AUTONOMOUS SPACECRAFT RENDEZVOUS AND DOCKING ON SAFE TRAJECTORIES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ALİ TEVFİK BÜYÜKKOÇAK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
AEROSPACE ENGINEERING

SEPTEMBER 2018

Approval of the thesis:

AUTONOMOUS SPACECRAFT RENDEZVOUS AND DOCKING ON SAFE TRAJECTORIES

submitted by **ALİ TEVFİK BÜYÜKKOÇAK** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences**   _____

Prof. Dr. Ozan Tekinalp
Head of Department, **Aerospace Engineering**   _____

Prof. Dr. Ozan Tekinalp
Supervisor, **Aerospace Engineering Dept., METU**   _____

Examining Committee Members:

Asst. Prof. Dr. Ali Türker Kutay
Aerospace Engineering Dept., METU   _____

Prof. Dr. Ozan Tekinalp
Aerospace Engineering Dept., METU   _____

Prof. Dr. Kemal Özgören
Mechanical Engineering Dept., METU   _____

Prof. Dr. Metin Uymaz Salamcı
Mechanical Engineering Dept., Gazi University   _____

Prof. Dr. Coşku Kasnakoğlu
Electrical and Electronics Engineering Dept., TOBB ETU   _____

Date: 07.09.2018

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name   : Ali Tevfik Büyükkoçak

Signature   :

ABSTRACT

AUTONOMOUS SPACECRAFT RENDEZVOUS AND DOCKING ON SAFE TRAJECTORIES

Büyükkoçak, Ali Tevfik
M.S., Department of Aerospace Engineering
Supervisor   : Prof. Dr. Ozan Tekinalp

September 2018, 92 pages

In this thesis, rendezvous and docking operation of a pair of low earth orbit spacecraft is addressed. Two different sets of equations for the nonlinear orbital relative motion of spacecraft are derived and simulation codes for this motion are developed. First, Hill-Clohessy-Wiltshire (HCW) equations are used in chaser-target spacecraft configuration with Model Predictive Control (MPC) algorithm including some safety considerations such as debris avoidance, direction of approach constraint and slow impact requirement. The HCW equations are linearized assuming a circular orbit, and used in MPC algorithm. All authority is given to the chaser spacecraft, and the target is kept passive. Parametric studies are implemented for different cases with several constraint combinations. According to these studies, best planning horizon length and optimal weighting parameter are selected for each case. The safe trajectory generated by MPC approach, which avoids a relatively moving debris represented as an obstacle, is tracked by a novel Lyapunov based control algorithm as well. The algorithm is based on dual quaternions for the motion parametrization and provides a combined control of both translational and rotational motion. Another set of relative motion dynamics including combined attitude and position is derived. An error dual quaternion and its derivative are generated from desired attitude and position information. While desired attitude trajectory is a time-dependent polynomial function, the reference position

trajectory is retrieved from MPC plan. Two control approaches are compared, and effectiveness of dual quaternion based control approach is demonstrated.

Keywords: Rendezvous and Docking, Model Predictive Control, Obstacle Avoidance, Constrained Optimization, Dual Quaternion Parametrization Based Attitude and Position Control

ÖZ

UZAY ARACININ GÜVENLİ YÖRÜNGELERDE OTONOM OLARAK YAKLAŞARAK KENETLENMESİ

Büyükkoçak, Ali Tevfik
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü
Tez Yöneticisi   : Prof. Dr. Ozan Tekinalp

Eylül 2018, 92 sayfa

Bu tezde, bir çift uzay aracının alçak yörüngedeki randevu ve kenetlenme problemi üzerine çalışılmıştır. İki farklı doğrusal olmayan yörüngedeki bağıl hareket denklem takımı elde edilmiş ve benzetim algoritmaları bu takımlarla geliştirilmiştir. İlk olarak, Hill-Clohessy-Wiltshire (HCW) denklemleri, yakalayıcı-hedef uzay aracı konfigürasyonunda tanımlanmıştır. Dairesel yörünge varsayımıyla doğrusallaştırılan bu denklemler Model Öngörülü Kontrol (MÖK) yaklaşımına dahil edilerek bazı güvenlik ve operasyon kısıtlarıyla birlikte randevu kontrolünde kullanılmışlardır. Bu kısıtlara örnek olarak randevu süresince karşılaşılan engellerden kaçınma, hedefe belirlenen bir doğrultuda yaklaşma ya da hedefe düşük hızlarda yaklaşarak, çarpmayı önleme verilebilir. Uygulamalarda sadece yakalayıcı araç kontrol edilmiş olup hedef pasif tutulmuştur. Farklı kısıtlar içeren dört farklı senaryo için parametrik çalışmalar yapılmıştır. Bu çalışmalar sonucunda en uygun planlama ufuk uzunluğu ve ağırlık parametresi her bir senaryo için seçilmiştir. Yörüngedeki pozisyon kontrolü kadar yönelim kontrolü de önem arz etmektedir. Bu sebeple hem pozisyonu hem de yönelimi beraber göstermeye yarayan İkili dörtleç parametrizasyonu yöntemi ile birlikte Lyapunov tabanlı doğrusal olmayan bir kontrol algoritması geliştirilmiştir. MÖK yaklaşımıyla elde edilen hareketli bir engelden kaçınma senaryolu randevu yörüngesi, yeni geliştirilen algoritmaya referans olarak verilip her iki yaklaşım da pozisyon kontrolü özelinde doğrulanmıştır. Bu yöntem için pozisyon ve yönelim bilgisini

birlikte içeren HCW denklemlerinden farklı bir bağıl hareket denklem takımı geliştirilmiş olup böylece her iki yaklaşımdaki benzetim kodları da doğrulanmıştır. Referans takibindeki hatayı ölçen bir İkili dörtleç ve onun türevi elde edilmiş ve kontrol algoritmasına dahil edilmiştir. Son olarak iki yaklaşım pozisyon kontrolü özelinde karşılaştırılmış ve yönelim kontrolünün pozisyon kontrolüne etkisi araştırılmıştır.

Anahtar Kelimeler: Randevu ve Kenetlenme, Model Öngörülü Kontrol, Engelden Kaçınma, Kısıtlı Optimizasyon, İkili Dörtleç Parametrizasyonu Tabanlı Yönelim ve Yörünge Kontrolü

To My Lovely Wife

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

MPC	Model Predictive Control
LQR	Linear Quadratic Regulator
HCW	Hill-Clohessy-Wiltshire
R-D	Rendezvous and Docking
QP	Quadratic Programming
LOS	Line of Sight
LTI	Linear Time Invariant
LTV	Linear Time Variant
LPV	Linear Parameter Variant
GPS	Global Positioning System
ISS	International Space Station
LVLH	Local Vertical Local Horizontal
PD	Proportional Derivative
LEO	Low Earth Orbit
DARE	Discrete Algebraic Riccati Equation
DQ	Dual Quaternion

LIST OF SYMBOLS

F	Force
m	Mass
a	Acceleration
G	Gravitational constant
r	Relative distance
μ	Gravitational parameter
x, y, z	Relative positions
h	Altitude
R_0	Total distance to earth center
e	Eccentricity
n	Orbital angular velocity
u_x, u_y, u_z	Inputs in LVLH frame
T_s	Time step or Sampling time
N	Planning horizon
J	Cost function
σ, β	Slow impact constraint shape factors
$\mathbf{1}_{n \times n}$	Matrix of ones
φ	Conical half angle of approach constraint shape
r_{debris}	Radius of circular obstacle
γ	Angle between chaser and obstacle
$x_{debris}, y_{debris}, z_{debris}$	Debris or obstacle positions in LVLH frame
$r_{avoidance}$	Radius of circle avoided until reaching approach cone
r_0	Initial relative position
I_n	n^{th} order identity matrix
$Q, R, \bar{Q}$	State, input, and terminal weight matrices respectively
α	Weighting parameter
$t_{docking}$	Docking time

r_I , r_B	Displacements in inertial and body-fixed frames respectively
w	Dual velocity
ω, v	Rotational and translational velocities respectively
J	Inertia
V	Lyapunov candidate function
K_p, K_d	Gain matrices
ω_{n_m} , ω_{n_J}	Natural frequencies of translational and rotational motion
ξ	Damping ratio
η	Attitude quaternion scalar part sign parameter

CHAPTER 1

INTRODUCTION

Main theme of this thesis is the development of control methods that yield safe and efficient motion planning and application of these plans successfully for spacecraft rendezvous and docking (R-D). This chapter creates a background about the work by giving an overview of the spacecraft rendezvous and docking problem, and summarizing the contributions made in this work.

In Section 1.1, spacecraft R-D problem is defined, and the critical aspects in this problem are underlined. Moreover, the motivation behind to make this process autonomous is explained. In Section 1.2, the studies previously conducted on this topic are reviewed. Section 1.3 states the original contributions made in this thesis. The chapter is concluded with Section 1.4 that presents the organization of the thesis.

1.1 Motivation

Rendezvous and docking missions become very frequent including several new mission types. In the future, a variety of new types of missions are expected to be implemented. These missions require a continuing demand of operations which enable more effective use of space and improve conditions of space assets in orbit. Rendezvous and docking may be mentioned as such operations in the first place. New missions to demonstrate space inspecting and servicing, debris removal, sample collecting, on-orbit assembly, and many other tasks are expected to be carried out in the coming years. To illustrate, a satellite running out of fuel, it must stop its operations unless it is refueled. Refueling satellites which are out of fuel is an option that requires

a proper R-D operation. However, since it is hard to transfer fuel to satellites which are not designed for refueling operations, missions are focused on moving these dying satellites to invaluable orbits [1], [2]. Because, if it is not possible to refuel such a satellite due to its design or capabilities, it is inevitable for it to become a space junk. At this point, again a R-D mission may be helpful to clean orbit from the space junks. Consequently, as access to space improves and mission frequency increases due to space commercialization, number of the missions with R-D operation is expected to grow significantly.

Not only rendezvous and docking but also formation flight research become very popular since it is believed to reduce costs and provide opportunities in new space applications [3]. Examples of these applications may be given as observation satellite constellations, distributed small satellites for communication infrastructure or earth mapping constellations.

The space adventure has gained a great acceleration with the travel to the moon in the 20th century. However, the residuals of these fascinating discoveries and new space missions undermine them. Space missions leave behind junk named as space debris, and the amount of this junk pile grows day by day. Earth orbits, especially the lower ones, have a huge amount of space objects moving with very high velocities. A collision with these objects may destroy space assets. There are several reasons for the increase in the possibility of space debris threat. With the space journeys starting from the 1950's, the rockets used to reach out of the atmosphere are only designed for the early stages of the mission. The outcomes of leaving rocket stages in orbit with the fuel and the pressure inside are never considered. Since the explosions of these rockets that may occur in orbits threaten the orbital health, next missions conducted with the rockets which evacuate the fuel and pressure inside after the mission.

Even very small particles can be dangerous due to their high speeds. Especially it takes longer for the debris in higher orbits to burn down in the atmosphere. The higher amount of space debris constitutes a higher possibility of a crash.

The space debris accumulating in valuable orbits make operations in these orbits very expensive. Planning and application of obstacle avoidance maneuvers require a significant amount of human power and fuel consumption. Although small obstacles can be stopped with shield-like structures, to avoid bigger objects, it is an obligation to implement avoidance maneuvers.

Although the most effective precaution is to prevent creating more and new space debris, clearing the orbits from them became an essential need for orbit health. Unless the precautions are taken, it is unavoidable for some orbits to lose their functionality.

In debris removal operations, targets are non-collaborating and usually rotating and wobbling around an axis. They also do not have a port to dock, and to clean the orbit from such objects, different approaches to capture should be evaluated. On the other hand, servicing and maintenance of operational spacecraft may be listed in the scope of R-D missions with cooperative targets. In such cases, both spacecraft are functional and, target has stable dynamics. Also, its flight data can be used, and this makes it collaborative or cooperative. For this kind of operations, it is required to approach the target spacecraft through its docking port or designated axis for berthing.

Rendezvous problem constitutes of relative motion dynamics, guidance, navigation and control of the spacecraft. In this thesis, the guidance and control problem along with the dynamics are handled.

Autonomy in space applications is desired because it is believed to allow higher mission frequency and reliability besides robustness. Possible scheduling conflicts and increase in operational costs are unavoidable due to the increasing number of missions in the space and its growing commercialization. This situation may cause an increase in the possibility of human errors as well. It is evaluated that automation can prevent such outcomes, and make space operations more efficient, by enabling higher numbers and types of missions in addition to improving robustness and reducing risk [4], [5].

For this type of autonomous missions, it is critical to understand the dynamics of rendezvous and docking operation for both mission planners and operators in addition

to engineers designing the computer algorithms that achieve the mission autonomously.

Rendezvous problem may be considered as the trajectory control of two point masses. In the literature, it is generally studied that one vehicle (chaser) is to be actively controlled, and the other (target) is kept passive like in rendezvous with a space station or a Mars Sample Return capture scenario [6]. In a spacecraft formation flying or R-D mission design, it is generally important to control the relative states of spacecraft rather than their absolute states. In addition, knowledge of the relative states of spacecraft in a formation is often far more accurate than the knowledge of the formation's absolute state. So, the problem may be reduced to control of relative motion between two objects. This requires the generation of a comprehensive relative motion model. The most known model of this relative motion is the Hill-Clohessy-Wiltshire equations. These equations constitute an LTI system, and they are derived for circular Keplerian orbits. In literature different models such as LTI, LTV and LPV models are implemented. However, for low Earth orbits that are designated orbits for debris removal operations, HCW is enough to model relative motion since near-circular orbits are the case. Linearized HCW equations written in Hill frame, i.e., the LVLH frame may be used in state space form in controller design. Since these equations are decoupled as in-plane and out of plane, motion in two planes may be treated separately.

In spacecraft control, decreasing fuel usage is the principal aim. Since attitude control torques are supplied by the reaction wheels, which take its energy from solar panels, it is not expected to limit the lifetime of a mission. As a result, trajectory or translational motion control has the highest priority especially for the rendezvous mission, and it should be fuel optimal. The need of optimality arises from the need of optimization with constraints, i.e., constrained optimization, to include safety and operational requirements into the problem. To illustrate, the chaser should approach its target within a safe trajectory and avoid any possible danger.

In the rendezvous phase, maneuvers with large durations and distances are implemented. This makes control methodologies focus on fuel efficiency in the

operation. When considered from this point of view, guidance, control and motion planning techniques requiring higher computational time but more fuel efficient plans may be applicable.

Not only time and fuel efficiency are required in rendezvous and docking operations which has a significant increase in number of applications, but also safe and robust algorithms. MPC makes it possible to solve the control problem as an optimization problem and its handling constraints systematically by including them to the problem enables safer plans. Unlike the other optimal control methods, it considers predicted future states inside its cost function to be minimized in a planning horizon. It is a discrete-time controller; therefore, it is required to discretize dynamic model. At each time step, it predicts the states within the prediction horizon and then builds a cost function including these predictions. Solution of minimization problem will be the stack of inputs that need to be applied at each time step in the control horizon. Between two time steps, the first optimal input of the solution will be applied, and at next step, the same calculations will be implemented for the same planning horizon. This is why MPC has another name of Receding Horizon Control.

In such an application, the problem has to be solved in real-time, i.e., within the sampling interval of the system, and with available hardware. Hardware selection is important because this application requires a high computational effort. This is why it is very suitable for space applications which constitutes a considerably slow environment, and in long-distance maneuvers on orbit, time steps can be taken as long as one minute. Indeed, constraints can be updated online which gives MPC an adaptive way of controlling. Contrary to fixed-gain control laws which require high control expenses, it is economical to use an online self-adapting control law.

Online optimization used in MPC approach gives an opportunity to handle time-varying or invariant constraints systematically. In rendezvous and docking operation there may be several possible constraints regarding safety and operational needs. To set an example, for cooperative spacecrafts, it is essential to approach in the field of view of sensors of the target and through docking port or for tumbling uncooperative objects chaser should approach through the axis of rotation. These may be achieved

by constraining the direction of approach of the chaser satellite. Another example may be given for scientific sample collection missions in which plume impingement to the sample may need to be avoided. Several other examples may be given, and some of them taken into account in this work may be seen in the following chapters.

Besides the trajectory control, in space applications controlling the orientation of the spacecraft is vital to sustain its operations effectively. First of all, most of the sensors and mission-related equipment require a stable orientation without any spinning or tumbling. To illustrate, for observatory satellites, it is required to keep camera direction on earth. Moreover, most of the satellites use solar panels to generate electricity during their operations. Consequently, orientation must be kept accordingly to get sunshine on the panels. One another mission type that requires precise attitude control is the R-D operation. Although for rendezvous phase, nominal attitude trajectories may be followed, in the docking phase controlling attitude considering target orientation may pose a challenging task, especially for the missions including uncooperative and tumbling or spinning targets. In addition, controlling attitude aggressively may affect position control since the same thrusters may be used. As a result, attitude or rotational motion should be controlled without losing accuracy in the translational motion control.

1.2   Literature Survey

R-D and formation flight of spacecraft are widely investigated problems in the literature. Some survey papers enlighten the experiences gained and methods considered to solve these problems [7], [8], [9], [10], [11], [12], [13], [14]. Literature which focus on spacecraft R-D control is comprehensive. Works in [15], [16], [17], [18], [19], [20], [21], [22], [23], and [24] with the references therein may be given as examples of the studies emphasize the control and guidance methods.

George William Hill was the first person who published a set of equations to represent the motion of the Moon relative to Earth in 1878 [25]. Hill's work constituted a base for Clohessy and Wiltshire's study in which equations derived by Hill are used to

define the relative translational motion for rendezvous and docking of two objects in orbit [16].

In [26], some cases include eccentric orbits. For these cases, target and chaser vehicles are modeled separately using general equations for Keplerian orbits given in [27], and their difference is counted as relative motion between them.

To determine relative motion, a number of approaches has been proposed. For example in Reference [28] by concurrently running Kalman filters on onboard computers of both satellites transmitting raw measurements to each other, states of satellites in the formation and the relative states of them are estimated. To do that, each satellite in the formation are equipped with a GPS receiver. It is shown in this study that in a formation up to four satellites from the distance of 4 *km*, 1.5 *mm* and 5 *μm/s* sensitivity can be achieved in position and velocity estimations respectively. As a result, sufficient information may be obtained to make rendezvous and docking operation autonomously based on the information gained from sensors.

Several control applications are considered in [29] for formation flying in leader-follower configuration. This formation flight configuration is very similar to rendezvous and docking operation. While in formation flight it is required to control relative distance according to formation necessities, in R-D this distance is tried to be set to zero. In the study, PD, Lyapunov based and integrator backstepping control techniques are shown to be succeeded in stable results in the formation control.

Another method proposed uses artificial potential functions for trajectory optimization [30], [31]. It depends on the definition of a potential function of which minimum exists at the desired relative position. Moreover, an obstacle-free rendezvous path is guaranteed by a dynamically reconfigurable control law which is guided by this artificial potential function method.

Many MPC approaches for relative motion control may be found in the literature. MPC controller uses linear relative motion model with different type of constraints, such as linear or quadratic, in an optimization problem which is solved to generate a control sequence over a planning horizon. In each step, the first sequence of generated input

needs to be applied, and in the next step all process is repeated with the receding horizon. In [17] and [32], variable horizon MPC is applied that solves a mixed-integer linear programming problem at each step. In [33], the spacecraft guidance problem in close proximity of ISS is solved with unconstrained MPC application. For attitude control, another controller constrained to keep ISS on LOS of the sensors onboard is used. In the same work, a control allocation scheme is also presented for thrust firing.

Again for safe, close proximity operations, in [34] MPC approach which solves a quadratic programming problem with quadratic constraints is proposed. While the problem presented becomes a non-convex optimization problem, the robustness of controller is improved by statistically modeling of positional uncertainty.

There is indeed an operation named as PRISMA in which MPC was employed and tried successfully in close proximity space operations. This shows the applicability of MPC methodology not only in simulations but also in real operations [35], [36], [37].

As the space debris get accumulated in high-valued orbits such as LEO's, a significant number of works are dedicated to find methods for the solution of debris removal problem by spacecraft designed to make rendezvous with the debris and capture them. Studies in [38], [39], and [40] may be given as examples of works in this context.

In robotics research area, obstacle avoidance problem is common and widely investigated [41], [42]. Obstacle avoidance application in spacecraft trajectory optimization is also addressed in the literature. In [43], a nonlinear optimal control methodology is used to implement a rendezvous operation with minimum fuel usage. Collision avoidance is also added to the optimization problem as inequality constraints in addition to other operational constraints. A method is proposed in [31] to optimize time to dock and terminal relative position which guarantees to avoid any obstacles intersected during the flight. Another work states possible obstacle regions may be defined as avoidance regions which are kept out during the flight, and flight path is constrained accordingly [44].

Orbital obstacle avoidance is also achieved by defining the problem as an optimization problem and using linear programming techniques to solve it. Optimal maneuver

number is found in [45] by the solution of trajectory optimization problem using linear programming with added constraints. In [46], obstacle avoidance is modelled as dynamically reconfigurable linear constraints, and fuel efficient maneuvers are implemented. Mixed-integer linear programming technique is used in [47] to solve trajectory optimization problem with obstacle avoidance requirement.

In [48], the complicated motion of tumbling objects or satellites with changing attitudes is addressed and difficulty to approach them is pointed out. For the case of LOS cone attached to the target which means that chaser must be kept in LOS of sensors placed on the target, changes in attitude of target should be taken into account since the LOS orientation changes as well with respect to the LVLH frame. Therefore, appropriate transformations should be performed. The same work proposes that an adaptive way of controlling may be achieved in the optimization by defining "dynamically reconfigurable" constraints in the problem. This would enable a replannable trajectory generation and desired motion can be shaped via constraints.

By considering the above requirements, a linear quadratic MPC approach with dynamic constraints in optimization is developed. Due to its replanning nature, disturbances and last minute changes in path-tracking commands, which may be caused by obstacles or anomalies in target motion, can be handled by this method with real-time application and fuel efficient maneuvers in MPC approach.

The success of the methods for both non-rotating and rotating targets or tumbling objects while avoiding a debris placed along the spacecraft's trajectory is shown in the literature. The success of MPC in handling disturbances is also demonstrated. MPC problem with dynamic constraints may be reduced to quadratic programming problem which requires a reasonable computational cost to be solved, and it makes the approach amenable for onboard computers. Avoidance problems generally handled with debris or obstacles which have fixed positions relative to the target. However, obstacle position may be uncertain, and its relative motion should be taken into account.

1.3 Contributions

In this thesis, MPC methodology is used to achieve a safe rendezvous and docking trajectory in the presence of a relatively moving obstacle between the chaser and target. A method is proposed to handle combined state and input constraints including the moving debris constraint updated in real-time inside the optimization problem. The efficiency of moving obstacle consideration instead of obstacle bounds fixed in position that represent avoidance zones is revealed.

Although safety in R-D is mostly concerned with the translational motion, in such operations, the interaction between attitude and position control may pose a challenge to overcome. To examine this problem, a control algorithm is developed based on dual quaternion parametrization with combined attitude and position information. The algorithm generates a Lyapunov based control law that includes additional terms with desired state and its derivative. In this aspect, it differs from PD-like approaches.

1.4 Outline

This thesis is organized as the construction of two separate methodologies and comparison of them in the end.

In Chapter 2, rendezvous and docking problem in terms of the translational motion dynamics is defined. Derivation of motion in orbit starts with a two-body problem, and the nonlinear dynamics arises in the problem. Relative motion of two objects in orbit may be found from differences of their individual orbital motion. HCW equations are derived to represent relative orbital motion. With some assumptions, it is possible to define these relative dynamics as an LTI system for a circular orbit. Finally, proper discretization is implemented for control applications.

