Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Μηχανολογων Μηχανικων

Τομεας Μηχανολογικων Κατασκευων και Αυτοματου Ελεγχου

Προβλεπτικός Ελεγκτής για την οδήγηση τετραπτέρυγου εναέριου οχήματος από πολλαπλά ενδιάμεσα σημεία

Διπλωματικη Εργασια

του

Άρη Κανελλόπουλου

Επιβλέπων: Κωνσταντίνος Ι. Κυριακόπουλος
Καθηγητής Ε.Μ.Π.

Εργαστηριο Αυτοματου Ελεγχου και Ρυθμισεως Μηχανων και Εγκαταστασεων

Αθήνα, Μάρτιος 2017

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Μηχανολογικών Κατασκευών και Αυτομάτου Ελέγχου
Εργαστήριο Αυτομάτου Ελέγχου και Ρυθμίσεως Μηχανών και Εγκαταστάσεων

Προβλεπτικός Ελεγκτής για την οδήγηση τετραπτέρυγου εναέριου οχήματος από πολλαπλά ενδιάμεσα σημεία

Διπλωματικη Εργασια

του

Άρη Κανελλόπουλου

Επιβλέπων: Κωνσταντίνος Κυριακόπουλος
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την ... Μαρτίου 2017.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Κων/νος Κυριακόπουλος	Ευάγγελος Παπαδόπουλος	Γεώργιος Βοσνιάκος
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2017

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Μηχανολογικών Κατασκευών και Αυτομάτου Ελέγχου
Εργαστήριο Αυτομάτου Ελέγχου και Ρυθμίσεως Μηχανών και Εγκαταστάσεων

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον Καθηγητή Κώστα Κυριακόπουλο για την ευκαιρία που μου προσέφερε να εργαστώ κάτω από την επίβλεψη και καθοδήγηση του. Ο χρόνος μου στο Εργαστήριο Αυτομάτου Ελέγχου υπήρξε ο καθοριστικός παράγοντας στην απόφαση μου να συνεχίσω περαιτέρω την ακαδημαϊκή μου πορεία. Βλέποντας το πάθος των μεταδιδακτορικών, διδακτορικών και προπτυχιακών φοιτητών για τη δουλειά μας, θα μου ήταν αδύνατη οποιαδήποτε άλλη απόφαση. Ιδιαιτέρως ευχαριστώ τον Γιώργο Ζωγόπουλο Παπαλιάκο για τα διάφορα mini courses που μου παρήχε ως labmate. Ευχαριστώ όλα τα μέλη του CSL και ειδικά: τον Μπάμπη, τον Νίκο, τον Πάρη. Τον Παναγιώτη Βλαντή για τις συμβουλές του σε όλα τα σημεία της έρευνας μου και τους Κώστα Βροχίδη και Χρήστο Μαυρίδη για την φιλία τους. Σε πιο ακαδημαϊκά πλαίσια, ευχαριστώ τον υποψήφιο διδάκτορα Shahab Heshmati-alamdari για την καθοδήγησή του σε ένα σημαντικότατο κομμάτι της δουλειάς μου.

Ευχαριστώ την οικογένεια μου για την απεριόριστη στήριξή που μου προσέφεραν, τόσο ψυχολογική όσο και οικονομική, καθώς και τους φίλους μου για την υπομονή που έδειξαν τις ατέλειωτες ώρες που με άκουγαν να παραλυρώ για το ‘‘control’’.

Τέλος, ευχαριστώ την Έφη Σκληρού για την υποστήριξη και βοήθειά της.

Περίληψη

Σε αυτή τη διπλωματική εργασία, εξετάζουμε το πρόβλημα της διέλευσης από μια αλληλουχία σημείων στον τρισδιάστατο χώρο για ένα τετραπτέρυγο Εναέριο Μη-Επανδρωμένο Όχημα. Δεδομένων των συντεταγμένων των σημείων, καθώς και της επιθυμητής σειράς διέλευσης, σχεδιάζονται οι τροχιές των μεταβλητών κατάστασης και ελέγχου που θα οδηγήσουν το όχημα σε αυτά.

Το παραπάνω πρόβλημα μοντελοποιήθηκε σαν Πρόβλημα Βελτίστου Ελέγχου στο πεδίο του συνεχούς χρόνου και λύθηκε με χρήση της μεθόδου Pseudospectral Optimal Control- της προσέγγισης των μεταβλητών κατάστασης και ελέγχου από βάση ορθογώνιων πολυωνύμων. Περαιτέρω, για να αντιμετωπίσουμε σφάλματα στη μοντελοποίηση ή εξωτερικές διαταραχές, εφαρμόζουμε ανατροφοδότηση κατάστασης σχεδιάζοντας τον ελεγκτή στο πλαίσιο του Προβλεπτικού Ελέγχου. Ο Προβλεπτικός Ελεγκτής λύνει επαναλαμβανόμενα το πρόβλημα βελτίστου ελέγχου κατά τη λειτουργία του συστήματος, χρησιμοποιώντας μέτρηση του πραγματικού συστήματος ως αρχική κατάσταση. Εφόσον τα επιθυμητά εσωτερικά σημεία δεν είναι σημεία ισορροπίας του συστήματος, τα συνήθη σχήματα Προβλεκτικού Ελέγχου για σταθεροποίηση συστήματος αποδείχθηκαν ανεπαρκή. Εκτενής έρευνα μας οδήγησε στην χρήση Προβλεπτικού Ελέγχου με Μεταβλητό Ορίζοντα, σύμφωνα με τον οποίο ο ορίζοντας πρόβλεψης αποτελεί μεταβλητή της βελτιστοποίσης σε κάθε επανάληψη. Στην δουλειά αυτή, επεκτάθηκαν τα προϋπάρχοντα θεωρήματα ευρωστίας του Προβλεπτικού Ελέγχου με Μεταβλητό Ορίζοντα σε μη-γραμμικά συστήματα.

Παρόλο που η πλειοψηφία της θεωρίας που παρουσιάζεται έχει εφαρμογή σε κάθε είδους εναέρια οχήματα, προσωμοιώσεις και συγκριτικές μελέτες διεξήχθησαν σε μοντέλο τετραπτέρυγου για την επαλήθευση των αλγορίθμων.

Abstract

In this thesis, we study the problem of traversing a series of points in the 3-Dimensional space with a quadrotor Unmanned Aerial Vehicle (UAV). Given the 3D coordinates of the points, as well as the desired sequence, the state and control input trajectories which lead the vehicle through the points are designed.

In that respect, the problem was formulated as an Optimal Control Problem (OCP). This continuous time OCP was solved using the method of Pseudospectral Optimal Control which consists of expanding the state and control variables over a space of orthogonal polynomials. Moreover, in order to account for modeling imperfection and external disturbances, state feedback was accomplished by implementing the controller in a Model Predictive Control (MPC) framework. In MPC schemes the OCP is solved online iteratively using the current state measurement of the system as the initial state. Since the desired interior waypoints are not equilibrium points of the system conventional MPC schemes proved inefficient. Extensive research led us to adopt a Variable-Horizon MPC approach (VH-MPC) in which the prediction horizon is a decision variable of the OCP solved in each time-step. Finally, a number of theorems presented in this work extend the preexisting robustness guarantees of VH-MPC to nonlinear continuous-time systems.

Although the majority of the theory presented in this work could be applied to any kind of aerial vehicle, simulations and comparative studies were conducted in a quadrotor model to illustrate the validity of our algorithms.

Περιεχόμενα

Κατάλογος Σχημάτων

Chapter 1

Introduction

1.1 Unmanned Aerial Vehicles

Unmanned Aerial Vehicles, also known as drones, are robotic platforms able to operate freely in the 3D space. UAVs may be designed to display different levels of autonomy: either by being teleoperated or by being able to perform certain tasks with little to no supervision from a human operator. UAVs vary both in size and capabilities, with larger ones, modeled after fixed-winged aerial vehicles, still in use during military operations and smaller ones, many of which are rotor based, used in everyday life.

Historically designed for military use, UAVs are now found in many civilian applications. Moreover, they constitute the preferred platform to conduct robotics research for a large part of the scientific community. As far as civilian operations are concerned, some of the most important areas are search and rescue [1], long term inspection missions [2], payload delivery [3] and surveillance in environments where human presence is impeded or even when navigation in indoors environments is required.

As mentioned before, the most defining characteristic used to categorize UAVs is the design of their wings. Thus, we introduce two major types of vehicles:

- Fixed Wing Aircrafts: modeled after airplanes (see Fig.1.1)
- Rotary Wing Aircrafts: modeled after helicopters (see Fig.1.2). Defining example of this type of UAV are the multirotors (see Fig.1.3).

Control of a UAV's position and attitude is achieved either by actuation of the flaps (in the case of fixed-wing UAVs) or by variating the rotation speed of some combination of the rotors (in rotary-wing UAVs). As a result, the problem of designing controllers for UAVs is made more difficult due to the system being

underactuated, forced to control all six of their states using less actuators; in the case of quadrotors four.

Figure 1.1: Fixed-wing Aircrafts.

Figure 1.2: Rotary Wing Aircraft modeled after a conventional helicopter.

Figure 1.3: Rotary Wing Aircraft. Example of a hexarotor.

1.2 Trajectory optimization

Future developments will call for robotic platforms that are able to move in complex, often cluttered or hard environments in order to accomplish a plethora of tasks. From mobile robots moving in a factory floor alongside workers to unmanned aircraft flying across countries and spacecrafts cruising toward distant planets in the Solar System, autonomous agents will have to plan a path and follow it. Trajectory planning involves designing a path compatible with the dynamics of the system. For example, paths for wheeled robots should respect their non-holonomic constraints. Aircraft trajectories must be constrained with respect to turning radii and velocities. Physical limitations should be considered, too. Trajectories through obstacles, such as walls, must be avoided.Finally, actuator limits should be taken into account as well.

A variety of methods have been used to solve the problem of trajectory design. A large class is based on geometric considerations. Examples of such algorithms are grid based methods, in which the robot's workspace is decomposed and each grid cell is identified with a certain point. At each grid point, the robot is allowed to move to adjacent grid points as long as the line between them is completely contained within the free space (in the sense of obstacle existence). This way the set of actions is discretized, and search algorithms (like A^*) are used to find a path from the start to the goal [4]. For a review of these methods, the reader is referred to [5]. Another method for trajectory planning is the use of navigation functions. A virtual potential field is applied over the robot's workspace, in which obstacles are assigned a high potential value, whereas the goal configuration a low one. This way, a virtual force is applied to the robot, attractive with respect to the goal and repulsive with respect to the obstacles [6].

However, we are often tasked with designing a trajectory which not only respects the system's constraints, but produces *optimal* results in some sense, either by minimizing or maximizing a quantity of importance. For this purpose, a large class of trajectory planning problems are formulated, and subsequently solved, as optimal control problems. According to these methods, We define a cost functional, a *function of functions* (in a sense of mapping from a function space into a scalar field), and we seek these functions that produce and extremal - minimum or maximum - value for this functional. An advantage of optimal control methods is the ease with which we may define various constraints on the variables. This way it becomes straightforward to model obstacles in the workspace or actuation limits.

The formulation of trajectory design problems as optimal control problems (denoted in the sequel as OCPs) require the following:

- A valid mathematical model of the system: No matter how rigorous we are when modeling the system, we cannot account for stochastic phenomena such as external disturbances and uncertainties in the system parameters.
- A performance measure: To evaluate the performance of the system, we design the aforementioned cost functional.
- The physical constraints: These may include limitations on the control input, obstacles, and even desired configurations we wish the system to achieve.

Trajectory planning via OCP is not without its disadvantages. First of all, solving the OCP is a really demanding process. With that in mind, a number of different methodologies have been developed. These methods are categorized as *indirect* or *direct*. Indirect methods involve deriving the Euler-Lagrange differential equations that describe the evolution of the trajectories in time and then using numerical discretization methods to solve them. On the other hand, direct methods involve discretizing the OCP and then solving numerically the derived static optimization problem.

Also, in most cases the solution of an OCP provides us with an open-loop control policy which fails to take into account external disturbances or discrepancies between the real system and the simplified model we provide. Therefore, we utilize a Model Predictive Control (MPC) scheme to account for this issue.

1.3 Model Predictive Control

Model Predictive Control (MPC) is an optimization based method for the feedback control of dynamical systems. Introduced in the chemical industry in the 1980s, rapid development of computational systems helped transition MPC to a broader class of problems. The difficulty of applying MPC schemes stems from their high computational burden compared to "classic" PID or LQR controllers.

MPC consists of iteratively solving an optimal control problem online, utilizing current updates on the system states as initial conditions. These methods are based on the use of dynamic models describing the plant's process. Generally, MPC is chosen in systems of high complexity, since most simpler systems can be adequately controlled with classic schemes. Therefore, models used in MPC have great complexity, high-frequency dynamics, or even contain discontinuities.

Through these dynamic models, it is possible to predict the system's states at some future time for a given control input. Therefore, in its conventional form, MPC consists of computing at time t_k the control strategy that minimizes a cost functional over a *prediction horizon* $[t_k, t_k+T_p]$ and applying this strategy over a *control horizon* $[t_k, t_k + T_c]$. This way, at each recalculation instant t_i $(i = k, k + 1, \ldots, N)$ the optimal control input and state trajectory is found in order to guide the system to the desired final configuration. Due to external disturbances or uncertainties in the model, discrepancies may appear between the predicted state and the actual state of the system. As a result, each recalculation of the optimal states and control, derives information from the true state of the system and utilizes it as an initial state for the optimization problem. Thus, MPC is considered a closed loop control scheme.

