvijek imao visoka očekivanja pa sam i zato dospjela do ovog trenutka, što si me inspirirao za obrazovanje i karijeru, bio moj idol i uzor otkad znam za sebe!

To my sister, my biggest and most faithful cheerleader, who taught me all the basics, and gave me a starting point from which it was easy to progress fast and far. Thank you for always being a proud sister!

To my friends - I am beyond grateful for having such wonderful people in my life. To Monika, for

our long-lasting friendship and our little traditions. Thank you for always supporting me and believing in me and my dreams! To Jing, for all our years of friendship, for your unique pieces of advice, and for always pushing me far in all my projects. To my ESTIEMers Mélanie, Pavle and Jaime, for the genuine friendship that is as old as my PhD journey - I have enjoyed all our moments together, serious and less serious conversations, through stress and laughs. To Sara, my fellow PhD student back in Croatia, for sharing the high level of passion and ambition, and always reminding me of my achievements! To Marina and Aleksa, thank you for all the trips, for rediscovering the world of cinema, and for the amazing nights to remember.

To my colleagues from L2S, who quickly became my friends. Joy, an incredible person who made the beginning of my thesis a real joy; Diego, for rediscovering the pleasure of coming to the lab and having supportive colleagues and sharing great moments with great people; Thibault, for linguistic debates, insightful mathematical discussions, and the *French point of view* in solving problems; Costas, a friendship that started with sharing the office for a short time - thank you for all the lessons in theoretical mathematics told as magical stories, and all the help with difficult situations. Not to forget anyone, thank you to all of you for your support and for making my time at L2S memorable!

Merci à tous mes amis français pour votre soutien et tous les moments partagés dans les rires et les larmes. Merci Manon d'avoir été à mes côtés et d'avoir toujours trouvé une solution et partagé nos moments de décompression, Jeanne et Pierre pour le partage de nos passions au-delà de la thèse. Merci Célia pour les moments précieux partagés toujours dans une très belle ambiance.

Finally, I would like to thank my mentors, with a more or less official role - Fernando, Goca and Alexandre, who believed in my competencies and offered me opportunities beyond the academy, and have kept supporting my path. Karine, ma mentore de *Femmes & Sciences*, qui m'a guidé dans mon parcours, mais aussi aidé à comprendre mes qualités personnelles et mes envies professionnelles.

Nonetheless, I want to thank all my teachers and professors, who have supported me and believed in my potential from my youngest age and throughout my formal and informal education. You have been not only a great motivation for learning and growth, but also a true inspiration, that I also want to have a meaningful impact on people and, possibly, this incredible world.

Thank you all from the bottom of my heart.

Résumé en français

L'utilisation des drones dans le contexte de l'agriculture de précision peut optimiser la gestion de l'exploitation et augmenter la productivité agricole tout en protégeant l'environnement. Toutefois, les drones présentent certaines limites qui doivent être prises en compte lors du développement de solutions de sa mission.

Le cadre du problème de la cartographie avec un ou plusieurs drones peut être divisé en deux sous-problèmes : la planification de la mission de cartographie et la pilotage du drone. La première étape définit la trajectoire à suivre pour couvrir la zone d'intérêt de manière efficace compte tenu des limites du drone, tandis que la seconde garantit que le suivi de la trajectoire planifiée est effectué avec succès.

Afin d'augmenter l'efficacité temporelle et d'assurer une mission à faible dépense énergétique, une nouvelle approche pour l'optimisation de la gestion de la batterie du drone lors de la planification de la mission de cartographie est proposée dans ce travail. La stratégie développée optimise l'utilisation des batteries disponibles pour la mission de cartographie en minimisant la distance de vol totale et en réduisant le nombre de remplacements de batteries. La suppression des remplacements de batterie inutiles réduit la durée totale de la mission, mais évite également les cycles redondants de recharge de la batterie.

L'étude des résultats en simulation et la validation expérimentale présentent une stratégie d'optimisation pour la gestion de la batterie des drones dans le cadre d'une planification de mission respectueuse de l'énergie pour les applications cartographiques. L'approche vise à réduire le temps de vol total en incorporant les remplacements de batterie nécessaires et en tenant compte des contraintes énergétiques dues aux capacités limitées des batteries, ainsi qu'en optimisant le choix de la station de base pour les remplacements de batterie. En minimisant les longs vols inutiles à destination et en provenance de la station de base, la stratégie améliore l'efficacité de la planification des missions. En outre, elle évalue stratégiquement les emplacements de base potentiels situés à la frontière de la

zone de cartographie afin d'identifier l'emplacement de base qui minimise la distance de vol globale de la mission pour chaque sous-chemin de drone.

La distribution des points d'intérêt résultant de la planification de la mission représente les sous-chemins de survol d'un drone doté de plusieurs batteries. Afin de suivre la trajectoire planifiée avec une erreur de suivi minimale, une approche de commande prédictive non linéaire pour un suivi de trajectoire robuste est développée. Cette approche est finalement étendue à une mission de cartographie impliquant plusieurs drones coopératifs, afin d'accroître l'efficacité de la mission et de réduire la durée de la cartographie, en particulier pour les grands champs. Dans ce cas, la coordination entre les drones engagés dans la même mission doit également être prise en compte. Garantir une erreur de poursuite minimale et maintenir une distance de sécurité entre les drones afin d'éviter les collisions sont les principaux défis d'une mission de cartographie bien menée avec plusieurs drones. Le MPC non linéaire a montré des résultats prometteurs pour le suivi de trajectoire en temps réel, principalement lorsqu'il est mis en œuvre dans la forme souvent irrégulière du champ qui conduit à une trajectoire de vol non linéaire. L'approche MPC non linéaire proposée traite avec succès le suivi de trajectoire de plusieurs UAV employés dans la même mission tout en gérant l'évitement des collisions. La couche supplémentaire d'attribution des priorités augmente encore l'efficacité de la commande non linéaire proposée, car elle élimine les manœuvres et les déviations inutiles. Il en résulte une meilleure prévisibilité de l'ensemble du système, ainsi que des changements moins brusques dans les trajectoires des drones.

Enfin, les approches proposées dans cette thèse garantissent l'achèvement de la mission de cartographie et sa sécurité, dans le cas d'un système de drone unique ou multiple.

Contents

List of Figures

List of Tables

List of Abbreviations

CPP Coverage Path Planning

DMPC Distributed Model Predictive Control

DOF degree of freedom

ICTs Information and Communications Technologies

ILP Integer linear programming

LQR Linear Quadratic Regulator

MILP mixed-integer linear programming

NDVI Normalized Difference Vegetation Index

NFZ No-Fly Zone

PA Precision Agriculture

PID Proportional-Integral-Derivative

RMSE Root Mean Square Error

RS Remote Sensing

SMC Sliding Mode Control

SoC state-of-charge

UAV Unmanned Aerial Vehicle

UGV Unmanned Ground Vehicle

VTOL vertical take-off and landing

Chapter 1

Introduction

1.1 Unmanned Aerial Vehicles in Precision Agriculture

The challenges emerged from the rapid world population growth are focused on ensuring enough resources, while avoiding threats of climate change. In order to provide food security, conventional agriculture needs to be transformed with improvements and innovation. According to [4] and [5], adopting Information and Communications Technologies (ICTs) can contribute to meeting the increasing need for faster and better solutions when it comes to agriculture. With the crisis in the agricultural sector, implementing ICTs can increase the financial gain for the farmers, while reducing total cost.

Smart farming and precision agriculture aim to support agriculture in facing challenges to meet the growing demand by using technology and innovation. While smart farming designates a general concept for adopting ICTs in order to increase efficiency in farming activities, Precision Agriculture (PA) focuses more closely on site-specific crop management [5]. Precision agriculture considers spatial variation within a field by leveraging geo-referenced information about relevant characteristics for crop production by means of novel technologies. Processing accurate and reliable data obtained with ICTs can provide insights considering specific field properties. Thus, processes for crop management can be planned in a more efficient way to account for crop yield optimization [6].

Alongside the agricultural robots, the evolution of smart farming and precision agriculture implies extensive development of Unmanned Aerial Vehicles (UAVs). The application of UAVs can significantly increase productivity while reducing the number of working hours as well as the stability and accuracy of the processes. As specified in [7], two main groups of PA tasks involving UAVs can be identified. Tasks that require some kind of physical interaction with the culture are distinguished as "crop spray-

ing" (e.g. irrigation, fertilization, pesticide application), while "Remote Sensing" (RS) indicates non-contact procedures to obtain information on the state of the crops (e.g. mapping, crop forecasting, weed recognition, health monitoring).

One of the most widely implemented RS techniques is aerial monitoring by analyzing aerial images. Those images can be captured by satellites, manned aircraft, or UAVs. Satellite-obtained images are not practical for most farmers due to their high cost, low availability and dependency on the weather conditions. Compared to satellite-obtained images, using an aircraft results in better image quality. Using human-crewed aircrafts is, however, very costly and requires flying at high altitudes and cloud penetration to obtain clear imagery. On the other hand, UAVs can fly at lower altitudes and are easy to operate, and therefore present a more cost-effective option for acquiring high-resolution images (as in the example of vineyard mapping in Figure 1.1).

Figure 1.1: Vineyard mapping with a UAV.

Aerial imagery is usually used for map generation in order to obtain information about the biophysical characteristics of the crop field. To obtain insights from the aerial images, a single capture is often not sufficient to cover the area of interest completely. It is, therefore, necessary to integrate multiple georeferenced images through mosaicing methods, with overlapping of 60-80% between subsequent images. To do so, a complex flight plan needs to determine precise waypoints where the image acquisition has to occur. In order to successfully assemble overlapping image captures, positioning with high precision is needed. Due to their capability of performing hovering while delivering high-resolution cameras, using UAVs progressively becomes the optimal choice for mapping.

By using UAVs equipped with a camera, farmers can continuously monitor specific crop parame-

ters, such as crop variability and stress conditions. Remote sensing methods in agriculture mostly rely on data extracted from electromagnetic radiation interaction with soil and crops. By measuring the reflected radiation of the crops, it is possible to generate agroclimatic models based on the Normalized Difference Vegetation Index (NDVI) or other vegetation indices, which allows the identification of crop subareas and their individual characteristics [4]. Remote sensing primarily focuses on measuring the radiation reflected by plants which correlates inversely with the amount of radiation absorbed by plant pigments. In particular, the NDVI can offer precise insights into biomass levels, which can provide guidance and understanding of the specific conditions, e.g. water stress, crop diseases, and nutrient deficiencies. This valuable information can aid the farmers with decision-making and processes for increasing crop productivity.

However, the technical limitations of UAVs also need to be taken into account when considering their agricultural mapping application. To ensure a successful mission for such an application, different factors need to be considered, including control of the UAV to enable stable flight with maintained constant velocity in order to obtain the necessary level of image quality, as well as high precision while visiting the predefined waypoints.

The current research explores challenges concerning UAV battery life and efficiency, flight time limitations, communication distance, payload, engine power, stability, and reliability [5]. In particular, aerial imagery needs to be obtained in the narrow time windows to acquire exploitable data for crop analysis. Thus, significant improvements need to be made in terms of reducing the mapping mission time. On the other hand, the battery capacity limits the duration of a single flight and at the same time increases the overall mission time due to the time necessary for battery recharge and/or battery replacement. Nonetheless, battery replacements directly affect mission safety as repeated take-offs and landings are required for each battery that is used for a flight segment. The highest flight safety risk arises at take-off and landing operations. Fully charged lithium-ion batteries currently in use can be used for a 20–30 min flight. However, battery management requires attention in order to preserve its initial capacity, as its proper use and maintenance can contribute to better battery health and reduce periodic replacements [8].

When it comes to mapping large dimension fields, employing multiple UAVs can overcome constraints linked to insufficient battery capacities and increase mission efficiency in terms of time and scalability to a large extent. By dividing the large field into subareas and allocating among UAVs in a fleet, challenges submerged from UAV limitations can be met. As a result, it is possible to obtain homogeneous images without change in daylight that can provide accurate insights about the observed

crop parameters. Significant research efforts therefore focus on improving the autonomy aspect of a UAV fleet mission, involving fewer operative tasks required from the operators on the ground by implementing advanced ground control systems [4, 7, 8]. Accordingly, the coordination of the UAV fleet must address the potential collision problem by implementing robust strategies for necessary and timely avoidance. This involves not only the detection of possible collision scenarios but also the execution of precise maneuvers to prevent accidents.

The problem framework for conducting RS aerial imaging with a single or multiple UAVs can be mainly divided into two subproblems: mapping mission planning and UAV control. The former step defines the path for covering the area of interest considering the UAV limitations, while the latter ensures that trajectory tracking of the planned path is successfully completed. The mapping mission requires integrating previously mentioned subproblems while addressing certain challenges that may arise, which will be described later in this chapter.

1.2 Guidance and Control of the UAVs

State of the art of application of UAVs in agricultural mapping missions regarding guidance and control is rich. Thus, a non-exhaustive state of the art will be presented here with the work relevant to the problem of this thesis.

1.2.1 Coverage Path Planning for Mapping Missions in Agriculture

Coverage Path Planning (CPP) problem falls within the domain of motion planning in robotics, focusing on generating a path for a robot to systematically explore every point of interest within a specified scenario. Despite the continuous advancements in technology, it is still challenging to ensure that UAVs perform the mapping mission completely in an autonomous manner. Human surveillance and support are required in the phases of take-off and landing, as well as supervising the mission execution, by monitoring the battery discharge level or handling the failure if it occurs. Therefore, appropriate CPP needs to be chosen in order to address challenges specific to a given UAV mission and ensure a high level of autonomy.

The planning phase of the optimal mapping mission for UAVs needs to address the challenges discussed in [9], which can be summarized into:

- **Time efficiency:** Coverage completeness should be ensured within a reasonable time frame. Time-efficient mission is essential for numerous reasons. It is indispensable to complete the

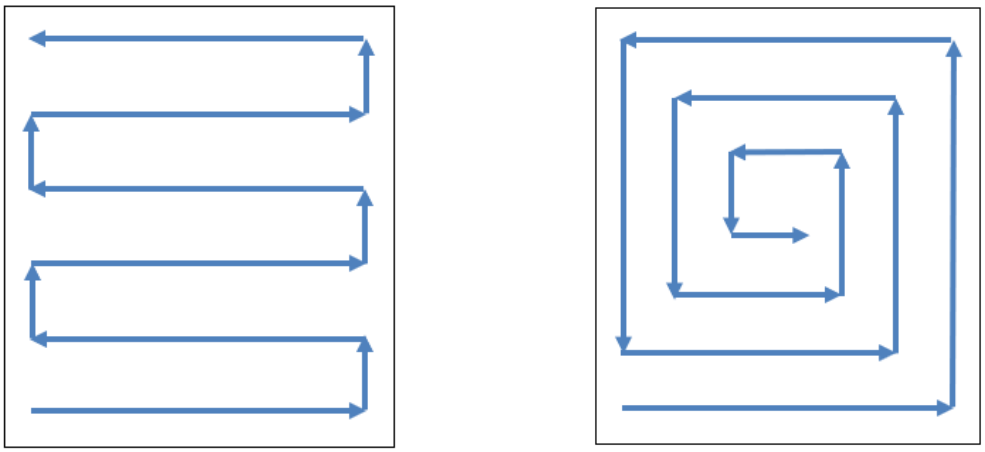
mapping mission within the least possible time, i.e. without change in daylight, as the resulting snapshots uniform in color can provide useful takeaways and aid with the decision-making process. In addition, reducing mission duration enhances cost efficiency by lowering overall expenses. This reduction includes decreased costs per flight hour, minimized staff expenses, and reduced drone wear — directly influenced by the number of take-off and landing cycles. Consequently, optimizing mission time enhances operational efficiency and results in significant cost savings across multiple facets of the mission.

- **Energy efficiency:** Limited capacity of the UAV batteries can hinder successful mission completion if not considered upfront. Thus, it is imperative to develop mission plans that prioritize minimizing energy consumption to ensure optimal battery utilization. An energy-aware mission strategy for UAVs involves integrating an understanding of the energy consumption model and battery limitations into the planning process. Knowledge about the battery capacity needed for a certain mission can help in determining the number of batteries or UAVs needed to complete the mission. This approach aims to mitigate unnecessary energy losses and maximize the efficient use of available battery capacity throughout the mission duration. By considering these factors upfront, energy-aware mission planning enhances the overall reliability, endurance, and effectiveness of the performed task.

- **Scalability:** Scalability in the context of optimization algorithms denotes their capacity to efficiently process and solve large and complex optimization problems within a limited computation time and capacity. It involves the ability of an algorithm to scale up and adapt to growing problem sizes and complexities while still delivering optimal solutions in a timely manner. The algorithm must demonstrate robustness in managing diverse scenarios, such as varying field sizes and multiple battery capacities, to formulate comprehensive and effective mission plans. This flexibility is especially critical for dynamically adjusting and replanning missions in real time to address unforeseen issues or changes in operational conditions.

CPP approaches are structured based on different criteria. First, a classic taxonomy defined in [10] depends on the area decomposition type. Three groups of CPP methods are differentiated: the methods with no decomposition and methods with exact or approximate cellular decomposition. Approximate decomposition methods are also known as grid-based methods. Methods without decomposition are applied to non-complex areas of interest that usually require only one UAV, and where simple geometrical patterns of sweep directions are explored, such as back-and-forth and the spi-

ral (Figure 1.2). In back-and-forth sweep direction, also known as zigzag movement or lawnmower pattern, the UAV performs straight movements with 180° turning maneuvers at the area borders. In contrast, with a spiral flight direction, the UAV flies in a circular pattern by decreasing the circle radius while flying towards the center.



(a) (b)

Figure 1.2: UAV sweep direction patterns: (a) Back and forth flight pattern (left-hand side); (b) Spiral flight pattern (right-hand side).

Exact cellular decomposition implies simplifying the coverage of the irregular-shaped area by dividing it into sub-areas and reducing the concavities. The approximate CPP approaches can be clustered considering the level of information available about the area of interest. Full information for the approximate approach requires algorithms that guarantee the complete coverage of the area of interest by visiting every cell. In that case, path planning is done offline in three steps: area decomposition, path planning, and flight execution. Once the area is decomposed into subareas or cells, path planning defines an optimal path considering the mission requirements and limitations. The last step involves executing the planned mission, provided no failures occur during the flight. On the other hand, partial information availability is applied in dynamic and constantly changing environments. Thus, the appropriate CPP method needs to ensure a balance between gathering information about the area and planning the path at the execution level. As these CPP techniques require performing area coverage under uncertainties, they are often sensor-based and bio-inspired, where flight plan is being updated based on the newly acquired data from the sensors. Finally, performance metrics are defined to evaluate mission success. Depending on the application, the common performance metrics for coverage mission assessment listed in [11] are the following: the total travelled distance, the time to complete a mission, the area coverage maximization, and the number of turning maneu-

vers. An appropriate trade-off needs to be found between distance minimization and area coverage maximization.

Depending on the mission-specific UAV application, different CPP approaches exist in the literature. The optimality of the chosen method is defined by the mission objective and constraints.

- **Distance minimization**

  A CPP algorithm for an arbitrary polygon area developed in [12] is designed for convex and concave areas of interest with the aim to shorten the flight distance by finding the optimal heading angle. In irregular and concave areas, mission complexity is reduced and continuity is ensured by considering the concave angle when defining the convex subareas that need to be connected in order to perform complete coverage.

  An improved method of the exact cellular decomposition considering the polygon area of interest is introduced in [1] and [2]. An example of the resulting coverage path derived from this method is given in Figure 1.3, where back-and-forth movements are aligned along the longest edge of each of the decomposed subregions.

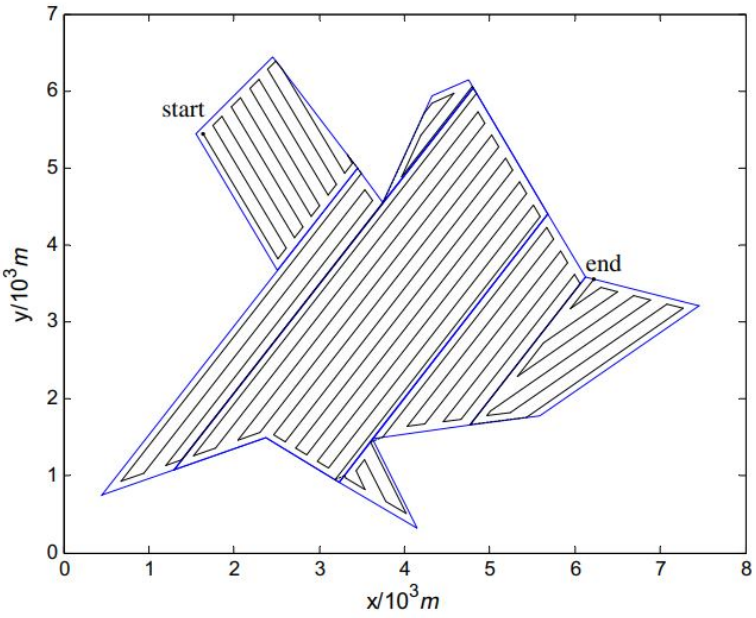


Figure 1.3: Example of the coverage path resulting from the exact cellular decomposition method in [1] and [2].

They study the impact of the turning motion on lowering efficiency in terms of route length, flight duration and energy consumption. Consequently, a convex decomposition algorithm based on the greedy recursive method minimizing the width sum is proposed. Resulting UAV paths form

a flight along the vertical direction of the subarea width to achieve the least number of turns. Finally, the minimum traversal algorithm of the undirected graph aims to connect the subareas by minimizing the repeated paths of the UAVs. In [13], a CPP algorithm fitting convex and non-convex area is proposed. For the latter, a decomposition method for the coverage of a concave polygon considering an interrupted path is developed. If there is no interruption, coverage can be planned for the convex hull of the considered polygon using the optimal line sweep calculated as the path with a minimized number of turns. On the other hand, if the path is interrupted, the initial polygon is decomposed and the optimal path is found for each subpolygon recursively. Authors in [14] present a CPP method for performing aerial surveys with fixed-wing UAVs in the presence of steady uniform wind that aims to determine the optimal sweep angle relative to the wind that minimizes flight time.

- **Coverage optimization**

  Grid-based method for CPP problem in [15] presents a scheduling framework for hexagonal sampling that results in optimal coverage density and spatially balanced sampling cycles for environmental monitoring, subject to limited energy. Furthermore, a partitioning method of the area of interest in [16] considers no-fly zones (NFZs) to be avoided. To improve the quality of coverage, partition borders are chosen to be on the grid-cell boundaries to reduce the percentage of uncovered area. The grid technique for mapping the area while efficiently avoiding NFZs proposed in [17] uses the self-adaptive parameter setting to tune the pattern and employ parallel computing, which results in improved performance.

  A learning-based adaptive path planning algorithm for accurate semantic segmentation is proposed by [18], linking the UAV altitude and semantic segmentation accuracy (similar to [19]), for ensuring high-resolution images. First, they define non-overlapping regions of the target field with associated waypoints in the 3D space as locations where the UAV camera covers the entire area. The back-and-forth path linking the waypoints serves as initialization for the proposed algorithm. The two-step adaptive strategy consists of assigning a semantic label to each pixel at every waypoint by using deep neural network, and deciding if the predetermined detail level is sufficient for desired accuracy. If the image resolution is not satisfied, UAV lowers the flying altitude at the next waypoint, and continues with the predetermined path otherwise.

- **Energy-aware coverage**

  As previously mentioned, there is an increasing importance of developing coverage path plan-

ning with minimal energy consumption for UAVs, which addresses challenges linked to UAV limited endurance. Energy-aware coverage planning in [20] and [21] relies on energy estimation models with the objective of finding the optimal flying speed. In [22], an energy-aware mission is planned by considering energy consumption for different segments of the planned path, straight lines and turns. Finally, the optimal path is the result of minimizing the sum of turning maneuvers, as they increase the energy consumption. In addition to minimizing the number of turns, method in [23] optimizes the area division by clustering to ensure equal energy consumption demand for each area section.

As it imposes major limitations to performance, battery capacity is a crucial element that needs to be accounted for when planning a mapping mission for UAVs. Due to the rather large size of the field, multiple batteries must be employed for a mapping mission. Therefore, the initial flight plan should consider battery energy constraints, and battery replacements must be planned to optimize the efficient use of batteries. As described in [24], the optimal UAV path should include the least possible number of take-offs and landings due to the cost per take-off-landing cycle, as well as the wear of the drone.

Significant research efforts have been made in optimizing the UAV coverage path under energy constraints. In [25], maximum flight endurance is computed based on battery and motor characteristics, while forecasting the remaining mission time ensures that the planned path can be completed. Depending on the size of an agricultural field, the entire mapping mission often cannot be performed using only one battery. Coverage path planning under energy constraints in [26] hence considers that each path should start and finish at the recharging base, while [27] takes into consideration energy constraints of the UAVs and weather forecast to plan a delivery mission accordingly. In [28], a rectangular area is covered using a back-and-forth motion along lines perpendicular to the sweep direction, and the proposed optimization problem aims to minimize the time of the longest route among the subroutes while respecting the battery constraints. Similarly, the algorithm in [29] provides optimal multi-UAV mission cost by minimizing the maximum individual cost of subpaths, that are initially assigned iteratively to each UAV until their maximum capacity is attained.

When considering battery capacity limitations, a battery replacement strategy also needs to be developed to optimize the overall mission performance. The battery replacement method proposed in [30] presents the algorithm for routing the quadcopter using the automatic battery replacement aerial system. It has been, however, developed for employment in nuclear power

plant monitoring in order to enable mission completion via the UAV in a single deployment, where mission configuration differs largely from that of the agricultural application.

Finally, the choice of the location for a recharging station can impact the mission duration, and consequently the energy consumption. In [31], the choice is done among potential charging base locations at the extremities of the vertices of the area. On the other hand, [24] introduces a mobile charging base along the road next to the field, which, however, requires the necessary infrastructure. Similarly, mission in [32] includes UAV and a supporting Unmanned Ground Vehicle (UGV) that serves as a mobile recharging station.

1.2.2 Control Methods for UAVs

Given a specific task that a UAV needs to accomplish and the associated environment, developing fully autonomous UAVs requires focusing on motion control. Motion control in robotics addresses the problem of reaching a desired position and tracking a desired trajectory by determining the dynamics generated by the robot or vehicle actuator, such as forces and torques. Motion control tasks include but are not limited to path planning, point stabilization, and trajectory tracking. More specifically, once a path is determined, control inputs are generated so that the vehicle accomplishes the task of tracking the referenced planned path [33].

The trajectory control problem, defined as ensuring a vehicle follows a predefined path in space, can be solved with trajectory tracking or path-following control strategies. The trajectory tracking has a position reference linked with time, where the reference is given in a specified temporal instant, whereas there is no time dependence in the path following, as the temporal dimension of the problem is removed [34]. By definition in [35]:

- **Trajectory-tracking problem:** Let $p_d : [0, \infty) \to \mathbb{R}^3$ be a given sufficiently smooth time-varying desired trajectory with its time-derivatives bounded. Design a controller such that all the closed-loop signals are bounded and the tracking error $\|p(t) - p_d(t)\|$, where p is the actual position and p_d desired position of the vehicle, converges to a neighborhood of the origin that can be made arbitrarily small.

- **Path-following problem:** Let $p_d(\gamma) \in \mathbb{R}^3$ be a desired path parameterized by virtual arc $\gamma \in \mathbb{R}$ and $v_r(\gamma) \in \mathbb{R}$ a desired speed assignment [1]. Suppose also that $p_d(\gamma)$ is sufficiently smooth with respect to γ and its derivatives (with respect to γ) are bounded. Design feedback control laws u_v,

[1] For simplicity of presentation, it is assumed that $v_r(\gamma) \in \mathbb{R}$ does not directly depend on time t.

u_ω ($v \in \mathbb{R}^3$ and $\omega \in \mathbb{R}^3$ denoting linear and angular velocities, respectively), and $\ddot{\gamma}$ such that all the closed-loop signals are bounded, the position of the vehicle converges to and remains inside a tube centered around the desired path that can be made arbitrarily thin, i.e., $\|p(t) - p_d(\gamma(t))\|$ converges to a neighborhood of the origin that can be made arbitrarily small, and the vehicle satisfies a desired speed assignment v_r along the path, i.e., the speed error $\dot{\gamma}(t) - v_r(\gamma(t))$ can be confined to an arbitrarily small ball.

However, since some of the path-following methods originate as adaptations of trajectory-tracking algorithms, some control-oriented algorithms may need a timing law for the virtual arc parameter $\gamma(t)$.

Because agricultural mapping tasks require the accurate position of the UAV at any given time, the trajectory tracking problem will be analyzed and implemented further in this thesis because of their temporal aspect.

In general, control techniques for trajectory tracking for UAVs can be classified into three groups: linear, nonlinear, and learning-based intelligent controllers [36].

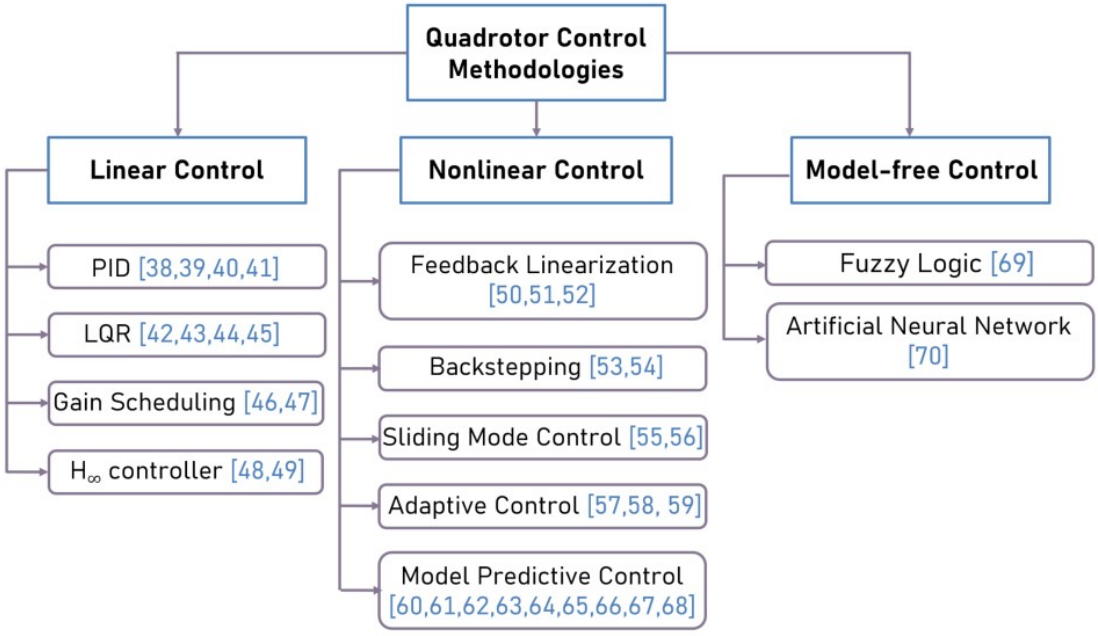


Figure 1.4: Control methodologies classification.

A non-exhaustive list of three control methodology groups used in quadrotor systems [37] [38] is illustrated in Figure 1.4 and described below.

1.2.2.1 Linear control

- **Proportional-Integral-Derivative (PID) Controller:**

  PID controller, a classical control method widely used in industrial systems due to its simplicity and ease of implementation, is nowadays largely employed in commercial quadrotors. In the literature, PID is mostly used for attitude control and flight stabilization [39]. Robust PID for position control of a quadrotor is developed in [40], with a procedure for reducing the power demanded by the controller. The robustness of a nonlinear PID controller proposed in [41] is assessed in the presence of disturbances in one of the actuators. Nevertheless, because of its simplified nature, the PID controller response exhibits overshooting and is often not an appropriate solution for a nonlinear underactuated system such as the UAV itself [42].

- **Linear Quadratic Regulator (LQR):**

  LQR algorithm is the optimal feedback controller for operating a dynamic system at a minimum cost. System output is obtained through feedback for closed-loop stabilization. LQR controller for linearized quadrotor model, as well as its parameter tuning for obtaining a satisfactory response, is presented in [43]. In [44], the authors have developed a discrete-time, finite-horizon LQR, where the nonlinear quadrotor system is linearized using a left-invariant error about a reference trajectory so that the optimal gain sequence can be computed offline. With a robust LQR presented in [45], the adaptive weighting matrix selection method ensures the flight stabilization of a small quadrotor under severe disturbances and model uncertainties. Fault-tolerant LQR in [46] guarantees the system stability in case of known faults or failures in actuators and/or sensors.

- **Gain Scheduling:**

  Gain scheduling is a control method where a nonlinear system is linearized at multiple operating points, and an appropriate linear controller is designed for each point. Gain scheduling, in combination with other control techniques, has been implemented for quadrotor control. Gain scheduling-based PID control for path tracking and fault tolerant control of a UAV is presented in [47]. An event-triggered control strategy in [48] is used to counteract the effect of the resulting time-varying delay with the provided gain-scheduling method.

- **H_∞ controller:**

  H_∞ controller is a linear robust control technique that deals with parameter uncertainties and

effects of external disturbances to ensure reaching the desired output for the UAV flight. However, the controller is designed in the frequency domain, which makes its gains difficult to adjust and computationally expensive. The H_∞ control problem is formulated as a mathematical optimization problem that minimizes the H_∞ norm of the closed-loop transfer function. In the literature, it has been implemented for stabilization and position control of the UAVs. In [49], the H_∞ controller was designed for stationkeeping and hovering of an unmanned helicopter. A 2 degree of freedom (DOF) H_∞ loop-shaping controller presented in [50] is used to control the attitude angles and vertical velocity, as well as to provide stabilization of a quadrotor UAV.

1.2.2.2 Nonlinear control

- **Feedback Linearization Control:**

  Feedback linearization is a nonlinear control methodology where the nonlinear dynamics of the system are linearized by model inversion so that feedback control can be applied. Nonetheless, as this approach requires a precise model, and the modeling of the errors and uncertainties is challenging, it is not robust to uncertainties in the system [51]. There are multiple applications of feedback linearization applied to UAVs, e.g. for stabilizing a highly nonlinear quadrotor system with a composite feedback linearized and LQR controller in [52], or for tracking control by combining the feedback linearization method and H_∞ algorithm in [53].

- **Backstepping Control:**

  Backstepping control uses a recursive algorithm for controlling nonlinear systems. It is decomposed into several steps to ensure the asymptotic stability of the system at each step. Despite its fast convergence, the disadvantage of this method is its lack of robustness. Particularly, backstepping requires the exact system dynamics and uncertainties to provide good performance [37]. Backstepping-based adaptive control of a quadrotor UAV with guaranteed transient and steady-state tracking performances is developed in [54], where the parametric inertia effects and drag uncertainties for attitude control are compensated using the least squares-based parameter identification algorithm in the indirect adaptive control design, and using a constructive Lyapunov analysis approach in the direct adaptive control scheme. The robust nonlinear controller in [55] combines the sliding mode control with the backstepping controller, where the backstepping sliding-mode controller is evaluated for position and yaw angle control of the quadrotor.

- **Sliding Mode Control:**

  Sliding Mode Control (SMC) is a robust nonlinear control strategy that applies a discontinuous control law to drive the system state to a sliding surface, where the dynamics of the system are simplified. The control signal switches from one state to another to ensure convergence of the states to the reference state. In [56], a second-order sliding mode control method for position and attitude tracking is designed by dividing the quadrotor system dynamics into two subsystems: a fully actuated subsystem and an underactuated subsystem. A sliding manifold is defined for each of the subsystems. In order to ensure asymptotic tracking of the command profile for the quadrotor, a continuous sliding mode control driven by sliding mode disturbance observer is presented in [57].

- **Adaptive Control:**

  Adaptive control is suitable for systems with unknown dynamics, as it directly compensates for parameter changes in system dynamics by adjusting the controller characteristics with the objective of maintaining overall system performances. Authors in [58] present an adaptive controller based on Lyapunov stability and its application to a quadrotor UAV. The adaptive controller is formulated for the command tracking problem, based on the reference model with linearized quadrotor dynamics, where parametric uncertainties are present in the form of actuator failures, more precisely, loss of a thrust portion in one or multiple propellers. Similarly, adaptive control for the trajectory tracking problem developed in [59] considers the input saturations and uncertain parameters with the known bounds, which are handled by using a combination of the smooth saturation function and smooth projection operator in the control design. In [60], authors propose quadrotor stabilization and trajectory tracking with dynamic changes in the quadrotor's center of gravity as the uncertain parameter. They propose an adaptive tracking controller based on the output feedback linearization to compensate for the changes in uncertain center of gravity.

- **Model predictive control:**

  Model predictive control (MPC) is a control strategy that repeatedly solves the optimization problem over a finite prediction horizon and determines the future control sequence on a moving horizon. MPC has shown promising results in trajectory tracking. Its ability to handle constraints makes it an easily adaptable solution for various problems [61]. Historical basis and general concepts of MPC are introduced in [62], as well as in [63]. Advantages in terms of accuracy and

response time of MPC-based controller compared to other control strategies are discussed in [64] and [65]. The advantages of the MPC controllers will be presented in detail later in the thesis. Even though plenty of research has been conducted on the linear control of UAVs, approaches in nonlinear model predictive control (NMPC) have drawn attention due to their ability to ensure application on real systems with minimal tracking error with reduced computational complexity. Experimental comparison of linear and nonlinear MPC in [66] shows better tracking results in smoothness for curved paths implemented on real-time systems, and [67] proves better disturbance rejection capability in addition to the tracking performance of nonlinear MPC in comparison to linear MPC. The authors in [68] present flatness-based MPC with feedforward linearization for improved robustness to modeling errors and known input time delays. Different robust MPC strategies for UAV trajectory tracking are developed in [69],[70] and [71]. Authors in [69] propose the control strategy consisting of a receding horizon scheme, with the optimization process that incorporates the minimum peak performance measure, optimality metric, for minimum possible deviation under the worst–case disturbance. On the other hand, robust nonlinear MPC in [70] is constructed to tackle external disturbances with the tightened state constraints based on the Lipschitz condition. Robust adaptive MPC introduced in [71] ensures robustness on two levels. First, the proposed identification method guarantees that the true parameter is always included in the updated uncertainty set, such that robust constraint satisfaction is guaranteed in the closed loop. Then, the tube MPC predicts the state propagation of a system with polytopes within which the guaranteed states exist, while respecting a bounded additive disturbance and bounded uncertain parameters.

1.2.2.3 Model-free control

For model-free controllers, no dynamics model is required. The system information is rather acquired from data obtained from the real system flights. These control methods are based on artificial intelligence and are largely used as a type of fuzzy logic that transforms ambiguous situations into approximations. The fuzzy controller in [72] is used to achieve stable hovering of the quadrotor in the presence of velocity disturbances. Authors in [73] compare the artificial neural network's direct inverse control with the PID control system, and highlight the advantage of the learning mechanism to overcome the limitation of PID tuning. In general, the model-free approach enables these controllers to be implemented on different UAV configurations. However, stability and robustness analysis remain uncertain and challenging when it comes to these methodologies.

1.2.3 Multi-UAV mapping missions

As suggested in [74], performing a mapping mission in PA with multiple UAVs can be structured into three parts:

- **Task subdivision and allocation**

  This is a multi-robot or multi-UAV coordination problem, where, given a global task T_0 and R robots, partition of T_0 needs to be found into R subtasks, that need to be allocated to robots or UAVs for execution by considering their individual capabilities. The goal is to achieve minimum overlapping between subtasks, which is ideally null, such that the original task is completely covered by the union of subtasks.

- **Path planning**

  Once the subtasks, i.e. subareas are allocated, each UAV needs to find a path to accomplish their assigned task by determining waypoints, such that the whole subarea is covered. Waypoints are the positions defined by the coordinates that identify a point in physical space where a task (e.g. taking a snapshot) needs to be accomplished. Coverage Path Planning (CPP) algorithms are used to find such path in the free workspace.

- **Multi-UAV control**

  A multi-UAV system working together to achieve a mutual goal is called a swarm or fleet of UAVs. Multi-UAV or swarm-control techniques are developed to enable UAVs to efficiently perform a given task in a cooperative manner by lowering energy costs and shortening flight times. In general terms, strong cooperation requires direct interaction in task execution, whereas in weak cooperation, each individual is assigned a localized subtask to perform. Strong cooperation in a multi-UAV system implies coordination between the UAVs, e.g. swarm or formation control, as well as the adaptive force or velocity control. Weak cooperation, on the other hand, can refer to considering collision avoidance between UAVs, assigning different level of priority to each UAV, or dealing with system management in case of failure occurrence, notably fault detection and diagnosis [75]. For a successful implementation of a multi-UAV system in agriculture it is important to facilitate the control and lower the effort necessary for operating UAVs simultaneously. Additionally, due to unpredictable weather conditions and possible perturbations, a lack of accurate data presents a need for real-time coordination.

When it comes to employing multiple UAVs for mapping an agricultural field, task allocation and path planning are considered as the first step. Since the two subproblems can be addressed in no particular order, and one may depend on the other, they are usually both handled by a single CPP strategy adapted for a specific multi-UAV mission.

Different strategies for cooperative multi-UAV coverage path planning can be found in the literature. Authors in [76] propose a method for the full coverage path planning divided into two parts. First, path planning is done for the complete area by selecting the covering operation mode for a single area, considering the determined criteria. Then, dispatching of the given operations areas to available multiple machines is done through an optimization algorithm integrating particle swarm optimization and genetic operation. A CPP algorithm for multiple UAVs that partitions the area of interest based on the individual capabilities of the vehicles is presented in [77], where the area is distributed and assigned proportionally to their relative capabilities and initial locations. In [78], the authors introduce a grid-based area decomposition method that transforms the area of interest into the grid cells by applying a rotation of the complete area in order to minimize the resulting grid and, consequently, completion time. Task allocation to multiple UAVs is formulated afterward as a mixed-integer linear programming (MILP) model that minimizes the completion time to cover the area and is solved with the randomized search heuristic algorithm to lower the computational time. A decentralized multi-UAV path planning method based on the two-layer coordinative framework is introduced in [79], a consensus-based algorithm suitable for a large number of UAVs in the system due to its scalability.

Once the mapping mission is planned and paths are determined for each UAV in the fleet, an appropriate control strategy is needed to ensure tracking of the planned individual paths. A robust and efficient control system for managing a swarm of UAVs effectively needs to address several points, namely coordination and collaboration. Control approaches for coordination and collaboration among UAVs to avoid collisions, optimize task execution, and achieve common objectives often involve decentralized decision-making algorithms.

In order to assure the safety of a multiple UAV system with a common goal, cooperative collision avoidance strategies involve techniques that enable multiple UAVs to work together to avoid collisions while navigating in shared airspace. According to [80], key strategies for collision avoidance can be clustered into 5 categories (Figure 1.5):

1. **Path planning:** In the case of the known environment, as well as its static and dynamic obstacles, the objective of a path planning algorithm is to find an optimal collision-free path. Path selection can be done with an online or offline planner, and it results in a curve connecting points of inter-

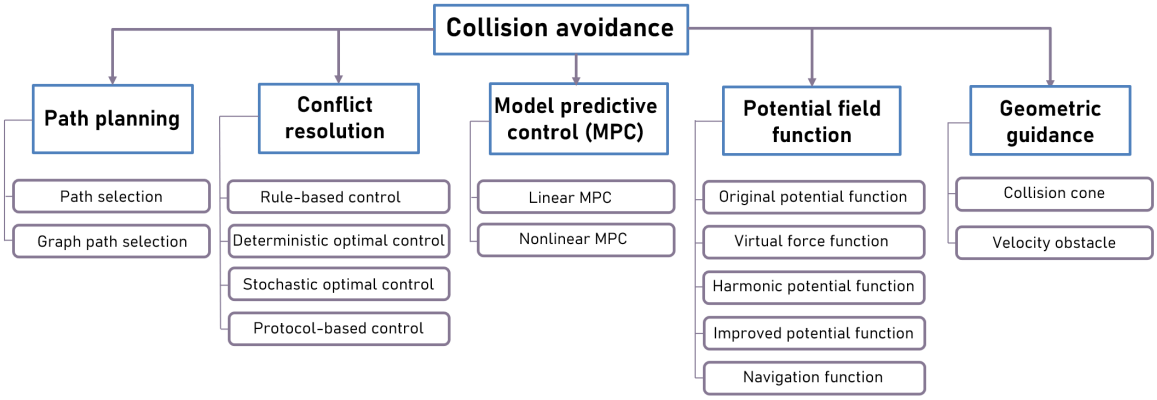


Figure 1.5: Collision avoidance strategies classification.

est, without having to consider a dynamical model of a UAV. Graph selection, on the other hand, requires transforming obstacles into polygons that need to be avoided. The problem is then solved by the graph-based optimization approach. A fundamental example of this approach in [81] introduces a collision avoidance algorithm for planning a safe path for a polyhedral object moving among known polyhedral objects. The obstacles are represented as forbidden regions, and trajectory is found by searching a network for vertices that can be reached safely without collision.