In Chapter 3, MPC methodology and constrained optimization are defined in detail. Receding horizon concept and optimization technique used are explained. In the presence of both state and input constraints, they should be combined to use only one

optimization variable in the optimization process which is the input in this case. In other words, constraints on states must be represented as constraints on inputs, and a proper transformation between them should be done. In addition, this constraint combination process is defined

In Chapter 4, safety and mission requirements are handled in motion planning. These requirements are applied as constraints in the optimization problem, and the plans, in which they are met, are generated accordingly. First, how mission requirements may be transformed into the optimization constraints is explained. Next, four cases which include different combinations of mission requirements are defined. Results for application of MPC to these cases and parametric studies implemented concludes the chapter including the comparison of MPC with a simpler LQR controller.

In Chapter 5, a novel combined attitude and position control algorithm is proposed. This algorithm uses dual-quaternions for parametrization. The concept of dual numbers and dual quaternions are presented briefly. For this nonlinear control algorithm, a new set of combined relative attitude and position equations of motion is developed. Finally, simulation results are compared with the MPC approach.

In Chapter 6, two different methods are compared for position control, and general results are discussed for each chapter. The thesis is concluded, and future directions of this work are discussed.

In the following chart, organization of the thesis may be seen with chapter contents. In brief, two different control approaches with different set of equations of motions are considered in parallel, and their results are compared in the end.
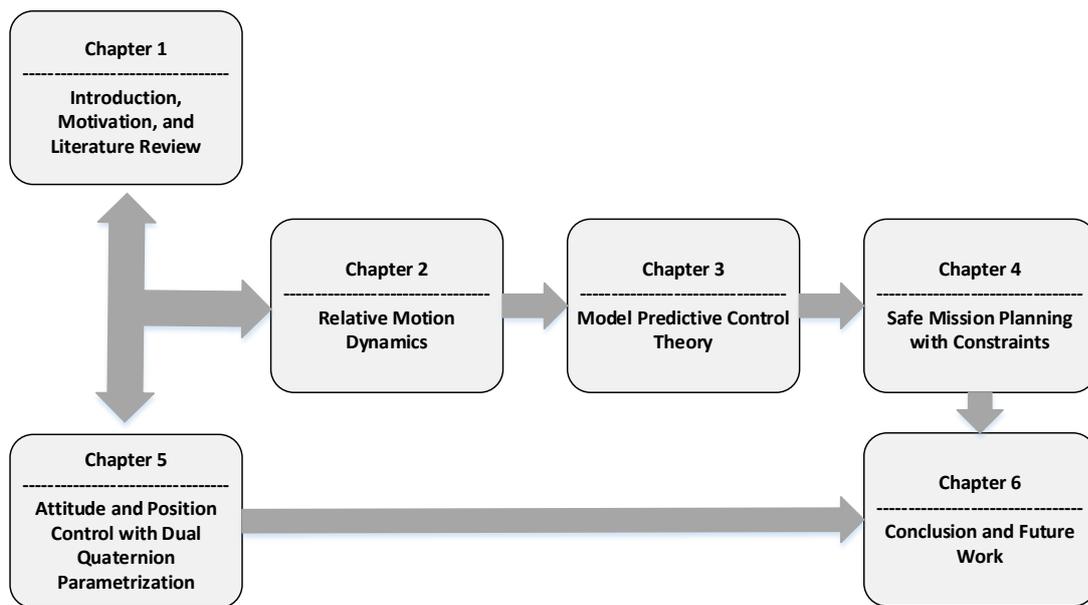
Figure 1.1: Flowchart of the thesis organization.

CHAPTER 2

RELATIVE MOTION DYNAMICS

This chapter describes the rendezvous and docking problem in terms of the translational dynamics. Relative motion dynamics that are required to simulate and control the motion of the chaser with respect to the target are investigated, and shown in detail.

In Section 2.1, the two-body problem that defines the motion of an object orbiting around a much bigger mass is presented.. In Section 2.2, equations of translational motion are derived. These nonlinear equations are linearized, and state space representation is obtained. Continuous state space matrices are discretized in the end. In Section 2.3, disturbances that may affect spacecraft operation are defined.

2.1 Two-Body Problem

The motion of satellites may be described with Newton's 2nd law of motion.

$$\sum F = ma \tag{2.1}$$

where F is the magnitude of force, m is the mass, and a is the total acceleration. When the two-body problem is considered as the attraction of two point masses, Newton's Law of Gravitation gives the amount of that attraction force,

$$F = G\frac{m_1 m_2}{r^2} \tag{2.2}$$

Here, G represents gravitational constant, m_1 and m_2 are the object masses, and $r = r_2 - r_1$ represents the distance between the objects shown in Figure 2.1.
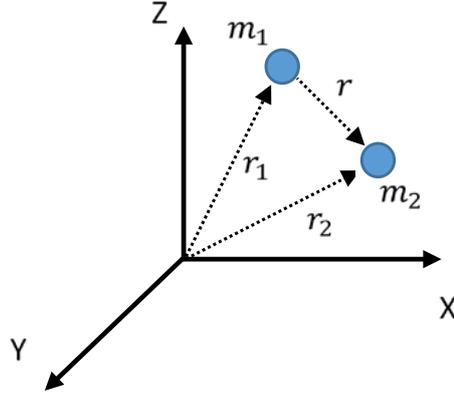
Figure 2.1: Two-body problem representation in an Inertial Frame.

To vectorize the forces, equations of motion belong to the masses may be written as,

$$m_1 \ddot{\vec{r}_1} = G \frac{m_1 m_2}{r^3} \vec{r} \tag{2.3}$$

$$m_2 \ddot{\vec{r}_2} = -G \frac{m_1 m_2}{r^3} \vec{r} \tag{2.4}$$

Here, double dots represent the second derivatives with respect to time. Relative acceleration of masses may be found by subtraction of Equation (2.3) from Equation (2.4).

$$\ddot{\vec{r}} = -G \frac{m_1 + m_2}{r^3} \vec{r} \tag{2.5}$$

or,

$$\ddot{\vec{r}} = -\frac{\mu}{r^3} \vec{r} \tag{2.6}$$

The gravitational parameter μ may be defined as $\mu = G(m_1 + m_2)$. In celestial body calculations, it is feasible to use the gravitational parameter of the body with much

bigger mass, i.e., $m_1 >> m_2$. For instance, in the calculations for bodies in low earth orbits (LEO), earth gravitational parameter is taken as $\mu_\oplus = 3.986004418 \times 10^{14} \, m^3 / s^2$, independent of flying body mass.

2.2   Orbital Relative Motion Model

Several mathematical models for spacecraft rendezvous may be found in the literature [49]. In this thesis, "target-chaser" configuration is used to define satellites on the formation. Relative motion between the chaser and target is represented in local vertical, local horizontal (LVLH) frame shown in Figure 2.2.

Where *x, y,* and *z* are the components of the chaser spacecraft position relative to the target in the LVLH frame. In this frame, the *x*-direction is radial, *y* is along-track, and *z* completes the orthogonal set.
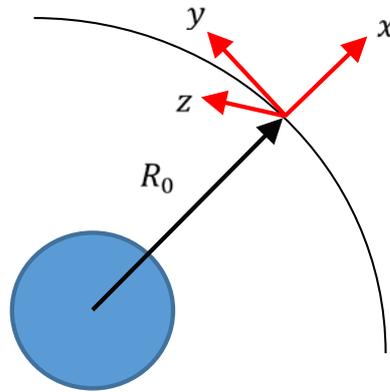
Figure 2.2: Local Vertical Local Horizontal (Hill) Frame.

Proposed orbit for the work in this thesis is a LEO, because debris removal missions are reasonable for valuable and crowded orbits. Indeed, LEO is the most possible orbit type for operational satellites on it to collide with the debris due to high demand in placing satellites there which in turn make these orbits the most crowded ones.

Table 2.1: Orbital parameters of the proposed orbit.

Name	Notation	Value	Unit
Altitude	h	550	[km]
Eccentricity	e	0	-
Orbital angular velocity	n	0.0011	[rad/s]

2.2.1 Nonlinear Dynamics

Chaser and target space vehicle configuration and the relativity between them presented in LVLH frame is shown in Figure 2.3.
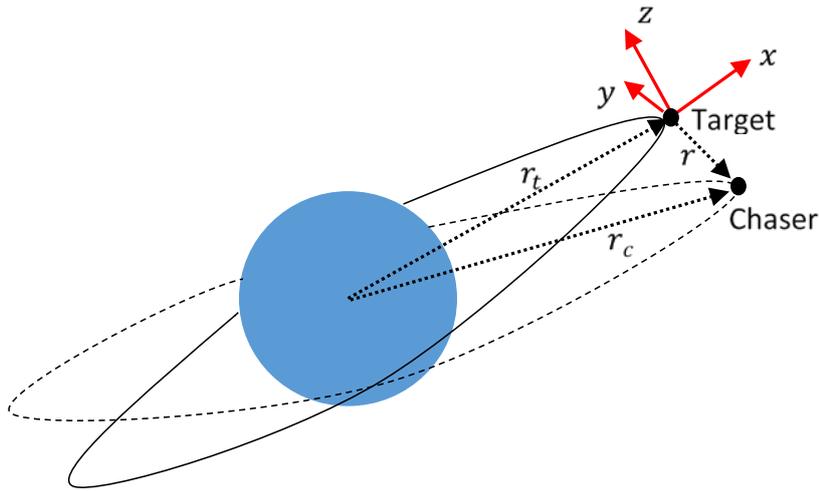
Figure 2.3: Chaser-Target spacecraft configuration represented on LVLH frame.

With the addition of control input as specific force, i.e., acceleration generated through thrusters to change chaser vehicle position, equations of motion of the chaser spacecraft represented in LVLH frame relative to Earth-Centered Inertial (ECI) frame, which has the x-direction on vernal equinox, z-direction on north pole, and y-direction completing the orthogonal set as lying on equatorial plane, are obtained as follows:

$$\frac{d^2\vec{r}_c}{dt^2} + \vec{u} = \frac{d^2\vec{r}_t}{dt^2} + \frac{d^2\vec{r}}{dt^2} + 2\left(\vec{n} \times \frac{d\vec{r}}{dt}\right) + \frac{d\vec{n}}{dt} \times \vec{r} + \vec{n} \times (\vec{n} \times \vec{r}) \qquad (2.7)$$

or,

$$-\frac{\mu}{r_c^3}\vec{r}_c + \vec{u} = -\frac{\mu}{r_t^3}\vec{r}_t + \frac{d^2\vec{r}}{dt^2} + 2\vec{n}\left(\vec{n} \times \frac{d\vec{r}}{dt}\right) + \frac{d\vec{n}}{dt} \times \vec{r} + \vec{n} \times (\vec{n} \times \vec{r}) \qquad (2.8)$$

where positions of target and chaser spacecraft together with the relative position and orbital angular velocity is given by the following:

$$\vec{r}_c = (r_t + x)\hat{i} + y\,\hat{j} + z\,\hat{k}$$
$$\vec{r}_t = r_t\,\hat{i} \qquad (2.9)$$
$$\vec{r} = x\hat{i} + y\,\hat{j} + z\,\hat{k}$$

Orbital angular velocity is given as $\vec{n} = n\hat{k}$ perpendicular to the orbital plane. As a result, vectorial representation of nonlinear equations of translational motion may be shown as below.

$$-\frac{\mu}{\left((r_t+x)^2 + y^2 + z^2\right)^{\frac{3}{2}}}\begin{bmatrix} r_t + x \\ y \\ z \end{bmatrix} + \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = -\frac{\mu}{r_t^3}\begin{bmatrix} r_t \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + 2\left(\begin{bmatrix} 0 \\ 0 \\ n \end{bmatrix} \times \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}\right)$$
$$+ \left(\begin{bmatrix} 0 \\ 0 \\ \dot{n} \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix}\right) + \begin{bmatrix} 0 \\ 0 \\ n \end{bmatrix} \times \left(\begin{bmatrix} 0 \\ 0 \\ n \end{bmatrix} \times \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}\right) \qquad (2.10)$$

In simulations, it is more realistic to use general nonlinear equations of the relative motion between chaser spacecraft and the passive target vehicle. These equations may be written for circular orbits replacing r_t by R_0 with Keplerian orbit assumption as the following:

$$\ddot{x} = 2n\dot{y} + n^2(R_0 + x) - \mu \frac{R_0 + x}{\left[\left(R_0 + x\right)^2 + y^2 + z^2\right]^{\frac{3}{2}}} + u_x$$

$$\ddot{y} = -2n\dot{x} + n^2 y - \mu \frac{y}{\left[\left(R_0 + x\right)^2 + y^2 + z^2\right]^{\frac{3}{2}}} + u_y \qquad (2.11)$$

$$\ddot{z} = -\mu \frac{z}{\left[\left(R_0 + x\right)^2 + y^2 + z^2\right]^{\frac{3}{2}}} + u_z$$

2.2.2 Linearization

In the above equations, only two-body gravitational equations of motion with no perturbations are considered. It is also assumed that the target is in a circular orbit about the Earth, and the relative distance between the target and the chaser is much smaller than the target's orbital radius. The above equations may be linearized around the origin of the LVLH frame as follows [26]:

$$\ddot{x} - 3n^2 x - 2n\dot{y} = u_x$$

$$\ddot{y} + 2n\dot{x} = u_y \qquad (2.12)$$

$$\ddot{z} + n^2 z = u_z$$

Where orbital mean angular velocity is [50]:

$$n = \sqrt{\frac{\mu}{R_0^3}} \qquad (2.13)$$

In general, Hill-Clohessy-Wiltshire equations are not very precise due to assumptions made to derive them. Especially for a target spacecraft moving in an eccentric orbit, other perturbations take place [51], [52]. However, for orbits such as LEO, feedback controllers are generally enough to compensate other effects.

18

2.2.3 State-Space Representation

In state space representation, the linearized HCW equations may be written as:

$$\dot{x} = Ax + Bu \tag{2.14}$$

Where $x \in \mathbb{R}^6$ is the state vector, and $u \in \mathbb{R}^3$ is the control input vector,

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.15}$$

$$x = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T \qquad u = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}^T \tag{2.16}$$

With the increasing distance between chaser and target spacecraft, the accuracy of the HCW equations decreases. On the other hand, for close proximity operations and feedback control applications, it is possible to use HCW equations without loss of generality.

2.2.4 Discretization

For discrete-time controllers, it is required to discretize the continuous state space representation of relative motion with appropriate sampling times that are chosen according to computational capacity. Discrete version of state space representation is presented as follows:

$$x(k+1) = A_d x(k) + B_d u(k) \tag{2.17}$$

Zero-order hold method is used to discretize the continuous state space model of relative translational motion given in Equation (2.14).

$$A_d = e^{A_c T_s}$$

$$B_d = B_c \int_0^{T_s} e^{A_c \tau} d\tau \tag{2.18}$$

The time between two consecutive steps, i.e., T_s is the sampling time of the discretization process. Explicit versions of discrete system and input matrices may be found in the literature widely as follows:

$$A_d = \begin{bmatrix}
4 - 3\cos(nT_s) & 0 & 0 & \frac{1}{n}\sin(nT_s) & \frac{2}{n}(1-\cos(nT_s)) & 0 \\
6(\sin(nT_s) - nT_s) & 1 & 0 & -\frac{2}{n}(1-\cos(nT_s)) & \frac{1}{n}(4\sin(nT_s) - 3nT_s) & 0 \\
0 & 0 & \cos(nT_s) & 0 & 0 & \frac{1}{n}\sin(nT_s) \\
3n\sin(nT_s) & 0 & 0 & \cos(nT_s) & 2\sin(nT_s) & 0 \\
-6n(1-\cos(nT_s)) & 0 & 0 & -2\sin(nT_s) & 4\cos(nT_s) - 3 & 0 \\
0 & 0 & -n\sin(nT_s) & 0 & 0 & \cos(nT_s)
\end{bmatrix} \tag{2.19}$$

$$B_d = \begin{bmatrix}
\frac{1}{n^2}(1-\cos(nT_s)) & \frac{2}{n}\left(T - \frac{1}{n}\sin(nT_s)\right) & 0 \\
\frac{2}{n^2}(\sin(nT_s) - nT_s) & -\frac{3}{2}T_s^2 + \frac{4}{n^2}(1-\cos(nT_s)) & 0 \\
0 & 0 & \frac{1}{n^2}(1-\cos(nT_s)) \\
\frac{1}{n}\sin(nT_s) & \frac{2}{n}(1-\cos(nT_s)) & 0 \\
\frac{2}{n}(\cos(nT_s) - 1) & -3T_s + \frac{4}{n}\sin(nT_s) & 0 \\
0 & 0 & \frac{1}{n}\sin(nT_s)
\end{bmatrix} \tag{2.20}$$

2.3 Disturbances

As mentioned previously, in addition to gravity and control forces, there are several disturbances that affect translational motion of earth orbiting space vehicles. These are more pronounced for eccentric orbits [51], [52]. Main disturbances on earth orbits may be mentioned as J_2-J_6 effects due to earth oblateness, 3rd body interactions such as moon's gravity, solar radiation pressure, and drag effect. These effects are widely investigated in the literature, and especially for LEO satellites, it is revealed that neglecting them does not cost much. Feedback control applications including MPC are generally enough to compensate them [48].

CHAPTER 3

MODEL PREDICTIVE CONTROL THEORY

This chapter discusses the main control theory applied in this thesis. Model Predictive Control (MPC) becomes popular in the literature for spacecraft applications. It takes the control problem as an optimization problem, and constraints on states or inputs can be added to the optimization problem itself. Its enabling to include future states in control methodology yields an effective and robust way of controlling.

In Section 3.1, MPC theory and receding horizon concept are explained. Mathematical backgrounds of MPC and optimization process are reviewed. In Section 3.2, constrained optimization with state and input constraints is discussed in detail. In Section 3.3, MPC methodology is examined in the sense of stability.

3.1   Control Algorithm

MPC is an optimal control problem in which an optimization in terms of mission parameters such as fuel, time, etc., is solved. In a mission, there may be several limitations which may constrain the states or inputs. Since these constraints limit the freedom in mission implementations, optimization problem should be designed such that the control approach is applicable for desired mission tasks.

MPC is a discrete-time controller. For each time step, it predicts the states within the prediction horizon, and then calculate the cost function to be minimized including these predictions in the same step.

MPC result gives a set of optimal inputs defined all over the planning horizon. In each time step first input among this set is applied. For the next time step, the same process is repeated with the same planning horizon length. In other words, this horizon shifts to the next step. For this reason, in literature name of "Receding Horizon Control" is widely used to refer MPC.
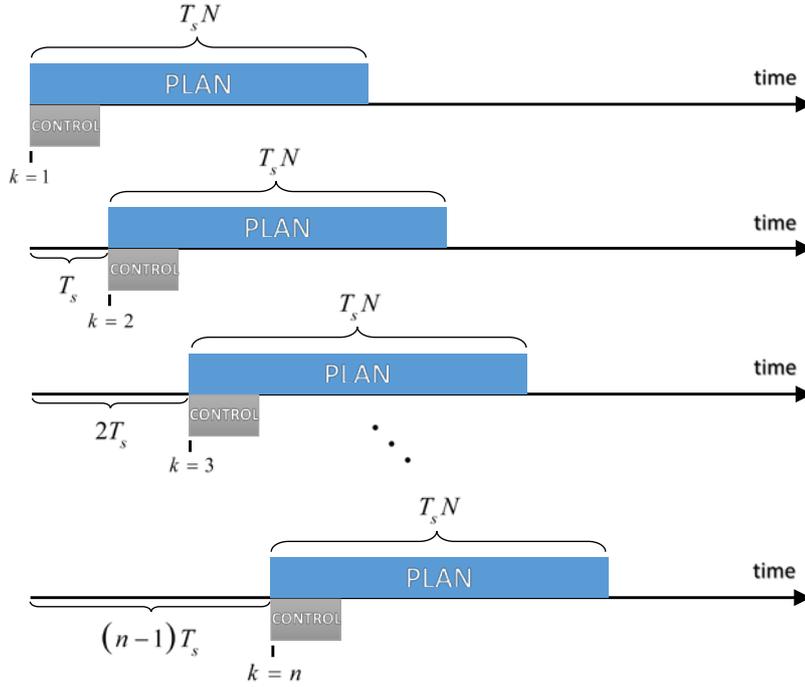
Figure 3.1: Receding Horizon Control approach.

Optimal control problem with constraints comprises a constrained optimization problem which needs to be solved numerically [9], [53]. This requires the solution of a complex optimization problem onboard in real-time with a proper discretization [54], [55].

To apply the MPC method, discretized HCW state space representation with sample time T_s is used. Notation for discrete system with subscript 'd' is not necessarily used anymore. In other words, discrete state space representation is used in the remaining sections as below.

$$\mathrm{x}(k+1) = A\mathrm{x}(k) + Bu(k) \tag{3.1}$$

Optimal control input may be found by the minimization of a quadratic cost function. Unlike the conventional Linear Quadratic problems which use a cost function of present states and inputs, MPC input minimizes a cost function constituted of predicted state and input values over a prediction horizon.

$$J(k) = \sum_{i=0}^{N-1} \left[\mathbf{x}^T(k+i|k)Q\mathbf{x}(k+i|k) + u^T(k+i|k)Ru(k+i|k) \right]$$
$$+ \mathbf{x}^T(k+N|k)\overline{Q}\mathbf{x}(k+N|k)$$

(3.2)

Vectors including *N* predictions of states and inputs predicted at step *k* may be defined as,

$$\mathbf{u}(k) = \begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N-1|k) \end{bmatrix} \qquad \mathbf{x}(k) = \begin{bmatrix} \mathbf{x}(k+1|k) \\ \mathbf{x}(k+2|k) \\ \vdots \\ \mathbf{x}(k+N|k) \end{bmatrix}$$

(3.3)

With the predicted states at discrete intervals in time, discrete-time state space model defined throughout the prediction horizon may be written as,

$$\mathbf{x}(k) = \mathcal{M}\mathbf{x}(k) + \mathcal{C}\mathbf{u}(k)$$

(3.4)

where,

$$\mathcal{M} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix} \qquad \mathcal{C} = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}$$

(3.5)

In the following, cost function to be minimized given in Equation (3.2), may be written in the matrix form as follows [56].

$$J(k) = x^T(k)Q\,x(k) + x^T(k)\tilde{Q}\,x(k) + \mathbf{u}^T(k)\tilde{R}\,\mathbf{u}(k)$$
$$= x^T(k)Q\,x(k) + \left[x^T(k)\mathcal{M}^T + \mathbf{u}^T(k)\mathcal{C}^T \right]\tilde{Q}\left[\mathcal{M}x(k) + \mathcal{C}\mathbf{u}(k) \right] + \mathbf{u}^T(k)\tilde{R}\,\mathbf{u}(k)$$
$$= x^T(k)Q\,x(k) + \left[x^T(k)\mathcal{M}^T + \mathbf{u}^T(k)\mathcal{C}^T \right]\left[\tilde{Q}\mathcal{M}x(k) + \tilde{Q}\mathcal{C}\mathbf{u}(k) \right] + \mathbf{u}^T(k)\tilde{R}\,\mathbf{u}(k) \quad (3.6)$$
$$= x^T(k)Q\,x(k) + x^T(k)\mathcal{M}^T\tilde{Q}\mathcal{M}x(k) + x^T(k)\mathcal{M}^T\tilde{Q}\mathcal{C}\mathbf{u}(k) + \mathbf{u}^T(k)\mathcal{C}^T\tilde{Q}\mathcal{M}x(k)$$
$$\quad + \mathbf{u}^T(k)\mathcal{C}^T\tilde{Q}\mathcal{C}\mathbf{u}(k) + \mathbf{u}^T(k)\tilde{R}\,\mathbf{u}(k)$$

or,

$$J(k) = \mathbf{u}^T(k)H\,\mathbf{u}(k) + 2x^T(k)F^T\mathbf{u}(k) + x^T(k)G\,x(k) \quad (3.7)$$

where,

$$H = \mathcal{C}^T\tilde{Q}\mathcal{C} + \tilde{R}$$
$$G = \mathcal{M}^T\tilde{Q}\mathcal{M} + Q \quad (3.8)$$
$$F = \mathcal{C}^T\tilde{Q}\mathcal{M}$$

$$\tilde{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & Q & 0 \\ 0 & \cdots & 0 & \overline{Q} \end{bmatrix} \quad \text{and} \quad \tilde{R} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & R & 0 \\ 0 & \cdots & 0 & R \end{bmatrix} \quad (3.9)$$

For unconstrained case, it is convenient to calculate an explicit solution for optimal input offline by equating the gradient of the cost function shown in Equation (3.7) to zero.

$$\mathbf{u}^*(k) = -H^{-1}F\,x(k) \quad (3.10)$$

It may be realized that optimal input given in Equation (3.10) is the same as the stack of LQR controller results at different time steps. When there are constraints on inputs and states, there is no explicit solution, and optimization problem should be solved in each time step. This requires an online implementation of controller unlike the application of a constant feedback gain policies. Consequently, the input that needs to be applied becomes the solution of the optimization problem shown as follows:

$$\mathbf{u}^{*}(k) = \min_{\mathbf{u}} J(k) \tag{3.11}$$

The optimal solution of Equation (3.11) is a stack vector of future control inputs throughout the planning horizon. At each time step, only the first time step input plan is applied, i.e., $u^{*}(k) = [\begin{matrix} 1 & 0 & 0 & \cdots \end{matrix}]\mathbf{u}^{*}(k)$.
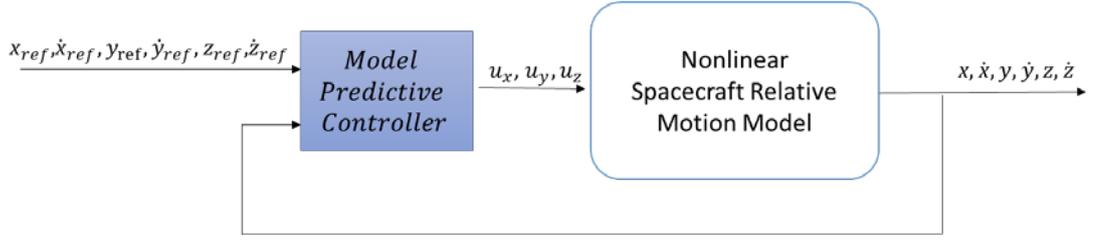
Figure 3.2: Block diagram of the controlled system where zero references are given to regulate the system to origin.

