An Evolutionary Framework for Multi-objective Trajectory Design and Robust Model Predictive Control in Long-range Rendezvous Missions

by

Sanaz Samsam, B.Sc.

A thesis submitted to the
Faculty of Graduate and Postdoctoral Affairs
in partial fulfillment of the requirements for the degree of

Master of Applied Science in Aerospace Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering
Department of Mechanical and Aerospace Engineering
Carleton University
Ottawa, Ontario
November, 2021

©Copyright
Sanaz Samsam, 2021
The undersigned hereby recommends to the
Faculty of Graduate and Postdoctoral Affairs
acceptance of the thesis

An Evolutionary Framework for Multi-objective Trajectory
Design and Robust Model Predictive Control in Long-range
Rendezvous Missions

submitted by Sanaz Samsam, B.Sc.

in partial fulfillment of the requirements for the degree of

Master of Applied Science in Aerospace Engineering

________________________
Professor Natallie Baddour, External Examiner

________________________
Professor Steve Ulrich, Internal Examiner

________________________
Professor Robin Chhabra, Thesis Supervisor

________________________
Professor Ronald Miller, Chair,
Department of Mechanical and Aerospace Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering
Department of Mechanical and Aerospace Engineering
Carleton University
November, 2021
Abstract

This thesis presents an optimization architecture for on-orbit servicing mission design in the long-range rendezvous phase. We develop a methodology to generate Pareto Optimal trajectories for long-range rendezvous of a servicing satellite with a moving target. The methodology employs a multi-impulse shape-based trajectory planning algorithm for in-plane orbit transfer, based on the two-body problem. We first derive the necessary and sufficient conditions that determine the set of smooth impulsive trajectories connecting the servicing satellite to the orbiting target. The Pareto Optimal trajectories from this set are then obtained using a constrained multi-objective optimization algorithm developed based on the Non-dominated Sorting Genetic Algorithm-II (NSGA-II). Transfer time and control effort are the two Pareto cost functions that are considered in the multi-objective optimization. To reduce the risk of collision in populated orbits and to remain in an orbital regime, we include restrictions on orbital elements as part of the constraints. Further, a maximum available impulse is considered as an upper-bound for velocity changes in an impulsive trajectory. The number of impulses along with the location of the first impulse in the parking orbit and the orbital parameters of the intermediate orbits form the set of design variables. We demonstrate the superiority of the developed trajectory planner by comparing its results with those obtained from another multi-objective evolutionary algorithm called the Multi-Objective Genetic Algorithm and an optimal Lambert approach. In an on-orbit servicing mission, a solution from the Pareto frontier set of optimal trajectories may be selected based on the mission requirements.

To robustly follow the generated reference trajectories in a $J_2$-perturbed orbital environment, we propose a Nonlinear Model Predictive Control (NMPC) scheme. The control signals are velocity increments at the time of applying each impulse, and the variable horizon is considered to be the time difference between every two impulses in the reference trajectory. To avoid singularities, the equinoctial orbital elements are used in the process model that includes the secular and first-order long-periodic effects
of the $J_2$-perturbation. The proposed objective function for the NMPC includes a norm of error between the reference and predicted satellite’s trajectory under $J_2$ perturbation, and a norm of control signals. To arrive near the target at the end of the transfer, the fixed-final-time, free-final-state optimization problem in the preceding to the last horizon is converted to a constrained free-final-time problem in the last horizon. The constraint is the final distance between the servicer and the target to ensure the chasing capability of the servicer (with a threshold of 10 km). Further, to demonstrate the robustness of the proposed controller, we model the following phenomena in the dynamic simulation of the servicer: (i) the first-order short-periodic effects due to the $J_2$ perturbation, (ii) a bounded uncertain acceleration to capture unmodelled dynamics, and (iii) application of the velocity increment over a short period of time. Finally, to decrease the computational demand due to the optimization process, an evolutionary optimization algorithm is used and embedded in the NMPC scheme. Simulation results are provided to illustrate the tracking effectiveness of the developed controller.
To my mother, Homa, who I would not be where I am today without her sacrifices.
Acknowledgments

First and foremost I would like to thank my supervisor Dr. Robin Chhabra for his guidance, valuable ideas and contributions towards the completion of my thesis. I could have not asked for a better supervisor to guide me throughout my graduate studies. I would like to thank my mother Homa and my grandparents for all they have done for me throughout my life. I would have not been where I am today without their hard work, kindness, and sacrifices. I would like to thank all my friends for their support and encouragement and to all the teachers and professors that I have had the privilege to be educated by. For them believing in me and pushing me to become the person I am today. Last but not least, I would like to thank my best friend for his love, supports, and patience. This research was financially supported in part by Natural Sciences and Engineering Research Council of Canada through the Canada Graduate Scholarship as well as Carleton University and the Department of Mechanical and Aerospace Engineering entrance scholarship and Teaching Assistantships.
# Table of Contents

Abstract iii  
Acknowledgments vi  
Table of Contents vii  
List of Tables xi  
List of Figures xii  

1 Introduction 1  
1.1 Motivation 1  
1.2 Thesis Objectives 2  
1.3 Statement of Contributions 3  
1.4 Thesis Organization 4  

2 Literature Review 6  
2.1 Optimal Trajectory Design 6  
2.1.1 Model 6  
2.1.2 Objective Function 8  
2.1.3 Approach and Solution 9  
2.2 $J_2$-Perturbed Orbits 10  
2.3 Model Predictive Control of Satellites 11  

3 Preliminaries 14  
3.1 Summary 14  
3.2 Classical Orbital Elements 14  
3.3 Equinoctial Orbital Elements 16
### 3.3 Conversion from Classical Elements to Equinoctial Elements

- Conversion from Classical Elements to Equinoctial Elements: 17
- Conversion from Equinoctial Elements to Classical Elements: 18
- Conversion from Classical Elements to Cartesian Coordinates: 18
- Conversion from Equinoctial Elements to Cartesian Coordinates: 19

### 3.4 J₂-Perturbed Motion of Satellites

- Central-Body Gravitational Potential: 21
- Averaged Equations of Motion: 24
- Approximation of Osculating Elements: 27

### 3.5 Multi-Objective Optimization

- Decision Variables: 29
- Constraints: 30
- Objective Functions: 30
- Multi-Objective Optimization Problem: 30
- Pareto Optimality: 31

### 4 Smooth Trajectories for Target Chasing

- Summary: 33
- Multi-Impulse Smooth Trajectories: 33
- Chasing a Target Satellite: 36
- Conclusion: 39

### 5 Trajectory Design Optimization

- Summary: 40
- Transfer Trajectory Optimization Problem: 40
  - Design Variables: 41
  - Constraints: 42
  - Cost Functions: 43
  - Optimization Problem: 43
- Non-Dominated Sorting Genetic Algorithm: 44
  - Initialization: 45
  - Cost Function Evaluation: 45
  - Sorting Operators: 46
  - Binary Tournament Selection: 49
  - Arithmetic Crossover: 49
  - Mutation: 49
List of Tables

6.1 Summary of the optimal results; Case Study 1 ........................................ 55
6.2 Summary of the optimal results; Case Study 2 ........................................ 58
6.3 Summary of the optimal results; Case Study 3 ........................................ 62
8.1 Information summary of the reference trajectory; Case Study 1 ................. 78
8.2 Information summary of the simulated trajectory; Case Study 1 ............... 79
8.3 Information summary of the reference trajectory; Case Study 2 ............... 83
8.4 Information summary of the simulated trajectory; Case Study 2 ............... 83
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Spacecraft trajectory optimization process</td>
<td>7</td>
</tr>
<tr>
<td>2.2</td>
<td>Velocity increments applied to a trajectory</td>
<td>8</td>
</tr>
<tr>
<td>2.3</td>
<td>Osculating orbital elements</td>
<td>12</td>
</tr>
<tr>
<td>3.1</td>
<td>ECI and local-vertical-local-horizontal reference frame</td>
<td>15</td>
</tr>
<tr>
<td>3.2</td>
<td>Classical Orbital Elements</td>
<td>16</td>
</tr>
<tr>
<td>3.3</td>
<td>Equinoctial reference frame</td>
<td>17</td>
</tr>
<tr>
<td>4.1</td>
<td>Oblique ellipse parameters in polar coordinate system</td>
<td>34</td>
</tr>
<tr>
<td>5.1</td>
<td>Multi-objective optimization architecture</td>
<td>45</td>
</tr>
<tr>
<td>5.2</td>
<td>Non-dominated sorting ranking.</td>
<td>48</td>
</tr>
<tr>
<td>5.3</td>
<td>Crowding distance for two objectives.</td>
<td>48</td>
</tr>
<tr>
<td>5.4</td>
<td>Mutation probability</td>
<td>50</td>
</tr>
<tr>
<td>5.5</td>
<td>Inverted Generational Distance</td>
<td>50</td>
</tr>
<tr>
<td>6.1</td>
<td>Case Study 1; (a) Unconstrained Pareto frontier (b) IGD$^+$ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier</td>
<td>56</td>
</tr>
<tr>
<td>6.2</td>
<td>Case Study 1; Dynamics of the optimal solution closest to the ideal point</td>
<td>57</td>
</tr>
<tr>
<td>6.3</td>
<td>Case Study 2; (a) Unconstrained Pareto frontier (b) IGD$^+$ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier</td>
<td>59</td>
</tr>
<tr>
<td>6.4</td>
<td>Case Study 2; Dynamics of the optimal solution closest to the ideal point</td>
<td>60</td>
</tr>
<tr>
<td>6.5</td>
<td>Case Study 3; (a) Unconstrained Pareto frontier (b) IGD$^+$ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier</td>
<td>61</td>
</tr>
<tr>
<td>6.6</td>
<td>Case Study 3; Dynamics of the optimal solution closest to the ideal point</td>
<td>62</td>
</tr>
<tr>
<td>7.1</td>
<td>Impulsive control signals</td>
<td>65</td>
</tr>
<tr>
<td>7.2</td>
<td>Block diagram of the closed-loop system</td>
<td>71</td>
</tr>
<tr>
<td>7.3</td>
<td>Single-objective genetic optimization architecture</td>
<td>73</td>
</tr>
</tbody>
</table>
7.4 Mutation probability ................................................. 75
8.1 Case Study 1; (a) Reference trajectory (b) Control signal (c) Projected 2-D transfer trajectory (d) 3-D transfer trajectory (e) Velocity increment differences from the reference (ordered sequentially in $x$, $y$, $z$ direction from top to bottom) ........................................... 80
8.2 Case Study 1; (a) Variation of $F_1$ (b) Variation of $F_2$ (c) Variation of $F_3$ (d) Variation of $F_4$ (e) Distance from the target ................................. 81
8.3 Case Study 2; (a) Reference trajectory (b) Control signal (c) Projected 2-D transfer trajectory (d) 3-D transfer trajectory (e) Velocity increment differences from the reference (ordered sequentially in $x$, $y$, $z$ direction from top to bottom) ........................................... 84
8.4 Case Study 2; (a) Variation of $F_1$ (b) Variation of $F_2$ (c) Variation of $F_3$ (d) Variation of $F_4$ (e) Distance from the target ................................. 85
Chapter 1

Introduction

1.1 Motivation

The accomplishments of the On-Orbit Servicing (OOS) missions in the past two decades corroborate their significance in space exploration and exploitation. These missions provide the operating satellites with a range of services, such as, refueling, repairing, upgrading, orbit modification, assembly, and debris removal. Such operations intend to increase the satellites’ lifetime and enhance their performance, as well as to reduce the operating costs associated with space programs. Examples of manned OOS missions are repairing the Hubble Space Telescope and assembling the International Space Station [1]. The world’s first unmanned OOS mission was conducted by Engineering Test Satellite (ETS-VII), which was equipped with a deployable robotic arm, to demonstrate the capability of docking to a target and the utilization of robotic technologies [2]. Other successful unmanned OOS missions have been performed. A series of the Experimental Spacecraft Systems (XSS-10 and XSS-11) demonstrated satellite-oriented servicing capabilities [3] in 2003. In 2005, the Demonstration of Autonomous Rendezvous Technology (DART) mission launched with the aim of supporting autonomous rendezvous and proximity operations [1], including station keeping, docking approach, circumnavigation, and collision avoidance maneuver. Subsequently, the Defense Advanced Research Projects Agency (DARPA) successfully executed the Orbital Expressed mission in 2007 and demonstrated an end-to-end robotic satellite servicing activity. The mission included autonomous rendezvous, capture, maintenance, and servicing of an experimental satellite [1]. The most recent attempt that will be commissioned in 2022 is On-orbit Servicing, Assembly, and Manufacturing 1 (OSAM-1) that is a mission to extend satellites’ lifespan
even if they are not designed to be serviced on orbit [4]. An autonomous servicing satellite (hereafter called servicer) that is equipped with robotic arms can perform valuable services, e.g., visual inspection, refueling, and debris removal.

The highly increasing demand for OOS operations suggests the necessity to move towards industrialization, whose main challenge is the design of an efficient mission architecture for continuous supply of multiple services to a number of target satellites (hereafter called target). The phase of a servicing mission that mostly contributes to fuel and time consumption is the long-range rendezvous [5]. Further, considering the target’s orbital motion in the long-range rendezvous will provide solutions that can decrease the duration and the required fuel in the short-range rendezvous. As a result, the design of the long-range rendezvous phase must be comprehensively studied as one of the crucial steps in the industrialization process. Long-range rendezvous in two-body context has been studied widely in the literature [6, 7, 8]. However, neglecting the effect of environmental perturbations results in inaccurate long-range rendezvous trajectory designs [9, 10, 11]. Hence, there is a need to design a controller so that the servicer be able to follow the designed trajectory in a perturbed environment.

1.2 Thesis Objectives

The long-range rendezvous phase of an OOS mission is studied in this work. The objectives of this thesis are to develop:

1. Transfer trajectories for chasing a dynamic target with minimum transfer time and control effort. The aim is to use multi-impulse smooth transfer trajectories for rendezvous with and chase a target. An evolutionary multi-objective optimization must be in place to obtain optimal trajectories considering both transfer time and control effort. The developed optimization framework should also be able to tackle constraints, e.g., avoiding highly populated orbital regions and limiting control effort during an impulse.

2. A nonlinear model predictive control to follow the proposed optimal transfer trajectories in perturbed environments. The controller must be able to both follow the multi-impulse smooth trajectory and arrive at the target at the final time of mission with a distance smaller than 10 km.
1.3 Statement of Contributions

The main contributions of this thesis can be summarized in the following.

1. In the first phase, we develop a multi-impulse shape-based transfer trajectory generator for chasing a target in the long-range rendezvous phase of an OOS mission. We propose a multi-objective constrained optimization architecture to minimize both the transfer time and control effort, using the concept of Pareto Optimality. Considering a dynamic target as apposed to a fixed landing point in multi-impulse trajectory design is addressed for the first time in this thesis.

2. The proposed architecture is based on the constrained multi-objective NSGA-II. Smoothness of the generated trajectories and avoiding some highly populated orbital regions are among the constraints that are considered in this phase. One journal publication in Aerospace Science and Technology and two conference papers were produced from this phase of the thesis.


   • S. Samsam, R. Chhabra, "Multi-impulse smooth trajectory design for long-range rendezvous with an orbiting target using multi-objective non-dominated sorting genetic algorithm”, Aerospace Science and Technology, manuscript accepted.

   • S. Samsam, R. Chhabra, "Multi-impulse shape-based trajectory optimization for target chasing in on-orbit servicing missions”, IEEE Aerospace Conference, 2021


3. In the second phase, a real-time Nonlinear Model Predictive Control (NMPC) is developed for a constrained spacecraft trajectory tracking problem specifically proposed for a long-range rendezvous mission. The objective function of the NMPC, which is a finite horizon optimal control, is a combination of
control efforts, tracking errors, and the final distance from the target. We embed an optimization technique based on the genetic algorithm in the NMPC framework to control impulsive trajectories. To avoid singularities of circular and/or equatorial orbits, we represent the orbital motion in the equinoctial elements. To demonstrate the robustness of the developed controller, we simulate the closed-loop system in a \( J_2 \)-perturbed environment with some orbital disturbances. The results of this phase was submitted for publication in Aerospace Science and Technology.


1.4 Thesis Organization

The remainder of this thesis are organized as follows.

Chapter 2 first presents a brief background and state-of-the-art in spacecraft trajectory design and optimization, including different implemented models, objective functions, and solutions. Then, recent developments in \( J_2 \)-perturbed orbits and control design for satellites are discussed.

Chapter 3 provides the reader with a variety of concepts that are important to the thesis. This chapter introduces equinoctial orbital elements and its conversions to different orbital elements. This is followed by the \( J_2 \)-perturbed equations of motion of satellites represented in equinoctial orbital elements, and equations of averaging. The other topic that is discussed in this chapter is the fundamental definitions for multi-objective optimization algorithms.

Chapter 4 derives a set of constraint equations to generate multi-impulse smooth transfer trajectories for a servicer chasing a target in the long-range rendezvous phase. The motion of the servicer and the target is formulated based on the two-body problem.

In Chapter 5, a multi-objective constrained optimization architecture is proposed to minimize both the transfer time and control effort, using the concept of Pareto Optimality. The proposed architecture is based on the constrained multi-objective NSGA-II.
Chapter 6 evaluates the proposed $N$-impulse optimal smooth trajectory generator in three case studies. To examine the efficiency of the proposed optimization algorithm, the results of NSGA-II is compared with that of the MOGA and the optimal Lambert approach developed for chasing a satellite, in this thesis.

In Chapter 7, a nonlinear model predictive control is proposed in order to follow a reference trajectory and catch a target satellite at the final time of the mission. To solve the optimization problem that is defined in the controller, the genetic algorithm is implemented.

Chapter 8 aims to evaluate the efficacy of the proposed nonlinear model predictive control to follow the reference trajectory and catch the target in two case studies. This chapter also studies the convergence of the implemented genetic algorithm for both of the case studies.

Finally, Chapter 8 includes some concluding remarks and possible future directions of this thesis.
Chapter 2

Literature Review

2.1 Optimal Trajectory Design

Various approaches for addressing the spacecraft trajectory optimization problem have been reviewed in [6, 12, 13]. In a survey by Shirazi et al. different aspects of the satellite transfer trajectory design are separately studied, including modeling, objective functions, and solutions [6]. In the following, the relevant research works conducted in these three aspects are surveyed in detail. Figure 2.1 is the visual illustration of different approaches to the trajectory optimization problem.

2.1.1 Model

In the design of transfer trajectories, two key characteristics of the spacecraft model must be considered: transfer type and equations of motion.

Transfer Type

Modelling the shape of the control input in an orbit transfer is an important step in trajectory optimization. Impulsive [14] and continuous [15] maneuvers are the two typical transfer types that enjoyed more attention in the literature. Although continuous transfer type captures a more accurate model of spacecraft motion, it adds to the complexity of the optimization problem, as the system always experiences non-zero inputs. On the other hand, the impulsive model reduces the computational effort, with an almost negligible compromise in the optimality of the objective function. Low specific impulse or high thrust level engines execute the impulsive maneuvers
that generate conic arcs as the result of the velocity increment $\Delta v$ at the impulse location [16]. Specifically when $\Delta v$ is tangent to the path of motion, the generated trajectory is considered smooth. A euphemistic illustration of planar conic arcs generated as the result of different velocity increments is depicted in Figure 2.2. Of the impulsive maneuvers, Hohmann transfer between two circular orbits is the most popular transfer method in which a two-impulse trajectory is generated. As another popular two-impulse transfer trajectory, one can refer to the Lambert transfer with the ability to generate a transfer orbit between any two arbitrary spatial points with a desire time interval. Improved versions of this approach have also been suggested in the literature, considering multiple orbital revolutions [17], orbital perturbations [18], and optimal Lambert solutions [19]. Beside the Lambert and Hohmann, shape-based approaches are other popular trajectory design methods for both impulsive and continuous maneuvers that work based on the trajectory’s geometry. The distinguished feature of shape-based approaches is having fewer design variables comparing to their counterparts that do not consider constraints. Shakouri et al. propose a set of constraint equations to generate multi-impulse shape-based transfer trajectories between eccentric orbits in a central gravity [20]. A favorite geometry for designing continuous
shape-based maneuvers is the use of spiral functions [21].

**Equations of Motion**

Motion of the Earth’s satellites is dominantly governed by the central gravity of the Earth (two-body problem). In addition, other factors, such as the gravity of other celestial objects [22], Earth oblateness [23], atmospheric drag [24], solar radiation pressure [25], etc., may be considered in the form of disturbing effects in the equations of motion. The magnitude of these disturbing forces changes with the satellite’s orbital altitude. For example, the dominant disturbances in the Low Earth Orbital (LEO) regime are perturbations induced by Earth’s oblateness (short periodic, long periodic, and mean change) and atmospheric drag [24]. Including such effects in the equations of motion depends on the application [26]. For example, in a rendezvous mission in LEO, comprising the two phases of long-range and short-range, the effects of the long-periodic, mean changes, and atmospheric drag dominate those of the short periodic in the long-range rendezvous [9]. On the other hand, in the short-range rendezvous relative maneuvers with respect to target need be designed that require including relative equations of motion for both position and attitude dynamics [27].