Until recently, there have been two ways to categorize MPC schemes:

- Based on time: While *continuous-time MPC* consists of using the system's dynamic model and defining a cost in the form of an integral, in *discrete-time MPC* the system model is sample-based and the cost is in the form of a discrete sum.

- Based on form: *Linear MPC*, where the model consists of linear equations (which is easily solvable with Linear Programming methods) and *Nonlinear MPC* where we take into account the nonlinear model of the system in question.

However, in this work we take into account another way to separate MPC. In conventional MPC schemes, the prediction horizon is chosen a priori and remains constant throughout. It has been shown that in this formulation, which will be denoted as fixed-horizon MPC (FH-MPC) there are certain limitations to the systems

and the control objectives we may tackle, especially when trying to build provable algorithms. In recent years, a significant body of work has dealt with MPC algorithms, where the prediction horizon may change online. The horizon could be considered a variable to be optimized, or it may decrease/increase in a predefined manner. Even though experimental work has shown that this Variable-Horizon MPC (VH-MPC) is of paramount importance to the control community, mathematical foundation has been established only in the linear or linearized case.

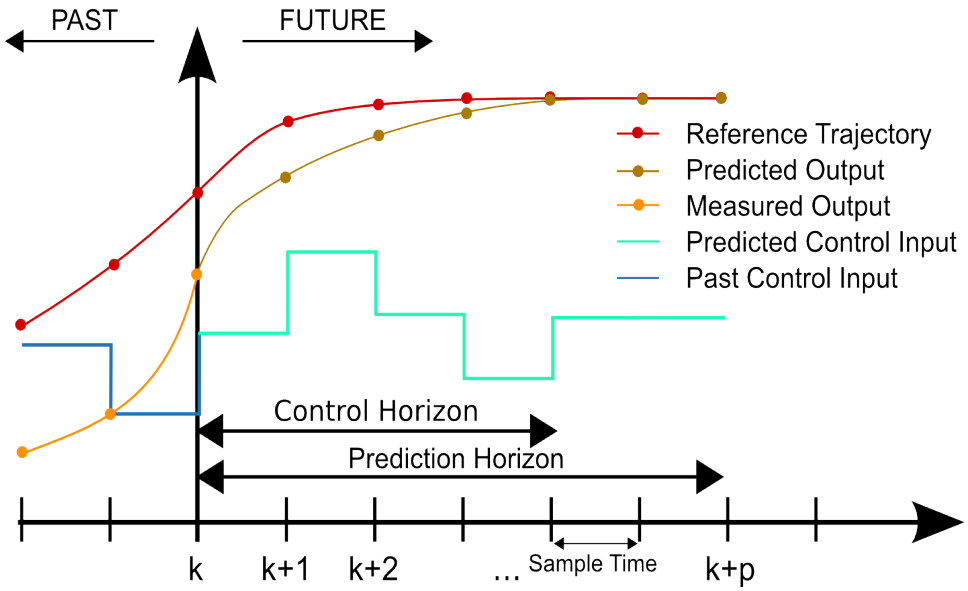

Figure 1.4: Conventional Fixed Horizon MPC

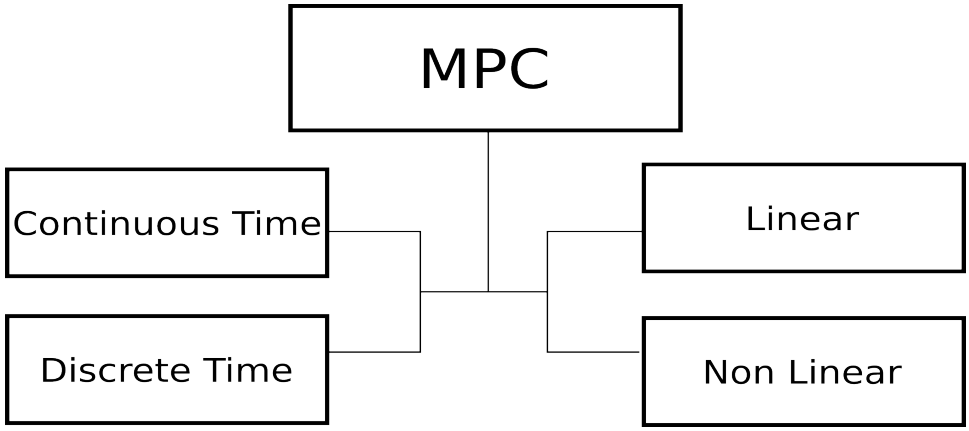

Figure 1.5: Categories of MPC

1.4 Thesis Outline

The thesis is structured as follows:

Chapter 2 provides the derivation of the state-space model for the Quadrotor UAV.

Chapter 3 consists of the formulation for the task of traversing the interior points. Subsequently, different approaches to the trajectory optimization problem are investigated. The indirect method, based on the Euler-Lagrange equations as derived utilizing calculus of variations techniques is presented. The method of Pseudospectral Optimal Control is introduced and utilized for the solution of our problem. Optimal trajectories are presented and discussed.

Chapter 4 presents the notion of Model Predictive Control and its shortcomings. The Variable-Horizon MPC is introduced and its theoretical framework is established. We derive bounds on the admissible disturbances under which nonlinear vehicle maneuvering problems are feasible. A single parameter γ is bounded to provide guaranteed convergence to the final set. Comparative results for different approaches and parameters are provided.

Chapter 5 concludes our work and showcases possible extensions and research directions.

Chapter 2

UAV Modeling

To begin constructing the differential equations describing the behavior of the vehicle, we define the following two reference frames:

- Earth inertial frame C_e
- Body-fixed frame C_b

In the sequel, we adopt the notation ${}^k_jR$ for the rotation matrix with dimension $\mathbf{R}^{3\times3}$ which denotes the orientation of frame C_j with respect to C_i. Also, we denote the linear position of frame C_j with respect to C_i by the vector in $\mathbf{R}^{3\times1}$, ${}^k_jT$.

For notation simplicity reasons, we define $\mathrm{c}\alpha = cos(\alpha)$ and $\mathrm{s}\alpha = sin(\alpha)$ for a given angle α.

The angular position of the UAV (defined by the orientation of C_b with respect to C_e) is given by three consecutive rotations about the main axes which take C_b into C_e. We choose the "roll-pitch-yaw" set of Euler angles in this work.

This way, the rotation matrix $\mathbf{R}(\phi)$ as a function of the Euler angles $\phi = [\phi,\ \theta,\ \psi]^T$ is:

$$\mathbf{R}(\phi)=\begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\phi - \mathrm{s}\psi\mathrm{c}\phi & \mathrm{c}\psi\mathrm{s}\theta\mathrm{c}\phi - \mathrm{s}\psi\mathrm{s}\phi \\ \mathrm{s}\psi\mathrm{c}\theta & \mathrm{s}\psi\mathrm{s}\theta\mathrm{s}\phi - \mathrm{c}\psi\mathrm{c}\phi & \mathrm{s}\psi\mathrm{s}\theta\mathrm{c}\phi - \mathrm{c}\psi\mathrm{s}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} \tag{2.1}$$

Also, we define the Jacobian relating the angular velocity $\boldsymbol{\omega} = [p,\ q,\ r]^T$ with the rate of change of the Euler angles $\dot{\boldsymbol{\phi}} = [\dot{\phi},\ \dot{\theta},\ \dot{\psi}]^T$ by the equation $\boldsymbol{\omega} = \mathbf{E}\dot{\boldsymbol{\phi}}$, where:

$$\mathbf{E}(\boldsymbol{\phi})=\begin{bmatrix} 1 & 0 & -\mathrm{s}\theta \\ 0 & \mathrm{c}\phi & \mathrm{s}\phi\mathrm{c}\theta \\ 0 & -\mathrm{s}\phi & \mathrm{c}\phi\mathrm{c}\theta \end{bmatrix}$$

Figure 2.1: Quadrotor UAV

Having defined the rotations for the system, we formulate the differential equations describing the UAV behavior. These are nothing but the a hybrid system of equations combining the Kinematic model of the platform, as well as the Dynamic model constructed via the Newton-Euler formalism. Therefore, we define the generalized state vector $\mathbf{x} = [\mathbf{P}^T,\ \mathbf{V}^T,\ \boldsymbol{\phi}^T,\ \boldsymbol{\omega}^T]^T$ with the following components:

- The linear body position relative to C_e: $\mathbf{P} = [x,\ y,\ z]^T$
- The linear velocity vector: $\mathbf{V} = [u,\ v,\ w]^T$
- The Euler angles: $\boldsymbol{\phi} = [\phi,\ \theta,\ \psi]^T$
- The angular velocities: $\boldsymbol{\omega} = \mathbf{E}\dot{\boldsymbol{\phi}}$

Under the following two assumptions:

- The origin of the body-fixed frame C_b coincides with the center of mass of the platform.
- The body-fixed frame axes are parallel to the principal axes of inertia. Thus,

the inertia matrix is diagonal:

$$\mathbf{I}=\begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

In vector form the state space first order differential equations are:

$$\frac{d}{dt}(\mathbf{P}) = \mathbf{V} \tag{2.2}$$

$$\frac{d}{dt}(\boldsymbol{\phi}) = \mathbf{E}^{-1}(\boldsymbol{\phi})\boldsymbol{\omega} \tag{2.3}$$

$$\frac{d}{dt}(\mathbf{V}) = m^{-1}{}_{c}^{e}\mathbf{R}(\mathbf{F}+\mathbf{F_D})+\mathbf{g} \tag{2.4}$$

$$\frac{d}{dt}(\boldsymbol{\omega}) = \mathbf{I}^{-1}(\boldsymbol{\tau}+\boldsymbol{\tau}_D-(\mathbf{I}\times\boldsymbol{\omega})) \tag{2.5}$$

where $\mathbf{g} = [0,\ 0,\ g]^T$ is the gravity acceleration vector ($g = -9.81m/s$). The force and torque vectors applied to the body through the rotors are denoted by $\mathbf{F} = [0,\ 0,\ F_z]^T$, $\boldsymbol{\tau} = [\tau_x,\ \tau_y,\ \tau_z]^T$, whereas $\mathbf{F_D}$ and $\boldsymbol{\tau_D}$ are the aerodynamic forces that act upon the body. Though they are presented here for completeness purposes, aerodynamic forces will not be taken into account by the controller. The feedback nature of MPC will ameliorate any issues that arise by this discrepancy in the model, as well as the stochastic nature of these forces.

Summing up, we derive the full state space model of the quadrotor in the form which will be used by the MPC scheme, expressing analytically the equations as:

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ z \\ u \\ v \\ w \\ \phi \\ \theta \\ \psi \\ p \\ q \\ r \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \\ (\mathrm{c}\psi\mathrm{s}\theta\mathrm{c}\phi+\mathrm{s}\psi\mathrm{s}\phi)\cdot(\frac{F_z}{m}) \\ (\mathrm{s}\psi\mathrm{s}\theta\mathrm{c}\phi-\mathrm{c}\psi\mathrm{s}\phi)\cdot(\frac{F_z}{m}) \\ (\mathrm{c}\theta\mathrm{c}\phi)\cdot\frac{F_z}{m}+g \\ p+\frac{\mathrm{s}\phi\mathrm{s}\theta}{\mathrm{c}\theta}q+\frac{\mathrm{c}\phi\mathrm{s}\theta}{\mathrm{c}\theta}r \\ \mathrm{c}\phi\cdot q-\mathrm{s}\phi\cdot r \\ \frac{\mathrm{s}\phi}{\mathrm{c}\theta}\cdot q+\frac{\mathrm{c}\phi}{\mathrm{c}\theta}\cdot r \\ \frac{I_{yy}-I_{zz}}{I_{xx}}\cdot q\cdot r+\frac{\tau_x}{I_{xx}} \\ \frac{I_{zz}-I_{xx}}{I_{yy}}\cdot p\cdot r+\frac{\tau_y}{I_{yy}} \\ \frac{I_{xx}-I_{yy}}{I_{zz}}\cdot p\cdot q+\frac{\tau_z}{I_{zz}} \end{bmatrix} \tag{2.6}$$

Via the optimal controller, we aim to produce the trajectories of the states $\boldsymbol{x} = [x,\ y,\ z,\ u,\ v,\ w,\ \phi,\ \theta,\ \psi,\ p,\ q,\ r]^T$ and the control input $\boldsymbol{u} = [F_z,\ \tau_x,\ \tau_y,\ \tau_z]^T$ that will guide the quadrotor through the desired objectives. For the derivation of

the kinematic and dynamic equations the reader is referred to [7] and the references therein.

It should be noted, that the forces acting on the body should be updated with frequency much greater than which we can solve an optimal control problem. Therefore, we utilize high-frequency stabilizing controllers to achieve trajectory tracking between updates.