2. **Conflict resolution:** Conflict detection and resolution algorithms are explored so that UAVs do not depend solely on a centralized solution to avoid collision. According to [82], conflict detection is a process of deciding when an action should be taken, whereas conflict resolution defines what action should be done and how. For example, authors in [83] introduce a decentralized cooperative policy as a base for conflict resolution in a multi-vehicle system, where each vehicle applies the policy based on locally available information. Each vehicle has its defined safety disc based on the motion equations, and a collision occurs whenever two safety discs overlap. In that case, an appropriately defined decentralized cooperative policy for conflict resolution is applied.

3. **Model predictive control:** Due to its capability to apply optimization methods for simultaneously handling constraints and a cost function, model predictive control (MPC) represents an efficient method for collision avoidance implemented on the control level. MPC strategies anticipate potential collisions based on the dynamical model and current trajectories of UAVs in the system. Intervehicle collision avoidance in [84] is ensured by the collision cost function in a nonlinear MPC framework combined with a priority strategy. A consensus-based decentral-

ized MPC approach in [85] decouples the collision avoidance constraint for each UAV to make decisions independently while ensuring consistency of coupled collision-avoidance constraint. MPC application to collision avoidance in a multi-UAV system will be further discussed later in the thesis.

4. **Potential field:** In this method, the artificial potential field is designed with two components: attractive field and repulsive field [86]. The attractive field is the value of the Euclidean distance from a certain position to the destination point, which naturally decreases as the UAV advances to its destination. The repulsive field represents the obstacles, and its value increases as the UAV approaches an obstacle. The resulting sum of both fields directs the path such that the UAV moves toward the lowest potential, such that it follows the gradient of potentials yielding to the path that is attracted by the target and repelled by obstacles. An optimized artificial potential field algorithm for multi-UAV collaborative trajectory planning and collision avoidance presented in [87] method considers the UAV companions as dynamic obstacles to accomplish collaborative trajectory planning. In [88], a probabilistic conflict detection algorithm using the generalized polynomial chaos method is developed for determining three-dimensional conflict-free trajectories considering wind uncertainty.

5. **Geometric guidance:** Geometric guidance methods consider the conflict geometry in order to reactively decide upon the collision avoidance control, with two main approaches: collision cone and velocity obstacle. In the collision cone approach, the UAV is represented by a point, and the obstacle is determined by a circle with a defined radius. The tangent lines of the circle form a cone between the UAV and the obstacle. The velocity obstacle approach is defined in a velocity space by moving the collision cone by the obstacle velocity. Authors in [89] introduce a nonlinear geometric guidance approach for obstacle avoidance. Rather than considering only the position guarantee, the guidance algorithm determines the velocity command along the tangent line over the collision cone. A real-time guidance law based on differential geometry is presented in [90] to avoid obstacles and guide formation reconfiguration for UAVs. The velocity obstacle approach was extended to three-dimensional obstacles in [91] by representing them with the pyramid cone method.

1.3 Objectives and contributions of the thesis

1.3.1 Objectives

The primary objectives of this thesis are to address and overcome the challenges associated with utilizing UAVs, particularly quadrotors, for enhancing efficiency in agricultural mapping tasks. First, it aims to understand the specific challenges linked to implementing UAVs in this context. Secondly, it focuses on developing an energy-aware path planning strategy that ensures optimal battery management, is time-efficient, safe, scalable, and minimizes disruptions related to battery replacements. Thirdly, the thesis seeks to design a control strategy for precise trajectory tracking, capable of handling nonlinearities and maintaining constant velocity to ensure stable flight and high-quality image capture. Lastly, it explores the development of a multi-UAV system for mapping missions, with a focus on effective coordination among UAVs to ensure collision avoidance while maintaining high levels of positional and velocity accuracy.

1.3.2 Contributions

1.3.2.1 Energy-aware battery management optimization for path planning of a UAV mapping mission

Contribution to the field of UAV agricultural mapping consists in introducing an innovative, energy-aware mission planning strategy that significantly improves efficiency and safety. The research underscores the need for optimal flight path planning to achieve comprehensive area coverage while respecting requirements for time- and energy-efficient, as well as a scalable solution. A key contribution is the development of a method for effective battery management, which involves waypoint allocation to optimize battery replacement schedules and minimize energy consumption. Additionally, by strategically positioning the base station relative to the mapping area, the approach reduces overall mission duration and flight distance. This leads to fewer battery replacements and a decrease in the number of high-risk take-offs and landings, resulting in enhanced mission safety. Additionally, the strategy supports long-term UAV operation by preserving battery health and optimizing performance, ensuring greater reliability and longevity of the UAV system in large-scale mapping applications.

1.3.2.2 Nonlinear MPC for UAV trajectory tracking

The proposed Nonlinear Model Predictive Control (NMPC) approach effectively manages trajectory tracking while maintaining a constant velocity, a crucial element for obtaining high-quality and reliable images. The presented work emphasizes the need to balance the constant velocity with the accuracy required for effective mapping. Simulation results show that adding a constant velocity cost can improve the smoothness of both the trajectory of the UAV and its control inputs, leading to improved image quality. Robustness assessment against external disturbances, such as wind and model uncertainties in thruster efficiency, demonstrates its ability to sustain performance under different conditions.

1.3.2.3 Nonlinear MPC for multi-UAV trajectory tracking with priority-allocation to ensure collision avoidance

The proposed distributed NMPC introduces a priority allocation mechanism that designates a single UAV to handle collision avoidance at any given time. This approach reduces the likelihood of unwanted coupled maneuvers and ensures that path adjustments do not trigger unnecessary changes across the entire multi-agent system, simplifying the control problem and conserving energy. Collision avoidance is integrated into the control problem as a penalty cost, which helps to relax the control constraints while ensuring safety. This method allows UAVs to follow their optimal paths unless imminent collision risks prompt avoidance maneuvers. The optimal control problem for multi-UAV agricultural mapping missions was evaluated with an emphasis on efficiency and robustness. The proposed strategy enhances mission efficiency by reducing computational time and demonstrates a degree of robustness against disturbances and uncertainties, showing the system's adaptability in dynamic conditions.

1.3.2.4 Implementation of the proposed energy-aware path planning on experimental setup

The contribution of this research work lies in the successful implementation of optimized mission planning strategies on a real UAV system within a real-world setting. The findings from these experiments reveal notable enhancements in the efficiency and effectiveness of agricultural mapping missions. The optimized path planning approach reduces battery consumption for non-mapping activities, such as traveling to and from the base station, resulting in fewer battery replacements and longer uninterrupted operation for quadrotors. Strategic positioning of the base station further decreases energy usage, enhancing the overall sustainability and operational efficiency of the missions.

1.4 List of publications

Conference articles

- **Novak, D.**, Tebbani, S., & Vande Wouwer, A. (2022, November). Distributed multi-objective state-dependent MPC for mapping mission with multiple UAVs. In *2022 10th International Conference on Systems and Control (ICSC)* (pp. 305-310). IEEE.
- **Novak, D.**, & Tebbani, S. (2023, June). Nonlinear MPC for the multi-UAV system with allocated priority for collision avoidance. In *2023 31st Mediterranean Conference on Control and Automation (MED)* (pp. 7-12). IEEE.
- **Novak, D.**, & Tebbani, S. (2024, July). A Comparative Study of NMPC Strategies for Prioritized Multi-UAV Trajectory Tracking with Collision Avoidance in Agricultural Field Mapping Missions. In *2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 1195-1200). IEEE.
- **Novak, D.**, Tebbani, S., Goricanec, J., Orsag, M. & Le Brusquet, L. (2024, July). Battery management optimization for an energy-aware UAV mapping mission path planning. In *2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)*. (pp. 2645-2650). IEEE.

Other publications

- **Novak, D.**, Tebbani, S., & Vande Wouwer, A. (2022, July). Distributed MPC for mapping mission with multiple UAVs. In *2022 41st Benelux Meeting on Systems and Control*.

Oral presentations

- **Novak, D.**, Tebbani, S., & Vande Wouwer, A.. (2022, May). Multi-UAV cooperative trajectory planning using distributed model predictive control. *ID2MOVE Symposium on Autonomous Systems*, Nivelles, Belgium.
- **Novak, D.** (2023, September). Drone trajectory optimization for mapping missions. *PhD day of the laboratory L2S*, Saint-Rémy-lès-Chevreuse, France.
- **Novak, D.**, & Tebbani, S. (2023, November) Nonlinear MPC for collision-avoidance trajectory tracking of a multi-UAV system in a mapping mission. *Journée du Comité Technique Commande Prédictive Non Linéaire*, Valence, France.

- **Novak, D.**, & Tebbani, S. (2024, February) Energy-aware UAV mapping mission optimization. *Seminar of the research group MODESTY at the laboratory L2S*, Gif-sur-Yvette, France.
- **Novak, D.**, & Tebbani, S. (2024, July) Energy-aware UAV mapping mission optimization. *Seminar on robotics and automation at the laboratory LARICS, IEEE Croatia section*, Zagreb, Croatia.

1.5 Thesis outline

The rest of the thesis is organized as follows.

Chapter 2: Modeling. This chapter introduces UAV classification and focuses on quadrotor modeling. The mathematical model of a quadrotor is then presented from a kinematics and dynamics perspective. Then, two energy consumption modeling approaches are issued from the electrical battery discharge model and aerodynamics-based energy consumption. Finally, a simplified nonlinear dynamic model of a quadrotor with 6 DOF is given for developing a model-based control strategy later in the thesis. Similarly, the battery consumption simplifications are issued from the previously elaborated energy consumption model.

Chapter 3: Energy-aware path planning for a UAV mapping mission. This chapter presents a path planning strategy based on battery management optimization. First, the chosen area decomposition approach results in the predefined flight path configuration. Then, in order to ensure a safe and energy-aware mapping mission, the proposed battery management optimization strategy defines the task allocation by assigning flight path segments to available batteries to minimize the total path distance and number of battery replacements needed. The mathematical problem formulation is followed by simulation results and discussion, where the proposed approach is compared to a benchmark method from the state of the art.

Chapter 4: Control strategy for UAV trajectory tracking. This chapter focuses on the development and application of a control strategy for UAV trajectory tracking, a critical aspect for ensuring precise navigation and mission success. The chapter begins with an introduction to the control scheme and the state-space model used to describe UAV dynamics. It then delves into the implementation of Nonlinear Model Predictive Control (NMPC) for trajectory tracking, including its mathematical formulation and considerations for output robustness and parameter tuning. Finally, the chapter applies NMPC to a UAV mapping mission, presenting simulation results that highlight the effectiveness and robustness of the proposed control strategy in various scenarios.

Chapter 5: Towards multi-UAV mapping mission. This chapter explores the extension to multi-

UAV mapping missions, where multiple UAVs must work together to complete a common task. The chapter begins with an introduction to the challenges and goals of multi-UAV missions. It then discusses path planning for multiple UAVs, ensuring that their trajectories are coordinated effectively. Coordination is a key aspect of the proposed distributed Nonlinear Model Predictive Control (NMPC) strategy, where the implemented passing priority allocation on the control level ensures successful collision avoidance. The mathematical formulation of this approach is presented, followed by simulation results that demonstrate its effectiveness and robustness in avoiding collisions while maintaining mission objectives.

Chapter 6: Application to mapping of an agricultural field. This chapter presents the practical application of the previously discussed energy-aware path planning optimization strategy to the real-world scenario of mapping an agricultural field. In order to match the needs of the experimental scenarios, the mathematical formulation developed in Chapter 3 is extended to include the choice of multiple base station locations directly at the optimization level. It then details an experimental study focused on a quadrotor conducting an energy-aware mapping mission, where efficient use of energy is critical. The results from field tests conducted at two sites are presented, showcasing the successful implementation of the proposed battery management optimization strategy in real agricultural environments.

Chapter 7: Conclusion and perspectives. This chapter concludes the thesis with final remarks and outlines several potential directions for future research.

Appendix A: Small-scale example solution for the battery management optimization. The first appendix illustrates a small-scale example case to showcase the impact of changing the path configuration on the optimization results for the battery management strategy.

Appendix B: Evaluating the impact of priority allocation in NMPC for collision avoidance. The second appendix demonstrates the benefits of implementing priority allocation when handling collision avoidance in a multi-UAV system.

Appendix C: Comparative study of NMPC strategies for prioritized trajectory tracking with collision avoidance in mapping missions. The third appendix presents a comparative study of three different collision avoidance strategies implementing NMPC.

Chapter 2

Modeling

2.1 UAV classification

Despite numerous possible classifications that exist for the categorization of the UAVs, the most significant criteria for their agricultural applications are wing type and autonomy level [7], as shown in Figure 2.1.

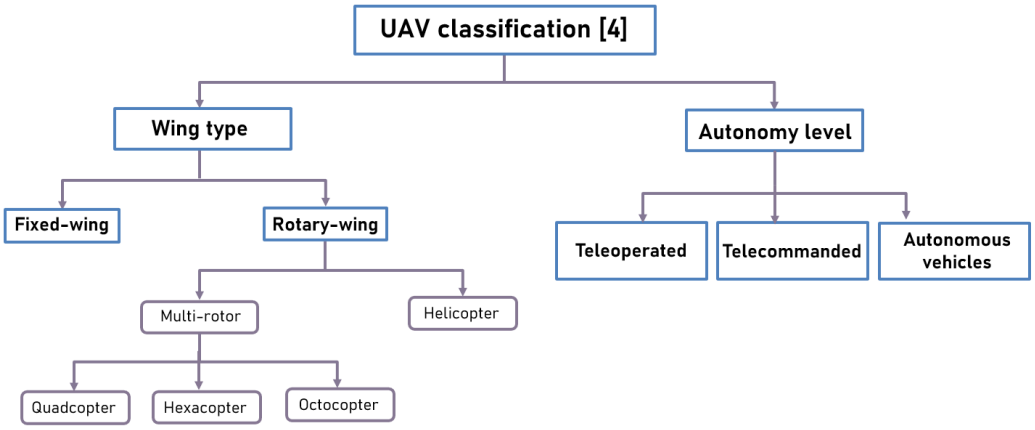


Figure 2.1: High-level classification of the UAVs.

One of the important criteria for agricultural UAVs classification is the autonomy level that impacts ease of operability, and can categorized into:

- Teleoperated vehicles. Teleoperated UAVs obtain references for each actuator from the pilot, such that the control resembles the one with an onboard pilot.
- Telecommanded vehicles. Telecommanded aircraft relies on an automatic controller on board to

maintain a stable flight. The vehicle trajectory, as well as the velocity and orientation commands are, on the other hand, defined by the ground operator.

- Autonomous vehicles. Autonomous UAV corresponds to the highest level of autonomy as it is able to perform a flight plan without the human intervention. The flight plan includes an ordered list of waypoints in form of four coordinates, relying on the GPS system on board the UAV. Provided coordinates that define the 3D environment are: longitude, latitude, altitude and heading angle (typically the deviation with respect to the North). Nonetheless, take-off and landing have not yet been fully autonomous in practice, due to number of requirements to establish safety, e.g. cleared accessible area, and therefore require assistance of the operator.

Furthermore, two main groups of UAVs that can be identified based on their wing type are fixed-wing (Figure 2.2a) and rotary-wing UAVs (Figure 2.2b). Their characteristics and differences are largely affected by aerodynamic principles.

(a) (b)

Figure 2.2: Wing-type classification of the UAVs: (a) Fixed-wing UAV; (b) Rotary-wing UAV (right-hand side).

Fixed-wing aircrafts, including airplanes, create airflow, hence the lift force, by shifting the aerodynamic surfaces (wings and ailerons) at high velocities. Therefore, they require flying at high altitudes to ensure a safe flight.

On the other hand, rotary-wing UAVs include helicopters and multi-rotors, that generate thrust in their rotors to create airflow for lifting the UAV. Because of their ability to perform hovering as well as vertical climb and descend, they are suitable for aerial photography or mapping as the capturing time can be increased to compensate for possibly poor weather and light conditions. Additionally, they can operate at low velocities which can minimize flight risks. Moreover, increased maneuverability is enabled by rotors performing the immediate rotations on the vertical axis. Multi-rotor UAVs are

further classified and named based on the number of rotors they have. A quadrotor or quadcopter has four rotors, the hexacopter has six rotors, whereas the octocopter has eight rotors.

In terms of payload capacity, as well as the flying range and endurance, fixed-wing UAVs exhibit higher capabilities. Nevertheless, multirotors, due to their high level of maneuverability, stability and ease of operation, have been predominantly used for civil applications, including agriculture, with the most common implementation of the quadrotors [34, 92].

Because of their above-mentioned suitability for application in mapping an agricultural field, quadrotors are considered the adequate UAV type hereafter. Therefore, UAV modeling will be focused solely on quadrotors.

2.2 Quadrotor mathematical model

A quadrotor or quadcopter is a type of unmanned aerial vehicle (UAV) that features two pairs of counter-rotating rotors and propellers positioned at the corners of a square frame, as illustrated in Figure 2.3. Similarly to a helicopter, it can execute vertical take-off and landing (VTOL), as well as hold a constant position while hovering.

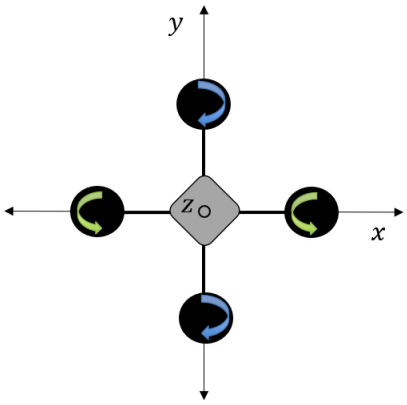


Figure 2.3: Quadrotor model with two pairs of counter-rotating rotors.

2.2.1 Kinematics

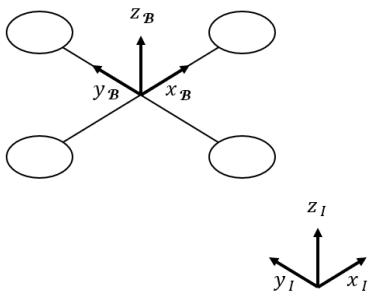


Figure 2.4: Reference frames for describing the quadrotor motions: Inertial and Body frame.

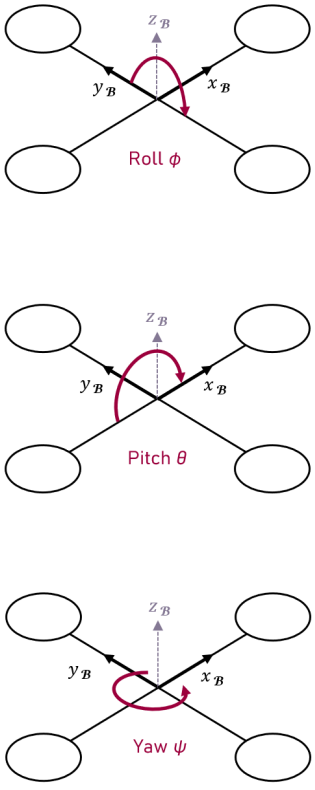


Figure 2.5: Euler angles, corresponding to rotations: roll ϕ, pitch θ and yaw ψ.

Kinematic model of a quadrotor describes the motion, i.e. changes in position and velocity in space over time. Defining the coordinate system is the first step in defining the relations in space. Two right-hand reference frames for describing motion of a quadrotor, shown in Figure 2.4, are the following:

- **Inertial frame $\mathcal{I}$:** earth-fixed coordinate system; defined with axes $x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}$, with $z_{\mathcal{I}}$ normally pointing upwards.

- **Body frame $\mathcal{B}$:** coordinate system attached to the quadrotor with axes $x_\mathcal{B}, y_\mathcal{B}, z_\mathcal{B}$. Center of the mass of the quadrotor corresponds to the origin of the body fixed frame.

Position vector $\boldsymbol{\xi}$ contains linear position vector $\mathbf{x}_\mathcal{I} = [x \;\; y \;\; z]^T$ and orientation $\boldsymbol{\eta}_\mathcal{I} = [\phi \;\; \theta \;\; \psi]^T$, both indicated with respect to inertial frame, as in (2.1).

$$\boldsymbol{\xi} = [\;\; \mathbf{x}_\mathcal{I} \;\; \boldsymbol{\eta}_\mathcal{I} \;\;]^T = [\;\; x \;\; y \;\; z \;\; \phi \;\; \theta \;\; \psi \;\;]^T \tag{2.1}$$

Linear velocity $\mathbf{v}_\mathcal{B} = [u \;\; v \;\; w]^T$ and angular velocity $\boldsymbol{\omega}_\mathcal{B} = [p \;\; q \;\; r]^T$ are expressed in the body frame, and contained in the vector $\boldsymbol{\nu}$, as in (2.2).

$$\boldsymbol{\nu} = [\;\; \mathbf{v}_\mathcal{B} \;\; \boldsymbol{\omega}_\mathcal{B} \;\;]^T = [\;\; u \;\; v \;\; w \;\; p \;\; q \;\; r \;\;]^T \tag{2.2}$$

Attitude of a UAV is described by Euler angles, corresponding to its orientation, which consists of three successive rotations, given in equations (2.3)-(2.5) and illustrated in Figure 2.5:

- Rotation $\mathbf{R}_\psi$ around the z-axis, called *yaw* or *heading*:

$$\mathbf{R}_\psi = \begin{bmatrix} cos\,\psi & -sin\,\psi & 0 \\ sin\,\psi & cos\,\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.3}$$

- Rotation $\mathbf{R}_\theta$ around the y-axis, called *pitch*:

$$\mathbf{R}_\theta = \begin{bmatrix} cos\,\theta & 0 & sin\,\theta \\ 0 & 1 & 0 \\ -sin\,\theta & 0 & cos\,\theta \end{bmatrix}, \tag{2.4}$$

- Rotation $\mathbf{R}_\phi$ around the x-axis, called *roll*:

$$\mathbf{R}_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\,\phi & -sin\,\phi \\ 0 & sin\,\phi & cos\,\phi \end{bmatrix}. \tag{2.5}$$

Once the reference frames are defined, the following transformation from one frame to the other determines the relation between the two frames:

$$\dot{\boldsymbol{\xi}} = \mathbf{J}_{\Theta}\boldsymbol{\nu} \tag{2.6}$$

In (2.6), $\mathbf{J}_{\Theta}$ represents the transformation matrix (2.7), with $\mathbf{R}$ being complete rotation matrix (2.8), $\mathbf{T}$ transfer matrix (2.9), and $\mathbf{0}_{3\times3}$ as a 3×3 zero matrix.

$$\mathbf{J}_{\Theta} = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{T} \end{bmatrix} \tag{2.7}$$

$$\mathbf{R} = \mathbf{R}_{\psi}\mathbf{R}_{\theta}\mathbf{R}_{\phi} = \begin{bmatrix} cos\theta cos\psi & sin\phi sin\theta cos\psi - cos\phi sin\psi & cos\phi sin\theta cos\psi + sin\phi sin\psi \\ cos\theta sin\psi & sin\phi sin\theta sin\psi + cos\phi cos\psi & cos\phi sin\theta sin\psi - sin\phi cos\psi \\ -sin\theta & sin\phi cos\theta & cos\phi cos\theta \end{bmatrix} \tag{2.8}$$

$$\mathbf{T} = \begin{bmatrix} 1 & sin\ \phi\ tan\ \theta & cos\ \phi\ tan\ \theta \\ 0 & cos\ \phi & -sin\ \phi \\ 0 & sin\ \phi\ /cos\ \theta & cos\ \phi\ /cos\ \theta \end{bmatrix} \tag{2.9}$$

Finally, the kinematic model of the quadrotor transformed into the inertial frame is given as in [93]:

$$\begin{cases} \dot{x} = w[s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta)] - v[c(\phi)s(\psi) - c(\psi)s(\phi)s(\theta)] + u[c(\psi)c(\theta)] \\ \dot{y} = v[c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta)] - w[c(\psi)s(\phi) - c(\phi)s(\psi)s(\theta)] + u[c(\theta)s(\psi)] \\ \dot{z} = w[c(\phi)c(\theta)] - u[s(\theta)] + v[c(\theta)s(\phi)] \\ \dot{\phi} = p + r[c(\phi)t(\theta)] + q[s(\phi)t(\theta)] \\ \dot{\theta} = q[c(\phi)] - r[s(\phi)] \\ \dot{\psi} = r\frac{c(\phi)}{c(\theta)} + q\frac{s(\phi)}{c(\theta)} \end{cases}, \tag{2.10}$$

where s denotes the sine, c the cosine, and t the tangent.

2.2.2 Rigid body dynamics

Assuming a quadrotor to be a rigid body, its dynamics need to factor in the effect of forces and torques acting on the vehicle.

Quadrotor models developed in this section do not include the effect of the disturbances and are thus considered nominal.

Following the Newton-Euler formalism, the dynamics of a rigid body under external forces applied

to the center of mass and expressed in the body fixed frame are defined as [94]:

$$\begin{bmatrix} mI_{3\times 3} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_{\mathcal{B}} \\ \dot{\boldsymbol{\omega}}_{\mathcal{B}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}_{\mathcal{B}} \times m\mathbf{v}_{\mathcal{B}} \\ \boldsymbol{\omega}_{\mathcal{B}} \times I\boldsymbol{\omega}_{\mathcal{B}} \end{bmatrix} = \begin{bmatrix} F \\ \tau \end{bmatrix}, \tag{2.11}$$

where m is the quadrotor mass, τ and F the rotor resulting torque and forces, respectively. $I \in R^{3\times 3}$ is inertia matrix:

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{2.12}$$

As the quadrotor is essentially symmetric about all three axes, therefore it is assumed that $I_{xy} = I_{xz} = I_{yz} = 0$, which implies:

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{2.13}$$

Following from (2.11), the first-level nonlinear dynamics of a quadrotor is given by equations (2.14-2.15):

$$F_{\mathcal{B}} = m\dot{\mathbf{v}}_{\mathcal{B}} = \boldsymbol{\omega}_{\mathcal{B}} \times (m\mathbf{v}_{\mathcal{B}}), \tag{2.14}$$

$$\tau_{\mathcal{B}} = I\dot{\boldsymbol{\omega}}_{\mathcal{B}} + \boldsymbol{\omega}_{\mathcal{B}} \times (I\boldsymbol{\omega}_{\mathcal{B}}), \tag{2.15}$$

with the inertial matrix I considered diagonal because of the symmetry of the vehicle. Vectors $F_{\mathcal{B}}$ and $\tau_{\mathcal{B}}$ constitute total forces and moments applied from each of the four rotors. The total angular moment of the rotors is assumed to be near zero, as the moments from the counter-rotating rotor pairs cancel when the yaw is held steady [95].

Torques generated by individual rotors $\Omega = [\Omega_1\ \Omega_2\ \Omega_3\ \Omega_4]^T$ enable control inputs $U = [U_1\ U_2\ U_3\ U_4]^T$, being thrust, roll, pitch and yaw, respectively, according to relations in the equations (2.16).

$$\begin{cases} U_1 = b\,(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 = bl\,(\Omega_4^2 - \Omega_2^2) \\ U_3 = bl\,(\Omega_3^2 - \Omega_1^2) \\ U_4 = d\,\Omega \end{cases} \tag{2.16}$$

In 2.16, b depicts the thrust coefficient, d drag coefficient, and l distance between the center of the quadrotor and the center of the rotor. Furthermore, the overall residual rotor angular speed Ω, considered in the gyroscopic torque as the quadrotor rolls or pitches is defined as:

$$\Omega = \Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2 \tag{2.17}$$

The full dynamics of a quadrotor in the inertial frame, considering its rotations, is modeled as in [96]:

$$\begin{cases} \ddot{x} = (cos\phi\ sin\theta\ cos\psi + sin\phi\ sin\psi)\ \frac{1}{m}\ U_1 \\ \ddot{y} = (cos\phi\ sin\theta\ sin\psi + sin\phi\ cos\psi)\ \frac{1}{m}\ U_1 \\ \ddot{z} = -g + (cos\phi\ cos\theta)\ \frac{1}{m}\ U_1 \\ \ddot{\phi} = \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) - \frac{J_r}{I_x}\dot{\theta}\Omega + \frac{l}{I_x}U_2 \\ \ddot{\theta} = \dot{\theta}\dot{\psi}\left(\frac{I_z - I_x}{I_y}\right) - \frac{J_r}{I_y}\dot{\phi}\Omega + \frac{l}{I_y}U_3 \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) + \frac{1}{I_z}U_4 \end{cases}, \tag{2.18}$$

with $I_{x,y,z}$ being body inertia and J_r rotor inertia. Accelerations in 3 axes $\ddot{x}$, $\ddot{y}$ and $\ddot{z}$ are modeled with Euler transformation and are influenced by thrust, denoted as U_1. Additionally, vertical acceleration $\ddot{z}$ needs to overcome the effects of gravitational acceleration g. Desired roll, pitch and yaw are reached by generating appropriate combination of rotor torques, as in relations $U \leftrightarrow \Omega$ in (2.16).

2.3 Energy consumption modeling

In order to efficiently plan an energy-aware mission for the UAVs, its performance evaluation needs to consider energy consumption. When it comes to UAVs, there are two possible approaches for energy consumption monitoring and prediction: based on the model of battery discharge, or power consumption linked with the aerodynamic principles.

2.3.1 Battery discharge model

Lithium-ion (Li-ion) batteries are the most commonly used rechargeable energy storage for autonomous vehicles, including UAVs [97]:

- Li-ion batteries hold a high energy density due to their electrodes made of lightweight lithium and carbon, in comparison to other chemical composition, such as lead-acid, NiCd (nickel-cadmium) or NiMH (nickel-metal hydride).
- Their low self-discharge rate leads to holding the charge for a longer time.
- Because of their long life cycle, the Li-ion batteries can endure hundreds of charge and discharge cycles without significant degradation in capacity.

The total available flight time available for a battery is reduced with the aging of the battery due to, for example, high-rate cycles, overcharge, overdischarge, etc. In order to precisely estimate the remaining battery capacity, it is necessary to know current battery conditions that can be represented by different indicators, such as state-of-charge (SoC). SoC of a battery or a single cell is a proportion of the charge available at a given time instant t. It is expressed as a percentage of the total charge of a fully charged battery. Because of its ease of implementation, the most commonly used method for calculating SoC is the Coulomb-counting [98]:

$$SoC(t) = SoC(t_0) - \frac{1}{3600 C_T} \int_{t_0}^{t} I_{batt}(t) dt, \tag{2.19}$$

where t_0, represents the initial time in seconds $[s]$, and C_T nominal capacity of the battery in Ampere-hours $[Ah]$ converted into seconds.

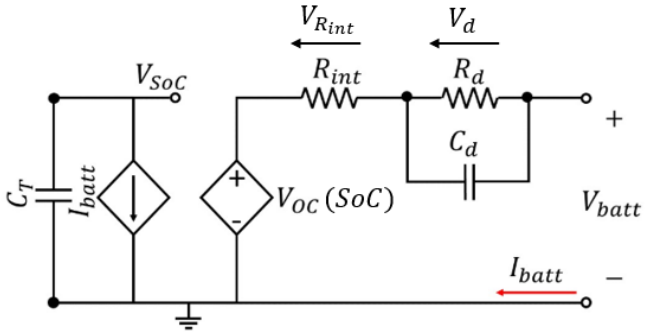


Figure 2.6: Electric circuit representation of a battery.

In the mathematical model representing the dynamical behaviour of battery types in use for the UAVs, such as Li-Ion, Li-Po or NiMh, the total voltage battery V_{batt} is a difference between the open voltage circuit $V_{OC}(SoC)$ and two voltage drops (Figure 2.6): voltage drop due to the internal battery resistance $V_{R_{int}}$ and dynamic response of the voltage V_d. Equations (2.20-2.23) illustrate the battery mathematical model:

$$V_{batt}(t) = V_{OC}(SoC(t)) - V_{R_{int}}(t) - V_d(t), \tag{2.20}$$

$$\dot{V}_{SoC}(t) = -\frac{I_{batt}(t)}{C_T}, \tag{2.21}$$

$$\dot{V}_d(t) = -\frac{1}{R_d C_d} V_d(t) + \frac{I_{batt}(t)}{C_d}, \tag{2.22}$$

$$V_{R_{int}}(t) = R_{int}\, I_{batt}(t), \tag{2.23}$$

where V_{SoC} represents the state-of-the charge voltage, R_{int} internal resistance of the battery, and parameters R_d and C_d are associated with the voltage dynamic response.

Evaluating and predicting the battery capacity is imperative for successful UAV mission completion. Accurate and precise measurement of the state-of-charge of the battery can be, therefore, helpful to support mission planning.

2.3.2 Aerodynamics-based energy consumption

According to [98], the largest part of the power consumed by a UAV, i.e. around of 85%, is used by the propulsion system to generate sufficient thrust for take-off, horizontal flight, hovering and landing. The remaining power is consumed by various sensors (GPS, accelerometers, compass, gyroscope), computer on board and payload, including cameras.

Total energy consumed by a UAV is intended to counteract the effects of gravity and drag forces due to forward motions and wind. The speed of each of the rotors of a multi-rotor (here: a quadrotor) is adjusted to achieve the necessary thrust to hover or move at the desired velocity. Mathematical model of the UAV energy consumption in this chapter is derived from [99]. The total required thrust for a UAV is therefore defined as:

$$T = (m_{body} + m_{batt} + m_{load})\, g + F_{drag}, \tag{2.24}$$

where m_{body}, m_{batt}, and m_{load} are the mass of the quadrotor body, battery, and payload, respectively, if any. Gravity is denoted as g. For further study, no payload will be considered, and the quadro-

tor body and battery will be joined as one component, total mass m. Drag force is denoted as F_{drag}, and is estimated as:

$$F_{drag} = \frac{1}{2}\rho v_a^2 C_D A, \tag{2.25}$$

with ρ as air density, v_a air speed, C_D the drag coefficient, and A projected area perpendicular to direction of the airspeed v_a.

In general, the minimum power needed for a steady-state hovering of a multi-rotor with n rotors of D diameter, without considering wind velocity, is given by:

$$P_{min,hov} = \frac{T^{3/2}}{\sqrt{\frac{1}{2}\pi n D^2 \rho}}, \tag{2.26}$$

On the other hand, when the UAV is flying at a certain velocity, as well as considering wind effects on the vehicle velocity, the minimum power for a horizontal flight is derived from the conservation of momentum. For small angles of attack σ, the following equation for minimum required power holds:

$$P_{min} = T(v_a \sin\sigma + v_i) \tag{2.27}$$

Induced velocity v_i required for a given thrust is found by solving the implicit equation below:

$$v_i = \frac{2T}{\pi n D^2 \rho \sqrt{(v_a \cos\sigma)^2 + (v_a \sin\sigma + v_i)^2}}. \tag{2.28}$$

Finally, a full nonlinear model of the power consumption during a quadrotor flight, considering aerodynamic aspects of a UAV flight, including induced, profile and parasite power, is obtained from [100]. The equations (2.29-2.31) represent power required for different phases of the flight, namely: P_{vert} for vertical take-off/landing, P_{hor} horizontal cruise and hovering P_{hov}.

- Power required for vertical take-off and landing:

$$P_{vert}(m, v_{vert}) = k_1 mg \left[\frac{v_{vert}}{2} + \sqrt{(\frac{v_{vert}}{2})^2 + \frac{mg}{k_2^2}}\right] + c_2(mg)^{3/2}, \tag{2.29}$$

- Power required for horizontal cruise:

$$P_{hor}(m, v_a) = (c_1 + c_2)[(mg - c_5(v_a \cos\sigma)^2)^2 + (c_4 v_a^2)^2]^{3/4} + c_4 v_a^3, \tag{2.30}$$

- Power required for hovering:

$$P_{hov}(m) = (c_1 + c_2)(mg)^{3/2}. \tag{2.31}$$

In the equations above, parameters k_1, k_2, c_1, c_2, c_4, c_5 are to be identified experimentally. From (2.30), it can be deduced that, for low values of the angle of attack, the power required for the horizontal flight is constant as the vehicle airspeed is also kept constant.

2.4 Considered quadrotor model

2.4.1 Dynamic model

For designing a suitable model-based control strategy for trajectory tracking later in the thesis, a nonlinear dynamic model of a quadrotor needs to map reliable quadrotor dynamics and, at the same time, provides a needed level of simplicity from the computational complexity perspective.

The nonlinear model of a quadrotor with 6 DOF is a simplification of the model in (2.18) and is described by the set of equations, similar to [101]:

$$\dot{p}(t) = v(t), \tag{2.32}$$

$$\dot{v}(t) = \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \alpha T \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} - \begin{bmatrix} A_x & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & A_z \end{bmatrix} v(t) + \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix}, \tag{2.33}$$

$$\dot{\phi}(t) = (K_\phi \phi_{ref}(t) - \phi(t))/\tau_\phi, \tag{2.34}$$

$$\dot{\theta}(t) = (K_\theta \theta_{ref}(t) - \theta(t))/\tau_\theta, \tag{2.35}$$

with rotation matrix $R \in SO(3)$, representing the orientation of the body-fixed reference frame with respect to the inertial reference frame (eq. (2.8)). Position is defined with the vector $p = [p_x, p_y, p_z]^\top$ and velocity with $v = [v_x, v_y, v_z]^\top$, both in the yaw-compensated inertial reference frame with a defined origin. Accordingly, the yaw angle, ψ, is regulated to zero with an inner-loop P-controller as in [101], and therefore not considered in the model for the trajectory tracking control. The full control scheme will be detailed in Chapter 4.

The attitude is denoted with ϕ, representing the roll angle, and θ pitch angle of the UAV in the inertial coordinate system. Yaw angle, ψ, is set to zero in the model equations, as previously mentioned. Vector $d = [d_x, d_y, d_z]^\top$ representing the external disturbances, such as wind acting on a UAV as torque, is added to the quadrotor system. Thruster efficiency α defines the model uncertainty parameter. It is equal to 1 in the nominal case, greater than one in case of underestimating real thruster efficiency, and less than 1 for its overestimation. Other parameters represent gravity, g, linear dumping terms, A_x, A_y, A_z, along each axis respectively, time constants, τ_ϕ, τ_θ, as well as the inner-loop gains for the attitude control, K_ϕ, K_θ. Control input vector, $u = [T, \phi_{ref}, \theta_{ref}]^\top$, consists of the total thrust T, reference roll ϕ_{ref}, and reference pitch θ_{ref} angles.

2.4.2 Energy consumption model

For the purpose of mapping an agricultural field, the mission plan needs to consider the energy consumption model of a UAV battery in order to prevent undesirable battery issues and ensure mission success.

With the aim of obtaining high-quality aerial images, the UAV velocity is supposed to be held constant. As the objective of the mapping mission planning is to cover a certain area of interest, the horizontal flight segments will be considered in this phase.

Following Eq. (2.30), it can be concluded that, for constant mass and parameter values, as well as the small angles of attack σ, assumed to be constant, the power consumption of the horizontal flight depends solely on the quadrotor velocity in a nonlinear manner. Furthermore, if the velocity remains constant at a value $v = v_c$, the power consumption during the horizontal flight for performing the mapping task, $P_{map} = P_{hor}$, is also constant over time, as shown in:

$$P_{map}(m, v_c) = (c_1 + c_2)[(mg - c_5(v_c \cos\sigma)^2)^2 + (c_4 v_c^2)^2]^{3/4} + c_4 v_c^3. \tag{2.36}$$

Based on the relationship:

$$P_{map} = \frac{E_{map}}{t_{map}}, \tag{2.37}$$

for the constant values of power consumption for mapping P_{map}, and energy E_{map}, it can be assumed that the battery consumption is represented by the time elapsed during the mapping task t_{map}. This assumption will be used in energy-aware mapping mission planning presented in Chapter 3.

2.5 Conclusion

Mathematical modeling of a UAV is essential for its accurate representation. Depending on the implementation and type of the UAV, different aspects need to be considered. For this thesis, the quadrotor model will be considered for further study. In order to develop an adequate control system, quadrotor dynamics needs to be modeled with sufficient accuracy, but also simplifications made with valid assumptions, with the aim of ensuring robustness. Furthermore, estimation of the power consumption is crucial for ensuring mission completion and safety, as well as to plan the optimal energy-aware mission. Eventually, to ensure the successful implementation of the designed concepts, they first need to be validated on an accurate model.

Chapter 3

Energy-aware path planning for a UAV mapping mission

3.1 Introduction

When it comes to planning a mapping mission performed by a UAV equipped with a camera, numerous aspects need to be considered in order to accomplish a given mission successfully. Besides the time efficiency, in planning an optimal mission, it is necessary to consider an energy-aware mission [9], namely due to major constraints linked to the limited battery capacity. Given large surfaces that need to be visited in order to map an entire agricultural field with limited resources, in terms of the number of available UAVs and batteries, the mission needs to be planned beforehand to guarantee its completion [26]. From the logistics perspective, it is essential to plan the mission accurately in advance so that it can be executed within the scheduled timeline without the need for important modifications. This also involves preparing all the necessary equipment, including batteries.

An energy-aware mission for UAVs implies completing the assigned task with the least possible energy consumption. Moreover, it is a mission planning approach that prioritizes the efficient use of energy resources to maximize flight time, achieve mission objectives effectively, and ensure safe return or landing. It involves integrating strategies and technologies that monitor, manage, and optimize energy consumption throughout the mission. Lowering the energy consumption consequently results in direct cost cuts, as well as preventing battery wear by reducing unnecessary recharging cycles.

There are two components of an energy-aware mission planning for the UAVs:

- **Battery Management:** It implies monitoring, prediction, and optimization of the battery discharge. Monitoring is a preliminary step that includes keeping continuous track of battery health, charge levels, and discharge rates. Prediction is crucial as it assumes estimating remaining flight time based on current consumption rates and mission requirements. Finally, improvements to the mission plan can be made with the optimization approach in terms of implementing efficient battery use practices. To address the energy constraints of a single UAV, it is essential to solve the task allocation problem by considering accurate battery parameters. Battery management considers subpaths as tasks that need to be allocated to multiple available batteries to ensure complete area coverage.

- **Path planning:** Optimal path calculation is a critical component of an energy-aware mission, aiming to determine the most energy-efficient route for a UAV to follow. This involves minimizing energy consumption while meeting the mission objectives, such as reaching defined waypoints, avoiding obstacles, or covering a designated area. An advanced step in path planning would consider adaptive navigation, i.e., adjusting the initially planned path in real-time based on dynamic conditions to conserve energy.

Due to the limited energy capacity of UAV batteries, it is crucial to schedule battery replacements in a manner that ensures the entire agricultural field is comprehensively covered. The battery capacities available for any given mission can vary significantly from one battery to another, as well as based on the specific conditions under which the mapping is conducted. Therefore, even when dealing with the same mapping field configuration, it is essential to plan battery replacements and waypoint allocations for each mission individually, using the most up-to-date information about the battery capacities. Variation in battery capacities arises from the use of diverse batteries, possibly from various manufacturers or with different usage histories, which can have varied capacities impacting their performance and endurance. The actual capacity and performance of batteries can also be influenced by specific conditions encountered during the mission.

Hence, strategic scheduling of battery replacements involves planning the timing and location of replacements to minimize disruptions and ensure continuous coverage of an agricultural field. Efficiency can be enhanced by reducing the frequency and duration of battery replacements, which contributes to the overall effectiveness of the mission. Waypoint allocation should be based on real-time data on battery levels to ensure efficient coverage without unnecessary detours. In addition, optimizing battery usage involves matching battery capacity with the mission-specific requirements, by

selecting batteries that are adequately charged and have the necessary capacity. This approach helps avoid mid-mission battery depletion. Energy-efficient flight paths should be designed to maximize coverage while minimizing energy consumption, avoiding redundant routes and unnecessary maneuvers. Reducing total flight time can be achieved by minimizing trips to the base station. Employing batteries with sufficient capacity and reducing the need for replacements helps eliminate unnecessary trips, thereby reducing overall flight time. Proximity to the recharging base station is also a crucial consideration to reduce the distance UAVs need to travel for battery replacements, thus minimizing total flight time.