### 2.1.2 Objective Function

Another step in spacecraft trajectory optimization process is defining objectives based on the mission requirements. Objective functions (or cost functions) may include fuel mass, total velocity increment, state errors, transfer time, or control effort. For a general spacecraft trajectory optimization, the objective function is formulated as
follows:

\[ J(x, u, t) = h(x(t_f), t_f) + \int_{t_0}^{t_f} H(x(t), u(t), t)dt, \]

where \( t_0, t_f, u, \) and \( x \) are the initial and final time, control effort, and system state, respectively. The function \( h \) represents the Mayer term, which demonstrates the cost related to the final states, and \( H \) is referred to as the Lagrange term or the accumulated cost associated with the states and control efforts in time \([28]\).

### 2.1.3 Approach and Solution

Solutions of the trajectory optimization problem are divided in two main categories: analytical and numerical approaches. Analytical approaches work based on the optimal control theory to determine a time history of control signals that minimizes a cost function, while satisfying a set of constraints \([29]\). Although the analytical approaches generate solutions with zero approximation, they are not necessarily attainable especially when the complexity of the model and the problem increases. As an alternative, numerical approaches are used to solve the satellite trajectory optimization problem, which are divided in two categories of direct and indirect methods \([30]\). The former attempts to find the minimum cost function over the states and inputs of the system, and the latter numerically solves Pontryagin’s minimum principle. Direct methods are more popular than indirect methods because of the ease of implementation, larger domain of convergence, and smaller problem size, indeed with some compromises in accuracy. Both methods attempt to minimize cost functions and constraint violations using discrete approximations based on some gradient- or heuristic-based approaches \([31]\). The high sensitivity of gradient-based methods to the initial guess of all system parameters renders them less desirable. Heuristic approaches offer alternative solutions to spacecraft trajectory optimization \([19, 32]\), whose examples include the Genetic Algorithm (GA) \([33, 34]\), the particle warm optimization \([35]\), and the simulated annealing \([34, 36]\).

Several methods exist to solve multi-objective optimization problems that can be divided in two general categories of decomposition and heuristic methods. In decomposition methods, a multi-objective optimization problem is converted to a single-objective problem \([37, 38, 39, 40, 41]\). Unlike heuristic methods, decomposition algorithms must be run several times to find a set of Pareto-optimal solutions. This makes heuristic methods faster and more reliable, comparing to decomposition
CHAPTER 2. LITERATURE REVIEW

Among all heuristic methods, Evolutionary Algorithms (EA), and more specifically GAs, are suitable to handle multi-objective optimization problems with constraints, especially for highly constrained problems in space applications [43]. A key feature of Multi-Objective EAs (MOEA) is maintaining a diverse set of solutions, since they work with a population of solutions. The Multi-Objective GA (MOGA), Non-dominated Sorting GA (NSGA), and Niched-Pareto GA (NPGA) are the pioneering MOEAs that are able to find multiple Pareto-optimal solutions in one single run [37]. Although these algorithms are proved effective in generating multiple non-dominated solutions, they still lack elitism that is an index for better convergence of an MOEA. Other algorithms, such as Strength Pareto EA (SPEA), Pareto Archived Evolution Strategy (PAES), and Pareto Envelope-based Selection Algorithm (PESA) have been introduced as elitist MOEAs [37]. Deb et al. suggest a non-dominated sorting-based multi-objective EA, called NSGA-II, which has attracted attention in the research community [42]. They have developed NSGA-II to tackle the drawbacks of the NSGA, which include high computational complexity, lack of elitism, and the need for a sharing parameter to preserve diversity. This algorithm also alleviates issues with the SPEA, PAES, and PESA by converging to a Utopian Pareto frontier while maintaining diversity of solutions. This algorithm also alleviates issues with the SPEA, PAES, and PESA by converging to a Utopian Pareto frontier while maintaining diversity of solutions. Deb et al. also developed the NSGA-III, specialized for many-objective problems, which substitutes the crowding distance sorting algorithm in the NSGA-II with a reference-point-based algorithm. To further enhance its performance, NSGA-III is combined with a specific multiple-shooting discretization and tested in a highly constrained multi-objective trajectory optimization task [43]. Although both the NSGA-II & III have been proved effective, the NSGA-II is more efficient when handling 2 objective functions [44].

2.2 $J_2$-Perturbed Orbits

Satellites in different orbital altitudes experience various disturbances due to other celestial objects, Earth’s oblateness, Earth’s atmosphere, solar radiation, etc. Among these disturbances, the most dominant one from the Low Earth Orbit (LEO) regime to the Geostationary Earth Orbit (GEO) regime, which are the regions with highly populated operative satellites, is the second zonal term in the spherical harmonic description of the Earth gravity [45]. The second zonal term’s effects, called $J_2$ perturbation, on satellite’s motion is at least 1000 times larger than those of other
terms, i.e., $J_n$ is $O(J_n^n)$ for $n \geq 3$ [46]. The osculating orbital elements refer to the orbital elements of a satellite that are perturbed through the accelerations induced by the $J_2$ perturbation, and they represent the true position and velocity vectors of a satellite. These elements poorly behave over time as a basis for prediction as their numerical integration is extremely slow. Also, despite being fast, their analytical integration contains large errors due to inherent approximations in the theories [47]. The effects of the $J_2$ perturbation in the orbital elements is captured in three types of induced motion, namely secular drift, short periodic effect, and long periodic effect. If the periodic effects are removed, then the new orbital elements are called mean orbital elements that only capture the secular drift. The mean orbital elements do not represent the true position and velocity vectors of a satellite but are well-behaved over time due to their averaging nature [48]. As depicted in Figure 2.3, the variations in osculating elements induced by periodic perturbations are much smaller than those induced by secular perturbations as the elapsed time increases. Note that the period of long periodic effect is in the order of the duration of the apsidal rotation, and over a short time this effect looks like a secular growth in the order of $J_2^2$. Also, the short periodic effect has a period in the order of the time for one satellite passage around the Earth.

In a study on $J_2$-perturbed orbits, Danielson et al. develop a semi-analytic approach for satellite orbit prediction in order to remove the periodic motion and find the mean element rates represented in equinoctial orbital elements [24]. Schaub and Alfriend [46] use the $J_2$-perturbed Hamiltonian to derive the state space equations for relative motions in spacecraft formation flying. They represent these equations using relative Delaunay elements and explore the impact of $J_2$ perturbation on relative orbits. Following that, Gim et al. derive the state transition matrices for mean and osculating elements in both classical orbital elements and equinoctial orbital elements [49, 50], using the 1st order long and short periodic variations due to $J_2$ obtained in [51]. Interested readers are referred to [52] to find more discussions on recent developments about the $J_2$-perturbed orbits.

### 2.3 Model Predictive Control of Satellites

The Guidance and Control (G&C) design for spacecraft has been receiving great attention recently. A survey paper on advanced G&C algorithms in space applications
CHAPTER 2. LITERATURE REVIEW

Figure 2.3: Osculating orbital elements

claims that designing optimization-based controllers established upon, e.g., dynamic programming, model predictive control, optimal control theory, etc., is becoming popular due to the high efficiency demand in space vehicles [13]. The developing concentration on this type of controllers is because of their ability to optimize a certain performance index during the control process. Also, their capability of being combined with other control schemes, e.g., adaptive methods [53] and disturbance observers [54], demonstrates their flexibility and functionality.

Among the optimization-based control strategies, Model Predictive Control (MPC) methods are of more interest due to their capability to handle mission related constraints [13]. The idea of MPC families is to optimize objective functions in finite time by calculating the optimal control signals using a constrained model. The MPC is equally applicable to linear and nonlinear models, respectively referred to as Linear MPC (LMPC), and Nonlinear MPC (NMPC). Despite being fast, the LMPC is not sufficient for trajectory control in the long-range rendezvous missions, since the traveled orbital distance in these missions is large and the linear models fail to accurately capture satellite dynamics [55].

In [56], the developed methodologies for solving the NMPC prior to the year 2017 are discussed. These methods that are mainly based on Non-Linear Programming (NLP) increase the complexity and the probability of converging to a local optimum. The drawbacks of these gradient-based methods can be tackled by implementing appropriate optimization algorithms. The efficacy of combining MPC (both linear and nonlinear) with evolutionary algorithms has been studied in different research areas [57, 58, 59, 60]. In [57], an NMPC formulation for trajectory generation and
tracking problems is proposed with a focus on real-time capability and robustness of autonomous vehicles and is solved using a genetic algorithm strategy. A hardware implementation of LMPC algorithms as well as its hardware in the loop testing for turbofan engine control is designed in [58] and the solutions are obtained using a genetic algorithm optimization procedure. Tian et al. [59] study the problem of cooperative search using a team of unmanned aerial vehicles and present an approach which combines MPC theory with GA to solve this problem. Finally, the efficiency of solving MPC problems using genetic algorithm in space applications is proved in [61], where the LMPC is used to control the attitude maneuver of a satellite based on a genetic algorithm.

One of the popular guidance approaches for space vehicles is reference trajectory tracking. In this approach, a predetermined transfer trajectory, which can be defined offline using trajectory optimization techniques, is prescribed to be followed by seeking control commands. There are several recent studies which have shown the effectiveness of the MPC in developing a reference tracking guidance method in real-time [55, 62]. In [55], an LMPC based constrained trajectory optimization algorithm has been applied for rendezvous and docking of two small satellites to achieve precise terminal soft docking with path constraints. Moreover, Chai et al. [62] propose both LMPC and NMPC schemes to solve the reconnaissance trajectory tracking problem. The performance of MPC in real-time applications is highly dependent on the accuracy of the prediction model. Due to difficulties in providing an accurate model of a system, there is a need to guarantee the robustness under high levels of modeling uncertainty. One of the applicable approaches that ensures a Robust MPC (RMPC) is to add a bounded uncertainty function to the dynamic model [56, 63]. A challenge in NMPC is that their closed-loop stability is not guaranteed. However, as is proved in [64], the stability of NMPC is guaranteed in the case of existing a zero state final constraint.
Chapter 3

Preliminaries

3.1 Summary

In this chapter, we review the basic definitions that are used throughout this thesis. In this regard, we first discuss the orbital motion of a satellite when exposed to the $J_2$ perturbation effects. In Section 3.2, we discuss the singularities in the perturbed satellites’ dynamics represented in the classical orbital elements. Following that in Section 3.3, the equinoctial orbital elements are introduced, using which the dynamical equations of the perturbed satellites are never singular. The relationship between these elements, the classical orbital elements, and the Cartesian elements is discussed in Section 3.3, whose content is mainly based on [24]. In Section 3.4, the $J_2$-perturbed motion of satellites is approximated by the mean orbital elements and the first order of periodic effects. This section is based on a work by Gim et al. [49]. Finally, we present the definitions that are needed in the development of the multi-objective optimization algorithm proposed in this thesis [37].

3.2 Classical Orbital Elements

There are different element sets for representing the orbital motion, e.g., the classical orbital elements (also known as Keplerian elements), equinoctial elements, Cartesian state vectors, Hill variables, cylindrical coordinates, and Deprit’s ideal elements [65]. In chapter 7, we interchangeably use classical orbital elements, equinoctial elements, and Cartesian position and velocity vectors with respect to Earth-Centered Inertial (ECI) reference frame. The ECI and Local-Vertical-Local-Horizontal (LVLH) reference frames are depicted in Figure 3.2. Herein, we first represent the classical orbital
elements and then discuss the singularities of the perturbed satellite dynamics in this orbital elements set.

The classical orbital elements are given by \((a, e, \iota, \Omega, \omega, \nu)\). Here, \(a\) is the semi-major axis, \(e\) is the eccentricity, \(\omega\) is the argument of perigee, \(\Omega\) is the Right Ascension of Ascending Node (RAAN), \(\iota\) is the inclination angle, and \(\nu\) is the true anomaly (see Figure 3.2). These orbital elements can be derived directly from the position and velocity vectors. To compute the effect of a perturbing acceleration on the osculating classical orbital elements one can apply Gauss’ form of Lagrange’s planetary equations [6]:

\[
\begin{align*}
\frac{da}{dt} &= \frac{2a^2}{\sqrt{\mu p}} \left( e \sin \nu f_r + \frac{p}{r} f_t \right), \\
\frac{de}{dt} &= \frac{1}{\sqrt{\mu p}} \left( p \sin \nu f_r + (p + r) \cos \nu + er \right) f_t, \\
\frac{d\iota}{dt} &= \frac{r \cos(\omega + \nu)}{\sqrt{\mu p}} f_n, \\
\frac{d\Omega}{dt} &= \frac{r \sin(\omega + \nu)}{\sqrt{\mu p} \sin \iota} f_n, \\
\frac{d\omega}{dt} &= \frac{1}{e \sqrt{\mu p}} \left( -p \cos \nu f_r + (p + r) \sin \nu f_t \right) - \frac{r \cos \iota \sin(\omega + \nu)}{\sqrt{\mu p} \sin \iota} f_n, \\
\frac{d\nu}{dt} &= \frac{\sqrt{\mu p}}{r^2} + \frac{1}{e \sqrt{\mu p}} \left( p \cos \nu f_r - (p + r) \sin \nu f_t \right),
\end{align*}
\]

where \(p = a(1 - e^2)\) is the semi-latus rectum, and \(f_r, f_t,\) and \(f_n\) are the components of
the perturbing acceleration in the radial, tangent, and normal directions, respectively. Although being intuitive, this method of defining an orbital state has a number of singularities that tend to complicate the equations of motion. For instance, at zero inclination the RAAN loses meaning. Similarly, for zero eccentricity the argument of perigee becomes in-distinguishable from the true anomaly. These singularities can be clearly seen in their equations of motion. Due to the existence of these singularities, the classical orbital elements are not necessarily the best set of states for numerical analysis [6].

3.3 Equinoctial Orbital Elements

The Equinoctial Element (EE) set is denoted by the vector \( \mathbf{x} = [a \ h \ k \ p \ q \ \lambda]^T \). These are osculating orbital elements and are shown with hat in the next section. For each equinoctial element set there are three associated vectors (\( \mathbf{f}, \mathbf{g}, \mathbf{w} \)) which define the equinoctial reference frame. These vectors form a right-handed orthonormal triad with the following properties:

1. \( \mathbf{f} \) and \( \mathbf{g} \) lie in the satellite orbit plane.

2. \( \mathbf{w} \) is parallel to the angular momentum vector of the satellite.
3. The angle between \( f \) and the ascending node is equal to the longitude of the ascending node.

The semi-major axis \( a \) is the same as the Keplerian semi-major axis. The eccentricity vector in the perifocal reference frame has a magnitude equal to the eccentricity and it points from the central body to perigee. Elements \( h \) and \( k \) are the \( g \) and \( f \) components, respectively, of the eccentricity vector in the equinoctial reference frame. The ascending node vector has a magnitude which depends on the inclination and it points from the central body to the ascending node. Elements \( p \) and \( q \) are the \( g \) and \( f \) components, respectively, of the ascending node vector in the equinoctial reference frame.

### 3.3.1 Conversion from Classical Elements to Equinoctial Elements

The EE is constructed based on the classical orbital elements as follows [66]

\[
\begin{align*}
a &= a, & h &= e \sin(\omega + \Omega), & k &= e \cos(\omega + \Omega), \\
p &= \tan(\iota/2) \sin(\Omega), & q &= \tan(\iota/2) \cos(\Omega), & \lambda = \Omega + \omega + M.
\end{align*}
\]
There are two auxiliary longitudes associated with the equinoctial element set: the eccentric longitude \( F \) and the true longitude \( L \). They are related to the Keplerian eccentric anomaly \( E \) and the true anomaly \( \nu \) by the equations:

\[
F = E + \omega + \Omega, \quad L = \nu + \omega + \Omega.
\]  
(3.8)

These auxiliary longitudes are used in converting from equinoctial elements to position and velocity. In addition, certain perturbations are modeled with Fourier series expansions in \( F \) or \( L \).

### 3.3.2 Conversion from Equinoctial Elements to Classical Elements

Conversely, the classical orbital elements can be obtained from the EE by [24]:

\[
a = a, \quad e = \sqrt{h^2 + k^2}, \quad i = 2 \arctan \sqrt{p^2 + q^2},
\]

\[
\sin \Omega = \frac{p}{\sqrt{p^2 + q^2}}, \quad \cos \Omega = \frac{q}{\sqrt{p^2 + q^2}}, \quad \omega = \zeta - \Omega, \quad M = \lambda - \zeta,
\]

(3.9)

where \( \zeta \) is defined by

\[
\sin \zeta = \frac{h}{\sqrt{h^2 + k^2}}, \quad \cos \zeta = \frac{k}{\sqrt{h^2 + k^2}}.
\]

(3.10)

The Keplerian eccentric and true anomalies are given by:

\[
E = F - \zeta, \quad \nu = L - \zeta.
\]

(3.11)

### 3.3.3 Conversion from Classical Elements to Cartesian Coordinates

The classical orbital elements can be converted to the Cartesian coordinates whenever required [16]:

\[
r = QO, \quad \dot{r} = Q\dot{O},
\]

\[
O = \frac{h^2}{\mu} \frac{1}{1 + e \cos \nu} \begin{bmatrix} \cos \nu & \sin \nu & 0 \\ -\sin \nu & e + \cos \nu & 0 \end{bmatrix}, \quad \dot{O} = \frac{\mu}{h} \begin{bmatrix} -\sin \nu & e + \cos \nu & 0 \end{bmatrix},
\]

(3.12)
\[ Q = \begin{pmatrix} 
\cos \omega \cos \Omega - \sin \omega \cos \iota \sin \Omega & - \sin \omega \cos \Omega - \cos \omega \cos \iota \sin \Omega & \sin \iota \sin \Omega \\
\cos \omega \sin \Omega + \sin \omega \cos \iota \cos \Omega & - \sin \omega \sin \Omega + \cos \omega \cos \iota \cos \Omega & - \sin \iota \cos \Omega \\
\sin \omega \sin \iota & \cos \omega \sin \iota & \cos \iota 
\end{pmatrix}, \]

where \( \nu \) is the true anomaly and is a function of the mean anomaly \( M \), \( \mu \) is the gravitational constant of the Earth (\( \approx 398601.2 \text{ km}^3\text{s}^{-2} \)), \( h \) is the orbital specific angular momentum \( h = \sqrt{\mu a(1-e^2)} \), and \( Q \) is the transformation matrix from the perifocal frame (\( z \)-axis perpendicular to orbital plane, \( x \)-axis pointing to periapsis of the orbit) to the ECI. We denote the set of position and velocity vectors in the Cartesian coordinates by the vector \( \mathbf{X} = [r^T \quad \mathbf{i}^T]^T \).