3.2 Constrained Optimization

Since the last term of summation in Equation (3.7) depends on the current and known states, it may be omitted from the optimization problem. Then, the cost function to be minimized may be rewritten as,

$$J(k) = \frac{1}{2}\mathbf{u}^{T}(k)H_{quad}\mathbf{u}(k) + f_{quad}^{T}\mathbf{u}(k) \tag{3.12}$$

where,

$$H_{quad} = 2H \quad \text{and} \quad f_{quad}^{T} = 2\mathbf{x}^{T}F^{T} \tag{3.13}$$

In the missions such as R-D, there may be several operational and safety constraints that need to be satisfied during the planning and control of the mission. Since MPC contains an optimization problem, these constraints may be applied to this problem which renders it a constrained optimization problem. In the optimization, cost function

given in Equation (3.12) is tried to be minimized in terms of the input as optimization variable with linear inequality constraints shown as follows:

$$\mathbf{u}^*(k) = \min_{\mathbf{u}} J(k)$$
$$subject\ to \quad \mathbf{x}(k+1) = A\mathbf{x}(k) + Bu(k)$$
$$A_x \mathbf{x} \le B_x$$
$$A_u u \le B_u$$

(3.14)

The problem in Equation (3.14) with the quadratic cost function to be minimized given in Equation (3.12) and linear inequality constraints constitutes a quadratic programming problem, and existing solvers can be used for constrained optimization in this problem. In this work, Matlab's *'quadprog'* solver is used with linear constraints. Quadratic programming solver of Matlab use *'interior-point-convex'* algorithm as default that has a low memory usage, and it is faster compared to other alternatives. Accuracy problem of that algorithm may be diminished by selecting smaller tolerance values. However, in this thesis, its results are quite acceptable.

Since the optimization variable is the control input in the defined problem, states and the constraints on them should be represented in terms of optimization variable. As a result, inequality constraint may be written as follows in which the only variable is input, and the current state is already known.

$$G_0 \mathbf{u}(k) \le E_0 \mathbf{x}(k) + w_0$$

(3.15)

Coefficient matrices are obtained to combine state and input constraints to represent them in terms of inputs only. Matrices for inputs and states may be derived separately and combined after as shown below.

$$G_0 = \begin{bmatrix} G_u \\ \cdots \\ G_x \end{bmatrix} \quad E_0 = \begin{bmatrix} E_u \\ \cdots \\ E_x \end{bmatrix} \quad w_0 = \begin{bmatrix} w_u \\ \cdots \\ w_x \end{bmatrix}$$

(3.16)

Matrices to represent constraints on inputs may be derived such that the inequality constraint of $A_u u \leq B_u$ is extended all over the planning horizon. Resultant constraint matrices are given as follows:

$$G_u = \begin{bmatrix} A_u & 0 & \cdots & 0 \\ 0 & A_u & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_u \end{bmatrix} \quad E_u = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad w_u = \begin{bmatrix} B_u \\ B_u \\ \vdots \\ B_u \end{bmatrix} \qquad (3.17)$$

Matrices to represent constraints on states are derived such that the state inequality constraint of $A_x x \leq B_x$ is extended through the planning horizon. Resultant constraint matrices are given as below.

$$G_x = \begin{bmatrix} 0 & 0 & \dots & 0 \\ A_x B & 0 & \dots & 0 \\ A_x AB & A_x B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_f A^{N-1} B & A_f A^{N-2} B & \dots & A_f B \end{bmatrix} \quad E_x = \begin{bmatrix} -A_x \\ -A_x A \\ -A_x A^2 \\ \vdots \\ -A_f A^N \end{bmatrix} \quad w_x = \begin{bmatrix} B_x \\ B_x \\ B_x \\ \vdots \\ B_f \end{bmatrix} \quad (3.18)$$

Equalities of $A_x = A_f$ and $B_x = B_f$ are considered as the case in this thesis. The right-hand side of the inequality constraint, i.e., $E_0 x(k) + w_0$ is computed in each time step with the knowledge of current state values, and resultant constraint matrix summation is treated as a constant. For the systems that include constraint matrices changing in time, representation in Equation (3.18) may be modified as in Equations (3.19) and (3.20). With this representation, linear inequality constraints in matrix forms with the entries changing in time can be predicted over the planning horizon and included in optimization process.

$$G_x = \begin{bmatrix} 0 & 0 & \dots & 0 \\ A_x(k+1)B & 0 & \dots & 0 \\ A_x(k+2)AB & A_x(k+2)B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A_x(k+N)A^{N-1}B & A_x(k+N)A^{N-2}B & \dots & A_x(k+N)B \end{bmatrix} \quad (3.19)$$

$$E_x = \begin{bmatrix} -A_x(k) \\ -A_x(k+1)A \\ -A_x(k+2)A^2 \\ \vdots \\ -A_f(k+N)A^N \end{bmatrix} \qquad w_x = \begin{bmatrix} B_x(k) \\ B_x(k+1) \\ B_x(k+2) \\ \vdots \\ B_f(k+N) \end{bmatrix} \qquad (3.20)$$

3.3 Stability

Although the MPC approach seems effective to be applied on R-D operations in which several safety and operational constraints may take place, closed-loop stability should still be ensured in the simulations. Stability in MPC applications may be guaranteed by the application of a terminal state constraint which imposes final state in the planning horizon to have desired value [57]. In R-D operation aim is to reach origin, and the terminal state may be given accordingly as $x_N = 0$. However, for the missions in which planning horizon length is not long enough to include the final desired state on it, it is hard to apply terminal state constraint in a feasible way. This constraint, on the other hand, may be relaxed by replacing it with a terminal cost term in the cost function [58]. Selection of this terminal cost weight matrix is made based on the solution of Discrete Algebraic Riccati Equation (DARE) given as,

$$A^T \bar{Q} A - \bar{Q} - A^T \bar{Q} B \left(B^T \bar{Q} B + R \right)^{-1} B^T \bar{Q} A + Q = 0 \qquad (3.21)$$

Addition of a terminal cost term, penalizes the controller unless the final state reaches a desired value in its plan. In R-D case, the controller is penalized until the final state reaches to origin. For this reason, this term may be renamed as "cost-to-go" term in the cost function.

CHAPTER 4

SAFE MISSION PLANNING

This chapter underlines safety considerations that should be taken into account during the mission planning. In an optimal control application, these considerations may be reflected as constraints in the optimization problem as soft or hard constraints. In this thesis, safety and operational constraints are given as hard constraints because of their importance and the urgency to achieve them.

In Section 4.1, constraints in trajectory generation are mentioned in detail. Their conceptual backgrounds and mathematical formulations are given. In Section 4.2, scenarios that are considered to simulate a complete rendezvous are introduced. Section 4.3 is dedicated to simulations of proposed methods for different rendezvous scenarios, and the comparison of MPC with a simpler LQR controller.

4.1 Constraints

This section justifies the applied safety and operational constraints. These include possible obstacles that may exist in the nominal trajectories, the requirement of specific approach directions, constraints in velocities for a slow impact with the target, and the input constraints.

In this thesis, MPC application includes the solution of an optimization problem by using linearly constrained quadratic programming (QP). This requires formulation of constraints as linear equality or inequalities. The constraints on inputs and states may

be represented separately as inequality constraints in the form given below for any variable v as follows:

$$A_{cons} \, v \le B_{cons} \tag{4.1}$$

In the optimization problem, the variable for optimization is inputs, and it is required to derive all constraints in terms of inputs including the constraints on states. This process is already mentioned in Section 3.2.

4.1.1 Constraints on Inputs

The constraints on control input may arise from the limit on available input that can be provided by actuators. In this application, inputs are the acceleration caused by the thrust generated through thrusters. These actuators may be chemically driven propulsion systems/rockets or electrical thrusters. Available thrust is limited due to actuator capacities. Instead of saturating the control inputs as in classical controllers, the limitation in available input is considered as a constraint in the optimization problem. This maintains controller to be aware of this limitation, and increases the possibility to avoid closing to the limit levels during the operation.

Since the problem is defined in LVLH frame, inputs are also applied in the same frame. It is assumed that available input level is the same for all axes in both negative and positive directions.

$$u_{min} \le u \le u_{max} \tag{4.2}$$

or,

$$|u| \le u_{available} \tag{4.3}$$

To adapt the input constraints to the representation given in Equation (4.1), two constraint matrices are formed so that the following inequality may be obtained.

$$A_u \, u \le B_u \qquad\qquad (4.4)$$

where the input is given as $u = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}^T$. A_u and B_u represent constraint matrices whose contents are shown below.

$$A_u = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \qquad B_u = \begin{bmatrix} u_{available} \\ u_{available} \\ u_{available} \\ u_{available} \\ u_{available} \\ u_{available} \end{bmatrix} \qquad (4.5)$$

The linear inequality constraint matrices of inputs are formed by considering actuator limits only. Another constraint on inputs may be for the avoidance of plume impingement to the target surface in close proximity. This is a vital issue especially for the missions with cooperative targets or with the targets that should not be damaged by thruster fire as in scientific data sample collecting missions. Such additional constraints on inputs may be implemented by the addition of related rows to constraint matrices given in (4.5). However, since the motivation of this work arises from autonomous debris removal operation idea, no such constraint is added in input actuation.

4.1.2   Constraints on States

Since the ideal sensation assumption is made based on the measurement of all states and feeding back them perfectly, operational and safety constraints are applied on states directly, not on the outputs. Linear inequality constraints on states may be represented in matrix form as below.

$$A_x \, x \le B_x \qquad\qquad (4.6)$$

In the following, three main constraints on states are introduced, and constraint matrices A_x and B_x are formed. Additional constraints on states or the same constraints on different dimensions may also be applied by the addition of associated rows to both matrices given in Equation (4.6).

4.1.2.1 Slow Approach Constraint

While berthing the target, high velocities and abrupt maneuvers should be avoided. Otherwise, it may result in hard impacts and undesired results for both the target and the chaser. To control the strength of impact, chaser velocity is decreased while closing to the target. This may be achieved by defining a decreasing function of relative distance as velocity constraint. The relative distance in three dimensions is given as the following:

$$r = \sqrt{x^2 + y^2 + z^2} \tag{4.7}$$

With the use of distance information which is updated on each step, a time-varying constraint may be formed as follows:

$$\left| \mathbf{x}_v \right| \leq \sigma \left(1 - e^{\beta r} \right) \mathbf{1}_{3 \times 1} \tag{4.8}$$

where $\mathbf{x}_v = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T$, σ and β are tuning parameters of constraint shape. In this thesis, the constraint on velocity is applied with the parameters of $\sigma = 100$ and $\beta = 0.00519$ so that no hard limitation is given on velocity of chaser when it is far from the target. The change of this constraint with the relative distance may be seen in Figure 4.1.

As the chaser becomes closer to the target, the amount of allowable velocity decreases, and in very close proximity goes to zero. Tuning parameters are selected such that for the long distances constraint does not undermine the feasibility of operation, and weight is given to proximity states.
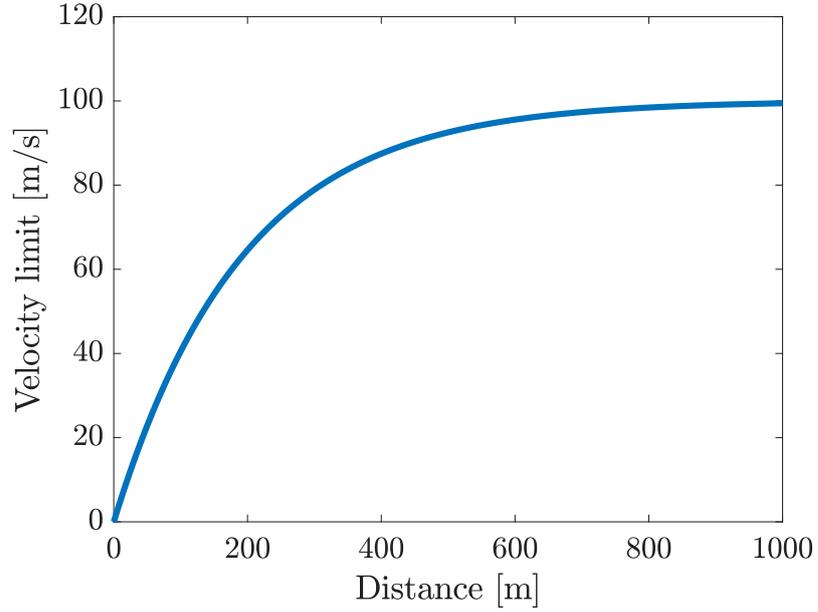
Figure 4.1: Decreasing allowable velocity profile as chaser becomes closer to the target.

If a particular direction of approach is defined, this velocity limitation may be reduced to the constraint on the velocity in this direction. For example, considering a radial approach to the target from $+x$ direction, this constraint may be applied as follows:

$$|\dot{x}| \leq \sigma\left(1 - e^{\beta r}\right) \tag{4.9}$$

With this constraint, it is provided that the chaser approaching the target becomes slower when it is close. To represent the velocity constraint in the form given in Equation (4.6), two constraint matrices are created as below such that $A_{slow}\, \mathrm{x} \leq B_{slow}$.

$$\underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix}}_{A_{slow}} \mathrm{x} \leq \underbrace{\begin{bmatrix} \sigma\left(1 - e^{\beta r}\right) \\ \sigma\left(1 - e^{\beta r}\right) \end{bmatrix}}_{B_{slow}} \tag{4.10}$$

It should be noted that these constraint matrices limit the velocity only in radial direction, i.e., along x-direction.

33

4.1.2.2 Approach Cone Constraint

The first path constraint to ensure a safe trajectory is on the region of approach. For a cooperative target, it is necessary for it to keep chaser in the Line of Sight (LOS) of the proximity sensors onboard which may be optical or radar sensors. This may be achieved by controlling the attitude of the target. Also, chaser trajectory may be planned to stay inside the LOS of the target which constrains the approach direction of chaser vehicle. On the other hand, for an uncooperative target, there is no such a limitation; however, there may be another path constraints or mission requirements that force chaser to implement approach maneuvers along some specific directions. For example, in debris removal operations it is possible for the debris to tumble or spin around a particular axis, and to capture such objects, it is essential to approach the target through its spinning/tumbling axis. Besides that, if the target has a docking port, again the chaser should approach its target through the docking port axis.

All path constraints mentioned above may be implemented by defining a region for approach that is formed around the direction of approach. Two proposed geometrical shapes for these regions are given in Figure 4.2. Given regions imply constraint zones whose centerlines are the axis of approach direction. In this thesis, radial approach from $+x$ direction is considered.
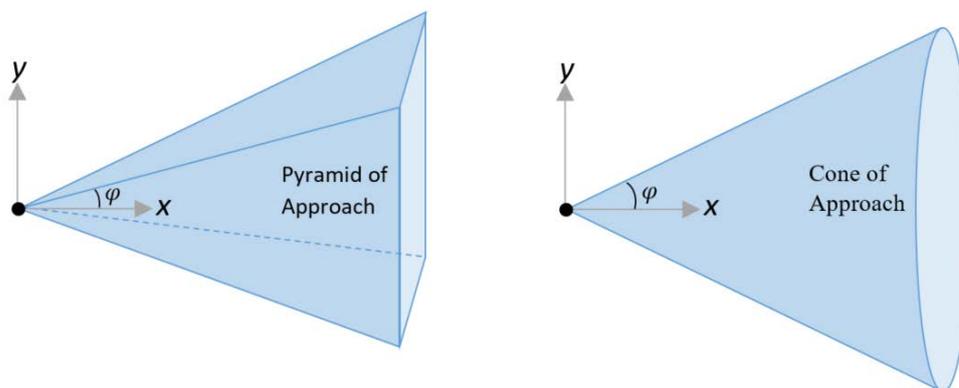
Figure 4.2: Options for direction of approach constraint geometries.

Since states in *z*-direction are decoupled, for the sake of simplicity relative motion only in the orbital *xy* plane is controlled in this thesis. Therefore, projections of both pyramid and cone constraints on this plane reduce to line equations in 2D. The constraint is applied such that the chaser stays between these two lines during the approach. The line constraints may be seen in Figure 4.2 on the right as the projection of cone into *xy* plane. From now on, since the cone is adopted as the shape for approach constraint, this path limitation is named as approach cone constraint. The angle φ is the conical half angle which defines hardness of the constraints, and it is selected as $\varphi = 45°$.

Consequently, the desired approach path should be in the region defined by $y < x$ and $-y < x$. To represent this approach cone constraint in the form given in Equation (4.6), two constraint matrices are created as below such that $A_{cone} \; \mathrm{x} \leq B_{cone}$.

$$\underbrace{\begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{cone}} \mathrm{x} \leq \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{B_{cone}} \tag{4.11}$$

It should be noted that besides its contribution to the trajectory shaping, the cone constraint prevents any overshoot in the direction of approach which considered as $+x$ direction in this work. This overshoot constraint may be represented as $x \geq 0$ in berthing phase. As a result, the chaser never goes behind the docking point.

4.1.2.3 Obstacle Avoidance Constraint

Additional constraints on states may arise from safety considerations. In space environment, besides the operational space assets, there are numerous remnants of previous space operations. Space debris may be hazardous for operational space vehicles especially when their planned trajectories intersect with these objects. The longer relative distance between chaser and target means the higher possibility of coming across a debris during the flight.

35

Collisions with such obstacles should strictly be avoided in the rendezvous path. To include this requirement of avoidance as a constraint in the optimization problem, a debris object between the chaser and target is assumed as a circular zone with radius of r_{debris} which should not be gotten inside [48]. As it is required to represent constraints as in Equation (4.6) with linear inequalities, obstacle avoidance constraints are applied as line equations which are formed as tangents to the debris circle as shown in Figure 4.3. During the relative motion between chaser and debris, at each time step, a new tangent line equation is calculated online, and chaser becomes constrained to stay on the safe side of that line. This dynamically reconfigurable obstacle avoidance constraint prevents collisions with objects which has uncertain motions.
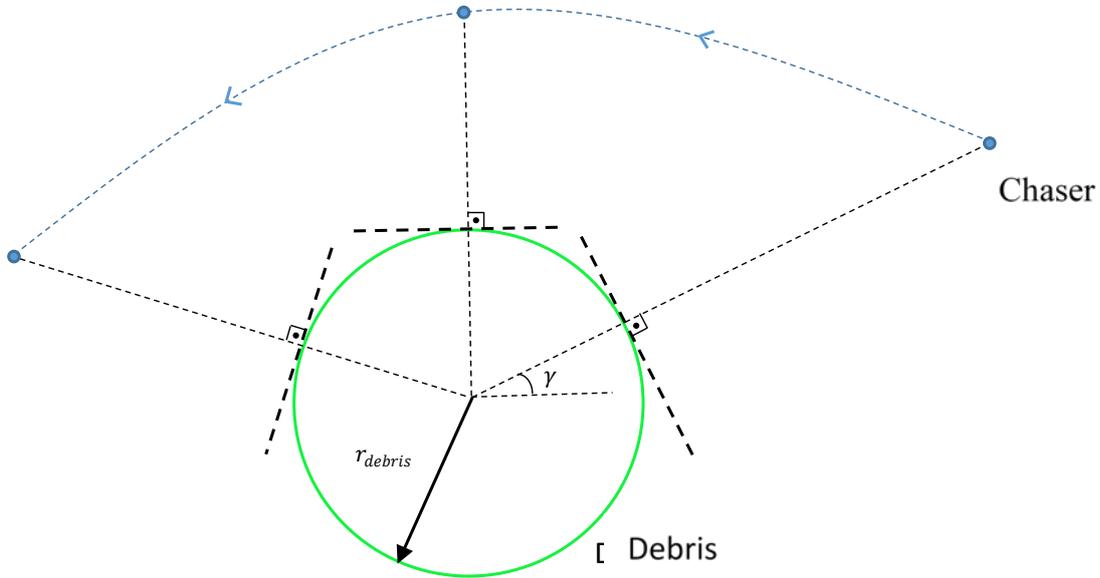
Figure 4.3: Lines tangent to the circular obstacle region as linear obstacle avoidance constraints.

The idea behind obstacle avoidance is based on keeping distance between the chaser spacecraft and obstacle always larger than the r_{debris}. First, γ is defined which is the angle between LVLH frame x-direction and the line which bonds chaser and obstacle.

$$\tan \gamma = \frac{y - y_{debris}}{x - x_{debris}} \tag{4.12}$$

$$\gamma = \tan^{-1}\left(\frac{y - y_{debris}}{x - x_{debris}}\right) \tag{4.13}$$

The distance between chaser and obstacle may be calculated by the use of positions of both and the angle γ as shown in Figure 4.4.
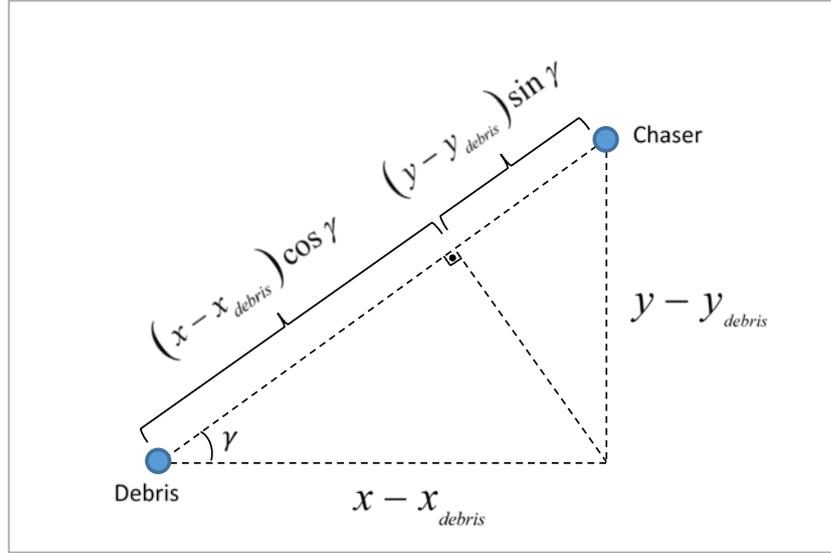
Figure 4.4: Geometrical representation of obstacle avoidance constraint elements in orbital plane.