Comprehensive battery assessment and route simulation during pre-mission planning are preceding steps to the practical implementation strategies. Conducting thorough checks of all batteries to determine their capacities and performance metrics is crucial. Running simulations is necessary to identify the most energy-efficient routes and optimal schedules for battery replacements, considering various scenarios and conditions.

An optimized energy-aware mission for the implementation of UAVs in agricultural mapping has multiple benefits. Efficient battery management and replacement scheduling can significantly extend the operational time of both the UAV and batteries, allowing for a longer life cycle. Additionally, cost savings can be achieved by reducing the number of battery replacements and minimizing unnecessary travel, which conserves energy and reduces operational costs. Mission success likelihood is increased as a UAV with sufficient energy can complete its tasks more reliably, improving data accuracy and coverage. Finally, optimized energy usage reduces the environmental footprint of the mapping mission by minimizing energy waste and enhancing overall operational efficiency.

In this study, battery management follows after path planning is done for the entire agricultural field, in order to address both aspects of energy-aware mission planning. The objective is to ensure a mapping mission that is completed with minimum resources used regarding energy and time. Furthermore, mission safety is imperative. Thus, the mission has to be planned such that the risk level is reduced during the entire flight of a UAV. Risks include any obstruction to the mission completion, such as insufficient battery level, UAV crash, etc.

The flowchart in Figure 3.1 structures the mission planning steps for an energy-aware mapping mission. The first phase contains two-step path planning, where area decomposition is followed by path configuration definition. Once the area is decomposed into ordered waypoints, task allocation is done based on the battery management principles. The energy consumption and mission time are minimized by determining the optimal battery replacement schedule, as well as a choice of the base

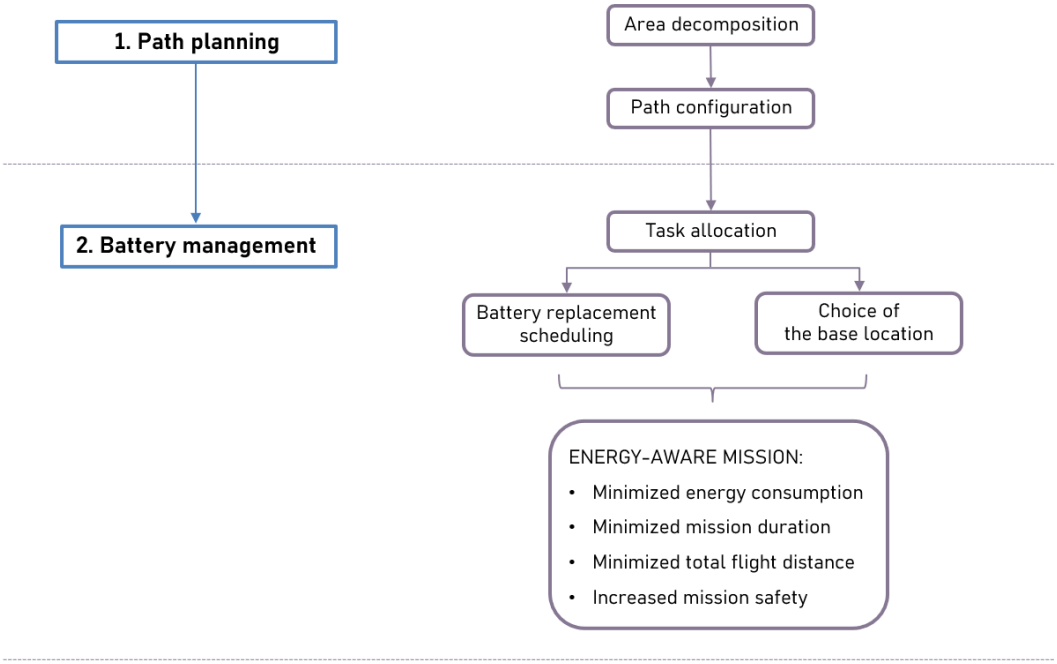


Figure 3.1: Energy-aware mapping mission planning flowchart

location to ensure smooth mission execution without unnecessary losses in terms of time and energy. Nonetheless, such choice of the base location and replacement schedule impacts the increased mission safety.

3.2 Path planning for a mapping mission

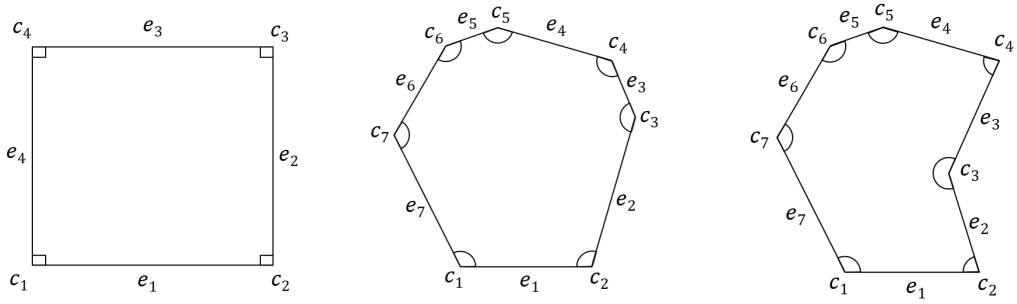
The first part of mission planning considers path planning, where the entire field is transformed into a single path, without considering battery limitations. This includes two steps: area decomposition and path configuration. Area decomposition has the objective to define waypoints that need to be visited to obtain a full and accurate map of the field. Once the area is decomposed into cells and waypoints are appointed as centers of the cell, path configuration is chosen such that the waypoints are linked in an optimal way, and the resulting path indicates a reference trajectory for the UAV flight.

Once the path planning has generated a single path that covers the complete field area, battery management will be used for task allocation to multiple batteries and/or UAVs, taking into account energy constraints.

3.2.1 Area decomposition

The first step in the mapping mission planning is to build a graph that represents the agricultural field, i.e. area of interest. This can be done by decomposing the field area into cells, which results in a full map of the area.

The area of interest is represented by a polygon with a sequence of p vertices $\{c_1, ..., c_p\}$ and p edges $\{e_1, ..., e_p\}$. Each vertex is described by three coordinates in 3D space $(c_x(i), c_y(i), c_z(i))$. An edge links two adjacent vertices, and its length can be expressed as $\|c_i - c_{i+1}\|$. The polygon can be of a convex or nonconvex (concave) shape, as in Figure 3.2.



(a) Rectangular polygon area (b) Convex polygon area (c) Concave polygon area

Figure 3.2: Convex and nonconvex polygon field area examples.

When decomposing an area of interest into cells for mapping purposes, it is necessary to consider UAV and camera characteristics, such as camera footprint, in order to ensure the required overlapping of snapshots taken at each waypoint. This is done namely by determining the flight altitude, as well as the number of waypoints and distance between each of them. Each waypoint is a node in a graph that represents the center of a cell and is linked with another one with a straight segment line. In addition to the waypoints denoting the points of interest for mapping, there is an additional node that designates the recharging base, and it acts as a departing and returning point for a UAV flight.

3.2.2 Flight path configuration

The graph, composed of all the waypoints covering the field, is created by linking the adjacent waypoints in a predetermined order based on the chosen path configuration, as well as constructing a two-way link between each waypoint node and the recharging base node. Figure 3.3 illustrates a portion of the graph with nodes representing ordered waypoints, where each waypoint k is linked to

waypoint $k+1$ with a segment line k. Moreover, every waypoint k is connected to the base node with line segments u_k and v_k, representing departing and returning segments, respectively.

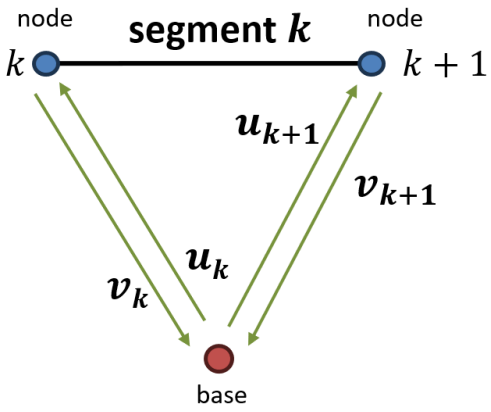


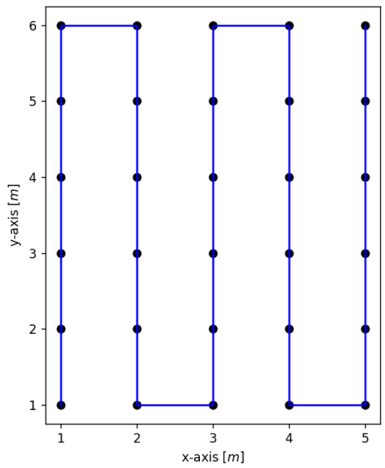
Figure 3.3: Graph composition: nodes k and $k+1$ linked with the segment line k and; segments u_k, u_{k+1} as departing and segments v_k, v_{k+1} as returning segments.

Path configuration defines the arrangement of the waypoints and results in an ordered list of coordinates, denoting the sequence of points that a UAV needs to visit. The order of the waypoints can affect the overall performance of the mapping mission. For example, differently defined edges or links on the graph directly impact the path length, hence, the energy consumption and mission completion guarantee. As stated in [2], the total energy consumed during a UAV flight largely depends on the turning maneuvers in the path. Therefore, energy-aware missions are planned such that the number of turns is minimized.

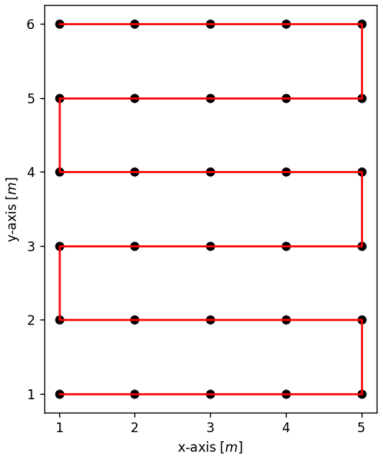
Also, the order of the waypoints outlines the temporal sequence of taking the snapshots, resulting in a certain level of uniformity and accuracy once all the images are consolidated into a map. Thus, the choice of an appropriate path configuration represents a pre-requirement to ensure the optimal mapping mission.

Once the area decomposition is done and the full area of interest is divided into waypoints, the waypoint ordering needs to be done in a way that provides the optimal flight path configuration. Optimal flight configuration ensures the minimum total flight distance resulting in minimum flight time and, consequently, minimum energy consumed. As introduced in the Chapter 1, most commonly implemented path configurations, or sweep directions, for agricultural mapping purposes are back-and-forth and circular flight patterns.

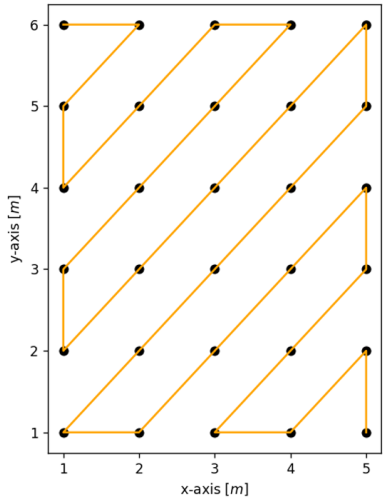
On a simplified rectangular example of field area $4m$ x $5m$, Figure 3.4 illustrates four possible



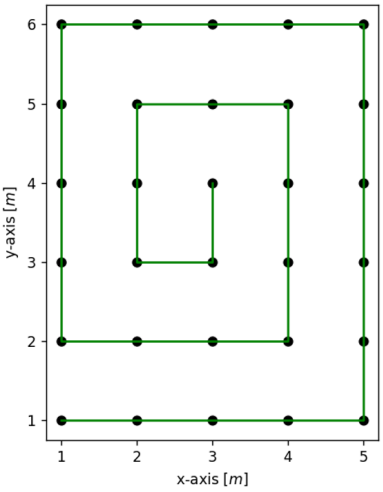
(a) Back-and-forth along the longest side



(b) Back-and-forth along the shortest side



(c) Back-and-forth along the diagonal



(d) Circular configuration

Figure 3.4: Flight path configuration possibilities for a rectangular field area $4m$ x $5m$.

configurations for the sweep direction of a mapping mission: back-and-forth movements along the longest side 3.4a, along the shortest side 3.4b, diagonal back-and-forth path 3.4c and circular path 3.4d. These configurations are evaluated in Table 3.1 in terms of total path distance and number of turns. It can be concluded that only the diagonal configuration differs in total path length that exceeds the other three possibilities. Regarding the number of turns, the back-and-forth direction along the longest side results in the minimum number of needed turning maneuvers. As these are the indicators of relative energy consumption, the optimal choice to reduce battery consumption in this case seems to be the back-and-forth path configuration along the longest field size. Additionally, this pattern facilitates a later image fusion, as the snapshots are taken in a sequence that follows the logical

order of map building.

Table 3.1: Resulting indicators for different path configurations of a rectangular field area $4m$ x $5m$.

	Total flight path length	Total number of turns
Back-and-forth along the longest side (Fig. 3.4a)	$29\ m$	8
Back-and-forth along the shortest side (Fig. 3.4b)	$29\ m$	10
Back-and-forth along the diagonal (Fig. 3.4c)	$37.28\ m$	16
Circular configuration (Fig. 3.4d)	$29\ m$	9

It is important to emphasize that the area decomposition and flight path configuration did not take into account preliminary battery limitations and can, therefore, be considered as optimal only if performed in a single flight. Often, there is only one UAV available to perform a given mission, and therefore, battery replacements or recharge need to be done along the mission. In order to respect the energy limitations of a single UAV, it is necessary to tackle the task allocation problem as a next step by allocating subpaths to individual batteries to complete area coverage. Battery management can facilitate proper task allocation with the objective of minimizing unexpected interruptions and increasing mission efficiency.

3.3 Battery management optimization for task allocation

In this work, the path planning phase, including area decomposition and flight path configuration, will not be further explored, as we will consider that the trajectory that links the waypoints is known and given beforehand. In order to complete the energy-aware mission planning as in the flowchart in Figure 3.1, once the path configuration and waypoint order are given, complete field coverage needs to be ensured by following an adequate management approach that considers the battery limitations.

3.3.1 Problem statement

As a preliminary step, the field area is decomposed into cells, resulting in a full area map, by considering the camera characteristics to define the flight altitude. Each cell has a centering waypoint that is connected to the next one with a straight segment line in a preordered sequence. The ordered sequence of the waypoints is defined by flight path configuration, such that the space coverage is performed with back-and-forth movements along the segment lines, as in the square field example in Figure 3.5.

The back-and-forth configuration of the mapping trajectory ensures reliable resulting images for photogrammetry purposes. In order to get a sufficient level of overlapping for image fusion and later image processing, back-and-forth movements provide the optimal results. Based on the required image resolution, camera characteristics, etc., it is a straightforward process to obtain such a UAV mapping trajectory, as the UAV moves along only one axis at once. That way, the obtained images are shifted along a single axis. The shift along the second axis is easily determined by knowing the distance between the mapping rows.

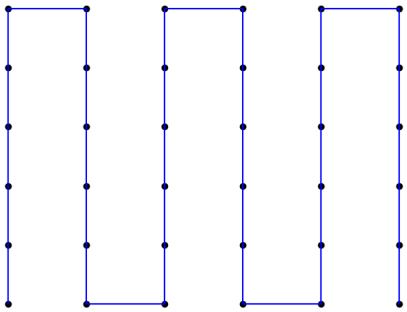


Figure 3.5: Example square field area coverage with back-and-forth movements.

Because of the limited energy capacity of a single battery on board the UAV, task allocation needs to consider battery replacements along the predefined path such that the whole agricultural field is covered. This planning phase includes segmenting the flight path into subpaths and allocating them according to available battery capacities. To enable an energy-aware mission, the subpath allocation needs to be performed in accordance with the appropriate battery management guidance that minimizes the energy consumed during the entire mission flight.

Battery management implies estimating accurate energy capacity for batteries available for the mission and optimizing their employment. Because capacities vary from one battery to another, battery replacements and waypoint allocation must be planned for each mission individually based on the latest battery capacity information. The objective of battery management is to select batteries for

the mission and schedule their replacement in an optimal manner to minimize the total flight time. Using only batteries with sufficient capacities and minimizing the number of replacements reduces the total flight time by eliminating unnecessary trips to the base station. This approach ensures that the UAV can cover more area per battery cycle, thereby enhancing the efficiency of the mission.

Moreover, selecting initial and final waypoints closer to the base further decreases the total flight distance and, consequently, the total flight time. Strategically positioning the waypoints near the base station not only saves energy but also allows for quicker battery replacements and less downtime. This is particularly important in large mapping areas where frequent returns to the base station could significantly increase the mission duration. By optimizing the battery usage and waypoint placement, the UAV can achieve maximum coverage with minimal interruptions, thus improving overall mission efficiency. Efficiently planned and executed energy-aware mapping mission minimizes both energy consumption and time. As the mission execution time is crucial for obtaining uniform images of the field, shorter mapping missions provide better final results for image fusion into a single map.

Another element of obtaining the optimal mission plan is to evaluate the recharging base station location choice. There are certain criteria that need to be examined before considering the candidate locations. First and foremost, position of the base needs to allow the operator easy access and sufficient space for equipment, as well as for the battery replacement task. Furthermore, terrain suitable for take-off and landing should be relatively flat, without significant incline, and without obstacles around the base. Thus, candidate base locations are commonly chosen at the extremities of the mapping area, as in the example in Figure 3.6. Finally, the best base location is chosen among the proposed potential locations, such that the above-mentioned criteria is satisfied and the total flight distance is minimized.

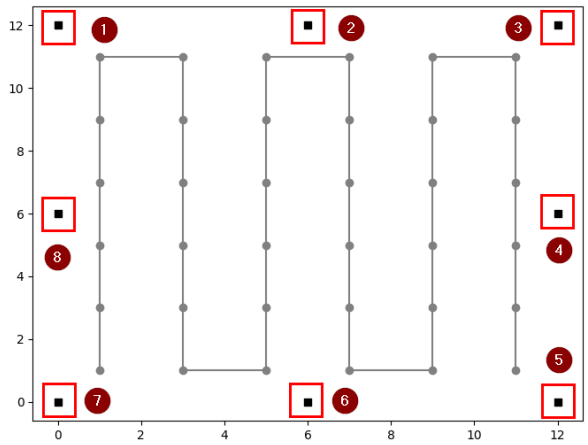


Figure 3.6: Example of the 8 candidate potential base locations at the extremities of a square field.

An additional important aspect of mission planning is flight safety. As take-offs and landings demonstrate the highest risk in the UAV flight, reducing their recurrence consequently increases mission safety. From the battery management perspective of the battery replacement scheduling, this can be done by lowering the number of batteries employed in a mapping mission.

In summary, battery management needs to ensure both an energy-aware mission as well as mission safety. Both aspects can be addressed by minimizing the number of battery replacements employed, which results from minimizing the total mission duration. Thus, the optimization strategy must define the subpath allocation, i.e., the battery replacement schedule and choice of the base location.

3.3.2 Optimization strategy

The battery management strategy proposed in this chapter aims to optimize the use of the available batteries for a given mapping mission. This is done by decomposing the mapping area into subpaths and allocating the associated waypoints to appropriate batteries simultaneously in a way that only the necessary number of eligible batteries is employed to complete the mission. This is done by minimizing the total flight distance, which includes mapping subpaths, as well as flights from and to the base station. Thus, unnecessary replacements are eliminated, along with flight time spent without performing the mapping task, which also improves mission safety since the take-off and landing represent high-risk parts of the flight plan.

3.3.2.1 Notation

The coverage problem is modeled as a graph with nodes representing the waypoints, i.e., points of interest that need to be visited by a UAV, and the segment lines as the edges of a graph, linking each waypoint to the adjacent one and the base. Waypoints are preordered to form the back-and-forth movement, and there is only one possible link between two successive waypoints.

Figure 3.7 illustrates the graph and introduces the notation that will be used to formulate the battery allocation problem mathematically. The binary variable x is defined for each segment-battery couple, which serves as a decision variable to determine whether a battery is employed to cover a certain segment or not (1 if used, 0 otherwise). Similarly, each node, i.e. waypoint, has a possible two-way link to the base location, where binary variables, denoted u and v, are introduced for each couple node-battery. The variable u equals 1 in case the considered waypoint marks the starting node of the subpath for the considered battery, while the variable v is set to 1 if the subpath ends with this particular node. Otherwise, these variables equal 0.

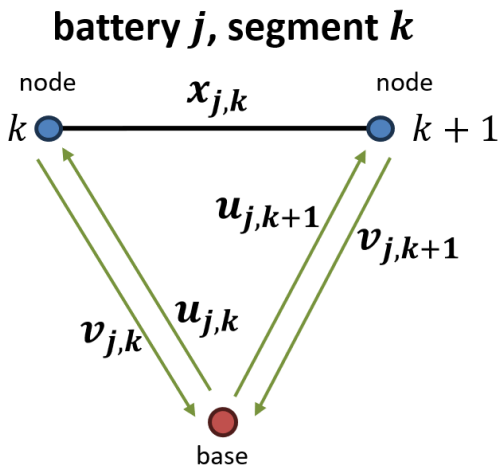


Figure 3.7: Definition of the used notation: segment k linking two successive waypoints and a two-way link to the base.

Table 3.2: Optimization parameters and decision variables

Optimization parameters	
$t_j\ [sec]$	Available capacity of battery $j \in \mathbb{J}$
$v_c\ [m/s]$	Constant velocity for mapping
$v_{max}\ [m/s]$	Maximum velocity for flying from/to base
$l_k\ [m]$	Length of the segment $k \in \{1, \ldots, K\}$
$l_{b,k}\ [m]$	Distance between the base and each waypoint $k \in \{1, \ldots, K+1\}$
Decision variables	
$x_{j,k}$	Allocation of the segment k to the battery j
$u_{j,k}$	Allocating node k as the first waypoint of the subpath for the battery j
$v_{j,k}$	Allocating node k as the last waypoint of the subpath for the battery j

The optimization strategy is defined for the mapping of $K+1$ waypoints joint by K segments, performed with J eligible batteries. As previously mentioned, for each battery $j \in \mathbb{J} = \{1, \ldots, J\}$, the decision variable $x_{j,k}$ determines allocation of the battery j to the segment $k \in \{1, \ldots, K\}$, joining waypoints k and $k+1$ (see Figure 3.7). Variables $u_{j,k}$ and $v_{j,k}$ represent the link of the node k from and to the base, respectively, and determine whether the node k is the first or last waypoint of a subpath allocated to the battery k. Similar logic is applied for the node $k+1$.

An example of an allocated subpath for battery j is illustrated in Figure 3.8. The associated subpath contains visiting N segments linking nodes in a preordered sequence from node #n to node #$(n+N)$, where $u_{jk} = 0,\ \forall k \neq n$, while $v_{jk} = 0,\ \forall k \neq n+N$, and $x_{jk} = 0,\ \forall k < n, \forall k \geq n+N$. Otherwise, $x_{jk}, u_{jk}, v_{jk} = 1$.

A structured list of parameters and decision variables for the considered battery management

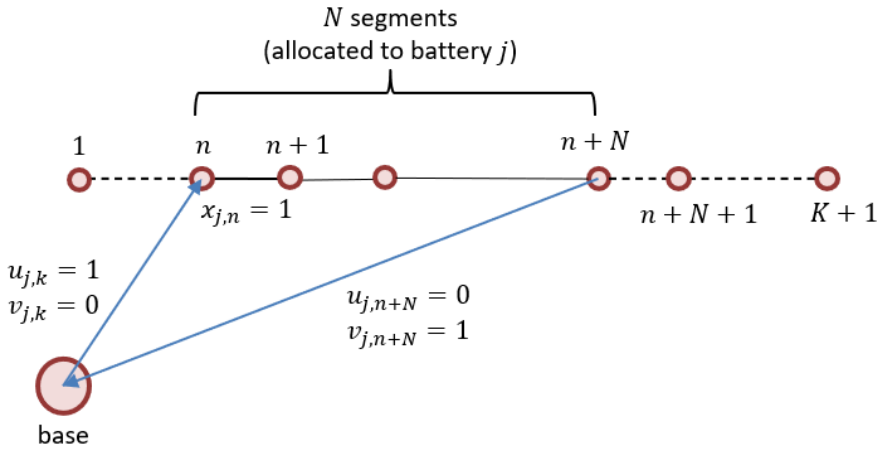


Figure 3.8: Example of a subpath with allocated waypoints $k \in [n...n+N]$ to battery j.

optimization problem is given in Table 3.2. Three types of binary decision variables are included in the optimization problem, for all the combinations between each battery and each segment: $x_{j,k} \in \{0,1\}$ for allocation of the segment $k \in \{1,\ldots,K\}$ to the battery $j \in \mathbb{J}$, variable $u_{j,k} \in \{0,1\}$ for determining whether the waypoint $k \in \{1,\ldots,K+1\}$ indicates the beginning of the subpath allocated to the battery $j \in \mathbb{J}$, and variable $v_{j,k} \in \{0,1\}$ that indicates the end of the subpath allocated to the battery $j \in \mathbb{J}$. Finally, the decision variable vector to be optimized comprises binary variables $x_{1..J,1..K}, u_{1..J,1..K+1}$ and $v_{1..J,1..K+1}$.

Battery capacity is defined as the duration available for the flight. Therefore, it is necessary to assume that the UAV is flying at approximately constant speed during different phases of the mission flight.

Assumption 1. ***Smooth trajectory.*** *Planned trajectory $p_d : [0,\infty) \to \mathbb{R}^3$ is assumed to be a sufficiently smooth time-varying trajectory with its time-derivatives bounded.*

Simplifications. In order to estimate energy consumption during the UAV flight, the following simplifications are made regarding the UAV velocity during the mapping mission:

- UAV flies at a constant velocity v_c when visiting segments $k \in \{1,\ldots,K\}$ to perform the mapping task.
- Maximum velocity v_{max} is considered when UAV is flying from and to the base station for the appointed segments $u_k, k \in \{1,\ldots,K+1\}$ and $v_k, k \in \{1,\ldots,K+1\}$.

Assumption 1 implies that the trajectory is *a priori* planned such that the UAV is able to maintain

a constant flying velocity during the entire flight. Thus, from the energy model equation (2.30), it can be derived that the power consumption depends on the UAV airspeed, and can be assumed constant when velocity is kept constant for sufficiently small angles of attack.

Implications stated above allow the conclusion that available battery capacity can be stated as remaining time for the flight. Similarly, energy consumed during a flight is equivalent to the time spent in the air and, therefore, results from the constant velocity and flight distance.

3.3.2.2 Integer linear programming (ILP) formulation

The battery management problem is to be formulated as an integer linear programming (ILP) optimization problem with binary decision variables. The mathematical formulation is given by a set of equations:

$$\min_{x,u,v} \sum_{j \in \mathbb{J}} \quad \left[\sum_{k=1}^{K} x_{j,k} l_k + \sum_{k=1}^{K+1} (u_{j,k} l_{b,k} + v_{j,k} l_{b,k}) \right] \tag{3.1}$$

subject to:

$$\sum_{k=1}^{K} x_{j,k} l_k \frac{1}{v_c} + \sum_{k=1}^{K+1} (u_{j,k} l_{b,k} + v_{j,k} l_{b,k}) \frac{1}{v_{max}} \leq t_j, \quad \forall j \in \mathbb{J} \tag{3.2}$$

$$\sum_{j=1}^{J} x_{j,k} = 1, \quad \forall k \in \{1, \ldots, K\} \tag{3.3}$$

$$\sum_{k=1}^{K+1} u_{j,k} \leq 1, \quad \forall j \in \mathbb{J} \tag{3.4}$$

$$\sum_{k=1}^{K+1} v_{j,k} \leq 1, \quad \forall j \in \mathbb{J} \tag{3.5}$$

$$\sum_{k=1}^{K+1} u_{j,k} = \sum_{k=1}^{K+1} v_{j,k}, \quad \forall j \in \mathbb{J} \tag{3.6}$$

$$\sum_{k=1}^{K+1} k v_{j,k} - \sum_{k=1}^{K+1} k u_{j,k} \geq \sum_{k=1}^{K+1} u_{j,k}, \quad \forall j \in \mathbb{J} \tag{3.7}$$

$$x_{j,k} + \sum_{l=k+1}^{K+1} u_{j,l} \leq 1, \quad \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K\} \tag{3.8}$$

$$x_{j,k} + \sum_{l=1}^{k} v_{j,l} \leq 1, \quad \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K\} \tag{3.9}$$

$$\sum_{k=1}^{K} x_{j,k} = \sum_{k=1}^{K+1} k v_{j,k} - \sum_{k=1}^{K+1} k u_{j,k}, \quad \forall j \in \mathbb{J} \tag{3.10}$$

$$x_{j,k} \leq u_{j,k} + x_{j,k-1}, \quad \forall j \in \mathbb{J}, \forall k \in \{2, \ldots, K\} \tag{3.11}$$

The objective of the proposed approach is to fragment the area of ordered waypoints into subpaths for each battery, ensuring that the required distances to the base are minimized and the battery capacity is sufficient for completing the assigned subpath flight (including departure and return to the base). It is assumed that the combined capacity of all available batteries is sufficient to cover the entire mapping area. This condition is to be verified beforehand.

The optimization problem has the objective of minimizing the cost function in (3.1), which represents the overall distance of the mission flight. It is stated as a sum of all segment distances, as well as the distances from and to the base station for all the assigned batteries. For every battery $j \in \mathbb{J}$, it includes distances of its respective allocated segments, if decision variable $x_{j,k} = 1$, distance from the base to the first waypoint of the allocated subpath when $u_{j,k} = 1$, and distance from the last waypoint of the allocated subpath back to the base location when $v_{j,k} = 1$.

The optimization problem is subject to constraints given in equations (3.2)-(3.11). Constraint (3.2) ensures that the total flight time planned for each battery does not exceed its initial capacity in terms of remaining flight time.

Constraint (3.3) states that each segment k should be allocated to exactly one battery, without repeating or neglecting a segment.

Constraints (3.4)-(3.7) guarantee that no more than one subpath is allocated to each battery. Constraints (3.4) and (3.5) limit each battery to assigning a maximum of one subpath beginning and ending waypoint, respectively, while (3.6) ensures the number of subpath beginnings and endings is equal for each battery. On the other hand, constraint (3.7) ensures that the ending waypoint of the subpath can only come after the starting waypoint. Thus, if a certain battery has an allocated subpath, this subpath necessarily has to be assigned its beginning and ending waypoint.

Constraint (3.8) ensures that there are no segments allocated to a battery before allocation of the first waypoint ($u_{j,k}$), and, similarly, constraint (3.9) allows no segment allocation after the last waypoint ($v_{j,k}$). Finally, constraints (3.10) and (3.11) ensure that the predefined order of the segments is

respected when allocated in a subpath, and no gaps between two allocated pre-ordered segments are allowed. That is, all the segments between the first and the last waypoint are allocated in a predetermined sequential order.

In order to determine the best base location for the respective mapping field, the above-defined optimization problem needs to be run iteratively for all of the predetermined base location coordinates, considering B possibilities. The best location is chosen as the one that results in the minimum cost (Eq. (3.1)). Finally, subpaths are allocated to the eligible batteries in reference to the resulting decision variables $x_{j,k}, u_{j,k}, v_{j,k}$. The iterative algorithm of the proposed formulation is given in Algorithm 1.

Algorithm 1 Battery management optimization

Initialize the ordered list of waypoints $k \in \{1, \ldots, K+1\}$
Initialize the list of coordinates for possible base locations $b \in \{1, \ldots, B\}$, with B number of locations
Initialize the list of eligible battery capacities $j \in \{1, \ldots, J\}$
Compute the segment length $l_k,\ \forall k \in \{1, \ldots, K+1\}$
Create an ordered list of segments $k \in \{1, \ldots, K\}$
for $b \leftarrow 1$ to B **do**
    Compute the *base-to-segment* lengths $l_{b,k},\ \forall k \in \{1, \ldots, K+1\}$
    Minimize the cost function (3.1) under constraints (3.2)-(3.11)
    return optimal cost
    if $b \neq 1$, cost$(b) \leq$ cost$(b-1)$ **then**
        Set $b\_optimal = b$
    else
        Keep $b\_optimal$
    end if
end for
return $b\_optimal$
return $x_{j,k},\ \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K\}$
return $u_{j,k}, v_{j,k},\ \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K+1\}$

The computational complexity of the given ILP optimization problem depends, namely, on the size of the mapping area, i.e., on the number of waypoints, as well as the number of available batteries with respect to their energy capacities.

3.4 Simulation results and discussion

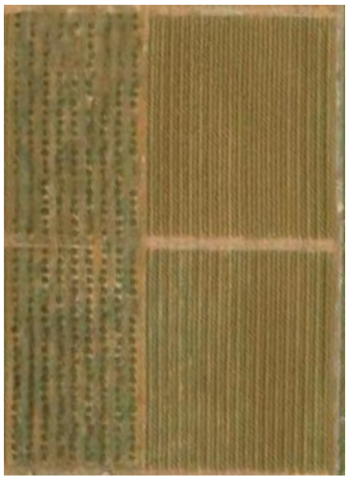
The battery management optimization strategy formulated in the previous section is to be initially validated through simulation results. Its performance can be evaluated by comparing it to an existing state-of-the-art approach. The objective is to ensure a safe and efficient energy-aware mission, i.e., to exhibit increased efficiency of the mission in terms of time, as well as reduced battery consumption

and level of flight risk.

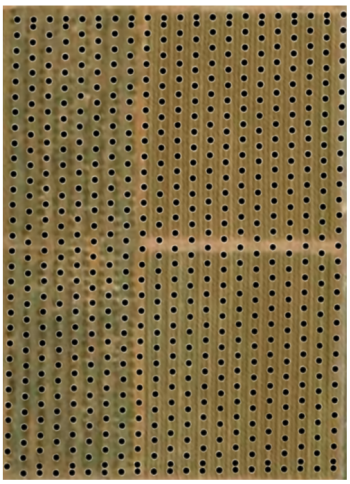
The small-scale example case results of the proposed battery management approach representing different configurations of the simplified area in Figure 3.4 are given in Appendix A. They illustrate the impact of the choice of path configuration on the results of the battery replacement optimization. From the small-scale example, it is seen that, under the assumption that the mapping velocity is kept constant, even at the turning maneuvers, the total energy consumption is affected mostly by the total distance between the segments, as well as the positioning of the base location.

However, a more realistic case example will be presented and analyzed in detail in this section. Therefore, the proposed approach is tested on an agriculture area example that is suitable for mapping with a UAV due to its large size. The test case represents an area separated into an olive orchard and a vineyard, which are to be mapped as a single agricultural field. A Google Maps image of the example test area and its waypoint distribution for mapping are shown in Figure 3.9.

The simulation setup parameters are given in Table 3.3.



(a) Google maps view of the example field



(b) Waypoint distribution for the example field

Figure 3.9: Google Maps view of the example mapping area with an olive orchard and a vineyard.

As a first step, the test case area is decomposed into cells with waypoints representing their centers, as shown in Figure 3.9b. In order to achieve a sufficient level of image quality and detail, the waypoints are *a priori* spaced $8m$ apart, with exceptions near the edge due to the inconsistency in dimensions after decomposition. Thus, the distance between each pair of waypoints is less or equal to the defined value (here: $l_k \leq 8m$). Because the considered mapping terrain is flat, the flying altitude remains constant during the mission, and is set to $20m$ for visiting all the waypoints. The mission is

[1] NB: Defined spacing is attempted to be honored, with certain exceptions, namely near the edges, in order for the mapping coverage consistency to be maintained.

Table 3.3: Simulation setup for the example mapping field

Mapping area decomposition	
Area dimensions	$(150x250)m$
Waypoint spacing [1]	$8m$
Number of waypoints	669
UAV mapping specifications	
Mapping altitude	$20m$
Constant mapping velocity v_c	$4m/s$
Maximum velocity v_{max}	$12m/s$
Eligible battery capacities t_j	$[14, 11, 13, 9]min$

considered to start after the initial take-off and end before the landing. Take-off and landing are to be done manually by the operator and, therefore, are not accounted for during the mission time. Thus, the altitude of the base location options is set to $1m$, which marks the start and end of the flight.

Finally, the test case area consists of 669 waypoints that need to be visited. During the mapping mission, the UAV flies at a defined constant velocity during the mapping mission (here: $v_c = 4m/s$), such that a snapshot is taken when it reaches the position of each waypoint. As illustrated in Figure 3.10, the choice for positioning the recharging base is to be made among 8 location options, one option at each vertex of the test case area, and one per each edge of the area. As the given terrain is flat and accessible, all 8 options are viable for consideration. Following this mapping definition, the distances between all the waypoints and all the possible base locations are calculated and initialized before solving the optimization problem.

Based on the experience of mapping the same test area without a predetermined mission plan for battery allocation, 4 batteries were to be employed, with their capacities varying but not exceeding 15 minutes. Four eligible batteries with different capacities will, therefore, be considered in simulations to validate the proposed battery management optimization strategy. The considered capacities of 14, 11, 13, and 9 minutes are assumed to be sufficient to perform the mapping of the entire test case area.

3.4.1 Proposed optimization strategy simulation results

Simulation results are obtained after solving the optimization problem using Python-MIP toolbox [102]. The solver executes the Branch-and-Cut algorithm for solving mixed integer problems, and providing the optimal solution in *finite time*.

Following the Algorithm 1, the optimal solution to battery allocation results from iteratively running

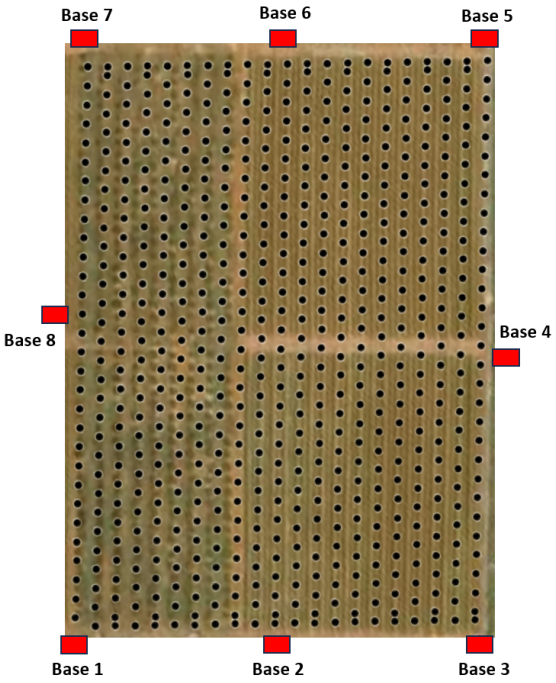


Figure 3.10: Example test case of the mapping area with 8 location options for the position of the recharging base.

the optimization for all the predetermined location possibilities. The final mission plan is ultimately determined for the base station location with the minimum cost.

The optimization problem for this test case has 8036 binary variables and 8717 constraints. The computational time for each base location simulation is given in Table 3.4. All the computations were done on a processor *Intel(R) Core(TM) i7-10610U CPU @ 1.80GHz 2.30 GHz*. As the mission planning is done offline as the preparatory step, the resulting computational times summing up to $27.40\ min$ are acceptable for the given application of the agricultural mapping.

Table 3.4: Optimization results for 8 potential base locations

Base location	Optimization cost	Computational time
1	$5546.75\ m$	$160.54\ sec$
2	**5487.72** m	$273.57\ sec$
3	$5648.55\ m$	$310.76\ sec$
4	$5571.31\ m$	$176.50\ sec$
5	$5534.42\ m$	$68.90\ sec$
6	$5491.41\ m$	$155.20\ sec$
7	$5651.78\ m$	$236.97\ sec$
8	$5579.66\ m$	$261.43\ sec$
Total computational time		$1643.87\ sec\ (27.40\ min)$

The optimal solution of the example problem shown in Figure 3.11 results in using two batteries instead of the anticipated four batteries for complete coverage of the field. The optimization algorithm reduces the flight distance required for battery replacements by selecting waypoints closer to the base as the start and end points of each subpath. By prioritizing shorter segments from/to the base station, energy consumption for support tasks, and not mapping itself, is minimized. Thus, spared energy is reallocated to cover the mapping segments with fewer batteries.

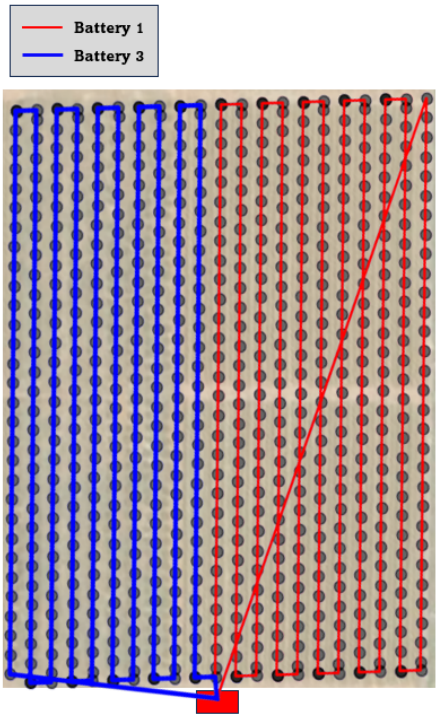


Figure 3.11: Optimization solution for the example test case area coverage.

The given solution considers option 2 for the base location, as it provides the best optimal solution among the resulting costs for each base location option. If planned adequately, two batteries are sufficient for this mapping mission, as demonstrated in Table 3.5.

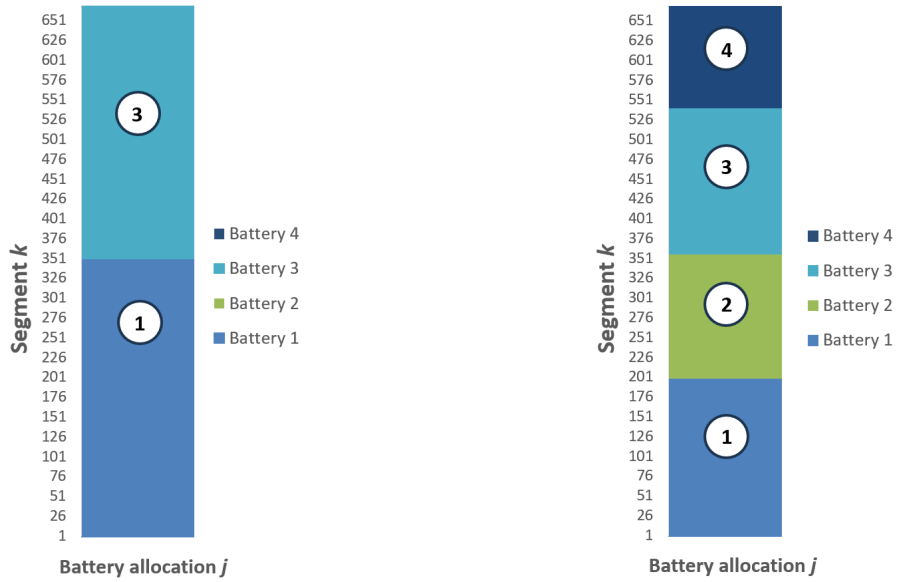
Table 3.5: Simulation results of battery capacities used for the example mapping mission

Battery j	Initial battery capacity	Battery employment duration	Remaining battery capacity
1	$14\ min$	$11.52\ min$	$2.48\ min$
2	$11\ min$	$0\ min$	$11\ min$
3	$13\ min$	$10.28\ min$	$2.72\ min$
4	$9\ min$	$0\ min$	$9\ min$

3.4.2 Comparison of the optimization approach and proportional strategy

The performance of the developed battery management optimization for energy-aware mission planning is assessed against the algorithm in [77] that addresses efficient coverage by dividing the mapping area proportionally depending on the relative capabilities of the UAVs. Among others, these capabilities include flying endurance and range, which can be assumed equivalent to the capacities of eligible batteries in the analyzed scenario. Hereafter, the described approach will be referred to as the proportional strategy.

To ensure an energy-aware efficient mapping mission, the main points to be considered in the following comparison of the two approaches, proposed optimization and proportional approach [77], are to evaluate the energy consumed during the flight in terms of batteries needed to complete the mission and the percentage of batteries used, as well as the total duration of the mission.