### 3.3.4 Conversion from Equinoctial Elements to Cartesian Coordinates

The first step in converting from equinoctial elements to position and velocity is to determine the equinoctial reference frame basis vectors \( \mathbf{f}, \mathbf{g}, \mathbf{w} \). Their components in \((3_x, 3_y, 3_z)\) system (see Figure 3.2) are

\[
\mathbf{f} = \frac{1}{1 + p^2 + q^2} \begin{bmatrix} 1 - p^2 + q^2 \\ 2pq \\ -2p \end{bmatrix},
\]

\[
\mathbf{g} = \frac{1}{1 + p^2 + q^2} \begin{bmatrix} 2pq \\ 1 + p^2 - q^2 \\ -2q \end{bmatrix},
\]

\[
\mathbf{w} = \frac{1}{1 + p^2 + q^2} \begin{bmatrix} 2p \\ 2q \\ 1 - p^2 - q^2 \end{bmatrix}. \tag{3.13}
\]

The second step is to find the eccentric and true longitudes \( F \) and \( L \), respectively.
To find the eccentric longitude $F$, the following equation should be solved:

$$\lambda = F + h \cos F - k \sin F.$$  \hfill (3.14)

This equation can be solved using a gradient-based method, e.g., Newton method. We define the following auxiliary quantities

$$n = \sqrt{\frac{\mu}{a^3}}, \quad b = \frac{1}{1 + \sqrt{1 - h^2 - k^2}},$$  \hfill (3.15)

where $n$ is called the mean motion. The true longitude $L$ is then given by:

$$\sin L = \frac{(1 - k^2 b) \sin F + h k b \cos F - h}{1 - h \sin F - k \cos F},$$
$$\cos L = \frac{(1 - h^2 b) \cos F + h k b \sin F - k}{1 - h \sin F - k \cos F}.$$  \hfill (3.16)

The next step is to compute the position $(X, Y)$ and velocity $(\dot{X}, \dot{Y})$ in the orbital plane of the satellite in the equinoctial reference frame. The radial distance of the satellite is given by:

$$r = \frac{a(1 - h^2 - k^2)}{1 + h \sin L + k \cos L}.$$  \hfill (3.17)

The position and velocity components are then given by:

$$X = r \cos L, \quad Y = r \sin L,$$
$$\dot{X} = -\frac{na(h + \sin L)}{\sqrt{1 - h^2 - k^2}}, \quad \dot{Y} = \frac{na(k + \cos L)}{\sqrt{1 - h^2 - k^2}},$$  \hfill (3.18)

where dots denote differentiation with respect to time. Now, we can compute the position and velocity vectors by

$$\mathbf{r} = X \mathbf{f} + Y \mathbf{g}, \quad \dot{\mathbf{r}} = \dot{X} \mathbf{f} + \dot{Y} \mathbf{g}.$$  \hfill (3.19)
3.4 \( J_2 \)-Perturbed Motion of Satellites

The Cartesian equation of motion for an artificial satellite in an inertial coordinate system is [47]:

\[
\dot{\mathbf{r}} = -\frac{\mu \mathbf{r}}{|\mathbf{r}|^3} + \mathbf{q} + \nabla \mathcal{R}.
\]  
(3.20)

Here, \( \dot{\mathbf{r}} = \frac{d^2 \mathbf{r}}{dt^2} \) is the acceleration vector and \( \mathbf{q} \) is the acceleration due to non-conservative perturbing forces, e.g., atmospheric drag and solar radiation pressure. Further, \( \mathcal{R} \) is a potential-like function called the disturbing function from which one can derive the acceleration due to conservative perturbing forces, e.g., central body spherical harmonics and third-body point-mass. If \( m \) and \( \Pi \) are the mass and potential energy of a satellite, respectively, then the disturbing function \( \mathcal{R} \) is

\[
\mathcal{R} = -\frac{\Pi}{m}.
\]  
(3.21)

In order to apply the generalized method of averaging, it is necessary to convert the equations of motion into a form giving the rates of change of the satellite orbital elements as a function of the orbital elements themselves. The equations of motion resulting from this conversion are called the Variation-of-Parameters (VOP) equations of motion. The generalized method of averaging can be applied to a wide variety of orbital element sets. However, the equinoctial elements were chosen since the variational equations for the equinoctial elements are nonsingular for all orbits. The derivation of these equations is discussed in some details in [67]. The VOP equations of motion are

\[
\dot{x}_i = n \delta_{i6} + \frac{\partial x_i}{\partial \mathbf{r}} - \sum_{j=1}^{6} (x_i, x_j) \frac{\partial \mathcal{R}}{\partial x_j},
\]  
(3.22)

where \( x_i \) is the \( i^{th} \) element of the vector \( \mathbf{x} \) \((i = 1, \cdots, 6)\), \( n = \sqrt{\frac{\mu}{a^3}} \) is the Kepler mean motion, \( \delta_{i6} \) is the Kronecker delta which is equal to one only when \( i = 6 \) and is zero otherwise. The partial derivatives of the equinoctial elements with respect to
velocity $\frac{\partial \dot{r}}{\partial \dot{r}}$ are given by:

\[ \begin{aligned}
\frac{\partial a}{\partial \dot{r}} &= \frac{2\dot{r}}{n^2 a} \\
\frac{\partial h}{\partial \dot{r}} &= \left( 2\dot{X}Y - X\dot{Y} \right) f - X\dot{X}g + \frac{k(qY - pX)w}{\eta_1 \eta_2} \\
\frac{\partial k}{\partial \dot{r}} &= \left( 2\dot{X}Y - Y\dot{X} \right) g - Y\dot{Y}f - \frac{h(qY - pX)w}{\eta_1 \eta_2} \\
\frac{\partial p}{\partial \dot{r}} &= \frac{\eta_3 Yw}{2\eta_1 \eta_2} \\
\frac{\partial q}{\partial \dot{r}} &= \frac{\eta_3 Xw}{2\eta_1 \eta_2} \\
\frac{\partial \lambda}{\partial \dot{r}} &= -\frac{2\dot{r}}{\eta_1} + \frac{(qY - pX)w}{\eta_1} + \frac{k\frac{\partial h}{\partial \dot{r}} - h\frac{\partial k}{\partial \dot{r}}}{1 + \eta_2},
\end{aligned} \]

(3.23)

where,

\[ \eta_1 = \sqrt{\mu a}, \quad \eta_2 = \sqrt{1 - h^2 - k^2}, \quad \eta_3 = 1 + p^2 + q^2. \]

Also, $(x_i, x_j)$ is called the Poisson matrix and its entries are called the Poisson brackets. The Poisson brackets of the element set $\mathbf{x}$ are defined by

\[ (x_i, x_j) \triangleq \frac{\partial x_i}{\partial r} \cdot \frac{\partial x_j}{\partial \dot{r}} - \frac{\partial x_i}{\partial \dot{r}} \cdot \frac{\partial x_j}{\partial r}. \]

(3.24)

It is evident that $(x_i, x_i) = 0$, $(x_i, x_j) = -(x_j, x_i)$. Having the fifteen independent Poisson brackets for the EE, the Poisson matrix $\mathcal{P}$ is [66]:

\[ \mathcal{P} = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & \frac{2\eta_1}{\mu} \\
0 & 0 & \frac{\eta_2}{\eta_1} & k\eta_3 & \frac{k\eta_3}{2\eta_1 \eta_2} & -\frac{k\eta_2}{\eta_1 (1 + \eta_2)} \\
0 & -\frac{\eta_2}{\eta_1} & 0 & \frac{k\eta_3}{2\eta_1 \eta_2} & -\frac{k\eta_3}{2\eta_1 \eta_2} & -\frac{k\eta_2}{\eta_1 (1 + \eta_2)} \\
0 & -\frac{k\eta_3}{2\eta_1 \eta_2} & \frac{k\eta_3}{2\eta_1 \eta_2} & 0 & \frac{\eta_1^2}{4\eta_1 \eta_2} & -\frac{\eta_1 \eta_2}{2\eta_1 \eta_2} \\
0 & -\frac{k\eta_3}{2\eta_1 \eta_2} & \frac{k\eta_3}{2\eta_1 \eta_2} & -\frac{\eta_1^2}{4\eta_1 \eta_2} & 0 & \frac{\eta_1 \eta_2}{2\eta_1 \eta_2} \\
-\frac{2\eta_1}{\mu} & \frac{h\eta_2}{\eta_1 (1 + \eta_2)} & \frac{k\eta_2}{\eta_1 \eta_2 (1 + \eta_2)} & \frac{p\eta_3}{2\eta_1 \eta_2} & \frac{q\eta_3}{2\eta_1 \eta_2} & 0
\end{pmatrix}. \]

(3.25)

The first element of (3.22) is the two-body effect, the second part indicates the Gaussian (or non-conservative) part, and the last element is for the Lagrangian (or conservative) effects. In this thesis, we only consider the two-body effect and the Lagrangian
part. The Lagrangian part of the VOP equations of motion contains the partial derivatives of the disturbing function \( R \) with respect to \( p \) and \( q \). The perturbations which contribute to \( R \) are not conveniently described in terms of \( p \) and \( q \), however. For these functions, it is better to write \( R \) as a function of \((a, h, k, \lambda)\) and the direction cosines \((\alpha, \beta, \gamma)\) of the symmetry axis of the perturbation. For central-body gravitational spherical harmonics, let \( z_B \) be the unit vector from the center of mass to the geographic north pole of the central-body. The direction cosines of \( z_B \) with respect to the equinoctial reference frame \((f, g, w)\) are then given by

\[
\alpha = z_B \cdot f, \quad \beta = z_B \cdot g, \quad \gamma = z_B \cdot w.
\] (3.26)

The quantities of \((\alpha, \beta, \gamma)\) are not independent but related by

\[
\alpha^2 + \beta^2 + \gamma^2 = 1.
\] (3.27)

Note that \((\alpha, \beta, \gamma)\) are functions of \( p \) and \( q \), since the unit vectors \((f, g, w)\) are functions of \( p \) and \( q \) through equations (3.13). Note also that \((\alpha, \beta, \gamma)\) are functions of \( t \), since \( z_B \) is a varying function of time. If the vector \( z_B \) along the geographic axis of the central-body is parallel to the \( Z_B \)-axis of Figure 3.3 at epoch, then the direction cosines of \( z_B \) at epoch are

\[
\alpha = -\frac{2p}{1 + p^2 + q^2}, \quad \beta = \frac{2q}{1 + p^2 + q^2}, \quad \gamma = \frac{1 - p^2 - q^2}{1 + p^2 + q^2}.
\]

The partial derivatives of \((\alpha, \beta, \gamma)\) with respect to \( p \) and \( q \) are:

\[
\frac{\partial \alpha}{\partial p} = -\frac{2(q\beta + \gamma)}{\eta_3},
\]
\[
\frac{\partial \alpha}{\partial q} = \frac{2p\beta}{\eta_3},
\]
\[
\frac{\partial \beta}{\partial p} = \frac{2q\alpha}{\eta_3},
\]
\[
\frac{\partial \beta}{\partial q} = -\frac{2(p\alpha - \gamma)}{\eta_3},
\]
\[
\frac{\partial \gamma}{\partial p} = \frac{2\alpha}{\eta_3},
\]
\[
\frac{\partial \gamma}{\partial q} = -\frac{2\beta}{\eta_3}.
\] (3.28)
CHAPTER 3. PRELIMINARIES

The partial derivatives of $\mathcal{R}$ with respect to $p$ and $q$ turns out to be

\[
\frac{\partial \mathcal{R}}{\partial p} = \frac{2}{\eta_3} (\mathcal{R}_{\alpha\gamma} + q\mathcal{R}_{\alpha\beta}),
\]

\[
\frac{\partial \mathcal{R}}{\partial q} = -\frac{2}{\eta_3} (\mathcal{R}_{\beta\gamma} + p\mathcal{R}_{\alpha\beta}),
\]

(3.29)

where the cross-derivative operator is defined as

\[
\mathcal{R}_{xy} = x \frac{\partial \mathcal{R}}{\partial y} - y \frac{\partial \mathcal{R}}{\partial x}.
\]

(3.30)

With this notation, the Lagrangian part of the VOP equations of motion in (3.22) become:

\[
\dot{a} = \frac{2a}{\eta_1} \frac{\partial \mathcal{R}}{\partial \lambda},
\]

\[
\dot{h} = \frac{\eta_2}{\eta_1} \frac{\partial \mathcal{R}}{\partial k} + \frac{k}{\eta_1 \eta_2} (p\mathcal{R}_{\alpha\gamma} - q\mathcal{R}_{\beta\gamma}) - \frac{h \eta_2}{\eta_1 (1 + \eta_2)} \frac{\partial \mathcal{R}}{\partial \lambda},
\]

\[
\dot{k} = -\frac{\eta_2}{\eta_1} \frac{\partial \mathcal{R}}{\partial h} - \frac{h}{\eta_1 \eta_2} (p\mathcal{R}_{\alpha\gamma} - q\mathcal{R}_{\beta\gamma}) - \frac{k \eta_2}{\eta_1 (1 + \eta_2)} \frac{\partial \mathcal{R}}{\partial \lambda},
\]

\[
\dot{p} = \frac{\eta_3}{2 \eta_1 \eta_2} \left[ p (\mathcal{R}_{hh} - \mathcal{R}_{\alpha\beta} - \frac{\partial \mathcal{R}}{\partial \lambda} - \mathcal{R}_{\beta\gamma}) \right],
\]

\[
\dot{q} = \frac{\eta_3}{2 \eta_1 \eta_2} \left[ q (\mathcal{R}_{hh} - \mathcal{R}_{\alpha\beta} - \frac{\partial \mathcal{R}}{\partial \lambda} - \mathcal{R}_{\alpha\gamma}) \right],
\]

\[
\dot{\lambda} = -\frac{2a}{\eta_1} \frac{\partial \mathcal{R}}{\partial a} + \frac{\eta_2}{\eta_1 (1 + \eta_2)} \left( h \frac{\partial \mathcal{R}}{\partial h} + k \frac{\partial \mathcal{R}}{\partial k} \right) + \frac{1}{\eta_1 \eta_2} \left( p\mathcal{R}_{\alpha\gamma} - q\mathcal{R}_{\beta\gamma} \right).
\]

(3.31)

3.4.1 Central-Body Gravitational Potential

The well-known expression for the disturbing function due to the gravitational field of the central body is [47]:

\[
\mathcal{R}(r, \phi, \psi) = -\frac{\mu}{r} \sum_{n=2}^{N} J_n \left( \frac{R_e}{r} \right)^n P_n \sin \phi
\]

\[
\frac{\mu}{r} \sum_{2}^{N} \sum_{m=0}^{\min(n,M)} \left( \frac{R_e}{r} \right)^n P_{nm} (\sin \phi) (C_{nm} \cos m\psi + S_{nm} \sin m\psi).
\]

(3.32)
Here,

\[ r = \text{radial distance from center of mass of central body} \]
\[ \phi = \text{geocentric latitude} \]
\[ \psi = \text{geocentric longitude} \]
\[ \mu = \text{central-body gravitational constant} \]
\[ R_e = \text{central-body mean equatorial radius} \]
\[ P_{nm} = \text{associated Legendre function of order } m \text{ and degree } n \]
\[ C_{nm}, S_{nm} = \text{geopotential constant coefficients} \]
\[ M = \text{maximum order of geopotential field} \quad (M \leq N) \]
\[ N = \text{maximum degree of geopotential field} \]

where \( J_n, C_{nm}, \) and \( S_{nm} \) are obtained experimentally from satellite observations and \( P_n \) is the associated Legendre function. For rotationally symmetric bodies, \( C_{nm} = S_{nm} = 0. \) Therefore, (3.32) can be simplified to

\[
\mathcal{R} = -\frac{\mu}{r^3} \sum_{n=2}^{N} J_n \left( \frac{R_e}{r} \right)^n P_n \sin \phi. \tag{3.34}
\]

The most dominant perturbing effect for Earth is due to the term \( J_2. \) The perturbing potential due to \( J_2 \) is given by

\[
\mathcal{R} = -\frac{\mu}{r^3} J_2 \left( \frac{3}{2} \sin \phi - \frac{1}{2} \right). \tag{3.35}
\]

The geocentric altitude can be expressed in terms of the equinoctial elements as \( \cos \phi = \alpha \cos L + \beta \sin L. \) The coefficient \( J_2 \) is called the second zonal harmonics coefficient and the numerical value of that is \( 1.08262545 \times 10^{-3}. \)

### 3.4.2 Averaged Equations of Motion

The generalized method of averaging is used to divide the VOP equations of motion (3.22) into a periodic part which can be integrated analytically and a slowly-varying part which can be integrated numerically using time steps several orders of magnitude longer than the time steps appropriate for integrating the osculating equations of motion. To apply the generalized method of averaging, we first assume that the
osculating orbital elements \( \hat{x}_i \) (that is shown without hat in the previous section) are related to a set of mean elements \( x_i \) by a near identity transformation:

\[
\hat{x}_i = x_i + \sum_{j=1}^{\infty} e^j \eta_j^i (a, h, k, p, q, \lambda, t)
\]  

\[(3.36)\]

Let \( \hat{f}(\hat{a}, \hat{h}, \hat{k}, \hat{p}, \hat{q}, \hat{\lambda}) \) be a function of osculating orbital elements using EE. Averaging this function over a \( 2\pi \) period of the satellite mean longitude \( \lambda \), where the other satellite orbital elements are held fixed, eliminates the long- and short-periodic effects and reveals secular effects \( f \) [24]:

\[
< \hat{f} > = f = \int_{-\pi}^{\pi} \hat{f}(\hat{a}, \hat{h}, \hat{k}, \hat{p}, \hat{q}, \hat{\lambda}, t) d\lambda.
\]  

\[(3.37)\]

For the central-body gravitational zonal harmonics, the appropriate averaging operator \( < \cdot > \) is (3.37) and the appropriate disturbing function \( R \) is (3.35). The contribution of the central-body gravitational second zonal harmonics (3.35) to the averaged equations of motion is

\[
U = \frac{J (\gamma^2 - \frac{1}{3})}{a^3 (1 - h^2 - k^2)^{\frac{3}{2}}}, \quad J = \frac{3 \mu R_e^2 J_2}{4}.
\]  

\[(3.38)\]

Then, the contribution of (3.38) to the mean equation of motion is

\[
\begin{align*}
\frac{da}{dt} &= 0, \\
\frac{dh}{dt} &= \frac{Jk [3\gamma^2 - 1 + 2\gamma(p\alpha - q\beta)]}{\eta_1 \eta_2^2 a^3}, \\
\frac{dk}{dt} &= -\frac{Jh [3\gamma^2 - 1 + 2\gamma(p\alpha - q\beta)]}{\eta_1 \eta_2^2 a^3}, \\
\frac{dp}{dt} &= -\frac{\eta_3 J\beta \gamma}{\eta_1 \eta_2^2 a^3}, \\
\frac{dq}{dt} &= -\frac{\eta_3 J\alpha \gamma}{\eta_1 \eta_2^2 a^3}, \\
\frac{d\lambda}{dt} &= \frac{J [(1 + \eta_2) (3\gamma^2 - 1) + 2\gamma(p\alpha - q\beta)]}{\eta_1 \eta_2^2 a^3}.
\end{align*}
\]  

\[(3.39)\]
Finally, the mean equation of motion will be a summation of (3.39) with the mean Keplerian motion $n$.

$$\dot{x} = \begin{pmatrix} 0 \\ \frac{Jk[3\gamma^2-1+2\gamma(p\alpha-q\beta)]}{\eta_1^2a^3} \\ -\frac{Jh[3\gamma^2-1+2\gamma(p\alpha-q\beta)]}{\eta_1^2a^3} \\ -\frac{\eta_2J\beta\gamma}{\eta_1^2a^3} \\ -\frac{\eta_2J\alpha\gamma}{\eta_1^2a^3} \\ -\frac{\eta_3J\alpha\gamma}{\eta_1^2a^3} \\ \frac{J[(1+\eta_2)(3\gamma^2-1)+2\gamma(p\alpha-q\beta)]}{\eta_1^2a^3} + n \end{pmatrix}. $$

### 3.4.3 Approximation of Osculating Elements

To have a more precise dynamics for the motion of a satellite, the first order of long and short periodic effects can be added to the mean motion. The Equation (3.36) can be rewritten as

$$\dot{x} \approx x + \Delta x_p, \quad (3.40)$$

where $\Delta x_p$ denotes the 1st order long and short periodic variations due to the primary gravitational perturbation $J_2$ and are obtained from the generating function in [51] via

$$W_1 = W_1^{lp} + W_1^{sp}, \quad (3.41)$$

where $W_1^{lp}$ and $W_1^{sp}$ are functions of the normalized Delaunay variables $(l, g, h, L, G, H)^T$ and the true anomaly $\nu$:

$$W_1^{lp} = -\left(\frac{1}{32G^3}\right)(1 - \frac{G^2}{L^2})(1 - 5\frac{H^2}{G^2})^{-1}(1 - 16\frac{H^2}{G^2} + 15\frac{H^4}{G^4})\sin 2g,$$

$$W_1^{sp} = \left(\frac{1}{4G^3}\right)(-1 + 3\frac{H^2}{G^2})(\nu - l + e\sin \nu) + 
\left(\frac{3}{8G^3}\right)(1 - \frac{H^2}{G^2})[\sin(2\nu + 2g) + e\sin(\nu + 2g) + \frac{e}{3}\sin(3\nu + 2g)]. \quad (3.42)$$
The Delaunay variables are defined as

\[ L = \sqrt{\mu a}, \quad G = L(1 - e^2)^{\frac{1}{2}}, \quad H = G \cos i, \]
\[ l = \text{mean anomaly}, \quad g = \text{argument of perigee}, \quad h = \text{right ascension}. \]

Using this generating function in (3.42), a single transformation is achieved by the Poisson bracket to obtain the periodic variations, \( \Delta x_p = -J_2(x, W_1) \), where

\[
(x_i, W_1) = \left( \frac{\partial x_i}{\partial l} \right) \left( \frac{\partial W_1}{\partial L} \right) + \left( \frac{\partial x_i}{\partial g} \right) \left( \frac{\partial W_1}{\partial G} \right) + \left( \frac{\partial x_i}{\partial h} \right) \left( \frac{\partial W_1}{\partial H} \right) - \left( \frac{\partial x_i}{\partial L} \right) \left( \frac{\partial W_1}{\partial l} \right) - \left( \frac{\partial x_i}{\partial G} \right) \left( \frac{\partial W_1}{\partial g} \right) - \left( \frac{\partial x_i}{\partial H} \right) \left( \frac{\partial W_1}{\partial h} \right). \]

Therefore, \( \hat{x} \) is obtained by factoring \( R_e^2 \)

\[
\hat{x} \approx x - (J_2 R_e^2)(x^{(lp)} + x^{(sp)}),
\]

where \( x^{(lp)} \) and \( x^{(sp)} \) denote the first order long- and short-periodic variations due to \( J_2 \).

### 3.5 Multi-Objective Optimization

Problems with multiple objectives arise in a natural fashion in most disciplines and their solution has been a challenge to researchers for a long time. Despite the considerable variety of techniques developed to tackle these problems, the complexities of their solution calls for search approaches, as an alternative. These approaches can be categorized into deterministic (e.g., Greedy, Hill-Climbing) and stochastic (e.g., Random Search, Simulated Annealing, Evolutionary Algorithm, Tabu Search) algorithms. A "good" multi-objective optimization algorithm should be able to satisfy the following goals:

- Preserve nondominated points in objective space and associated solution points in decision space.
- Continue to make algorithmic progress towards the Pareto Front in objective function space.
- Maintain diversity of points on Pareto frontier (phenotype space) and/or of Pareto Optimal solutions (genotype space).
- Provide the decision maker “enough” but limited number of Pareto Optimal points for selection, resulting in decision variable values.

In order to understand these goals of a multi-objective optimizer and its attainment, we start the discussion with single-objective optimization problems. The single-objective optimization problem as presented in Definition 3.5.1 continues to be addressed by many search techniques including numerous evolutionary algorithms.