The constraint which imposes chaser vehicle to stay out of obstacle zone may be represented as in Equation (4.14). In other words, the distance between the chaser spacecraft and obstacle must always be kept larger than the radius of the circular obstacle avoidance region.

$$\left(x - x_{debris}\right)\cos\gamma + \left(y - y_{debris}\right)\sin\gamma \geq r_{debris} \tag{4.14}$$

To represent this approach cone constraint in the form given in Equation (4.6), two constraint matrices are created such that $A_{debris\atop avoidance}\ \mathrm{x} \leq B_{debris\atop avoidance}$.

$$\underbrace{\left[-\frac{\cos \gamma}{r_{debris}} \quad -\frac{\sin \gamma}{r_{debris}} \quad 0 \quad 0 \quad 0 \quad 0 \right]}_{\substack{A_{debris} \\ avoidance}} \mathbf{x} \le \underbrace{-\frac{\cos \gamma}{r_{debris}} x_{debris} -\frac{\sin \gamma}{r_{debris}} y_{debris} -1}_{\substack{B_{debris} \\ avoidance}} \qquad (4.15)$$

4.2 Scenarios to Apply

In the proximity of cooperative or uncooperative space assets, a safe and fuel or time efficient trajectory should be generated to approach target as close as possible. This can be achieved by applying constraints mentioned in Section 4.1. Different scenarios may be created by applying combinations of aforementioned constraints. In this thesis, four different cases or scenarios are considered for rendezvous operation. In all cases, available thrust constraint is applied as it is not optional due to actuator limitations, and slow impact constraint is imposed in all scenarios as well. It should be noted that in slow approach constraint, parameters are changing with time and distance as well as in debris avoidance constraint for moving debris case. The distance between chaser and target in orbital plane for 2D case with L_2 norm is given as,

$$r = \sqrt{x^2 + y^2} \qquad (4.16)$$

The following chart explains the methodology used to create scenarios. The first decision depends on the requirement of a specific approach direction. If there is no such a requirement, operation is implemented with slow impact and input constraints only, and named as Case 1. Note that it is the only case without any path constraint. The second decision is made under the requirement of obstacle or debris avoidance. If there is no such obstacle danger, a path constraint to approach the target within an approach cone is defined in addition to Case 1, and this case is named as Case 2: Approach Cone Case. The third and final decision is on the relative motion of obstacle. If there is a relatively stationary obstacle in addition to Case 2, it is named as Case 3: Fixed Debris Case. If the obstacle has a relative motion, this case is the fourth case with moving debris.
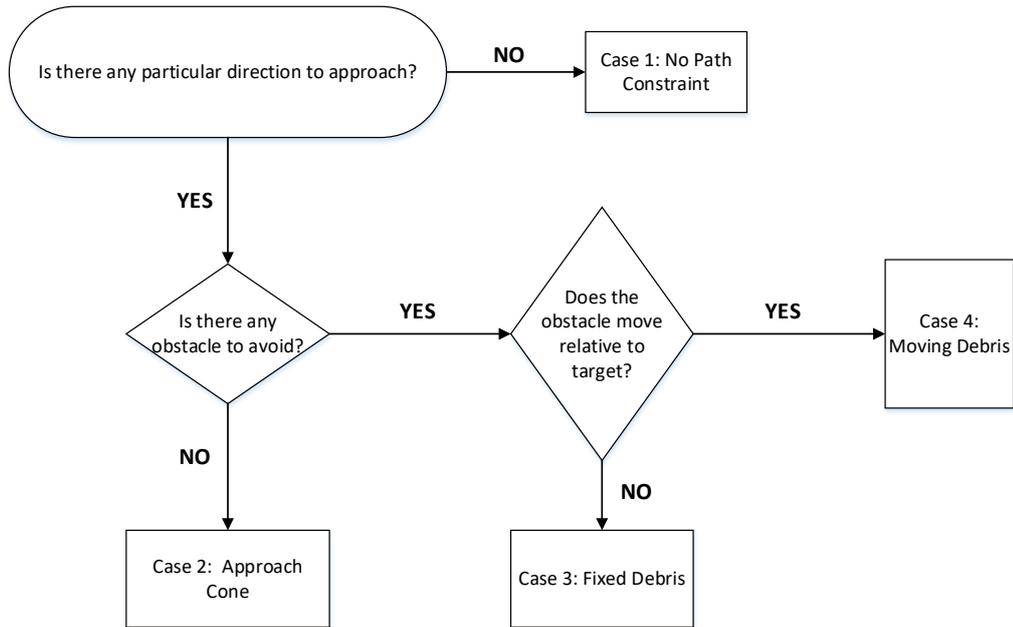
Figure 4.5: Flow chart to decide which scenario needs to be considered.

4.2.1 Case 1: No Path Constraints

Fully controlled target can arrange its orientation to make a docking with the approaching chaser. No path constraint such as approach direction is applied. Only state constraint is the one which imposes a slow impact on the target.

$$A_x = A_{slow} \qquad B_x = B_{slow} \tag{4.17}$$

Another example to this kind of scenario may be given as a space object capture which does not tumble or wobble.

4.2.2 Case 2: Approach Cone Constraint

For a collaborating target, approach should be performed through its docking port. For tumbling objects, it may not be easy to capture them as in the case for cooperative targets. The approach such objects may be performed through their axis of rotation similar to the docking port case. Instead of approaching through the docking port, this

time, the most stationary part of the target, which is on the spinning axis, is considered as the point to dock.

For such missions which require a specific direction of approach, the rendezvous problem is split into two phases. If the chaser initial position is out of the prescribed approach cone, its first aim is to get inside that cone. This is achieved by defining an avoidance circle of radius $r_{avoidance}$ which is determined by the initial relative distance r_0 according to the following formula:

$$r_{avoidance} = 100\left(1-\left(e^{-0.1r_0}\right)\right) \tag{4.18}$$

To get the inside of the approach cone, target vehicle follows a circular path until it reaches the cone border. Once it touches to the cone border, control system switches to regulate the position to the origin of the LVLH frame where the target is positioned. From Figure 4.6, a sample prescribed path may be seen with the red line, and the virtual path which is formed to take the chaser inside the approach cone is represented with the blue line.
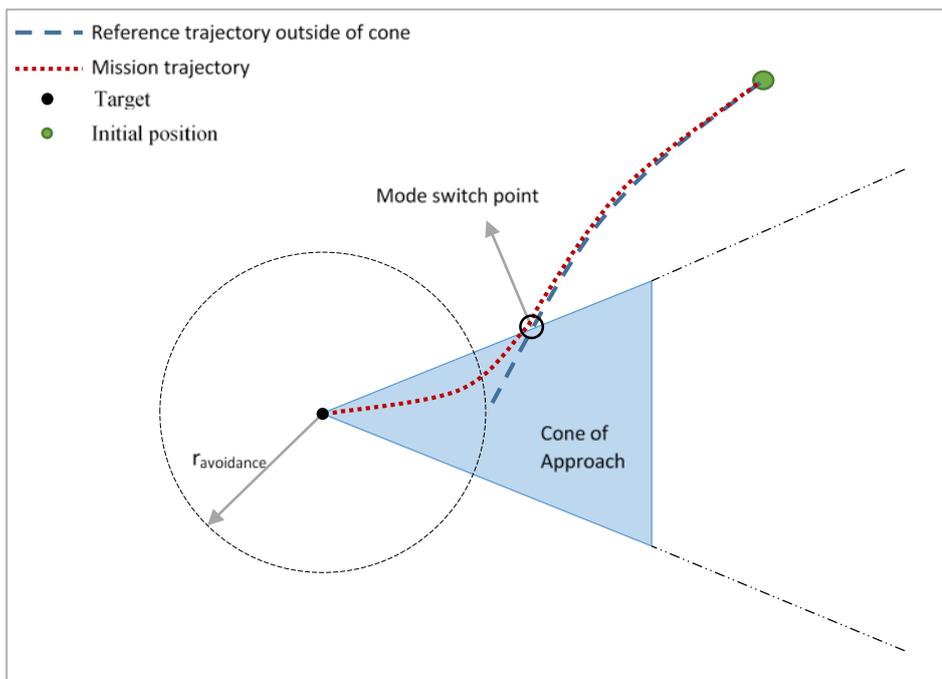
Figure 4.6: Methodology to get inside approach cone first, and then continue to rendezvous.

In this case, there are two control modes. The first mode is following the virtual path until to get inside approach cone, and the second mode is to reach the target. While the latter is the regulation problem defined in Chapter 3, the former is a tracking problem which may be solved by modifying the regulation problem. Change in desired states may be expressed with the same discrete system matrix A as follows:

$$\mathrm{x}_d\left(k+1\right) = A\,\mathrm{x}_d\left(k\right) \tag{4.19}$$

Tracking error between desired and current states may be defined as,

$$e_{track} = \mathrm{x} - \mathrm{x}_d \tag{4.20}$$

Using Equations (4.19) and (4.20), error based state space representation may be given as follows:

$$
\begin{aligned}
e_{track}\left(k+1\right) &= \mathrm{x}\left(k+1\right) - \mathrm{x}_d\left(k+1\right) \\
&= A\,\mathrm{x}\left(k\right) + B\,u\left(k\right) - A\,\mathrm{x}_d\left(k\right) \\
&= A\,e_{track}\left(k\right) + B\,u\left(k\right)
\end{aligned}
\tag{4.21}
$$

The same analogy introduced in Chapter 3 may be used to regulate tracking error dynamics given in Equation (4.21). However, constraints on states must this time be represented in terms of error. The modified constraint matrices on error is given as follows:

$$A_e e_{track} \leq B_e \tag{4.22}$$

Where,

$$A_e = A_x \quad \text{and} \quad B_e = B_x - A_x x_d \tag{4.23}$$

As stated in Chapter 3, since the optimization variable is the inputs, constraints on states must also be represented by inputs. The same analogy defined in Section 3.2 is used with new constraint matrices given in Equation (4.23) to combine state and input constraints.

Resultant state constraint matrices with slow impact and approach cone constraints are given as follows:

$$A_x = \begin{bmatrix} A_{cone} \\ A_{slow} \end{bmatrix} \qquad B_x = \begin{bmatrix} B_{cone} \\ B_{slow} \end{bmatrix} \qquad (4.24)$$

4.2.3   Case 3: Addition of an Obstacle without Relative Motion

In this case, in addition to Case 2, approach to target is performed considering a possible crash danger with the debris orbiting close to the target. The position of this obstacle, which is assumed as a debris, is kept stationary or fixed relative to the target. In other words, in LVLH frame obstacle position does not change with time.

Besides the state constraint matrices given in Case 2, additional debris avoidance constraint matrices are added. Resultant state constraint matrices are defined as,

$$A_x = \begin{bmatrix} A_{cone} \\ A_{slow} \\ A_{fixed\ debris\ avoidance} \end{bmatrix} \qquad B_x = \begin{bmatrix} B_{cone} \\ B_{slow} \\ B_{fixed\ debris\ avoidance} \end{bmatrix} \qquad (4.25)$$

4.2.4   Case 4: Addition of an Obstacle with Relative Motion

More realistic approach for obstacle avoidance case in rendezvous operation is to consider obstacles as having individual motions different than the target. Unlike the above stationary debris case, this time obstacle has a relative motion with respect to the target. Moreover, it follows an unmodeled path which is assumed to be predicted by an algorithm and fed into the control system. This requires the real-time sensing of the obstacle position. In other words, obstacle or debris position terms included in constraint matrices changing in time.

In addition, at each time step, predicted motion of obstacle is included in optimization problem over the horizon. The resultant state constraint matrices for moving debris case are obtained by the integration of related constraint matrices.

$$A_x(k) = \begin{bmatrix} A_{cone} \\ A_{slow} \\ A_{\substack{moving\ debris \\ avoidance}}(k) \end{bmatrix} \qquad B_x(k) = \begin{bmatrix} B_{cone} \\ B_{slow} \\ B_{\substack{moving\ debris \\ avoidance}}(k) \end{bmatrix} \qquad (4.26)$$

4.3 Applications for Different Scenarios

Simulations for the scenarios explained in the previous section are implemented with nonlinear dynamics of relative motion given in Equation (2.11). While for the control algorithm, a linear version of this nonlinear model linearized in Equation (2.12) is used. In the simulations, orbital parameter is chosen as $n = 0.0011 \ rad\,/\,s$ which implies that spacecraft are flying in a circular orbit at 550 km of altitude. The sampling period is selected as $T_s = 4\,s$. Control inputs are limited to $u = 0.5\,m\,/\,s^2$.

The weight matrices of states and inputs namely Q and R may be listed among the most important tuning parameters in an optimal control problem. For parametric studies, the weighting matrices are selected as,

$$Q = \begin{bmatrix} 10^3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10^3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10^3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{-1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{-1} \end{bmatrix} \qquad R = \alpha \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (4.27)$$

where, α is the relative weight between two matrices. α determines the importance of inputs relative to the states. Terminal cost weight matrix $\overline{Q}$ is chosen as the solution of Discrete Algebraic Riccati Equation (DARE) given below to ensure optimality at each step, and stability in the final step as mentioned in Section 3.3.

$$
\bar{Q} = \begin{bmatrix}
1.0047 \times 10^3 & 0 & 0 & 0.0094 \times 10^3 & 0 & 0 \\
0 & 1.0047 \times 10^3 & 0 & 0 & 0.0094 \times 10^3 & 0 \\
0 & 0 & 1.0047 \times 10^3 & 0 & 0 & 0.0094 \times 10^3 \\
0.0094 \times 10^3 & 0 & 0 & 0.0189 \times 10^3 & 0 & 0 \\
0 & 0.0094 \times 10^3 & 0 & 0 & 0.0189 \times 10^3 & 0 \\
0 & 0 & 0.0094 \times 10^3 & 0 & 0 & 0.0189 \times 10^3
\end{bmatrix} \quad (4.28)
$$

Simulations are carried out for different planning horizons and optimization weight matrices. Two performance metrics are used as given in Equation (4.29) to evaluate performance of controllers in different cases. The former indicates the fuel consumption while the latter is the indication of energy used.

$$
J_1 = \sum_{k=0}^{t_{docking}} |u_x(k)| + |u_y(k)| \qquad J_2 = \sum_{k=0}^{t_{docking}} \sqrt{u_x(k)^2 + u_y(k)^2} \qquad (4.29)
$$

Parameters for performance evaluations are chosen as planning horizon N and weighting parameter α. Time to dock is another performance criterion in addition to the performance metrics given above.

4.3.1 Case 1: No Path Constraints

Only input and slow impact constraints are applied in this case. Nonlinear simulations are conducted to make a rendezvous with the target starting from the initial position of $(400, 200, 0)\, m$.

Performance metrics for different planning horizons with constant weighting parameter $\alpha = 1$ are shown in Table 4.1. From the table, it may be observed a convex history in the performance metrics with increasing planning horizon. Because docking times are the same for three most efficient planning horizon lengths, it is reasonable to select the shortest possible planning horizon. Indeed, longer horizons lead higher computational burden. In other words, since at each step, optimization problem is solved in this planning horizon, a too long horizon means higher computational cost. For this reason, horizon length is chosen as $N = 15$. In simulations, control and prediction horizons are taken to be the same.

Table 4.1: Performance of controller with different planning horizon lengths for no path constraint case.

	$N=5$	$N=15$	$N=25$	$N=35$
J_1	14.4470	12.5206	12.5206	12.5206
J_2	11.3329	9.6997	9.6997	9.7001
$t_{docking} \, [s]$	78.0000	69.2000	69.2000	69.2000

For the selected prediction horizon $N = 15$, different weighting parameters are applied. Results are given in Table 4.2 including two performance metrics and the docking time. A rise in docking time and a decrease in costs are observed with increasing α. Therefore, optimum weighting parameter is selected as $\alpha = 1$.

Table 4.2: Performance of controller with different weight matrix ratios for no path constraint case.

	$\alpha = 10^{-6}$	$\alpha = 10^{-3}$	$\alpha = 1$	$\alpha = 10^{3}$	$\alpha = 10^{6}$
J_1	12.5207	12.5207	12.5206	12.4545	11.0809
J_2	9.7001	9.7001	9.6997	9.6755	8.4159
$t_{docking} \, [s]$	69.2000	69.2000	69.2000	70.0000	>100.0000

With the application of selected planning horizon length and weighting parameter, simulation of R-D to the target is implemented on a path generated by the MPC. Since, next cases include additional complex constraints, this case is selected for the comparison with Linear Quadratic Regulator (LQR) methodology in which constraints are not considered. Two R-D scenarios are simulated in parallel with the same initial conditions to see the efficiency and applicability of the MPC algorithm. Weighting parameter for LQR is selected as $\alpha = 5 \times 10^{6}$. In LQR case, input limits are applied to saturate them for the value of $u = 0.5 \, m/s^{2}$ which is the constrained value in MPC approach. Same parametric studies are conducted for LQR approach as well, and performances of two controllers are compared in Table 4.3.

Table 4.3: Comparison of MPC and LQR performances for no path constraint case.

	MPC	LQR
J_1	12.5206	14.2456
J_2	9. 6997	11.0573
$t_{docking}\,[s]$	69.2000	>100.0000

Results of regulated states are given in Figure 4.7. In addition, there is no obstacle avoidance requirement, and the controller is free to choose any path to dock.
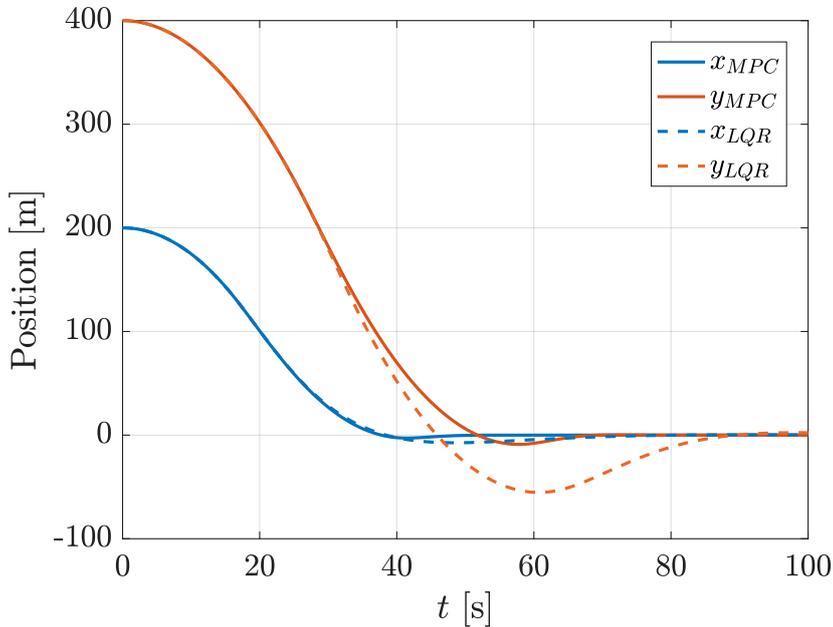
Figure 4.7: Time histories of regulated relative positions for no path constraint case.

Overshoots in the *y*-direction are observed for both methods after reaching origin which may be dangerous for the operation by increasing crash possibility. Indeed, by getting beyond the target, chaser may lose its ability to complete mission, because the target may become out of FOV of the sensors placed in the chaser vehicle or hit the target since chaser's path intersects with its nominal trajectory along $+y$ direction. Control accelerations applied to achieve this operation are presented in Figure 4.8. It may be observed that control inputs obey the limitations imposed.
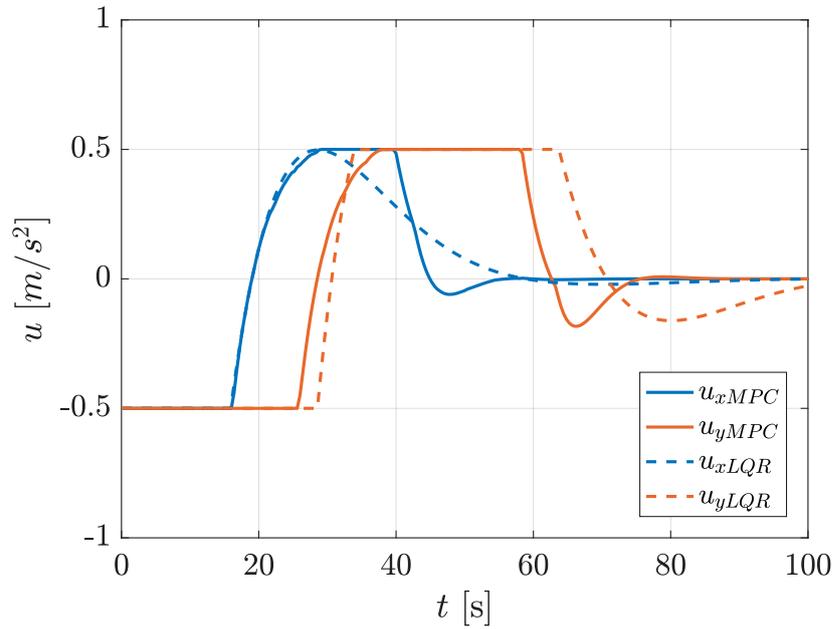
Figure 4.8: Time histories of control accelerations in orbital plane for no path constraint case.

The complete rendezvous path is given in Figure 4.9. Overshoots are evident, and they undermine the safety of the operation. However, MPC seems more efficient and applicable considering parametric study results and huge overshoot in LQR approach.
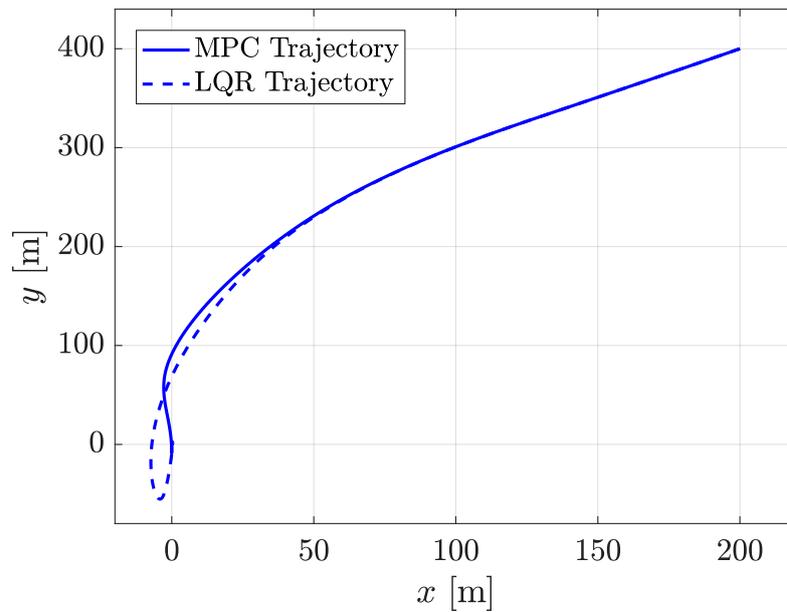
Figure 4.9: Complete rendezvous path with overshoots in both *x* and *y* directions.

This case is a demonstration of the applicability of proposed control algorithms and simulation codes. Other cases include harder path constraints. Moreover, this case gives an opportunity to compare constrained and unconstrained results for specific state constraints.

4.3.2 Case 2: Approach Cone Constraint

In addition to input and slow impact constraints, approach cone constraint is also applied in this case. In the parametric study, which is implemented to get the best planning horizon and weighting parameter, nonlinear simulations are conducted to make a rendezvous with the target starting from initial position of $(400, 200, 0)\ m$.