(a) Proposed strategy

(b) Proportional strategy

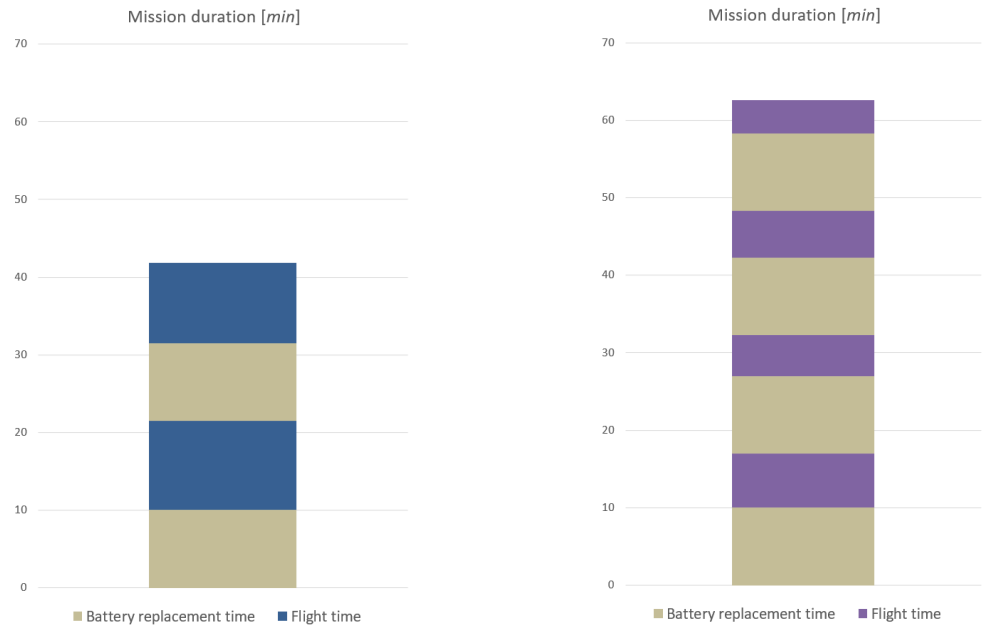
Figure 3.12: Comparison of the proposed optimization and proportional strategy for the test case battery allocation.

Figure 3.12 and associated Table 3.6 show the battery allocation results of the proposed optimization approach in comparison to the state-of-the-art proportional strategy. For reference, in the proportional strategy, all the batteries destined for the mission are employed for mapping. Waypoints are distributed among the eligible batteries proportionally to their available capacities. In this case, all batteries consume around 50% of their initial capacities, which leaves a sufficient security buffer for both longer and shorter trips from/to the recharging base. On the other hand, by optimizing the total

Table 3.6: Simulation results comparison of battery usage in the optimization and proportional strategy

Battery j	Initial battery capacity	Percentage of the battery used	
		PROPOSED STRATEGY	**PROPORTIONAL STRATEGY** [77]
1	$14\ min$	82.3 %	49.6 %
2	$11\ min$	0 %	48.3 %
3	$13\ min$	79.1 %	46.5 %
4	$9\ min$	0 %	47.2 %

distance and duration of the mission and prioritizing lower overall energy consumption, a more accurate path plan can be made beforehand, where fewer batteries are utilized with the optimal waypoint allocation according to the available capacities. To maintain consistency in comparison, the selected base location is the same in both approaches.



(a) Optimization strategy

(b) Proportional strategy

Figure 3.13: Comparison of the total mission duration for the optimization and proportional strategy.

Table 3.7 highlights the efficiency of the proposed optimization strategy in terms of flight duration and battery savings in comparison to the proportionally allocated waypoints. In both cases, the total mission duration includes an additional 10 minutes per used battery for battery placement and replacement, as well as the supporting tasks needed for starting the flight with a new battery. The visual representation of a mission planning comparison using optimization and proportional strategy are shown in Figure 3.13. In the test case simulations, the total flight distance is reduced by 9%, while the

Table 3.7: Simulation results comparison of the mission efficiency indicators in the optimization and proportional strategy

	OPTIMIZATION STRATEGY	**PROPORTIONAL STRATEGY**	% Savings
Number of batteries used	2	4	50%
Total flight distance	$5487.72\ m$	$6029.36\ m$	9%
Total mission duration	$41\ min\ 48\ sec$	$62\ min\ 34\ sec$	33.2%

total mission duration is cut by 33.2%. The overall flight duration affects battery consumption, as well as the efficiency and quality of the mapping mission. Cutting the number of battery replacements in half can greatly enhance mission duration efficiency, as the replacement process takes considerable time. Moreover, by reducing the number of batteries used, the mission risk level is also reduced, as take-offs and landings present the highest risk of a UAV crash.

The primary advantage of the proposed strategy lies in reducing the total number of batteries used. Hence, unnecessary replacements that decrease overall mission efficiency in terms of time are avoided. Additionally, minimizing the number of charging cycles is crucial to maintaining battery health and efficiency in the long term.

3.5 Conclusion

Optimal flight path planning is crucial in UAV agricultural mapping to enable comprehensive coverage and high-quality data collection. A systematic approach to mission planning requires ensuring mission safety and cost efficiency. Apart from the mission duration, an optimal mission should consider the energy needed to complete the area coverage task. Herein presented energy-aware mission planning considers an optimization approach to battery management, i.e. waypoint allocation to determine optimal battery replacement schedule, such that the mapping task is completed with minimal energy consumption.

Mapping mission planning should result in a time-efficient, energy-efficient, and scalable path plan, namely for large mapping areas. The proposed optimization problem aims to minimize the overall flight distance, taking into account the necessary battery replacements due to the limited capacities of available batteries. It also considers selecting the best location for the base station, with respect to the distance from the mapping area. When compared to the state-of-the-art approach that divides the mapping area proportionally according to available battery capacities, the optimization strategy results in fewer battery replacements and shorter total flight distances. That way, both mission dura-

tion and energy consumption are reduced. Consequently, mission risk is lowered, as there are fewer high-risk take-offs and landings. In the long term, aside from the advantages listed above, the proposed approach also preserves battery health and maximizes its performance by reducing redundant charging cycles.

The next chapter will introduce the proposed control strategy that ensures the tracking of the trajectory that was planned beforehand, following the optimization strategy presented here, while maintaining the constant velocity.

Chapter 4

Control strategy for UAV trajectory tracking

4.1 Introduction

The optimal path for the UAV is determined to minimize the mission duration and energy consumption. Once the initial path plan is set, a UAV trajectory is generated such that it tracks the waypoints with a certain level of accuracy while respecting particular constraints, e.g. actuator limitations. Thus, an adequate control technique needs to be developed for trajectory tracking.

When selecting a control strategy for trajectory tracking, the main requirements concern accuracy in terms of position error, as well as mission-specific demands, such as system limitations, velocity tracking, etc. Nonetheless, the computational complexity level needs to be evaluated in order for the control strategy to be applied to a real system.

While thorough research has been conducted on linear control of the quadrotors, where its dynamics is approximated with a linear model, its high level of nonlinearities in the system dynamics makes it difficult to achieve the needed level of precision in a given environment. The quadrotor is a nonlinear system due to the coupling of the aerodynamic forces of the four rotors. In addition to that, because they are light and small in size, this class of aircraft is especially prone to nonlinearities like atmospheric turbulence and reacts sensitively to system degradation, as it causes uncertainties in the system model. Lastly, it is an underactuated system as it has 6 degrees of freedom that are controlled by only four actuators (rotors). Thus, nonlinear control strategies allow for counteracting

these challenges successfully under the condition of knowing the model parameters [103].

Model predictive control (MPC) has shown promising results in trajectory tracking due to its ability to handle constraints, which makes it an easily adaptable solution for various problems. A literature comparison of the MPC-based controller and other control strategies leads to the conclusion that MPC is superior in terms of accuracy and response time. The comparison of tracking performances of the MPC and LQR controllers in [64] shows better accuracy and airworthiness, which translates into flight safety reliability. In [104], authors show that the nonlinear MPC largely outperforms differential-flatness-based control by analyzing accuracy in detail. It also shows that MPC is able to perform dynamically *infeasible* trajectories using future predictions, making it more suitable for tracking time-optimal trajectories that violate the rotor thrust constraints.

As stated above, due to the highly nonlinear system, as well as often nonlinear trajectories, the nonlinear approaches to MPC have drawn attention with their ability to reduce the computational complexity and ensure application on real systems with better tracking performance. Theoretical and application developments in NMPC and its challenges are presented in [105]. For example, authors in [67] prove better disturbance rejection capability in addition to the tracking performance of nonlinear MPC in comparison to linear MPC for different types of maneuvers. In order to improve computational time, they employ a real-time iteration scheme based on Gauss-Newton to approximate the optimization problem and iteratively improve the solution during the runtime of the process. Finally, their approach to nonlinear MPC demands less computational time than linear MPC. Furthermore, an NMPC in [106] provides real-time solutions with a computational time lower than the sampling time in real experiments. The authors propose an approach that uses direct multiple shooting and nonlinear programming formulation to include the internal states and the control actions as optimization variables guaranteeing computational efficiency. Similarly, the direct multiple shooting approach has been successfully adapted for real-time optimization of NMPC in [107].

Different challenges regarding NMPC, namely receding horizon control theory, modeling for NMPC, computational aspects of online optimization, and application issues, are discussed in [108].

4.1.1 Control scheme

Complete control for trajectory tracking of the UAV considered in this thesis is illustrated with the cascade control scheme in Figure 4.1, as in [101].

The cascade control structure considered in this chapter includes low-level inner-loop controller for attitude control, and outer-loop nonlinear MPC controller for trajectory tracking. The estimated states

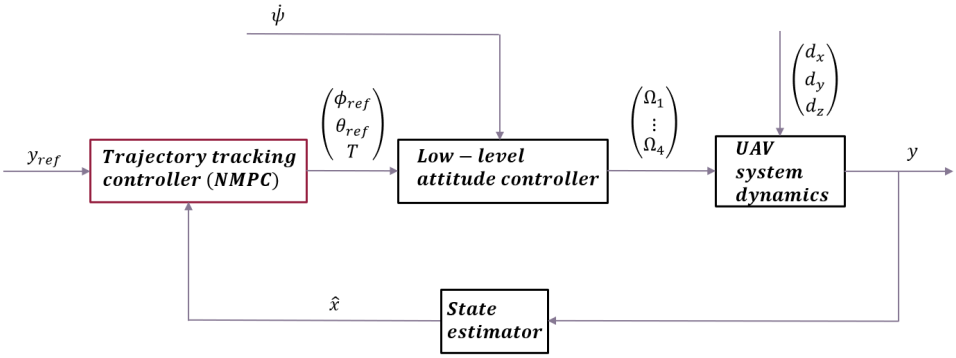


Figure 4.1: Considered cascade control scheme.

of the system $\hat{x}$ are considered initialization states for the trajectory tracking control of the output y. MPC needs to be designed such that the UAV system is tracking the reference output y_{ref} with high level of precision, while rejecting external disturbances d and deals with parameter uncertainties in the system model. Inner-loop controller handles the attitude control, and more precisely, tracks the reference roll and pitch angles ϕ_{ref}, θ_{ref}, as well as the thrust T by adjusting rotor speed Ω_i for i-th of the 4 rotors. A system model of a quadrotor supposes that the UAV heading is aligned with the x-axis of the inertial frame, and therefore, the yaw angle is set to zero ($\psi = 0$). The first-order inner-loop approximation is assumed to provide sufficient information to the MPC to consider the low-level controller behavior. External disturbances added to the UAV system dynamics are represented with the vector $d = [d_x, d_y, d_z]^\top$.

The synthesis of the low-level controller, as well as the state estimation, are assumed to be known and given, and will not be considered for detailed analysis in this thesis. The research work will focus on developing the NMPC trajectory tracking controller.

4.1.2 State-space model

The full model of quadrotor UAV dynamics is presented in Chapter 2. The discretized model is derived from the simplified continuous model (2.32)-(2.35). The discretization is performed by the Runge-Kutta method of 4-th order (RK4), which ensures sufficient accuracy.

- **The Fourth Order-Runge Kutta Method (RK4)**

In RK4, the solution to a first-order differential equation given by $y' = f_c(x, y)$, with initial condition $y(x_0) = y_0$ is to be approximated. The following slope approximations are considered to estimate the slope at time t_0 (while assuming that only an approximation to $y(t_0)$ is known and denoted as $y^*(t_0)$:

$$\begin{cases} k_1 = f_c(y^*(t_0), t_0) \\ k_2 = f_c(y^*(t_0) + k_1 \frac{h}{2}, t_0 + \frac{h}{2}) \\ k_3 = f_c(y^*(t_0) + k_2 \frac{h}{2}, t_0 + \frac{h}{2}) \\ k_4 = f_c(y^*(t_0) + k_3 h, t_0 + h) \end{cases} \tag{4.1}$$

with sampling time h. The slope estimates can be described as follows:

- k_1 is the slope at the beginning of the time step h.
- k_2 is the slope estimate at the midpoint, halfway through the time step h, using the estimate k_1.
- k_3 is another slope estimate at the midpoint, halfway through the time step h, using the estimate k_2.
- k_4 is an estimate of the slope at the endpoint at $t_0 + h$.

The final estimate of $y^*(t_0 + h)$ is obtained as a weighted sum of these slopes:

$$y^*(t_0 + h) = y^*(t_0) + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k4) \tag{4.2}$$

The resulting discrete nonlinear function with the sampling time $h = T_e$ to be used in further study is finally:

$$x_{k+1} = f(x_k, u_k, d_k), \quad k \geq 0, \tag{4.3}$$

where k represents the time step. The control inputs u_k, UAV states x_k and disturbances d_k are considered constant between the two time steps.

The above-defined discrete dynamic model derived from the continuous model (Eq. (2.32)-(2.35)) will be used in this chapter for developing an NMPC for trajectory tracking. The system dynamics involves:

- 8 state variables: $\mathbf{x} = [p, v, \phi, \theta]^\top$,
- 6 output variables: $\mathbf{y} = [p, v]^\top$,
- 3 control input variables: $u = [T, \phi_{ref}, \theta_{ref}]^\top$,

with p and v representing position and velocity vector, respectively, defined in a given coordinate system.

This is a mass-free model, as the input variable thrust T is considered as the respective acceleration rather than the corresponding force. Therefore, the developed NMPC controller needs to be robust to mass changes and uncertainties, as well as the loss of thruster efficiency due to the decline of battery voltage over time. Thruster efficiency is modeled as a parameter uncertainty α in equation (2.33).

4.2 Nonlinear model predictive control for trajectory tracking

4.2.1 Model predictive control

Model predictive control (MPC) is a control strategy that repeatedly solves a finite horizon optimization control problem subject to state and input constraints by considering the model of the system that is to be controlled. An open-loop optimization problem is formulated with receding horizon implementation.

MPC uses the system model to predict future behavior, given estimates or measurements of the current state and intended future control inputs that are chosen to optimize the predicted cost.

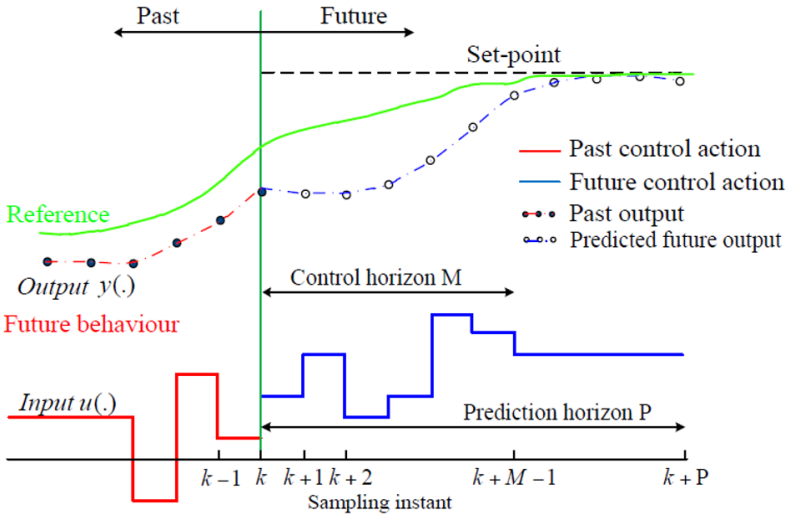


Figure 4.2: Model predictive control methodology [3].

As illustrated in Figure 4.2, given a reference trajectory, the sequence of predicted control inputs and predicted output of a system are computed for the chosen finite control and prediction horizons at each time step or sample time, in a manner that minimizes the tracking error between predicted

output and reference trajectory based on the defined optimization problem. In further discussion, the control horizon is considered to be equal to the prediction horizon.

Once the optimization problem is solved for a specified prediction horizon, only the first control input of the optimal control sequence is implemented, and following the receding-horizon principle, the optimization is repeated at the next time step using newly available information on the system state. This repetition is beneficial for reducing the gap between the predicted and the actual system response, as in a closed-loop operation. Also, it ensures a certain level of inherent robustness to the uncertainty that can arise from imperfect knowledge or mapping of the real system behavior to the system model, also known as multiplicative uncertainty, as well as the additive uncertainty caused by disturbances that can appear in the system dynamics.

As the early-developed MPC control strategies lacked formal guarantees of nominal stability, such as closed-loop stability in the absence of uncertainties, this challenge was tackled by imposing additional conditions on predicted model states, known as equality terminal constraints. The aim of such conditions is to ensure that the desired steady state is reached at the end of a finite prediction horizon. The effect of these constraints can be related to an infinite horizon, thus ensuring various stability and convergence properties [109]. The stability and optimality of different forms of MPC are distilled in [110] to provide a solid basis for further research.

In this thesis, a nonlinear MPC will be considered to address the quadrotor reference trajectory tracking problem.

4.2.2 Mathematical formulation of NMPC

4.2.2.1 Notation

Before introducing the mathematical formulation of the nonlinear MPC for trajectory tracking, the necessary notation is presented beforehand. System variables, as well as the tuning parameters to be selected for the optimal NMPC performance, are given in Table 4.1.

4.2.2.2 NMPC for trajectory tracking

Nonlinear MPC (NMPC) for trajectory tracking aims to compute the future control inputs u_k, that minimize the following cost function J, at time kT_e along the prediction horizon N_p starting from time kT_e and under input constraints:

Table 4.1: Trajectory tracking NMPC system variables and tuning parameters

NMPC system variables	
$\hat{\mathbf{y}}_k$	Predicted system output for positions p and velocities v at time instant k
$\mathbf{y}_k^{ref}$	Reference output states at time instant k
u_k	Control input at time instant k
Δu_k	Variation between two consecutive control inputs at time instant k and $k-1$
$\|\hat{v}_k\|_2$	Norm of the predicted output velocity at time instant k
v_c^{ref}	Reference constant mapping velocity
NMPC tuning parameters	
N_p	Prediction horizon
T_e	Sampling time
Q	Weight matrix for the output cost
R	Weight matrix for the input cost
W	Weight scalar for the velocity cost

$$J(u_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}_{k+n} - \mathbf{y}_{k+n}^{ref}\|_Q^2 + \|\Delta u_{k+n-1}\|_R^2 \right], \quad k \geq 0 \tag{4.4}$$

subject to:

$$u \in \mathcal{U}, \tag{4.5}$$

where the first part of the cost function designates the output cost, which minimizes the tracking error, i.e. the error between the predicted system output $\mathbf{y}_k$ and reference output $\mathbf{y}_k^{ref}$. The second part of the cost function accounts for the input cost, where the control smoothness is to be ensured by minimizing the consecutive control input change Δu_k in a computed sequence. $\mathcal{U}$ is the set containing bounded functions with given lower bound u_{min} and upper bound u_{max}, i.e. $u_{min} \leq u \leq u_{max}$, representing dynamical limitations of the system.

To summarize, two costs to be minimized in the presented NMPC formulation for trajectory tracking are the following:

- Weighted **output cost** (tracking error):

  $\|\hat{\mathbf{y}}_{k+n} - \mathbf{y}_{k+n}^{ref}\|_Q^2$, where $\|\cdot\|_Q^2$ represents the quadratic norm with respect to the positive semi-definite weight matrix Q, i.e. $\|y_k\|_Q = \sqrt{y_k^\top \, Q \, y_k}$.

- Weighted **input cost** (control smoothness):

  $\|\Delta u_{k+n-1}\|_R^2$, where $\|\cdot\|_R^2$ represents the quadratic norm with respect to the positive semi-definite

weight matrix R, i.e. $\|u_k\|_R = \sqrt{u_k^\top \, R \, u_k}$.

4.2.2.3 NMPC with constant velocity

Besides the classical reference trajectory tracking, a control strategy suitable for the specific task of agricultural mapping needs to ensure that the constant velocity of the quadrotor is respected along its trajectory. Constant velocity is essential for enabling uniform and high-quality snapshots of the agricultural area.

The proposed NMPC formulation, therefore, considers velocity cost, which is added to the previously defined classical NMPC for trajectory tracking.

The cost function to be minimized under input constraints is defined as follows:

$$J(u_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}_{k+n} - \mathbf{y}_{k+n}^{ref}\|_Q^2 + \|\Delta u_{k+n-1}\|_R^2 + \|\|\hat{v}_{k+n}\|_2 - v_c^{ref}\|_W^2 \right], \quad k \geq 0, \tag{4.6}$$

$$u \in \mathcal{U}, \tag{4.7}$$

where the last term defines velocity cost, such that the error between the reference constant velocity and the norm of predicted output velocity is minimized.

Finally, the proposed NMPC cost function consists of:

- Weighted **output cost** (tracking error):

  $\|\hat{\mathbf{y}}_{k+n} - \mathbf{y}_{k+n}^{ref}\|_Q^2$, with the weight matrix Q,

- Weighted **input cost** (control smoothness):

  $\|\Delta u_{k+n-1}\|_R^2$, with the weight matrix R,

- Weighted **velocity cost** (constant velocity error):

  $\|\|\hat{v}_{k+n}\|_2 - v_c^{ref}\|_W^2$, with the weight scalar W.

4.2.2.4 Output robustness

The aim of the control law is to track the reference output $\mathbf{y}_{ref}$ with minimal error. Formulated NMPC uses the discretized formulation of the model (2.32)–(2.35) to compute the predicted output of the quadrotor over the prediction window, which is denoted $\hat{\mathbf{y}}$. As the model output $\mathbf{y}^{model}$ considers a

nominal model, without any parameter uncertainties or external disturbances, we need to address the robustness of the controller.

To account for possible additive external disturbances d or multiplicative errors due to parameter uncertainties, the error between the model output, $\mathbf{y}^{model}$, and the output of the real system $\mathbf{y}$ is considered to perform an offset-free trajectory. Along the prediction horizon starting from time kT_e, at time $(n+k)T_e$, the predicted output is computed following the bias equation:

$$\hat{\mathbf{y}}_{k+n} = \mathbf{y}^{model}_{k+n} + n(\mathbf{y}_k - \mathbf{y}^{model}_k), \quad k \geq 0, \quad n \in [1, N_p] \tag{4.8}$$

The model output $\mathbf{y}^{model}$ is given by a nominal model, without parameter uncertainties, with thruster efficiency $\alpha = 1$, nor external disturbances, $d = 0$.

It is assumed that the real system's output is error- and noise-free, and all the state variables are measured. Disturbances acting on the system d are assumed to be bounded, but unknown and unmeasured. Therefore, to obtain an offset-free control with NMPC, model output $\mathbf{y}^{model}$ is augmented with the measured error and multiplied by the prediction horizon time step n, to account for propagation of the respective error towards the end of the prediction horizon N_p.

4.2.2.5 Parameter tuning

MPC is an optimization-based control approach, where its performance depends on the choice of tuning parameters, such as weight matrices in the cost function and prediction horizon. Tuning such parameters can be challenging because they are related to the closed-loop performance of the system in a complex manner.

Effective tuning of the NMPC parameters requires a balance between prediction accuracy, control smoothness, and computational feasibility. Iterative tuning and extensive testing are essential for achieving optimal trajectory tracking performance. By systematically adjusting the prediction horizon, sampling time, cost weights, and input constraints, the resulting tuning parameters are set to meet the specific needs of the application, which is, in this case, accurate mapping of an agricultural field. Outdoor conditions need to be considered to ensure the robustness of the NMPC.

Here is a comprehensive list of tuning parameters for the studied case:

- **Prediction Horizon** (N_p) is the number of future steps in a window over which predictions are made. A longer prediction horizon can improve performance by considering a longer future trajectory but increases computational load.

- **Sampling Time** (T_e) is a time interval between two consecutive control actions. The chosen value should be small enough to capture system dynamics accurately without causing excessive computational burden. It depends on the dynamics of the system and its response time.

- **Cost Function Weights** (state weights Q, control input weights R, constant velocity weight W) define the importance of each individual cost in the cost function. A balance needs to be achieved between penalizing deviations from the desired trajectory and reference velocity, as well as the excessive change in control efforts.

- **Input Constraint Handling** imposes limits on control inputs (thrust T, reference roll ϕ_{ref}, reference pitch θ_{ref}). Tuning ensures that constraints are realistically set to avoid infeasibility while protecting the system from unsafe conditions.

Different guidelines for selecting and tuning MPC parameters exist. The simplest method to achieve the desired performances is through a manual adjustment in trial and error simulations. This method is often sufficient if done *ad hoc* for a given system and its application based on experience and prior knowledge. However, there are more advanced strategies for tuning MPC parameters in an optimal way. For example, authors in [111] present a tuning approach based on Artificial-Neural-Network (ANN). The main advantage of this approach is reaching closed-loop stability and online parameter tuning using Particle Swarm Optimization (PSO) and Online Sequential Extreme Learning Machine(OS-ELM). Similarly, a tuning strategy for NMPC with implemented reinforcement learning is developed in [112], which is implemented for trajectory tracking control of UAVs to facilitate a generic and system-independent tuning process.

In this work, parameter tuning is done manually using the trial and error method.

4.3 UAV mapping mission trajectory tracking with NMPC

The objective of this study is to evaluate the performance of the proposed Nonlinear Model Predictive Control (NMPC) in trajectory tracking for UAVs (quadrotors), specifically in the context of mapping agricultural fields. The increasing demand for precision agriculture has highlighted the need for efficient and accurate mapping techniques. Quadrotors, with their agility and maneuverability, present a viable solution for detailed aerial surveys. NMPC, known for its ability to handle multi-variable control problems and constraints, alongside with the system nonlinearities, is employed to ensure the quadrotor follows a predefined path with high precision. First, the mapping mission planning phase

described in Chapter 3 results in an ordered set of waypoints that need to be visited by a UAV. Once it is determined, trajectory tracking of the planned path needs to be ensured with a suitable control strategy.

In this section, we present the results of the simulation experiments, which were designed to assess the effectiveness of the NMPC algorithm in various scenarios typical for agricultural field mapping. These scenarios include back-and-forth trajectories in different conditions (under external disturbances and model uncertainties). The performance metrics used to evaluate the MPC algorithm include trajectory accuracy, control effort, and computational efficiency.

The following results are derived from simulations using a nonlinear quadrotor model given in equations (2.32)–(2.35), after discretization, both in a nominal case and incorporating realistic environmental conditions such as wind disturbances and model disturbances. Robustness is evaluated in case of external disturbances d and uncertain parameter for thruster efficiency α.

Key aspects covered in this results section include:

- **Trajectory Tracking Accuracy:** Evaluation of the quadrotor's ability to follow the desired path within an allowed error margin.
- **Control Effort:** Analysis of the control inputs required by the NMPC algorithm to maintain the desired trajectory.
- **Computational Performance:** Assessment of the algorithm's computational requirements, ensuring timely and efficient processing for maintaining the desired trajectory.
- **Robustness to Disturbances:** Examination of the tracking performance under uncertainties in the model parameters, and external disturbances, such as wind.

These results demonstrate the potential of NMPC for enhancing the precision and efficiency of quadrotor-based agricultural mapping, contributing to the advancement of precision agriculture technologies.

4.3.1 Simulation results and discussion

The parameter values of the system can be identified through classic system identification techniques. Here, they were chosen empirically as in [113], and are shown in Table 4.2.

Simulations have been conducted for a test case that represents back-and-forth movements of the quadrotor when performing the mapping task. An illustrative example of the UAV path is shown in

Table 4.2: Model parameter values

Model parameter		Value
Gravitation	g	$9.81m/s^2$
Linear damping terms	A_x, A_y, Az	$(0.1, 0.1, 0.2)s^{-1}$
Inner-loop gains	K_ϕ, K_θ	1
Time constants	τ_ϕ, τ_θ	$(0.7, 0.5)s$

Figure 4.3. The reference positions of the visiting waypoints, defined as 3D relative coordinates from a given starting point, are specified in Table 4.3. As this flight plan represents a segment of the full mission plan, take-off and landing are not included, and altitude is held constant at 20 m.

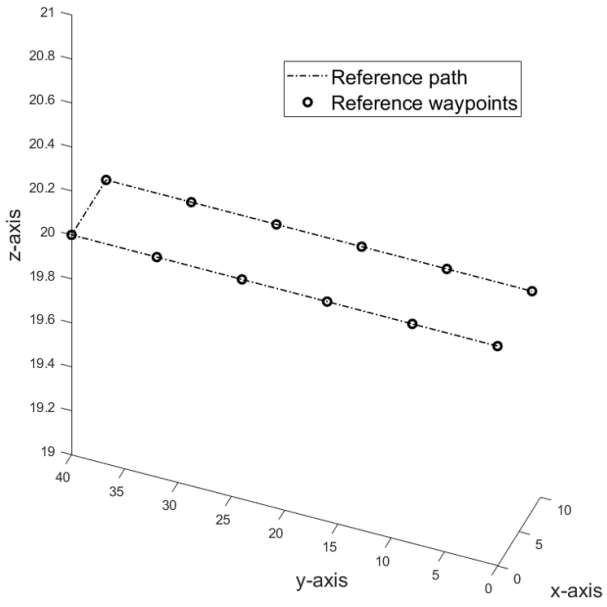


Figure 4.3: Reference path and waypoints for the UAV mapping mission.

Tuning parameters and constraints on control inputs are given in Table 4.4. The chosen values, based on the trial-and-error approach, are compatible with the implementation on a UAV model type DJI, that is suitable for agricultural mapping. Prediction horizon N_p and sampling time T_e are adjusted to the system dynamics and reference trajectory. Weight matrices Q and R are normalized and chosen as a trade-off between tracking accuracy and control input changes within imposed bounds. The choice of the tracking velocity cost W will be discussed in detail. Control inputs are bounded, such that the value of thrust varies approximately $(0.5 - 1.5)\ g$, in order to maintain the reference altitude despite the gravity g, and reference angles ϕ_{ref} and θ_{ref} do not enable aggressive maneuvers, and to remain within $|30|°$ during the mapping flight, as it is the case for the DJI quadrotor model Matrice 350

Table 4.3: Relative positions of the reference waypoints for the UAV trajectory tracking

Waypoint	Relative coordinates
$start(1)$	$(0, 0, 20)\ m$
2	$(0, 8, 20)\ m$
3	$(0, 16, 20)\ m$
4	$(0, 24, 20)\ m$
5	$(0, 32, 20)\ m$
6	$(0, 40, 20)\ m$
7	$(8, 40, 20)\ m$
8	$(8, 32, 20)\ m$
9	$(8, 24, 20)\ m$
10	$(8, 16, 20)\ m$
11	$(8, 8, 20)\ m$
$finish(12)$	$(8, 0, 20)\ m$

RTK [114].

Table 4.4: NMPC tuning parameter values and control input bounds

Tuning parameter		Value
Prediction horizon	N_p	10
Sampling time	T_e	$0.1\ s$
Output cost weight	Q	$diag(1, 1, 5, 0, 0, 0)$
Control input cost weight	R	$diag(1, 10^2, 10^2)$
Control input		**Bounds**
Thrust	T	$5\ m/s^2 \leq T \leq 15\ m/s^2$
Reference roll	ϕ_{ref}	$\phi_{ref} \leq \lvert 30° \rvert$
Reference pitch	θ_{ref}	$\theta_{ref} \leq \lvert 30° \rvert$

All the simulation results are obtained by solving the proposed optimal control problem with the Matlab *fmincon* algorithm. The control problem, at time $t = 0\ s$, is initialized with the initial state vector $\mathbf{x} = [0, 0, 20, 0, 5, 0, 0, 0]^\top$, where position vector $p = [0, 0, 20]^\top m$, velocity vector $v = [0, 5, 0]^\top m/s$, roll angle $\phi = 0°$ and pitch angle $\theta = 0°$. The reference constant velocity is set at $5\ m/s$, and initialized in the mapping direction along y-axis. Optimization, at time $t = 0\ s$, is initialized with the values $u = [9.81 m/s^2, 0°, 0°]^\top$, where control inputs represent thrust T, and reference angles roll ϕ_{ref} and pitch θ_{ref}, respectively. At time $t = kT_e$, the MPC algorithm is initialized with the optimal values of the control inputs computed at time $(k-1)T_e$.

4.3.1.1 Nominal case

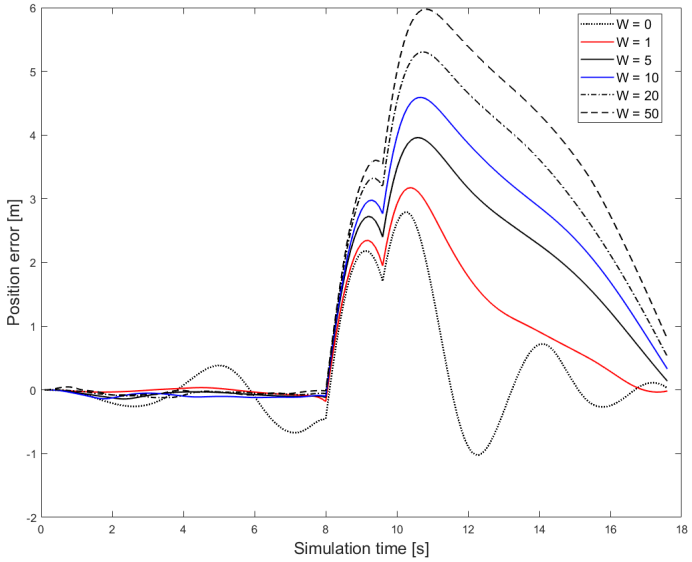
A nominal case implies a matching of the system model and the real system, i.e., the absence of external disturbances and model uncertainties. Thus, in the nonlinear model equations (2.32)-(2.35), $d = [0, 0, 0]^T$ and $\alpha = 1$.

Table 4.5: NMPC solution evaluation

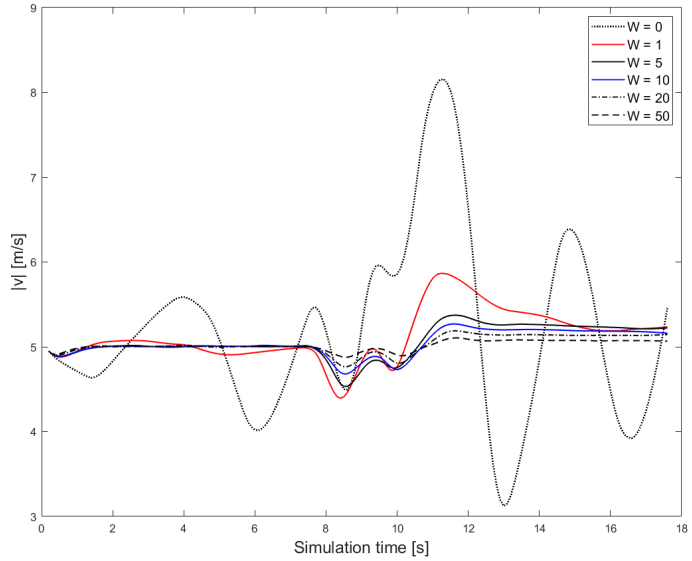
Absolute position tracking error [m]			
Velocity cost weight W	min	max	mean
$W = 0$	6.21e-05	2.79	0.60
$W = 1$	6.21e-05	3.17	0.75
$W = 5$	6.21e-05	3.96	1.27
$W = 10$	6.21e-05	4.59	1.54
$W = 20$	6.21e-05	5.30	1.83
$W = 50$	6.21e-05	5.98	2.12
Absolute velocity tracking error [m/s]			
Velocity cost weight W	min	max	mean
$W = 0$	2.00e-03	3.16	0.78
$W = 1$	3.90e-03	0.87	0.23
$W = 5$	3.21e-04	0.47	0.15
$W = 10$	8.06e-04	0.32	0.12
$W = 20$	3.16e-05	0.24	0.08
$W = 50$	9.81e-05	0.12	0.05
CPU solving time [s]			
Velocity cost weight W	min	max	mean
$W = 0$	0.05	1.59	0.27
$W = 1$	0.05	1.52	0.20
$W = 5$	0.05	1.14	0.22
$W = 10$	0.05	1.25	0.22
$W = 20$	0.06	1.38	0.24
$W = 50$	0.06	1.83	0.28

In addition to classical trajectory tracking, the proposed NMPC includes the velocity tracking cost in order to maintain the constant velocity during the entire mapping mission. Therefore, its performance depends on the choice of the weight for constant velocity tracking W. The test case illustrated in Figure 4.3 and Table 4.3 is tested for different values of W. Good tracking performance requires low tracking error while tracking the reference constant velocity. By increasing the value of the weigh scalar W, the absolute position error increases. In contrast, the absolute velocity error decreases, as the constant velocity cost is given more importance in the optimization. CPU solving time is not directly affected by

W, as no regularity is found in the results.



(a)



(b)

Figure 4.4: Resulting errors for different values for velocity tracking weight W (a) Position tracking error; (b) Norm of velocity.

Minimal variations in constant velocity enable good-quality snapshots during the mapping flight. NMPC without velocity tracking ($W = 0$) and various levels of velocity tracking importance ($W = 1, 5, 10, 20, 50$) are compared in Table 4.5, considering absolute position tracking error, absolute veloc-

ity tracking error and computational time (CPU solving time).

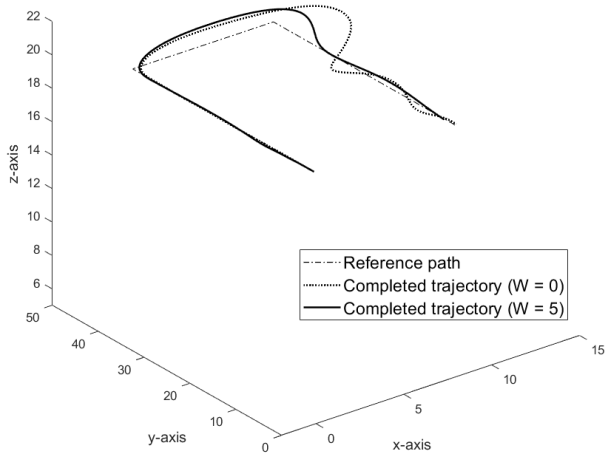


Figure 4.5: Reference path and waypoints for the UAV mapping mission.

Development of the position error and norm of the velocity along the trajectory are given in Figure 4.4a and 4.4b. Smaller variations from the reference velocity imply a more important position tracking error. The effects of the turning maneuvers are visible starting from $t = 8s$, as both position and velocity norm errors increase. The velocity norm is smoothened by applying higher values of the weight W. However, by increasing W, position error also increases. There are no large improvements in the constant velocity when comparing results of $W > 5$. Thus, as mapping mission requires both position accuracy and smooth flight, $W = 5$ represents a good trade-off between position and velocity tracking.

Completed trajectories for both cases are illustrated in Figure 4.5. As the reference path is rectangular, turning points demand a change in velocity. Thus, the resulting trajectory without constant velocity tracking results in deviations in velocity norm, but also more aggressive changes of direction and, consequently, position deviations.

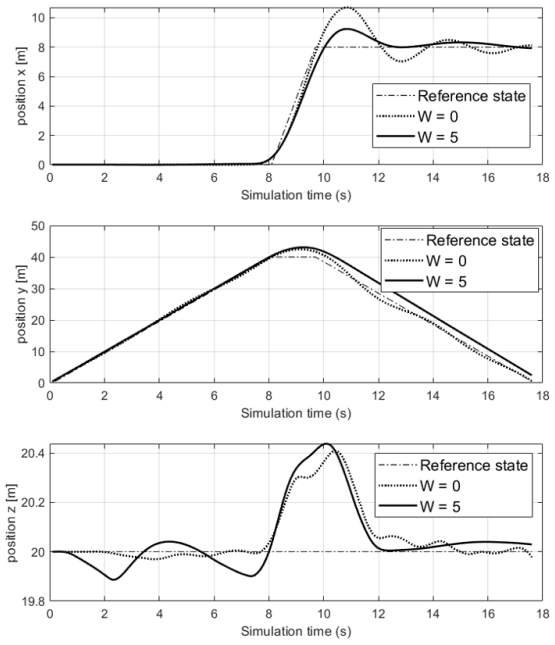


Figure 4.6: Reference and resulting positions in x, y and z-direction for $W = 0$ and $W = 5$.

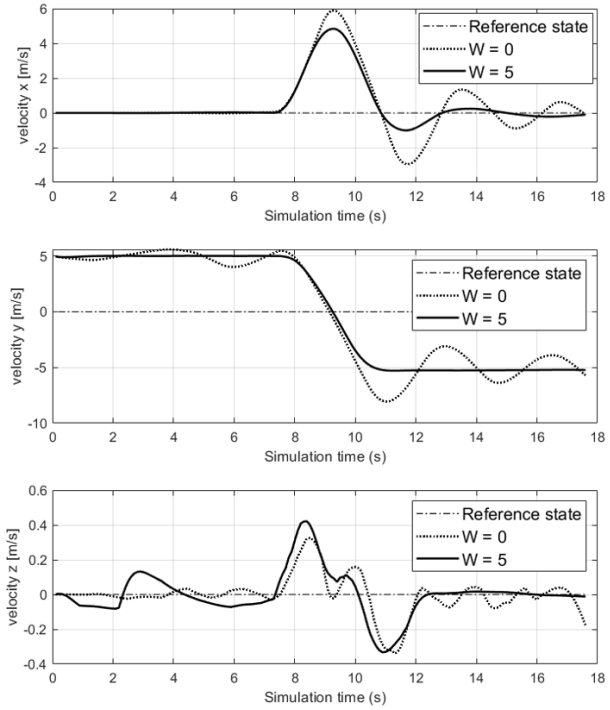


Figure 4.7: Reference and resulting velocity components in x, y and z-direction for $W = 0$ and $W = 5$.

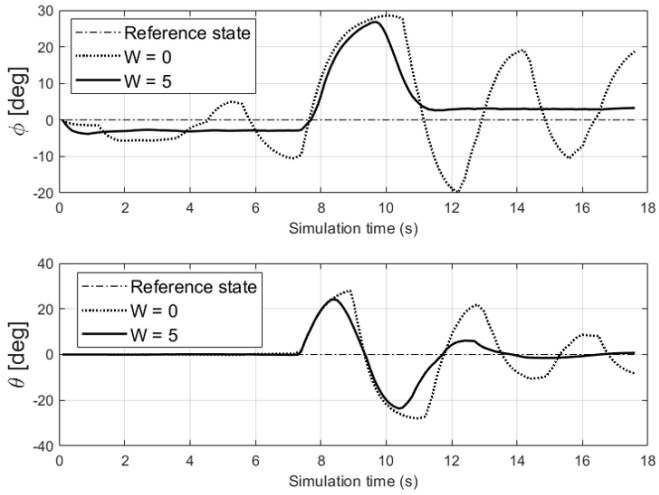


Figure 4.8: Reference and resulting roll ϕ and pitch θ angles for $W = 0$ and $W = 5$.

Figures 4.6, 4.7, and 4.8 exhibit the comparison of resulting states and deviation from the reference for cases with and without the constant velocity control. Even though the norm of position error is larger for $W = 5$, state changes between consecutive time instants are less abrupt, which indicates a more stable flight. A stable flight is imperative for a mapping mission, which also implies small Euler angles. Figure 4.8 shows that by imposing the constant velocity norm, the angle changes of ϕ and θ are less important, and remain largely within the imposed bounds of $30°$. Finally, velocity components in all 3 directions are stabilized faster after necessary maneuvers, which is illustrated in Figure 4.7. A change in velocity behavior is seen at $t = 8s$, when the turning maneuver takes place. Before the turning, velocities in all three directions are rather smooth, with only a slight difference between the case $W = 0$ and $W = 5$. However, as the turning maneuver is being handled, by controlling the constant velocity, all velocity components are also smoothened.

The resulting values for control inputs are shown in Figure 4.9. When comparing the thrust values for the two cases, it can be concluded that, in the case where $W = 0$, the values are saturated while during the turning maneuver and continue to vary to a larger extent than is the case for $W = 5$. When observing the reference angles ϕ_{ref} and θ_{ref}, both control inputs reach the saturation point of $30°$ at the beginning of the turn, at $t = 8s$, for both values of weight W. However, their values remain around 0 for the largest part of the trajectory. Overall changes in control inputs between the consecutive time instants are smoothened in the case of constant velocity tracking, where $W = 5$.