Definition 3.5.1. (General Single-Objective Optimization Problem) A general single-objective optimization problem is defined as minimizing (or maximizing) a scalar function \( f(x) \) subject to constraints \( g_i(x) \leq 0, i = 1, \ldots, m, \) and \( h_j(x) = 0, j = 1, \ldots, p \), where \( x \in \Omega \subseteq \mathbb{R}^n \). A solution minimizes (or maximizes) \( f(x) \), where \( x \) is an \( n \)-dimensional decision variable from the universe \( \Omega \).

Observe that \( g_i(x) \leq 0 \) and \( h_j(x) = 0 \) represent constraints that must be fulfilled while optimizing (minimizing or maximizing) \( f(x) \). The universe \( \Omega \) contains all possible \( x \) that can be used to satisfy an evaluation of \( f(x) \) and its constraints. Of course, \( x \) can be a vector of continuous or discrete variables as well as \( f \) being continuous or discrete.

Although single-objective optimization problems may have a unique optimal solution, Multi-Objective Problems (MOPs) (as a rule) present a possibly uncountable set of solutions. The multi-objective optimization problem can be defined as the problem of finding a vector of decision variables which satisfies constraints and optimizes a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria which are usually in conflict with each other. Hence, the term “optimize” means finding solutions which would give acceptable values for all objective functions in the eyes of the decision maker. In order to develop an understanding of MOPs, a series of definitions are required that are discussed in the following.

3.5.1 Decision Variables

The decision variables are the numerical quantities for which values are to be chosen in an optimization problem. These quantities are denoted by \( x_j, j = 1, 2, \ldots, n \). The vector \( x \) of \( n \) decision variables is represented by \( x = [x_1, x_2, \ldots, x_n]^T \in \mathbb{R}^n \).
3.5.2 Constraints

In most optimization problems there are always restrictions imposed by the particular characteristics of the environment or available resources (e.g., physical limitations, time restrictions, etc.). These restrictions must be satisfied in order to consider a certain solution acceptable. All these restrictions in general are called constraints, and they describe dependencies among decision variables involved in the problem. These constraints are expressed in the form of mathematical inequalities \( g_i(x) \leq 0, \ i = 1, \ldots, m \), or equalities \( h_j(x) = 0, \ j = 1, \ldots, p \). Note that \( p \) must be less than \( n \) because if \( p \geq n \) the problem is said to be over-constrained.

3.5.3 Objective Functions

In order to know how “good” a certain solution is, it is necessary to have some criteria to evaluate it. These criteria are expressed as computable functions of the decision variables, called objective functions. In real-world problems, some functions are in conflict with others, and some must be minimized while others are maximized. The multiple objectives being optimized almost always conflict, placing a partial, rather than total, ordering on the search space. The objective functions are designated by \( f_1(x), f_2(x), \ldots, f_k(x) \), where \( k \) is the number of objective functions in the MOP being solved. Therefore, the objective functions form a vector function \( f(x) \) which is defined by \( f(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T \in \mathbb{R}^k \).

3.5.4 Multi-Objective Optimization Problem

The MOPs are those problems where the goal is to optimize \( k \) objective functions simultaneously. This may involve the maximization or minimization of all objective functions, or a combination of maximization and minimization. An MOP global minimum (or maximum) problem is formally defined in the following.

**Definition 3.5.2.** (General MOP) : A general MOP is defined as minimizing (or maximizing) \( f(x) = [f_1(x), \ldots, f_k(x)]^T \) subject to \( g_i(x) \leq 0, \ i = 1, \ldots, m \), and \( h_j(x) = 0, \ j = 1, \ldots, p \), where \( x \in \Omega \subseteq \mathbb{R}^n \). An MOP solution minimizes (or maximizes) the components of a vector \( f(x) \), where \( x \) is an \( n \)-dimensional decision variable from the universe \( \Omega \).

It is noted that constraints must be fulfilled while minimizing (or maximizing) \( f(x) \)
and Ω contains all possible x that can be used to satisfy an evaluation of f(x). The objective functions may be linear or nonlinear and continuous or discrete in nature.

3.5.5 Pareto Optimality

In MOPs, the set of solutions is found through the use of Pareto Optimality theory. Having several objective functions, the notion of “optimum” changes, because in MOPs, the aim is to find good compromises (or “trade-offs”) rather than a single solution as in global optimization. The notion of “Pareto optimum” is generalized by Vilfredo Pareto [68]. The formal definition in a multi-objective minimization problem is provided in the following.

**Definition 3.5.3.** (Pareto Dominance) A vector \( u = [u_1, \cdots, u_k]^T \) is said to dominate another vector \( v = [v_1, \cdots, v_k]^T \) (denoted by \( u \preceq v \)) if and only if \( u \) is partially less than \( v \), i.e., \( \forall i \in \{1, \cdots, k\}, u_i \leq v_i \) and \( \exists i \in \{1, \cdots, k\} \) such that \( u_i < v_i \).

**Definition 3.5.4.** (Pareto Optimality) A solution \( x \in \Omega \) is said to be Pareto Optimal in \( \Omega \) if and only if there is no \( x' \in \Omega \) for which \( v = f(x') \) dominates \( u = f(x) \).

This definition means that \( x^* \) is Pareto Optimal if there exists no feasible vector \( x \) which would decrease some criterion without causing a simultaneous increase in at least one other criterion. The concept of Pareto Optimality is integral to the theory and solving of MOPs.

**Definition 3.5.5.** (Pareto Optimal Set) For a given MOP with the objective functions \( f(x) \), the Pareto Optimal Set, \( P^* \), is defined as

\[
P^* := \{ x \in \Omega | \nexists x' \in \Omega \text{ s.t. } f(x') \preceq f(x) \}.
\]

Pareto Optimal solutions are those solutions within the genotype search space (decision space) whose corresponding phenotype objective vector components cannot be all simultaneously improved. These solutions are also termed non-inferior, admissible, or efficient solutions, with the entire set represented by \( P^* \). Their corresponding vectors in the objective space are termed nondominated and they form the Pareto frontier set \( PF^* \).

**Definition 3.5.6.** (Pareto frontier) For a given MOP with the objective functions \( f(x) \), and Pareto Optimal Set, \( P^* \), the Pareto frontier \( PF^* \) is defined as

\[
PF^* := \{ u \in \mathbb{R}^k | u = f(x) \text{ s.t. } x \in P^* \}.
\]
When plotted in objective space, the nondominated vectors in $PF^*$ form a hyper-surface in $\mathbb{R}^k$. In general, it is not easy to find an analytical expression for this hyper-surface. The normal procedure to generate the Pareto frontier is to compute many points in $\Omega$ and their corresponding image under $f$. When there is a sufficient number of these points, it is then possible to determine the nondominated points and to produce the Pareto frontier.
Chapter 4

Impulsive Smooth Trajectories for Target Chasing

4.1 Summary

In this chapter, we derive a set of necessary and sufficient conditions to generate multi-impulse smooth transfer trajectories for a servicer chasing a target in the long-range rendezvous phase. As discussed before, including the condition of chasing a target rather than landing at a predefined point increases the chance of finding a trajectory with a shorter transfer time or better control effort. The motion of the servicer and the target is formulated based on the two-body problem, and smoothness requirement dictates that they must move in a plane. We model an impulse in an impulsive trajectory with an instantaneous velocity change. The proposed transfer trajectories are constructed by joining a number of co-planar co-focal elliptical arcs.

4.2 Multi-Impulse Smooth Trajectories

Motion of a satellite in a central gravity is fully described by the set of classical orbital elements denoted by the vector \( \mathbf{q} = [a, e, \iota, \Omega, \omega, \nu]^T \). Here, \( a \) is the semi-major axis, \( e \) is the eccentricity, \( \iota \) is the inclination, \( \Omega \) is the Right Ascension of Ascending Node (RAAN), \( \omega \) is the argument of perigee, and \( \nu \) is the true anomaly. In a given plane in the three-dimensional space, identified by known \( \iota \) and \( \Omega \), the satellite’s motion is fully described by only four orbital elements, i.e., \( a, e, \omega, \) and \( \nu \). Using the polar coordinates of the plane \((r, \theta)\), depicted in Figure 4.1, \( \nu \) can be replaced by \( \theta = \nu - \omega \) to introduce the vector \( \mathbf{p} = [a, e, \theta, \omega]^T \) as the set of planar orbital elements. The
parameters $r$, $\nu$, $\theta$, $a$, $e$, and $\omega$ are shown in Figure 4.1, where the origin is the Earth’s center.

Conversely, knowing the satellite’s position $(r, \theta)$ and its speed $(\dot{r}, \dot{\theta})$ relative to the Earth-Centered Inertial (ECI) coordinate frame, we find the planar orbital elements by [16, p. 210]:

\begin{align*}
a &= \frac{\mu r}{2\mu - r(r^2 + r^2\dot{\theta}^2)}, \quad (4.1) \\
e &= \sqrt{1 - \frac{r^4\dot{\theta}^2}{\mu a}}, \quad (4.2) \\
\omega &= \cos^{-1}\left\{\frac{1}{e} \frac{a}{r} ((1 - e^2) - 1)\right\} - \theta, \quad (4.3)
\end{align*}

where $\mu$ is the gravitational constant of the Earth ($\approx 398601.2$ km$^3$s$^{-2}$).

When an impulse takes place, the satellite’s velocity changes instantaneously resulting in a sudden variation of the orbital elements and accordingly the satellite’s trajectory. The velocity change at an intersection is denoted by $\Delta \mathbf{v}$. Given (4.1)-(4.3), we can observe how the radial ($\Delta \dot{r}$) and tangential ($r\Delta \dot{\theta}$) components of $\Delta \mathbf{v}$ vary the orbital elements of a satellite. A trajectory is called smooth, if the curves before and after an impulse intersect and share an identical tangent direction at the intersection point [69]. The smoothness condition at the intersection of any two sequential arcs
can be formulated in the polar coordinates as follows [20]:

\begin{align}
  r^- &= r^+ , \\
  \theta^- &= \theta^+ , \\
  \left( \frac{dr}{d\theta} \right)^- &= \left( \frac{dr}{d\theta} \right)^+ .
\end{align}

The + and − signs correspond to the conditions right after and right before an impulse, respectively. Using the relationship between \( r \) and the planar orbital elements [16, p. 182], the tangent direction of a trajectory is calculated by

\begin{align}
  \frac{dr}{d\theta} &= e \sin(\theta + \omega) \\
  r &= \frac{a(1 - e^2)}{1 + e \cos(\theta + \omega)} .
\end{align}

The conditions in (4.4) and (4.5) imply that the two curves must have an intersection at the impulse location, and the polar coordinates of the satellite remain unchanged over the course of an impulse. Further, (4.6) demonstrates that at the location of the impulse the direction of the velocity vector does not change, and only its magnitude must vary. This condition implies that the two curves before and after an impulse must remain in a plane.

**Problem 4.2.1 (Smoothness).** Given two locations in two co-planar orbits and an integer \( N \), find the necessary and sufficient conditions for the smoothness of an \( N \)-impulse trajectory for a satellite joining these two locations.

For a general \( N \)-impulse smooth co-planar maneuver, \( N - 1 \) arcs need to be designed to generate a transfer trajectory, each of which must satisfy (4.4)-(4.6) at the impulse locations. We denote the initial and final orbital locations by \( p_1 \) and \( p_{N+1} \), respectively, and we index the intermediate orbital elements by \( i = 2, \ldots, N \). The vectors \( p_i = [a_i, e_i, \theta_i, \omega_i]^T, i = 1, \ldots, N + 1 \), are the set of planar orbital elements describing the full satellite trajectory starting from \( p_1 \) and ending at \( p_{N+1} \). The parameter \( \theta_i \), for \( i = 1, \ldots, N \), describes the location of the impulse where the satellite leaves the \( i^{th} \) orbit, and based on (4.5) it is equal to the satellite’s polar angle immediately after the impulse. Note that \( \theta_{N+1} = \theta_N \) that indicates the point of entry to the final orbit. To guarantee the smoothness of an \( N \)-impulse transfer trajectory from a
known initial orbital location $p_1$ to a given final orbital location $p_{N+1}$, we impose the following set of $2N$ nonlinear equations based on (4.4)-(4.6), for $i = 1, \ldots, N$:

$$f_{i1}(p_i, p_{i+1}) := e_i \sin(\theta_i + \omega_i) + e_i e_{i+1} \sin(\omega_i - \omega_{i+1})$$

$$- e_{i+1} \sin(\theta_i + \omega_{i+1}) = 0,$$

$$(4.9)$$

$$f_{i2}(p_i, p_{i+1}) := a_i (1 - e_i^2) [1 + e_{i+1} \cos(\theta_i + \omega_{i+1})]$$

$$- a_{i+1} (1 - e_{i+1}^2) [1 + e_i \cos(\theta_i + \omega_i)] = 0.$$  

$$(4.10)$$

The number of unknowns in this set of equations is $4N - 5$ including the elements of vectors $p_2$ to $p_N$ minus $\theta_N$ that must be equal to the known $\theta_{N+1}$ based on (4.5).

## 4.3 Chasing a Target Satellite

**Problem 4.3.1** (Chasing). Given co-planar orbital locations of a servicer and a target at the time $t = 0$, find $N$-impulse smooth transfer trajectories for the servicer to intercept the target at its point of entry to the target’s orbit.

The time-dependent orbital elements of the servicer in the parking orbit and the target are denoted by $p_S(t) = [a_S, e_S, \theta_S(t), \omega_S]$ and $p_T(t) = [a_T, e_T, \theta_T(t), \omega_T]$, respectively. In this problem, knowing $\theta_S(0)$ and $\theta_T(0)$ we seek the necessary and sufficient conditions to generate $N$-impulse smooth transfer trajectories for the servicer, such that $p_1$ is the location of the first impulse in the parking orbit and $p_{N+1}$ is the point of entry of the servicer to the target’s orbit intercepting the target. That is, we must impose the following condition in addition to the smoothness conditions in (4.9)-(4.10):

$$p_{N+1} = p_T(t_f),$$  

$$(4.11)$$

where $t_f$ indicates the servicer’s arrival time. For an $N$-impulse trajectory, $t_f$ depends on the intermediate orbital elements, and is calculated by
CHAPTER 4. SMOOTH TRAJECTORIES FOR TARGET CHASING

\[ t_f(p_1, \ldots, p_N) = \sum_{i=1}^{N} \Delta t_i(p_i, \theta_{i-1}), \]  
(4.12)

\[ \Delta t_i(p_i, \theta_{i-1}) = \frac{M_i^- - M_i^+}{\sqrt{\frac{\mu}{a_i^3}}}, \quad i = 1, \ldots, N \]  
(4.13)

where \(M_i^+\) and \(M_i^-\) are the mean anomalies of the servicer at the \(i^{th}\) impulse in the \((i + 1)^{st}\) and \(i^{th}\) orbit, respectively. Note that \(M_0^+\) is equal to \(M_S(0)\), the servicer’s initial mean anomaly in its parking orbit. To calculate the mean anomalies, we use the following set of equations \((i = 1, \ldots, N)[16, \text{p. 160}]:

\[ M_i^+ = E_i^+ - e_i \sin E_i^+, \]  
(4.14)

\[ E_i^+ = 2 \tan^{-1}\left( \sqrt{\frac{1 - e_{i+1}}{1 + e_{i+1}}} \tan \frac{\theta_i + \omega_{i+1}}{2} \right), \]  
(4.15)

\[ M_i^- = E_i^- - e_i \sin E_i^-, \]  
(4.16)

\[ E_i^- = 2 \tan^{-1}\left( \sqrt{\frac{1 - e_i}{1 + e_i}} \tan \frac{\theta_i + \omega_i}{2} \right). \]  
(4.17)

The parameters \(E_i^+\) and \(E_i^-\) are the eccentric anomalies of the servicer at the \(i^{th}\) impulse in the \((i + 1)^{st}\) and the \(i^{th}\) orbit, respectively. At the time \(t_f\) the mean anomaly of the target satellite \(M_T(t_f)\) is calculated by [16, p. 158]:

\[ M_T(t_f) = \sqrt{\frac{\mu}{a_T^3}} t_f + M_T(0), \]  
(4.18)

where \(M_T(0)\) is the target’s initial mean anomaly in its orbit. To ensure interception of the target, the condition on the final polar angles of the servicer and target in (4.11) can be substituted by

\[ M_T(t_f) = M_N^+. \]  
(4.19)

Note that \(M_T(t_f)\) is a function of the number of impulses \(N\), intermediate orbital elements \(p_i\), and the initial location of the servicer \(\theta_S(0)\).

Comparing the chasing problem with smooth trajectory generation, the location of the initial and final impulses, \(\theta_1\) and \(\theta_N = \theta_{N+1}\), are unknown; however, the condition in (4.19) must be added to (4.9) and (4.10) to ensure interception of the target. In
CHAPTER 4. SMOOTH TRAJECTORIES FOR TARGET CHASING

38

summary, to generate an $N$-impulse smooth trajectory to chase a target, the following constraints should be satisfied:

\begin{align}
    f_{i1}(p_i, p_{i+1}) &= 0, \quad i = 1, \cdots, N \quad (4.20) \\
    f_{i2}(p_i, p_{i+1}) &= 0, \quad i = 1, \cdots, N \quad (4.21) \\
    f_3(p_1, \cdots, p_{N+1}) := (M_T(0) - M_N^+) + \sum_{i=1}^{N} \left( \frac{a_i}{a_T} \right)^2 (M_i^- - M_{i-1}^+) &= 0. \quad (4.22)
\end{align}

The constraint equation in (4.22) is the result of substituting (4.12) and (4.18) in (4.19). In the problem of $N$-impulse smooth trajectory generation for target chasing, $2N+1$ constraints in (4.20)-(4.22) should be satisfied, while we have $4N-3$ unknowns, i.e., $p_i$ ($i = 2, \cdots, N$) and $\theta_1$. Including the location of the first impulse among the unknowns allows us to study its effect on the long-range rendezvous maneuvers, which will be discussed later.

Some additional assumptions must be considered to find a unique solution to (4.20)-(4.22), since the number of unknowns is more than that of the equations for $N \geq 3$. We implement the Newton method to numerically solve for an $N$-impulse smooth trajectory in chasing problems (see Algorithm 1). Once we have a solution,

\begin{algorithm}
\caption{Smooth trajectory generation for chasing problem.} \label{alg:smooth}
\begin{algorithmic}[1]
\Data: $a_1, e_1, \omega_1, a_T, e_T, \omega_T, \theta_S(0), \theta_T(0), N$, plus $2N-4$ unknowns
\Result: Impulse locations and intermediate orbital elements: $\theta_1$, $p_i$ ($i = 2, \cdots, N$)
\For{$i \leftarrow 1$ to $N$}
\State Form the system of nonlinear equations from (4.20)- (4.21);
\EndFor
\State Add (4.22) to the system of equations, using (4.14)-(4.17).
\State Initialize the Newton algorithm using $\text{ga}()$.
\State Solve using Newton algorithm.
\end{algorithmic}
\end{algorithm}

the magnitude of the velocity change at the $i^{th}$ impulse location $\Delta v_i = \|\Delta v_i\|$ is calculated based on the orbital elements of the smooth transfer trajectory by

\begin{equation}
    \Delta v_i = v_i^+ - v_i^-,
\end{equation}
where,
\[ v_i^+ = \sqrt{\mu \left( \frac{2}{r_i} - \frac{1}{a_{i+1}} \right)} \quad \text{and} \quad v_i^- = \sqrt{\mu \left( \frac{2}{r_i} - \frac{1}{a_i} \right)}. \] (4.24)

The velocity magnitude before and after the \( i^{th} \) impulse are denoted by \( v_i^- \) and \( v_i^+ \), respectively. The radius \( r_i \) is the radial component of the location of the servicer at the \( i^{th} \) impulse in polar coordinates, based on (4.8). Note that at the \( i^{th} \) impulse location, the direction of \( \Delta v_i \) is the same as the servicer’s velocity direction due to the smoothness conditions.

### 4.4 Conclusion

In this chapter, we formed a set of equations needed for designing a family of multi-impulse smooth transfer trajectories for chasing a target in a long-range rendezvous mission. The transfer trajectories were designed for planar motions in a two-body context. The set of \( 2N+1 \) equations were solved by fixing a set of unknowns and using the Newton algorithm, which was initialized using a predefined MATLAB function.
Chapter 5

Trajectory Design Optimization

5.1 Summary

In this chapter, we propose a multi-objective constrained optimization architecture to minimize both the transfer time and control effort of the smooth trajectories we defined in Chapter 4, using the concept of Pareto Optimality. The design variables, constraints, cost functions, and the optimization problem are first discussed. Then, the optimization architecture is developed based on the constrained multi-objective NSGA-II, with some modifications in sorting and mutation operators. In addition, we implement the modified Inverted Generational Distance (IGD+) as the convergence measure along with the modified NSGA-II.

5.2 Transfer Trajectory Optimization Problem

For $N \geq 3$, since we have $2N - 4$ unknowns more than the number of equations in (4.20)-(4.22), there exist infinitely many smooth trajectories for chasing a target. Each set of $2N - 4$ unknowns results in a feasible transfer trajectory with a specific control effort and transfer time. In this chapter, we implement a multi-objective non-dominated sorting genetic algorithm to find the set of Pareto Optimal smooth transfer trajectories. Employing optimization helps enhance the performance of the derived multi-impulse smooth trajectories for chasing a target in the long-range rendezvous phase of OOS.