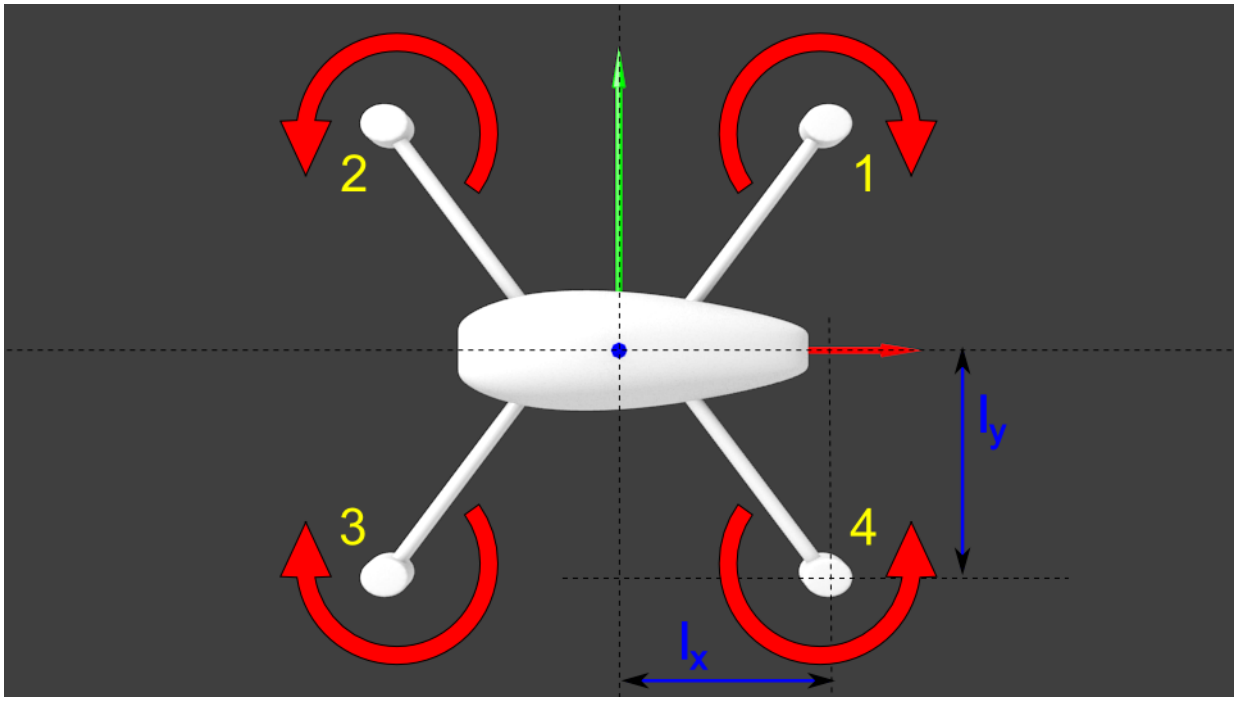

Figure 2.2: Geometric representation of a Quadrotor

Chapter 3

Controller Design

3.1 Optimal Control Problem formulation

In this section, we formulate the optimal control problem (OCP), which will produce the trajectories and control input which lead the quadrotor through the desired interior points.

Given an n-tuple of vectors in Cartesian space, denoting interior points ($\mathbf{x_1}\ \mathbf{x_2}\ \ldots \mathbf{x_n}$), we plan a trajectory passing through these points optimally with respect to a certain cost function.

We formulate the following Optimal Control Problem:

$$\min J(\mathbf{x}, \mathbf{u}, t) = \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}, \tau) dt \tag{3.1}$$

Subject to:

Differential constraints:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) \tag{3.2}$$

This constraint is the derived state-space model (2.6), which describes the kinematics-dynamics of the quadrotor.

Initial constraints:

$$\mathbf{x}(t_0) = \mathbf{x}_0 \tag{3.3}$$

Which denotes the equilibrium point from which the maneuver begins. This vector contains information on every state of the system since $\mathbf{x}_0 \in \mathbf{R}^n$

Terminal constraints:

$$\mathbf{x}(t_f) = \mathbf{x}_f \tag{3.4}$$

Denoting the equilibrium point where the maneuver ends. Again, we specify the final state completely $\mathbf{x}_f \in \mathbf{R}^n$

Interior point constraints:

$$\mathbf{x}^q(t_i) = \mathbf{x}_i^q \tag{3.5}$$

where q states are constrained, and $n-q$ states are to be optimized. This constraint denotes the desired waypoints. Therefore, $\mathbf{x}_f \in \mathbf{R}^q$

Also, the obvious constraint: $t_{i-1} < t_i$, $\forall i \in (1, 2 \dots N)$

Solution via the Indirect Method: Using techniques from the calculus of variations, we may derive the Euler-Lagrange equations, a set of Ordinary Differential Equations which solve the optimal control problem above [8].

First, we must express the interior point constraints (3.5) as a general equality constraint of the form $C(\boldsymbol{x}, \boldsymbol{u}) = 0$. We define the equation:

$$C(\boldsymbol{x}, \boldsymbol{u}) = \mathbf{x}^q(t_i) - \mathbf{x}_i^q = 0 \tag{3.6}$$

and adjoin them to the cost index (3.1) by a set of multipliers ν.

$$\bar{J} = \nu^T C + \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}, \tau) dt$$

Following standard optimal control procedure, defining the Hamiltonian $H = L + \boldsymbol{\lambda}^T \boldsymbol{f}$ and calculating the first variation of the augmented cost index, we get:

$$\delta \bar{J} = \delta(\nu^T C) + \delta \int_{t_0}^{t_f} (H - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}}) dt$$

Splitting the integral at t_i:

$$\begin{aligned}\delta \bar{J} =& \nu^T \frac{\partial C}{\partial t_i} dt_i + \nu^T \frac{\partial C}{\partial \boldsymbol{x}(t_i)} d\boldsymbol{x} - \lambda^T \delta \boldsymbol{x}\big|_{t_i+}^{t_0} - \lambda^T \delta \boldsymbol{x}\big|_{t_0}^{t_i-} \\ &+ (H - \lambda^T \dot{\boldsymbol{x}})\big|_{t=t_i-} dt_i - (H - \lambda^T \dot{\boldsymbol{x}})\big|_{t=t_i+} dt_i + \int_{t_0}^{t_f} \left[(\dot{\lambda^T} + \frac{\partial H}{\partial \boldsymbol{x}}) \delta \boldsymbol{x} + \frac{\partial H}{\partial u} \right] dt\end{aligned}$$

Now, we utilize the relation between a variable's differential and its variation, namely $d\boldsymbol{x}(t_k) = \delta \boldsymbol{x}(t_k) + \dot{\boldsymbol{x}} dt_k$.

By eliminating the variations $\delta \boldsymbol{x}(t_i-)$ and $\delta \boldsymbol{x}(t_i+)$ and regrouping terms, we get:

$$\begin{aligned}\delta \bar{J} = \lambda^T \delta \boldsymbol{x}\big|_{t=t_f} + \lambda^T(t_i+) - \lambda^T(t_i-) + \nu^T \frac{\partial C}{\partial x(t_i)} d\boldsymbol{x}(t_i) \\ + H(t_i-) - H(t_i+) + \nu^T \frac{\partial C}{\partial t_i} dt_i + \lambda^T \delta \boldsymbol{x}\big|_{t=t_0}\end{aligned}$$

Finally, we choose $\lambda(t_i-)$ and $H(t_i-)$ in such a way as to produce stationary values of the cost index for different interior states and times. To accomplish that, we cause their coefficients to vanish:

$$\begin{aligned} \lambda^T(t_i-) &= \lambda^T(t_i+) + \nu^T \frac{\partial C}{\partial \boldsymbol{x}(t_i)} \\ H(t_i-) &= H(t_i+) - \nu^T \frac{\partial C}{\partial t_i} \end{aligned} \tag{3.7}$$

Therefore, equations (3.7) take the role of transversality conditions on time t_i. As a result, while most indirect optimal control methods based on the Euler Lagrange equations require solving a Two-Point Boundary Value Problem, problems with interior point constraints (as well as other types of constraints, but these cases go beyond the scope of this thesis) require solving a Multi-Point Boundary Value Problem.

There are inherent issues in dealing with these kinds of problems which made this solution method, despite being mathematically sounder, unsuitable for this work:

- The costate functions $\boldsymbol{\lambda}(t)$ as well as the Hamiltonian $H(t)$ take arbitrary values which have no physical meaning along the optimal trajectory.
- According to (3.7) jumps appear on these values which drive them from a certain nonintuitive value to another one.
- It has been known to the control community since the 1960s that the Ordinary Differential Equations derived in optimal control theory are extremely sensitive to small variations in the initial guess [9].
- Applying an MPC controller on such a problem would require solving the equations at each recalculation instant, forcing us to guess valid initial values for these two qualities repeatedly.

In order to avoid producing infeasible solutions due to erroneous guesswork, ultimately, we chose to approach the OCP by a *direct method*. In direct transcription methods, the continuous time variable is discretized over a grid and at each grid point the continuous functions of the problem are discretized *first*, and then solved via well-known static optimization techniques, such as interior point methods or Sequential Quadratic Programming.

3.2 Pseudospectral Optimal Control

General formulation: In the pseudospectral approximation to the optimal control problem, the computational interval $[t_0, t_f]$, be it fixed or free, must be transformed to $[-1, 1]$ to facilitate solution of the integral cost by a quadrature. These sets are related by the affine transformation:

$$t = \frac{(t_f - t_0)\tau + (t_f + t_0)}{2} \tag{3.8}$$

As a result, we may choose discrete nodes in $\tau \in [-1, 1]$ to solve the problem and subsequently map them to the interval $t \in [t_0, t_f]$

This way, the cost functional is reformulated as:

$$J(\mathbf{x}, \mathbf{u}, t_f) = \frac{2}{(t_f - t_0)} \int_{-1}^{1} L(\mathbf{x}, \mathbf{u}, \tau) d\tau$$

In any pseudospectral method, both the system's states and control variables are approximated by a finite set of interpolating orthogonal polynomials. We choose the Legendre Pseudospectral Method, in which the following holds:

$$\mathbf{x}(\tau) \approx \mathbf{X}(\tau) = \sum_{i=1}^{N} \mathcal{L}_i(\tau)\mathbf{X}(\tau_i)$$
$$\mathbf{u}(\tau) \approx \mathbf{U}(\tau) = \sum_{i=1}^{N} \mathcal{L}_i(\tau)\mathbf{U}(\tau_i)$$

where $\mathcal{L}_i(\tau)(i = 1, \ldots, N)$ is an orthogonal basis of N Lagrange polynomials defined by:

$$\mathcal{L}_i(\tau) = \frac{1}{N(N+1)P_N(\tau_i)} \frac{(\tau^2 - 1)\dot{P}_N(\tau)}{\tau - \tau_i}$$

Where $P_N(\tau)$ is the N^th degree Legendre polynomial.

Next, we define the collocation points of the method. The Legendre-Gauss-Lobatto points are chosen to enable better approximation of the constraints as well as the discretized cost index. The LGL nodes are:

- $\tau_0 = -1$
- For $\tau_i \in (-1, 1)$ the points are the roots of the derivative of the Legendre polynomials $\dot{P}_N(\tau)$:
- $\tau_f = 1$

along with their respective weights:

$$w_i = \frac{2}{N(N+1)} \frac{1}{[P_N(\tau_i)^2]}$$

Subsequently, we perform the quadrature approximation to the cost index integral:

$$J(\boldsymbol{X}(\tau), \boldsymbol{U}(\tau), t_f) = \frac{t_f - t_0}{2} \sum_{i=1}^{N} L(\tau_i) w_i$$

Finally, one of the most important advantages of Pseudospectral control, is its ability to transform the differential constraints of the problem to algebraic oven the whole time interval through the following matrix multiplication:

$$\dot{\boldsymbol{X}}(\tau) = \sum_{i=1}^{N} \dot{\mathcal{L}}(\tau) \boldsymbol{X}(\tau_i) = \sum_{i=1}^{N} D_{kl} \boldsymbol{X}(\tau_i) \tag{3.9}$$

Where the differentiation matrix D_{kl} is given by:

$$D_{kl} = \begin{cases} \frac{P_N(\tau_i)}{P_N(\tau_l)} \frac{1}{\tau_i - \tau_l} & , i \neq l \\ \frac{-N(N+1)}{4} & , i = l = 0 \\ \frac{N(N+1)}{4} & , i = l = N \\ 0 & \text{otherwise} \end{cases}$$

Having successfully transcribed the continuous OCP into the Nonlinear Program (NLP):

$$J(\boldsymbol{X}, \boldsymbol{U}, tf) = \frac{t_f - t_0}{2} \sum_{i=0}^{N} L(\boldsymbol{X}, \boldsymbol{U}) w_i$$

subject to:

$$\sum_{i=0}^{N} D_{kl} \boldsymbol{X}(\tau_i) - \frac{t_f - t_0}{2} f(\boldsymbol{X}, \boldsymbol{U}) = 0 \qquad \text{the discretized differential equations (2.6)}$$

$$e(\boldsymbol{X}_0, \boldsymbol{X}_f, t_0, t_f) = 0 \quad \text{discrete event constraints at initial/final times}$$

we are able to solve it using well-known methods such as the Sequential Quadratic Programming or the Interior Point Method.

Interior waypoint application: In order to implement the methodology of Pseudospectral Optimal Control to our problem, we present the notion of *knots*. According to this extension to the method, we split the problem in phases from t_0 to t_1 (when the system passes through the first point), from t_1 to t_2 and so on. Therefore, the choice of LGL nodes is made in order to incorporate the notion

of double Lobatto points i.e. two boundary points on top of one another. The framework that has been developed allows for information exchange between the phases (including the discontinuities inherent to our problem)

The mathematical framework of pseudospectral knotting methods can be derived directly from the one presented above, therefore the inclusion of its equations are omitted.