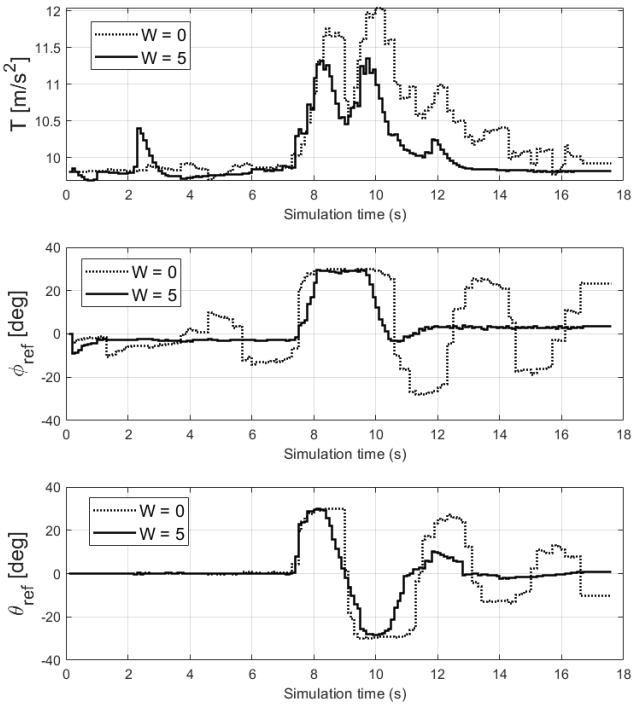


Figure 4.9: Resulting control input values for thrust T, reference roll ϕ_{ref} and pitch θ_{ref} angles, for $W = 0$ and $W = 5$.

4.3.1.2 Robustness assessment

Ensuring robustness in NMPC is crucial for reliable operation in uncertain environments. Robust NMPC enhances safety, performance, and efficiency, making it a vital component for applications requiring high levels of precision and reliability, such as agricultural mapping.

Robustness assessment evaluates the ability of NMPC to handle uncertainties and disturbances. In real-world scenarios, systems are often subject to model inaccuracies and external disturbances. Robustness assessment ensures that the NMPC can maintain satisfactory performance under such conditions.

The robustness of the proposed NMPC is assessed by considering two aspects: model uncertainty and external disturbances. Model uncertainty represents variations between the actual system and the prediction model. In this evaluation, thruster efficiency α is considered to be an uncertain parameter. External disturbances are defined as a vector $d = [d_x, d_y, d_z]^\top$, and represent, for example, the wind in 3 directions acting on a UAV as torque.

Both model uncertainty and disturbances are bounded and handled with a bias in equation (4.8),

as the error between the model output $\mathbf{y}^{model}$ and the real system output $\mathbf{y}$.

The robustness evaluation is performed by analyzing the results of 50 Monte Carlo simulations. Test cases are generated with random values of constant external disturbances $d = [d_x, d_y, d_z]^\top$, and uncertainty of the model parameter of the thruster efficiency α are considered for all the test cases. The external disturbance values vary in the range $d_x = [-3, 3]\, m/s^2$, $d_y = [-3, 3]\, m/s^2$, $d_z = [0, 2]\, m/s^2$, while for the thruster uncertainty $\alpha = [0.8, 1.2]$.

Simulation parameters are the same as in the nominal case and are given in the Table 4.3 and Table 4.4. All the test cases are compared for the NMPC with velocity cost weights $W = 0$ and $W = 5$.

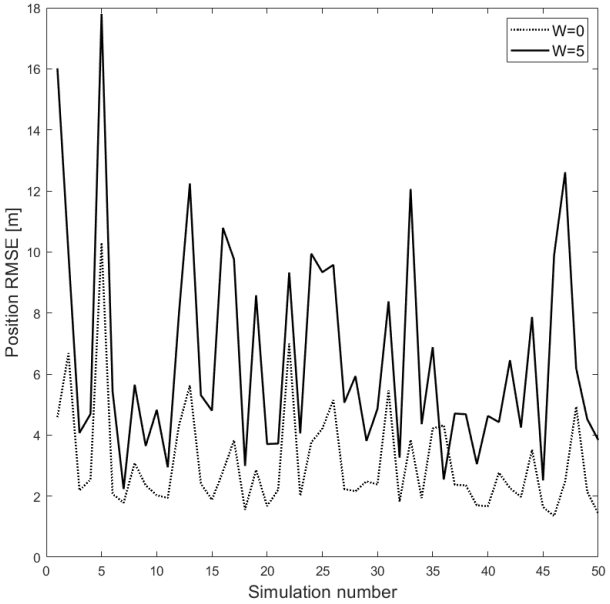


Figure 4.10: Position RMSE for 50 Monte Carlo simulations for $W = 0$ and $W = 5$, random values of constant external disturbances $d = [d_x, d_y, d_z]^\top$, and model parameter uncertainty of the thruster efficiency α.

Figure 4.10 illustrates the position Root Mean Square Error (RMSE) along the trajectory for 50 test cases and both values of the weight parameter W. Similarly, Figure 4.11 shows the values of the average norm of velocity $|v|$ along the same trajectory. Reference constant velocity is set to $v_c^{ref} = 5\, m/s$. Based on the two graphs, position RMSE reaches overall higher values if $W = 5$, whereas the resulting velocity deviates further from the reference constant velocity if $W = 0$, as expected.

Robustness evaluation of the Monte Carlo simulations is given in Table 4.6, where minimum, maximum, and mean values are given for position RMSE, average values of the norm of velocity, and average CPU time for solving the optimization. As the constant velocity remains the priority for acquiring

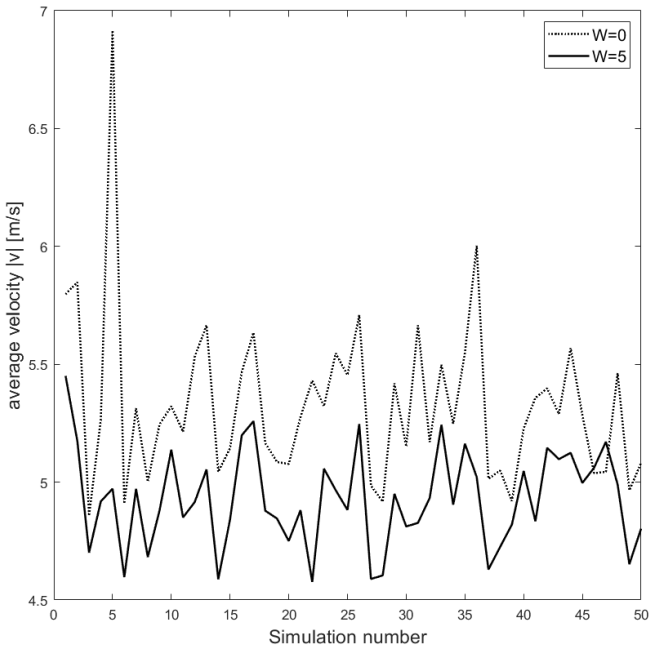


Figure 4.11: Average norm of velocity for 50 Monte Carlo simulations for $W = 0$ and $W = 5$, random values of constant external disturbances $d = [d_x, d_y, d_z]^\top$, and model parameter uncertainty of the thruster efficiency α.

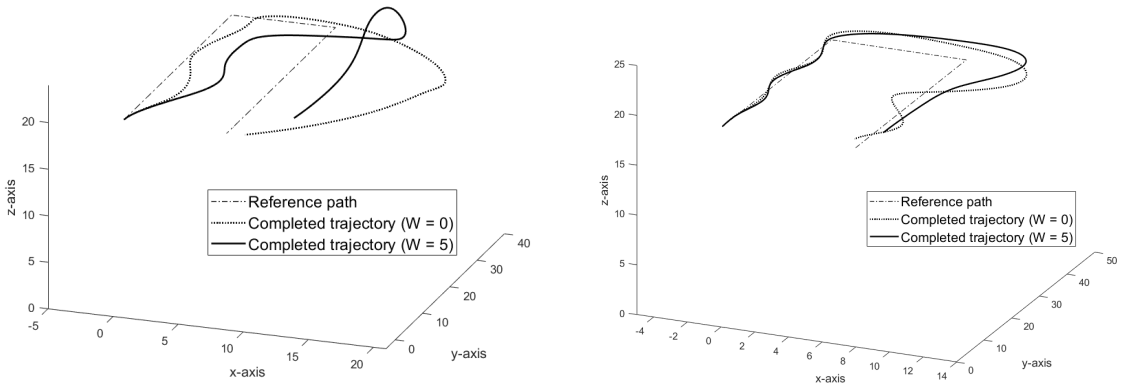
Table 4.6: Monte Carlo simulations robustness evaluation

Position RMSE [m]			
Velocity cost weight W	min	max	mean
$W = 0$	1.35	10.30	3.08
$W = 5$	2.23	17.80	6.52
Average velocity [m/s]			
Velocity cost weight W	min	max	mean
$W = 0$	4.86	6.92	5.33
$W = 5$	4.58	5.45	4.93
CPU solving time [s]			
Velocity cost weight W	min	max	mean
$W = 0$	0.18	0.49	0.34
$W = 5$	0.27	0.58	0.41

good-quality images, the value of the weight $W = 5$ represents a trade-off between small deviations from the reference velocity $v_c^{ref} = 5\, m/s$ and smaller position accuracy, as well as slightly longer average computational CPU time. These results represent the worst and the best case considering added disturbances and model uncertainties. Based on these results, when constant velocity cost is active,

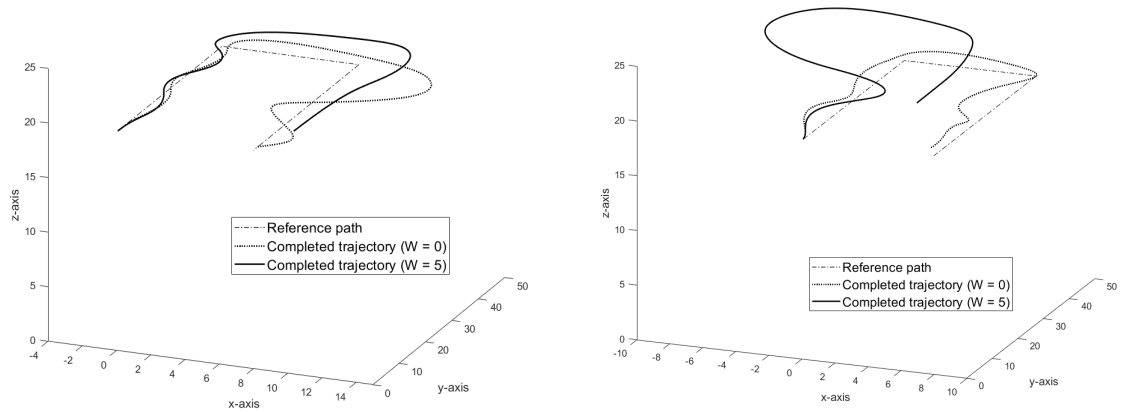
lower average velocity is prioritized rather than reaching the values over reference constant velocity. Consequently, smoother and more stable flight can be ensured.

Depending on the specific mission requirements, the acceptable variations need to be defined. Consequently, we can determine the bounds on the values of disturbances and uncertainties against which the proposed NMPC satisfies the robustness.



(a) (b)

Figure 4.12: Reference and resulting trajectories, for $W = 0$ and $W = 5$. (a) Test case 22; (b) Test case 23.



(a) (b)

Figure 4.13: Reference and resulting trajectories, for $W = 0$ and $W = 5$. (a) Test case 42; (b) Test case 47.

Figures 4.12a, 4.12b, 4.13a and 4.13b illustrate deviations from the reference trajectory for the test

Table 4.7: Robustness assessment for test cases 22, 23, 42 and 47

Test case	External disturbances d	Model uncertainty α	Position RMSE	Average velocity
22	$[3.0, -2.8, 1.9]\ N$	1.1		
$W = 0$			$7.00\ m$	$5.43\ m/s$
$W = 5$			$9.32\ m$	$4.58\ m/s$
23	$[0.9, 0, 1.5]\ N$	1.2		
$W = 0$			$2.01\ m$	$5.32\ m/s$
$W = 5$			$4.05\ m$	$5.05\ m/s$
42	$[0.7, 0.5, 1.8]\ N$	0.9		
$W = 0$			$2.26\ m$	$5.40\ m/s$
$W = 5$			$6.45\ m$	$5.15\ m/s$
47	$[-1.8, 2.0, 1.8]\ N$	0.9		
$W = 0$			$2.48\ m$	$5.04\ m/s$
$W = 5$			$12.61\ m$	$5.17\ m/s$

cases number 22, 23, 42 and 47, respectively. These test cases are chosen to illustrate the effects of different combinations of high and low values of external disturbances and model uncertainty.

Values for the 4 mentioned test cases are listed in Table 4.7. External disturbances d represent constant wind force caused by wind velocity, and are assumed to be bounded as in [115]. Uncertainty on thruster efficiency α affects the resulting trajectory as it varies from 80 %, in case of overestimation, to 120 % for underestimating the real thruster performance.

The first three examples show that despite the higher position RMSE, the resulting trajectory is smoother and more appropriate for mapping when the constant velocity cost is imposed, i.e., $W = 5$, in comparison to $W = 0$. Test case 47 exhibits high tracking error and a resulting trajectory that largely deviates from its reference if constant velocity is tracked. In this case, classical NMPC tracking with $W = 0$ presents better results. Overall results show that the higher values of external disturbances deteriorate tracking performance. Also, when comparing values in cases 23 and 42, with similar values of d, trajectories are somewhat degraded for test case 42, where thruster efficiency is overestimated.

Nonetheless, from Figure 4.10, test case number 5 results in the highest position RMSE, $10.3\ m$ and $17.8\ m$, for $W = 0$ and $W = 5$, respectively. The resulting trajectory illustrated in Figure 4.14 does not follow the planned path within reasonable limits. With the uncertain parameter value of the thruster efficiency $\alpha = 0.8$, it can be concluded that, due to the overestimation of the thruster, the UAV is not capable of reaching the reference state. Therefore, the resulting trajectory has very low, if any, position accuracy. It seems important to estimate this parameter to improve the performance of the control.

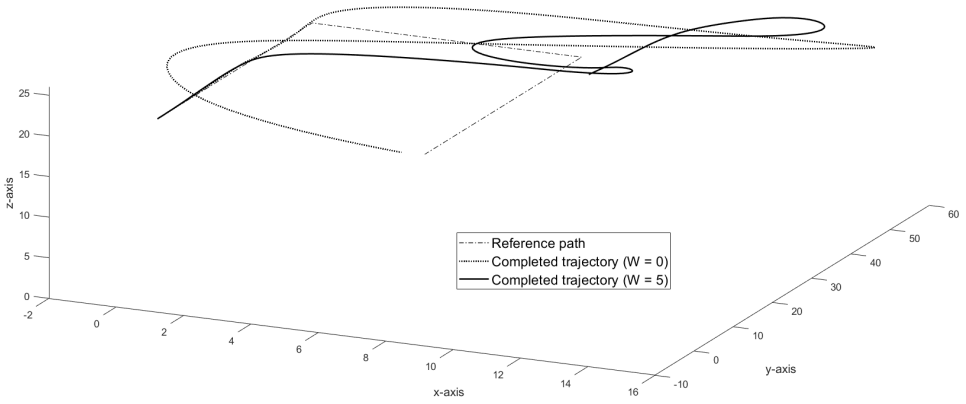


Figure 4.14: Reference and resulting trajectories, for $W = 0$ and $W = 5$. Test case 5.

Based on the provided analysis of the simulation results, it can be concluded that an underperforming thruster poses the greatest problem to the robustness of the proposed NMPC approach. Additionally, high values of external disturbances can further deteriorate the tracking performance. However, average velocity approaches the reference constant values closer if constant velocity is activated in the NMPC cost function.

4.4 Conclusion

The proposed Nonlinear Model Predictive Control (NMPC) for agricultural mapping effectively addresses the challenge of trajectory tracking with a constant velocity. This constant velocity is crucial for acquiring high-quality images, ensuring that the agricultural mapping data is reliable and accurate.

Simulation results demonstrate a comparison between scenarios with and without the constant velocity cost. The findings highlight the importance of the compromise between maintaining a constant velocity for superior image acquisition and mapping accuracy. When implementing constant velocity cost, the resulting trajectory, as well as the control inputs of the quadrotor, are smoothened.

Additionally, the robustness of the NMPC was evaluated against external disturbances, such as wind and model uncertainty of the thruster efficiency parameter, showcasing its ability to maintain performance under varying conditions. However, to ensure the proposed NMPC satisfies robustness, acceptable variations need to be defined based on specific mission requirements. While the results

are promising, they indicate that further improvements can be achieved by incorporating a trajectory planning step. This would help generate smoother reference trajectories, potentially enhancing the overall performance of the NMPC. A robust NMPC can also be developed to further mitigate the deterioration caused by uncertainties and disturbances. Also, the estimation of the uncertainty parameter α and external disturbances can allow for a greater efficiency of the proposed control strategy.

Chapter 5

Towards multi-UAV mapping mission

5.1 Introduction

While single-UAV missions have demonstrated significant benefits in agricultural applications, multi-UAV systems offer enhanced performance and scalability. By deploying multiple drones simultaneously, large areas can be covered more efficiently, reducing the time required for data collection. Multi-UAV systems also provide redundancy, ensuring that if one UAV fails, others can continue the mission without interruption.

In a cooperative multi-UAV mission, all the UAVs aim to satisfy a common objective. The successful implementation of cooperative mapping tasks in multi-UAV systems involves a two-layered approach: strategic and tactical. The strategic layer considers mission and path planning, such that the mapping of an entire field is divided for each UAV in an optimal manner and without collision of multiple planned paths. The tactical layer involves choosing the control strategy suitable for tracking planned paths simultaneously for each UAV in the system by avoiding *ad hoc* collision risk. This bifurcation ensures that the UAVs operate efficiently and effectively, achieving the mission objectives while adapting to specific conditions. For example, if path planning results in reference mapping paths of the UAVs in a cooperative mission that do not intersect, collision avoidance needs to be ensured on a tactical level, i.e. control level. When a UAV needs to stop the mapping mission and return to the base prematurely, the risk of collision may arise.

This chapter provides a foundation for a comprehensive cooperative multi-UAV mission for agricultural mapping by understanding the complexities and benefits of such a problem. It also identifies the main challenges and proposes an opening toward more complex and realistic problem configu-

rations.

Concepts presented in previous chapters will be extended to the multi-UAV system. The proposed path planning for a multi-UAV mission resembles planning for multiple available batteries in Chapter 3. The main focus of this chapter will be placed on developing a suitable control strategy for trajectory tracking that considers coordination between the UAVs, namely by addressing the collision avoidance problem.

5.2 Path planning for multiple UAVs

Path planning is a strategic component of multi-UAV agricultural mapping missions. It involves determining the optimal collision-free routes for UAVs to ensure comprehensive coverage of an agricultural field while minimizing travel time and energy consumption.

Multi-UAV path planning is modeled as a graph in order to leverage graph theory and optimization algorithms to efficiently plan and coordinate the cooperative mission. Similarly to Chapter 3, where the problem is modeled for a single UAV equipped with multiple batteries, obtained results can be interpreted for multiple UAVs instead of the batteries.

As in Chapter 3, the field graph contains nodes that represent waypoints that need to be visited by a set of UAVs to map the entire field, whereas the edges represent possible flight paths for the UAVs, linking the nodes in a predetermined order, as well as linking the waypoints with the recharging base candidates denoting potential mission starting and ending locations. The objective of the path planning optimization problem is to determine a flight plan regarding relevant metrics such as energy consumption or expected travel distance and time, considering individual UAV capabilities as limitations.

As a cooperative multi-UAV mission is considered here, coordination of the UAVs needs to be ensured on a strategic level. First, we assume reliable data exchange between UAVs, i.e. flight plans for all the UAVs are known in the system. The second point concerns task allocation, by distributing tasks among UAVs to balance workload and optimize the performance of the entire system and mission itself.

There are numerous challenges and considerations to be addressed. The non-exhaustive list includes the following:

- Scalability: efficient management of the path planning for a large number of UAVs covering large areas.

- Real-time adaptation: dynamical path adapting in response to environmental changes or new information.

Path planning for the multi-UAV mapping mission is defined as the task allocation problem, where the waypoints of the field area are distributed among available UAVs, while respecting their battery capacity levels. The proposed optimization problem is defined as minimizing the cost in eq. (3.1), under the constraints in eq. (3.2)-(3.11), as developed in Chapter 3 for a single-UAV mission. The set of UAVs is interpreted as a set of their respective batteries $\mathbb{J}$.

Unlike the energy-aware battery allocation in Chapter 3, where the overall flight distance directly impacts the total mission time, mission time is significantly reduced here due to the possibility of simultaneous flight of all available UAVs, i.e., through parallelization of the missions for the UAVs in the system. However, energy consumption still remains a priority, and by minimizing the flight distance, we can achieve the lowest possible battery consumption. As the optimization problem defined beforehand results in reducing the number of batteries or UAVs employed for the mapping, mission safety is consequently increased by eliminating unnecessary take-offs and landings. Additionally, using fewer UAVs minimizes possible collision risk, and facilitate coordination of the multi-UAV system.

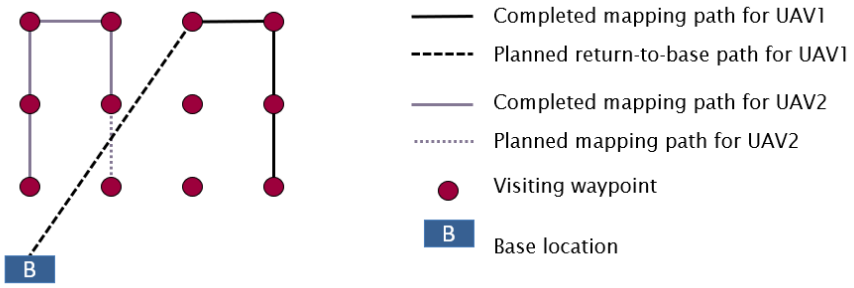


Figure 5.1: Example of collision risk when one UAV is returning to base, while the other is completing its planned mission.

Every multi-UAV mission needs to consider the risk of collision, whether it is of a higher or a lower level. In an agricultural mapping mission, where the mapping back-and-forth path configuration excludes the risk on the strategic planning level, control on the tactical level needs to prevent any eventual unintended collision during the mission. On a tactical level, a collision can happen due to multiple reasons. For example, if a UAV completes the flight of its planned subpath, during the return-to-base flight, the trajectory of this UAV might intersect with the one of another UAV that is still completing its mapping, as in Figure 5.1. Moreover, in case of insufficient battery for path completion, return-to-

base needs to occur before it was initially planned, and the updated flight plan can be in collision with another UAV. Also, the risk of collision can arise simply due to external disturbances, such as wind, or system uncertainties, where the accuracy of the tracking deteriorates, and the resulting trajectory deviates from the reference path. Therefore, collision avoidance will be addressed in this chapter on the control level.

5.3 Passing priority allocation

In order to enhance the multi-UAV mission performance and safety, attributing different levels of passing priority to the UAVs involved in the same mission can aid in resolving a collision conflict. Priority allocation has been discussed mainly on the path planning level as in [116], where passing priority is implemented for the UAV formations in complex environments. The priority while planning a surveillance mission was discussed in [117]. In order to evaluate the trajectories generated by the proposed trajectory planner, tasks are assigned a different priority level, regarding energy consumption, flight risk, and surveillance area priority.

On the other hand, the benefits of the priority strategy on the trajectory tracking level were presented in [84]. The goal of such implementation is to eliminate unnecessary maneuvers in the formation control, where undesired chain maneuvers are avoided as fewer UAVs need to handle collision avoidance. As the risk of collision arises, not all the concerned agents need to react and alternate their trajectories accordingly to avoid the foreseen collision. Collision avoidance maneuvers often include abrupt changes in control inputs, leading to more significant energy loss and deviation from the planned path. As their behaviors are coupled, unplanned changes in the path of one agent can greatly affect changes and the need for the response of the other one. Thus, the overall impact of collision avoidance efforts could be reduced by allocating the passing priority.

Priority allocation rules need to be clearly defined, where the hierarchy of the UAVs in a multi-UAV system is determined along the mission. In the agricultural mapping mission, higher passing priority should be allocated to the UAV that is performing the mapping flight, i.e. visiting the waypoints with a certain level of position accuracy. As it should not deviate from the reference trajectory, the other UAV that finishes its flight and flies back to the base station location needs to handle collision avoidance, and is, therefore, assigned a lower passing priority.

Communication between the UAVs in the system, as well as the information exchange are assumed to be established. Only the UAV with lower passing priority needs to ensure safety distance between

UAVs to prevent a collision. It takes into account the predicted paths of other UAVs and reacts when their paths come in close proximity.

The UAV with a higher priority does not consider the risk of collision and continues to follow its path without changes and disruptions, focusing solely on its trajectory tracking and arrival at the desired position. The other UAV with lower priority, thus, handles collision avoidance as its primary concern before going back on the reference path as the risk of collision has passed. This way, not all agents need to perform complex maneuvers. Fewer alternations in their paths lead to increased safety and a decrease in the energy needed for the flight, as less energy is consumed for direct paths. Thus, the priority benefits can also be interpreted from the battery consumption perspective.

5.4 Collision avoidance control strategy

5.4.1 Distributed NMPC for collision avoidance

Distributed Model Predictive Control (DMPC) aims to lower the computational complexity of the centralized optimal control problem by dividing it into subproblems for each subsystem, here UAV. Subsystems are coupled through dynamics [118], constraints [119], or cost functions [120].

In DMPC, each UAV in the system is an agent in set $\mathcal{N}$, and implements its own control strategy to track the reference output. A distributed approach to MPC has been extensively researched and used in applications such as field mapping. In distributed MPC [121], each UAV computes its own optimal control inputs with respect to the predicted behavior of the system based on the model, as well as the predicted behavior of neighboring UAVs, while respecting constraints. Thus, it provides better scalability and tractability compared to centralized coordination [122].

Additionally, collision avoidance needs to be considered on a tactical level in case of risk of multiple UAVs being in close proximity. Therefore, to ensure the safe flight of all the UAVs, additional measures need to be included while defining a control strategy suitable for the mission.

Different approaches to collision avoidance in multi-UAV systems have been proposed in the literature using DMPC. In [123, 124, 125], collision avoidance is imposed as a nonlinear constraint while computing the optimal trajectory for each UAV. In [126], collision avoidance is handled in the cost function, and its significance can be adjusted depending on the need for collision avoidance. Staying within a defined safe flight corridor is another approach to avoid collision and smoothen the trajectory. In [127], a feasible flight corridor is constructed based on a graph of interconnections in the multi-obstacle environment. Flight corridors in [128] are defined by a radius around the sphere waypoint,

which allows the UAV to perform more natural turns while remaining inside the imposed corridor.

5.4.2 Proposed collision avoidance strategy

The distributed approach implies that each UAV i in the set $\mathcal{N}$ computes its own control inputs based on the predicted outputs $\hat{\mathbf{y}^i}$ over a certain prediction horizon of length $N_p \cdot T_e$. Predicted outputs are acquired based on the model (2.32)–(2.35). The objective of the control law is to track the reference output $\mathbf{y}^i_{ref}$ with precision.

The real system's output is presumed to be free from errors and noise, with all state variables being measured accurately. The disturbances d acting on the system are considered bounded, albeit unknown and unmeasured. Consequently, to achieve offset-free control using Nonlinear Model Predictive Control (NMPC), the model output $\mathbf{y}^{i,model}$ is augmented with the measured error and multiplied by the prediction horizon time step n (equation 5.1, similarly to the approach considered in Chapter 4). This approach compensates for the propagation of the corresponding error over the prediction horizon N_p. As it is the case in NMPC for a single UAV, presented in Chapter 4, the model output $\mathbf{y}^{i,model}$ is provided by a nominal model, without uncertainties, $\alpha_i = 1$, nor external disturbances, $d_i = 0$.

$$\hat{\mathbf{y}}^i_{k+n} = \mathbf{y}^{i,model}_{k+n} + n(\mathbf{y}^i_k - \mathbf{y}^{i,model}_k), \quad k \geq 0, \quad n \in [1, N_p], \quad i \in \mathcal{N} \tag{5.1}$$

Control is handled in a distributed manner, with output information exchanged between the UAVs. The schematic diagram in Figure 5.2 illustrates the distributed control with allocated priority for the case of a multi-UAV system with 2 UAVs. Here, UAV2 is given a higher passing priority and is computing its control input according to the NMPC with constant velocity based on its own predicted output, as in Chapter 4. On the other hand, UAV1 is given the lower passing priority and needs to handle collision avoidance in addition to trajectory tracking by considering both its own predicted output and the predicted output of the UAV2. Hereafter, the focus will be on the control strategy of UAV1.

The proposed DMPC formulation for the collision avoidance problem is formulated as a multi-objective NMPC, similarly to [129], where cost function criteria have state dependency imposed by defined conditions.

In distributed MPC for agricultural mapping, each UAV is an individual subsystem that solves its optimal control problem autonomously in a distributed manner while considering the estimated states of other UAVs in order to complete their cooperative mission while avoiding collision.

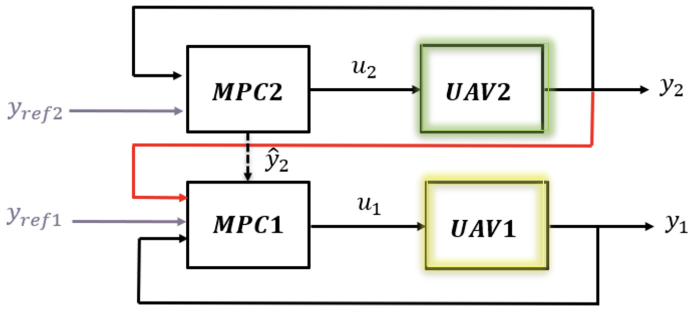


Figure 5.2: Distributed MPC schematic diagram.

As two desired objectives are in conflict, a compromise needs to be reached. When one UAV is in the proximity of another UAV, the importance of collision avoidance increases. On the other hand, if the agent has no threat of encountering another one, this objective can be neglected. The proposed solution is, therefore, taking into account an additional time-varying, state-dependent decision criterion for collision avoidance with constant weights, while solving the optimal control problem for trajectory tracking. The main advantage of the proposed approach is to transform the optimization problem into an unconstrained one, reducing the computational burden and complexity, especially for online control determination.

All UAVs compute their own control inputs to track the reference trajectory. When there is no risk of collision, all the UAVs solve the optimal control problem with an emphasis on trajectory tracking by minimizing the error between the desired and actual outputs. However, if the risk of collision arises, UAVs take action with respect to the attributed priority in the multi-UAV system. Only the UAV with lower passing priority maximizes the distance between them to prevent a collision. This UAV takes into account the predicted paths of the respective neighboring UAV and reacts when their paths come in close proximity. After they pass the intersecting point at risk, the higher importance returns to trajectory tracking in the optimal control problem. As collision avoidance is handled in the cost function, the optimal problem remains unconstrained. However, the bounds on the control inputs are still considered. The aim of this approach is to reduce the computational complexity to allow online implementation. Moreover, as collision avoidance is handled only if there is a risk of intersecting paths, the tracking accuracy is not affected when unnecessary.

Hereafter, the proposed strategy will be presented in more detail for the case of 2 UAVs.

5.4.3 Mathematical formulation of distributed NMPC for collision avoidance

5.4.3.1 Notation

Table 5.1: Trajectory tracking NMPC variables and tuning parameters for a multi-UAV system

NMPC variables for a multi-UAV system	
$\mathcal{N}$	Set of UAVs in the multi-UAV system
$\hat{\mathbf{y}}_k^i$	Predicted system output for UAV i at time instant k
$\mathbf{y}_k^{i,ref}$	Reference output states for UAV i at time instant k
u_k^i	Control input for UAV i at time instant k
Δu_k^i	Variation between two consecutive control inputs for UAV i at time instant k and $k-1$
$\|\hat{v}_k^i\|_2$	Norm of the predicted output velocity for UAV i at time instant k
$v_c^{i,ref}$	Reference constant mapping velocity for UAV i
$d_{ij,k}$	Real distance between the UAVs i and j at time instant k, $i,j \in \mathcal{N}$
d_{min}	Minimal distance allowed between a pair of UAVs
A_{ij}	State-dependent sigmoid function depending on d_{ij} and taking values $[0,1]$
NMPC tuning parameters for a multi-UAV system	
N_p	Prediction horizon
T_e	Sampling time
Q_i	Weight matrix for the output cost for UAV i
R_i	Weight matrix for the input cost for UAV i
W_i	Weight scalar for the velocity cost for UAV i
G_{ij}	Weight scalar for the collision avoidance cost between a pair of UAVs, $i,j \in \mathcal{N}$
d_s	Safety distance between a pair of UAVs, $i,j \in \mathcal{N}$
S	Security factor for safety distance
γ	Tuning parameter for the sigmoid function A_{ij}

The notation necessary for the proposed control approach is given in Table 5.1, including the system variables, as well as the tuning parameters to be selected for the optimal NMPC performance. The table includes variables and parameters, as is the case in a single-UAV mission, as well as additional

ones that are introduced to address collision avoidance.

5.4.3.2 Distributed NMPC for constant velocity trajectory tracking

When the priority is allocated to a pair of UAVs in a cooperative mapping mission, higher passing priority is given to the UAV that is completing its mapping task at the moment of collision risk. That way, this UAV continues to track its reference trajectory without deviations. The NMPC for the UAV with the highest allocated priority considers tracking cost, control cost, and velocity cost, as it was defined for a single-UAV mission in Eq. 4.6. We assume that the UAV with the lower priority has sufficient level of energy to complete the collision avoidance maneuver and land safely. The cost function of the considered NMPC to be minimized is as follows:

$$J_i(u^i_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i} + \|\Delta u^i_{k+n-1}\|^2_{R_i} + \|\|\hat{v}^{i,ref}_{k+n}\|_2 - v^{i,ref}_c\|^2_W \right], \tag{5.2}$$

subject to:

$$u \in \mathcal{U}, \tag{5.3}$$

where $\mathcal{U}$ is the set containing bounded functions with given lower bound u_{min} and upper bound u_{max}, i.e. $u_{min} \leq u \leq u_{max}$, representing dynamical limitations of the system.

The cost function includes three costs: tracking error, control smoothness, and constant velocity error costs. The importance of maintaining the constant velocity for the mapping is addressed in the last part of the cost function, as in a single-UAV mapping mission. Therefore, the control strategy of the UAV with the higher passing priority focuses on completing the mapping task, without needing to deviate from its reference path to avoid potential collision.

5.4.3.3 Distributed NMPC for collision avoidance

In the proposed distributed NMPC strategy, collision avoidance is handled by a UAV with the lower passing priority, which is supposedly not performing the mapping task in the moment of collision risk. In case of reference path modifications due to unforeseen conditions, the lower passing priority is allocated to the UAV, which has its reference path changed. This UAV needs to handle collision avoidance by ensuring that the minimal safety distance between the cooperative UAVs is respected.

Generally, collision avoidance is ensured by imposing a nonlinear constraint (as in [123]), where it ensures that minimal safety distance is respected along the trajectory. However, relaxing the collision

avoidance constraint can be transformed into a state-dependent penalty cost. That way, we proceed with simplifying the solving of the optimization problem. The resulting problem is an unconstrained optimization with the constraints imposed exclusively on the control inputs, which serve as control saturation, and therefore, managing the feasibility of solving the optimization problem online can be avoided.

Thus, the newly formulated cost function includes a new criterion whose weight depends on the proximity of the UAV that needs to be prevented, and is formulated as follows:

$$J_i(u^i_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i} + \|\Delta u^i_{k+n-1}\|^2_{R_i} - \sum_{j\in\mathcal{N}, j\neq i} A_{ij}\|d_{ij,k+n}\|^2_{G_{ij}} \right], \tag{5.4}$$

subject to:

$$u \in \mathcal{U}. \tag{5.5}$$

When a collision avoidance task is allocated to the lower-passing priority UAV, the constant velocity is irrelevant. In this NMPC formulation (5.4), in addition to minimizing the tracking error and successive change in control inputs as in (4.4), the last term in the cost function aims to maximize the distance between the UAVs with the weight factor G_{ij}. Depending on the immediate distance $d_{ij,k}$ at time k, weight A_{ij} can take values in the interval $[0, 1]$:

$$A_{ij} = \frac{1}{1 + e^{\gamma \cdot D}}\ , \tag{5.6}$$

where

$$D = d_{ij} - d_{min}, \tag{5.7}$$

$$d_{min} = d_s * (1 + S). \tag{5.8}$$

Here, minimum safety distance d_{min} is multiplied by the safety factor S to account for the prediction and to ensure respecting the defined safety distance d_s. Depending on the difference D between the distance d_{ij} and d_{min}, A_{ij} can take a value as in Figure 5.3.

A_{ij} is a sigmoid function that helps avoid numerical issues due to the choice of a binary term

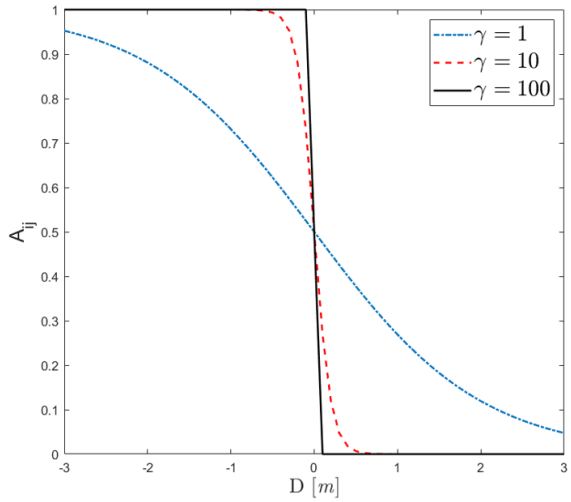


Figure 5.3: Weight function A_{ij} for different values γ.

(switch between 0 and 1 depending on the distance), similar to the sigmoid activation function in the neural network. Tuning parameter γ determines the change from 0 to 1, i.e. determines how long during the collision avoidance maneuver the UAV needs to deviate from its reference path before focusing on the trajectory tracking.

In summary, the proposed NMPC cost function consists of:

- Weighted **output cost** (tracking error):

  $\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i}$, with the weight matrix Q_i,

- Weighted **input cost** (control smoothness):

  $\|\Delta u^i_{k+n-1}\|^2_{R_i}$, with the weight matrix R,

- Weighted **collision avoidance cost** (Maximization of the distance between UAVs):

  $A_{ij}\|d_{ij,k+n}\|^2_{G_{ij}}$, with the weight scalar G_{ij} and a state-dependent parameter A_{ij}.

Algorithmic representation of the NMPC for the UAV with the lower passing priority that implements the proposed collision avoidance strategy is given in Algorithm 2.

To summarize, the proposed control strategy for a multi-UAV system consists in solving the optimization problem with an additional criterion in the cost function. As long as the two UAVs are at a safety distance, without the risk of collision, the control strategy resembles the classical NMPC. However, if the distance starts to pose a safety risk, the term in A_{ij} takes a more important value, and the penalty cost to maximize distance becomes active, and collision avoidance becomes a priority. On the

Algorithm 2 NMPC for trajectory tracking with collision avoidance for a lower passing priority UAV

1: Initialize states $\mathbf{x_0^i}$, $\mathbf{x_0^j}$ and control inputs u_0^i,
2: $\quad i, j \in \mathcal{N},\ j \neq i$
3: Get output positions $p^i(t_0)$, $p^j(t_0)$ and control inputs $u^i(t_0)$,
4: Compute $d_{ij} = \|p^i(t_0) - p^j(t_0)\|_2$
5: Compute $A_{ij}(d_{ij})$
6: Minimize (5.4)
7: Return $y^i(t), u^i(t), \quad t = (t_0, t_0 + N_p)$
8: Apply $u^i(t_0)$
9: Compute $y^i(t_k), \quad k = 1$
10: Set $t_1 \rightarrow\ t_0, y^i(t_1) \rightarrow\ y^i(t_0), u^i(t_1) \rightarrow\ u^i(t_0)$
11: Repeat Step 3. - Step 10.
12: **until** End of simulation

other hand, priority allocation makes sure that only one UAV is handling collision avoidance, whereas the other UAV implements the control strategy suitable for mapping, as presented in Chapter 4.

5.5 Simulation results and discussion

In this study, passing priority is given to UAV2, which aims to track the trajectory with accuracy, while maintaining the reference constant velocity. On the other hand, UAV1 is attributed a lower passing priority, and therefore, responsible for collision avoidance as the risk arises along the trajectory.

The layout of the reference paths of two UAVs in the studied mission is given in Figure 5.4 and the relative coordinates of the starting and ending waypoints in Table 5.2. The reference altitude for both UAVs remains constant at $20\ m$, as the returning-to-base flight is done at the same altitude as mapping, and prior to landing.

Table 5.2: Reference waypoints for a 2-UAV mapping mission

Waypoint	UAV1 (lower priority)	UAV2 (higher priority)
$start$	$(11, 2, 20)m$	$(1, 1, 20)m$
$finish$	$(-4, 26, 20)m$	$(16, 26, 20)m$

Model parameter values for both UAVs in the mission are chosen as in a single-UAV case (Table 4.2). Tuning parameters and control input bounds for the simulation test case are given in Table 5.3.

Tuning parameters for collision avoidance safety distance d_s and γ are chosen according to the size of the UAV. Here, simulations are performed for the UAV model DJI Matrice 350 RTK, with the dimensions (430×420×430) mm, when folded, with propellers, and diagonal wheelbase of 895 mm. Therefore, the safety distance is chosen at $d_s = 1\ m$ to allow for secure passage of the two UAVs. The distance is calculated between the centers of gravity of the two UAVs.

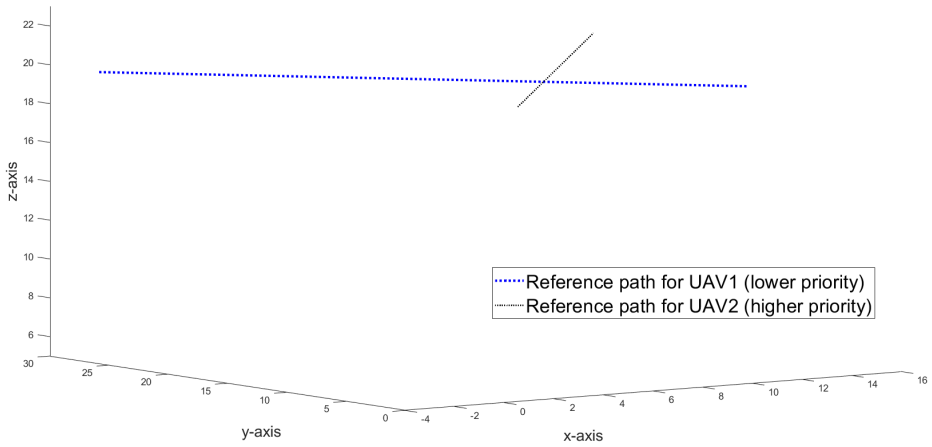


Figure 5.4: Reference path for the mapping mission with two UAVs (UAV1 - blue, UAV2 - black).

Table 5.3: NMPC tuning parameter values and control input bounds for $\mathcal{N}$ = 2

Tuning parameter		Value
Prediction horizon	N_p	10
Sampling time	T_e	$0.1\ s$
Output cost weight	$Q_1 = Q_2$	$diag(1, 1, 5, 0, 0, 0)$
Control input cost weight	$R_1 = R_2$	$diag(1, 10^2, 10^2)$
Velocity cost weight	W_2	1
Collision avoidance cost weight	G_{12}	5
Safety distance	d_s	$1\ m$
Security factor	S	0.5
Tuning parameter for A_{ij}	γ	5
Reference constant velocity	$v_c^{2,ref}$	$3\ m/s$
Control input		Bounds (UAV1, UAV2)
Thrust	T	$5\ m/s^2 \leq T \leq 15\ m/s^2$
Reference roll	ϕ_{ref}	$\phi_{ref} \leq \lvert 30° \rvert$
Reference pitch	θ_{ref}	$\theta_{ref} \leq \lvert 30° \rvert$

All simulation results in this chapter were obtained by solving the proposed distributed NMPC control problem using *fmincon* algorithm in Matlab. The initial state vectors for two UAVs in the control problem are $\mathbf{x_1} = [11, 2, 20, -1.5, 2.4, 0, 0, 0]^\top$, and $\mathbf{x_2} = [1, 1, 20, 1.5, 2.4, 0, 0, 0]^\top$. The reference constant velocity for the higher priority UAV2 is set at $3\ m/s$, according to the reference path configuration in this simulation case. Velocity components of both UAVs are initialized with reference to the desired velocity size and direction.