Performance metrics for different planning horizons with constant weighting parameter $\alpha = 1$ are shown in Table 4.4. The same results are obtained for the last three planning horizon values. Since docking times are equal for these three most efficient planning horizon times, it is reasonable to select the shortest possible planning horizon as in the previous case to avoid high computational burden. As a result, horizon length is chosen as $N = 15$. In simulations, control and prediction horizons are taken to be the same.

Table 4.4: Performance of controller with different planning horizon lengths for the case with approach cone constraint only as path constraint.

	$N=5$	$N=15$	$N=25$	$N=35$
J_1	16.8275	13.3176	13.3176	13.3176
J_2	12.5385	9.9767	9.9767	9.9767
$t_{docking}\ [s]$	93.2000	78.0000	78.0000	78.0000

For the selected prediction horizon $N = 15$, performance metrics are calculated for different weighting parameters together with the time past between initial position and docking. Results are given in Table 4.5, including two performance metrics and the docking time. From the results, it may be concluded that $\alpha = 1$ results in better

performance metrics and shorter docking time. Despite yielding smaller fuel and energy metrics, higher α values cause long docking times as well.

Table 4.5: Performance of controller with different weight matrix ratios for the case with approach cone constraint only as path constraint.

	$\alpha = 10^{-6}$	$\alpha = 10^{-3}$	$\alpha = 1$	$\alpha = 10^{3}$	$\alpha = 10^{6}$
J_1	13.3177	13.3177	13.3176	13.2624	10.2424
J_2	9.9769	9.9769	9.9767	9.9414	8.0151
$t_{docking}\,[s]$	78.0000	78.0000	78.0000	82.0000	>100.0000

With the application of selected planning horizon length and weighting parameter, simulation of R-D to the target through an approach cone is implemented on a path generated by the MPC. It may be observed from the figures that the rendezvous with the target spacecraft is realized successfully within the allowed approach zone. In Figure 4.10 and Figure 4.11, regulated relative distances and applied control acceleration inputs are given. Besides the states, control inputs also obey the limitations imposed. A different profile than the first case is observed due to the application of approach cone constraint.
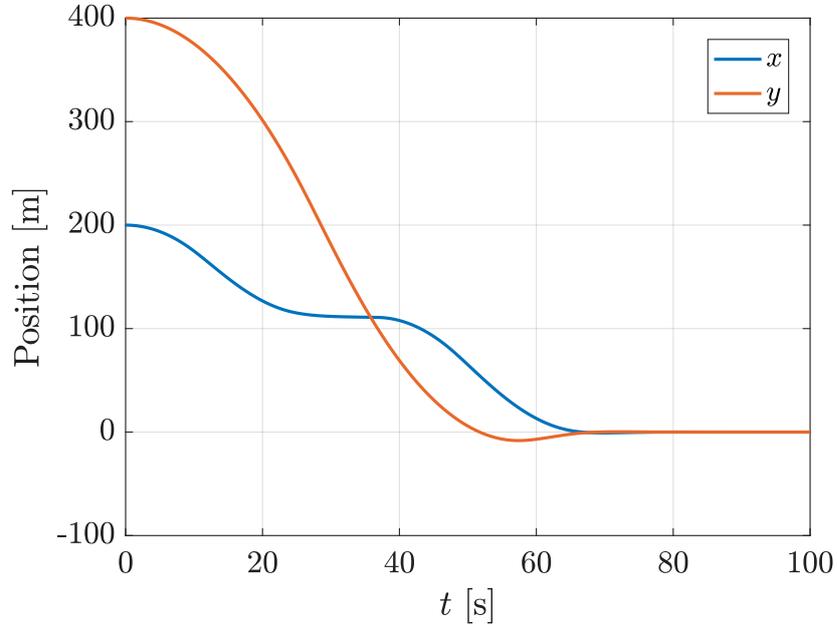
Figure 4.10: Time histories of regulated relative position states for approach cone constraint case.
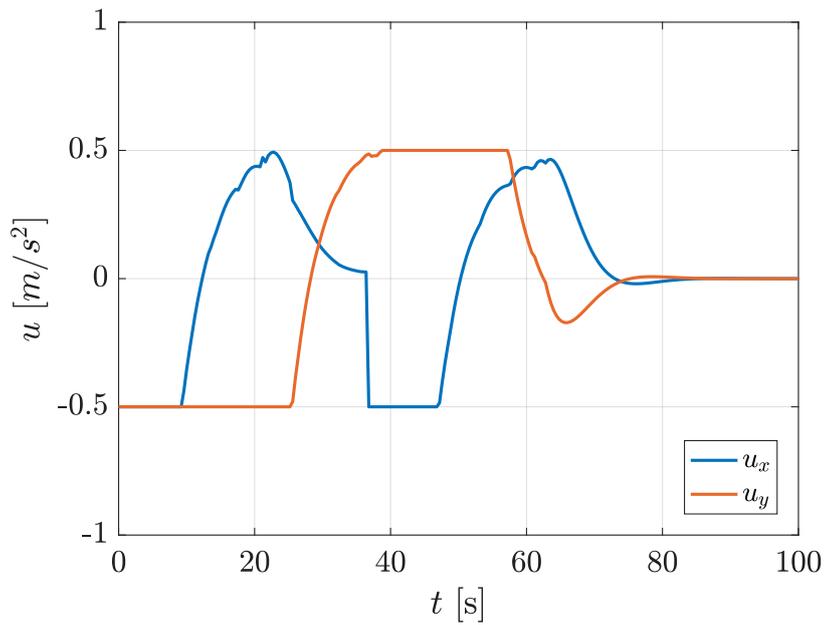
Figure 4.11: Time histories of control accelerations in orbital plane for approach cone constraint case.

The complete rendezvous path is given in Figure 4.12. Only path constraint, in this case, is to approach target through a prescribed cone. Chaser becomes steady after its first touch to the origin, i.e., the target vehicle, and no overshoot exists. Moreover, constraints are successfully satisfied, and no constraint violation is observed.
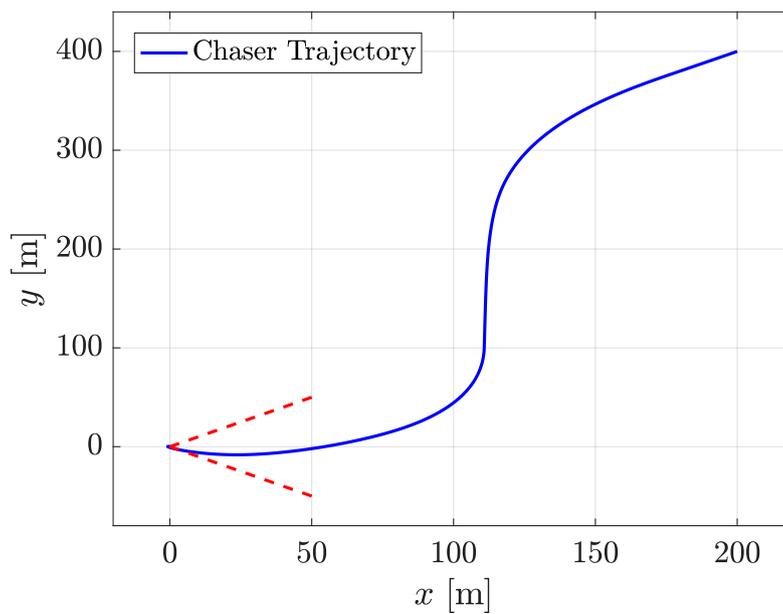
Figure 4.12: The complete rendezvous trajectory with obeying approach cone constraint only as path constraint.

The same simulation parameters are used to control the chaser vehicle starting from different initial positions. In Figure 4.13, results are given. Similar success in satisfying constraints and preventing overshoot is achieved for all initial conditions.
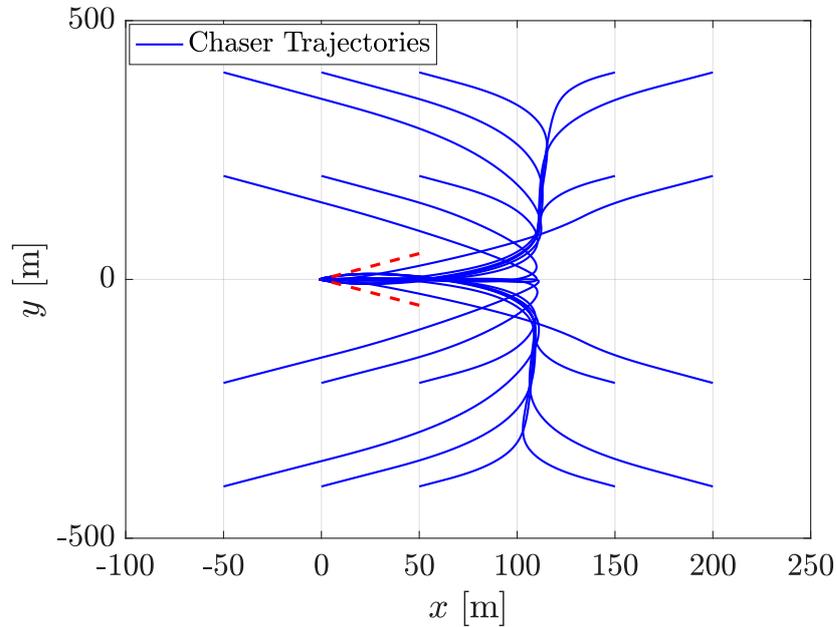
Figure 4.13: Rendezvous paths starting from different initial positions for approach cone constraint case.

4.3.3 Case 3: Addition of an Obstacle without Relative Motion

A debris is placed along the possible trajectories of the chaser vehicle as an obstacle on the position of $(80, 0, 0)$ m. The obstacle position is kept fixed relative to the target, and represented with a circle, the radius of $r_{debris} = 10$ m. Remaining requirements are the same with the Case 2. In the parametric study, nonlinear simulations are conducted to make a rendezvous with the target starting from initial position of $(400, 200, 0)$ m. Performance metrics for different planning horizons with constant weighting parameter $\alpha = 1$ are shown in Table 4.6. A convex history in the performance metrics with increasing planning horizon is observed. Since docking times are the same for the three most efficient options, $N = 15$ is selected for the planning horizon length as it is the most efficient choice. In simulations, control and prediction horizons are the same.

51

Table 4.6: Performance of controller with different planning horizon lengths for stationary debris case.

	$N=5$	$N=15$	$N=25$	$N=35$
J_1	14.4021	14.2718	14.2739	14.2739
J_2	10.8484	10.9906	10.9918	10.9918
$t_{docking}\,[s]$	97.2000	90.4000	90.4000	90.4000

For different weighting ratios, i.e., parameter α, performance metrics are calculated together with the time spent to dock. Results are shown in Table 4.7 with prediction horizon of $N=15$. Convex results are obtained, and $\alpha=1$ is concluded to have better performance and shorter docking time.

Table 4.7: Performance of controller with different weight matrix ratios for stationary debris case.

	$\alpha=10^{-6}$	$\alpha=10^{-3}$	$\alpha=1$	$\alpha=10^3$	$\alpha=10^6$
J_1	14.2741	14.2741	14.2718	14.4536	14.6224
J_2	10.9919	10.9919	10.9906	11.0926	11.2154
$t_{docking}\,[s]$	90.4000	90.4000	90.4000	94.4000	>100.0000

With the application of selected planning horizon length and weighting parameter, simulation of R-D to the target with obstacle avoidance consideration is implemented. It may be observed from the Figure 4.14 that relative positions are successfully regulated. Collision avoidance maneuver may clearly be seen especially in *x*-direction.

Control accelerations applied to achieve a collision-free and safe path are presented in Figure 4.15. An aggressive input application may be seen especially in *x*-direction. Reason for this profile is the obstacle avoidance task in the mission. Despite the aggression in the control, input accelerations still obey the limitations imposed successfully, and bring chaser to the origin safely.
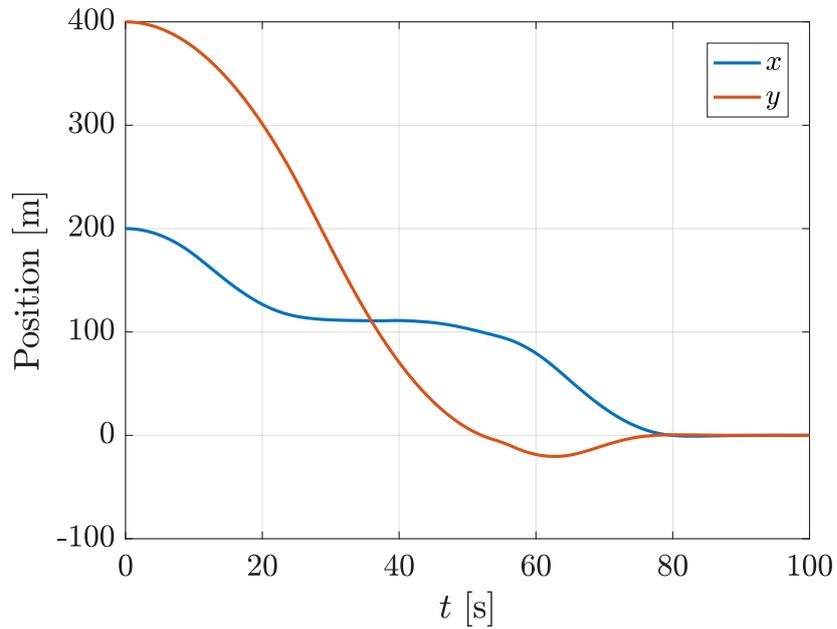
Figure 4.14: Time histories of regulated relative position states for stationary debris case.
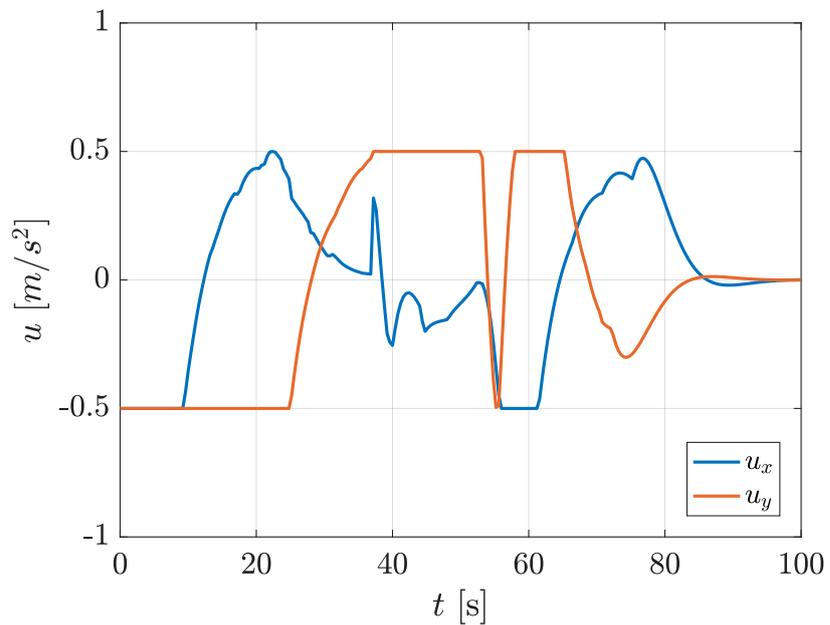
Figure 4.15: Time histories of control accelerations in orbital plane for stationary debris case.

The collision-free rendezvous path is given in Figure 4.16. Obstacle position is chosen based on the nominal path of obstacle-free missions by aiming intersection of debris with the chaser trajectory. Other state and input constraints are also applied as in

previous cases. Chaser approaches the target within the prescribed approach cone, and avoids the obstacle. It becomes steady after avoiding debris and reaching the target. All constraints are met, and a successful rendezvous is achieved.
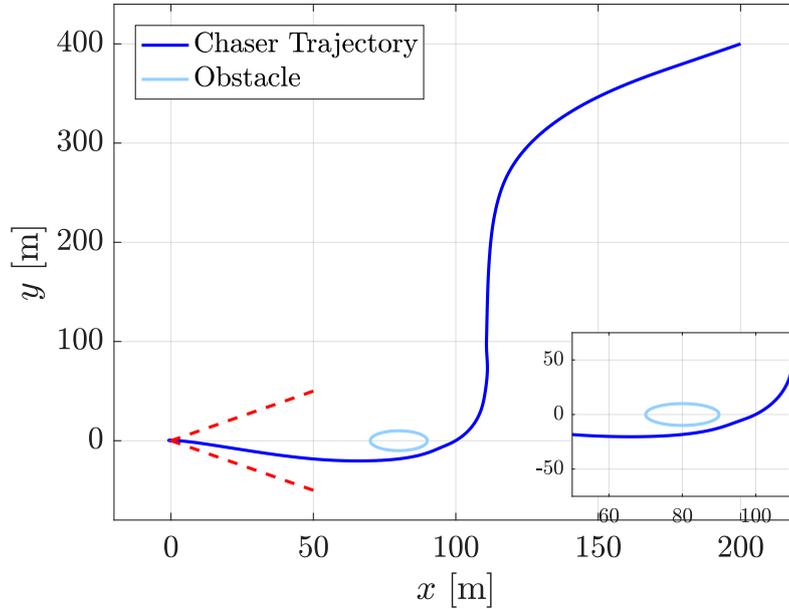
Figure 4.16: The complete rendezvous trajectory with collision avoidance constraint for stationary debris case.

The same simulation parameters are used to control the chaser vehicle starting from different initial positions, and debris position is kept the same as well. In Figure 4.17, results are given. Similar success in satisfying constraints and avoiding any collision without an overshoot in origin is achieved for all initial conditions.

As a next step, R-D operation is implemented for a chaser vehicle starting from initial position of $(400, 200, 0)\, m$ by avoiding obstacles placed in different positions. Obstacle placement is done based on the nominal collision-free trajectory in Case 2 to put debris against the chaser vehicle. State and input constraints in Case 2 is applied in addition to obstacle avoidance. For different obstacle positions, collision-free rendezvous paths are given in Figure 4.18, Figure 4.19, and Figure 4.20.
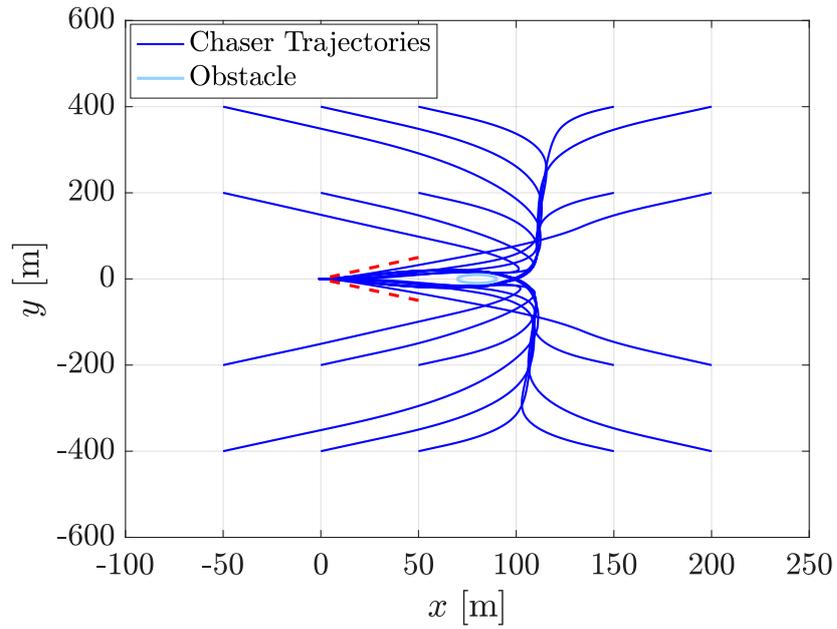
Figure 4.17: Rendezvous operation starting from different initial positions for stationary debris case.
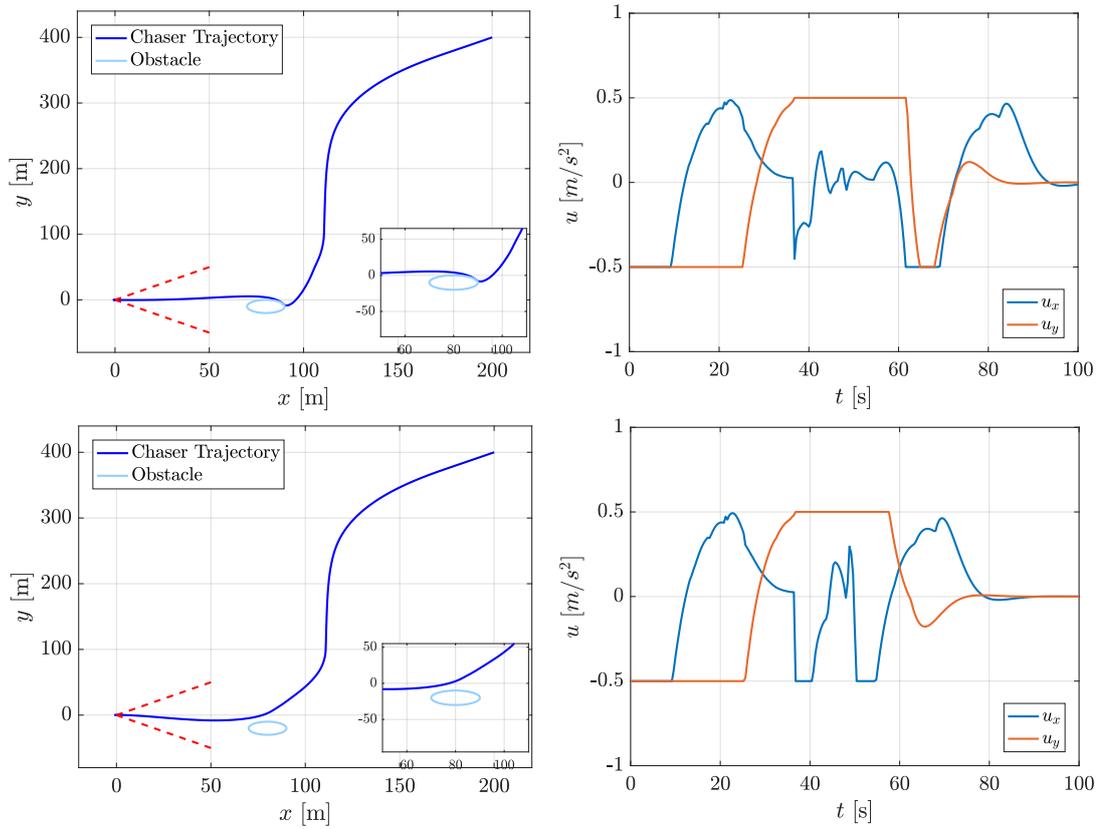
Figure 4.18: Rendezvous path of chaser spacecraft avoiding several different stationary debris positions.

Sharp avoidance of the debris is a clear demonstration of sensing the debris near its proximity. All constraints, including debris avoidance, are successfully satisfied, and no overshoot or constraint violation is observed. Chaser approaches the target within the prescribed approach cone, and avoids the obstacle. Next, it becomes steady after avoiding debris and reaching the target.