Problem 5.2.1 (Optimality). Given the control effort and transfer time as Pareto cost functions, and the number of impulses $N$ along with the set of $2N - 4$ unknowns
as design variables, find the set of Pareto-optimal trajectories satisfying (4.20)-(4.22). The optimization must also include constraints on the orbital elements of the transfer trajectories.

The remainder of this section details the employed constrained multi-objective optimization, including the description of design variables, Pareto cost functions, and constraints, to find the optimal transfer trajectories.

### 5.2.1 Design Variables

Since \( N \) is a part of design variables and its change linearly affects the total number of design variables, a maximum number of impulses \( N_{\text{max}} \) is fixed. This introduces a total number of \( 2N_{\text{max}} - 3 \) design variables in the trajectory design optimization. When \( N = 2 \), there is a unique solution to (4.20)-(4.22). If \( 3 \leq N \leq N_{\text{max}} \), a subset of \( 2N - 4 \) design variables identifies the solution to (4.20)-(4.22). The collection of all design variables is denoted by the vector \( \mathbf{x} \in \mathbb{R}^{2N_{\text{max}}-3} \). For an \( N \), we have the freedom to choose the design variables from the vectors \( \mathbf{p}_i, i = 1, \ldots, N \). To capture the required unknowns to solve (4.20)-(4.22) for every value of \( 3 \leq N \leq N_{\text{max}} \), we have to cautiously choose the design variables to include enough information. Any two variables from the components of \( \mathbf{p}_2 \) and \( \mathbf{p}_3 \), and \( \theta_1 \) (9 variables in total) can be chosen to be two of the design variables for \( N_{\text{max}} = 3 \), e.g., the vectors \( \mathbf{x} = [N, \theta_1, \omega_2] \) or \( \mathbf{x} = [N, a_2, e_3] \) are acceptable choices. However, some parameters are preferred over the others, e.g., since \( \theta_i \) and \( \omega_i \) appear in the argument of trigonometric functions they are more reasonable choices comparing to \( a_i \) and \( e_i \). As it is evident in (4.9)-(4.10), knowing \( \theta_1 \) and \( \omega_2 \) for \( N = 3 \) gives explicit answers for \( a_2 \) and \( e_2 \), without needing a nonlinear solver. In the case of \( N_{\text{max}} = 4 \), we keep the first two design variables as the ones chosen for \( N_{\text{max}} = 3 \) and add two more design variables from \( \mathbf{p}_2 \), \( \mathbf{p}_3 \), and \( \mathbf{p}_4 \), i.e., the vectors \( \mathbf{x} = [N, \theta_1, \omega_2, \omega_3, \omega_4] \) or \( \mathbf{x} = [N, \theta_1, \omega_2, e_3, e_4] \) can serve as the vector of design variables. But as discussed, the former choice is preferred. This process can be extended to any maximum number of impulses considered in the optimization. Note that in any design candidate \( \mathbf{x} \) if \( N < N_{\text{max}} \), only a subset of design variables identifies the cost functions.
5.2.2 Constraints

In every real-world optimization problem, certain physical or manufacturing requirements ought to be satisfied, leading to a set of constraints on the design variables. In the problem of trajectory design optimization, four different types of constraints are considered:

1. To meet the smoothness and chasing requirements of the $N$-impulse transfer trajectories, (4.20)-(4.22) are the first set of constraints.

2. A belt of orbits is defined to be a set of neighbouring elliptic orbits of the Earth. Such a belt can be identified by the following inequalities on $a$, $e$, and $\omega$:

$$a_L < a < a_U \text{ and } e_L < e < e_U \text{ and } \omega_L < \omega < \omega_U,$$

where the subscripts $L$ and $U$ indicate the lower and upper bound for an orbital element, respectively. Therefore, to avoid a certain belt of orbits, a set of inequalities are imposed on the parameters of the intermediate orbital elements ($i = 2, \ldots, N$):

$$a_i \geq a_U \text{ or } a_i \leq a_L, \text{ or } e_i \geq e_U \text{ or } e_i \leq e_L, \text{ or } \omega_i \geq \omega_U \text{ or } \omega_i \leq \omega_L. \quad (5.1)$$

A belt of orbits may represent a populated orbital region where the risk of collision with other spatial objects is high. Note that avoiding a belt of orbits guarantees that the servicer never uses any orbit in the belt for transfer; however, it may cross the belt in its path to the target.

3. To ensure that the servicer stays within an orbital region, constraints may also be applied on functions of the intermediate orbital elements. For example, to always stay within LEO throughout a long-range rendezvous with a target, the following inequalities must be satisfied to ensure that the maximum and minimum altitude of the servicer remain between 100-2000 kilometers. For $i = 2, \ldots, N$:

$$a_i(1 + e_i) \leq 8371\text{km and } a_i(1 - e_i) \geq 6471\text{km.} \quad (5.2)$$

4. The maximum amount of available $\Delta v$ at each impulse is subject to some limitations. Defining $\Delta v_U$ as the maximum available velocity increment at each
impulse, the following set of inequalities must be satisfied for \( i = 1, \cdots, N \) to find a feasible transfer trajectory.

\[
\Delta v_i \leq \Delta v_U. \tag{5.3}
\]

### 5.2.3 Cost Functions

Both total Transfer Time (TT) and Control Effort (CE) are critical performance indexes in the servicer trajectory planning problem. For any impulsive trajectory design in the two-body problem context, TT and CE can be respectively computed by the following functions:

\[
J_t = t_f(p_1, \cdots, p_N), \tag{5.4}
\]

\[
J_c = \sum_{i=1}^{N} |\Delta v_i(p_i, p_{i+1})|. \tag{5.5}
\]

The functions \( t_f \) and \( \Delta v_i \) are already defined in (4.12) and (4.23), respectively.

### 5.2.4 Optimization Problem

Let \( J \) denote the collection of the objective functions \( J_t \) and \( J_c \), i.e.,

\[
J = [J_t, J_c]^T. \tag{5.6}
\]

The optimality problem in Problem 5.2.1 can now be formulated as the following multi-objective constrained optimization:

\[
\mathbf{x}^* = \arg \min_{\mathbf{x}} J(\mathbf{x}) \tag{5.7}
\]

subject to (4.20)-(4.22) and (5.1)-(5.3).

Here, \( \mathbf{x}^* \) denotes a member of the set of optimal solutions. The set of equality constraints in (4.20)-(4.22) are already taken into consideration when defining the design variables as a subset of parameters appearing in \( p_2 \) to \( p_{N_{\text{max}}} \) along with \( \theta_1 \) and \( N \).

Let \( g_j(\mathbf{x}) \) for \( j = 1, \cdots, m \) be a set of functions of the design variables. A constrained optimization problem with a set of inequality constraints in the form of
\[ g_j(x) \leq 0 \quad (j = 1, \ldots, m) \] can be converted into an unconstrained problem. Note that the constraints in (5.1)-(5.3) can be reformatted to this form. Let \( \Phi \) be a cost function defined as:

\[
\Phi(x; \alpha_t, \alpha_c) = J(x) + [\alpha_t, \alpha_c]^T \sum_{j=1}^{m} G_j(x),
\]

where \( G_j(x) = \max\{0, g_j(x)\} \) and \( \alpha_t, \alpha_c \gg 1 \) are two penalty constants. The following unconstrained multi-objective optimization problem is equivalent to (5.7):

\[
x^* = \arg \min_x \Phi(x; \alpha_t, \alpha_c). \tag{5.8}
\]

In a multi-objective optimization, if two objectives are cooperative, i.e., changing design variables has the same effect on both, the optimal solution is a single point. However, if we deal with two competitive objectives, we have multiple optimal solutions whose collection is called the Pareto frontier set. In the trajectory optimization problem in (5.8), the Pareto cost functions \( J_t \) and \( J_c \) are normally competitive and the solutions to (5.8) form the Pareto frontier set. A solution \( x^* \) in the Pareto frontier set dominates every other feasible solution to (5.8) in the space of objectives. Point A dominates point B in the Pareto sense, if A is better than B in at least one objective function and not worse with respect to all other objective functions. Then, Pareto frontier set is the collection of all of the solutions that cannot dominate one another \[70\], i.e., no solution is superior to others.

### 5.3 Non-Dominated Sorting Genetic Algorithm

Non-dominated Sorting Genetic Algorithm (NSGA-II) is a guided random search multi-objective optimization algorithm with the capability of exploring the diverse regions of the solution space. This algorithm proposes a direct approach using the concept of Pareto-optimality for solving multi-objective problems with four main advantages to its direct rival multi-objective optimization algorithms: (i) it uses a fast non-dominated sorting procedure, (ii) it is an elitist-preserving approach, (iii) it uses an operator for exploring diverse regions, and (iv) it provides strategies for handling constraints \[42\]. The flowchart in Figure 5.1 demonstrates the implementation of
NSGA-II to find the optimal transfer trajectories from (5.8). The steps of NSGA-II for finding the optimal answers are elaborated in the following.

### 5.3.1 Initialization

Considering $N_{\text{max}}$ number of impulses, $2N_{\text{max}} - 3$ design variables must be initialized for $K$ number of population. This initialization can be performed whether randomly or by a specific pattern. For example, if there are some "good" solutions available, they can be included in the initial population. The original NSGA-II sorts the population after the initialization. However, we perform sorting after producing the children to save some computations.

### 5.3.2 Cost Function Evaluation

To evaluate the cost functions, the nonlinear solver in Algorithm 1 calls the initialized design variables to find $K$ smooth transfer trajectories that chase a dynamic target by $N \leq N_{\text{max}}$ number of impulses based on (4.20)-(4.22). The Newton method as the nonlinear solver has a fast convergence near the optimal point, but it is too sensitive
to the initialization of the parameters computed by this solver. To address this drawback, the \texttt{ga()} function from MATLAB with a low number of iterations is used to find a set of initial parameters for which the functions $f_{i1}$, $f_{i2}$, and $f_3 \ (i = 1, \ldots , N)$ are close enough to zero. The transfer time and control effort are calculated based on (4.12) and (4.23) after computing the feasible intermediate orbital elements from the Newton solver. Moreover, those solutions which do not satisfy the constraints in (5.1)-(5.3) are penalized using $\alpha_t$ and $\alpha_c$ in the cost function $\Phi$ to have less chance to be part of the next generation.

5.3.3 Sorting Operators

The NSGA-II uses the rank and crowding distance operations in the space of objectives including CE and TT to sort the individuals in a generation [42]. For each individual $\mathbf{x}_k \ (k = 1, \ldots , K)$ in a generation, the rank operator is evaluated based on the following algorithm. Two entities are calculated: (i) domination count $n_{\mathbf{x}_k}$ that is the number of individuals dominating $\mathbf{x}_k$, and (ii) a set of individuals $S_{\mathbf{x}_k}$ that $\mathbf{x}_k$ dominates. Initially, $n_{\mathbf{x}_k} = 0$ and $S_{\mathbf{x}_k} = \emptyset$. Then, we form $S_{\mathbf{x}_k}$ and compute $n_{\mathbf{x}_k}$ by comparing all individuals in the generation. If $n_{\mathbf{x}_k} = 0$, the rank of $\mathbf{x}_k$, denoted by $\text{rank}(\mathbf{x}_k)$, is assigned to be 1. For every $\mathbf{x}_k$ with rank 1, we reduce the domination count of every element in the set $S_{\mathbf{x}_k}$ by 1. Excluding the rank 1 individuals, if the recomputed $n_{\mathbf{x}_k} = 0$, then $\text{rank}(\mathbf{x}_k) = 2$. Repeating the same procedure, we rank all of the individuals in the generation. We denote the set of all individuals with rank $l$ by $\text{PF}_l$ (see Figure 5.2). Clearly, the lower the rank, the closer the individuals are to the Pareto-optimal solutions. The algorithmic evaluation of the rank operator is detailed in 2.

The Crowding Distance (CD) operator guides the algorithm at each stage towards a uniform Pareto frontier set, and helps generate diverse solutions. In a generation, the CD operator is evaluated for an individual $\mathbf{x}_k$, based on the following algorithm. The two extreme individuals in $\text{PF}_l$ are identified:

$$\mathbf{x}_l^t = \arg \max_{\mathbf{x}_k \in \text{PF}_l} J_t(\mathbf{x}) \quad \& \quad \mathbf{x}_l^c = \arg \max_{\mathbf{x}_k \in \text{PF}_l} J_c(\mathbf{x}).$$

For the extreme individuals $\mathbf{x}_l^t$ and $\mathbf{x}_l^c$ with rank $l$, we define the CD to be infinity. We sort the members of $\text{PF}_l$ based on the value of one of the objective functions.
**Algorithm 2:** Ranking [42].

**Data:** $x_k$, $J_l(x_k)$, $J_c(x_k)$ for $k = 1, \cdots, K$

**Result:** $S_{x_k}$ and $n_{x_k}$ for $k = 1, \cdots, K$, and $PF_l$

**for** $x_k$, $k \leftarrow 1$ to $K$ **do**

$S_{x_k} = \emptyset$

$n_{x_k} = 0$

**for** $x_{k'}$, $k' \leftarrow 1$ to $K$ **do**

- **if** $x_k$ dominates $x_{k'}$ **then**
  - $S_{x_k} = S_{x_k} \cup \{x_{k'}\}$
  - **else**
    - $n_{x_k} = n_{x_k} + 1$

**end**

**end**

**if** $n_{x_k} = 0$ **then**

- $\text{rank}(x_k) = 1$
- $PF_1 = PF_1 \cup \{x_k\}$

**end**

$l = 1$ **while** $PF_l \neq \emptyset$ **do**

- $Q = \emptyset$ **Used to store the member of the next rank**

  **for** $x_k \in PF_l$ **do**

  - **for** $x_{k'} \in S_{x_k}$ **do**

    - $n_{x_{k'}} = n_{x_{k'}} - 1$

  **end**

  **end**

  **if** $n_{x_{k'}} = 0$ **then**

  - $\text{rank}(n_{x_{k'}}) = l + 1$
  - $Q = Q \cup \{n_{x_{k'}}\}$

  **end**

  $l = l + 1$

  $PF_l = Q$

**end**
Then, for the individuals in $\text{PF}_l$ except the extremes we define the CD to be

$$\text{CD}(x_k) = \frac{J_c(x_{k+1}) - J_c(x_{k-1})}{J_c(x^*_l) - J_c(x^*_t)} + \frac{J_t(x_{k+1}) - J_t(x_{k-1})}{J_t(x^*_l) - J_t(x^*_t)}. \quad (5.9)$$

Note that $x_{k-1}$, $x_k$, and $x_{k+1}$ are three consecutive individuals in the set $\text{PF}_l$. The higher the CD, the more preferred individual is in $\text{PF}_l$, since it identifies rare regions in the search space. Figure 5.3 demonstrates the crowding distance in a set $\text{PF}_l$, and the algorithmic evaluation of the CD operator is detailed in 3.

**Algorithm 3: Crowding Distance [42].**

<table>
<thead>
<tr>
<th>Data:</th>
<th>$J_t(x_k)$ and $J_c(x_k)$ for $x_k \in \text{PF}_l$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result:</td>
<td>CD($x_k$) for $x_k \in \text{PF}_l$</td>
</tr>
<tr>
<td>$x^*<em>l = \arg \max</em>{x_k \in \text{PF}_l} J_t(x)$</td>
<td></td>
</tr>
<tr>
<td>$x^*<em>t = \arg \max</em>{x_k \in \text{PF}_l} J_c(x)$</td>
<td></td>
</tr>
<tr>
<td>$\text{PF}_l = \text{sort}(\text{PF}_l, J_t(x) \text{ or } J_c(x) )$</td>
<td></td>
</tr>
<tr>
<td>for $x_k \in \text{PF}_l$ do</td>
<td></td>
</tr>
<tr>
<td>if $x_k = x^<em>_l$ or $x_k = x^</em>_t$ then</td>
<td></td>
</tr>
<tr>
<td>CD($x_k$) = $\infty$</td>
<td></td>
</tr>
<tr>
<td>else</td>
<td></td>
</tr>
<tr>
<td>CD($x_k$) = $\frac{J_c(x_{k+1}) - J_c(x_{k-1})}{J_c(x^<em>_l) - J_c(x^</em><em>t)} + \frac{J_t(x</em>{k+1}) - J_t(x_{k-1})}{J_t(x^<em>_l) - J_t(x^</em>_t)}$</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>

Figure 5.2: Non-dominated sorting ranking. Figure 5.3: Crowding distance for two objectives.
5.3.4 Binary Tournament Selection

In any step of the optimization that we need to select parents, we use binary tournament selection method. In this method, two individuals are randomly chosen from the population and the one with the lower rank is kept in the process. If the two individuals have the same rank, then the one with higher CD is preferred to become the parent. To select other parents, we repeat the same procedure [71].

5.3.5 Arithmetic Crossover

Crossover is an inheritance operator, that causes the children to have attributes from all parents. Initially, a crossover rate (between 0 and 1) is fixed that is the indicator of the execution probability of this operator. During the optimization, each time that the crossover operator is called, a random number between 0 and 1 is generated. If this number is smaller than the crossover rate, then this operator is executed. During crossover, a random $2N_{max} - 3$ dimensional vector $\beta$ whose elements are between 0 and 1 is generated, based on which we define the children $x_{c1}, x_{c2}$ of the two selected parents by the binary tournament selection, $x_{p1}, x_{p2}$.

\[
x_{c1} = \beta \ast x_{p1} + (1 - \beta) \ast x_{p2}, \quad \& \quad x_{c2} = \beta \ast x_{p2} + (1 - \beta) \ast x_{p1},
\]

where the operator $\ast$ refers to the element-wise multiplication of two vectors and the vector $1$ is the $2N_{max} - 3$ dimensional vector whose elements are all equal to 1 [71].

5.3.6 Mutation

In the iteration $1 \leq d \leq D$, where $D$ is the maximum number of iterations, a number of individuals are randomly selected based on the binary tournament selection to be mutated with a probability, to explore the search region. A mutation probability distribution is fixed at the initialization phase of the NSGA-II algorithm. This distribution allows the mutation operator to execute with a higher probability at the beginning of the optimization, which leads to avoiding convergence to local optima. In this thesis, the mutation probability $PM$ in the iteration $d$ is defined by (see Figure 5.4)
\[ PM(d) = PM_0(1 + \cos \frac{\pi d}{D}), \]

where \( PM_0 \) is a parameter, fixed at the beginning of the optimization loop. Each time that the mutation operator is called for an individual, \( 2N_{\text{max}} - 3 \) random numbers between 0 and 1 are generated. If any of these numbers is smaller than \( PM(d) \), then the corresponding design variable in the individual is randomly reassigned within its bounds [71].

### 5.3.7 Replacement

After applying the crossover and mutation operators on the current population, \( K \) children are generated and added to the current population. This new population of \( 2K \) individuals is sorted first based on their rank and then their crowding distance. To maintain a fix number for the population in each generation, \( K \) best individuals are truncated. Note that the new \( K \) individuals must be sorted again, since the sorting operators depend on all members of a population. This new \( K \) individuals are the population for the next generation (iteration) [42].

### 5.3.8 Termination Criteria and Convergence

In an optimization, several termination criteria can be considered: (i) maximum number of iterations, (ii) maximum number of iterations with no improvement, (iii)
maximum allowed CPU time, and (iv) reaching an admissible fitness. For multi-objective problems, another criterion that is widely used to measure both convergence and diversity is Inverted Generational Distance (IGD) [72]. This performance metric is defined in the following. Let PF\(^*\) denote the ideal Pareto frontier, defined as the one-dimensional hyper-plane in the performance space according to the minimum values of \(J_t\) and \(J_c\) and the constraints over them. If no constraints are specified for the objective functions, the PF\(^*\) includes two lines parallel to the objective axes passing through the minimum values of \(J_t\) and \(J_c\). The two lines meet at a point closest to the origin, coined as the ideal point (Figure 5.5). Considering a set of reference points \(P^*\), i.e., uniformly distributed points along the ideal Pareto frontier, and PF\(_1\) as the set of non-dominated solutions generated by the optimization algorithm, the modified Inverted Generational Distance (IGD\(^+\)) is defined as:

\[
\text{IGD}^+ = \sum_{P \in P^*} \frac{\text{dist}(P, \text{PF}_1)}{|P^*|}.
\]

Here, \(\text{dist}(P, \text{PF}_1)\) is the minimum Euclidean distance \(d(P, Y)\), between \(P \in P^*\) and the point \(Y\) in PF\(_1\), and \(|P^*|\) is the cardinality of the set \(P^*\). The Euclidean distance \(d(P, Y)\) for the two-objective problem is calculated by

\[
d(P, Y) = \sqrt{\sum_{j=1}^{2} \max(P_j - Y_j, 0)^2}.
\]

As opposed to the IGD, the modified measure IGD\(^+\) can differentiate the quality of PF\(_1\) when it is nondominated by the reference points in \(P^*\). If \(|P^*|\) is large enough, IGD\(^+\) can measure both diversity and convergence of the optimization algorithm. The details of finding the reference points \(P^*\) are explained in [72]. Minimizing IGD\(^+\) is a convergence criterion that indicates how far we are from the ideal Pareto frontier. In this thesis, minimizing the IGD\(^+\) is numerically studied to ensure the convergence of the optimizations in different case studies.