Discussion: Initially introduced in the early 2000s [10], new concepts and applications of Pseudospectral Optimal Control continue to appear in the literature. The basic issue we faced during this thesis was the higher computational burden that comes with the method compared to classic collocation techniques (which propagate the states using the Euler method). However, in recent years, a number of research groups have addressed this issue, leading to publications which propose frameworks for better real time implementation of the method [11].

In this work, there are specific reasons we chose to solve the optimal control problem with the Pseudospectral Optimal Control

- A powerful tool in the Pseudospectral method is the theoretical framework supporting it. With the development of the *Covector Mapping Principle* [12], the indirect continuous Hamiltonian and costate functions of the OCP are directly related with the KKT dual variables of the discretized problem's NLP. Subsequently, we are certain that the optimal solutions to the discrete problem are equal to the continuous ones. Furthermore, the dual variables obtained by the Pseudospectral method can be used as valid initial guesses for an indirect method.

- The LGL grid contains nodes that are more cluttered close to the origin of the maneuver. That way, implemented in an MPC framework, we obtain a more accurate optimal solution for the first steps of the control horizon.

- By inspecting Eq.(3.8) we note that the interior times t_i and final time t_f don't affect the dimension of the NLP. As a result, Pseudospectral methods allow as to extend the prediction horizon of the MPC with no cost in the computational time.

3.3 Trajectory optimization results

In this section, simulated results are presented where a quadrotor is tasked to pass through a series of waypoints.

The cost index we wish to minimize is:

$$J(x, u, t_f) = \int_{t_0}^{t_f} (1 + \gamma \|\boldsymbol{u}(\tau)\|) d\tau$$

The importance of the particular formulation of the cost function will become apparent in Chapter 4.

The quadrotor begins its maneuver from a resting state

$$\boldsymbol{x}_0 = [0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0]^T$$

The Cartesian coordinates of the desired interior points are:

$$\begin{aligned} \boldsymbol{x}_1 &= [10,\ 10,\ 15]^T \\ \boldsymbol{x}_2 &= [10,\ 10,\ 0]^T \\ \boldsymbol{x}_3 &= [13,\ 18,\ 20]^T \\ \boldsymbol{x}_4 &= [10,\ 10,\ 15]^T \end{aligned}$$

Finally, the maneuver ends in the fully defined state:

$$\boldsymbol{x}_f = [10,\ 10,\ 5,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0]^T$$

The optimization problem is solved using the Pseudospectral Optimal Control method, as is implemented in the software PSOPT [13]

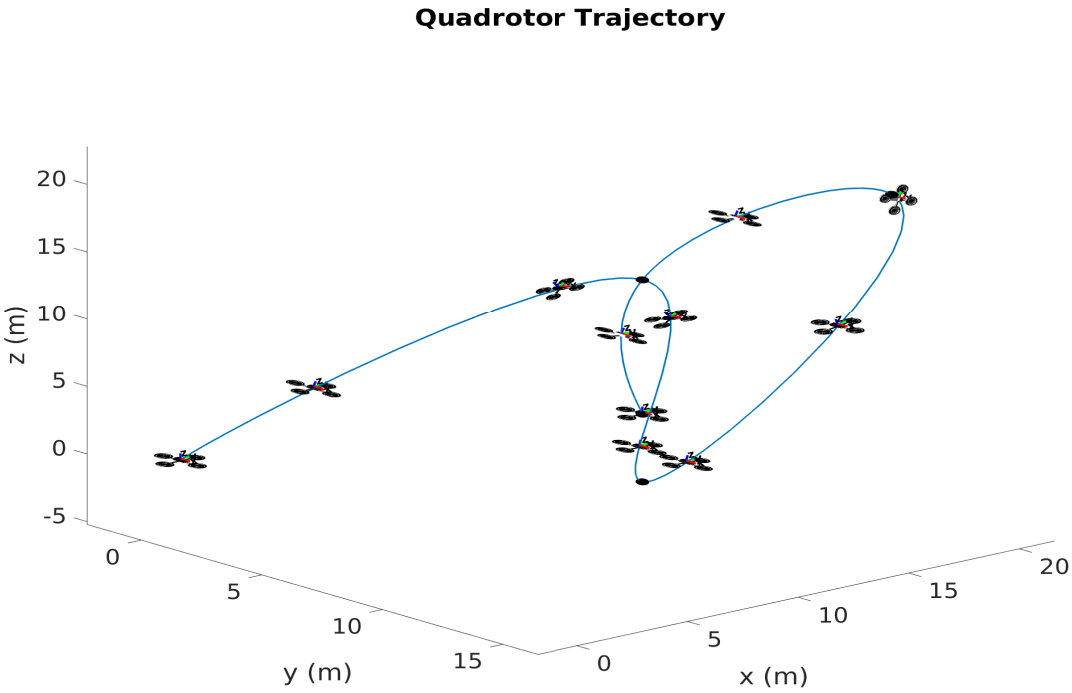

Figure 3.1: 3D trajectory of the quadrotor without external disturbances by utilizing the Pseudospectral Optimal Control method.

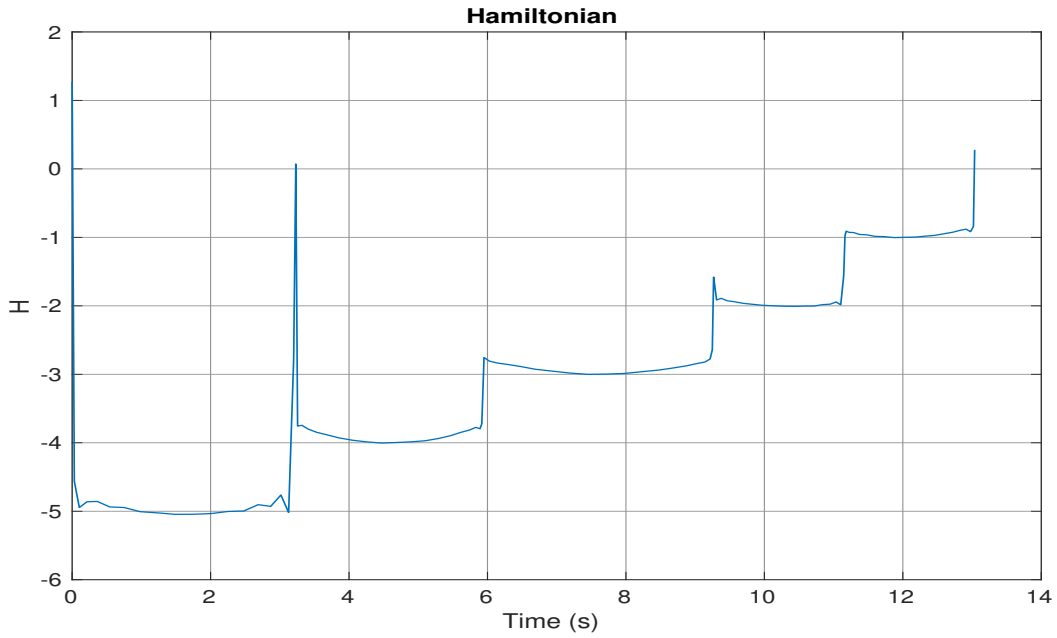

Figure 3.2: Optimal Hamiltonian as produced according to the Covector Mapping Principle. It can be seen that random value jumps appear at each phase. However, we note that $\frac{\partial H}{\partial t}$ is constant, in agreement with the Euler-Lagrange theory.

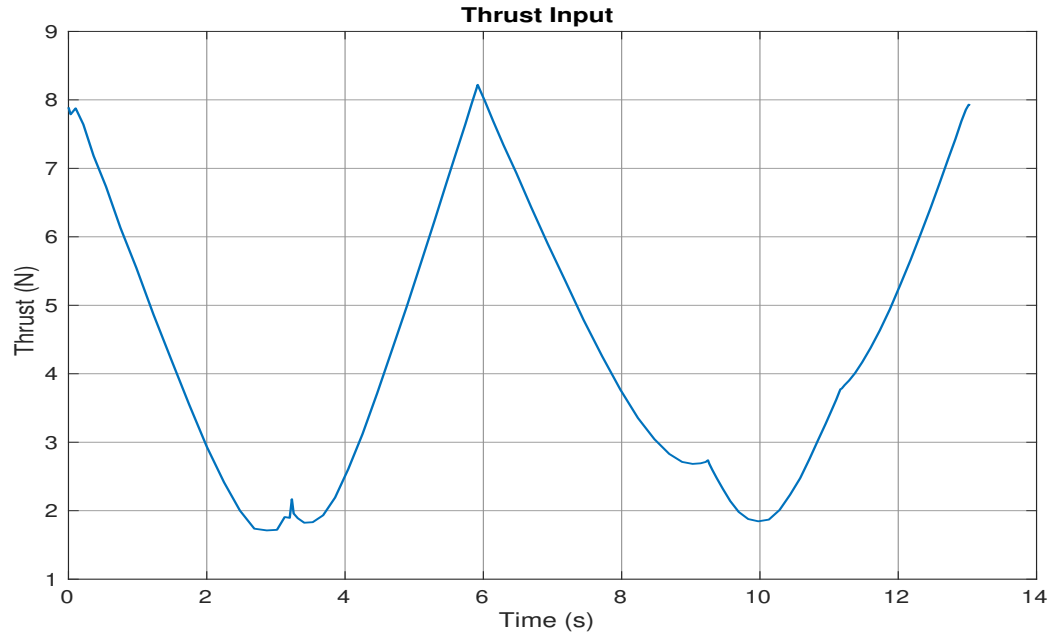

Figure 3.3: Optimal thrust input.

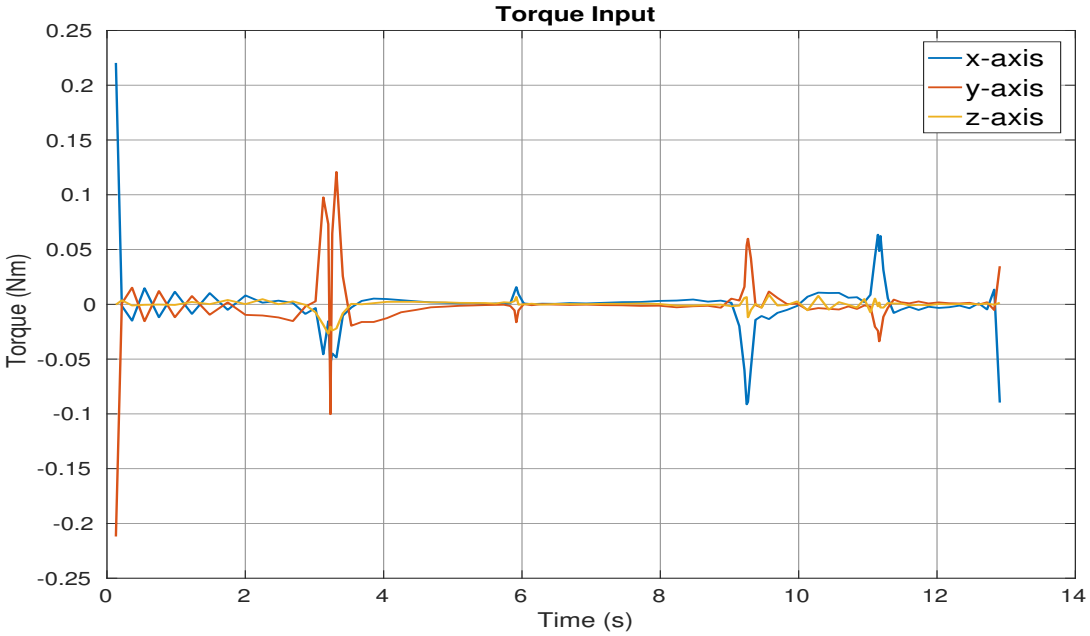

Figure 3.4: Applied torque input in the three axes of the quadrotor.

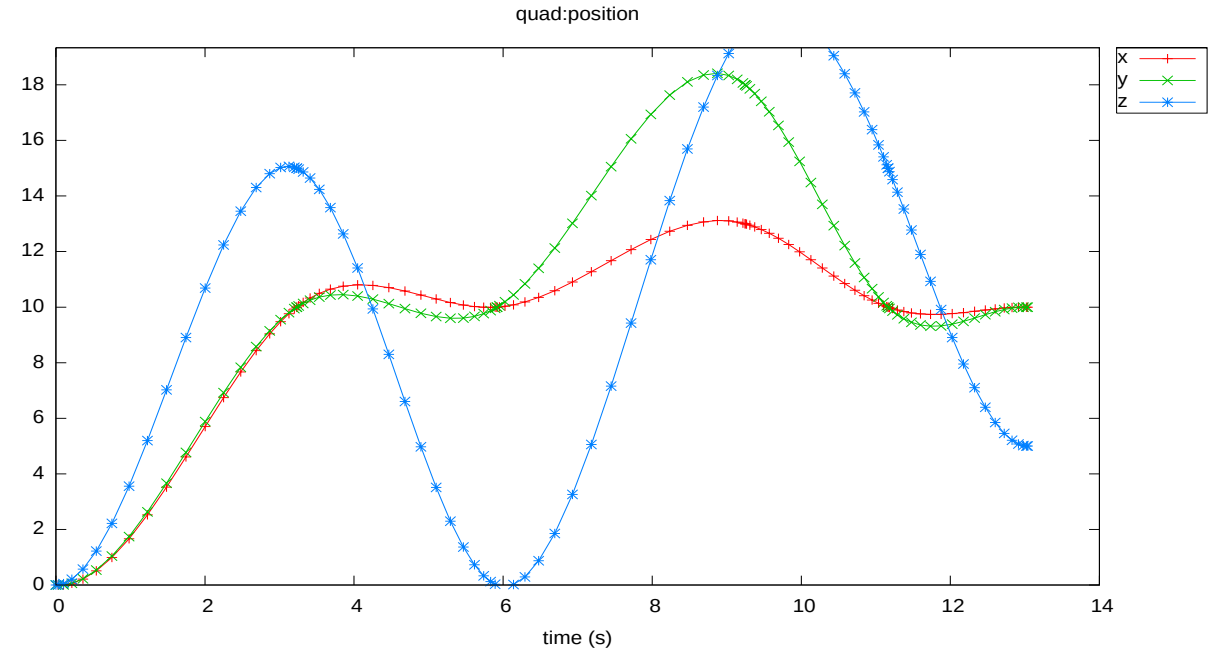

Figure 3.5: Optimal positions. We present the variables as produced by a run of the Pseudospectral Optimal Controller. The asymmetric grid, finer near the Lobatto nodes can be observed.