Even in the cases with obstacles placed very close proximity of the target, chaser achieves to stay in prescribed approach cone, and completes the rendezvous.
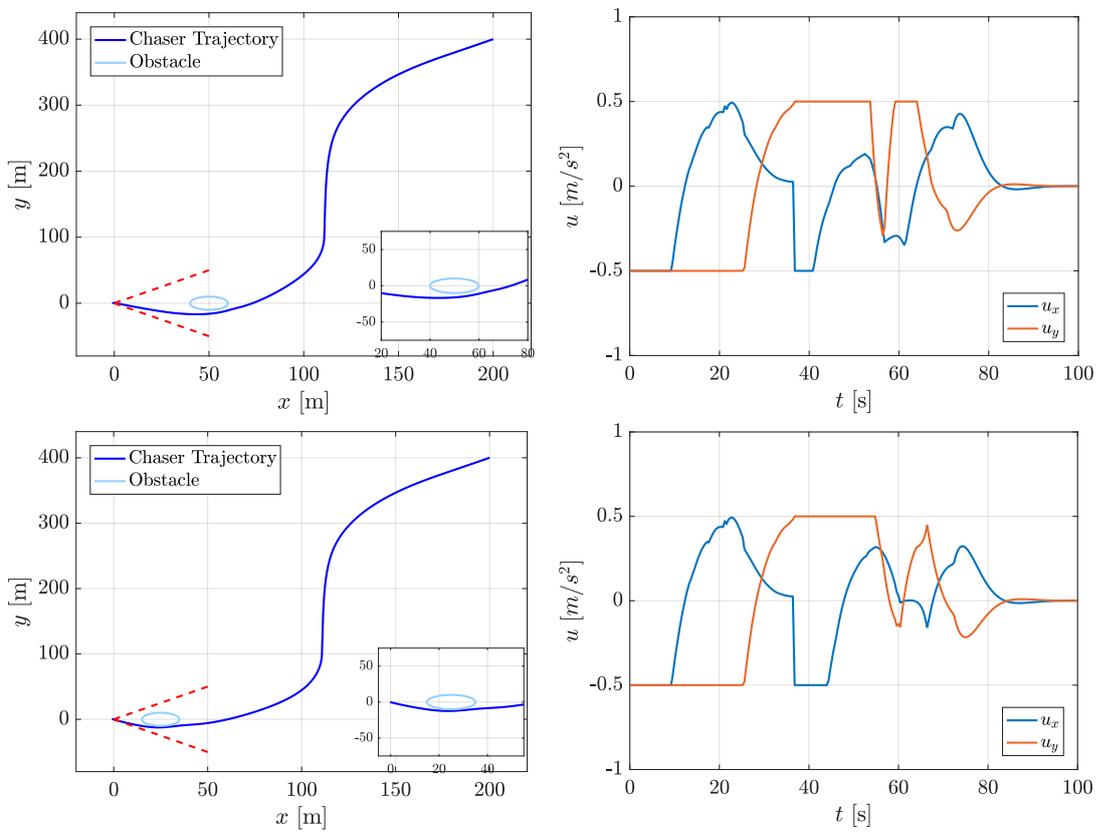
Figure 4.19: Rendezvous path of chaser spacecraft avoiding different debris positions close to target.

When debris is placed close to starting point of the chaser, more restrained input profiles are observed.
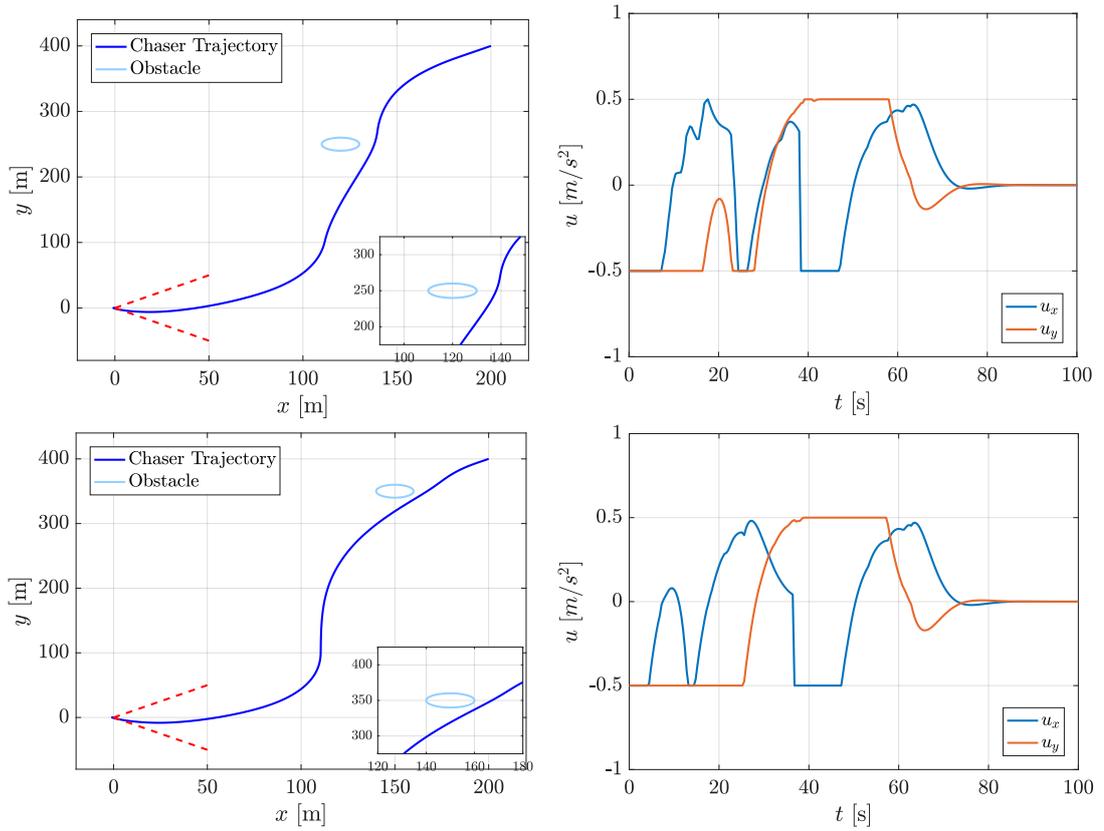
Figure 4.20: Rendezvous path of chaser spacecraft avoiding different debris positions close to chaser initial position.

4.3.4 Case 4: Addition of an Obstacle with Relative Motion

Apart from Case 3, the debris placed along the possible trajectories of the chaser vehicle as an obstacle that moves freely between chaser and the target. For the obstacle motion, an elliptical path is generated. A prediction is assumed to be provided to control system including the predicted motion of the obstacle at each time step. Indeed, this predicted obstacle motion is used in the generation of an optimal control plan over the horizon. To update the control plan according to obstacle avoidance requirement, the debris needs to be sensed online in real-time. Obstacle motion is given in orbital plane according to the representation given below. To give obstacle an oscillatory motion, debris position in *y*-direction is defined as follows:

$$y(k)_{debris} = 30*\cos\left(0.091\left(t(k)-37\right)\right) \tag{4.30}$$

The aim in this representation is the consideration of motion of obstacle after chaser vehicle enters the approach cone, i.e., sensing the debris in close proximity. Obstacle is given an elliptical path with necessary parameters to intersect with the nominal chaser trajectory without the presence of any obstacle as in Figure 4.12 of Case 2.

$$\frac{\left(x(k)_{debris}-75\right)^2}{5^2} + \frac{y(k)_{debris}^2}{30^2} = 1 \tag{4.31}$$

Since the obstacle moves relatively, it is not necessarily as conservative as in the fixed debris case. As a result, debris is represented as a moving circle with radius of $r_{debris} = 5\ m$. Remaining requirements such as state and input constraints are the same with the Case 2. In the parametric study, nonlinear simulations are conducted to make a rendezvous with the target starting from initial position of $(400, 200, 0)\ m$. Performance metrics for different planning horizons with constant weighting parameter $\alpha = 1$ are shown in Table 4.8. A convex history in the performance metrics with increasing planning horizon is observed. Considering docking times together with the costs, $N = 15$ is selected for the planning horizon length. In simulations, control and prediction horizons are taken to be the same.

Table 4.8: Performance of controller with different planning horizon lengths for moving debris case.

	N=5	N=15	N=25	N=35
J_1	13.9191	13.8384	13.8385	13.8385
J_2	10.4509	10.4893	10.4893	10.4894
$t_{docking}\ [s]$	>100.0000	86.4000	86.4000	86.4000

For the selected prediction horizon $N = 15$, performance metrics are calculated for different weighting parameters together with the docking times. Results are given in Table 4.9, including two performance metrics and times to dock. Convexity in results and docking times show that $\alpha = 1$ is the best value for the weighting parameter.

58

Table 4.9: Performance of controller with different weight matrix ratios for moving debris case.

	$\alpha = 10^{-6}$	$\alpha = 10^{-3}$	$\alpha = 1$	$\alpha = 10^{3}$	$\alpha = 10^{6}$
J_1	13.8390	13.8390	13.8384	13.6965	10.1367
J_2	10.4899	10.4899	10.4893	10.3823	8.3887
$t_{docking} [s]$	86.4000	86.4000	86.4000	90.4000	>100.0000

With the application of selected planning horizon length and weighting parameter, simulation of R-D with avoidance of a moving obstacle relative to the LVLH frame is implemented. Figure 4.21 shows the position states. The difference between with and without obstacle cases is evident. Relative positions are successfully regulated. Control accelerations applied to achieve R-D operation by avoiding a collision with moving debris are presented in Figure 4.22. The nominal case is added to observe the effects of moving obstacle case on inputs. An aggressive input application may be seen compared to nominal case in which there is no obstacle avoidance requirement. Despite the aggression in control inputs, applied accelerations still successfully obey the limitations imposed, and bring the chaser to the origin without any overshoot.
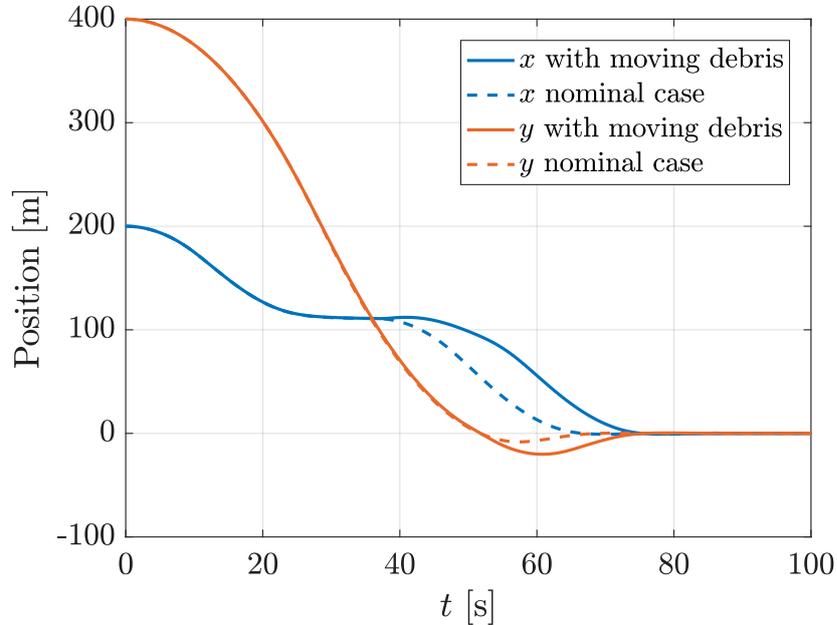
Figure 4.21: Comparison of regulated position states in orbital plane for the nominal case with direction of approach constraint only as path constraint and moving debris avoidance case.

59
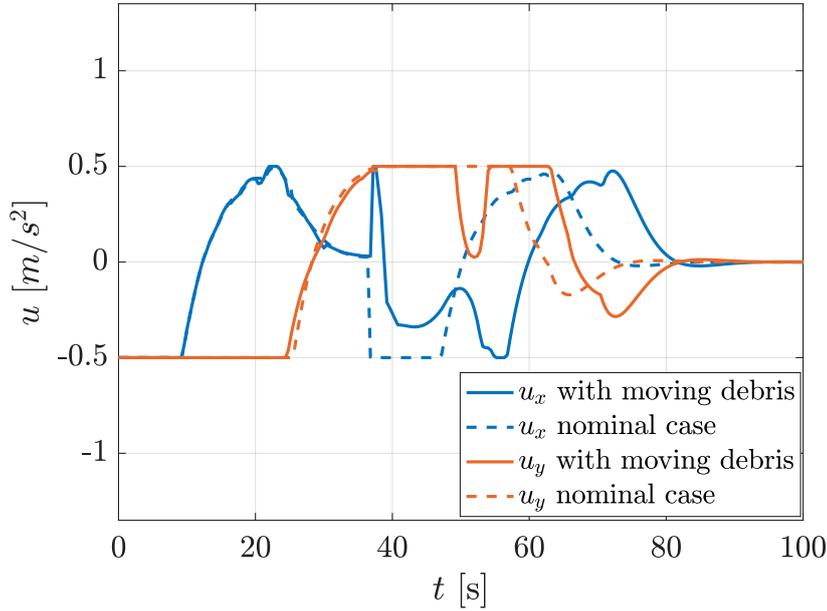
Figure 4.22: Comparison of time histories of control accelerations in orbital plane for the nominal case with direction of approach constraint only as path constraint and moving debris avoidance case.

Debris motion is represented after the chaser gets inside the approach cone. The time when the debris is started to be represented may be considered as the time when debris is sensed by the chaser sensors as well. Debris is assumed to follow an elliptical path. In the Figure 4.23, avoidance maneuver of the chaser spacecraft may be seen in 1.5 s intervals.

The complete rendezvous path generated by MPC including relatively moving obstacle avoidance maneuvers is given in Figure 4.24. Obstacle motion is generated based on the nominal path of obstacle-free missions by aiming intersection of debris path with the chaser trajectory. Nominal trajectory without any obstacle consideration is also plotted in the same figure. All aforementioned input and state constraints are also applied. As a result, chaser approaches the target within the prescribed approach cone, and avoids the moving obstacle. As in the all other cases with approach cone constraint, it becomes steady after avoiding debris and reaching to the target without overshoot.

60
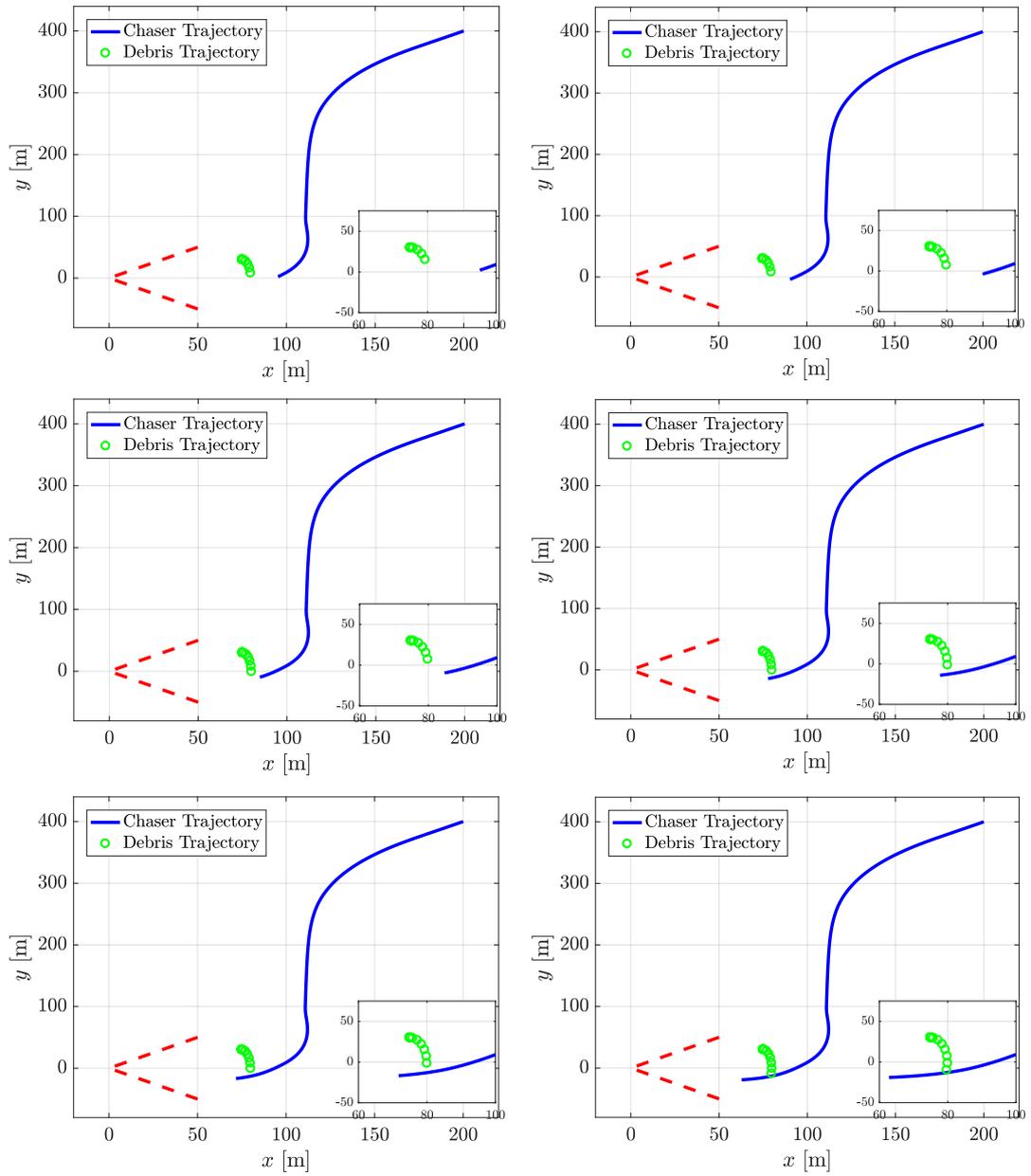
Figure 4.23: Rendezvous path of chaser spacecraft avoiding a relatively moving debris shown in 1.5 seconds intervals.
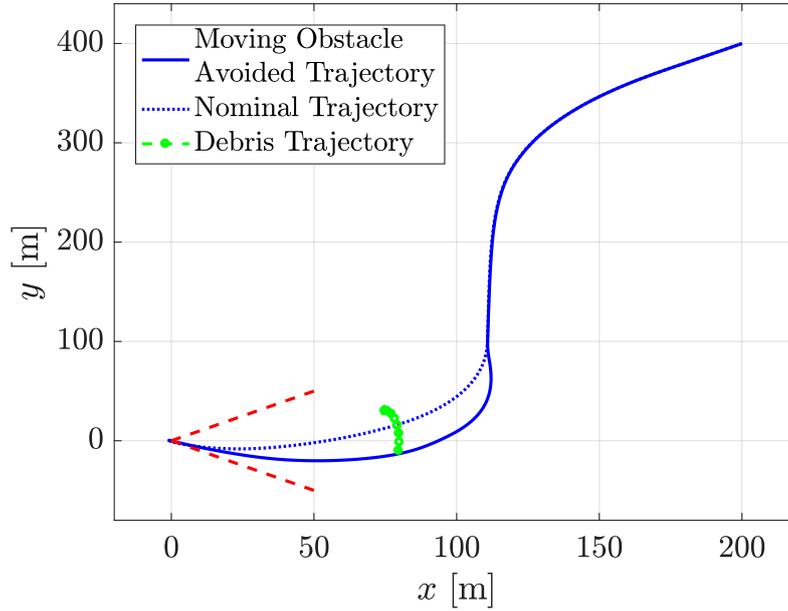
Figure 4.24: The complete rendezvous trajectory with collision avoidance constraint for moving debris, and comparison with the nominal case (Case 2).

4.3.5 Comparison of Case Results

In all cases, R-D mission is completed successfully. Moreover, input and slow impact constraints are met in each application. Although saturation-like profiles are observed in inputs, they are not long-lasting, and considering the nonlinear relative motion dynamics, it is acceptable to have such profiles in the presence of considerably low available inputs as in this study.

From results, it is observed that having a constraint that limits the direction of approach such as approach cone constraint helps to prevent overshoots that may exist after reaching origin. In Table 4.10, parametric study results which include two different performance costs and the time past between chaser initial position and target are revealed for all four cases. Results are for operations started from $(400, 200, 0)\ m$.

Table 4.10: Comparison of parametric study results including performance metrics and docking times for all four cases.

Case Numbers	Presence of Path Constraints	J_1	J_2	$t_{docking}\,[s]$
1	No positional constraints	12.5206	9. 6997	69.2000
2	Only Cone	13.3176	9.9767	78.0000
3	Cone and fixed debris	14.2718	10.9906	90.4000
4	Cone and moving debris	13.8384	10.4893	86.4000

It is clear that applying additional constraints makes the operation costlier, and docking times are increasing as well. When Cases 1 and 2 are compared, it is clear that only difference is the addition of approach cone constraint, and this causes an increase in both parametric costs and docking time as the amount of 8.8 seconds.

Case 3 is the same with Case 2 except the addition of a debris as an obstacle stationary in LVLH frame. Again, both performance metrics and the docking time are increased. The increase in time is 12.4 seconds.

In Case 4, the debris does not have a stationary position anymore. An elliptical path in the LVLH frame is defined for the obstacle. Although the only difference between Cases 3 and 4 is the relative mobility of the debris, it may be seen that moving debris case is more efficient than the stationary debris case based on performance metric results. Indeed, docking time is 4.4 seconds less for the Case 4. This phenomenon may be explained with the conservativeness in stationary debris case. If the obstacle motion is assumed as bounded in some fixed zone, this zone must be selected as a considerably bigger avoidance region to ensure that no collision takes place. This bound area was selected as a circle with radius of $r_{debris} = 10\ m$ in Case 3, However, for moving obstacle case, since the obstacle or debris is sensed in real-time, it is not necessary to define a "no enter zone" with a considerably big area. Therefore, debris may be represented as a relatively smaller area which is a circle with radius of $r_{debris} = 5\ m$ in Case 4. To conclude, instead of bounding the motion of an obstacle in a fixed zone, moving obstacle consideration leads to a more efficient mission trajectory.

It should be noted that approach angle is not tried to be set to zero during approach in none of the scenarios because an uncooperative target is assumed; however, some operations may pose a constraint to approach with the zero angle of approach. For such operations, an "end game" must be applied considering additional attitude control requirements besides the translational path control after reaching close proximity of the target with proposed MPC algorithm.

CHAPTER 5

DUAL-QUATERNION BASED ATTITUDE AND POSITION CONTROL

Up to now, the only translational motion is examined with the assumption of attitude is controlled somehow. Because safety considerations such as obstacle avoidance are mostly concerned with the translational motion of spacecrafts. In missions such as R-D and formation flying, while controlling the relative position, it is also necessary to control the orientation of spacecraft properly to keep the target within the field of view (FOV) of the sensors onboard. Coupling in translational and rotational motions poses a challenging task to overcome. MPC helps to achieve such operations by satisfying several safety constraints. However, both high demand in computational power for MPC and requirement of attitude control yield a need to generate alternative approaches.

Space vehicles usually require smooth, low jerk, attitude maneuvers. The attitude control of spacecraft is a nonlinear problem which requires nonlinear control methods. Most common nonlinear control approaches use energy-like Lyapunov functions [59], [27]. Combined attitude and position may also be controlled with the same approach. This chapter includes a novel Lyapunov function based nonlinear control algorithm which uses dual-quaternion parametrization to control both position and orientation, i.e., pose. The algorithm is based on the error dual quaternion formulation that takes time-dependent desired attitude and position trajectories into account. A Lyapunov candidate function is defined based on this error dual quaternion kinematics. The formulation adds new terms containing desired attitude and position information, and their derivatives. The algorithm is based on the previously developed attitude control algorithm to track a time-dependent attitude trajectory [60]. The algorithm is modified

65

and extended to dual quaternions to control both attitude and position. With that algorithm, it is desired to achieve translational trajectory plan generated by MPC approach while tracking challenging attitude trajectories as well.

In Section 5.1, dual-quaternion parametrization based attitude and position representation are defined in addition to the conventional quaternion based attitude representation. In Section 5.2, derivation of error dual-quaternion that is required to track given trajectory is presented. Section 5.3 includes the relative motion dynamics derivation by the use of dual-quaternion representation. In Section 5.4, the proposed nonlinear tracking control algorithm is developed. Finally, simulation results are presented and discussed in Section 5.5.

5.1 Attitude and Position Parametrization

In the following, dual quaternions are presented with the background of conventional quaternion parametrization. Attitude and position based on these methods are examined.