5.5.1 Nominal case

A nominal scenario denotes the absence of external disturbances and model uncertainties for both UAVs in the cooperative multi-UAV system. Consequently, in the nonlinear model equations (2.32)-(2.35), the disturbance vectors $d_1 = d_2$ are set to $[d_x, d_y, d_z]^T = [0, 0, 0]^T$, and the parameters $\alpha_1 = \alpha_2$ are equal to 1, as there is no uncertainties about the thruster efficiency.

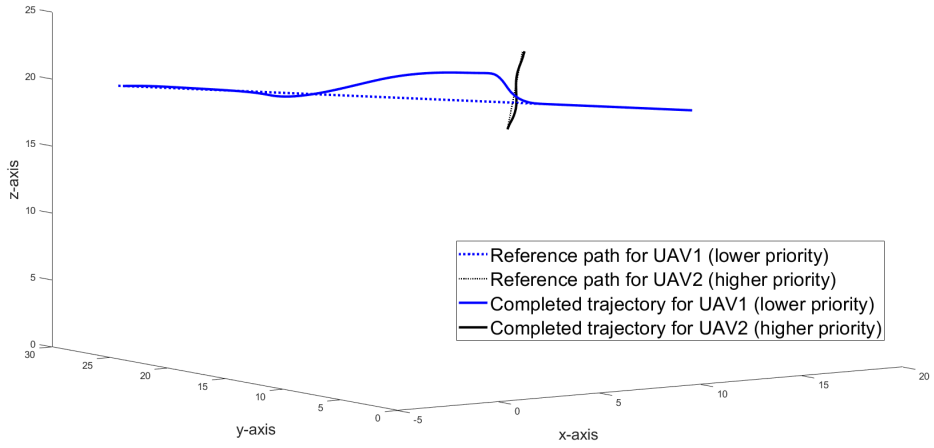


Figure 5.5: Resulting trajectories for the two UAVs, where UAV1 (blue) is handling collision avoidance.

Figure 5.5 illustrates resulting trajectories for both UAVs in a simulation test case with intersecting paths. The UAV with allocated passing priority keeps tracking its reference path, whereas the UAV that handles collision starts to deviate from its reference path before the intersection point, regarding the imposed safety distance. While it is assumed that all the states are measured and the UAV is aware of the states of the other UAV due to the full exchange of information, the predictive nature of the control strategy directly implies this anticipation in collision avoidance maneuvers.

As the risk of collision decreases, the respecting UAV changes the focus of its control strategy to trajectory tracking and aims to reach the final reference position with a high level of accuracy.

More precise insights can be seen in Figure 5.6, where a separate view of position for each axis is linked to time. It can be seen that the risk of collision arises at the intersection of the two reference paths around $t = 3.5\ s$. UAV1, therefore, starts the collision avoidance maneuver at $t = 3\ s$ in all three directions, in order to avoid the collision in due course. In the given example, the altitude of the UAV1 increases by approx. $1m$, but there are changes in $x-$ and $y-$ directions as well at the beginning of the collision risk. It takes about $5s$ for the UAV1 to return to its reference position.

Safety distance is defined considering the UAV size and is set to $d_s = 1\ m$. The distance between the two UAVs is computed along the trajectory and shown in Figure 5.7. The minimal distance value amounts to $1.04\ m$, which means that the collision is successfully avoided.

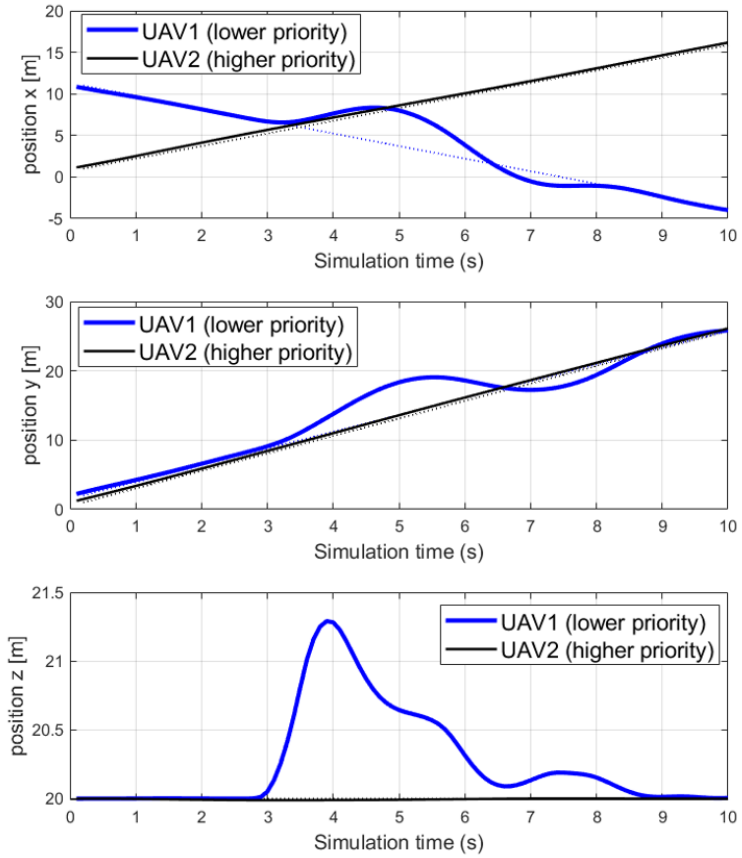


Figure 5.6: Reference and resulting positions in x, y and z-direction for UAV1 (blue) and UAV2 (black).

The resulting velocity components in three directions are shown in Figure 5.8. UAV2, which is performing the mapping, has its velocity stable along the trajectory that does not change the direction. It is the result of the constant velocity cost part of the NMPC for the higher-priority UAV. The other UAV, UAV1, due to the collision avoidance maneuvers that start around $t = 3\ s$, its velocity also deviates from that point and stabilizes slowly towards the end. A similar behavior can be seen in the resulting roll ϕ and pitch θ angles, where the leading UAV holds the low values of both angles, whereas the lower priority UAV changes the values of the angles while remaining within the imposed bounds.

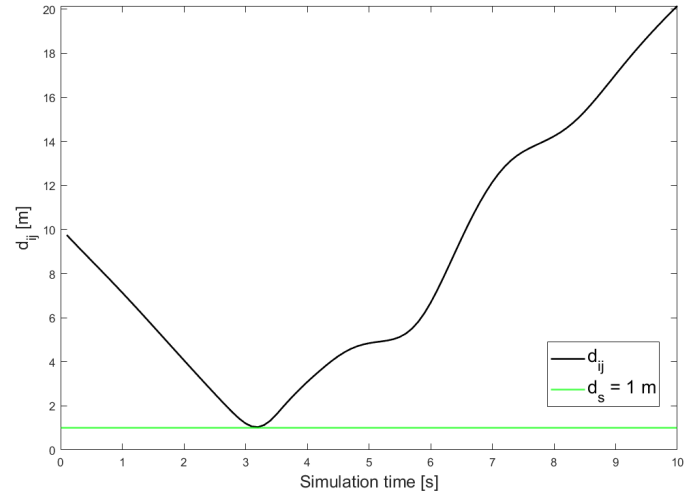


Figure 5.7: Distance between the two UAVs in the mission along their trajectories.

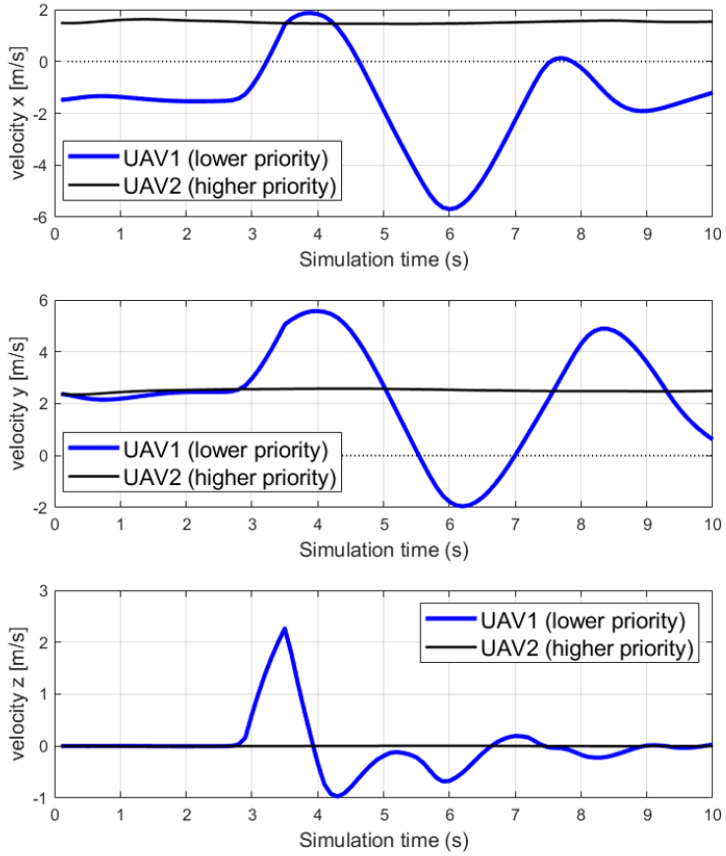


Figure 5.8: Reference and resulting velocity components in x, y and z-direction for UAV1 (blue) and UAV2 (black).

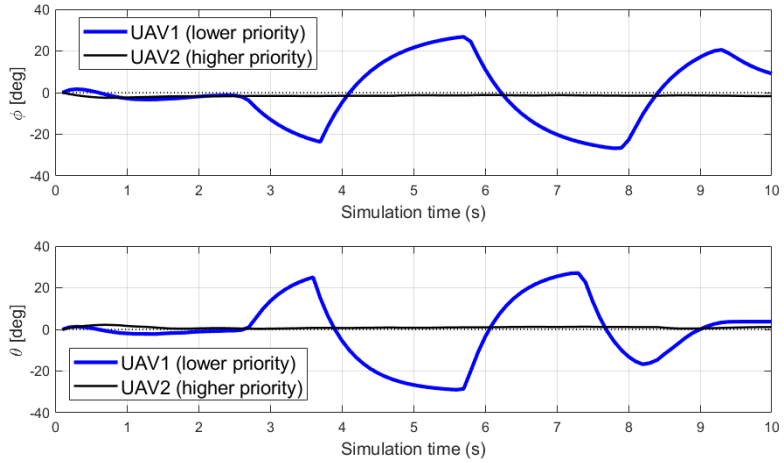


Figure 5.9: Reference and resulting roll ϕ and pitch θ angles for UAV1 (blue) and UAV2 (black).

The norm of velocities for both UAVs is shown in Figure 5.10. As there are no requirements imposed on keeping the constant velocity for UAV1, the velocity starts to deviate from the initial values as the collision maneuver starts to happen. As this maneuver involves abrupt changes in direction, the constant velocity is not maintained. However, the importance of introducing the priority is emphasized as the higher priority UAV both tracks its reference trajectory without the need for deviation and keeps its reference constant velocity. Therefore, it continues to perform its mapping task seamlessly.

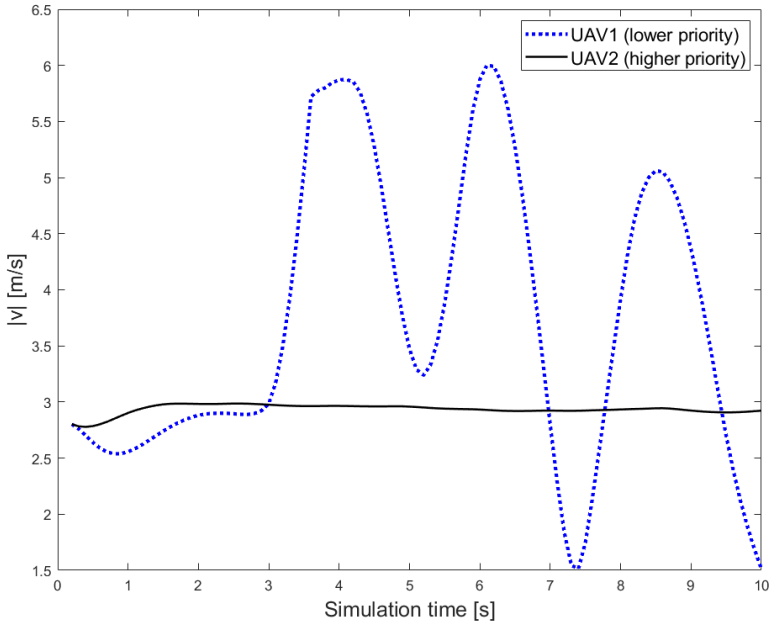


Figure 5.10: Norm of velocity for UAV1 (blue) and UAV2 (black).

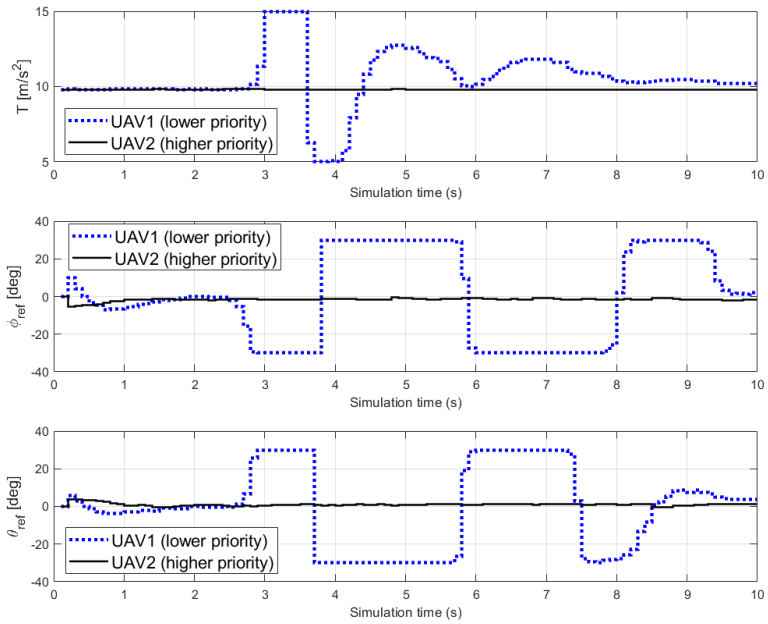


Figure 5.11: Resulting control input values for thrust T, reference roll ϕ_{ref} and pitch θ_{ref} angles, for UAV1 (blue) and UAV2 (black).

Control inputs of the UAV handling collision avoidance in Figure 5.11 reach saturation point. More specifically, saturation of all the control inputs happens around $t = 3s$, just as the UAV is beginning to perform the collision avoidance maneuver. It seems to be rather abrupt, and therefore, the advantage of the passing priority is highlighted. That way, the UAV with the higher passing priority avoids aggressive changes in trajectory and, as such, has its resulting control smoothened. Nonetheless, the saturation is reached only for the reference angles of the UAV1, while its resulting angles (Figure 5.9) in reality attain smoother transitions.

Finally, control inputs in Figure 5.11 can be interpreted from the energy consumption perspective. Changes in control inputs that are captured for UAV1 result in higher energy consumption. By allocating priority, we decrease the need for abrupt changes for one of the UAVs in the system, which results in lower energy consumption for the entire multi-UAV system.

The comparison between collision avoidance with and without priority allocation is given in Appendix B, where the benefits of prioritized collision avoidance are demonstrated in more detail.

5.5.2 Robustness assessment

In order to test the robustness of the previously presented control strategies for collision avoidance, the analysis of the Monte Carlo simulations will be presented here. Test cases include constant external disturbances d, and uncertainty of the model parameter of the thruster efficiency α.

Monte Carlo simulations included 50 test cases with randomly added external disturbances d_1, d_2, and thruster efficiency α_1, α_2, different for the two UAVs. The external disturbance values vary in the range $d_{x,1}, d_{x,2} = [-3, 3]\ m/s^2$, $d_{y,1}, d_{y,2} = [-3, 3]\ m/s^2$, $d_{z,1}, d_{z,2} = [0, 2]\ m/s^2$, while for the thruster uncertainty $\alpha_1, \alpha_2 = [0.8, 1.2]$.

The robustness is evaluated for the proposed collision avoidance strategy with distributed NMPC for the lower passing priority UAV, and is compared with the NMPC, where collision avoidance is handled as a nonlinear constraint, as in the equations below:

$$J_i(u^i_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i} + \|\Delta u^i_{k+n-1}\|^2_{R_i} \right], \tag{5.9}$$

subject to:

$$\|d_{ij,k+n}\|_2 \geq d_s, \quad i,j \in \mathcal{N}, \quad j \neq i \tag{5.10}$$

$$u \in \mathcal{U}, \tag{5.11}$$

where the cost function contains the trajectory tracking and control smoothness cost, and collision avoidance is ensured by respecting the nonlinear constraint in equation 5.10, which requires that the distance between the UAVs i and j is greater than the imposed safety distance d_s along the entire trajectory.

As the robustness of the higher passing priority UAV was assessed in Chapter 4, this part will focus on the results for collision avoidance robustness assessment for the lower passing priority UAV, which is UAV1 in this simulation case. The configuration of the simulation case remains as it was for the nominal case, and all the parameter values, control input bounds and reference positions are given in Table 5.3 and 5.2, respectively. The reference intersecting paths are illustrated in Figure 5.4.

As the two objectives are in conflict, the trade-off between position accuracy and collision avoidance, which causes deviations from the reference trajectory, needs to be reached. As the lower-level priority UAV is not performing the mapping mission, but is flying to the base location, only the final position error of the simulation case will be considered, rather than the average RMSE along the trajec-

tory. Additionally, when choosing the best NMPC strategy for the use case, computational complexity plays a major role, as the control approach needs to enable a real-time implementation.

Results of the same test cases are compared between the proposed NMPC collision avoidance strategy, referred to as a *penalty cost*, and the *nonlinear constraint* added to a classical NMPC.

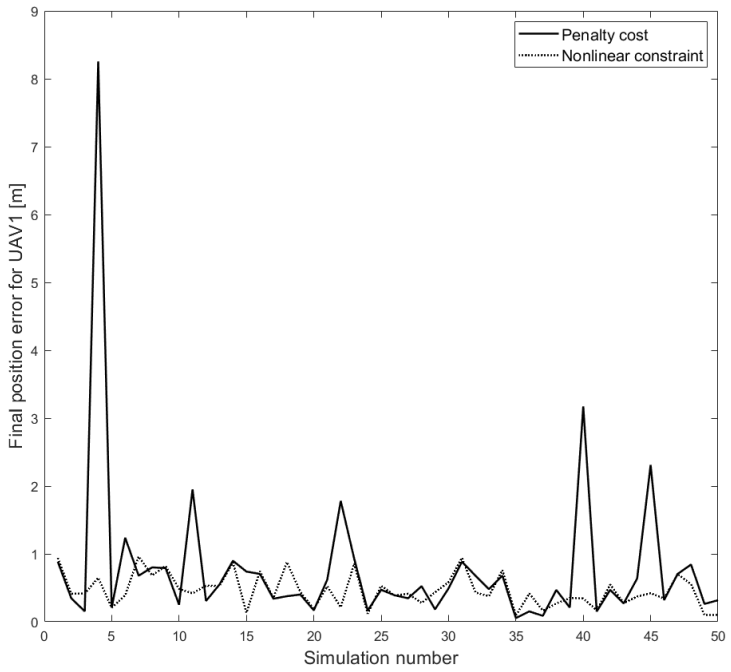


Figure 5.12: Final position error at the end of simulation time for UAV1 for 50 Monte Carlo simulations with random values of constant external disturbances d_1, and model parameter uncertainty of the thruster efficiency α_1.

Figure 5.12 shows the resulting position error for UAV1 with lower priority at the end of each simulation. When considering all the test cases, the mean final position error for penalty cost is $0.78\ m$, whereas it amounts to $0.46\ m$ for the constraint approach. Even though that can be translated to nearly $40\ \%$ less for the nonlinear constraint, the final error is not a focus of the collision avoidance problem. Moreover, such a difference arises from the large deviation in case 4, whereas in most cases, the results are quite similar.

On the other hand, Figure 5.13 shows that computational complexity in terms of CPU time is significantly lower for the relaxed optimal problem, where collision avoidance is handled as a penalty cost, in comparison with the nonlinear constraint. With a mean value of $0.65s$ for the penalty cost, in comparison with a mean value of $1.35s$ for constraint, the computational complexity is lowered by $52\ \%$ with the proposed unconstrained approach. This is an important insight for the potential real-time

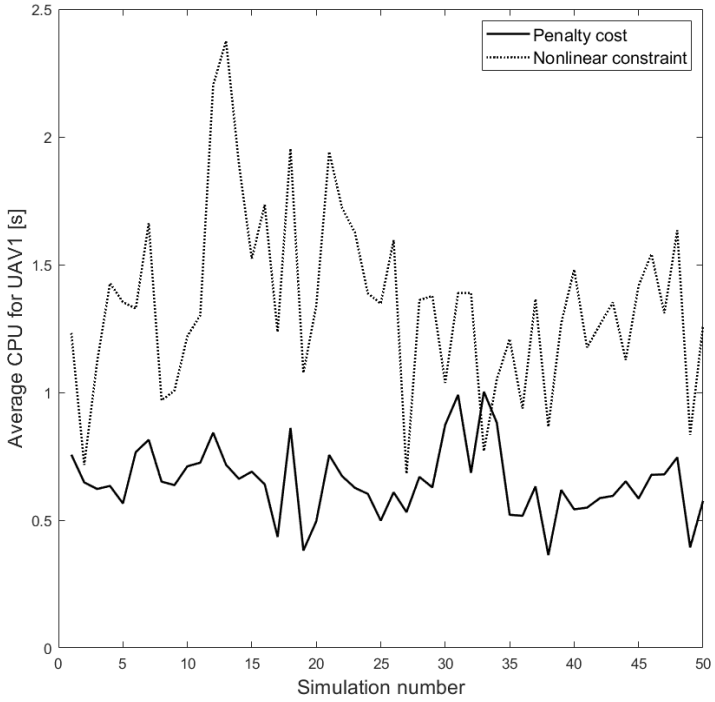


Figure 5.13: Resulting average CPU time for solving the NMPC control problem for UAV1 for 50 Monte Carlo simulations with random values of constant external disturbances $d1$, and model parameter uncertainty of the thruster efficiency α_1.

implementation.

In Figure 5.14, the green line illustrates the imposed safety distance $d_s = 1\ m$. When analyzing the minimal distance along the trajectories in all the Monte Carlo test cases, it can be concluded that, in the case of the penalty cost, the safety distance is not respected for 5 cases. However, in the worst-case scenarios, where minimal distance reaches the lowest values, the penalty cost performs better than the nonlinear constraint, as is the case in simulation number 4 and 45.

When comparing all three indicators: final position error, minimal distance, and computational time, the proposed unconstrained NMPC for collision avoidance, where the sole constraints are bounds on the control variables, seems to represent a good compromise between the accuracy, performance and ability to handle collision avoidance.

However, in order to fully ensure collision avoidance in real conditions, additional measures, e.g., flight corridor, should be established. That way, the ambiguity of the false prediction due to large disturbances and uncertainties can be removed. The additional analysis where the flight corridor is considered is given in Appendix C. Results insinuate that the flight corridor is sufficient for cases where

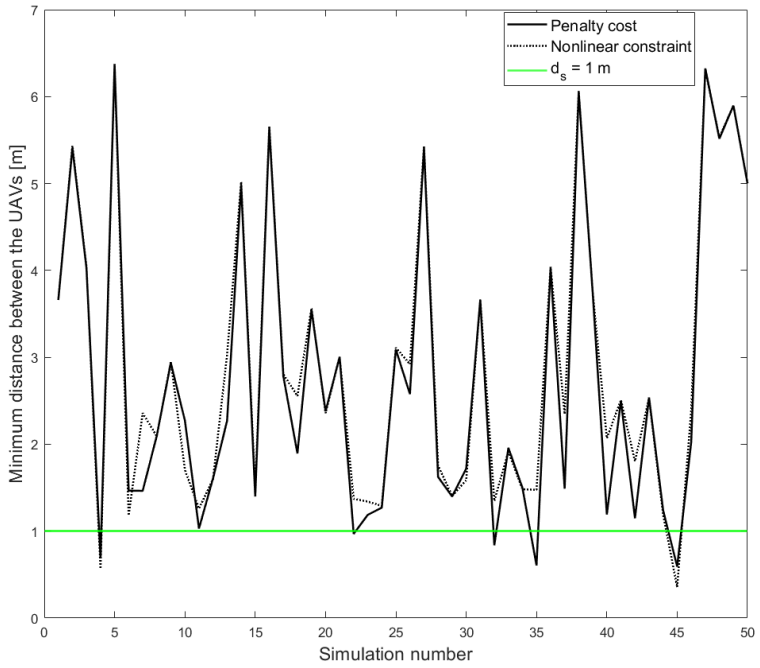


Figure 5.14: Resulting minimal distance between the two UAVs along their trajectories for 50 Monte Carlo simulations with random values of constant external disturbances d_1, d_2, and model parameter uncertainty of the thruster efficiency α_1, α_2.

the planned paths of the UAVs do not intersect. This method can also help enhance the robustness of such cases.

Appendix C also compares different collision avoidance strategies (including penalty cost, nonlinear constraint, and flight corridor). However, the method detailed in this chapter is chosen as the one that has demonstrated the best compromise between performance, computation time, and implementation complexity, which is to handle collision avoidance as a penalty cost.

5.6 Conclusion

Employing multiple UAVs in a cooperative mission with a shared task can significantly reduce time and effort for the mission completion. However, certain challenges need to be addressed, such as coordination between the UAVs that need to ensure safety by avoiding possible collisions.

The proposed distributed NMPC introduces the priority allocation mechanism, where only one UAV is assigned the responsibility of managing collision avoidance at any given time. By allocating the priority, there is a reduced risk of unwanted and unexpected coupled maneuvers, and path alterna-

tions do not provoke additional changes for the entire multi-agent system. This strategy simplifies the control problem and minimizes unnecessary maneuvers, thereby conserving energy.

In the proposed control approach, collision avoidance was incorporated into the control problem as a penalty cost, effectively relaxing the control problem while maintaining safety. This method ensures that the UAVs remain on their optimal paths unless a collision is imminent, in which case the avoidance maneuvers are triggered.

The optimal control problem for a multi-UAV agricultural mapping mission was evaluated with a focus on efficiency and robustness. The proposed approach significantly improves the efficiency of the mapping mission by reducing the overall computational time required for solving the optimal control problem. The simulation results demonstrate a certain level of robustness of our control strategy against disturbances and uncertainties, showcasing the system's ability to adapt and maintain performance in dynamic and unpredictable environments. However, in certain cases, the safety distance was not respected. Therefore, additional mechanisms should be considered to improve the robustness level.

This study sets the groundwork for the implementation of multiple UAVs in a cooperative mapping mission. Nonetheless, further exploration and development in this dynamic field is envisioned.

Chapter 6

Application to mapping of an agricultural field

6.1 Introduction

The rapid integration of UAVs, particularly quadrotors, into precision agriculture, has opened new opportunities for efficient and detailed agricultural mapping. In order to enhance the capabilities of a single quadrotor for agricultural mapping, three critical challenges are addressed with the proposed energy-aware mission planning: path planning optimization, battery allocation optimization, and base location optimization. A systematic experimental approach in this chapter was aimed at demonstrating how these optimizations could improve the efficiency, coverage, and quality of agricultural mapping missions.

The traditional methods of agricultural monitoring often face limitations in spatial resolution, temporal frequency, and susceptibility to environmental conditions. In contrast, quadrotors equipped with advanced sensors and imaging technologies can provide high-resolution, real-time data essential for precise agricultural tasks and beyond. However, the operational constraints of quadrotors, particularly related to battery life and flight path efficiency, necessitate innovative solutions to maximize their potential.

The energy-aware battery management algorithm for path planning, developed in Chapter 3, was extended for experimental use cases and implemented on a real system to minimize flight distance and optimize coverage. Battery allocation optimization strategy was employed to extend mission du-

ration, and strategic base location positioning was utilized to reduce energy consumption and downtime. The results of these optimizations were evaluated through field experiments, focusing on key performance metrics such as battery usage, mission duration, and overall mission efficiency.

6.2 Experimental study

6.2.1 Quadrotor energy-aware mapping mission

For this experimental study, multiple base locations can be used in the same mission, i.e., they can be combined when allocated to one battery subpath, such that the operator moves across the field area. Nonetheless, the requirements for accessibility remain the same, with an additional possibility of moving from one location to another seamlessly and in a reasonable amount of time.

The optimization strategy for battery allocation in Chapter 3 is extended with the base location choice included in the optimization problem. Therefore, certain adjustments need to be made to allow for a multi-base choice optimization. Unlike the approach modeled in Chapter 3, where the choice of the base location was made iteratively, for the experimental part, this choice is included within the optimization problem. The main motivation for such modification is to allow different base locations to be chosen within the same mission. While the UAV takes off at one base location, it can land at a different location, to further optimize the mission time and battery use. The time for subpath mission completion allows the operator to move across the field area if necessary.

In that sense, in addition to K segments that are linking $K+1$ nodes, K_b segments linking each waypoint to each potential base location $b \in \{1, \ldots, B\}$ need to be introduced. Thus, node k_b now denotes each waypoint k individually attributed to each base b, $k_b \in \{1, \ldots, K_b\}$, where $K_b = (K + 1) \times B$. Coordinates of the base location possibilities are predefined.

Modified decision variables are given in Table 6.1. The decision variable $x_{j,k}$ is not considered by the choice of the base location, as it designates covering the waypoints without their link to the base station.

Table 6.1: Decision variables for a multi-base battery allocation optimization problem

Decision variables	
$x_{j,k}$	Allocation of the segment k to the battery j
u_{j,k_b}	Allocating node k_b as the first waypoint of the subpath for the battery j
v_{j,k_b}	Allocating node k_b as the last waypoint of the subpath for the battery j

The modified ILP optimization problem is as follows:

$$\min_{x,u,v} \sum_{j \in \mathbb{J}} \quad \left[\sum_{k=1}^{K} x_{j,k} l_k + \sum_{k_b=1}^{K_b} (u_{j,k_b} l_{b,k_b} + v_{j,k_b} l_{b,k_b}) \right] \tag{6.1}$$

subject to:

$$\sum_{k=1}^{K} x_{j,k} l_k \frac{1}{v_c} + \sum_{k=1}^{K_b} (u_{j,k_b} l_{b,k_b} + v_{j,k_b} l_{b,k_b}) \frac{1}{v_{max}} \leq t_j, \quad \forall j \in \mathbb{J} \tag{6.2}$$

$$\sum_{j=1}^{J} x_{j,k} = 1, \quad \forall k \in \{1, \ldots, K\} \tag{6.3}$$

$$\sum_{k_b=1}^{K_b} u_{j,k_b} \leq 1, \quad \forall j \in \mathbb{J} \tag{6.4}$$

$$\sum_{k_b=1}^{K_b} v_{j,k_b} \leq 1, \quad \forall j \in \mathbb{J} \tag{6.5}$$

$$\sum_{k_b=1}^{K_b} u_{j,k_b} = \sum_{k_b=1}^{K_b} v_{j,k_b}, \quad \forall j \in \mathbb{J} \tag{6.6}$$

$$\sum_{k_b=1}^{K_b} k v_{j,k_b} - \sum_{k_b=1}^{K_b} k u_{j,k_b} \geq \sum_{k_b=1}^{K_b} u_{j,k_b}, \quad \forall j \in \mathbb{J}, \quad k = k_b - (\left\lfloor \frac{k_b}{(K+1)} \right\rfloor - 1)(K+1) \tag{6.7}$$

$$x_{j,k} + \sum_{l=(k+(b-1)(K+1))+1}^{b \cdot (K+1)} u_{j,l} \leq 1, \quad \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K\}, \forall b \in \{1, \ldots, B\} \tag{6.8}$$

$$x_{j,k} + \sum_{l=1+(b-1)(K+1)}^{k+(b-1)(K+1)} v_{j,l} \leq 1, \quad \forall j \in \mathbb{J}, \forall k \in \{1, \ldots, K\}, \forall b \in \{1, \ldots, B\} \tag{6.9}$$

$$\sum_{k=1}^{K} x_{j,k} = \sum_{k=1+(b-1)(K+1)}^{b\cdot(K+1)} k v_{j,k} - \sum_{k=1+(b-1)(K+1)}^{b\cdot(K+1)} k u_{j,k}, \quad \forall j \in \mathbb{J}, \forall b \in \{1, \ldots, B\} \tag{6.10}$$

$$x_{j,k} \leq u_{j,k_b} + x_{j,k-1}, \quad \forall j \in \mathbb{J}, \forall k \in \{2, \ldots, K\}, k_b = k + (b-1)(K+1), \forall b \in \{1, \ldots, B\} \tag{6.11}$$

The equations (6.1)-(6.11) resemble equations (3.1)-(3.11) presented in Chapter 3, with the choice of the base location linked to the decision variables u_{j,k_b} and v_{j,k_b}. In this extension, all the $K+1$ nodes are linked to B bases. Therefore, in equations where the decision variable $x_{j,k}$ is placed with decision variables u_{j,k_b} and/or v_{j,k_b}, there is a need of normalizing subscripts k and k_b. This is done via equations where $k = f(k_b)$ in Eq. 6.7 and, similarly, $k_b = f(k)$ in Eq. 6.11.

The modified problem considers a 2D trajectory for each segment, as well as for the base, i.e., all the waypoints, including the base, are set at the equivalent altitude. Thus, vertical take-off and landing are not included in the optimization problem. We consider a certain security level for initial battery capacity in order to allow for safe take-off and landing.

Because take-off and landing are executed out of the optimization solution scope, and the altitude is kept constant throughout the entire mapping task, the evaluated problem considers a 2D path layout.

6.2.2 Experimental setup

The experimental validation of the proposed optimization strategy is performed on a quadrotor with a mounted multispectral camera. DJI quadrotor model Matrice 350 RTK of dimensions (430×420×430) mm (L×W×H), when folded, with propellers, and with a diagonal wheelbase of $895mm$, is suitable for performing mapping missions in an agricultural setting, and is shown in Figure 6.1. With two TB65 batteries, the total weight comes to approx. $6.47\ kg$ can reach the horizontal velocity of $23\ m/s$. Acquiring NDVI images, a measure of the amount and vigor of vegetation on the land surface, is provided with two Micasence RedEdge MX Multispectral NDVI cameras, shown in Figure 6.2. The gimbal of the UAV ensures camera stabilization despite the changes in attitude.

To facilitate the validation of the path planning optimization with battery allocation, the mapping velocity, as well as the velocity of flying from/to the base location, are set to be equal due to the characteristics of the DJI mission implementation system on the DJI Pilot 2 app, i.e. $v_c = v_{max}$ in all the experiments. Trajectories are tracked using a DJI internal guidance system that prioritizes the accu-

racy of the given waypoints in a mission route in a continuous mode, without stopping at waypoints, and, therefore, cannot maintain the reference's constant velocity when performing the turning maneuvers, unlike the NMPC for trajectory tracking proposed in Chapter 4. As the UAV slows down at each turn, the time difference needs to be accounted for in the optimization problem. Thus, the input velocity $v_c = v_{max}$ for the battery allocation algorithm is considered to be lower than in the real-time implementation.

(a) (b)

Figure 6.1: DJI Matrice 350 RTK with mounted multi-spectral camera: (a) Top view (left-hand side); (b) Front view (right-hand side).

Figure 6.2: Two Micasence RedEdge MX Multispectral NDVI cameras that are mounted on the DJI quadrotor.

DJI Matrice 350 RTK uses a pair of TB65 Li-Ion batteries. Based on experience, a pair of batteries can perform a flight of approx. $30\ min$ on full discharge. To evaluate the proposed battery allocation optimization approach, we have 3 batteries available to be employed in each mapping mission. Their capacities vary as they have a different initial state-of-charge (SoC), which is lower than 100%. Open-

circuit voltage for SoC estimation with a charge and discharge cut-off voltage of a Li-ion cell of approx. $4.2\ V$ and $2.5\ V$, respectively, given in [130], can be simplified, such that the discharge is done in a linear manner for the most part. DJI batteries can be recharged to 70% in $20\,\mathrm{min}$ with a fast charge.

Because of the linearization of the battery discharge behavior, energy consumption will be considered linear, regardless of the SoC at the beginning of the experiments.

6.2.3 Experimental validation

Experimental validation was done on two test sites in Croatia with different shapes and characteristics. At the test site Jazbina, the area of interest represents a part of the vineyard that was mapped and resulted in an NDVI image for the purpose of vegetation density analysis. The second test site, Borongaj, is an area that is interesting for an infrastructural project. The point cloud resulting from LiDAR imaging can be of interest for distinguishing vegetation, ground, and different specific objects.

For both test sites, the battery allocation optimization problem was evaluated and compared with a benchmark solution from [77], where the allocation is done with regard to the proportional capacities of available batteries.

6.2.3.1 Test site Jazbina, Croatia

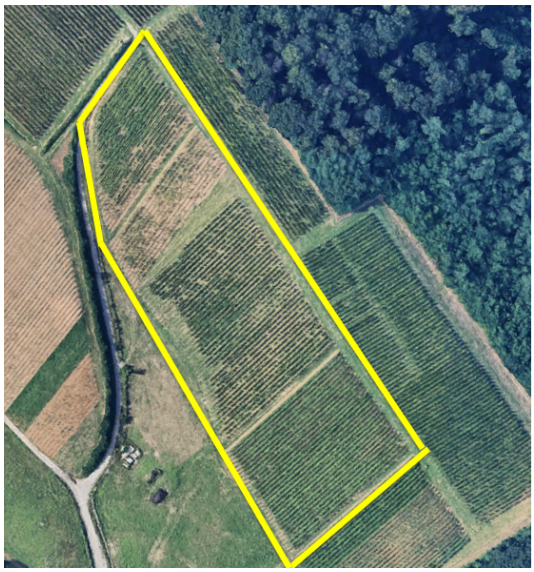


Figure 6.3: Test site Jazbina, area of interest for the mapping task framed in yellow.

Jazbina is a vineyard test site of the Faculty of Agriculture at the University of Zagreb. For this experiment, the chosen area of interest framed in yellow is shown in Figure 6.3. The area of interest has an

irregular shape and is accessible by a road on the left side. The base locations can be placed along the edges of the field, as the terrain is flat enough for the operator to perform take-off and landing.

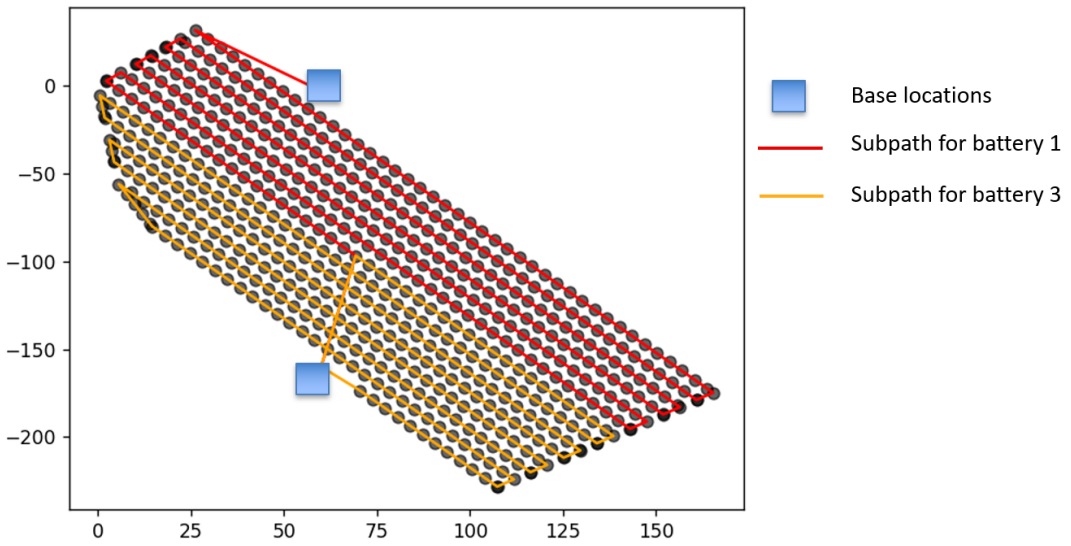


Figure 6.4: Optimal solution for the battery allocation in simulation for the area of interest at the test site Jazbina with two base locations (red: subpath for battery 1, yellow: subpath for battery 3).

Of the three initially available batteries, two were employed in the mapping mission, and the optimization solution is given in Figure 6.4. Area decomposition alongside the longer edge resulted in 546 waypoints that are distanced at $6\ m$, with the exceptions near the edges, where the waypoints are placed more closely. There are two potential base station locations. One is placed close to the first waypoint, and the other is located closer to the last waypoint in order to prevent long passages for flights from/to the base station. Following the multi-base battery allocation optimization formulated in this chapter, both base locations can be used during a given mission. Regarding the size of the optimization problem, there are 10920 decision variables and 11482 constraints for this test case. Figures 6.5 illustrate the two-stage complete mapping mission in a Google Earth view.

(a) (b)

Figure 6.5: Google Earth view on planned paths for mapping of the test site Jazbina: (a) Battery 1 (left-hand side); (b) Battery 3 (right-hand side).

Table 6.2 represents the solution after simulation and is compared with the results after the experiment performed on a real quadrotor system. Out of three possible batteries, only two were used. Optimization resulted in not needing to employ battery 2.

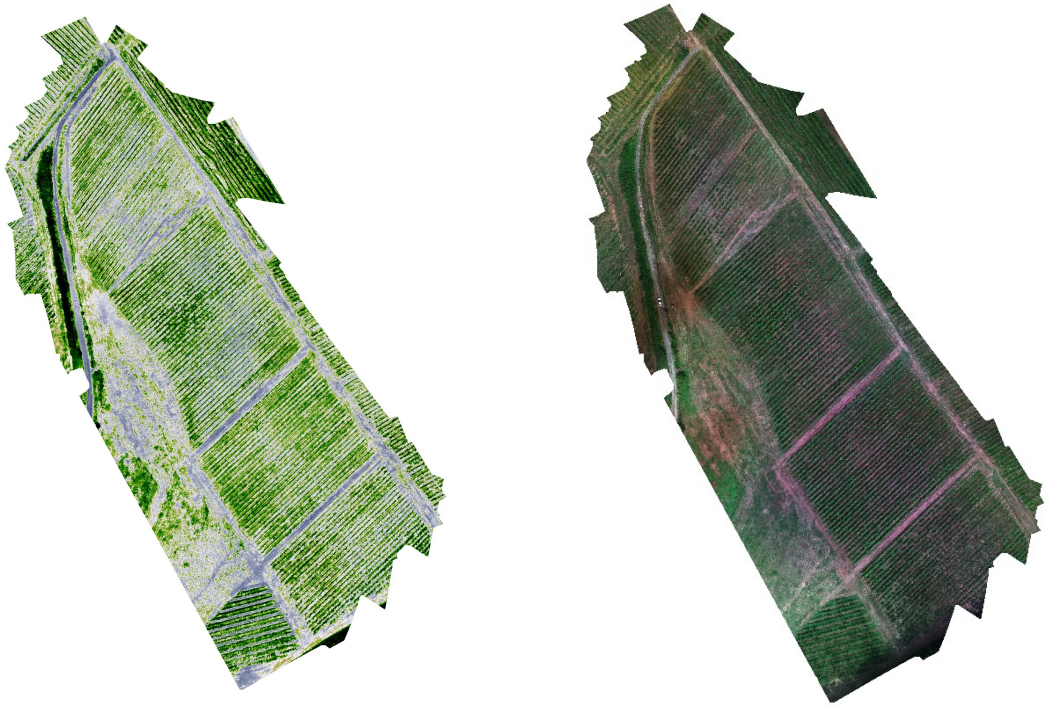
In experiments, initial capacities are reduced in relation to the actual capacities in order to account for the safe take-off and landing of each battery. The beginning of the mission and measurement, therefore, starts after the initial take-off and ends before the landing, both of which are at the position of a given base location and at the altitude of the mapping mission at $50\ m$. Velocity is considered to be constant in the simulations during the mapping task for each battery. However, quadrotor type DJI cannot maintain the reference velocity at turning waypoints and, therefore, utilizes more time and, consequently, more battery. Thus, the constant velocity v_c set in simulations is decreased by $20\ \%$. This way, the resulting mission times in simulations are slightly overestimated when compared to the experimental mission duration. The final percentage of each of the two batteries employed in the mission is superior to the estimated consumption, but it is essential to preserve the minimum level of battery capacity for safe mission completion. The experimental results are, therefore, satisfying as they represent a small difference compared to the simulations, but on the safe side. The real total mission flight time amounts to just under $13\ min$.