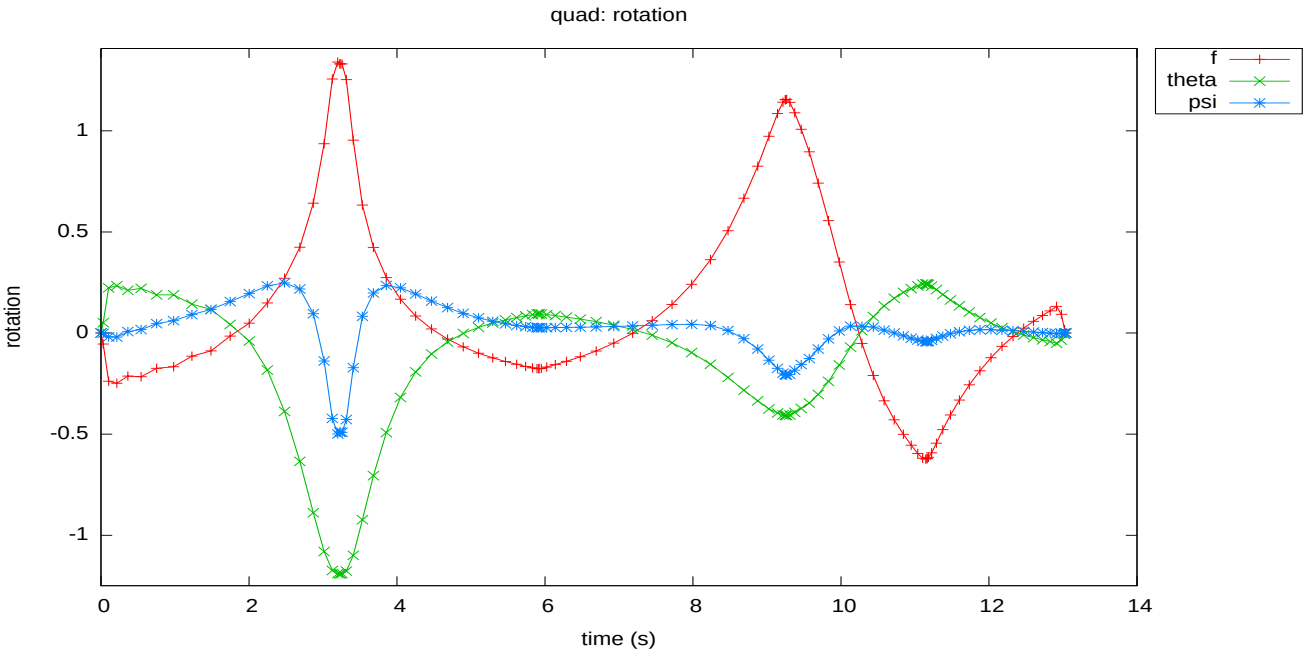

Figure 3.6: Optimal angles.

Chapter 4

Variable-Horizon Model Predictive Control

4.1 Background

In its most general formulation a Model Predictive Control scheme is a simple and intuitive method. It requires an explicit system model, utilized as a predictor over a finite time prediction horizon. The MPC scheme provides an optimal input trajectory over the horizon which, minimizing a predefined cost index for the predicted behavior of the system, guides us to a final configuration. Since the resulting output of the system will be different from the predicted output (due to disturbances), MPC is solved iteratively. As a result, MPC is a closed loop controller. [14]

The mathematical formulation of the general MPC algorithm is as follows:

$$\min_{\boldsymbol{u}} J(t_k) = \int_{t_k}^{t_k+T_p} L(\boldsymbol{x}, \boldsymbol{u}) d\tau$$

subject to:

$$\hat{\boldsymbol{x}}(t_k) = \boldsymbol{x}(t_k)$$

$$\dot{\hat{\boldsymbol{x}}}(\tau) = f(\hat{\boldsymbol{x}}(\tau), u(\tau)), \text{ for } \tau \in [t_k, t_k + T_p]$$

$$\hat{\boldsymbol{x}}(t_k + T_p) = \boldsymbol{x}_f$$

where $\hat{\boldsymbol{x}}$ is the predicted state. Therefore, we seek at each time step t_k to find the input $u(t)$, which will minimize the cost index J. The optimizations constraints force the optimizer to use the real (measured) state at t_k as an initial value, constrain the system to evolve according to the predicted dynamics $f(\cdot, \cdot)$ and force the state at the end of the horizon to enter the final configuration $\boldsymbol{x}_f$.

However, this conventional formulation of MPC assumes that the final configuration we wish to reach is an invariant set. Both experimental work and theoretical

research holds this assumption [15], [16], [17]. In the problem we face, the interior points we want to reach are not inside invariant sets (not being equilibrium points of the system).

We introduce the notion of *Variable-Horizon MPC*(VH-MPC), where find both the optimal control input and the optimal prediction horizon over which we must optimize. Although a variety of problems in MPC (especially those dealing with vehicle guidance) require the use of variable-horizon [18] the mathematical guarantees are seldom derived. The first theoretical work on VH-MPC was presented in [16]. It was later extended in [19], [20]. In all cases, the authors considered Linear-MPC (due to the access we have to the system's state transition matrices). Inherent to the definition of VH-MPC is the idea of *completion*. Since we are unable to prove asymptotic stability of the algorithm to a non-equilibrium point, we prove that the state will be driven to the terminal set in finite number of steps.

In the sequel, we combine the work of [21] with that of [19]. In [21], where the authors approach the question of stability by guaranteeing recursive feasibility and convergence of the MPC. In [19] the authors formulate a a cost index penalizing time-to-go to the target set, as well as control effort:

$$\min_{\boldsymbol{u},\, t_f} J(t_k) = \int_{t_k}^{t_f(\boldsymbol{x}(t_k))} (1 + \gamma \|\boldsymbol{u}(\tau; \boldsymbol{x}(t_k))\|) d\tau$$

The choice of this cost index is quite intuitive. Instead of penalizing the error from the final state, we use the time it takes for the system to reach the target. However, to avoid saturation of the actuators and to provide smoother trajectories, we also penalize the control effort. This way, the convergence of the scheme will be investigated based on a single variable, γ, which will turn the optimization problem more "time-optimal" or "energy-optimal".

Since our goal is to create a theoretical framework for nonlinear systems, we use theorems from [22], where connections between the initial state of an OCP and the final states were presented.

4.2 Robust Nonlinear VH-MPC

4.2.1 Preliminaries

In order to perform the robustness analysis of the MPC algorithm, we also present some properties for the general class of nonlinear systems, of which (2.6) is an example. This kind of systems can be expressed by the following ordinary differential vector equation:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) \tag{4.2}$$

where $\boldsymbol{x}$ denotes the system's state and $\boldsymbol{u}$ the control input. These variables belong to subsets of $\mathbb{R}^n$ and $\mathbb{R}^m$ respectively:

$$\boldsymbol{x} \in \boldsymbol{X} \subseteq \mathbb{R}^n$$
$$\boldsymbol{u} \in \boldsymbol{U} \subseteq \mathbb{R}^m$$

where, in the case of the quadrotor, $n = 12$ and $m = 4$.

Equation (4.2) describes the *nominal* system, in which no stochastic behavior is considered. Assuming additive disturbances, we also define the *real* system described by the equation:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{w}(t) \tag{4.3}$$

where $w \in \boldsymbol{W}$ denotes the unknown disturbance acting on the system which belongs to the bounded set $\boldsymbol{W} \subset \mathbb{R}^n$. Henceforth, we denote by $\hat{\boldsymbol{x}}(t_{k+j}|t_k)$ the vector of predicted state of the system at time t_{k+j}, based on the measurement of the system at time t_k, applying a control profile $\boldsymbol{u}(t)$ on the nominal model:

$$\hat{\boldsymbol{x}}(t_{k+j}|t_k) = \boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+j}} f(\boldsymbol{x}, \boldsymbol{u}) d\tau \tag{4.4}$$

It is obvious that $\hat{\boldsymbol{x}}(t_k|t_k) = \boldsymbol{x}(t_k)$. We also denote as $\boldsymbol{x}(t_{k+j})$ the real state of the system at time t_{k+j}, applying the same control profile $\boldsymbol{u}(t)$ on the real model:

$$\boldsymbol{x}(t_{k+j}) = \boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+j}} (f(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{w}(\tau)) d\tau \tag{4.5}$$

4.2.2 Definitions

In this subsection, assumptions regarding the system as well as definitions of some basic notions are presented:

Assumption 1: f is locally Lipschitz in $\boldsymbol{X}$ i.e., there is a positive constant $L_f < \infty$, such that for every $\boldsymbol{u} \in \boldsymbol{U}$, $\|f(\boldsymbol{x}_1, \boldsymbol{u}) - f(\boldsymbol{x}_2, \boldsymbol{u})\| \leq L_f \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|$, and there is a constant $0 \leq M < \infty$, such that for every $\boldsymbol{u} \in \boldsymbol{U}$, $\|f(\boldsymbol{x}, \boldsymbol{u})\| \leq M$.

Definition 1: Given a system subject to (4.3), we define its reachable target set $\mathbb{A}(\boldsymbol{x}_0)$, starting from state $\boldsymbol{x}_0$, after some time period s, as:

$$\mathbb{A}(\boldsymbol{x}_0) = \{\boldsymbol{x} : \boldsymbol{x}(s) \in R(s : \boldsymbol{x}_0) \cap \boldsymbol{X}_t\}$$

where $\boldsymbol{X}_t$ is some predefined terminal target set and for every $\boldsymbol{x}_0 \in \mathbb{R}^n$ and s $\geq t_0$: $R(s; \boldsymbol{x}_0) = \{\boldsymbol{x}(s), \boldsymbol{x}(\cdot)$ is a trajectory of the system on $[t_0, s], \boldsymbol{x}(t_0) = \boldsymbol{x}_0\}$

Definition 2: Given two compact sets $\mathcal{P}$ and $\mathcal{Q}$, the Hausdorff distance $H(\mathcal{P}, \mathcal{Q})$ is defined as [23]:

$$H(\mathcal{P}, \mathcal{Q}) = \max\{H^+(\mathcal{P}, \mathcal{Q}), H^+(\mathcal{Q}, \mathcal{P})\} \tag{4.6a}$$

$$H^+(\mathcal{P}, \mathcal{Q}) = \inf\{a \geq 0; \mathcal{P} \subset \mathcal{Q} + a \cdot \mathcal{B}\} \tag{4.6b}$$

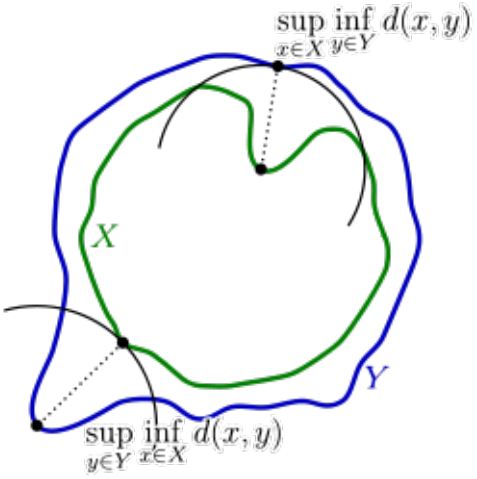

Figure 4.1: Geometric representation of the Hausdorff distance for two sets in $\mathbb{R}^2$. The Hausdorff distance is the longest distance you can be forced to travel by an adversary who chooses a point in one of the two sets, from where you then must travel to the other set.

Lemma 1: Given two initial states $\boldsymbol{x}$, $\boldsymbol{x}' \in \mathbb{R}^n$, the following Lipschitz property of the reachable sets holds:

$$H(\mathbb{A}(\boldsymbol{x}), \mathbb{A}(\boldsymbol{x}')) \leq L_H |\boldsymbol{x} - \boldsymbol{x}'| \tag{4.7}$$

Proof: See proof of Theorem 2.1 in [22].