5.1.1 Quaternions

This section includes a brief introduction to quaternions. Attitude motion may be parametrized by the unit quaternions without any singularity. A unit quaternion represents a rotation around an axis λ with the amount of angle θ. This axis of rotation is shown with the orthogonal set of i, j, and k.

$$q = \left[\lambda_x \sin\left(\frac{\theta}{2}\right) \quad \lambda_y \sin\left(\frac{\theta}{2}\right) \quad \lambda_z \sin\left(\frac{\theta}{2}\right) \quad \cos\left(\frac{\theta}{2}\right) \right]^T \tag{5.1}$$

or,

$$q = \lambda_x \sin\left(\frac{\theta}{2}\right) i + \lambda_y \sin\left(\frac{\theta}{2}\right) j + \lambda_z \sin\left(\frac{\theta}{2}\right) k + \cos\left(\frac{\theta}{2}\right) \tag{5.2}$$

66

Multiplication of two quaternions to obtain a new attitude representation is presented with the following multiplication operation:

$$q \otimes p = q_4 \boldsymbol{p} + p_4 \boldsymbol{q} + \boldsymbol{q} \times \boldsymbol{p} + q_4 p_4 - \boldsymbol{q}^T \boldsymbol{p} \qquad (5.3)$$

Where $q = \boldsymbol{q} + q_4$ and $p = \boldsymbol{p} + p_4$ with bold vectorial parts and scalar parts with subscripts. Quaternion multiplication may be reduced to the multiplication of matrices defined in $\mathbb{R}^{4\times4}$ and $\mathbb{R}^{4\times1}$ respectively.

$$q \otimes p = \left[q \right]_{\otimes} p \qquad (5.4)$$

Where the matrix form of regular quaternion multiplication may be shown as,

$$\left[q \right]_{\otimes} = \begin{bmatrix} \boldsymbol{q}^{\times} + q_4 I & \boldsymbol{q} \\ -\boldsymbol{q}^T & q_4 \end{bmatrix} \qquad (5.5)$$

here, $\boldsymbol{q}^{\times}$ denotes the skew-symmetric matrix representation of cross product operation.

$$\boldsymbol{q}^{\times} = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \qquad (5.6)$$

5.1.2 Dual Quaternions

Dual quaternions give the opportunity to combine relative position information with the regular attitude quaternions. A dual quaternion q_{dq} is formed by two parts as real part q_1 and dual part q_2 for $\epsilon \neq 0$ and $\epsilon^2 = 0$ as in the case for dual numbers.

$$q_{dq} = q_1 + \epsilon\, q_2 \qquad (5.7)$$

A unit quaternion q representing orientation and the position vector r may be combined to represent both position and orientation with a unit dual quaternion. It is also convenient to show this dual quaternion in $\mathbb{R}^{8\times1}$, column matrix format [61]. This

representation is given in Equation (5.8). Dual part may be found by a translation in inertial frame followed by a rotation of the amount of q or first the rotation of q followed by a translation in body frame.

$$q_{dq} = \begin{bmatrix} q \\ \frac{1}{2} r_I \otimes q \end{bmatrix}_{8\times1} = \begin{bmatrix} q \\ \frac{1}{2} q \otimes r_B \end{bmatrix}_{8\times1} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}_{8\times1} \tag{5.8}$$

Dual quaternion multiplication may be reduced to the multiplication of two matrices defined in $\mathbb{R}^{8\times8}$ and $\mathbb{R}^{8\times1}$ with modified quaternion multiplication operator $\bar{\otimes}$.

$$q \,\bar{\otimes}\, p = \begin{bmatrix} [q_1]_\otimes & 0_{4\times4} \\ [q_2]_\otimes & [q_1]_\otimes \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = [q]_{\bar{\otimes}} \, p \tag{5.9}$$

It is quite useful to define the operator of $[a]_{\bar{\otimes}}^*$ as in Equation (5.10), which adds commutativity to dual quaternion matrix multiplication [61].

$$[q]_{\bar{\otimes}} \, p = [p]_{\bar{\otimes}}^* \, q = \begin{bmatrix} [p_1]_\otimes^* & 0_{4\times4} \\ [p_2]_\otimes^* & [p_1]_\otimes^* \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{5.10}$$

This operator's conventional quaternion version is given as,

$$[p]_\otimes^* = \begin{bmatrix} \left[p^\times \right]^T + p_4 I & p \\ -p^T & p_4 \end{bmatrix} \tag{5.11}$$

Dual quaternion conjugate may be written as,

$$q_{dq}^* = \begin{bmatrix} q_1^* \\ q_2^* \end{bmatrix} \tag{5.12}$$

where $q_1^* = -\boldsymbol{q_1} + q_{1_4}$ and $q_2^* = -\boldsymbol{q_2} + q_{2_4}$.

5.2   Derivation of Error Dual Quaternion

From now on, all letters which are used to symbolize quaternions, represent a dual quaternion column-matrix form defined in $\mathbb{R}^{8\times1}$. Some parts may have dual quaternion multiplications without multiplication symbol. Error dual quaternion e associated with the desired pose d and the current pose q, may be defined as follows:

$$q = d \,\overline{\otimes}\, e \tag{5.13}$$

or,

$$e = d^* \,\overline{\otimes}\, q \tag{5.14}$$

Using the chain rule, error dual quaternion derivative may be shown as below.

$$\dot{e} = \dot{d}^* \,\overline{\otimes}\, q + d^* \,\overline{\otimes}\, \dot{q} \tag{5.15}$$

Kinematics used to relate regular quaternion, and rotational velocity is valid for dual quaternions also with the dual velocity definition [62].

$$\dot{q} = \frac{1}{2} q \,\overline{\otimes}\, w \quad or \quad \dot{q} = \frac{1}{2}[w]^*_{\otimes} q \tag{5.16}$$

Where dual velocity is given as $w = \omega + \epsilon v$, or with the matrix form, it may be represented as $w = \begin{bmatrix} \omega^T & 0 & v^T & 0 \end{bmatrix}^T$. By the substitution of Equations (5.14) and (5.16) into Equation (5.15), simplifications may be made on the derivative of error dual quaternion e as below.

$$\dot{e} = \dot{d}^* de + \frac{1}{2} d^*[w]^*_{\otimes} de \tag{5.17}$$

$$\dot{e} = \left[\dot{d}^* d + \frac{1}{2} d^*[w]^*_{\otimes} d \right] e \tag{5.18}$$

$$\dot{e} = \left[\dot{d}^* d + \frac{1}{2} d^*[d]_{\overline{\otimes}} w \right] e \tag{5.19}$$

By defining $s = \dot{d}^* d$, finalized and simplified error kinematics may be obtained as,

$$\dot{e} = \left[s + \frac{1}{2} w \right] e \qquad (5.20)$$

This differential equation of attitude and position propagation of the error dual quaternion is used for spacecraft pose control in next sections.

5.3 Dual Quaternion Representation of Relative Motion Dynamics

In the following, $(\ )^c$ and $(\ )^t$ indicate that the quantity is given in chaser body frame and LVLH frame respectively relative to inertial frame. $(\ )_{ct}^c$, on the other hand, indicates that the quantity is written in chaser's body-fixed frame relative to the LVLH frame. Translational and rotational dynamics may be combined by defining cross product operator for dual quaternions.

$$q \, \overline{\times} \, p = \begin{bmatrix} \boldsymbol{q_1} \times \boldsymbol{p_1} \\ \boldsymbol{q_1} \times \boldsymbol{p_2} + \boldsymbol{q_2} \times \boldsymbol{p_1} \end{bmatrix} \qquad (5.21)$$

Translational and rotational equations of motion in chaser body frame are given as follows:

$$\begin{aligned} m\dot{v} + \omega \times mv &= F \\ \mathbf{J}\dot{\omega} + \omega \times \mathbf{J}\omega &= T \end{aligned} \qquad (5.22)$$

Where, m is the mass of the spacecraft, $\mathbf{J}$ is the inertia matrix, F is the force, and T is the torque acting on the spacecraft. These two equations are combined with the help of operator given in Equation (5.21), and dual quaternion form of them may be written as [63],

$$\overline{J}\dot{w}_c^c + w_c^c \, \overline{\times} \, \overline{J}w_c^c = u_c^c \qquad (5.23)$$

70

Where $\bar{J}$ is the invertible matrix with anti-diagonal blocks defined in $\mathbb{R}^{8 \times 8}$. u is the combined input, and w is the combined translational and rotational velocities or dual velocity,

$$w = \begin{bmatrix} \omega \\ 0 \\ v \\ 0 \end{bmatrix} \qquad u = \begin{bmatrix} F \\ 0 \\ T \\ 0 \end{bmatrix} \qquad \bar{J} = \begin{bmatrix} 0_{3\times3} & 0 & mI & 0 \\ 0 & 0 & 0 & 1 \\ \mathbf{J} & 0 & 0_{3\times3} & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.24}$$

Relative coupled motion dynamics may be written as [63],

$$\dot{w}_{ct}^c = -\bar{J}^{-1}\left(w_c^c \times \bar{J}w_c^c\right) + \bar{J}^{-1}u_c^c - q_{ct}^* \otimes \dot{w}_t^t \otimes q_{ct} + \dot{w}_{ct}^c \times \left(q_{ct}^* \otimes w_t^t \otimes q_{ct}\right) \tag{5.25}$$

If the relative pose is measurable, Equation (5.25) may be modified using $w_c^c = w_{ct}^c + q_{ct}^* \otimes w_t^t \otimes q_{ct}$ to write dynamics in terms of target motion on LVLH frame and relative motion on chaser body-fixed frame as below.

$$\dot{w}_{ct}^c = -\bar{J}^{-1}\left(\left(w_{ct}^c + q_{ct}^* \otimes w_t^t \otimes q_{ct}\right) \times \bar{J}\left(w_{ct}^c + q_{ct}^* \otimes w_t^t \otimes q_{ct}\right)\right) + \bar{J}^{-1}u_c^c$$
$$- q_{ct}^* \otimes \dot{w}_t^t \otimes q_{ct} + w_{ct}^c \times \left(q_{ct}^* \otimes w_t^t \otimes q_{ct}\right) \tag{5.26}$$

Other equations that helps to derive above relative dynamic equation are given as follows:

$$q_c = q_t \otimes q_{ct}$$
$$\dot{q}_{ct} = \frac{1}{2} q_{ct} \otimes w_{ct} \tag{5.27}$$

It should be noted that relative coupled dynamics are represented in chaser body frame, and they should be transformed into LVLH frame to compare results with the MPC approach obtained in Chapter 4.

5.4 Lyapunov Based Control Algorithm

The nonlinear control of attitude and position may be realized with a properly selected Lyapunov function. A positive definite Lyapunov candidate function is given as follows:

$$V\left(s, w, (e^T e - 1)\right) = \frac{1}{2}\left(2s + w\right)^T K_p^{-1} \overline{J}\left(2s + w\right) + e^T e - 1 \tag{5.28}$$

where, $V\left(s = 0, w = 0, (e^T e - 1) = 0\right) = 0$ as the spacecraft reaches desired attitude and position. Derivation of the Lyapunov candidate function leads to the following:

$$\dot{V} = \left(2s + w\right)^T K_p^{-1} \overline{J}\left(2\dot{s} + \dot{w}\right) + 2\dot{e}^T e \tag{5.29}$$

Error dual quaternion derivative on the right-hand side of the summation in Equation (5.29) may be found by using the property defined in Equation (5.10).

$$\dot{e} = [e]_{\bar{\otimes}}^* \left[s + \frac{1}{2}w \right]$$
$$\dot{e}^T = \left[s + \frac{1}{2}w \right]^T [e]_{\bar{\otimes}}^{*\,T} \tag{5.30}$$

Derivative of w may be retrieved from the combined dynamics in Equation (5.23). Substitution of Equations (5.26) and (5.30) into Lyapunov function in Equation (5.29) yields,

$$\dot{V} = \left(2s + w\right)^T \left[K_p^{-1} \overline{J}\left[2\dot{s} + \overline{J}^{-1}\left(u_c^c - \left(w_{ct}^c + q_{ct}^* \bar{\otimes} w_t^t \bar{\otimes} q_{ct} \right) \bar{\times} \overline{J}\left(w_{ct}^c + q_{ct}^* \bar{\otimes} w_t^t \bar{\otimes} q_{ct} \right) \right) \right. \right.$$
$$\left. \left. - q_{ct}^* \bar{\otimes} \dot{w}_t^t \bar{\otimes} q_{ct} + w_{ct}^c \bar{\times} \left(q_{ct}^* \bar{\otimes} w_t^t \bar{\otimes} q_{ct} \right) \right] + [e]_{\bar{\otimes}}^{*\,T} e \right] \tag{5.31}$$

For an asymptotically stable system, this derivative must be decaying. By equating the derivative to a negative definite function given in Equation (5.32), this decay may be guaranteed.

$$\dot{V} = -\left(2s + w\right)^T K_p^{-1} K_d \left(2s + w\right) \tag{5.32}$$

The equalization of these two functions yields,

$$-K_p^{-1}K_d(2s+w) = K_p^{-1}\left[2\overline{J}\dot{s} + u_c^c - \left(w_{ct}^c + q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right)\overline{\times} \overline{J}\left(w_{ct}^c + q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right)\right. \\ \left. - \overline{J}\left(q_{ct}^* \overline{\otimes} \dot{w}_t^t \overline{\otimes} q_{ct}\right) + \overline{J}\left(w_{ct}^c \overline{\times}\left(q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right)\right)\right] + \overline{e}$$

(5.33)

where $\overline{e} = [e]_{\overline{\otimes}}^{*T} e$, and taking u to the left-hand side, the combined control input may be found as follows:

$$u_c^c = \left(w_{ct}^c + q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right)\overline{\times} \overline{J}\left(w_{ct}^c + q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right) \\ - \overline{J}\left(2\dot{s} - q_{ct}^* \overline{\otimes} \dot{w}_t^t \overline{\otimes} q_{ct} + w_{ct}^c \overline{\times}\left(q_{ct}^* \overline{\otimes} w_t^t \overline{\otimes} q_{ct}\right)\right) \\ - K_p\overline{e} - K_d(2s+w)$$

(5.34)

It should be noted that the control input comprises the derivative of desired attitude and position. Moreover, it is represented on chaser body-fixed frame. The positive definite gain matrices may be chosen as,

$$K_p = \begin{bmatrix} \omega_{n_m}^2 I_4 & 0_{4\times4} \\ 0_{4\times4} & \omega_{n_J}^2 I_4 \end{bmatrix}\overline{J} \quad K_d = \begin{bmatrix} 2\xi\omega_{n_m} I_4 & 0_{4\times4} \\ 0_{4\times4} & 2\xi\omega_{n_J} I_4 \end{bmatrix}\overline{J}$$

(5.35)

$\omega_{n_m}, \omega_{n_J}$, and ξ may be chosen by considering response performances [60]. In this work, different values for translational and rotational motions are selected, and weight between these motions is determined.

5.5   Simulation Results

Automated R-D operation may be implemented by generating time-dependent trajectories and tracking of them. The desired attitude and position trajectories may be defined using polynomials [64], or employing an optimization routine based on a cost function with constraints on control torques and forces [65], [66]. Trajectory planning may also be carried out with the intention of avoiding certain attitudes [64] or positions. The objective of simulations is to track time-dependent trajectories of attitude and position. This may be achieved by keeping the unit error dual quaternion

value as $e = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. Another equilibrium point that represents the same motion is $e = [-1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. Between these two options the one with the minimum path length, i.e., avoidance of unwinding phenomenon should be selected. Although in this study a precise tracking is implemented, unreasonable path selections may still be a problem and should be avoided. This may be achieved by the implementation of a proper switch between error dual quaternion definitions according to the sign of scalar part of the attitude quaternion as shown below [63]:

$$\eta = \begin{cases} 1, & e_{1_4} \geq 0 \\ -1, & e_{1_4} < 0 \end{cases} \tag{5.36}$$

Regarding the sign of the scalar part in attitude error quaternion, vectorial part is modified as below to ensure the avoidance of unwinding.

$$e_{1_{new}} = \eta \, e_1 \tag{5.37}$$

Then simulations are conducted with updated control algorithm including modified error dual quaternion of $e_{dq} = e_{1_{new}} + \epsilon e_2$ where modification is on attitude part of $e_{1_{new}} = \boldsymbol{e_{1_{new}}} + e_{1_4}$, and position related part is kept same as $e_2 = \boldsymbol{e_2} + e_{2_4}$. This turns control system stated in Section 5.4 into a discontinuous controller with a switch mechanism. Chaser spacecraft's physical properties are selected as,

$$\text{Mass: } m = 100 \, kg$$

$$\text{Moment of Inertia: } \mathbf{J} = \begin{bmatrix} 10 & -0.3 & -0.7 \\ -0.3 & 18 & 0.2 \\ -0.7 & 0.2 & 8 \end{bmatrix} kg \cdot m^2 \tag{5.38}$$

Attitude quaternion is comprised of,

$$\boldsymbol{q} = \lambda \sin(\theta/2) \qquad q_4 = \cos(\theta/2) \tag{5.39}$$

74

The rotation angle is taken as a function of time $\theta = a + bt + ct^2 + dt^3$. To calculate coefficients, initial and final conditions are specified as $\theta_0 = 0$, $\dot{\theta}_0 = 0$, $\theta_f = 2\pi/3$, $\dot{\theta}_f = 0$ and $\lambda = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T / \sqrt{14}$ as in [60]. Desired position trajectory is taken from MPC results of Case 4 in Section 4.3.4 which is a rendezvous trajectory with relatively moving obstacle avoidance. The desired path may be seen in Figure 4.24. In other words, the same translational plan generated by MPC is desired to be tracked while controlling the attitude also at the same time by the same control algorithm. This MPC plan is fed to the dual quaternion feedback by cubic interpolation between the solutions. The final time for simulations is selected as $t_f = 100\,s$. Controller parameters to define gain matrices K_p and K_d are selected as the following:

$$\omega_{n_m}^2 = 10\ m/s^2 \qquad \omega_{n_J} = 0.5\ rad/s \qquad \xi = 0.7 \qquad (5.40)$$

Since control application and simulations are conducted in chaser body-fixed frame, simulation results such as dual velocity, position, and inputs should be transformed into LVLH frame to compare results with the MPC approach results obtained in Chapter 4. For this reason, at the beginning of simulation desired trajectory obtained with MPC and dual quaternion generated with these values are transformed to the chaser body-fixed frame. Next, simulations are conducted to follow given attitude and position trajectories in the chaser frame. Finally, input and state results are converted by following transformations to represent them in the LVLH frame and to compare them with MPC results.

$$u_c^t = q_{ct} \,\overline{\otimes}\, u_c^c \,\overline{\otimes}\, q_{ct}^* \qquad (5.41)$$

$$w_{ct}^t = q_{ct} \,\overline{\otimes}\, w_{ct}^c \,\overline{\otimes}\, q_{ct}^* \qquad (5.42)$$

The time histories of resulting desired and realized dual quaternions and the differences between them are given in Figure 5.1. It should be noted that these are quaternion entries only, not angles or positions. Tracking error is found as the

component-wise difference between the real and dual parts where the real part is the attitude quaternion, and dual part may be considered as the position quaternion.
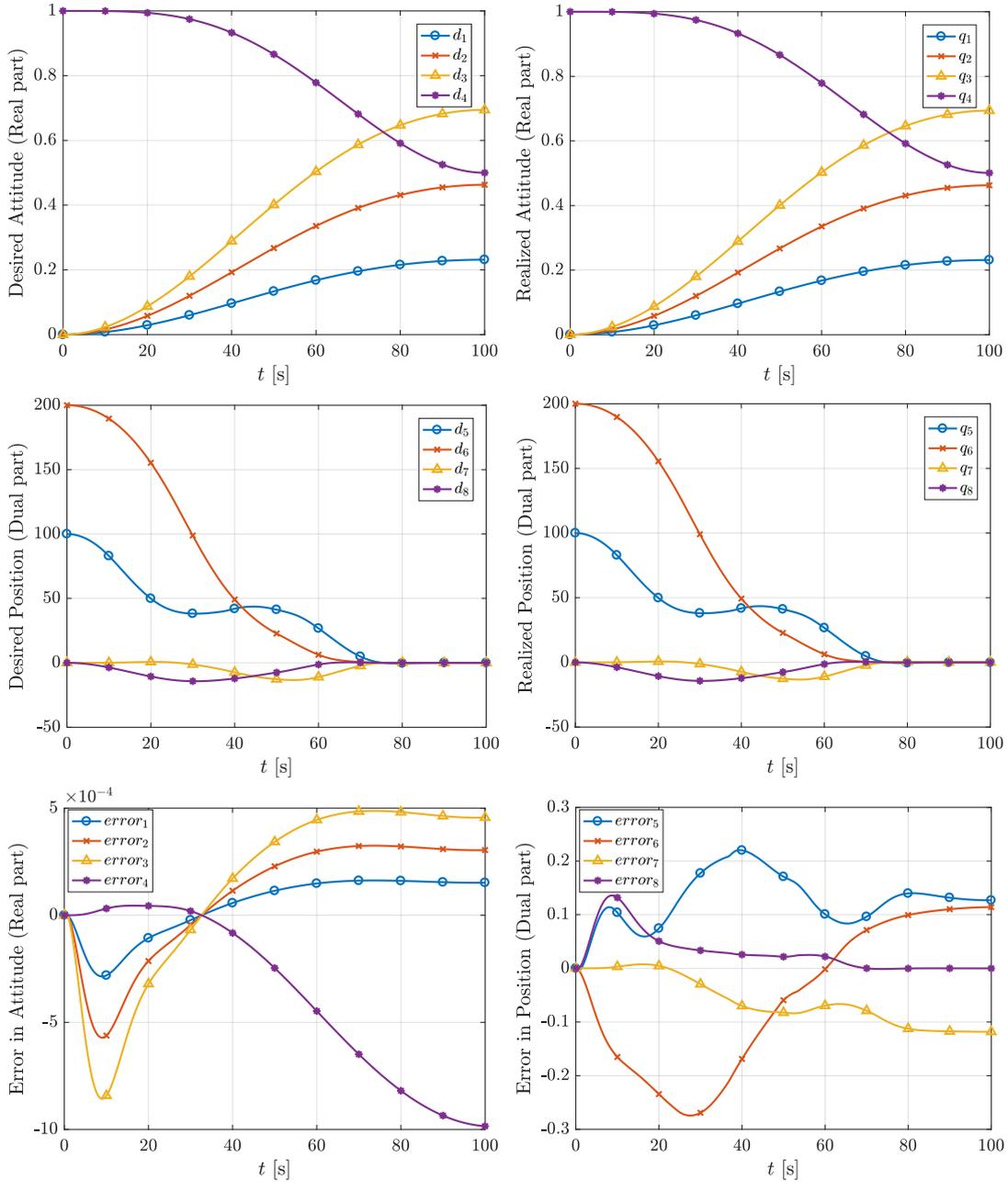

Figure 5.1: Time histories of the desired and realized dual quaternions as attitude and position parts, and the error in the realization of these parts.

In the realization of the desired attitude quaternion, sensitivity with the amount of $\pm 10^{-4}$ is obtained. To evaluate the sensitivity in position tracking, it is reasonable to

look at the relative position time histories. In Figure 5.2, realized position and the difference between desired and realized positions are given respectively.
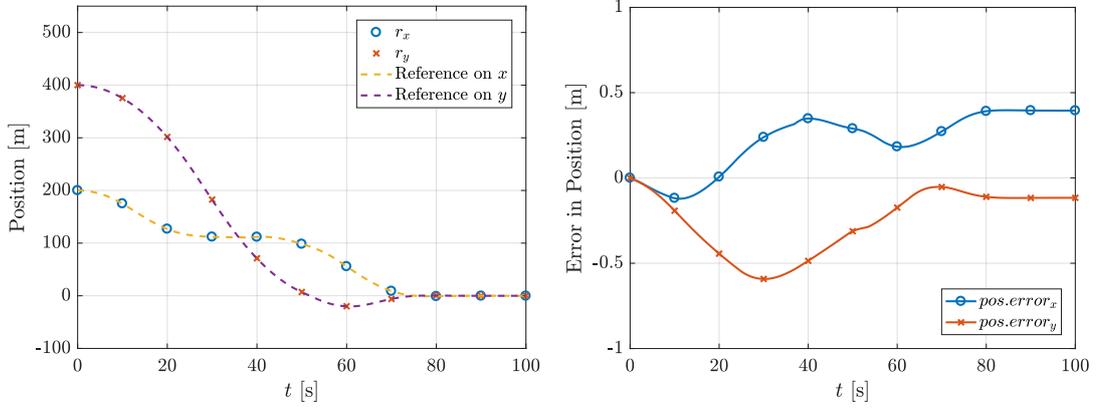

Figure 5.2: Position of chaser vehicle relative to target in *x* and *y* directions while avoiding moving debris, and error in relative position realization.