### 5.4 Conclusion

In this chapter, we proposed a multi-objective constrained optimization based on the NSGA-II to identify the set of Pareto Optimal trajectories among the smooth
trajectories derived in the previous chapter. Transfer time and control effort were considered as the two Pareto cost functions. The constraints were restrictions on the maximum permitted amount of $\Delta v_i$ and the orbital elements corresponding to the intermediate orbits. The design variables were the orbital elements of the intermediate orbits, and the number and the location of the impulses. To study the convergence of the modified NSGA-II, the $IGD^+$ was implemented.
Chapter 6

Simulation Results

6.1 Summary

In this chapter, we aim to evaluate the proposed $N$-impulse optimal smooth trajectory generator, presented in Chapters 4 and 5, in three case studies. In all case studies, we solve the unconstrained multi-objective optimization problem and investigate the effect of constraints in the formation of the Pareto frontier. To examine the efficiency of the proposed optimization algorithm, we compare the resulted unconstrained Pareto frontier with that of the MOGA. The optimization parameters of the MOGA and the modified NSGA-II are selected the same, wherever possible. Moreover, the unconstrained solutions closest to the ideal points are compared with the optimal Lambert approach developed for chasing a satellite, in this thesis. In the proposed Lambert approach, the design variables are the transfer time and the location of the first impulse, and the control effort is the objective function. The optimization performed to find the optimal Lambert solutions uses a single-objective GA with the same parameters. Note that the trajectories generated by the Lambert approach are not necessarily smooth. However, the proposed 2-impulse smooth trajectories can be considered as a specific case of the Lambert approach, where the smoothness constraint is applied.

6.2 Simulation Setup

In the following case studies, it is assumed that $\theta_S(0) = 270$ deg, $\theta_T(0) = 0$ deg, $N_{max} = 5$. In the constrained optimizations, we consider the following constraints: (i) $\Delta v_i \leq 6$ km/sec for $i = 1, \cdots, N$, and (ii) staying within LEO regime, i.e., (5.2).
The typical value of the crossover rate used in the literature is $[0.8, 0.95]$ [42]. For constant mutation functions, the rate is selected to be reciprocally proportional to the number of design variables, which is almost 0.4 in our case studies; hence, $PM_0 = 0.4$. Based on an investigation conducted on the optimization time and the quality of the produced solutions, the number of population and the maximum number of iterations are obtained to be $K = 50$ and $D = 200$, respectively. Moreover, the optimal number of reference points in calculating the $IGD^+$ is found to be around $|P^*| = 100$. To ensure convergence, the behaviour of $IGD^+$ over the course of iterations is monitored. We consider convergence of $IGD^+$ to a number less than 1 as the indicator of the multi-objective optimization convergence.

### 6.3 Case Study 1

In this case study, the initial and final orbital elements are:

$$a_S = 13756 \text{ km} \quad e_S = 0.5 \quad \omega_S = 10 \text{ deg}$$

$$a_T = 13756 \text{ km} \quad e_T = 0$$

This case study is simulated with and without constraints. The Pareto Optimal solutions for the unconstrained optimization along with the convergence criterion $IGD^+$, as well as the transfer trajectories for the two extreme solutions are shown in Figures 6.3(a)-6.3(d), respectively. The Pareto frontier set in this case only includes solutions with three and four number of impulses. The initial/final locations of the servicer and target, and the location of impulses are shown in Figures 6.3(c)-6.3(e). The blue-dashed curve is the proposed optimal smooth transfer trajectory. It can be observed in Figure 6.3(c), which is the trajectory for the solution with the minimum time and maximum control effort, the servicer loiters in an inner orbit with high eccentricity until it catches the target. However, in the solution with minimum control effort and maximum time, the servicer waits in the first orbit for a long period of time and transfers to the target orbit using elliptic arcs with low eccentricity. The optimal transfer time and control effort for this case study are shown in Table 6.1. The minimum transfer time corresponds to a trajectory with $N = 3$, which is almost 16 times faster than the second extreme solution. However, the extreme solution with minimum $J_c$ has four impulses and demonstrates a control effort that is ten
times smaller than that of the other extreme solution. It is worth noticing that all of the 50 initial population become non-dominating solutions. As we yet to apply the constraints, Figure 6.3(a) shows all the possible solutions to chase a target with the given initial condition, even if they are not feasible. Figure 6.3(a) also depicts the Pareto Frontier obtained using MOGA (blue stars). Both algorithms have the same maximum iteration and initial population. However, the computational time of solving the problem using MOGA is almost 2 times more than that of the NSGA-II, due to the increased computational complexity of the MOGA. Further, the MOGA can converge to only 10 optimal solutions for this case study, while all of them are dominated by the solutions of NSGA-II. To have a reasonable comparison between the proposed optimal trajectory design and the optimal Lambert, the solution closest to the ideal point is compared with the optimal Lambert in Figure 6.3(e). In the proposed optimal Lambert, the servicer waits in the initial orbit to find the right location for the first impulse, which is shown with the green star. Then, it catches the target with the Lambert trajectory shown with the red curve. It can be interpreted from Table 6.1 that the $J_c^*$ of the Lambert solution is 4 times bigger than that of the proposed optimal transfer trajectory, with almost 30% smaller transfer time. The Pareto Optimal solutions after applying constraints in (5.2) are shown in Figure 6.3(f). For the constrained optimization we find 40 non-dominated solutions out of 50 population, with three impulses. We plot the changes in the planar orbital elements in time for the trajectory closest to the ideal point in Figure 6.2 to visualize the dynamics of the optimal solution.

<table>
<thead>
<tr>
<th>Method</th>
<th>$N$</th>
<th>$J_c$ [km/s]</th>
<th>$J_t$ [s] ($\times 10^3$)</th>
<th>Design variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extreme point 1</td>
<td>3</td>
<td>12.46</td>
<td>1.72</td>
<td>$\omega_2, \theta_1$</td>
</tr>
<tr>
<td>Extreme point 2</td>
<td>4</td>
<td>1.25</td>
<td>28.21</td>
<td>$\omega_2, \omega_3, \omega_4, \theta_1$</td>
</tr>
<tr>
<td>Closest to ideal point</td>
<td>3</td>
<td>1.97</td>
<td>6.03</td>
<td>$\omega_2, \theta_1$</td>
</tr>
<tr>
<td>Optimal Lambert</td>
<td>2</td>
<td>7.74</td>
<td>4.36</td>
<td>$t_f, \theta_1$</td>
</tr>
</tbody>
</table>
Figure 6.1: Case Study 1; (a) Unconstrained Pareto frontier (b) IGD+ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier
6.4 Case Study 2

In this case study, the transfer trajectories from an eccentric orbit to a circular orbit in LEO are studied. The initial and final orbital elements are:

\[ a_S = 7000 \text{ km} \quad e_S = 0.05 \quad \omega_S = 10 \text{ deg} \]

\[ a_T = 7500 \text{ km} \quad e_T = 0 \]

The determined unconstrained Pareto Optimal solutions are shown in Figure 6.4(a). Also, the transfer trajectories for the two extremes (both minimum transfer time and minimum control effort are for \( N = 3 \)) and the solution closest to the ideal point (\( N = 4 \)) are depicted in Figures 6.4(b), 6.4(c), and 6.4(d), respectively. Given the termination and convergence criteria defined for the modified NSGA-II (Figure 6.4(b)), 50 Pareto Optimal trajectories were found in this case study. Evidently, the trajectory corresponding to the minimum transfer time drastically reduces the mission duration (almost 9 times), with a large compromise in the control effort (almost 43 times), comparing to the one with minimum control effort. Figure 6.4(a) also depicts the Pareto Frontier obtained using the MOGA (blue stars). Similar to the previous case study, the Pareto frontier consists of 17 solutions, all of which are dominated by the NSGA-II solutions. The optimal solution closest to the ideal point is compared with that of the optimal Lambert approach in Table 6.2 and Figure
6.4(e), proving that the Lambert solution is not efficient. The optimal control effort and the corresponding transfer time in the Lambert approach are almost 8.6 and 1.3 times larger than those of the proposed optimal trajectory generator, respectively. Figure 6.4(f) shows the Pareto frontier after enforcing the trajectories to be in LEO and considering the limitation in $\Delta v_i$. Here, the number of constrained optimal solutions is 50 and they include most of the three-impulse unconstrained solutions with $J^*_c \leq 0.7$. The dynamics of the planar orbital elements in time for the trajectory closest to the ideal point is depicted in Figure 6.4.

Table 6.2: Summary of the optimal results: Case Study 2

<table>
<thead>
<tr>
<th>Method</th>
<th>$N$</th>
<th>$J_c$ [km/s]</th>
<th>$J_t$ [s] ($\times 10^3$)</th>
<th>Design variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extreme point 1</td>
<td>3</td>
<td>11.14</td>
<td>1.13</td>
<td>$\omega_2, \theta_1$</td>
</tr>
<tr>
<td>Extreme point 2</td>
<td>3</td>
<td>0.26</td>
<td>10.02</td>
<td>$\omega_2, \theta_1$</td>
</tr>
<tr>
<td>Closest to ideal point</td>
<td>4</td>
<td>1.59</td>
<td>3.46</td>
<td>$\omega_2, \omega_3, \omega_4, \theta_1$</td>
</tr>
<tr>
<td>Optimal Lambert</td>
<td>2</td>
<td>13.64</td>
<td>4.58</td>
<td>$t_f, \theta_1$</td>
</tr>
</tbody>
</table>

6.5 Case Study 3

In this case study, transferring between two circular orbits in LEO has been studied.

$$a_S = 6500 \text{ km} \quad e_S = 0$$

$$a_T = 8300 \text{ km} \quad e_T = 0$$

After satisfying the termination criteria based on Figure 6.5(b), 50 optimal transfer trajectories were found in this case study (see Figure 6.5(a)), with the extreme solutions correspond to $N = 4$ (Figure 6.5(d)) and $N = 2$ (Figure 6.5(c)), and the solution closest to the ideal point with $N = 4$ (Figure 6.5(e)). Regarding Table 6.3, the CE for extreme point 1 is almost 11 times smaller than that for extreme point 2. However, the transfer time for the extreme solution corresponding to minimum $J_t$ is almost 2.5 times smaller than that of the extreme two-impulse solution. Figure 6.5(a)
Figure 6.3: Case Study 2; (a) Unconstrained Pareto frontier (b) IGD+ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier
CHAPTER 6. SIMULATION RESULTS

Figure 6.4: Case Study 2; Dynamics of the optimal solution closest to the ideal point also depicts the Pareto Frontier solutions obtained using MOGA (blue stars). Like the previous case studies, the converged solutions to the first rank Pareto frontier are only 5, and all of these solutions are dominated by the NSGA-II solutions. In the Lambert trajectory shown in Figure 6.5(e), the servicer waits approximately a period before applying the first impulse (green star), which generates the Lambert trajectory (red curve). The $J_c$ and $J_t$ for the Lambert solution are respectively 2.6 and 2.1 times larger than those for the optimal solution closest to the ideal point (Figure 6.5(e)). Moreover, although the number of the design variables of the smooth two-impulse trajectory generator is less than that of the optimal Lambert, it outperforms the Lambert solution in both aspects of the objectives and the computational cost. The constrained Pareto-optimal solutions in LEO are also presented in Figure 6.5(f), which includes 17 solutions with $J_c \leq 0.9$ from the unconstrained Pareto-optimal trajectories. The planar orbital element changes versus time are depicted in Figure 6.6.

6.6 Computational Complexity

As indicated in [42], the computational complexity of the NSGA-II for a problem with two objective functions is $\mathcal{O}(2K^2)$. The Newton method is implemented to evaluate the objective functions by solving a system of $2N_{\max} + 1$ nonlinear equations with the
Figure 6.5: Case Study 3; (a) Unconstrained Pareto frontier (b) IGD+ (c) Extreme point 1 (d) Extreme point 2 (e) Ideal point vs. Lambert (f) Constrained Pareto frontier
Table 6.3: Summary of the optimal results; Case Study 3

<table>
<thead>
<tr>
<th>Method</th>
<th>N</th>
<th>$J_c$ [km/s]</th>
<th>$J_t$ [s] ($\times 10^3$)</th>
<th>Design variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extreme point 1</td>
<td>2</td>
<td>0.81</td>
<td>4.52</td>
<td>-</td>
</tr>
<tr>
<td>Extreme point 2</td>
<td>4</td>
<td>8.76</td>
<td>1.81</td>
<td>$\omega_2, \omega_3, \omega_4, \theta_1$</td>
</tr>
<tr>
<td>Closest to ideal point</td>
<td>4</td>
<td>2.92</td>
<td>2.67</td>
<td>$\omega_2, \omega_3, \omega_4, \theta_1$</td>
</tr>
<tr>
<td>Optimal Lambert</td>
<td>2</td>
<td>7.66</td>
<td>5.74</td>
<td>$t_f, \theta_1$</td>
</tr>
</tbody>
</table>

computational complexity $\mathcal{O}((2N_{max} + 1)^3)$ [73]. It is initialized using $\text{ga}()$ with the computational complexity $\mathcal{O}(F(2N_{max} + 1))$, where $F$ is the number of population multiplied by the number of generations. Overall, the computational complexity of the proposed algorithm for solving (5.8) is $\mathcal{O}(K^2N_{max}^3)$.

6.7 Conclusion

To demonstrate the efficacy of the proposed methodology in Chapter 5, we compared the obtained optimal solutions with those generated by the MOGA and a developed optimal Lambert approach for chasing a target, in three case studies. In addition to the inherent complexity of the MOGA that resulted in slower optimization process, it was evident from the case studies that the proposed optimization based on the
NSGA-II was superior to the MOGA. The size of the Pareto frontier using the NSGA-II was larger and its solutions dominated those that obtained from the MOGA. In the Lambert approach, the design variables were the transfer time and the impulse location in the initial orbit, while the objective function was the control effort. In the last case study, it was shown that the computational complexity of the optimal Lambert is higher than that of the proposed optimal trajectory. This was because of the fact that the number of design variables in the optimal Lambert problem was more than that of the proposed trajectory generator, despite having the same number of impulses. Overall, the multi-impulse smooth trajectory design methodology provided improved solutions in terms of both transfer time and control effort, as observed from Tables 6.1-6.3.
Chapter 7

Nonlinear Model Predictive Control

7.1 Summary

A nonlinear model predictive control is proposed in this chapter in order to follow an impulsive smooth reference trajectory and catch a target satellite at the final time of the mission, in a $J_2$-perturbed environment. The satellite dynamics is simulated under the induced effects of the $J_2$ perturbation and a bounded random acceleration to compensate for the error of neglecting other types of perturbations. The impulsive control signals are modelled with a constant accelerations over the burning time, which is the time that the thruster is on. To solve the optimization problem that is defined in the controller, a genetic algorithm is implemented and is discussed in this chapter.

7.2 Nonlinear Process Model

In Chapter 5, the trajectories were designed in the context of the two-body problem without involving any disturbances. In this chapter, we consider the following control problem to compensate for the errors of neglecting the most dominant $J_2$ perturbations in an orbit transfer.

**Problem 7.2.1** (Tracking multi-impulse chasing trajectories). Given a desired multi-impulse smooth trajectory for chasing a target, derived from (4.20)-(4.22), find $\Delta v_i$ at every impulse time $t_i$ such that the satellite’s trajectory under $J_2$ perturbations remains close to the desired trajectory and the satellite arrives near the target at the end of transfer.
To address this problem, a Nonlinear Model Predictive Control (NMPC) is proposed in this section. In most of the predictive control families, the idea is to calculate a control sequence minimizing an objective function using a model to predict the process output at future time instants (horizon) and applying the first control signal at each step (receding strategy) [74]. In the following, we introduce the nonlinear process model for the multi-impulse trajectory tracking problem and the objective function that should be minimized in the proposed NMPC. We also provide an optimization platform based on the GA to compute control signals. The variable horizon that is considered in this control problem is the time difference between every two impulses denoted by $T_i, i = 1, \cdots, N$ (Figure 7.1). Consider the following discrete-input nonlinear system governing the evolution of the mean EE for perturbed satellites:

$$
\dot{x}(t) = A(x(t)) + \sum_{i=1}^{N} \frac{\partial \epsilon(\hat{x})}{\partial (\hat{x})} B(\hat{x}(t)) u_i \delta(t - t_i).
$$

(7.1)

Here, $x \in \mathbb{R}^6$ is the state space vector and includes mean equinoctial orbital elements, $A : \mathbb{R}^6 \to \mathbb{R}^6$ is the nonlinear function of secular variation effects induced by $J_2$ perturbations given in (3.40), $u \in \mathbb{R}^3$ is the control input in the spacecraft orbital frame, $\delta$ is the Dirac-delta function, $t_i$ is the time of applying the $i^{th}$ control impulse, and $N$ is the maximum number of control impulses. Also, $\epsilon(\hat{x})$ is a function that transforms osculating elements to mean elements ($x = \epsilon(\hat{x})$), and can be approximated by a $6 \times 6$ matrix whose diagonal elements are equal to 1 and the off-diagonal terms being of order $J_2$ or smaller [75]. Moreover, $B$ is the $6 \times 3$ control influence matrix of Gauss's variational equations that relates accelerations in the spacecraft orbital frame to changes in the orbital elements [76]. In the control application, it is acceptable to approximate the term $\frac{\partial \epsilon(\hat{x})}{\partial (\hat{x})} B(\hat{x}(t))$ with $B(x(t))$ that is
The nonlinear process model that is considered for the evolution of the satellite in $J_2$ perturbed orbits includes the solution to (7.1) in addition to the first order long periodic effects, i.e., (A.1)-(A.7).

\[ \hat{x} \approx x - J_2 R_e^2 x^{lp}, \quad (7.5) \]

where $x^{lp} = [a^{lp} \ h^{lp} \ k^{lp} \ p^{lp} \ q^{lp} \ \lambda^{lp}]$. Since the NMPC feedback control law is

\[ B(x(t)) = \begin{pmatrix}
\frac{2}{m_2} (k \sin L - h \cos L) & 2W \\
-\eta_2 \cos L & \frac{2W}{n\eta_2} \\
-\eta_2 \cos L & \frac{2W}{n\eta_2} \\
0 & 0 \\
0 & 0 \\
-\eta_2 (W(h \sin L + k \cos L) + 2\eta_2) & -\eta_2 \frac{1+W}{1+\eta_2} (h \cos L - k \sin L) & -\eta_2 \frac{W}{n\eta_2} (p \cos L - q \sin L)
\end{pmatrix} \]

where,

\[ W = 1 + k \cos L + h \sin L, \quad L = \theta + \omega + \Omega. \quad (7.3) \]

The impulsive control signal $u_i \delta(t - t_i)$ (as shown in Figure 7.1) is the velocity increment $\Delta v_i$ over a small amount of time. As the impulsive control may cause severe damages to the satellite’s equipment and it is not achievable practically, we assume that the thruster is on for a certain amount of time. This burn time $\Delta t_{burn}$ is a function of satellite’s initial mass $m_0$, thruster $T$, specific thrust $I_{sp}$, and an approximation of the magnitude of velocity increment $\Delta v_{burn}$ [16].

\[ \Delta t_{burn} = \frac{m_0 I_{sp} g_0}{T} (1 - e^{-\frac{\Delta v_{burn}}{I_{sp} g_0}}), \quad (7.4) \]

where $g_0 \approx 9.81 \text{msec}^{-2}$ is the sea-level standard acceleration of gravity. Also, for a typical servicer satellite $T \approx 10 \text{kN}$, $I_{sp} \approx 300 \text{sec}$, and $\Delta v_{burn} \approx 1 \text{kmsec}^{-1}$ [16]. Then, the velocity increment will be a constant amount over course of $\Delta t_{burn} \approx 30 \text{sec}$, and is simply calculated as $\Delta v = \Delta t_{burn} u$. The nonlinear process model that is considered for the evolution of the satellite in $J_2$ perturbed orbits includes the solution to (7.1) in addition to the first order long periodic effects, i.e., (A.1)-(A.7).
most conveniently described in discrete time, a discrete-time formulation of the equations of motion is obtained. Considering a sampling time step $T_s$ and a sampling instant $j$, by applying the Euler’s approximation to the vehicle kinematics and dynamics, the following discrete-time model can be constructed from (7.1) and (7.5):

\[
\begin{align*}
a(j + 1) &= a(j) + T_s \left( \frac{2}{n\eta_2} \left( k(j) \sin L(j) - h(j) \cos L(j) \right) u_r + \frac{2W(j)}{n(j)\eta_2(j)} u_\theta \right) \\
h(j + 1) &= h(j) + T_s \left( \frac{3\mu J_2 R_e^2 k(j)}{4\eta_2^3(j) \eta_1(j) a^3(j)} (3\gamma^2(j) - 1 + 2\gamma(j)(p(j) \alpha(j) - q(j) \beta(j))) \\
&\quad - \eta_2(j) \cos \frac{L(j)}{n(j) a(j)} u_r \right) \cos L(j) u_\theta - \frac{\eta_2(j) k(j)}{n(j) a(j) W(j)} (p(j) \cos L(j) - q(j) \sin L(j)) u_h \\
k(j + 1) &= k(j) + T_s \left( -\frac{3\mu J_2 R_e^2 h(j)}{4\eta_2^3(j) \eta_1(j) a^3(j)} (3\gamma^2(j) - 1 + 2\gamma(j)(p(j) \alpha(j) - q(j) \beta(j))) \\
&\quad + \eta_2(j) \sin \frac{L(j)}{n(j) a(j)} u_r \right) \sin L(j) u_\theta + \frac{\eta_2 h}{n(j) a(j) W(j)} (p(j) \cos L(j) - q(j) \sin L(j)) u_h \\
p(j + 1) &= p(j) + T_s \left( -\frac{3\mu J_2 R_e^2 q(j) \eta_3(j) \beta(j) \gamma(j)}{4\eta_2^3(j) \eta_1(j) a^3(j)} + \eta_2(j) \eta_3(j) \sin \frac{L(j)}{2n(j) (j) a W(j)} u_h \right) \\
q(j + 1) &= q(j) + T_s \left( -\frac{3\mu J_2 R_e^2 q(j) \eta_3(j) \alpha(j) \gamma(j)}{4\eta_2^3(j) \eta_1(j) a^3(j)} + \eta_2(j) \eta_3(j) \cos \frac{L(j)}{2n(j) a(j) W(j)} u_h \right) \\
\lambda(j + 1) &= \lambda(j) + T_s \left( -\frac{3\mu J_2 R_e^2}{4\eta_2^3(j) \eta_1(j) a^3(j)} \left( (3\gamma(j)^2 - 1)(\eta_2(j) + 1) - 2\gamma(j)(p(j) \alpha(j) - q(j) \beta(j)) \right) \\
&\quad - \eta_2(j) \cos \frac{L(j)}{n(j) a(j) W(j)} \left( h(j) \sin L(j) + k(j) \cos L(j) \right) \right) + n(j) \left( -\frac{\eta_2(j) (1 + W(j))}{n(j) a(j) W(j)} \left( h(j) \cos L(j) - k(j) \sin L(j) \right) u_\theta \right) \\
&\quad - \frac{\eta_2(j)}{n(j) a(j) W(j)} (p(j) \cos L(j) - q(j) \sin L(j)) u_h \right) \right)
\end{align*}
\]

Equation (7.6) can then be rewritten in a more compact form and the first-order
long-periodic effect is added to form the discrete-time nonlinear process model:

\[
x(j + 1) = f(x(j), u(j)) \in \mathbb{R}^6, \\
\dot{x}(j + 1) = x(j + 1) - J_2 R_e^2 x^{(lp)}(j + 1).
\]  \(7.7\)

### 7.3 Objective Function

To find the objective function, we need the following set of information provided by the desired multi-impulse trajectory (hereinafter called the reference trajectory): (i) the number of impulses \(N\), (ii) the time of applying each impulse \((t_i, i = 1, \cdots, N)\), (iii) the prediction horizon \(T_i, i = 1, \cdots, N\), (iv) the \(\Delta v\) from (4.23) and the resulted acceleration \(u_d = \frac{\Delta v}{\Delta t_{\text{burn}}}\), and (v) the conversion of the reference trajectory to the Cartesian coordinates \((X_d(t))\).