Lemma 2: Considering a system measurement at time t_k, as well as an upper bound on the norm of the admissible uncertainties W, i.e.: $\forall \boldsymbol{w} \in \boldsymbol{W}, \|\boldsymbol{w}\| \leq W$ then, for a given control profile $\boldsymbol{u}(t)$, the difference between the predicted state based on the nominal model and the real system at some time $t_{k+1} \geq t_k$, is bounded by:

$$\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq \frac{W}{L_f}(e^{L_f \cdot (t_{k+1} - t_k)} - 1) \tag{4.8}$$

Proof: Formulating the difference between the real and the predicted state, we get:

$$\begin{aligned}
&\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| = \|\boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} (f(\boldsymbol{x},\boldsymbol{u}) + w(\tau)d\tau \\
&- \boldsymbol{x}(t_k) - \int_{t_k}^{t_{k+1}} f(\hat{\boldsymbol{x}},\boldsymbol{u})d\tau\| \\
=&\| \int_{t_k}^{t_{k+1}} (f(\boldsymbol{x},\boldsymbol{u}) + w(\tau)d\tau - \int_{t_k}^{t_{k+1}} f(\hat{\boldsymbol{x}},\boldsymbol{u})d\tau\| \\
=&\| \int_{t_k}^{t_{k+1}} (f(\boldsymbol{x},\boldsymbol{u}) - f(\hat{\boldsymbol{x}},\boldsymbol{u}))d\tau - \int_{t_k}^{t_k+1} w(\tau)d\tau\|
\end{aligned}$$

Utilizing the triangle inequality and the Lipschitz property of the system:

$$\begin{aligned}
&\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \\
&\leq \| \int_{t_k}^{t_{k+1}} (f(\boldsymbol{x},\boldsymbol{u}) - f(\hat{\boldsymbol{x}},\boldsymbol{u})d\tau\| + \| \int_{t_k}^{t_{k+1}} w(\tau)d\tau\| \\
&\leq \int_{t_k}^{t_{k+1}} \|f(\boldsymbol{x},\boldsymbol{u}) - f(\hat{\boldsymbol{x}},\boldsymbol{u})\|d\tau + \int_{t_k}^{t_{k+1}} \|w(\tau)\|d\tau \\
&\leq \int_{t_k}^{t_{k+1}} L_f \cdot \|\boldsymbol{x}(\tau) - \hat{\boldsymbol{x}}(\tau)\|d\tau + \int_{t_k}^{t_{k+1}} W d\tau \\
&= \int_{t_k}^{t_{k+1}} L_f \cdot \|\boldsymbol{x}(\tau) - \hat{\boldsymbol{x}}(\tau)\|d\tau + W \cdot (t_k - t_{k+1})
\end{aligned}$$

Therefore, we have reached the following inequality:

$$\begin{aligned}
\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq &\int_{t_k}^{t_{k+1}} L_f \cdot \|\boldsymbol{x}(\tau) - \hat{\boldsymbol{x}}(\tau)\|d\tau \\
&+ W \cdot (t_k - t_{k+1})
\end{aligned}$$

Applying the Gronwall-Bellman inequality, we get:

$$\begin{aligned}
\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq &\int_{t_k}^{t_{k+1}} L_f \cdot w(\tau - t_k) e^{L_f \cdot (t_{k+1} - \tau)} d\tau \\
&+ W \cdot (t_{k+1} - t_k)
\end{aligned}$$

Integrating by parts the right-hand side integral, we get:

$$\|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq \frac{W}{L_f} e^{L_f \cdot (t_{k+1}-\tau)} - \frac{W}{L_f}$$
$$+ W(t_{k+1} - t_k) - W(t_{k+1} - t_k)$$
$$\Rightarrow \|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq \frac{W}{L_f}(e^{L_f \cdot (t_{k+1}-t_k)} - 1)$$

which completes the proof.

As is usual in MPC frameworks, the proof of stability consists of two separate parts. First, a feasibility analysis is presented through which we obtain an upper bound on the admissible disturbances. Then, based on these results, a convergence analysis provides guarantees on stability or ultimate boundedness [21], [15]. It should be noted that in order to prove stability in MPC, the target set is required to be invariant. However, since this is not our case, only finite time completion of the problem, is provable [24], [19].

4.2.2.1 Feasibility

In this section the following argument is presented: Given a solution of the OCP at some triggering instance t_k, it is possible to calculate the maximum deviation from the final target under bounded disturbances for the next triggering instance t_{k+1}. Therefore, by specifying the tolerance area around the target, the upper bound on the admissible disturbances may be derived.

Consider the real system (4.3) of the unmanned quadrotor. The Nonlinear Model Predictive Controller of Chapter 3 guides the system to a desired final state, if the disturbances are bounded by:

$$W \leq \frac{L_f}{L_H}(e^{L_f \cdot \delta t} - 1)^{-1} \cdot \rho \tag{4.9}$$

where ρ to be a positive predefined tolerance.

Proof: Consider the solution of the optimization process of Chapter 3 at some time t_k i.e., the control profile $\boldsymbol{u}^*(\tau; \boldsymbol{x}(t_k))$ and the optimal final time $t_f^*(\boldsymbol{x}(t_k))$ which guides the nominal system from an initial measured state $\boldsymbol{x}(t_k)$ to the desired final state $\boldsymbol{x}_d$. We predict the nominal state at the next calculation instant $t_{k+1} = t_k + \delta t$. This state belongs to the nominal optimal trajectory as described in (4.4). The nominal system's remaining time to reach the target, beginning at t_{k+1} is $t_{rem} = t_f^*(\boldsymbol{x}(t_k)) - \delta t$. Due to the disturbances, the measured state of the system will be according to (4.5).

We know that for $s = t_{rem}$, for (4.7) the following holds:

$$\mathbb{A}(\hat{\boldsymbol{x}}(t_{k+1}|t_k)) = \mathbb{A}(\boldsymbol{x}(t_k)) = \boldsymbol{x}_d$$

meaning that the reachable target set at time t_k is a singleton containing the waypoint, while:

$$\mathbb{A}(\boldsymbol{x}(t_{k+1})) = \bar{\boldsymbol{X}}_d$$

where $\bar{\boldsymbol{X}}_d$ is an extended set around $\boldsymbol{x}_d$. These equalities mean that even if the real system cannot be guided to the target state the way the nominal can, it could utilize the remaining time to approach it. In order to ensure that the distance between $\bar{\boldsymbol{X}}_d$ and $\boldsymbol{x}_d$ is smaller that the tolerance ρ:

$$\begin{aligned} &H(\bar{\boldsymbol{X}}_d, \boldsymbol{x}_d) \leq \rho \Rightarrow \\ &H(\mathbb{A}(\boldsymbol{x}(t_{k+1})), \mathbb{A}(\hat{\boldsymbol{x}}(t_{k+1}|t_k))) \leq \rho \end{aligned}$$

Due to Lemma 1:

$$\Rightarrow L_H \|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \leq \rho$$

And ultimately, utilizing Lemma 2:

$$\begin{aligned} &\Rightarrow L_H \frac{W}{L_f}(e^{L_f \cdot \delta t} - 1) \leq \rho \\ &\Rightarrow W \leq \frac{L_f}{L_H}(e^{L_f \cdot \delta t} - 1)^{-1} \cdot \rho \end{aligned}$$

which completes the proof.

Remark 1: During the proof of feasibility, the system was given a horizon of $t_{rem} = t_f^*(\boldsymbol{x}(t_k)) - \delta t$ to approach the target. If there exists an upper bound on the horizon t_{max}^*, and $t_f^*(\boldsymbol{x}(t_k)) \leq t_{max}^*$, then the system can perform a longer maneuver to get closer to the target. This fact makes our analysis stricter but creates no problems on our proofs.

4.2.2.2 Convergence

In order to prove convergence of the algorithm, a Lyapunov function must be shown to decrease. Following previous work on MPC stability analysis, it is proven that for two consecutive triggers of the OCP, the optimal cost decreases as long as the parameter γ is within a previously calculated range. By bounding this parameter, the controller is permitted to apply enough control effort to overcome the disturbances, without leading to excessive energy consumption or overly aggressive maneuvers. First, the following lemma is presented:

Lemma 3: The optimal final time when optimizing the same cost function is Lipschitz with respect to the initial state, with Lipschitz constant L_t.

$$\left\|t_f^*(\boldsymbol{x}) - t_f^*(\boldsymbol{x}')\right\| \leq L_t \left\|\boldsymbol{x} - \boldsymbol{x}'\right\| \tag{4.10}$$

Proof: See the proof of Theorem 3.1 in [22].

We note that the optimal control profile as well as the optimal final time depend solely on the initial state.

Consider the real system (4.3) of the unmanned quadrotor. The Nonlinear Model Predictive Controller under admissible disturbances, guides the system to its target set in finite time if:

$$\gamma \leq \frac{L_t/L_H\rho + L_t \cdot M \cdot \delta t}{sup\|\boldsymbol{u}\| \cdot t^*_{max}}$$

Proof: Consider $J^*(t)$, the optimal cost, as a candidate Lyapunov-like function. we wish to choose the variable γ such that the cost to reach the target decreases between two consecutive triggering instances. Practically, we wish to shift the weight towards time-optimal solutions (rather that energy-optimal), so as to let the system use enough control effort to account for the disturbances. Take the optimal cost from initial time t_k, along with its corresponding initial measured state $\boldsymbol{x}(t_k)$:

$$J^*(t_k) = \int_{t_k}^{t^*_f(\boldsymbol{x}(t_k))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\tau; \boldsymbol{x}(t_k))\|) d\tau$$

Obviously, at the next calculation instant $t_{k+1} = t_k + \delta t$, the cost is:

$$J^*(t_{k+1}) = \int_{t_{k+1}}^{t^*_f(\boldsymbol{x}(t_{k+1}))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\tau; \boldsymbol{x}(t_{k+1}))\|) d\tau$$

where, $\boldsymbol{x}(t_{k+1})$ can be derived if we apply the optimal control $u^*(\tau)$ to (4.5). To simplify the notation, the dependency on running time τ will be dropped unless explicitly needed. Consider the difference between the optimal costs above:

$$\begin{aligned}
J^*(t_{k+1}) - J^*(t_k) = & \int_{t_{k+1}}^{t^*_f(\boldsymbol{x}(t_{k+1}))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_{k+1}))\|) d\tau \\
& - \int_{t_k}^{t^*_f(\boldsymbol{x}(t_k))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\|) d\tau \\
= & \int_{t_{k+1}}^{t^*_f(\boldsymbol{x}(t_{k+1}))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_{k+1}))\|) d\tau \\
& - \int_{t_{k+1}}^{t^*_f(\boldsymbol{x}(t_k))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\|) d\tau \\
& - \int_{t_k}^{t_{k+1}} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\|) d\tau
\end{aligned}$$

Since any part of the cost function is positive, we may bound the equation above as

such:

$$
\begin{aligned}
J^*(t_{k+1}) - J^*(t_k) &\leq \int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_{k+1}))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_{k+1}))\|) d\tau \\
&- \int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_k))} (1 + \gamma \cdot \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\|) d\tau \\
&= t_f^*(\boldsymbol{x}(t_{k+1})) - t_{k+1} + \gamma \int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_{k+1}))} \|\boldsymbol{u}^*(\boldsymbol{x}(t_{k+1}))\| d\tau \\
&- t_f^*(\boldsymbol{x}(t_k)) + t_{k+1} - \gamma \int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_k))} \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\| d\tau \\
&\Rightarrow J^*(t_{k+1}) - J^*(t_k) \leq t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(\boldsymbol{x}(t_k)) \\
&+ \gamma \left(\int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_{k+1}))} ||\boldsymbol{u}^*(\boldsymbol{x}(t_{k+1}))|| d\tau - \int_{t_{k+1}}^{t_f^*(\boldsymbol{x}(t_k))} \|\boldsymbol{u}^*(\boldsymbol{x}(t_k))\| d\tau \right)
\end{aligned}
$$

Since we assume that the control input is bounded, i.e. $\boldsymbol{u} \in \boldsymbol{U}$, it is correct to use the following inequality:

$$inf\|\boldsymbol{u}\| \leq \|\boldsymbol{u}\| \leq sup\|\boldsymbol{u}\|$$

Therefore we may further bound the cost difference:

$$
\begin{aligned}
J^*(t_{k+1}) - J^*(t_k) &\leq t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(t_k) \\
+\gamma \|(sup\|\boldsymbol{u}\| \cdot t_f^*(\boldsymbol{x}(t_{k+1})) - inf\|\boldsymbol{u}\| \cdot t_f^*(\boldsymbol{x}(t_k)) \\
-t_{k+1}(sup\|\boldsymbol{u}\| - inf\|\boldsymbol{u}\|)\|) \\
\Rightarrow J^*(t_{k+1}) - J^*(t_k) &\leq t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(t_k) \\
+\gamma \left(sup\|\boldsymbol{u}\| \cdot t_f^*(\boldsymbol{x}(t_{k+1})) - inf\|\boldsymbol{u}\| \cdot t_f^*(\boldsymbol{x}(t_k)) \right)
\end{aligned}
$$

We may assume that $inf\|\boldsymbol{u}\| = 0$ and also that there is an upper bound to the time-to-go/horizon t_{max}^*.

$$J^*(t_{k+1}) - J^*(t_k) \leq t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(\boldsymbol{x}(t_k)) + \gamma \left(sup\|\boldsymbol{u}\| \cdot t_{max}^* \right) \tag{4.11}$$