As may be observed from the figures above, position control is realized quite accurately. Accelerations and torques applied to realize desired trajectories are presented in Figure 5.3. Please note that, since translational motion is on orbital plane, no input in the out of plane (*z*-direction) is observed. It may be concluded from these results that the realized trajectory is quite accurate and acceptable.
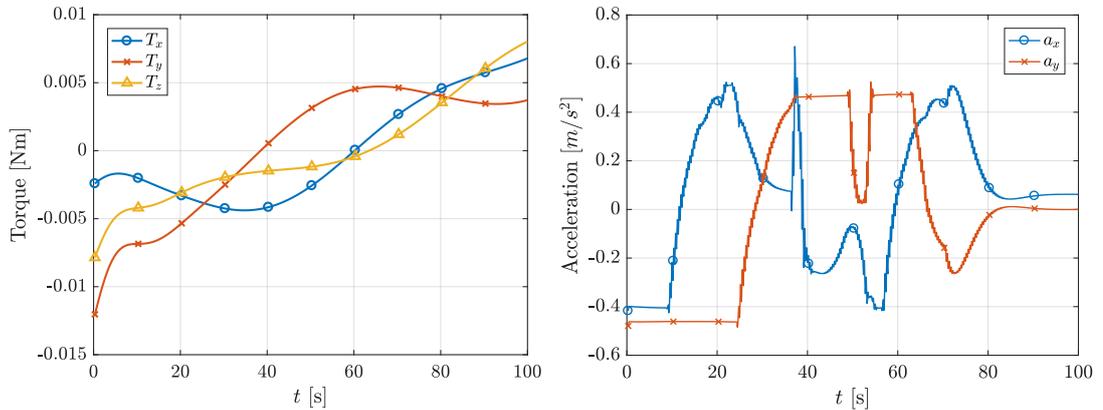

Figure 5.3: Control inputs for relative attitude and position tracking.

It may also be observed from Figure 5.3 that the control accelerations applied to avoid the moving debris are very sharp. However, inputs obey the constraints that are imposed by the actuators limitations.

In Figure 5.4 the specific forces or accelerations generated by the MPC method and dual quaternion based nonlinear tracking controllers are presented in two different directions. It may be observed from these figures that they are quite close to each other.
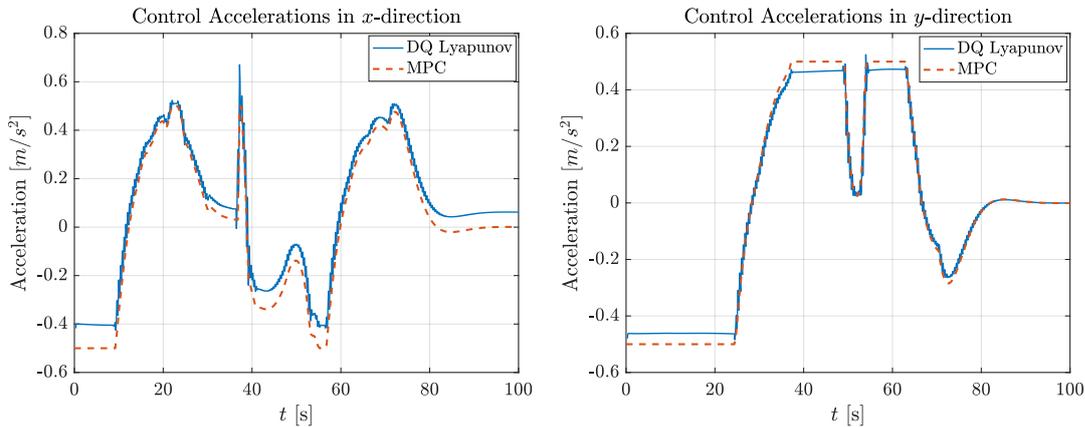
Figure 5.4: Comparison of control inputs of MPC and dual quaternion based nonlinear tracking controllers.

The similarity between two control input sets is evident. MPC control set is smoother which is believed to be because of the awareness of the approaching obstacle predicted far before. MPC also results in slightly lower control effort. However, it is clear that by applying MPC in real time, the computational load would be much bigger than the case for a fixed control policy. The realized trajectory of both approaches and the proof of their similarities are shown in Figure 5.5. Not much difference is observed, and it may be deduced that two different control methods are verified with two different relative motion dynamics representations. All constraints, including debris avoidance, are successfully satisfied in both cases, and no overshoot or constraint violation is observed. Thus, the chaser approaches the target safely within the specified approach cone which was shown in Figure 4.24.
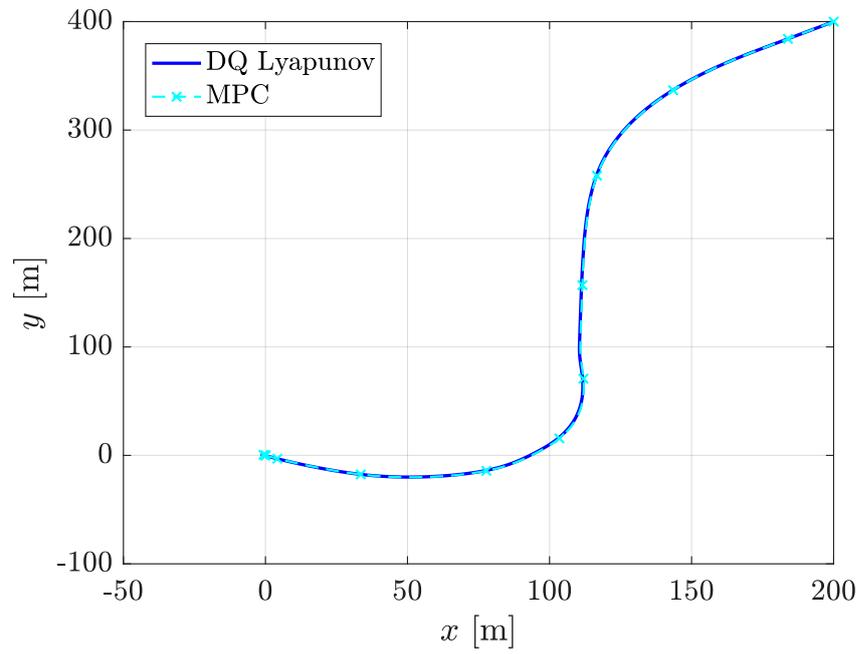
Figure 5.5: Comparison of followed rendezvous paths for MPC and dual quaternion based nonlinear tracking controllers.

CHAPTER 6

CONCLUSION

6.1 Concluding Remarks

In this thesis, spacecraft relative motion planning and control are examined. Rendezvous problem is solved by controlling the chaser spacecraft with the use of Constrained Model Predictive Control approach and simulating relative motion with nonlinear HCW equations. Constraints on both states and inputs arisen from safety and operational considerations are combined, and handled in optimization problem together. Simulation results show that operation is implemented safely in obedience to the constraints imposed such as available input, slow impact at docking, the direction of approach, and collision avoidance constraints. It is demonstrated that constraints on states keep the chaser within the prescribed approach cone without any overshoot, and help to achieve a safe trajectory by avoiding a possible collision with relatively stationary or moving space debris obstacles.

Parametric studies including two separate performance metrics to observe fuel and energy consumptions together with the docking times are carried out for four different mission scenarios with various constraint combinations to find the best prediction horizon length and weighting parameter between state and input weight matrices. Results are tabulated, and best values for each scenario are found. It is observed that unnecessarily large horizons should be avoided because of both computational burden and high fuel consumption.

Another control algorithm that uses dual quaternions to parametrize both attitude and position together is developed with a Lyapunov based nonlinear method. This dual quaternion representation is used to derive a set of equations of combined rotational and translational relative motions as well. Two control approaches with separate relative dynamics representations are compared concerning the relative position accuracy and acceleration inputs. As a result, both simulation codes using different nonlinear sets of equations of relative motion and both control methodologies are verified.

Results of MPC application revealed that the control approach is quite suitable for such an R-D mission that is planned to be implemented autonomously and includes vital safety considerations. Although slow impact and available input considerations are operational constraints that are unavoidable to consider, application of additional safety constraints results in a safe and efficient implementation of the mission.

Approach cone helps chaser vehicle to approach the target within a previously determined approach direction which may be selected based on mission requirements. Indeed, it prevents any possible overshoot, and keeps the chaser inside the prescribed mission path until it touches the target and reaches the steady state in the origin of the LVLH frame.

Obstacle avoidance is crucially important especially for crowded orbits such as LEO's in which thousands of space debris are orbiting. Both options of bounding obstacle in a fixed obstacle zone and considering it as an individual object that is sensed in real-time are considered. Since the former requires to be more conservative in terms of the size of the avoidance region, the latter results in more efficient plans.

Dual quaternion based nonlinear control algorithm successfully tracks the MPC translational path plan. Comparison of input time histories of both control methods shows that MPC application demands slightly less control effort. However, it should be remembered that MPC application consumes much more computational energy compared to the Lyapunov based controller. Controlling both attitude and position is taken as the advantage of dual quaternion parametrization, and an attitude trajectory

plan is also tracked in parallel to the position control. Similar success in attitude tracking is observed as well as in the position tracking. As a result, it may be stated that using MPC in motion planning, and apply Lyapunov based controller as the control application may result in an efficient, safe, and accurate R-D mission implementation.

6.2 Future Directions

This work constitutes an inception to a solution of autonomous rendezvous and docking operation proposed to be applicable to missions such as debris removal. The suggested methodology of control and motion planning is shown to be quite suitable to implement missions with accurate attitude and position trajectory tracking in a safe and efficient manner. Future work may focus on complex attitude control cases in an R-D mission within the close proximity of a tumbling and uncooperative object, i.e., "end game" operation. In this thesis, a safe and fuel-efficient path planning is realized for rendezvous and docking phases, and it is shown that challenging attitude trajectories can be tracked while following rendezvous path accurately. However, it is more complicated than controlling attitude only when the target has an unstable tumbling or wobbling motion. An approach including strategies to plan and control both attitude and position accordingly to reach desired rotational and translational states should be developed. Moreover, thruster placement and number should be considered during this process. In this work, they were assumed to generate enough specific force in each direction.

Another aspect to consider while implementing such operations is the mechanisms to dock or capture the target, and the collaboration of attitude and position controllers onboard with these mechanisms. Robot arms and tethering mechanisms may be given as examples to these mechanisms. Controlling the motions of such capture mechanisms is another topic that needs to be investigated in R-D applications.

REFERENCES

[1] A. B. Bosse, W. J. Barnds, M. A. Brown, N. G. Creamer, A. Feerst, C. G. Henshaw, A. S. Hope, B. E. Kelm, P. A. Klein, F. Pipitone, B. E. Plourde and B. P. Whalen, "SUMO: Spacecraft for the universal modification of orbits," in *Spacecraft Platforms and Infrastructure*, Aug 2004..

[2] S. Eberle, A. Ohndorf and R. Faller, "On-Orbit Servicing Mission Operations at GSOC," in *SpaceOps 2010 Conference*, 2010.

[3] R. Burns, C. Sabol and C. A. McLaughlin, "Satellite Formation Flying Design and Evolution," *Journal of Spacecraft and Rockets,* 2001.

[4] J. A. Starek, B. Acikmese, I. A. Nesnas and M. Pavone, "Spacecraft Autonomy Challenges for Next-Generation Space Missions," 2013.

[5] D. Geller, "Orbital rendezvous: When is autonomy required?," *Journal of Guidance, Control, and Dynamics,* vol. 30, no. 4, p. 974–981, 2007.

[6] P. R'egnier, C. Koeck, X. Sembely, B. Frapard, M. Parkinson and R. Slade, "Rendezvous GNC and system analyses for the Mars Sample Return mission," in *56th Int. Astronautical Congress*, Japan, Oct 17–21 2005.

[7] D. P. Scharf, F. Y. Hadaegh and S. R. Ploen, "A survey of spacecraft formation flying guidance and control. Part II: control," in *American Control Conference*, 2004.

[8] D. P. Scharf, F. Y. Hadaegh and B. H. Kang, "A Survey of Spacecraft Formation Flying Guidance," in *International Symposium Formation Flying*, 2002.

[9] J. T. Betts, "Survey of Numerical Methods for Trajectory Optimization. , 21(2):193–207, 1998.," *Journal of Guidance, Control, and Dynamics,* vol. 21, no. 2, pp. 193-207, 1998.

[10] D. C. Woffinden and D. K. Geller, "The road to autonomous orbital rendezvous," *Advances in the Astronautical Sciences,* vol. 129, p. 2395–2415, 2008.

[11] J. R. Wertz and R. Bell, "Autonomous Rendezvous and Docking Technologies: Status and Prospects," in *SPIE AeroSense Symposium*, Aug 2003.

[12] A. Flores-Abad, O. Ma, K. Pham and S. Ulrich, "A review of space robotics technologies for on-orbit servicing.," *Progress in Aerospace Sciences,* vol. 68, pp. 1-26, 2014.

[13] Y. Luo, J. Zhang and G. Tang, "Survey of orbital dynamics and control of space rendezvous," *Chinese Journal of Aeronautics,* vol. 27, no. 1, pp. 1-11, 2014.

[14] D. Zimpfer, P. Kachmar and S. Tuohy, "Autonomous Rendezvous, Capture and In-Space Assembly: Past, Present and Future," in *1st Space Exploration Conference: Continuing the Voyage of Discovery*, Orlando ,FL, Jan 2005.

[15] W. Fehse, Introduction to Automated Rendezvous and Docking of Spacecraft, Cambridge University Press, 2003.

[16] W. Clohessy and R. Wiltshire, "Terminal Guidance System for Satellite Rendezvous," *Journal of the Aerospace Sciences,* vol. 27, no. 9, pp. 653-658, 1960.

[17] L. Breger and J. How, "Safe trajectories for autonomous rendezvous of spacecraft," *Journal of Guidance, Control, and Dynamics,* vol. 31, no. 5, 2008.

[18] K. T. Alfriend, S. R. Vadali, P. Gurfil, J. P. How and L. S. Breger, "Spacecraft Formation Flying," *Elsevier Astrodynamics Series,* 2010.

[19] J. Goodman, "History of space shuttle rendezvous and proximity operations," *Jounal of Spacecraft and Rockets,* vol. 43, no. 5, pp. 944-956, 2006.

[20] Y. Luo and G. Tang, "Spacecraft optimal rendezvous controller design using simulated annealing," *Aerospace Science and Technology,* vol. 9, pp. 732-737, 2005.

[21] A. Miele, M. Weeks and M. Circia, " Optimal trajectories for spacecraft renrdezvous," *Journal of Optimization Theory and Applications,* vol. 132, no. 3, pp. 353-376, 2007.

[22] E. Kong, S. Nolet and D. Miller, "Autonomous docking algorithm development and experimentation using the SPHERES testbed," in *SPIE Defense and Security*, 2004.

[23] M. Polites, "Technology of automated rendezvous and capture in space," *Jounal of Spacecraft and Rockets,* vol. 36, no. 2, pp. 280-291, 1999.

[24] E. N. Hartley, "A tutorial on model predictive control for spacecraft rendezvous," in *European Control Conference (ECC)*, Linz, Austria, 2015.

[25] G. W. Hill, "Researches in the Lunar Theory," *American Journal of Mathematics,* vol. 1, pp. 5-26, 1878.

[26] F. Gavilan, R. Vazquez and E. F. Camacho, "Chance-constrained Model Predictive Control for Spacecraft Rendezvous with Disturbance Estimation," *Control Engineering Practice,* 2012.

[27] B. Wie, Space Vehicle Dynamics and Control, 1998.

[28] O. Montenbruck and S. Leung, "Real-Time Navigation of Formation-Flying Spacecraft Using Global-Positioning-System Measurements," *Journal of Guidance, Control and Dynamics,* 2005.

[29] P. J. Nicklasson and R. Kristiansen, "Spacecraft formation flying: A review and new results on state feedback control," *Acta Astronautica,* 2009.

[30] N. Martinson, J. Munoz and G. Wiens, "A new method of guidance control for autonomous rendezvous in a cluttered space environment," in *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, August 2007.

[31] N. Martinson, "Obstacle avoidance guidance and control algorithms for spacecraft maneuvers," in *AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, 2009.

[32] A. Richards and J. How, "Performance evaluation of rendezvous using model predictive control," in *AIAA Guidance, Navigation, and Control Conference*, 2003.

[33] L. Singh, S. Bortolami and L. Page, "Optimal guidance and thruster control in orbital approach and rendezvous for docking using model predictive control," in *AIAA Guidance, Navigation, and Control Conference*, 2010.

[34] M. Holzinger, J. Di Matteo, J. Schwartz and M. Milam, "Passively safe receding horizon control for satellite proximity operations," in *47th IEEE Conf. on Decision and Control*, 2008.

[35] R. Larsson, S. Berge, P. Bodin and U. Jönsson, "Fuel efficient relative orbit control strategies for formation flying and rendezvous within PRISMA," *Advances in the Astronautical Sciences,* vol. 125, pp. 25-40, 2006.

[36] P. Bodin, R. Larsson, F. Nilsson, C. Chasset, R. Noteborn and M. Nylund, "PRISMA: An In-Orbit Test Bed for Guidance, Navigation, and Control Experiments," *Journal of Spacecraft and Rockets,* vol. 46, no. 3, p. 615–623, 2009.

[37] S. D'Amico, P. Bodin, M. Delpech and R. Noteborn, Distributed Space Missions for Earth System Monitoring, New York, NY: Springer, 2013.

[38] S. Kawamoto, T. Makida, F. Sasaki, Y. Okawa and S.-İ. Nishida, "Precise numerical simulations of electrodynamic tethers for an active debris removal system," *Acta Astronautica,* vol. 59, no. 5, p. 139–148, 2006.

[39] S.-I. Nishida, S. Kawamoto, Y. Okawa, F. Terui and S. Kitamura, "Space debris removal system using a small satellite," *Acta Astronautica,* vol. 65, no. 1, pp. 95-102, 2009.

[40] J. C. Liou, N. L. Johnson and N. M. Hill, "Controlling the growth of future LEO debris populations with active debris removal," *Acta Astronautica,* vol. 66, no. 5, p. 648–653, 2010.

[41] S. M. LaValle, Planning Algorithms, Cambridge University Press, 1996.

[42] M. H. Maia and R. K. H. Galvao, "On the use of mixed-integer linear predictive control with avoidance constraints," *International Journal of Robust and Nonlinear Control,* vol. 19, p. 822–828, 2008.

[43] R. Epenoy, "Fuel optimization for continuous-thrust orbital rendezvous with collision avoidance constraint," *Journal of Guidance, Control, and Dynamics,* vol. 34, no. 2, 2011.

[44] C. Ranieri, "Path-Constrained Trajectory Optimization for Proximity Operations," in *AIAA/AAS Astrodynamics Specialist Conference*, Honolulu, HI, August 2008.

[45] Y. Ulybyshev, "Trajectory optimization for spacecraft proximity operations with constraints," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, August 2011.

[46] J. B. Mueller, P. R. Griesemer and S. Thomas, "Avoidance maneuver planning incorporating station-keeping constraints and automatic relaxation," *Journal of Aerospace Information Systems,* vol. 10, no. 6, pp. 306-322, 2013.

[47] A. Richards, T. Schouwenaars, J. How and E. Feron, "Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming," *AIAA Journal of Guidance, Control, and Dynamics,* vol. 25, no. 4, 2002.

[48] S. Di Cairano, H. Park and I. Kolmanovsky, "Model Predictive Control approach for guidance of spacecraft rendezvous and proximity maneuvering," *INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL,* vol. 22, p. 1398–1427, 2012.

[49] T. E. Carter, "State Transition Matrices for Terminal; Rendezvous Studies: Brief Survey and New Example," *Journal of Guidance,Control and Dynamics,* vol. 21, no. 1, pp. 148-155, 1998.

[50] P. K. Seidelmann and S. E. Urban, Explanatory Supplement to the Astronomical Almanac, Mill Valley, CA: University Science Books, 2013.

[51] G. Inalhan, M. Tillerson and J. P. How, "Relative dynamics and control of spacecraft formations in eccentric orbits," *Journal of Guidance, Control, and Dynamics,* vol. 25, no. 1, pp. 48-59, 2002.

[52] M. Humi and T. Carter, "Orbits and Relative Motion in the Gravitational Field of an Oblate Body," *Journal of Guidance Control and Dynamics ,* vol. 31, pp. 522-532, 2008.

[53] I. M. Ross and F. Fahroo, "Issues in the real-time computation of optimal control," *Mathematical and Computer Modelling,* vol. 43, no. 9-10, pp. 1172-1188, 2006.

[54] G. Hull, "Conversion of Optimal Control Problems into Parameter Optimization Problems," *Journal of Guidance, Control, and Dynamics,* vol. 20, no. 1, pp. 57-60, 1997.

[55] J. Vlassenbroeck and R. V. Dooren, "A Chebyshev Technique for Solving Nonlinear Optimal Control Problems," *IEEE Transactions on Automatic Control,* vol. 33, no. 4, pp. 333-340, 1988.

[56] J. A. Rossiter, Model-Based Predictive Control A Practical Approach, CRC Press, 2013.

[57] S. Keerthi and E. G. Gilbert, "Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations," *Journal of optimization theory and applications,* vol. 57, no. 2, p. 265–293, 1988.

[58] C. Jewison, "Guidance and Control for Multi-stage Rendezvous and Docking (Doctoral dissertation)," 2017.

[59] B. Wie, H. Weiss and A. Arapostathis, "Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotation," *Journal of Guidance, Control, and Dynamics,* vol. 12, no. 3, 1989.

[60] O. Tekinalp and A. Tekinalp, "Tracking Control of Spacecraft Attitude on Time Dependent Trajectories," in *AIAA/AAS Astrodynamics Specialist Conference*, Long Beach, CA, Sep 13-16, 2016.

[61] U. Lee and M. Mesbahi, "Dual Quaternion based Spacecraft Rendezvous with Rotational and Translational Field of View Constraints," in *AIAA/AAS Astrodynamics Specialist Conference*, San Diego, CA, August 4-7, 2014.

[62] N. Filipe and P. Tsiotras, "Adaptive Position and Attitude-Tracking Controller for Satellite Proximity Operations Using Dual Quaternions," *Journal of Guidance, Control, and Dynamics,* vol. 38, no. 4, pp. 566-677, 2015.

[63] J. Wang and Z. Sun, "6-DOF robust adaptive terminal sliding mode control for spacecraft formation flying," *Acta Astronautica,* vol. 73, pp. 676-687, 2012.

[64] A. Caubet and J. D. Biggs, "Optimal Attitude Motion Planner for Large Slew Maneuvers Using a Shape-Based Method," in *AIAA Guidance Navigation Control Conference*, Boston, MA, August 19-22, 2013.

[65] K. D. Bilimoria and B. Wie, "Time-Optimal Three-Axis Reorientation of a Rigid Spacecraft," *Journal of Guidance, Control, and Dynamics,,* vol. 16, no. 3, pp. 446-452, 1993.

[66] G. A. Boyarko, M. Romano and O. A. Yakimenko, "Time-Optimal Reorientation of a Spacecraft Using an Inverse Dynamics Optimization Method," *Journal of Guidance, Control, and Dynamics,* vol. 16, no. 3, pp. 1197-1207, 2011.