As the quadrotor is performing a continuous flight without stopping at each waypoint, the images are acquired with a timer set to make a snapshot with the multi-spectral camera every $1\ sec$. Once the mapping mission is finalized, the snapshots acquired in a logical order are fused into a complete

Table 6.2: Simulation and experimental results of battery capacities used for the optimized mapping mission plan at the test site Jazbina

Battery j	1	2	3
Initial battery capacity	$13\ min$	$8\ min$	$11\ min$
Battery employment duration (simulation, $v_c = 4\ m/s$)	$7\ min\ 18\ sec$	$0\ min$	$6\ min\ 42\ sec$
Battery employment duration (experiment, $v_c = 5\ m/s$)	$6\ min\ 43\ sec$	$0\ min$	$6\ min\ 15\ sec$
Battery percentage at the beginning of the experiment	$43\ \%$	$27\ \%$	$37\ \%$
Battery percentage at the end of the experiment	$22\ \%$	$27\ \%$	$20\ \%$

image. Figures 6.6a and 6.6b represent the final result as NDVI and RGB images, respectively. For agricultural purposes, NDVI is an interesting indicator as it quantifies vegetation of an agricultural field. In this case, the intensity of the color green in Figure 6.6a represents the vegetation density. The lighter green indicates lower-density vegetation, such as grass, whereas darker green indicates the lush vegetation, here, the vines.



(a) (b)

Figure 6.6: Google Earth view on planned paths for mapping of the test site Jazbina: (a) NDVI (left-hand side); (b) RGB (right-hand side).

- **Comparison of the optimal solution against the proportional approach**

  The optimal solution previously presented was evaluated against the proportional benchmark approach from [77]. Because the proportional approach does not involve optimization, a single

base location is considered. The proportional solution for three available batteries is shown in Figure 6.7.

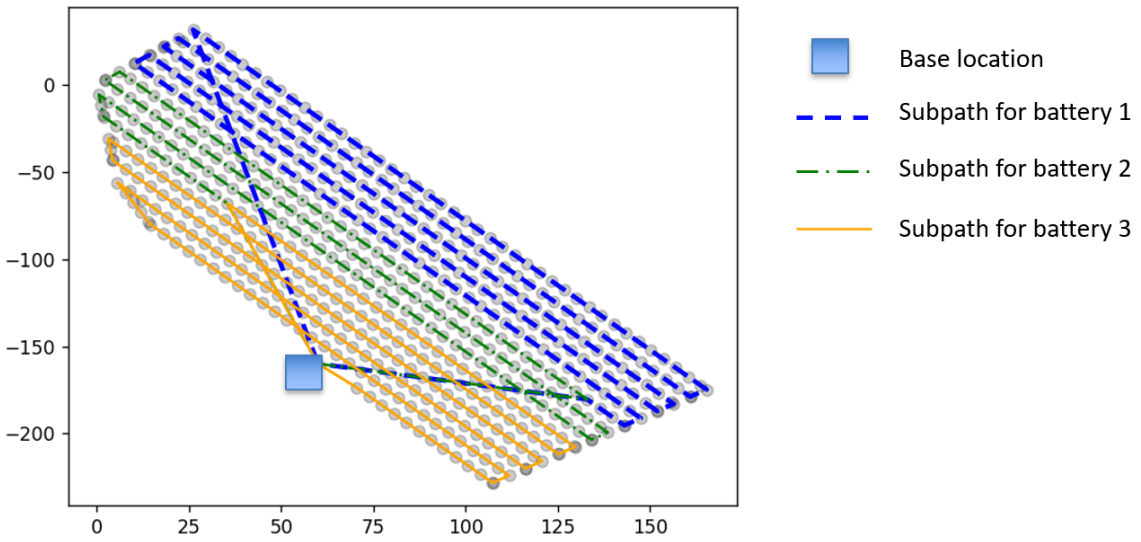


Figure 6.7: Proportional approach solution for the battery allocation in simulation for the area of interest at the test site Jazbina with one base location (dashed blue: subpath for battery 1, dash-dot green: subpath for battery 2, yellow: subpath for battery 3).

The results of the comparison are given in Table 6.3. In the proportional scenario, all the batteries are employed for the equivalent of approx. $50\,\%$ of their initial capacities. On average, before each mission subpath flight, the preparation lasts $5\ min$, which includes battery placement or replacement, mission route upload, as well as take-off and landing. With regard to employment time that varies from approx. $3\ min$ to $7\ min$, the preparation time of $5\ min$ represents an important portion of total mission time. Therefore, by reducing the number of batteries employed in the mission, the total mission duration in this particular case was reduced by $23\,\%$. This result also emerges from the base location optimization, as the optimal approach uses both possibilities in order to enhance the mission efficiency. In addition, the level of mission safety was also increased in the optimized plan, as there were fewer take-offs and landings performed.

Table 6.3: Comparison of the experimental results for optimal and proportional strategy for the test site Jazbina

		Optimal strategy	Proportional strategy
Battery 1 Initial capacity: $13\ min$	Mission preparation time Mission employment duration	$5\ min$ $6\ min\ 43\ sec$	$5\ min$ $5\ min\ 52\ sec$
Battery 2 Initial capacity: $8\ min$	Mission preparation time Mission employment duration	$0\ min$ $0\ min$	$5\ min$ $3\ min\ 43\ sec$
Battery 3 Initial capacity: $11\ min$	Mission preparation time Mission employment duration	$5\ min$ $6\ min\ 15\ sec$	$5\ min$ $5\ min\ 9\ sec$
Total mission duration		$22\ min\ 58\ sec$	$29\ min\ 44\ sec$

- **Comparison of the optimal solution against the suboptimal area configuration**

Based on the study in [2], the optimal configuration of a CPP problem is the back-and-forth (BF) area decomposition along the longest side of the area of interest because it enables the minimum number of turns. Turning maneuvers require change in acceleration and reducing velocity at turning points. By reducing the turning points, the velocity profile during the mapping mission is more uniform, and a shorter time is required to complete the mission. Consequently, this configuration results in a lower energy consumption.

The optimal configuration result in Figure 6.4 is compared to the solution of the battery allocation optimization for a path configuration perpendicular to the optimal configuration, shown in Figure 6.8. This configuration is issued along the shortest side of the mapping field and includes the same two base location choices. Table 6.4 reflects the increase in the total mission duration when comparing the optimal horizontal configuration to the suboptimal vertical configuration with more turning maneuvers. Both solutions were issued from the proposed optimization strategy. However, the horizontal configuration results in $10\ \%$ lower total mission duration time when compared to the vertical BF configuration. This difference results from a higher number of turns, as the DJI quadrotor reduces its velocity to $v = 1\ m/s$ at each turn.

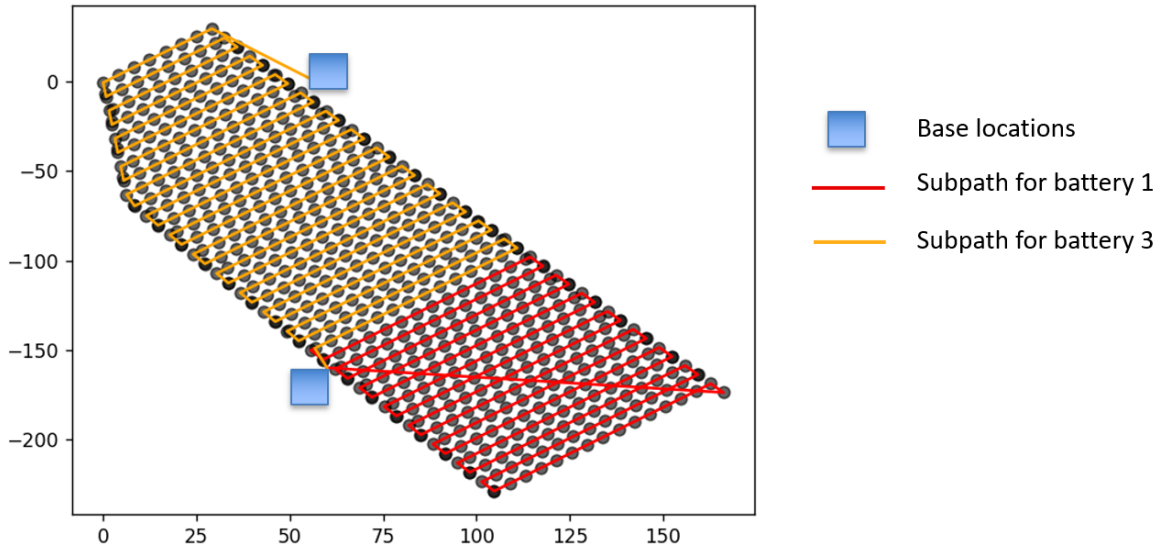


Figure 6.8: Optimal solution for the battery allocation in simulation for the suboptimal vertical configuration of the area of interest at the test site Jazbina with two base locations (red: subpath for battery 1, yellow: subpath for battery 3).

Table 6.4: Comparison of the experimental results for the optimal horizontal and suboptimal vertical BF path configuration for the test site Jazbina

		Horiz. BF configuration	Vert. BF configuration
Battery 1 Init. capacity: $13\ min$	Mission preparation time Mission employ. duration	$5\ min$ $6\ min\ 43\ sec$	$5\ min$ $6\ min\ 35\ sec$
Battery 2 Init. capacity: $8\ min$	Mission preparation time Mission employ. duration	$0\ min$ $0\ min$	$0\ min$ $0\ min$
Battery 3 Init. capacity: $11\ min$	Mission preparation time Mission employ. duration	$5\ min$ $6\ min\ 15\ sec$	$5\ min$ $8\ min\ 53\ sec$
Total mission duration		$22\ min\ 58\ sec$	$25\ min\ 28\ sec$

6.2.3.2 Test site Borongaj, Croatia

The second test site, Borongaj, represents an area that is interesting for infrastructural projects. The full mission area is shown in Figure 6.9.

For mission planning, area decomposition is done along the longest side of the area of interest, such that the number of turns is minimized. The area decomposition results in 435 waypoints dis-

tanced at $6\ m$, except for the edges, where the density of the waypoints is higher. Mission planning results are compared between the proposed optimization strategy and the proportional solution in order to evaluate the benefits of the proposed approach further. For the optimal approach, there are two possible base locations included in the mapping mission, whereas one base location is determined for the proportional approach.

Figure 6.9: Test site Borongaj, area of interest for the mapping task framed in yellow.

Table 6.5: Experimental results of battery capacities used for the optimized mapping mission plan at the test site Borongaj

Battery j	1	2	3
Initial battery capacity	$10\ min$	$8\ min$	$6\ min$
Battery employment duration	$4\ min\ 33\ sec$	$5\ min\ 48\ sec$	$0\ min$
Battery percentage at the beginning of the experiment	$33\ \%$	$27\ \%$	$20\ \%$
Battery percentage at the end of the experiment	$18\ \%$	$7\ \%$	$20\ \%$

Figure 6.10 illustrates the solution of the optimal approach in simulation, and Figure 6.11 the result from the simulated proportional approach. As the area of interest is fairly small, low initial battery capacities are chosen for this evaluation. This case represents an interesting challenge when there is no sufficient time for battery recharge, and available residual capacities can be used to complete the mapping of a certain area. Initial capacities and consumption of each battery is given in Table 6.5, whereas the resulting mission times are given in Table 6.6. The total mission duration is shortened for $19\ \%$ for the optimal approach in comparison to the proportional one. Again, the largest difference results from an additional battery replacement.

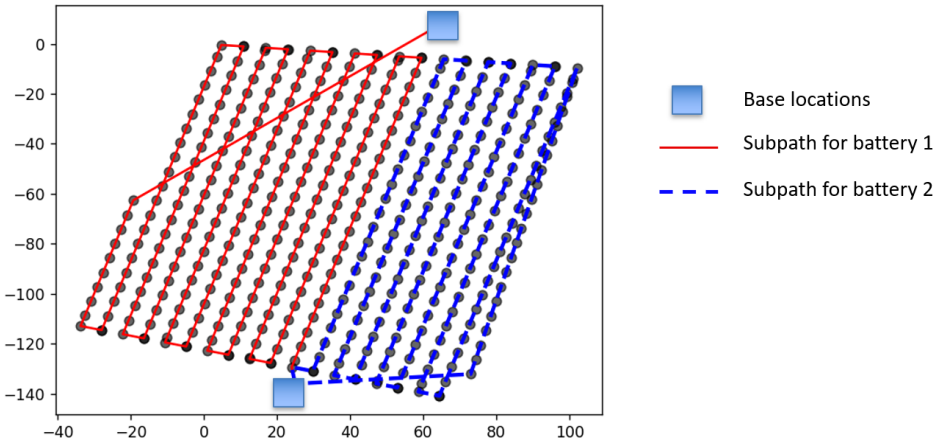


Figure 6.10: Optimal solution for the battery allocation in simulation for the area of interest at the test site Borongaj with two base locations (red: subpath for battery 1, dashed blue: subpath for battery 2).

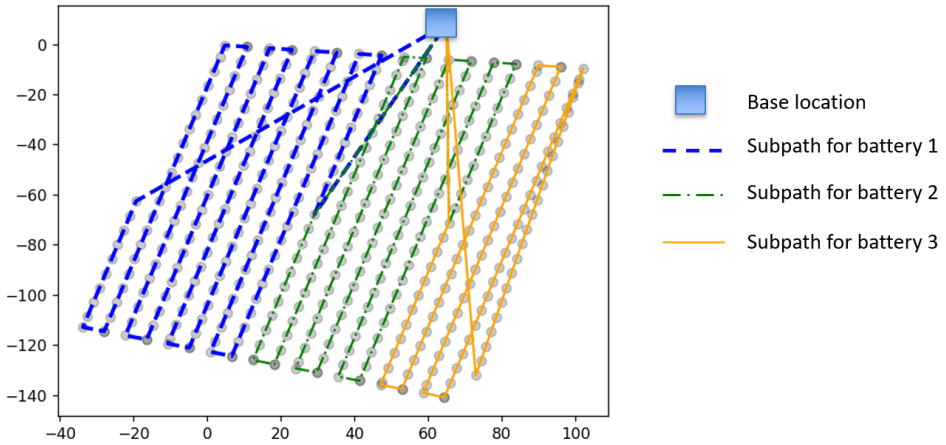


Figure 6.11: Proportional approach solution for the battery allocation in simulation for the area of interest at the test site Borongaj with one base location (dashed blue: subpath for battery 1, dash-dot green: subpath for battery 2, yellow: subpath for battery 3).

The resulting image from UAV mapping is a highly detailed and accurate representation of the site obtained with LiDAR, as shown in Figure 6.12. It can serve several purposes in infrastructural planning as it provides a comprehensive overview of the current state of the infrastructure, including roads, buildings, and natural features.

Figure 6.12: Resulting point cloud obtained with LiDAR representing detailed mapping area at Borongaj test site.

Table 6.6: Comparison of the experimental results for optimal and proportional strategy for the test site Borongaj

		Optimal strategy	Proportional strategy
Battery 1 Initial capacity: $10\ min$	Mission preparation time Mission employment duration	$5\ min$ $4\ min\ 33\ sec$	$5\ min$ $4\ min\ 11\ sec$
Battery 2 Initial capacity: $8\ min$	Mission preparation time Mission employment duration	$5\ min$ $5\ min\ 48\ sec$	$5\ min$ $2\ min\ 52\ sec$
Battery 3 Initial capacity: $6\ min$	Mission preparation time Mission employment duration	$0\ min$ $0\ min$	$5\ min$ $3\ min\ 3\ sec$
Total mission duration		$20\ min\ 21\ sec$	$25\ min\ 6\ sec$

6.3 Conclusion

This experimental study was presented to validate the proposed battery management optimization approach that was done in simulations. In that manner, the benefits of planning the mission in advance, in terms of batteries that affect logistics, were presented. The findings from these experiments highlighted significant improvements in the efficiency and effectiveness of agricultural mapping missions. The optimized path planning aims to consume less battery for tasks that do not include mapping, such as flying to and back to the base station. This was shown in the performed experiments by

reducing the mission flight time by around 20%, as well as using 2 instead of 3 available batteries in both test cases.

Consequently, fewer batteries are needed to complete the mission, and quadrotors are able to operate for longer periods without interruptions. Additionally, the strategic base location positioning contributed to a noticeable reduction in energy consumption, further enhancing the sustainability and operational efficiency of the missions.

The performed experiments show that the total mission duration depends on two factors: the number of batteries employed in a mission, and the path configuration issued from the initial area decomposition. Battery replacement task, as well as individual mission preparation, require significant time, which extends the mission duration. On the other hand, the initial path configuration needs to minimize the number of turning maneuvers, as they also influence the mission efficiency, mainly because of the reduction of speed needed for each turn.

Overall, the integration of these energy-aware mission planning strategies into the use of quadrotors for agricultural mapping represents a significant advancement in precision agriculture. By addressing the critical challenges of path planning, battery management, and base location, these optimizations not only improve the efficiency and coverage of mapping missions but also pave the way for more sustainable and effective use of UAVs in agriculture.

Chapter 7

Conclusion and perspectives

7.1 Conclusion

The motivation for this PhD thesis originates from the growing need to enhance agricultural yields while simultaneously integrating environmental sustainability with economic viability. As global food demand increases, there is an urgent requirement to optimize agricultural practices that not only maximize output but also protect the environment. This research aims to address this by developing a comprehensive approach to agricultural mapping with the use of a single or multiple UAVs. The final goal is to provide valuable, actionable information to farmers, based upon which they can proceed with the decision-making process.

The contribution of this manuscript is threefold.

First, the proposed battery management optimization strategy ensures that mission completeness is considered in the planning phase. Aside from ensuring that the mission is completed with available resources, the introduced planning strategy optimizes the mapping mission from a time, energy, and safety perspective. Consideration of the base location further enhances all of the above-mentioned criteria.

As a second contribution, following the mission plan issued from the optimization strategy, the proposed control strategy based on nonlinear MPC handles accurate trajectory tracking while maintaining the constant mapping velocity. An added layer in the control scheme that mitigates parameter uncertainties in the UAV dynamics model, as well as the external disturbances, improves the robustness of the tracking control. Following multiple simulation results with random values of uncertainties and disturbances, the introduced NMPC is proven to provide good results in unknown conditions.

Lastly, to improve the efficiency of the mapping mission, a multi-UAV system is considered for performing the mapping task to further decrease the time needed for its completion. The main challenge that arises from cooperative multi-UAV missions is to ensure coordination between the UAVs, i.e. successful collision avoidance to guarantee the mission safety. In this thesis, the proposed NMPC control strategy incorporates prioritized trajectory tracking with collision avoidance. By allocating priority in the multi-UAV system, unnecessary maneuvers are eliminated, as only one UAV handles the collision avoidance. Thus, apart from enhanced safety, the energy needed for collision avoidance is also decreased, alongside the computational time for computing the appropriate control inputs. By relaxing the collision avoidance constraint, the proposed multi-objective NMPC is tested in unknown conditions by adding uncertainties and disturbances.

Finally, the successful implementation of the battery management optimization strategy on the real system and in real settings suggests the advantages of the proposed developments in the mission planning phase.

7.2 Perspectives

The developments in this thesis open plenty of possibilities for future work.

Regarding the planning phase of the mapping mission, unforeseen events, such as insufficient battery levels for completion or faults in the system, must be anticipated and accounted for. These unexpected challenges can disrupt the mission's progress, leading to incomplete data collection or system failures. To mitigate these risks, it is essential to implement an online replanning strategy that allows for real-time adjustments to the mission flight plan. This approach ensures that any *ad hoc* modifications necessary to accommodate changing conditions or emergencies can be made promptly, thereby maintaining the operational efficiency and effectiveness of all the UAVs involved in the mission.

Similarly, the criteria for online priority allocation require thorough analysis and integration between the planning and control phases of the mission flight. This is crucial to ensure that resources, such as UAVs and their associated tasks, are dynamically and efficiently allocated in response to real-time changes in mission priorities or environmental conditions. By refining these criteria, the system can prioritize critical tasks, such as data collection in high-risk areas or the immediate return of UAVs facing potential system failures. Integrating this priority allocation seamlessly between the planning and control phases can enhance the overall adaptability and resilience of the mission, enabling more effective management of unforeseen challenges and optimizing mission outcomes.

Furthermore, No-fly zones (NFZs) need to be carefully considered in the path planning phase to ensure safe and efficient UAV operations. These zones, which may include areas with restricted airspace or locations with potential hazards, such as nearby infrastructure, must be identified and incorporated into the mission's flight paths. For instance, static obstacles like an electric grid situated in the middle of an agricultural field that needs to be mapped pose significant risks to the UAVs. By factoring in such obstacles during the path planning stage, the UAVs can be directed to navigate around them, thus preventing collisions and ensuring the continuity of the mapping mission.

When it comes to the control strategy for trajectory tracking, one of the primary challenges of NMPC is its high computational load. Future work should focus on developing approaches to reduce this complexity, possibly through algorithmic optimizations or leveraging high-performance computing resources. This is essential to enable real-time implementation, ensuring that the control strategy can be effectively applied in dynamic and time-sensitive environments. Moreover, as agricultural mapping systems become larger and more complex, scaling NMPC to handle these systems effectively will be crucial, and potential solutions should be explored.

With regard to robustness guarantees, combining NMPC with other control methodologies, such as adaptive control or robust control, can enhance its performance and robustness. Future research should explore these integrations to create more versatile and resilient control systems.

Overall, while the proposed NMPC demonstrates significant potential for improving agricultural mapping, ongoing research and development will be essential to address these challenges and fully realize its capabilities.

Appendix A

Small-scale example solution for the battery management optimization

Results of the optimization problem simulations for a small-scale example are given in this appendix, showcasing differences in results that are caused by different path configurations, namely: back-and-forth movements alongside the longest side (Figure A.1), back-and-forth movements alongside the shortest side (Figure A.2), circular path (Figure A.3), and back-and-forth movements alongside the diagonal (Figure A.4). All the results include a single base location option at the origin of the reference coordinate system. There are three available batteries for the mission flight, with capacities of $0.1min$, $0.05min$, and $0.01min$, respectively. Available capacities are chosen to be small enough to represent the need for battery allocation on a small area of the field, $4m$ x $5m$. The results of the battery allocation, with the overview of the used and remaining capacities in terms of percentage $\%$, are given in Table A.1. To interpret the results accurately, it is necessary to keep in mind that the second battery has exactly half of the capacity of the first battery. In all the cases, the third battery was not employed. This is not a realistic case, as it only represents the impact of the change in the path configuration on the flight mission time, as well as the battery consumption.

From the results, it can be concluded that for an area of a rather small scale, the choice of path configuration impacts the mission duration only slightly. The main factors of the different results are, by far, the position of the base station and the total distance between the waypoints. In a real setting, if the velocity is not maintained constant, the flight duration, as well as the energy consumption can be impacted by the number of turns as well.

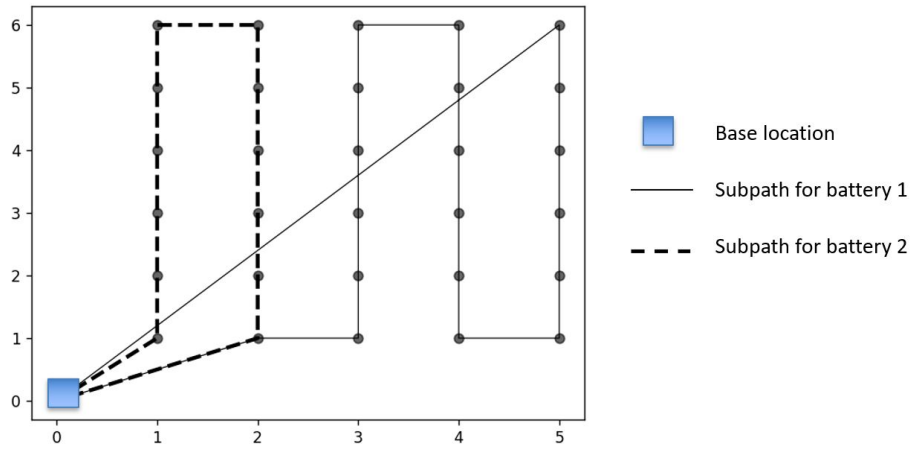


Figure A.1: Optimization results for a flight path configuration of a rectangular field area $4m$ x $5m$: back-and-forth movements alongside the longest side.

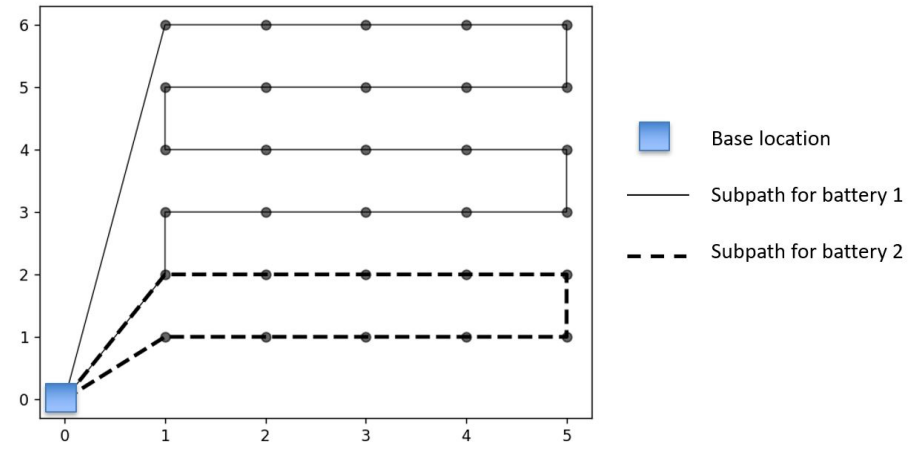


Figure A.2: Optimization results for a flight path configuration of a rectangular field area $4m$ x $5m$: back-and-forth movements alongside the shortest side.

Table A.1: Simulation results of battery capacities used for the small-scale example mapping mission

	Figure A.1			Figure A.2		
Battery j	1	2	3	1	2	3
Battery employment %	73.95 %	83.47 %	0 %	78.22 %	70.14 %	0 %
Remaining battery %	26.05 %	16.53 %	100 %	21.78 %	29.86 %	100 %
Total mission duration	$0.1159\ min$			$0.1132\ min$		
	Figure A.3			Figure A.4		
Battery j	1	2	3	1	2	3
Battery employment %	65.89 %	95.08 %	0 %	97.55 %	96.94 %	0 %
Remaining battery %	34.11 %	4.92 %	100 %	2.45 %	3.06 %	100 %
Total mission duration	$0.1134\ min$			$0.1460\ min$		

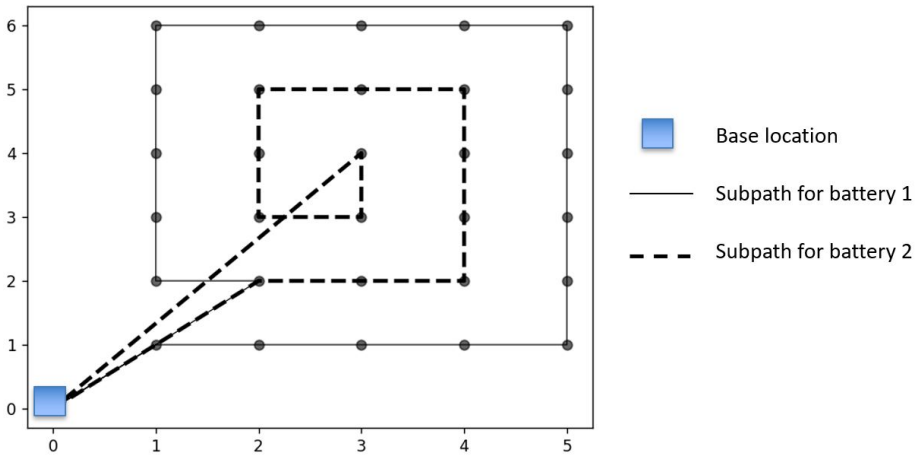


Figure A.3: Optimization results for a flight path configuration of a rectangular field area $4m$ x $5m$: circular movements.

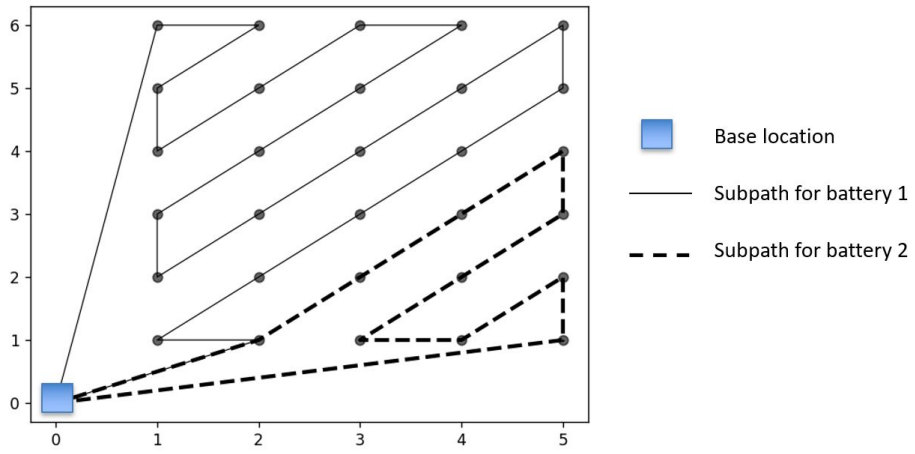


Figure A.4: Optimization results for a flight path configuration of a rectangular field area $4m$ x $5m$: back-and-forth movements alongside the diagonal.

Appendix B

Evaluating the impact of priority allocation in NMPC for collision avoidance

For clarity and simplicity, simulation results of nonlinear trajectory optimization are presented for a multi-UAV system with 2 agents. The simulation is performed for Crazyflie 2.0 Nano Quadcopters, a small and lightweight quadcopter measuring 92 mm between diagonally opposed motor shafts. In all the scenarios, the parabolic configuration of desired paths is defined as follows, $p^{ref} = [p_x^{ref}, p_y^{ref}, p_z^{ref}]^\top$ with:

$$p_x^{ref}(t_k) = (\sqrt{p_x^0} + \frac{\sqrt{p_x^f} - \sqrt{p_x^0}}{T_f} \cdot t_k)^2 \tag{B.1}$$

$$p_y^{ref}(t_k) = p_y^0 + \frac{p_y^f - p_y^0}{T_f} \cdot t_k \tag{B.2}$$

$$p_z^{ref}(t_k) = p_z^0 + \frac{p_z^f - p_z^0}{T_f} \cdot t_k \tag{B.3}$$

This trajectory represents a path between two consecutive waypoints, where the UAVs take snapshots of the field, where the initial position is $[p_x^0, p_y^0, p_z^0]^\top$, and final desired one is $[p_x^f, p_y^f, p_z^f]^\top$ at final time T_f. Reference velocity is zero, $v_{ref} = 0$. Bounds on the control inputs were also considered: $|T| \leq 15$, $|\phi_{ref}| \leq 50°$ and $|\theta_{ref}| \leq 50°$. Values of the parameters used in the simulation study

are given in Table B.1. Weighting matrices are chosen as $Q_1 = Q_2 = diag(10^2, 10^2, 10^2, 10, 10, 10)$, $R_1 = R_2 = diag(1, 1, 1)$.

Table B.1: Parameter values

τ_ϕ, τ_θ	$(0.7, 0.5)s$	g	$9.81m/s^2$
K_ϕ, K_θ	1	A_x, A_y, Az	$(0.1, 0.1, 0.2)s^{-1}$
T_e	$0.1s$	$[p^0_{x,1}, p^0_{y,1}, p^0_{z,1}]$	$[1, 1, 3]m$
T_f	$10s$	$[p^0_{x,2}, p^0_{y,2}, p^0_{z,2}]$	$[5, 1, 3]m$
d_s	$0.3m$	$[p^f_{x,1}, p^f_{y,1}, p^f_{z,1}]$	$[5, 2, 3]m$
G_{12}	200	$[p^f_{x,2}, p^f_{y,2}, p^f_{z,2}]$	$[2, 2, 3]m$

Simulation results for the nominal case, i.e. without model mismatch nor external disturbances ($d = 0$) are presented for the prediction horizon $N_p = 10$. In this case, Agent 1 has a lower priority and successfully avoids collision by alternating its path at the intersection point. Table B.2 shows the absolute final tracking errors and minimum distance between the UAVs along the trajectory. The minimum distance between the agents is slightly below the safety distance $d_s = 0.3m$ and this can be viewed as a reaction delay. Figure B.2 shows the resulting trajectories with and without consideration of the priority. If there is no priority, both agents alternate their trajectories around the point of intersection. On the other hand, if the passing priority is given to Agent 2, there are no aggressive maneuvers nor alternations of the path for this agent. All collision avoidance efforts are taken by Agent 1, that prioritizes maximizing the distance between the two agents when collision risk arises. Thus, Agent 1 ensures that the safety distance is respected along the trajectory.

Figures B.3 and B.4 illustrate the change in control inputs when implementing passing priority, for Agent 1 and Agent 2 respectively. By removing the collision avoidance term for Agent 2 with passing priority, its control inputs become smooth and its real trajectory does not deviate from the initially planned path. Since Agent 1 always considers collision avoidance, its control inputs still have higher variation in comparison to Agent 2. However, considering the priority helps in smoothing the control inputs of Agent 1. It could be further improved by tuning the weights for the input cost in the cost function.

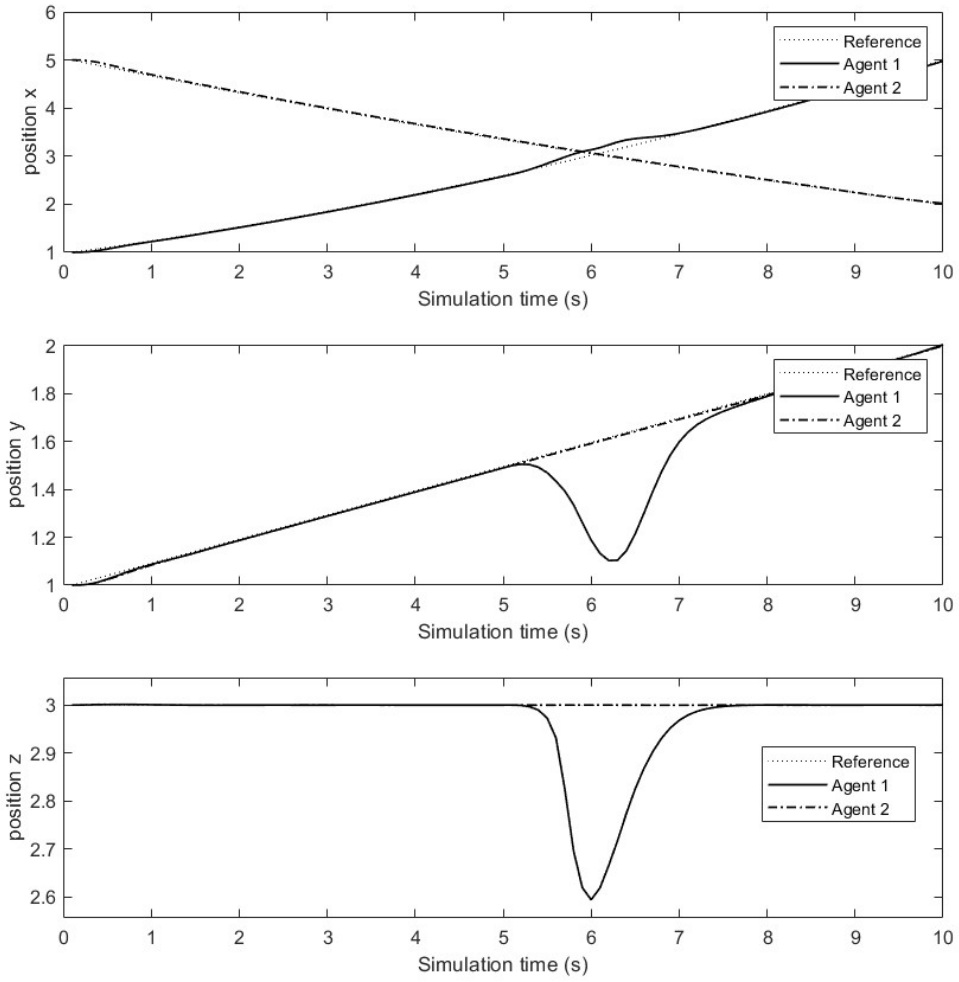


Figure B.1: Reference and resulting trajectories.

Table B.2: Final tracking errors and minimum distance - nominal case

$\|\epsilon_1(T_f)\|$	$\|\epsilon_2(T_f)\|$	d_{min}
$0.01m$	$0.01m$	$0.298m$

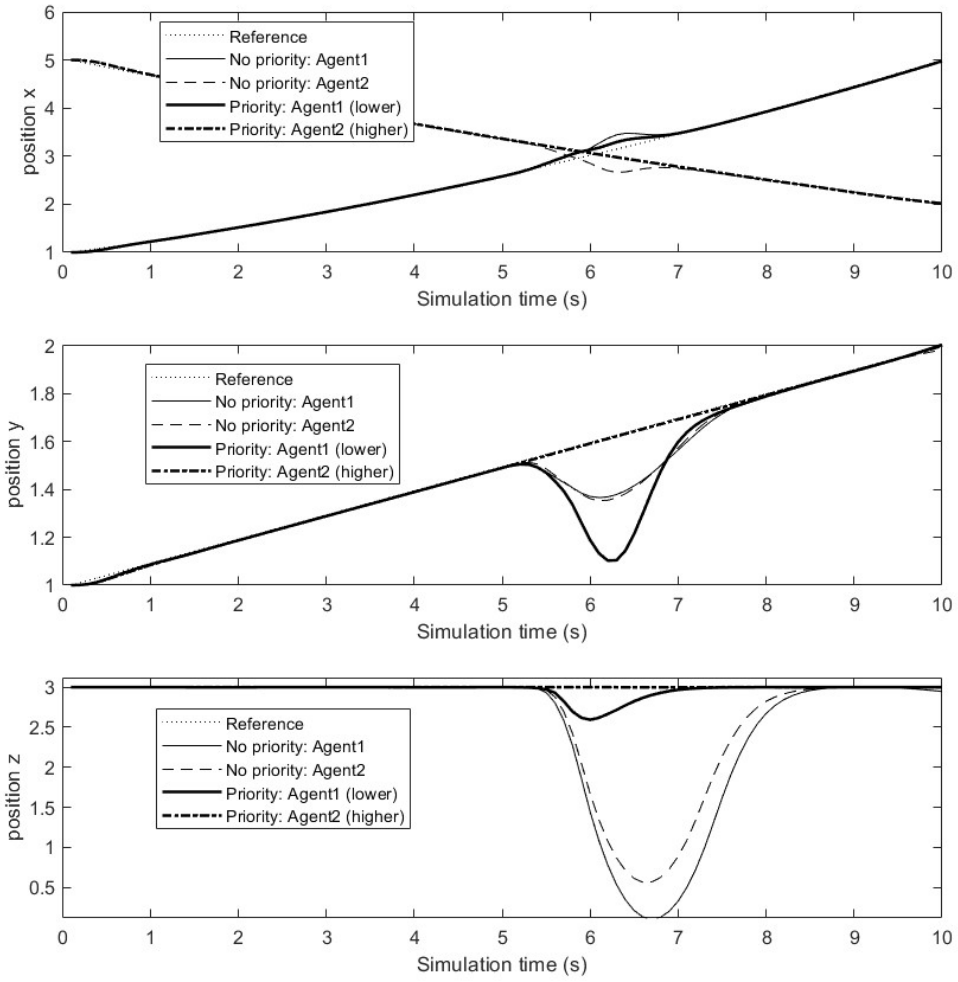


Figure B.2: Outputs for Agent 1 and Agent 2.

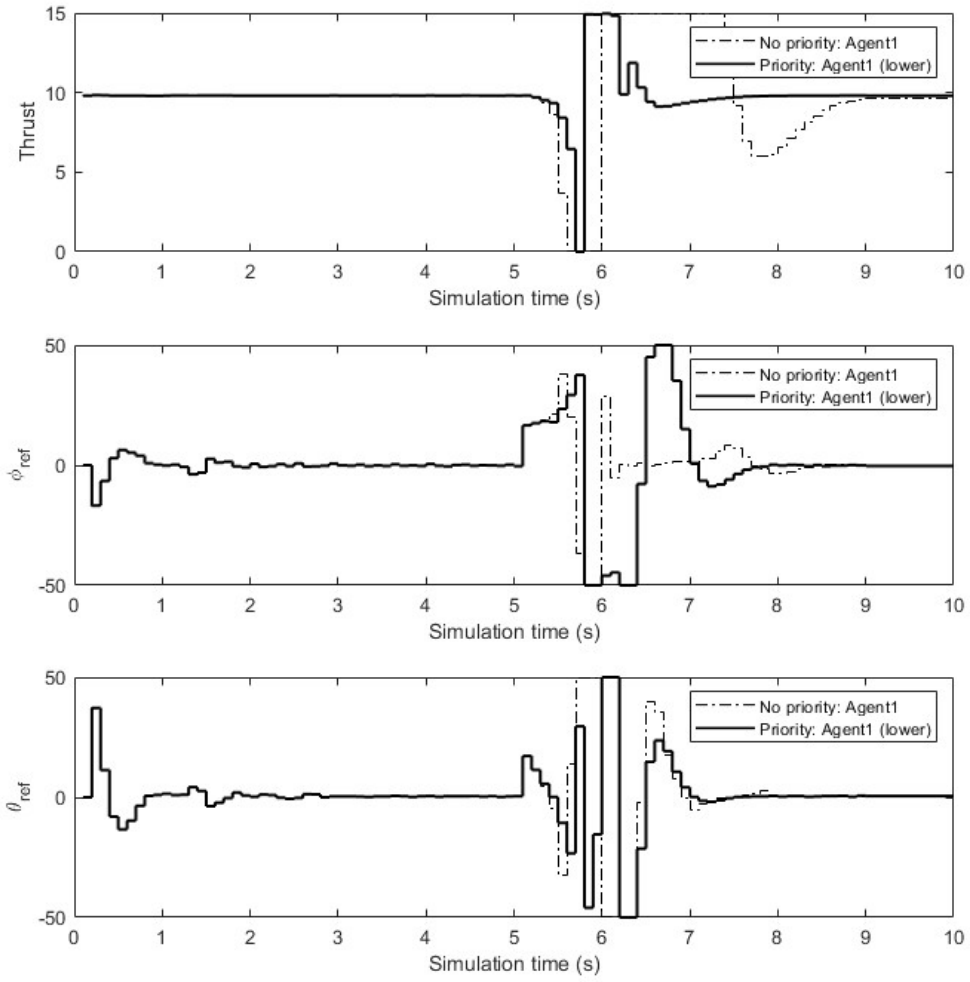


Figure B.3: Control inputs for Agent 1.