In order to prove convergence, we require that (4.11) is non-positive:

$$
\begin{aligned}
t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(\boldsymbol{x}(t_k)) + \gamma \left(sup\|\boldsymbol{u}\| \cdot t_{max}^* \right) &\leq 0 \\
\Rightarrow \gamma \leq \frac{t_f^*(\boldsymbol{x}(t_k)) - t_f^*(\boldsymbol{x}(t_{k+1}))}{sup\|\boldsymbol{u}\| \cdot t_{max}^*}
\end{aligned}
$$

Given that $\gamma \geq 0$:

$$
\begin{aligned}
\|\gamma\| &\leq \left\| \frac{t_f^*(\boldsymbol{x}(t_k)) - t_f^*(\boldsymbol{x}(t_{k+1}))}{sup\|\boldsymbol{u}\| \cdot t_{max}^*} \right\| \\
\Rightarrow \|\gamma\| &\leq \frac{\left\| t_f^*(\boldsymbol{x}(t_k)) - t_f^*(\boldsymbol{x}(t_{k+1})) \right\|}{sup\|\boldsymbol{u}\| \cdot t_{max}^*} \\
&= \frac{\left\| t_f^*(\boldsymbol{x}(t_{k+1})) - t_f^*(\boldsymbol{x}(t_k)) \right\|}{sup\|\boldsymbol{u}\| \cdot t_{max}^*}
\end{aligned}
$$

Using Lemma 3 we obtain:

$$\|\gamma\| \leq \frac{L_t \|\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k)\|}{sup\|\boldsymbol{u}\| \cdot t^*_{max}} \tag{4.12}$$

where $\boldsymbol{x}(t_{k+1})$ denotes the measured state at t_{k+1}. Consider the predicted state at the same time, under the optimal control $\boldsymbol{u}^*(\boldsymbol{x}(t_k))$, which is:

$$\hat{\boldsymbol{x}}(t_{k+1}|t_k) = \boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} f(\boldsymbol{x}, \boldsymbol{u}^*(\boldsymbol{x}(t_k)))d\tau$$
$$\Rightarrow \boldsymbol{x}(t_k) = \hat{\boldsymbol{x}}(t_{k+1}|t_k) - \int_{t_k}^{t_{k+1}} f(\boldsymbol{x}, \boldsymbol{u}^*(\boldsymbol{x}(t_k)))d\tau$$

So, we construct the state difference:

$$\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k) = \boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k) + \int_{t_k}^{t_{k+1}} f(\boldsymbol{x}, \boldsymbol{u}^*(\boldsymbol{x}(t_k)))d\tau$$

Taking the norm of the above expression, using the triangle inequality, as well as the lemma 2, we get:

$$\begin{aligned}\|\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k)\| &\leq \|\boldsymbol{x}(t_{k+1}) - \hat{\boldsymbol{x}}(t_{k+1}|t_k)\| \\ &\quad + \|\int_{t_k}^{t_{k+1}} f(\boldsymbol{x}, \boldsymbol{u}^*(\boldsymbol{x}(t_k)))d\tau\| \\ \Rightarrow \|\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k)\| &\leq \frac{W}{L_f}(e^{L_f(t_{k+1}-t_k)} - 1) \\ &\quad + \int_{t_k}^{t_{k+1}} \|f(\boldsymbol{x}, \boldsymbol{u}^*(\boldsymbol{x}(t_k)))\|d\tau\end{aligned}$$

By utilizing Assumption 1:

$$\begin{aligned}\|\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k)\| &\leq \frac{W}{L_f}(e^{L_f(t_{k+1}-t_k)} - 1) + \int_{t_k}^{t_{k+1}} \|M\|d\tau \\ \Rightarrow \|\boldsymbol{x}(t_{k+1}) - \boldsymbol{x}(t_k)\| &\leq \frac{W}{L_f}(e^{L_f(t_{k+1}-t_k)} - 1) + M \cdot (t_{k+1} - t_k)\end{aligned} \tag{4.13}$$

Using (4.13) in (4.12), we get:

$$\gamma \leq \frac{L_t(\frac{W}{L_f}(e^{L_f\delta t} - 1) + M \cdot \delta t)}{sup\|\boldsymbol{u}\| \cdot t^*_{max}}$$

Finally, taking into account Theorem 1:

$$\gamma \leq \frac{L_t/L_H\rho + L_t \cdot M \cdot \delta t}{sup\|\boldsymbol{u}\| \cdot t^*_{max}}$$

which completes the proof.

4.3 Comparative Results

The following cases will be analyzed and compared. First, a comparison between the behavior of the system when a classical fixed-horizon MPC and when the proposed Variable-Horizon MPC are utilized under the same conditions. Then, considering the Variable-Horizon Controller, a comparative study is presented for different magnitudes of disturbance in order to validate Theorem 1. Ultimately, the validity of Theorem 2 is illustrated through a case in which different values of the constant γ are considered.

Case A: Fixed and Variable Horizon

First, we examine a fixed-horizon MPC scheme. It is shown in Fig.4.2 that the quadrotor is unable to reach even the first waypoint in finite time. Employing the proposed Variable-Horizon MPC of this paper leads the quadrotor successfully to the first waypoint (see Fig.4.3). Therefore, even in the absence of additive external disturbances the fixed-horizon MPC fails to converge to the target.

In order to better examine the effects of the various parameters, the following graphs consider the first part of the trajectory, i.e. from the initial resting position to the first waypoint, which is a non-invariant target.

Case B: VH-MPC under disturbances of different magnitude

Consider the Variable-Horizon case, in which additive disturbances are taken into account. The derivation of the various Lipschitz constants through mathematical theorems and simulation results is omitted due to space limitations. In accordance to Theorem 1, $\rho = 0.5$, and $\delta t = 0.1 sec$. The upper bound is found to be $W = 4.5$. It can be seen in Fig.4.4 that applying a greater disturbance, $W = 6$ leads the system to infeasibility. On the other hand, applying a disturbance of $W = 3.5$, the algorithm produces feasible trajectories (see Fig.4.5)

Case C: Feasible VH-MPC with different values of γ

Finally, applying a feasible disturbance, it becomes apparent from Fig.4.6 that for larger γ than Theorem 2 predicts the quadrotor fails to converge to the target. On the contrary, shifting the weight towards time-optimality, the control input becomes large enough to steer the system to the desired state (see Fig.4.7). The simulations agree with the theoretical bound of $\gamma = 10^{-4}$. The derivation procedure of bound M is omitted due to size limitations.

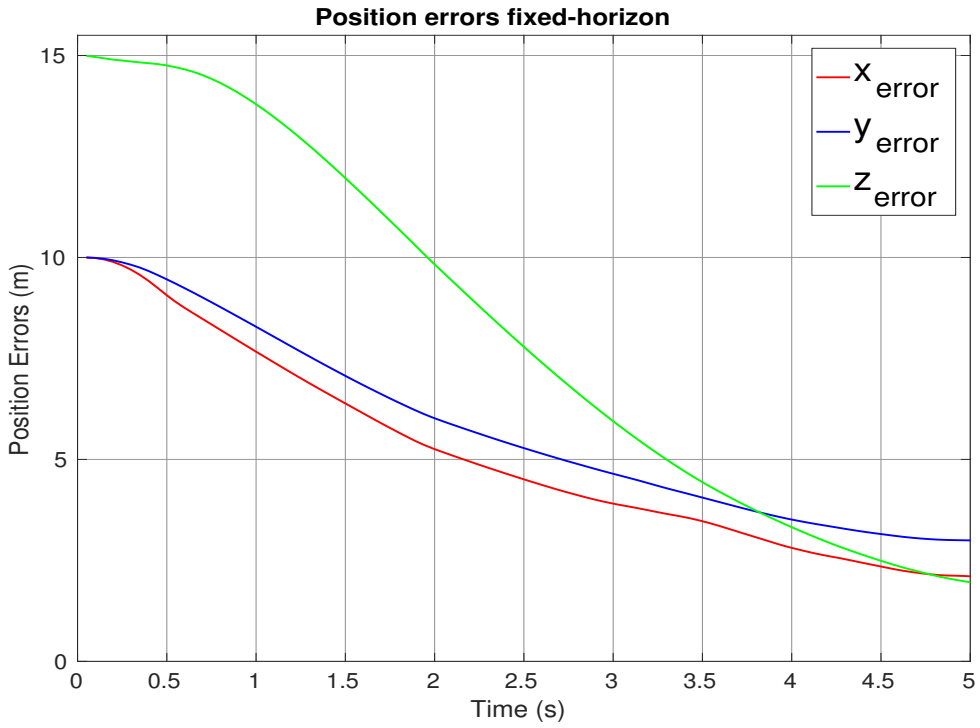

Figure 4.2: Position error when a fixed-horizon MPC is employed. In order for the trajectory towards the target to end at a predefined horizon, the system reduces its speed and hovers in place unable to complete the maneuver and reach the first waypoint.

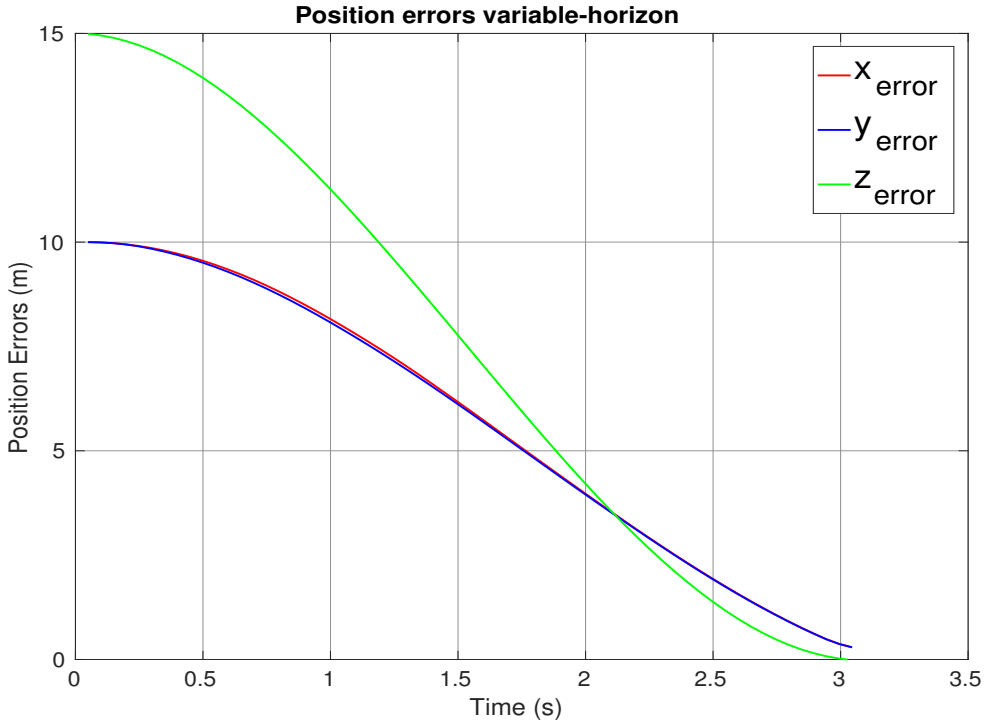

Figure 4.3: Position error with respect to the first desired waypoint employing the proposed variable-horizon MPC of this work. Under the proposed scheme, the system is lead successfully to the first desired waypoint.

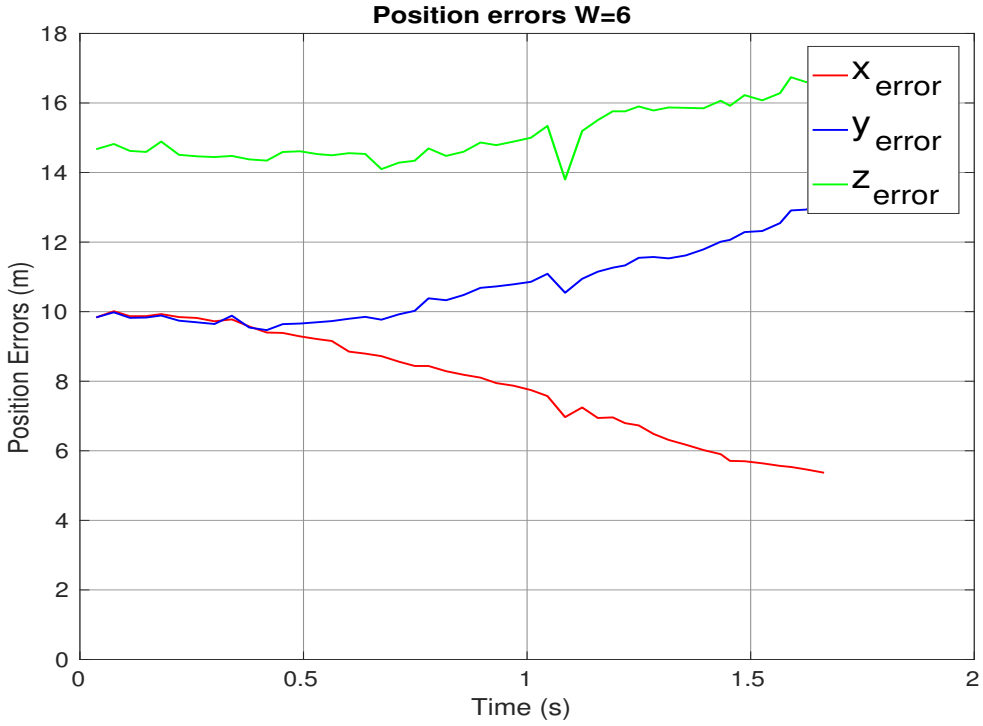

Figure 4.4: Position error with a disturbance greater than the computed theoretical bound according to Theorem 1. The optimizer is unable to find a feasible solution to the problem, and the algorithm fails to guide the quadrotor to the target.