According to (7.7), the prediction of the dynamics conducted at the \(j^{th}\) time instant for \(\tau\) time samples in the future is calculated as

\[
x(j + \tau + 1|j) = f(x(j + \tau|j), u(j + \tau|j)), \\
\dot{x}(j + \tau + 1|j) = x(j + \tau + 1|j) - J_2 R_e^2 x^{(lp)}(j + \tau + 1|j),
\]  \(7.8\)

where the notation \(j + \tau|j\) indicates the predicted value of the variable at the instant \(j + \tau\) calculated at time instant \(j\).

The proposed objective function for the NMPC includes a norm of error between the desired and predicted satellite's trajectory under \(J_2\) perturbation using the process model and a norm of control signals. To formulate the objective function, we use the Cartesian coordinates of the servicer and the target, due to the required numerical uniformity in the optimization problem. The equations that are needed to convert the EE to the classical orbital elements and then to the Cartesian coordinates are given in (3.7) and (3.12), respectively. We denote the actual trajectory of the servicer at time \(t\) in a perturbed orbit, represented in the Cartesian coordinate, by \(\dot{X}(t)\). By introducing the error vector \(\delta X = \dot{X} - X_d\) and the correction vector \(\delta u = u - u_d\), the objective function for the NMPC to minimize the deviation from the desired trajectory with minimum control effort at the time \(t_i\) and in the prediction horizon.
$T_{i+1}$ can be defined as

$$F_i(\delta X, \delta u) = \sum_{\tau=1}^{\bar{T}_{i+1}} \delta X^T(\bar{t}_i+\tau|\bar{t}_i)Q\delta X(\bar{t}_i+\tau|\bar{t}_i) + \sum_{\tau=1}^{\bar{T}_{i+1}-1} \delta u^T(\bar{t}_i+\tau|\bar{t}_i)R\delta u(\bar{t}_i+\tau|\bar{t}_i).$$  

(7.9)

Here, we have $\bar{t}_i = \frac{t_i}{T_s}$, $\bar{T}_{i+1} = \frac{T_{i+1}}{T_s}$ for $i = 1, \cdots, N-2$. Note that $\delta X(\bar{t}_i + \tau|\bar{t}_i)$ and $\delta u(\bar{t}_i + \tau|\bar{t}_i)$ are the prediction of the error vector $\delta X$ and the correction vector $\delta u$ based on the nonlinear process dynamics (7.8) and converted into the Cartesian coordinate. Also, $\mathbf{Q} \in \mathbb{R}^{6 \times 6}$ and $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ are symmetric positive definite matrices.

The discrete time horizon over which $F_i$ is minimized is $\tau = 1, \cdots, \bar{T}_{i+1}$. Note that the objective function $F_i$ does not include the first interval $T_1$ and the time instants between impulses, since there is no control signal to compensate the deviation from the desired trajectory at the times other than $t_i$. That makes the objective function to be calculated $N-2$ times at the sampling instants $\bar{t}_i$. Therefore, to guarantee following the reference trajectory with minimum control effort in the time intervals $T_{i+1}$ ($i = 1, \cdots, N-2$), the following fixed-final-time constrained optimization is solved at time $t_i$:

$$\delta u^* = \arg \min_{\delta u} F_i(\delta X, \delta u), \quad \text{for} \quad i = 1, \cdots, N-2$$

subject to: (i) (7.8) with the initial condition $\dot{\mathbf{X}}(\bar{t}_i|\bar{t}_i) = \dot{\mathbf{X}}(t_i)$, and

(ii) $\|u(j + \tau + 1|j)\| \leq u_{max}$,

(7.10)

where $u_{max}$ stands for the upper bound of the input vector. However, this optimization in $T_N$ does not promise arriving at the target at the final time of transfer, which is the main goal in the long-range rendezvous mission, considered in this thesis. Therefore, we add the (soft) constraint of zero distance between the servicer and the target at the final time to provide the chasing capability of the servicer. This constraint is augmented in the cost function with a large penalty constant absorbed into a weighting matrix. Moreover, since the resulting optimization problem may become over-constrained or difficult to solve while the final time is fixed, we consider the final time $t_f$ or equivalently $T_N$ among the design variables in the final horizon. Hence, the optimization problem in the final horizon becomes a free-final time constrained problem to ensure the chasing capability of the servicer. Denoting $\delta \mathbf{X}_T = \dot{\mathbf{X}} - \mathbf{X}_T$, where $\mathbf{X}_T$ is the target’s state trajectory in the Cartesian coordinate, we define the
objective function at the time $t_{N-1}$ in the final horizon $T_N$ as

$$
F_{N-1}(\delta X, \delta u) = \sum_{\tau=1}^{\tilde{T}_N} \delta X^T(\hat{t}_{N-1} + \tau|\hat{t}_{N-1})Q\delta X(\hat{t}_{N-1} + \tau|\hat{t}_{N-1}) + \\
\sum_{\tau=1}^{\tilde{T}_N-1} \delta u^T(\hat{t}_{N-1} + \tau|\hat{t}_{N-1})R\delta u(\hat{t}_{N-1} + \tau|\hat{t}_{N-1}) + \\
\delta X^T(\hat{t}_{N-1} + t_N|\hat{t}_{N-1})H\delta X(\hat{t}_{N-1} + t_N|\hat{t}_{N-1}),
$$

(7.11)

where $H \in \mathbb{R}^{6 \times 6}$ is a symmetric positive definite matrix. Note that $Q$, $H$, and $R$ are the weighting matrices to distinguish between the importance of following the desired path, penalizing the control, and catching the target satellite. The NMPC trajectory tracking algorithm minimizes the objective function subject to the dynamic constraints, path constraints, and terminal condition over the prediction horizon $\tau = 1, 2, ..., \tilde{T}_N$. Then, the following free-final-time constrained optimization in the last horizon must be solved at the time $t_{N-1}$:

$$
(\delta u^*, t_N^*) = \arg \min_{\delta u, t_N} F_{N-1}(\delta X, \delta u),
$$

subject to: (i) (7.8) with the initial condition $\dot{X}(\hat{t}_{N-1}|\hat{t}_{N-1}) = \dot{X}(t_{N-1})$, and

(ii) $\|u(j + \tau + 1|j)\| \leq u_{max}$.

(7.12)

The optimization problem given by (7.10) and (7.12) should be solved at $t_i$ ($i = 1, \cdots, N - 1$), thereby calculating the optimal $t_N$ and the constant optimal control signals over the course of $\Delta t_{\text{burn}}$. The last control signal $u_N$ is reserved to match the servicer’s and target’s velocities at the time of intersecting the target to complete the rendezvous mission. The overall NMPC algorithm and architecture are shown in Algorithm 4 and Figure 7.2, respectively. As can be seen from Figure 7.2, the state and control reference, i.e., $X_d$ and $u_d$, are obtained based on equations derived in Chapter 4. The closed-loop tracking guidance law is then achieved based on the designed NMPC. The NMPC optimization problem in (7.10) and (7.12) is solved using a genetic algorithm approach, which will be discussed in Section 7.4. The control input $u(t)$ is then calculated by combining the reference control $u_d$ and the correction control vector $\delta u^*(t)$.

To investigate the performance of the proposed controller and show its robustness against unmodelled dynamics, in this thesis we include a realistic dynamic simulation
for the servicer in the control loop. In addition to secular and 1st order long-periodic effects, the short-periodic effects and a bounded random uncertainty that is one order of magnitude smaller than the $J_2$ perturbations are simulated (see Figure 7.2). To feedback the actual servicer states, the last calculated states of the servicer in the plant’s dynamics in the current horizon is used as the initial states for the nonlinear process model of NMPC in (7.1) for the next horizon.

### 7.4 GA-Based Optimization

The performance of the NMPC mainly depends on the implemented optimization algorithm. To solve the optimization problem defined in (7.10) and (7.12), we implement a single-objective GA. This GA method is a guided random search algorithm with the capability of exploring the diverse regions of the solution space. The employed GA method is explained in the flowchart depicted in Figure 7.3. In the following, we elaborate the steps of the algorithm to find the optimal solutions in the proposed NMPC.
Algorithm 4: Nonlinear Model Predictive Control.

**Data:** Servicer’s initial location $x_0$, Reference trajectory and control $X_d$ and $u_d$, target’s trajectory $X_T$, $N$, $T_i$, $i = 1, \cdots, N-1$, and sampling time $T_s$.

**Result:** Real control signal $u$, Real state space $X$, and new transfer time $t_N$.

$t_N = \infty$, $j = 1$, $i = 2$.

$x(j) = x_0$.

while $j \leq t_N$ do
  if $j = \bar{t}$ then
    if $i \neq N-1$ then
      $\delta u(j) = \text{GA} \left( F \left( \delta X(j), \delta u_{\text{candidate}}(j) \right), T_i \right)$. $\delta X(j)$ is the feedback from the plant’s dynamics to the control’s dynamics.
      $i = i + 1$.
    else
      $[\delta u(j), t_N] = \text{GA} \left( F \left( \delta X(j), \delta u_{\text{candidate}}(j), t_{N_{\text{candidate}}} \right) \right)$. 
    end
  end
  else
    $\delta u(j) = 0$.
  end

$u(j) = \delta u(j) + u_d(j) \pm O(J_2/10)$.

$x(j + 1) = x(j) + T_s \left( A(x(j)) + I_{6 \times 6} B(x(j)) u^T(j) \right)$.

$\dot{x}(j + 1) = x(j + 1) - J_2 R_c^2 \left( x_{lp}(j + 1) + x_{sp}(j + 1) \right)$.

Calculating the plant’s dynamics.

$x(j) \rightarrow X(j)$.

$\delta X(j) = X(j) - X_d(j)$.

$j = j + 1$.

end
7.4.1 Initialization

The three design variables $\delta u_x, \delta u_y$ and $\delta u_z$, which are the elements of the vector $\delta u$ in the Cartesian coordinate, must be initialized for $Z$ number of population at the time $t_i, i \neq N - 1$. The correction vector $\delta u$ is selected to be a small random vector in the initialization phase, and it is added to the desired control input $u_d(t_i)$. For the last horizon, the time of catching the target specified $t_N$ is another design variable beside $\delta u$, to minimize the distance between the two satellites. For initializing this variable, we add a small random number to the transfer time we already have from the reference trajectory.

7.4.2 Objective Function Evaluation

The next step is to evaluate the objective function for the initialized design variables. The objective function in the horizons $T_i, i \neq N$, is derived by (7.9), which measures the deviation of the servicer from the reference trajectory and the total control effort. Beside that, the objective function in the last interval also measures the distance of the servicer from the target at the final time $t_f$ (which is among the initialized design

Figure 7.3: Single-objective genetic optimization architecture
variables). This performance criterion is prioritized through multiplying it by a large weighting matrix in (7.11).

7.4.3 Binary Tournament Selection

In any step of the optimization that we need to select parents, we use binary tournament selection method. In this method, two individuals are randomly chosen from the population and the one with the lower objective function is kept in the process. To select other parents, we repeat the same procedure.

7.4.4 Arithmetic Crossover

Crossover is an inheritance operator, that causes the children to have attributes from all parents. Initially, a crossover rate (between 0 and 1) is fixed that is the indicator of the execution probability of this operator. During the optimization, each time that the crossover operator is called, a random number between 0 and 1 is generated. If this number is smaller than the crossover rate, then this operator is executed. During crossover, a random vector $\beta$ (4-dimensional for the last horizon and 3-dimensional for other horizons) whose elements are between 0 and 1 is generated, based on which we define the children $y^c_1, y^c_2$ of the two selected parents by the binary tournament selection, $y^p_1, y^p_2$.

$$
y^c_1 = \beta \cdot y^p_1 + (1 - \beta) \cdot y^p_2, \quad \& \quad y^c_2 = \beta \cdot y^p_2 + (1 - \beta) \cdot y^p_1,
$$

where $y$ is a vector of all design variables, the operator $\ast$ refers to the element-wise multiplication of two vectors, and the vector $1$ is the 3- or 4-dimensional vector whose elements are all equal to 1.

7.4.5 Mutation

In the iteration $1 \leq d \leq D$, where $D$ is the maximum number of iterations, a number of individuals are randomly selected based on the binary tournament selection to be mutated with a probability, to explore the search region. A mutation probability distribution is fixed at the initialization phase of the GA. This distribution allows the mutation operator to execute with a higher probability at the beginning of the
optimization, which leads to avoiding convergence to local optima. In this thesis, the mutation probability PM in the iteration $d$ is defined by (see Figure 7.4)

$$PM(d) = PM_0(1 + \cos \frac{\pi d}{D}),$$

where $PM_0$ is a parameter, fixed at the beginning of the optimization loop. Each time that the mutation operator is called for an individual, 3 random numbers (or 4 random numbers for the last horizon) between 0 and 1 are generated. If any of these numbers is smaller than $PM(d)$, then the corresponding design variable in the individual is randomly reassigned within its bounds.

### 7.4.6 Replacement

After applying the crossover and mutation operators on the current population, $Z'$ children are generated and added to the current population. After evaluating the objective function of this new $Z'$ individuals, they should be replaced with $Z'$ number of the members of the old population to have a fixed number of individuals at each iteration. The replacement operation is similar to the binary selection except that the higher objective function is chosen and replaced with the better offspring.

### 7.4.7 Termination Criteria and Convergence

In an optimization, several termination criteria can be considered: (i) maximum number of iterations, (ii) maximum number of iterations with no improvement, (iii)
maximum allowed CPU time, and (iv) reaching an admissible fitness. Further, the convergence of a single objective heuristic algorithm is proved by a decreasing behaviour of the evaluated objective function over the course of iterations (for a minimization problem). In this thesis, minimizing the objective function is numerically studied to ensure the convergence of the optimization in a case study. Further, the maximum number of iterations is considered as a termination criterion.

7.5 Conclusion

In this chapter, a robust NMPC was developed and applied to solve the online optimal tracking in a long-range rendezvous problem in $J_2$ perturbed environments. The reference trajectory was a multi-impulse smooth transfer trajectory designed in the two-body context that we proposed in Chapter 4. The control signals were non-zero constant accelerations over the burning time $\Delta t_{\text{burn}}$, since the impulsive control signal damages the satellites’ structure. For the real-time servicer dynamics, secular, 1st order long-periodic, and 1st order short-periodic effects induced by $J_2$ perturbation were considered. Further, to compensate for the error of neglecting other types of perturbations, a bounded random acceleration with one order of magnitude smaller effects comparing to the $J_2$ perturbation was added to the control signal. The optimization method applied in this chapter was based on the GA that was designed to find the impulsive controllers and the final time in the final horizon. The objective functions of the unconstrained fixed-final-time optimization problems defined before the final horizon were the measure of deviation from the reference trajectory and the desired control input. To satisfy the chasing requirement, the optimization in the final horizon was a free-final-time problem with a terminal constraint.
Chapter 8

Simulation Results

8.1 Summary

In this chapter, we aim to evaluate the efficacy of the proposed nonlinear control strategy presented in Chapter 7 using two different 5-impulse smooth reference trajectories. The optimal control signals and the mission transfer time are obtained using the genetic algorithm, whose convergence is studied. We simulate the closed-loop control system including the perturbed trajectories using MATLAB.

8.2 Case Study 1

In this case study, the initial and final co-planar orbits are assumed to be in the plane specified by the inclination of $\iota = 5$ deg and RAAN of $\Omega = 10$ deg. These orbits are defined based on the planar orbital elements as:

- $a_S = 7000 \, \text{km}$
- $e_S = 0.05$
- $\omega_S = 10 \, \text{deg}$
- $\theta_S(0) = 270 \, \text{deg}$

- $a_T = 7500 \, \text{km}$
- $e_T = 0$
- $\theta_T(0) = 0 \, \text{deg}$

Table 8.1 summarizes the co-planar 5-impulse smooth reference trajectory considered in this study to catch the target from the servicer’s parking orbit. The trajectory is described based on the (planar) classical orbital elements of the intermediate orbits, the time between each impulses $T_i$, and the magnitude of the velocity increment at the time of each impulse $\Delta v_i$. Figure 8.2(a) depicts the reference trajectory in the plane of the orbits. The cyan star and the small red square show the initial locations of the servicer and target, and the black circles are the impulse locations. In the reference
trajectory, the servicer catches the target at the final time \( t_f = 6391 \) sec. In the

<table>
<thead>
<tr>
<th>Orbit number ( i )</th>
<th>( a ) [km] (( \times 10^3))</th>
<th>( e )</th>
<th>( \omega ) [deg]</th>
<th>( T_i ) [sec]</th>
<th>( \Delta v_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>0.05</td>
<td>10</td>
<td>89.7</td>
<td>0.213</td>
</tr>
<tr>
<td>2</td>
<td>6.64</td>
<td>0.06</td>
<td>-46.47</td>
<td>1885</td>
<td>0.255</td>
</tr>
<tr>
<td>3</td>
<td>6.41</td>
<td>0.026</td>
<td>-37.41</td>
<td>2809</td>
<td>0.1458</td>
</tr>
<tr>
<td>4</td>
<td>7.93</td>
<td>0.174</td>
<td>108.13</td>
<td>1156</td>
<td>0.812</td>
</tr>
<tr>
<td>5</td>
<td>6.39</td>
<td>0.173</td>
<td>-175.8</td>
<td>451.3</td>
<td>0.66</td>
</tr>
<tr>
<td>6</td>
<td>7.5</td>
<td>0</td>
<td>60</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

NMPC, the weighting matrices appearing in \( F_i \), for \( i = 1, \cdots, 3 \), corresponding to the trajectory error and control correction are designed to be \( Q = I_{6 \times 6} \) and \( R = 10I_{3 \times 3} \), respectively. For the last horizon, we design these matrices to be \( Q = 0.01I_{6 \times 6} \) and \( R = 0.1I_{3 \times 3} \), and the weighting matrix corresponding to the final distance from the target is selected to be \( H = 10I_{6 \times 6} \). These parameters are obtained based on trial and error according to an investigation conducted on the quality of the produced solutions. Note that the weighting/penalty matrix \( H \) is designed orders of magnitude larger than the other two matrices to emphasise the importance of catching the target in the last interval. The time between each impulse \( T_i \) is the prediction horizon for the NMPC. The time of impulse (control action) \( t_i, i = 1, \cdots, 4 \), can be obtained from \( T_i \) in Table 8.1, and for \( i = 5 \) the optimal time of the last impulse is calculated as part of the optimization of \( F_4 \). Moreover, the sampling time \( T_s \), which is used to calculate the time instants \( \bar{t}_i \), is set to be \( T_s = 0.1 \) sec.