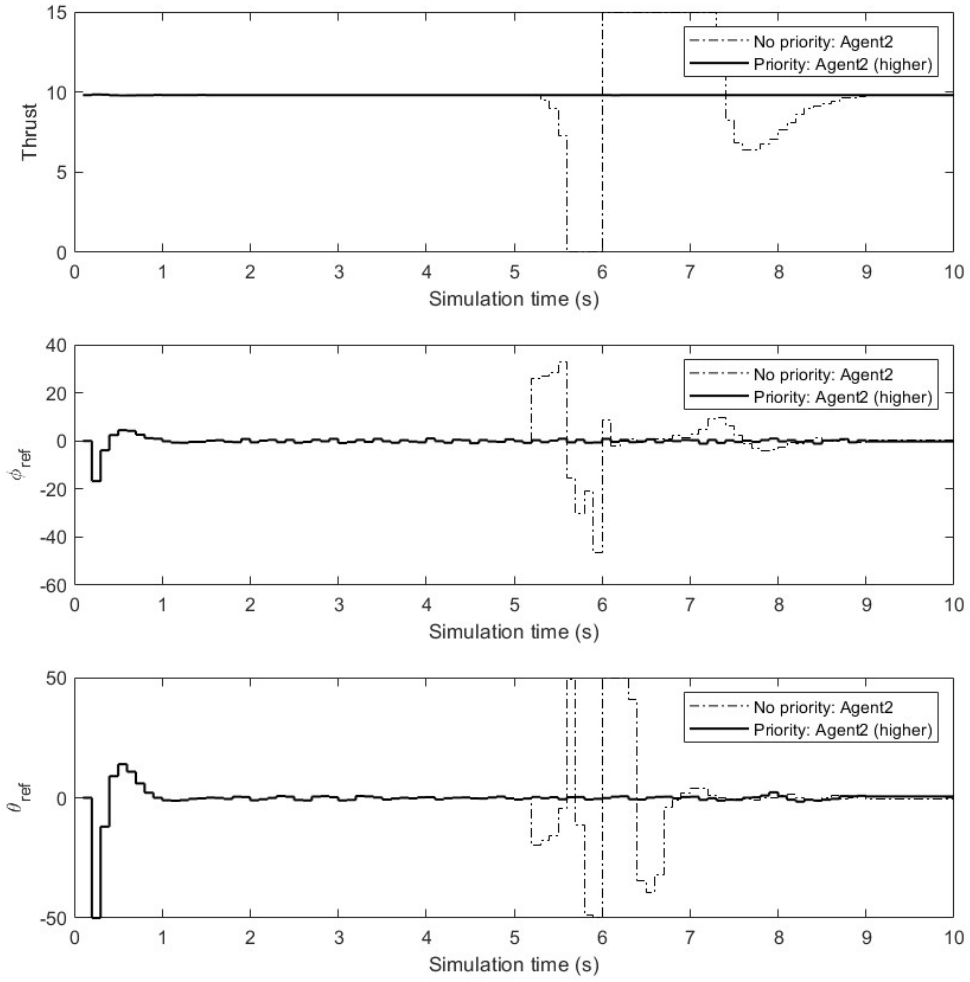


Figure B.4: Control inputs for Agent 2.

Appendix C

Comparative study of NMPC strategies for prioritized trajectory tracking with collision avoidance in mapping missions

C.1 Multi-UAV mapping mission

A simplified layout of the reference paths for agricultural mapping with two UAVs is given in Figure C.1 and the coordinates of the waypoints in Table C.1.

Table C.1: Reference waypoints

Waypoint	UAV1	UAV2
$start$	$(0, 0, 0)m$	$(0.5, 0.5, 0)m$
1	$(1, 0, 3)m$	$(1, 3, 3)m$
2	$(2, 0, 3)m$	$(2, 3, 3)m$
3	$(3, 0, 3)m$	$(3, 3, 3)m$
4	$(3, 1, 3)m$	$(3, 2, 3)m$
5	$(2, 1, 3)m$	$(2, 2, 3)m$
6	$(1, 1, 3)m$	$(1, 2, 3)m$
$finish$	$(0, 0, 0)m$	$(0, 0.5, 0)m$

As this is a homogeneous multi-UAV mission, both UAVs have the same camera parameters and

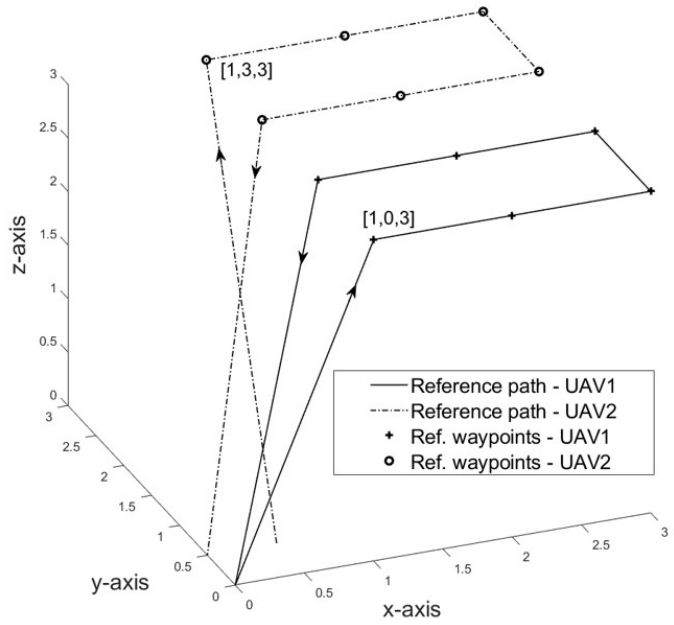


Figure C.1: Reference path for the mapping mission with two UAVs.

therefore, fly at identical and constant reference altitude. Even though the planned paths do not intersect and the distance between agents' waypoints is at least 1 m at all times, a risk of possible collision needs to be considered as it is present at take-off and landing from and to the points in close proximity, or in case of deviation from the planned path. Deviation can be caused by numerous reasons, such as external disturbances, model uncertainties, emergency landing due to fault in the system, etc.

To avoid unnecessary maneuvers and path alternations, only one of the two UAVs will handle collision avoidance. Hierarchical passing priority is allocated to a pair of the UAVs, such that the UAV with higher passing priority focuses only on the trajectory tracking in its control strategy, whereas the UAV with lower passing priority needs to avoid collision in addition to the trajectory tracking. Resulting in smoothened trajectories of both UAVs, fewer alternations account for augmented safety and less energy consumed during the flight. At the planning level, higher priority should be given to the UAV with a lower battery level, smaller overall path distance, or based on another determined criterion. In this study (Figure C.1), priority is given to UAV2, while UAV1 is responsible for collision avoidance when the risk appears along the trajectory.

Control is handled in a distributed manner, with output information exchanged between the UAVs. Model uncertainties and external disturbances are also considered when describing the real system.

C.2 Distributed NMPC strategies for collision avoidance

Collision avoidance can be addressed in several ways. This study aims to compare collision avoidance handling as a nonlinear constraint, an additional criterion in the cost function and a safe flight corridor.

All the listed strategies for collision avoidance will be applied only to the UAV with the lower-passing priority, whereas classical NMPC will be applied to the higher-passing priority UAV. Cost function of the classical NMPC minimizes the tracking error and change in control inputs:

$$J_i(u^i_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i} + \|\Delta u^i_{k+n-1}\|^2_{R_i} \right], \tag{C.1}$$

where Q_i and R_i are weight matrices, and $y^{i,ref}$ the reference output for UAV i. The first term considers trajectory tracking, while the second term aims to smooth the variation in control inputs.

C.2.1 Collision avoidance as a nonlinear constraint

A nonlinear constraint that prevents collision between the UAVs is imposed as a minimum distance that needs to be satisfied between the positions of the UAVs along their trajectories. Besides the classic NMPC (C.1), a nonlinear constraint for collision avoidance needs to be satisfied, where the distance between the UAVs i and j, d_{ij}, is kept greater than the defined safety distance d_s over the prediction horizon. Thus, at each sampling time $(k+n)T_e$, following constraint is considered:

$$\|d_{ij,k+n}\|_2 \geq d_s, \quad i,j \in \mathcal{N}_i, \quad j \neq i \tag{C.2}$$

C.2.2 Collision avoidance in the cost function

Relaxing the collision avoidance constraint can be transformed into a penalty cost. Thus, the newly formulated cost function includes a new criterion whose weight depends on the proximity from the UAV that needs to be avoided. It is expressed as in [113]:

$$J_i(u^i_{k,\dots,k+N_p-1}) = \sum_{n=1}^{N_p} \left[\|\hat{\mathbf{y}}^i_{k+n} - \mathbf{y}^{i,ref}_{k+n}\|^2_{Q_i} + \|\Delta u^i_{k+n-1}\|^2_{R_i} - \sum_{j\in\mathcal{N}_i, j\neq i} A_{ij}(d_{ij,k+n}) \|d_{ij,k+n}\|^2_{G_{ij}} \right] \tag{C.3}$$

In (C.3), in addition to minimizing the tracking error and successive change in control inputs as in (C.1), the last term in the cost function aims to maximize the distance between the UAVs with the

weight factor G_{ij}. Depending on the distance d_{ij} and tuning factor γ, weight A_{ij} can take values in the interval $[0, 1]$.

As the highest risk of collision may appear during the take-off and landing, the collision avoidance term outweighs the trajectory tracking term in the cost function. On the other hand, minimizing the tracking error is a priority during the rest of the mission. Therefore, A_{ij} is a sigmoid function that helps avoid numerical issues due to the choice of a binary term (switch between 0 and 1 depending on the distance).

C.2.3 Collision avoidance through a flight corridor

As a safety mechanism to reject disturbances and model uncertainties, the flight corridor is introduced as a nonlinear constraint in the optimal control problem to ensure trajectory tracking within allowed limits. A flight corridor is represented as a tube with a set radius from the desired trajectory. Alone, it can ensure collision avoidance as each UAV stays inside its own corridor, which is constructed such that the corridor of the UAV does not intersect with the planned trajectory of the other UAVs in the system.

In this case, the control input is chosen to minimize the classical NMPC as in (C.1) while respecting additional constraints to remain in the defined boxed corridor:

$$| p_\bullet - p_\bullet^{ref} | \leq \beta, \tag{C.4}$$

where $p_\bullet$ are the actual positions p_x, p_y or p_z, and $p_\bullet^{ref}$ position references along 3 axes. In order to ensure collision avoidance, the choice of the corridor width in 3 axes, β, depends on the size of UAV, as well as the *a priori* planned paths of all the UAVs.

C.3 Simulation results and discussion

C.3.1 Mapping mission trajectory tracking

In order to assess the robustness of the previously presented control strategies for trajectory tracking with collision avoidance in a mapping mission (Figure C.1), the analysis of the Monte Carlo simulations will be presented here. Random values of constant external disturbances $[d_x, d_y, d_z]$, and uncertainty of the model parameter of the thruster efficiency α are considered for all the test cases. Simulation parameters are given in Table C.2.

Table C.2: Parameter values

g	$9.81m/s^2$	T_e	$0.1s$
A_x, A_y, Az	$(0.1, 0.1, 0.2)s^{-1}$	N_p	10
K_ϕ, K_θ	1	T_f	$10s$
τ_ϕ, τ_θ	$(0.7, 0.5)s$	d_s	$0.55m$
$Q_1 = Q_2$	$diag(10^2, 10, 10^2, 10, 10^3, 10)$	S	10%
$R_1 = R_2$	$diag(1, 1, 1)$	β	$0.3m$
G_{12}	100		

Parameters d_s and β are chosen according to the size of the UAV and the mission path plan. Here, simulations are performed for DJI Mavic 3, with the dimensions 347,5 x 283 x 107,7 mm (with propellers).

Monte Carlo simulations were conducted for 50 test cases with randomly added bounded external disturbances and uncertainties on thruster efficiency, denoted d_1, α_1 and d_2, α_2 for UAV1 and UAV2, respectively. The external disturbance values vary in the range $d_{x,1}, d_{x,2} = [-1, 1]\ m/s^2$, $d_{y,1}, d_{y,2} = [-1, 1]\ m/s^2$, $d_{z,1}, d_{z,2} = [0, 0.4]\ m/s^2$, while for the thruster uncertainty $\alpha_1, \alpha_2 = [0.7, 1.35]$. All the optimization problems were solved by the same algorithm (*fmincon* of Matlab).

It is important to verify whether the minimum safety distance d_s is respected along the trajectories in all the test cases. In order to challenge collision avoidance, the imposed safety distance $d_s = 0.55m$ is superior to the distance between the reference paths of the two UAVs. Figure C.2 shows the minimum reached distance between the UAVs along the mission. Nonlinear constraint and penalty cost strategies successfully avoid entering the imposed collision risk zone, whereas the flight corridor strategy violates the safety distance in certain test cases because the planned reference paths are *a priori* configured in overly close proximity to each other.

Table C.3 shows the root mean square error (RMSE) of the tracking error along the trajectory for the UAV that handles collision avoidance. The best results in terms of the tracking error, both the error value and the uniformity of the results, are exhibited for the flight corridor. As the flight corridor restricts the deviations from the reference path, it ensures that the UAV remains within the imposed limits, even in the presence of high model uncertainty and external disturbances. Collision avoidance in the cost function criterion shows the highest average tracking error, as it outweighs the trajectory tracking term at take-off and landing, when the UAVs are in high-risk collision zone.

When selecting the adequate control strategy, computational complexity needs to be considered. CPU time needed to solve the optimal control problem is an indicator of the complexity. Based on the results shown in Table C.4, collision avoidance as the nonlinear constraint indicates the highest

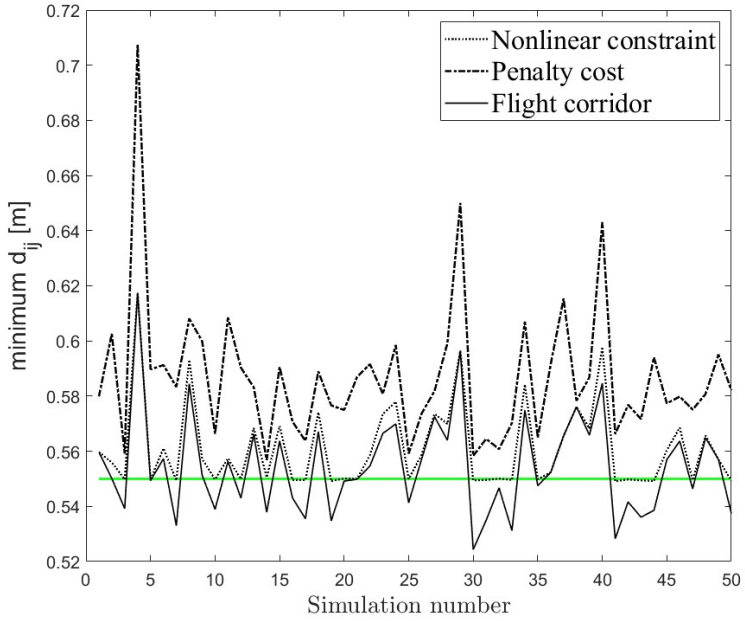


Figure C.2: Minimum distance between the UAVs along the trajectory. $d_s = 0.55m$ (in green).

Table C.3: RMSE for each strategy based on Monte Carlo simulation results

RMSE [m]	min	max	mean
Nonlinear constraint	0.08	0.85	0.15
Penalty cost	0.30	0.97	0.51
Flight corridor	0.08	0.23	0.13

computational complexity among all strategies. It should be noted that resulting CPU time is given only as an indication for comparison, and as such is not compatible with real-time application.

Table C.4: Average CPU time for each strategy based on Monte Carlo simulation results

Average CPU time [s]	min	max	mean
Nonlinear constraint	0.67	1.58	0.89
Penalty cost	0.54	1.06	0.78
Flight corridor	0.29	0.65	0.46

When comparing the two indicators, root mean square tracking error and computational complexity, flight corridor seems to be the highest-performing collision avoidance strategy for the presented path configuration.

Bibliography

[1] Y.-S. Jiao, X.-M. Wang, H. Chen, and Y. Li, "Research on the coverage path planning of UAVs for polygon areas," in *2010 5th IEEE Conference on Industrial Electronics and Applications*. IEEE, 2010, pp. 1467–1472.

[2] Y. Li, H. Chen, M. J. Er, and X. Wang, "Coverage path planning for UAVs based on enhanced exact cellular decomposition method," *Mechatronics*, vol. 21, no. 5, pp. 876–885, 2011.

[3] X. Yang, G. Liu, A. Li, and V. D. Le, "A predictive power control strategy for DFIGs based on a wind energy converter system," *Energies*, vol. 10, no. 8, p. 1098, 2017.

[4] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, "A compilation of UAV applications for precision agriculture," *Computer Networks*, vol. 172, p. 107148, 2020.

[5] 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. 107017, 2022.

[6] D. L. Corwin, "Site-specific management and delineating management zones," in *Precision agriculture for sustainability and environmental protection*. Routledge, 2013, pp. 135–157.

[7] J. Del Cerro, C. Cruz Ulloa, A. Barrientos, and J. de León Rivas, "Unmanned aerial vehicles in agriculture: A survey," *Agronomy*, vol. 11, no. 2, p. 203, 2021.

[8] J. Kim, S. Kim, C. Ju, and H. I. Son, "Unmanned aerial vehicles in agriculture: A review of perspective of platform, control, and applications," *IEEE Access*, vol. 7, pp. 105 100–105 115, 2019.

[9] K. Kumar and N. Kumar, "Region coverage-aware path planning for unmanned aerial vehicles: A systematic review," *Physical Communication*, p. 102073, 2023.

[10] H. Choset, "Coverage for robotics–a survey of recent results," *Annals of mathematics and artificial intelligence*, vol. 31, pp. 113–126, 2001.

[11] T. M. Cabreira, L. B. Brisolara, and F. J. Paulo R, “Survey on coverage path planning with unmanned aerial vehicles,” *Drones*, vol. 3, no. 1, p. 4, 2019.

[12] J. Li, H. Sheng, J. Zhang, and H. Zhang, “Coverage path planning method for agricultural spraying UAV in arbitrary polygon area,” *Aerospace*, vol. 10, no. 9, p. 755, 2023.

[13] M. Torres, D. A. Pelta, J. L. Verdegay, and J. C. Torres, “Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction,” *Expert Systems with Applications*, vol. 55, pp. 441–451, 2016.

[14] M. Coombes, W.-H. Chen, and C. Liu, “Boustrophedon coverage path planning for UAV aerial surveys in wind,” in *2017 International conference on unmanned aircraft systems (ICUAS)*. IEEE, 2017, pp. 1563–1571.

[15] T. Li, C. Wang, C. W. de Silva *et al.*, “Coverage sampling planner for UAV-enabled environmental exploration and field mapping,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2509–2516.

[16] A. Ghaddar, A. Merei, and E. Natalizio, “Pps: Energy-aware grid-based coverage path planning for UAVs using area partitioning in the presence of NFZs,” *Sensors*, vol. 20, no. 13, p. 3742, 2020.

[17] Y. Gong, K. Chen, T. Niu, and Y. Liu, “Grid-based coverage path planning with NFZ avoidance for UAV using parallel self-adaptive ant colony optimization algorithm in cloud IoT,” *Journal of Cloud Computing*, vol. 11, no. 1, p. 29, 2022.

[18] F. Stache, J. Westheider, F. Magistri, M. Popović, and C. Stachniss, “Adaptive path planning for UAV-based multi-resolution semantic segmentation,” in *2021 European Conference on Mobile Robots (ECMR)*. IEEE, 2021, pp. 1–6.

[19] J. M. Peña, J. Torres-Sánchez, A. Serrano-Pérez, A. I. De Castro, and F. López-Granados, “Quantifying efficacy and limits of unmanned aerial vehicle (UAV) technology for weed seedling detection as affected by sensor resolution,” *Sensors*, vol. 15, no. 3, pp. 5609–5626, 2015.

[20] T. M. Cabreira, C. Di Franco, P. R. Ferreira, and G. C. Buttazzo, “Energy-aware spiral coverage path planning for UAV photogrammetric applications,” *IEEE Robotics and automation letters*, vol. 3, no. 4, pp. 3662–3668, 2018.

[21] D. Datsko, F. Nekovar, R. Penicka, and M. Saska, “Energy-aware multi-UAV coverage mission planning with optimal speed of flight,” *IEEE Robotics and Automation Letters*, 2024.

[22] T. M. Cabreira, P. R. Ferreira, C. Di Franco, and G. C. Buttazzo, "Grid-based coverage path planning with minimum energy over irregular-shaped areas with UAVs," in *2019 international conference on unmanned aircraft systems (ICUAS)*. IEEE, 2019, pp. 758–767.

[23] S. Liu, X. Li, M. Meng, and X. Gong, "Energy-aware coverage path planning of multi-UAV based on relative distance scaling cluster method," in *2023 International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE, 2023, pp. 709–714.

[24] R. I. Mukhamediev, K. Yakunin, M. Aubakirov, I. Assanov, Y. Kuchin, A. Symagulov, V. Levashenko, E. Zaitseva, D. Sokolov, and Y. Amirgaliyev, "Coverage path planning optimization of heterogeneous UAVs group for precision agriculture," *IEEE Access*, vol. 11, pp. 5789–5803, 2023.

[25] R. Schacht-Rodríguez, J.-C. Ponsart, C. D. Garcia-Beltran, C. M. Astorga-Zaragoza, and D. Theilliol, "Mission planning strategy for multirotor UAV based on flight endurance estimation," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2019, pp. 778–786.

[26] M. Wei and V. Isler, "Coverage path planning under the energy constraint," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 368–373.

[27] A. Thibbotuwawa, G. Bocewicz, P. Nielsen, and B. Zbigniew, "Planning deliveries with UAV routing under weather forecast and energy consumption constraints," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 820–825, 2019.

[28] G. S. Avellar, G. A. Pereira, L. C. Pimenta, and P. Iscold, "Multi-UAV routing for area coverage and remote sensing with minimum time," *Sensors*, vol. 15, no. 11, pp. 27 783–27 803, 2015.

[29] M. A. Luna, M. S. Ale Isaac, A. R. Ragab, P. Campoy, P. Flores Peña, and M. Molina, "Fast multi-UAV path planning for optimal area coverage in aerial sensing applications," *Sensors*, vol. 22, no. 6, p. 2297, 2022.

[30] H. Fesenko, I. Kliushnikov, V. Kharchenko, S. Rudakov, and E. Odarushchenko, "Routing an unmanned aerial vehicle during npp monitoring in the presence of an automatic battery replacement aerial system," in *2020 IEEE 11th international conference on dependable systems, services and technologies (DESSERT)*. IEEE, 2020, pp. 34–39.

[31] R. Santin, L. Assis, A. Vivas, and L. C. Pimenta, "Matheuristics for multi-UAV routing and recharge station location for complete area coverage," *Sensors*, vol. 21, no. 5, p. 1705, 2021.

[32] K. Yu, A. K. Budhiraja, and P. Tokekar, “Algorithms for routing of unmanned aerial vehicles with mobile recharging stations,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5720–5725.

[33] M. A. Kamel, A. T. Hafez, and X. Yu, “A review on motion control of unmanned ground and aerial vehicles based on model predictive control techniques,” *Journal of Engineering Science and Military Technologies*, vol. 2, no. 1, pp. 10–23, 2018.

[34] B. Rubí, R. Pérez, and B. Morcego, “A survey of path following control strategies for UAVs focused on quadrotors,” *Journal of Intelligent & Robotic Systems*, vol. 98, no. 2, pp. 241–265, 2020.

[35] A. P. Aguiar and J. P. Hespanha, “Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty,” *IEEE transactions on automatic control*, vol. 52, no. 8, pp. 1362–1379, 2007.

[36] R. Amin, L. Aijun, and S. Shamshirband, “A review of quadrotor UAV: control methodologies and performance evaluation,” *International Journal of Automation and Control*, vol. 10, no. 2, pp. 87–103, 2016.

[37] M. Maaruf, M. S. Mahmoud, and A. Ma’arif, “A survey of control methods for quadrotor UAV,” *International Journal of Robotics and Control Systems*, vol. 2, no. 4, pp. 652–665, 2022.

[38] H. T. Nguyen, T. V. Quyen, C. V. Nguyen, A. M. Le, H. T. Tran, and M. T. Nguyen, “Control algorithms for UAVs: A comprehensive survey,” *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 7, no. 23, pp. e5–e5, 2020.

[39] C. A. Cárdenas R, V. H. Grisales, C. A. Collazos Morales, H. D. Cerón-Muñoz, P. Ariza-Colpas, and R. Caputo-Llanos, “Quadrotor modeling and a pid control approach,” in *Intelligent Human Computer Interaction: 11th International Conference, IHCI 2019, Allahabad, India, December 12–14, 2019, Proceedings 11*. Springer, 2020, pp. 281–291.

[40] R. Miranda-Colorado and L. T. Aguilar, “Robust PID control of quadrotors with power reduction analysis,” *ISA transactions*, vol. 98, pp. 47–62, 2020.

[41] J. Moreno-Valenzuela, R. Pérez-Alcocer, M. Guerrero-Medina, and A. Dzul, “Nonlinear PID-type controller for quadrotor trajectory tracking,” *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 5, pp. 2436–2447, 2018.

[42] P. Burggräf, A. R. Pérez Martínez, H. Roth, and J. Wagner, "Quadrotors in factory applications: Design and implementation of the quadrotor's P-PID cascade control system: Modeling and implementation," *SN Applied Sciences*, vol. 1, no. 7, p. 722, 2019.

[43] E. Okyere, A. Bousbaine, G. T. Poyi, A. K. Joseph, and J. M. Andrade, "LQR controller design for quad-rotor helicopters," *The Journal of Engineering*, vol. 2019, no. 17, pp. 4003–4007, 2019.

[44] M. R. Cohen, K. Abdulrahim, and J. R. Forbes, "Finite-horizon LQR control of quadrotors on $SE\_2(3)$," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5748–5755, 2020.

[45] A. S. Elkhatem and S. N. Engin, "Robust LQR and LQR-PI control strategies based on adaptive weighting matrix selection for a UAV position and attitude tracking control," *Alexandria Engineering Journal*, vol. 61, no. 8, pp. 6275–6292, 2022.

[46] B. Yu, Y. Zhang, J. Yan, Y. Qu, and Z. Liu, "Fault tolerant control using linear quadratic technique against actuator faults in a UAV," in *Proceedings of the 32nd Chinese Control Conference*. IEEE, 2013, pp. 6294–6299.

[47] J. Qiao, Z. Liu, and Y. Zhang, "Gain scheduling based PID control approaches for path tracking and fault tolerant control of a quad-rotor UAV," *International Journal of Mechanical Engineering and Robotics Research*, vol. 7, no. 4, pp. 401–408, 2018.

[48] A. González, A. Cuenca, V. Balaguer, and P. García, "Event-triggered predictor-based control with gain-scheduling and extended state observer for networked control systems," *Information Sciences*, vol. 491, pp. 90–108, 2019.

[49] J. Gadewadikar, F. Lewis, K. Subbarao, and B. M. Chen, "Structured H-infinity command and control-loop design for unmanned helicopters," *Journal of guidance, control, and dynamics*, vol. 31, no. 4, pp. 1093–1102, 2008.

[50] M. Chen and M. Huzmezan, "A combined MBPC/2 DOF H infinity controller for a quad rotor UAV," in *AIAA guidance, navigation, and control conference and exhibit*, 2003, p. 5520.

[51] S. B. da Cunha, "On the robustness of feedback linearization," *International Journal of Dynamics and Control*, pp. 1–14, 2024.

[52] C.-C. Chen and Y.-T. Chen, "Feedback linearized optimal control design for quadrotor with multi-performances," *IEEE Access*, vol. 9, pp. 26 674–26 695, 2021.

[53] A. Mokhtari, A. Benallegue, and B. Daachi, "Robust feedback linearization and GH/sub/spl infin//controller for a quadrotor unmanned aerial vehicle," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 1198–1203.

[54] N. Koksal, H. An, and B. Fidan, "Backstepping-based adaptive control of a quadrotor UAV with guaranteed tracking performance," *ISA transactions*, vol. 105, pp. 98–110, 2020.

[55] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, "Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor UAV," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 8, pp. 5044–5056, 2016.

[56] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, "Second order sliding mode control for a quadrotor UAV," *ISA transactions*, vol. 53, no. 4, pp. 1350–1356, 2014.

[57] L. Besnard, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," *Journal of the Franklin Institute*, vol. 349, no. 2, pp. 658–684, 2012.

[58] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor UAVs: A design trade study with flight evaluations," *IEEE Transactions on control systems technology*, vol. 21, no. 4, pp. 1400–1406, 2012.

[59] T.-T. Tran, S. S. Ge, and W. He, "Adaptive control of a quadrotor aerial vehicle with input constraints and uncertain parameters," *International Journal of Control*, vol. 91, no. 5, pp. 1140–1160, 2018.

[60] I. Palunko and R. Fierro, "Adaptive control of a quadrotor with dynamic changes in the center of gravity," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 2626–2631, 2011.

[61] B. Li, C. Song, S. Bai, J. Huang, R. Ma, K. Wan, and E. Neretin, "Multi-UAV trajectory planning during cooperative tracking based on a fusion algorithm integrating MPC and standoff," *Drones*, vol. 7, no. 3, p. 196, 2023.

[62] B. C. Camacho, E. F., *Model predictive control*. Springer London, 2004.

[63] J. B. Rawlings, D. Q. Mayne, M. Diehl *et al.*, *Model predictive control: theory, computation, and design*. Nob Hill Publishing Madison, WI, 2017, vol. 2.

[64] A. Ortiz, S. Garcia-Nieto, and R. Simarro, “Comparative study of optimal multivariable LQR and MPC controllers for unmanned combat air systems in trajectory tracking,” *Electronics*, vol. 10, no. 3, p. 331, 2021.

[65] K. Yang, X. Tang, Y. Qin, Y. Huang, H. Wang, and H. Pu, “Comparative study of trajectory tracking control for automated vehicles via model predictive control and robust H-infinity state feedback control,” *Chinese Journal of Mechanical Engineering*, vol. 34, pp. 1–14, 2021.

[66] E. Kayacan, W. Saeys, H. Ramon, C. Belta, and J. M. Peschel, “Experimental validation of linear and nonlinear MPC on an articulated unmanned ground vehicle,” *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 5, pp. 2023–2030, 2018.

[67] M. Kamel, M. Burri, and R. Siegwart, “Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3463–3469, 2017.

[68] M. Greeff and A. P. Schoellig, “Flatness-based model predictive control for quadrotor trajectory tracking,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 6740–6745.

[69] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes, “Robust model predictive flight control of unmanned rotorcrafts,” *Journal of Intelligent & Robotic Systems*, vol. 81, pp. 443–469, 2016.

[70] K. Zhang, Y. Shi, and H. Sheng, “Robust nonlinear model predictive control based visual servoing of quadrotor UAVs,” *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 2, pp. 700–708, 2021.

[71] A. Didier, A. Parsi, J. Coulson, and R. S. Smith, “Robust adaptive model predictive control of quadrotors,” in *2021 European Control Conference (ECC)*. IEEE, 2021, pp. 657–662.

[72] D. Abeywardena, L. Amaratunga, S. Shakoor, and S. Munasinghe, “A velocity feedback fuzzy logic controller for stable hovering of a quad rotor UAV,” in *2009 International Conference on Industrial and Information Systems (ICIIS)*. IEEE, 2009, pp. 558–562.

[73] J. Muliadi and B. Kusumoputro, “Neural network control system of UAV altitude dynamics and its comparison with the PID control system,” *Journal of Advanced Transportation*, vol. 2018, pp. 1–18, 2018.

[74] A. Barrientos, J. Colorado, J. d. Cerro, A. Martinez, C. Rossi, D. Sanz, and J. Valente, “Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots,” *Journal of Field Robotics*, vol. 28, no. 5, pp. 667–689, 2011.

[75] C. Ju, J. Kim, J. Seol, and H. I. Son, "A review on multirobot systems in agriculture," *Computers and Electronics in Agriculture*, vol. 202, p. 107336, 2022.

[76] J. Li and W. Ren, "A multi-machine cooperation full coverage path planning method in agriculture," in *2023 4th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*. IEEE, 2023, pp. 713–716.

[77] I. Maza and A. Ollero, "Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms," in *Distributed Autonomous Robotic Systems 6*. Springer, 2007, pp. 221–230.

[78] S. W. Cho, H. J. Park, H. Lee, D. H. Shim, and S.-Y. Kim, "Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations," *Computers & Industrial Engineering*, vol. 161, p. 107612, 2021.

[79] Z. Cheng, L. Zhao, and Z. Shi, "Decentralized multi-UAV path planning based on two-layer coordinative framework for formation rendezvous," *IEEE Access*, vol. 10, pp. 45 695–45 708, 2022.

[80] S. Huang, R. S. H. Teo, and K. K. Tan, "Collision avoidance of multi unmanned aerial vehicles: A review," *Annual Reviews in Control*, vol. 48, pp. 147–164, 2019.

[81] T. Lozano-Pérez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Communications of the ACM*, vol. 22, no. 10, pp. 560–570, 1979.

[82] J. Tang, S. Lao, and Y. Wan, "Systematic review of collision-avoidance approaches for unmanned aerial vehicles," *IEEE Systems Journal*, vol. 16, no. 3, pp. 4356–4367, 2021.

[83] L. Pallottino, V. G. Scordio, A. Bicchi, and E. Frazzoli, "Decentralized cooperative policy for conflict resolution in multivehicle systems," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1170–1183, 2007.

[84] Z. Chao, L. Ming, Z. Shaolei, and Z. Wenguang, "Collision-free UAV formation flight control based on nonlinear MPC," in *2011 international conference on electronics, communications and control (ICECC)*. IEEE, 2011, pp. 1951–1956.

[85] Y. Kuriki and T. Namerikawa, "Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control," *SICE Journal of Control, Measurement, and System Integration*, vol. 8, no. 4, pp. 285–294, 2015.

[86] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.

[87] J. Sun, J. Tang, and S. Lao, “Collision avoidance for cooperative UAVs with optimized artificial potential field algorithm,” *IEEE Access*, vol. 5, pp. 18 382–18 390, 2017.

[88] Y. Matsuno, T. Tsuchiya, J. Wei, I. Hwang, and N. Matayoshi, “Stochastic optimal control for aircraft conflict resolution under wind uncertainty,” *Aerospace Science and Technology*, vol. 43, pp. 77–88, 2015.

[89] A. Mujumdar and R. Padhi, “Reactive collision avoidance of using nonlinear geometric and differential geometric guidance,” *Journal of guidance, control, and dynamics*, vol. 34, no. 1, pp. 303–311, 2011.

[90] J. Seo, Y. Kim, and A. Tsourdos, “Differential geometry based collision avoidance guidance for multiple UAVs,” *IFAC proceedings volumes*, vol. 46, no. 19, pp. 113–118, 2013.

[91] C. Y. Tan, S. Huang, K. K. Tan, and R. S. H. Teo, “Three dimensional collision avoidance for multi unmanned aerial vehicles using velocity obstacle,” *Journal of Intelligent & Robotic Systems*, vol. 97, no. 1, pp. 227–248, 2020.

[92] M. F. F. Rahman, S. Fan, Y. Zhang, and L. Chen, “A comparative study on application of unmanned aerial vehicle systems in agriculture,” *Agriculture*, vol. 11, no. 1, p. 22, 2021.

[93] F. Sabatino, “Quadrotor control: modeling, nonlinearcontrol design, and simulation,” 2015.

[94] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Design and control of an indoor micro quadrotor,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA’04. 2004*, vol. 5. IEEE, 2004, pp. 4393–4398.

[95] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, “Quadrotor helicopter flight dynamics and control: Theory and experiment,” in *AIAA guidance, navigation and control conference and exhibit*, 2007, p. 6461.

[96] M. Idrissi, M. Salami, and F. Annaz, “A review of quadrotor unmanned aerial vehicles: applications, architectural design and control algorithms,” *Journal of Intelligent & Robotic Systems*, vol. 104, no. 2, p. 22, 2022.

[97] B. Saha, E. Koshimoto, C. C. Quach, E. F. Hogge, T. H. Strom, B. L. Hill, S. L. Vazquez, and K. Goebel, "Battery health management system for electric UAVs," in *2011 aerospace conference*. IEEE, 2011, pp. 1–9.

[98] R. Schacht-Rodriguez, G. Ortiz-Torres, C. Garcia-Beltran, C. Astorga-Zaragoza, J.-C. Ponsart, and D. Theilliol, "Soc estimation using an extended kalman filter for UAV applications," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 179–187.

[99] J. K. Stolaroff, C. Samaras, E. R. O'Neill, A. Lubers, A. S. Mitchell, and D. Ceperley, "Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery," *Nature communications*, vol. 9, no. 1, p. 409, 2018.

[100] Z. Liu, R. Sengupta, and A. Kurzhanskiy, "A power consumption model for multi-rotor small unmanned aircraft systems," in *2017 international conference on unmanned aircraft systems (ICUAS)*. IEEE, 2017, pp. 310–315.

[101] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart, "Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system," *Robot Operating System (ROS) The Complete Reference (Volume 2)*, pp. 3–39, 2017.

[102] H. G. Santos and T. Toffolo, "Mixed integer linear programming with python," *Accessed: Apr*, 2020.

[103] S. De and D. Guida, "Control design for an under-actuated UAV model," *FME Transactions*, vol. 46, no. 4, pp. 443–452, 2018.

[104] S. Sun, A. Romero, P. Foehn, E. Kaufmann, and D. Scaramuzza, "A comparative study of nonlinear MPC and differential-flatness-based control for quadrotor agile flight," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3357–3373, 2022.

[105] R. Findeisen, F. Allgöwer, and L. T. Biegler, *Assessment and future directions of nonlinear model predictive control*. Springer, 2007, vol. 358, no. 7.

[106] B. S. Guevara, L. F. Recalde, J. Varela-Aldás, V. H. Andaluz, D. C. Gandolfo, and J. M. Toibero, "A comparative study between nmpc and baseline feedback controllers for UAV trajectory tracking," *Drones*, vol. 7, no. 2, p. 144, 2023.

[107] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

[108] F. Allgöwer and A. Zheng, *Nonlinear model predictive control*. Birkhäuser, 2012, vol. 26.

[109] B. Kouvaritakis and M. Cannon, "Model predictive control," *Switzerland: Springer International Publishing*, vol. 38, pp. 13–56, 2016.

[110] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[111] H. Moumouh, N. Langlois, and M. Haddad, "A novel tuning approach for MPC parameters based on artificial neural network," in *2019 IEEE 15th International Conference on Control and Automation (ICCA)*. IEEE, 2019, pp. 1638–1643.

[112] M. Mehndiratta, E. Camci, and E. Kayacan, "Automated tuning of nonlinear model predictive controller by reinforcement learning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3016–3021.

[113] D. Novak and S. Tebbani, "Nonlinear MPC for the multi-UAV system with allocated priority for collision avoidance," in *2023 31st Mediterranean Conference on Control and Automation (MED)*. IEEE, 2023, pp. 7–12.

[114] https://enterprise.dji.com/matrice 350-rtk/specs, "DJI Matrice 350 RTK."

[115] L. Qian and H. H. Liu, "Path-following control of a quadrotor UAV with a cable-suspended payload under wind disturbances," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 3, pp. 2021–2029, 2019.

[116] L. Zhu, Y. Wang, and Z. Wu, "An adaptive priority allocation for formation UAVs in complex context," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 2, pp. 1002–1015, 2020.

[117] H. Teng, I. Ahmad, A. Msm, and K. Chang, "3D optimal surveillance trajectory planning for multiple UAVs by using particle swarm optimization with surveillance area priority," *IEEE Access*, vol. 8, pp. 86 316–86 327, 2020.

[118] X. Liu, Y. Shi, and D. Constantinescu, "Distributed model predictive control of constrained weakly coupled nonlinear systems," *Systems & Control Letters*, vol. 74, pp. 41–49, 2014.

[119] A. Richards and J. P. How, “Robust distributed model predictive control,” *International Journal of control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[120] D. H. Shim, H. J. Kim, and S. Sastry, “Decentralized nonlinear model predictive control of multiple flying robots,” in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, vol. 4. IEEE, 2003, pp. 3621–3626.

[121] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, “Distributed model predictive control,” *IEEE control systems magazine*, vol. 22, no. 1, pp. 44–52, 2002.

[122] F. Mohseni, E. Frisk, and L. Nielsen, “Distributed cooperative MPC for autonomous driving in different traffic scenarios,” *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 2, pp. 299–309, 2020.

[123] Z. Niu, X. Jia, and W. Yao, “Communication-free MPC-based neighbors trajectory prediction for distributed multi-UAV motion planning,” *IEEE Access*, vol. 10, pp. 13 481–13 489, 2022.

[124] 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.

[125] B. Li, C. Song, S. Bai, J. Huang, R. Ma, K. Wan, and E. Neretin, “Multi-UAV trajectory planning during cooperative tracking based on a fusion algorithm integrating MPC and standoff,” *Drones*, vol. 7, no. 3, p. 196, 2023.

[126] A. Belkadi, H. Abaunza, L. Ciarletta, P. Castillo, and D. Theilliol, “Design and implementation of distributed path planning algorithm for a fleet of UAVs,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 6, pp. 2647–2657, 2019.

[127] D. Ioan, S. Olaru, I. Prodan, F. Stoican, and S.-I. Niculescu, “From obstacle-based space partitioning to corridors and path planning. a convex lifting approach,” *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 79–84, 2019.

[128] G. Rousseau, C. S. Maniu, S. Tebbani, M. Babel, and N. Martin, “Minimum-time B-spline trajectories with corridor constraints. application to cinematographic quadrotor flight plans,” *Control Engineering Practice*, vol. 89, pp. 190–203, 2019.

[129] A. Bemporad and D. M. de la Peña, “Multiobjective model predictive control,” *Automatica*, vol. 45, no. 12, pp. 2823–2830, 2009.

[130] H. Zhou, J. Luo, and Z. Yu, “Co-estimation of SOC and SOH for Li-ion battery based on MIEKPF-EKPF fusion algorithm,” *Energy Reports*, vol. 10, pp. 4420–4428, 2023.

Titre: Optimisation de trajectoire de drone(s) pour des missions de cartographie

Mots clés: commande prédictive, planification de trajectoire, drones, optimisation, évitement de collision, agriculture de précision

Résumé: L'utilisation des drones dans le contexte de l'agriculture de précision peut optimiser la gestion de l'exploitation et augmenter la productivité agricole tout en protégeant l'environnement. Toutefois, les drones présentent certaines limites qui doivent être prises en compte lors du développement de solutions de sa mission. Le cadre du problème de la cartographie avec un ou plusieurs drones peut être divisé en deux sous-problèmes : la planification de la mission de cartographie et la pilotage du drone. La première étape définit la trajectoire à suivre pour couvrir la zone d'intérêt de manière efficace compte tenu des limites du drone, tandis que la seconde garantit que le suivi de la trajectoire planifiée est effectué avec succès.
Afin d'augmenter l'efficacité temporelle et d'assurer une mission à faible dépense énergétique, une nouvelle approche pour l'optimisation de la gestion de la batterie du drone lors de la planification de la mission de cartographie est proposée dans ce travail. La stratégie développée optimise l'utilisation des batteries disponibles pour la mission de cartographie en minimisant la distance de vol totale et en réduisant le nombre de remplacements de batteries. La suppression des remplacements de batterie inutiles réduit la durée totale de la mission, mais évite également les cycles redondants de recharge de la batterie.
La distribution des points d'intérêt résultant de la planification de la mission représente les sous-chemins de survol d'un drone doté de plusieurs batteries. Afin de suivre la trajectoire planifiée avec une erreur de suivi minimale, une approche de commande prédictive non linéaire pour un suivi de trajectoire robuste est développée. Cette approche est finalement étendue à une mission de cartographie impliquant plusieurs drones coopératifs, où la sécurité de la mission est assurée principalement en tenant compte de l'évitement des collisions.

Title: Drone(s) trajectory optimization for mapping missions

Keywords: predictive control, trajectory planning, UAVs, optimization, collision avoidance, precision agriculture

Abstract: Using Unmanned Aerial Vehicles (UAVs) in the context of Precision Agriculture (PA) can optimize farming management and increase agricultural productivity while protecting the environment. However, UAVs have certain limitations that must be considered when developing solutions.
The problem framework for conducting mapping with a single or multiple UAVs can be divided into two subproblems: mapping mission planning, and UAV control. The former step defines the path for covering the area of interest in an efficient manner considering the UAV limitations, while the latter ensures that trajectory tracking of the planned path is successfully completed.
In order to increase time efficiency and ensure an energy-aware mission, a novel approach for UAV battery management optimization of the mapping mission planning is proposed in this work. The developed strategy optimizes the use of batteries available for the mapping mission by minimizing the total flight distance and reducing the number of battery replacements. Removing unnecessary battery replacements reduces the overall mission time, but also avoids redundant battery recharging cycles.
The resulting waypoint distribution from the mission planning represents the subpaths for a UAV with multiple batteries. In order to follow the planned path with minimal tracking error, a nonlinear predictive control approach for robust trajectory tracking is introduced. This approach is finally extended to a mapping mission involving multiple cooperative UAVs, where mission safety is ensured primarily by considering collision avoidance.