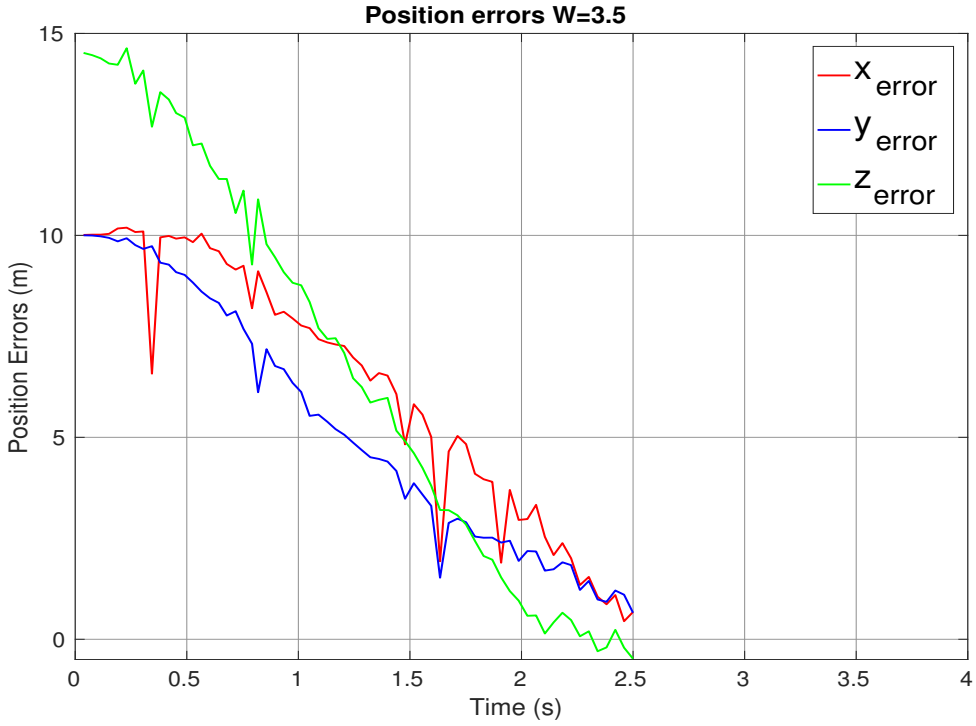

Figure 4.5: Position error with a disturbance smaller that the theoretical bound. The system is kept within a feasible region and the algorithm is able to guide the quadrotor to the target.

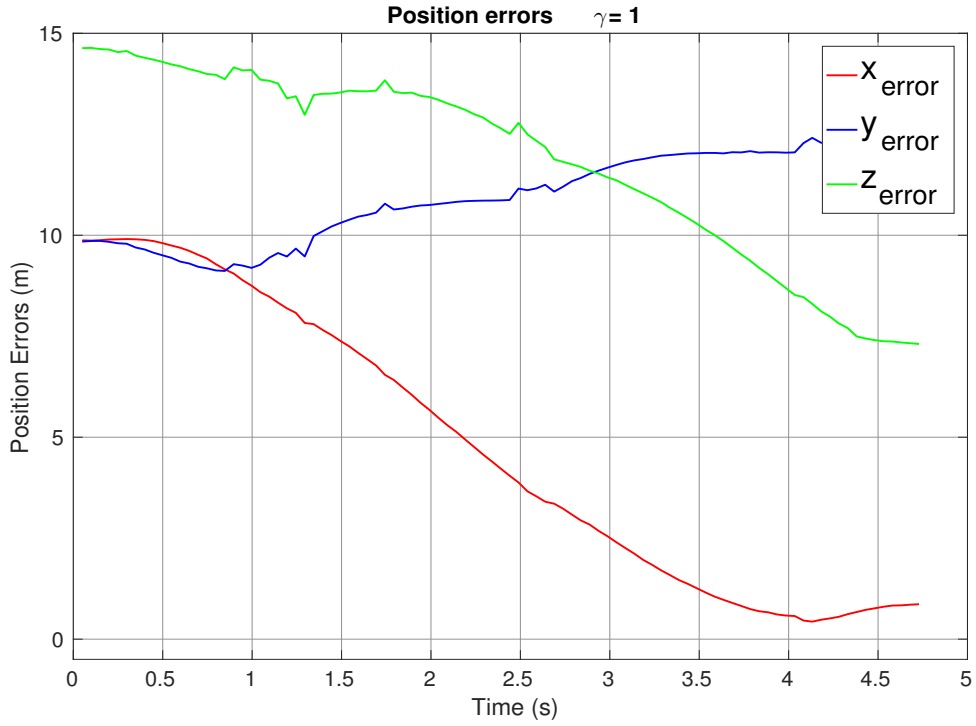

Figure 4.6: Position error when the optimization is equal parts energy-optimal and time-optimal. Although the optimization algorithm is able to produce feasible optimal trajectories, the disturbances prevent the system from reaching the target, i.e. the optimal cost is non-decreasing at each step.

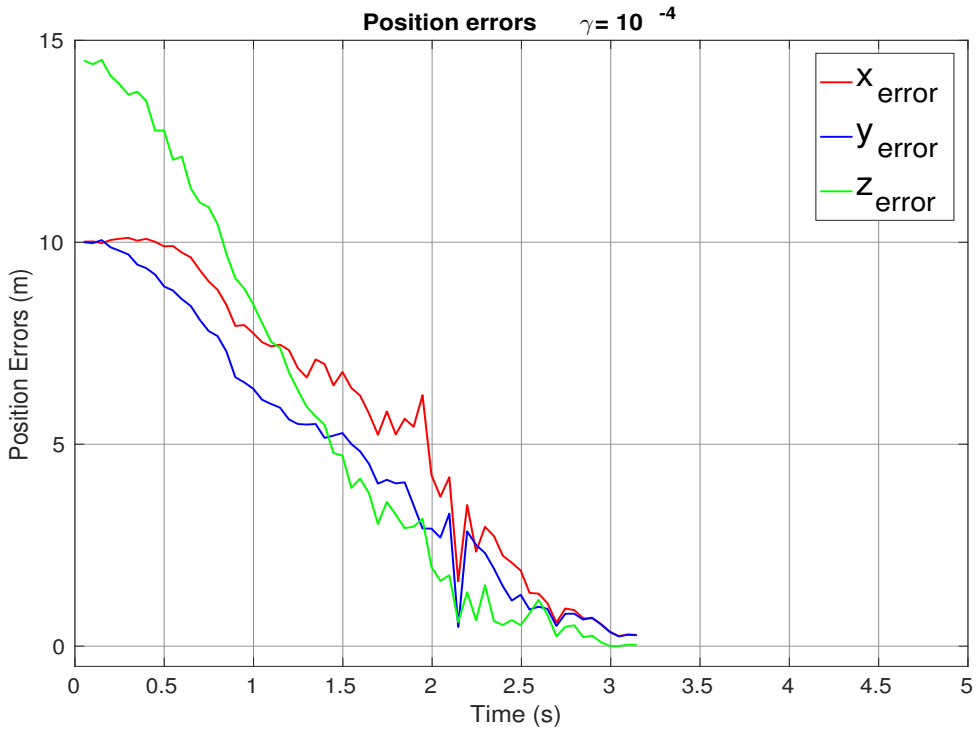

Figure 4.7: Position error when the optimization is primarily time-optimal. The control effort produced by the optimization algorithm is enough to surpass the disturbance and guide the quadrotor to the target. Consequently, the optimal cost is decreasing at each step.

Chapter 5

Conclusion and future work

5.1 Conclusion

In this thesis, we investigated the problem of traversing a series of points in the Cartesian space for a quadrotor UAV. Our work focused in two research directions. Initially, we focused on the problem of trajectory optimization. Given the kinematic/dynamic model of the vehicle, we utilized theorems from optimal control to derive the first order conditions that describe the solution. Due to issues inherent to the classic formulation of optimal control problems, we turned to the newer framework of Pseudospectral Optimal Control. This way, we presented the solution to the original problem.

Subsequently, our focus shifted to the closed loop implementation of the optimizer, which would enable us to account for external disturbances encountered in real life applications. In that respect, the framework of Model Predictive Control was chosen. After extensive literature review, it became apparent that there were gaps in the mathematical foundation of MPC. While experimental work in the usage of MPC schemes to vehicle maneuvering problems-problems where our goal is not to regulate a system around an equilibrium point, but rather to guide it to some set-has been presented before, feasibility, convergence and robustness guarantees have been provided only for linear systems. As a result, manual tuning of the different gains in the optimal cost was necessary. In this work, we provide bounds on the admissible disturbances as well as on a single parameter that guarantees completion of the problem.

5.2 Future Work

The work conducted in this thesis could be extended in a number of different directions. First of all, the framework of Pseudospectral Optimal Control is to this day a very active field in trajectory optimization and optimal control. Computational improvements on the optimizer, which may include the use of intermediate sub-optimal solutions, the combination of indirect and direct methods etc, could decrease the computational burden of the method dramatically.

Further research in Variable-Horizon Model Predictive Control could lead to improvements on the completion guarantees, allowing us to consider more general terminal constraints. Also, via the use of reachability theorems, we could include obstacles in the paths. Additionally, better approximations on the different constants of the system used in the theorems developed here, could lead to easier computation of the various sets. Furthermore, upon completing a working framework of guarantees for the VH-MPC, combining it with the work on Event Triggered MPC would be straightforward.

Another extension to the problem would be to allow for the sequence of interior points to be subject to optimization. Preliminary research we conducted in that direction pointed us towards the formulation of Mixed Integer Nonlinear Programming problems, or approximations via Motorized Travelling Salesman paradigms.

Bibliography

[1] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 46–56, 2012.

[2] M. Fumagalli, R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli, and L. Marconi, "Modeling and control of a flying robot for contact inspection," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 3532–3537.

[3] S. Kim, S. Choi, and H. J. Kim, "Aerial manipulation using a quadrotor with a two dof robotic arm," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4990–4995.

[4] S. Karaman and E. Frazzoli, "Optimal kinodynamic motion planning using incremental sampling-based methods," in *49th IEEE conference on decision and control (CDC)*. IEEE, 2010, pp. 7681–7687.

[5] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[6] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos, "Nonholonomic navigation and control of cooperating mobile manipulators," *IEEE Transactions on robotics and automation*, vol. 19, no. 1, pp. 53–64, 2003.

[7] T. Bresciani, "Modelling, identification and control of a quadrotor helicopter," *MSc Theses*, 2008.

[8] A. E. Bryson, *Applied optimal control: optimization, estimation and control*. CRC Press, 1975.

[9] J. T. Betts, *Practical methods for optimal control and estimation using nonlinear programming*. SIAM, 2010.

[10] I. M. Ross and M. Karpenko, "A review of pseudospectral optimal control: from theory to flight," *Annual Reviews in Control*, vol. 36, no. 2, pp. 182–197, 2012.

[11] I. M. Ross and F. Fahroo, "Issues in the real-time computation of optimal control," *Mathematical and computer modelling*, vol. 43, no. 9, pp. 1172–1188, 2006.

[12] ——, "A perspective on methods for trajectory optimization," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2002, p. 4727.

[13] V. M. Becerra, "Solving complex optimal control problems at no cost with psopt," in *2010 IEEE International Symposium on Computer-Aided Control System Design*. IEEE, 2010, pp. 1391–1396.

[14] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practice—a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[15] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Novel event-triggered strategies for model predictive controllers," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 3392–3397.

[16] 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.

[17] S. J. Qin and T. A. Badgwell, "An overview of nonlinear model predictive control applications," in *Nonlinear model predictive control*. Springer, 2000, pp. 369–392.

[18] P. Vlantis, P. Marantos, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Quadrotor landing on an inclined platform of a moving ground vehicle," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 2202–2207.

[19] A. Richards and J. P. How, "Robust variable horizon model predictive control for vehicle maneuvering," *International Journal of Robust and Nonlinear Control*, vol. 16, no. 7, pp. 333–351, 2006.

[20] R. C. Shekhar, M. Kearney, and I. Shames, "Robust model predictive control of unmanned aerial vehicles using waysets," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 10, pp. 1898–1907, 2015.

[21] S. Heshmati-alamdari, G. K. Karavas, A. Eqtami, M. Drossakis, and K. J. Kyriakopoulos, "Robustness analysis of model predictive control for constrained

image-based visual servoing," *IEEE International Conference on Robotics and Automation*, pp. 4469–4474, 2014.

[22] V. Veliov, "Lipschitz continuity of the value function in optimal control," *Journal of optimization theory and applications*, vol. 94, no. 2, pp. 335–363, 1997.

[23] D. Burago, Y. Burago, and S. Ivanov, *A course in metric geometry*, vol. 33.

[24] J. P. How, B. BEHIHKE, A. Frank, D. Dale, and J. Vian, "Real-time indoor autonomous vehicle test environment," *IEEE control systems*, vol. 28, no. 2, pp. 51–64, 2008.