The parameters of the genetic algorithm used in the NMPC architecture are designed as explained in the following. The typical value of the crossover rate used in the literature is \([0.8, 0.95]\) [42], and for constant mutation functions the rate is selected to be reciprocally proportional to the number of design variables, which is almost 0.3 in our case study. Hence, the crossover rate and \( PM_0 \) are chosen 0.8 and 0.3, respectively. Based on an investigation conducted on the optimization time and the quality of the produced solutions, the number of population and the maximum number of iterations are obtained to be \( Z = 20 \) and \( D = 100 \) for the GA before the
last horizon, and $Z = 100$, $D = 200$ for the last horizon. Further, another termination criterion for the optimizations is to have a constant objective value (within a defined tolerance) for 50 consecutive iterations. Figure 8.2(b) shows the obtained control signals in the Cartesian coordinates, after solving (7.10) and (7.12) using the GA. As one can observe, the control signal is a constant acceleration over the burning time $\Delta t_{\text{burn}} = 30$ sec. In addition, in the last horizon we also find the optimal $t_N$ to be $7266.4$ sec, which indicates that the actual trajectory must be almost $14.6$ min longer than the reference trajectory to catch the target.

The trajectory of the servicer is simulated under the secular and first order long-periodic and short-periodic effects induced by $J_2$, and bounded random accelerations included to capture other types of perturbations with one order of magnitude smaller effects comparing to the $J_2$ perturbation. The reference trajectory along with the actual trajectory of the servicer in three-dimension are shown in Figure 8.2(c), and their projections onto the $x$-$y$-plane are observed in Figure 8.2(d).

<table>
<thead>
<tr>
<th>Orbit number $i$</th>
<th>$\vec{a}$ [km] ($\times 10^3$)</th>
<th>$\vec{e}$</th>
<th>$\vec{\omega}$ [deg]</th>
<th>$T_i$ [sec]</th>
<th>$\Delta v_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>0.05</td>
<td>10</td>
<td>89.7</td>
<td>0.216</td>
</tr>
<tr>
<td>2</td>
<td>6.68</td>
<td>0.051</td>
<td>67.03</td>
<td>1885</td>
<td>0.300</td>
</tr>
<tr>
<td>3</td>
<td>6.45</td>
<td>0.029</td>
<td>-2.38</td>
<td>2809</td>
<td>0.530</td>
</tr>
<tr>
<td>4</td>
<td>6.68</td>
<td>0.0451</td>
<td>186.93</td>
<td>1156</td>
<td>0.661</td>
</tr>
<tr>
<td>5</td>
<td>7.82</td>
<td>0.121</td>
<td>-31.52</td>
<td>1326.6</td>
<td>0.852</td>
</tr>
<tr>
<td>6</td>
<td>7.5</td>
<td>0</td>
<td>60</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The difference between the velocity increment vectors of the perturbed trajectory and the reference are shown in Figure 8.2(e). The magnitudes of the velocity increment vectors at each impulse for the simulated trajectory are also shown in Table 8.2. Comparing the total velocity increment (control effort) for these two trajectories, the total velocity increment of the perturbed trajectory is slightly greater than that of the reference trajectory for 0.468 km/sec. This increment in the control effort is predictable since the controller has to compensate for the perturbing effects. As can be interpreted from Figures 8.2(c)-8.2(d), the servicer follows the trajectory before
Figure 8.1: Case Study 1; (a) Reference trajectory (b) Control signal (c) Projected 2-D transfer trajectory (d) 3-D transfer trajectory (e) Velocity increment differences from the reference (ordered sequentially in $x$, $y$, $z$ direction from top to bottom)
Figure 8.2: Case Study 1; (a) Variation of $F_1$ (b) Variation of $F_2$ (c) Variation of $F_3$ (d) Variation of $F_4$ (e) Distance from the target
CHAPTER 8. SIMULATION RESULTS

the 4\textsuperscript{th} impulse. However, in the last horizon after the 4\textsuperscript{th} impulse, servicer deviates from the reference. This is because of prioritizing catching the target to following the reference trajectory by giving more weight to the termination criterion in the last horizon. Due to the randomness of the added perturbation in the simulation, we computed the actual trajectory under the obtained optimal control input multiple times. The average distance between the two satellites at the final time of mission is calculated to be almost 4 km, with this distance changing in the range [3, 5] km. Table 8.2 summarizes the averaged classical orbital elements in each intermediate orbit to compare with those corresponding to the reference trajectory. The classical orbital elements are averaged since the $J_2$ effects (especially the short-periodic effect) change the classical orbital elements over time.

Although the stability of the NMPC is guaranteed when there is a terminal zero-state constraint [64], it is worth studying the performance of the optimizer at each prediction horizon. The convergence of single-objective evolutionary algorithms is shown using a decreasing behaviour of the objective function versus the number of objective function evaluations. In this regard, Figures 8.2(a)-8.2(d) show the convergence of the GA when optimizing $F_i$, $i = 1, \ldots, 4$. The variation of distance from the target with respect to the number of objective function evaluations in the last interval is also given in Figure 8.2(e). This figure indicates that the algorithm can converge to an acceptable distance from the target despite starting from an initial guess that resulted in a distance in the order of thousands of kilometers.

8.3 Case Study 2

In this case study, the initial and final orbital elements are similar to those of the previous case study. The 5-impulse smooth reference trajectory considered in the second case study is described in Table 8.3, and Figure 8.3(a) depicts the reference trajectory in the plane of orbits. The total transfer time for the reference trajectory to catch the target is $t_f = 12558$ sec, with a total control effort of $\Delta v = 1.4673$ km/sec. The parameters of the NMPC and the implemented genetic algorithm for this case study (including the weighting matrices, sampling time, crossover rate, mutation rate, etc.) are similar with that of Case Study 1.

The control signals obtained after solving the optimization problems (7.10)-(7.12)
are given in Figure 8.3(b). As can be observed, control signals are constant accelerations over the burning time $\Delta t_{\text{burn}} = 30$ sec. In addition, the time of the last impulse that is obtained from the last optimizer is $t_N = 12799$ sec, which is only 4 min longer than the time of reference trajectory to catch the target. Similar to the previous case study, the trajectory of the servicer is simulated under the $J_2$ perturbation and a bounded random acceleration. The reference trajectory along with the simulated actual trajectory of the servicer in three-dimension are shown in Figure 8.3(c), and their projections onto the $x$-$y$-plane are observed in Figure 8.3(d).

<table>
<thead>
<tr>
<th>Orbit number $i$</th>
<th>$a$ [km] ($\times 10^3$)</th>
<th>$e$</th>
<th>$\omega$ [deg]</th>
<th>$T_i$ [sec]</th>
<th>$\Delta v_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.00</td>
<td>0.050</td>
<td>10.00</td>
<td>1476.7</td>
<td>0.278</td>
</tr>
<tr>
<td>2</td>
<td>6.58</td>
<td>0.020</td>
<td>123.08</td>
<td>138</td>
<td>0.275</td>
</tr>
<tr>
<td>3</td>
<td>7.00</td>
<td>0.047</td>
<td>$-1.11$</td>
<td>4903.5</td>
<td>0.178</td>
</tr>
<tr>
<td>4</td>
<td>6.71</td>
<td>0.034</td>
<td>$-61.92$</td>
<td>3538.3</td>
<td>0.712</td>
</tr>
<tr>
<td>5</td>
<td>7.15</td>
<td>0.048</td>
<td>$-161.50$</td>
<td>2501.8</td>
<td>0.024</td>
</tr>
<tr>
<td>6</td>
<td>7.5</td>
<td>0</td>
<td>60</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Orbit number $i$</th>
<th>$\bar{a}$ [km] ($\times 10^3$)</th>
<th>$\bar{e}$</th>
<th>$\bar{\omega}$ [deg]</th>
<th>$\bar{T}_i$ [sec]</th>
<th>$\bar{\Delta v}_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.20</td>
<td>0.050</td>
<td>9.98</td>
<td>1476.7</td>
<td>0.124</td>
</tr>
<tr>
<td>2</td>
<td>7.00</td>
<td>0.080</td>
<td>8.92</td>
<td>138</td>
<td>0.159</td>
</tr>
<tr>
<td>3</td>
<td>7.00</td>
<td>0.032</td>
<td>10.05</td>
<td>4903.5</td>
<td>1.7835</td>
</tr>
<tr>
<td>4</td>
<td>6.68</td>
<td>0.042</td>
<td>64.07</td>
<td>3538.3</td>
<td>0.866</td>
</tr>
<tr>
<td>5</td>
<td>6.98</td>
<td>0.135</td>
<td>124.23</td>
<td>2742.2</td>
<td>0.021</td>
</tr>
<tr>
<td>6</td>
<td>7.5</td>
<td>0</td>
<td>60</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Figure 8.3(e) depicts the differences between the velocity increment vectors of the reference and the perturbed trajectory. The magnitudes of the velocity increment vectors at each impulse for the simulated trajectory are also shown in Table 8.4.
Figure 8.3: Case Study 2; (a) Reference trajectory (b) Control signal (c) Projected 2-D transfer trajectory (d) 3-D transfer trajectory (e) Velocity increment differences from the reference (ordered sequentially in $x$, $y$, $z$ direction from top to bottom)
Figure 8.4: Case Study 2; (a) Variation of $F_1$ (b) Variation of $F_2$ (c) Variation of $F_3$ (d) Variation of $F_4$ (e) Distance from the target
Comparing the total velocity increment (control effort) for these two trajectories, the total velocity increment of the perturbed trajectory is 2 times greater than that of the reference trajectory. Again, this increment was predictable as the thruster should be able to compensate for the perturbing effects. As can be interpreted from Figures 8.3(c)-8.3(d), the servicer follows the trajectory before the 4\textsuperscript{th} impulse. The deviation of the perturbed trajectory from the reference is due to prioritizing catching the target to following the reference trajectory in the last horizon. Due to the randomness of the added perturbation in the simulation, we computed the actual trajectory under the obtained optimal control input multiple times. The average distance between the two satellites at the final time of mission is calculated to be almost 7 km, with this distance changing in the range [5, 9] km. The averaged classical orbital elements in each intermediate orbit for the perturbed trajectory are given in Table 8.4. Finally, the convergence of the implemented optimizer is studied in Figures 8.3(a)-8.3(d). Moreover, 8.3(e) shows the decreasing behaviour of the distance between the servicer and the target over a course of function evaluations.

8.4 Conclusion

The proposed NMPC in Chapter 7 was simulation in two case studies with different 5-impulse reference trajectories. The results indicated that the proposed controller was able to follow the reference and catch the target with a final distance of less than 10 km. This is an acceptable final distance for a long-range rendezvous, since the main goal of such missions is to just arrive at the target’s orbit. The transfer time and the control effort of the optimally controlled trajectory for the first case study were slightly greater (15 min and 0.468 km/sec, respectively) comparing to those of the reference trajectory. For the second case study, the control effort of the controlled trajectory was two times greater than that of the reference. However, the transfer time of the controlled trajectory was only 4 min longer than the time that the reference trajectory would catch the target. Finally, the convergence analysis for both case studies, shown in Figure 8.2 and Figure 8.4, indicated that the GA was well-suited for the proposed NMPC. Therefore, it is effective to use the developed guidance algorithm for impulsive control signals to achieve online optimal trajectory tracking in long-range rendezvous missions.
Chapter 9

Conclusion and Future Directions

In this thesis, we developed a multi-impulse shape-based trajectory optimization methodology for the long-range rendezvous phase of a servicer that chases a target in an OOS mission. In this process, three main problems were tackled. First, a family of $N$-impulse smooth transfer trajectories were designed to connect any two co-planar co-focal spatial points in a central-gravity. Then, the $N$-impulse smooth trajectory generator was revised to include trajectories that intercept a target satellite in its orbit, where the initial impulse location was not considered fixed anymore. Finally, we proposed a multi-objective constrained optimization to identify the set of Pareto Optimal trajectories among the smooth trajectories derived in the second phase. Transfer time and control effort were considered as the two Pareto cost functions. The Pareto frontier set of solutions gives the designer the ability to choose a trajectory based on the mission priorities. For example, if there is an urgent need for servicing, the servicer chooses a trajectory that has less transfer time with some compromises in fuel consumption. Beside the constraints on the shape of the transfer trajectories and the two boundary conditions, restrictions on the maximum permitted amount of $\Delta v_i$ and the orbital elements corresponding to the intermediate orbits were included. The latter restriction reduces the risk of collision in populated orbital regions and ensures that the servicer remains within the LEO regime. The design variables considered in the optimization process were the orbital elements of the intermediate orbits, and the number and the location of the impulses. A non-dominated sorting genetic algorithm, NSGA-II, was implemented to solve the constrained multi-objective optimization problem. To study the convergence of the proposed algorithm in our problem, IGD$^+$ was used. The proposed trajectory design methodology has less design variables compared to other approaches using models with approximately
the same level of complexity, e.g., the continuous thrust models. To demonstrate the efficacy of the proposed methodology, we compared the obtained optimal solutions with those generated by the MOGA and a developed optimal Lambert approach for chasing a target, in three case studies. Overall, the multi-impulse smooth trajectory design methodology provided improved solutions in terms of both transfer time and control effort.

In the second phase of this thesis, a real-time robust NMPC was developed and applied to solve the online optimal tracking in a long-range rendezvous problem in $J_2$-perturbed environments. The reference trajectory was the multi-impulse smooth transfer trajectory designed in the two-body context that we proposed in the first phase of this thesis. The control signals were non-zero constant accelerations over the burning time $\Delta t_{\text{burn}}$, since the impulsive control signal damages the satellites’ structure. For the real-time servicer dynamics, secular, 1st order long-periodic, and 1st order short-periodic effects induced by $J_2$ perturbation were considered. Further, to compensate for the error of neglecting other types of perturbations, a bounded random acceleration with one order of magnitude smaller effects comparing to the $J_2$ perturbation was added to the control signal. The optimization method applied in this thesis was based on the GA that was designed to find the impulsive controllers and the final time in the final horizon. The objective functions of the unconstrained fixed-final-time optimization problems defined before the final horizon were the measure of deviation from the reference trajectory and the desired control input. To satisfy the chasing requirement, the optimization in the final horizon was defined to be a free-final-time problem with a terminal constraint. The efficiency of the proposed NMPC was then proved in two case studies. The results demonstrated that the proposed NMPC was able to both follow the reference trajectory and arrive at the target in an acceptable range.

A possible future direction of this research is developing an optimization framework for service scheduling in OOS missions with multiple servicers and multiple targets. The optimization framework should be able to assign a number of servicers to a number of targets based on a priority function such that the overall transfer time and control effort of the servicing mission remain optimal. The trajectories used for transferring these multiple servicers to multiple targets are the Pareto Optimal multi-impulse smooth transfer trajectories that we designed in this thesis. These trajectories must be accurately followed under the $J_2$ perturbation. In addition to the $J_2$
perturbation, other types of environmental disturbances, e.g., atmospheric drag and solar radiation, can be considered in the satellite’s dynamics to have a more realistic approximation of the satellite motion in space. Another future direction of this thesis is to consider perturbed motion for non-cooperative targets. Moreover, one can implement other control schemes and compare their results with the proposed NMPC in this thesis.
Bibliography


Appendix A

The First Order Long- and Short-Periodic Effects

\[ a^{(lp)} = 0, \]  
\[ \lambda^{(lp)} = \left( \frac{\zeta_1 \zeta_2}{8a^2 \eta^4 (1 + \eta) \sigma_2^2 (\sigma_3^2 - \sigma_1^2)^2} \right) \{ 4\eta^2 (\sigma_3^2 - \sigma_1^2) \Theta + (1 + \eta) \Pi \}, \]  
\[ L^{(lp)} = \lambda^{(lp)} - \left( \frac{\Theta}{4a^2 \eta^4 (1 + \eta) \sigma_2^2} \right) \times \{ \zeta_1 \zeta_2 [3(1 + \eta) + 2n^2] + (1 + \eta)[2(\tau_1 \tau_2 + \tau_2 \tau_1) + \epsilon_1^2 (\tau_1 \tau_2)] \}, \]  
\[ h^{(lp)} = - \left( \frac{1}{8a^2 \eta^4 \sigma_2^2 (\sigma_3^2 - \sigma_1^2)^2} \right) \times \{ 2\eta^2(q \zeta_2 - p \zeta_1)(\sigma_3^2 - \sigma_1^2)^2 \Theta + k \zeta_1 \zeta_2 \Pi \}, \]  
\[ k^{(lp)} = - \left( \frac{1}{8a^2 \eta^4 \sigma_2^2 (\sigma_3^2 - \sigma_1^2)^2} \right) \times \{ 2\eta^2(q \zeta_1 + p \zeta_2)(\sigma_3^2 - \sigma_1^2)^2 \Theta + h \zeta_1 \zeta_2 \Pi \}, \]  
\[ p^{(lp)} = - \left( \frac{\sigma_3}{16a^2 \eta^4 \sigma_2^2 (\sigma_3^2 - \sigma_1^2)^2} \right) \times \{ 5q \sigma_3^2 \zeta_1 \zeta_2 + (k \zeta_2 + h \zeta_1)(\sigma_3^2 - \sigma_1^2)^2 \Theta \}, \]  
\[ q^{(lp)} = - \left( \frac{\sigma_3}{16a^2 \eta^4 \sigma_2^2 (\sigma_3^2 - \sigma_1^2)^2} \right) \times \{ 5p \sigma_3^2 \zeta_1 \zeta_2 - (k \zeta_1 - h \zeta_2)(\sigma_3^2 - \sigma_1^2)^2 \Theta \}, \]
APPENDIX A. THE FIRST ORDER LONG- AND SHORT-PERIODIC EFFECTS

\[ a^{(sp)} = - \left( \frac{(\sigma_3^2 - 2\sigma_1^2)}{a\eta^2\sigma_2^2} \right) \left[ (1 + \epsilon_2)^3 - \eta^3 \right], \quad (A.8) \]

\[ \lambda^{(sp)} = - \left( \frac{3(3\sigma_3^2 - 2\sigma_1^2)}{2a^2\eta^4(1 + \eta)\sigma_2^2} \right) \left[ (1 + \epsilon_2)(2 + \epsilon_2) + \eta^2 \right] \quad (A.9) \]

\[ L^{(sp)} = \lambda^{(sp)} + \left( \frac{3(3\sigma_3^2 - 2\sigma_1^2)}{2a^2\eta^4(1 + \eta)\sigma_2^2} \right) \left[ (1 + \epsilon_2)^2 + \eta(1 + \eta) \right], \quad (A.10) \]

\[ h^{(sp)} = - \left( \frac{h(\sigma_3^2 - 2\sigma_1^2)}{2a^2\eta^2(1 + \eta)\sigma_2^2} \right) - \left( \frac{3(3\sigma_3^2 - 2\sigma_1^2)}{2a^2\eta^4\sigma_2^2} \right) \times \left\{ \left[ (L - \lambda) + \epsilon_3 \right] \right\}, \quad (A.11) \]

\[ k^{(sp)} = - \left( \frac{k(\sigma_3^2 - 2\sigma_1^2)}{2a^2\eta^2(1 + \eta)\sigma_2^2} \right) - \left( \frac{3k(3\sigma_3^2 - 2)}{2a^2\eta^4\sigma_2^2} \right) \times \left\{ k(1 + \epsilon_2) + \left[ \eta^2 + (1 + \epsilon_2)(2 + \epsilon_2) \right] \cos L \right\} \]

\[ p^{(sp)} = \left( \frac{3q\sigma_3}{2a^2\eta^4\sigma_2} \right) \left[ (L - \lambda) + \epsilon_3 \right], \quad (A.13) \]

\[ q^{(sp)} = - \left( \frac{3p\sigma_3}{2a^2\eta^4\sigma_2} \right) \left[ (L - \lambda) + \epsilon_3 \right]. \quad (A.14) \]

Here,

\[ \Theta = 1 + \frac{5\sigma_3^2}{2(\sigma_3^2 - \sigma_1^2)} \quad (A.15) \]

\[ \Pi = 28 - 150\sigma_1^2 + 290\sigma_1^4 - 215\sigma_1^6 + 60\sigma_1^8 - 7\sigma_1^{10} \]

\[ \eta = \sqrt{(1 - \kappa^2 - \kappa^2)}, \quad \epsilon_1 = \sqrt{k^2 + \lambda^2}, \quad \epsilon_2 = k \cos L - \lambda \sin L, \quad \epsilon_3 = k \sin L - \lambda \cos L, \]

\[ \sigma_1 = \sqrt{q^2 + p^2}, \quad \sigma_2 = 1 + \sigma_1, \quad \sigma_3 = 1 - \sigma_1, \]

\[ \tau_1 = q \cos L + p \sin L, \quad \tau_2 = q \sin L - p \cos L \]

\[ \zeta_1 = qk + ph, \quad \zeta_2 = qh - pk. \]