DESIGN OF GUIDANCE SYSTEM FOR AN AIR-LAUNCH ROCKET

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

BY

TAHİR YANIK

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

SEPTEMBER 2023

Approval of the thesis:

DESIGN OF GUIDANCE SYSTEM FOR AN AIR-LAUNCH ROCKET

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

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

Prof. Dr. Serkan Özgen
Head of Department, **Aerospace Engineering** _____

Assoc. Prof. Dr. Halil Ersin Söken
Supervisor, **Aerospace Engineering, METU** _____

Examining Committee Members:

Assist. Prof. Dr. Ali Türker Kutay
Aerospace Engineering, METU _____

Assoc. Prof. Dr. Halil Ersin Söken
Aerospace Engineering, METU _____

Prof. Dr. Klaus Werner Schmidt
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Ramazan Yeniçeri
Aeronautical Engineering, Istanbul Technical University _____

Assist. Prof. Dr. Yakup Özkazanç
Electrical and Electronics Engineering, Hacettepe University _____

Date: 06.09.2023

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

Name, Surname: Tahir Yanık

Signature :

ABSTRACT

DESIGN OF GUIDANCE SYSTEM FOR AN AIR-LAUNCH ROCKET

Yanık, Tahir
M.S., Department of Aerospace Engineering
Supervisor: Assoc. Prof. Dr. Halil Ersin Söken

September 2023, 83 pages

Small satellites are of great interest nowadays due to their increased capability and low cost. However, their launch into the desired orbit is usually performed in shared flights. A promising method to launch them is to use dedicated rockets of smaller sizes that are launched in the air. This study deals with alternative guidance system designs that use Explicit Guidance (E Guidance) and Iterative Guidance Mode (IGM) algorithms to steer the rocket launched in the air. The optimal trajectory for the launch vehicle is obtained using numerical methods, and the outputs of the closed-loop algorithms are compared to each other, and the optimal solution to investigate their performance. A hypothetical rocket is modeled, and the guidance system is tested for different payloads and release conditions to examine its robustness.

Keywords: airlaunch rocket, ascent guidance, satellite launch vehicle, small satellite

ÖZ

HAVADAN FIRLATILAN BİR FIRLATMA ARACININ GÜDÜM SİSTEMİ TASARIMI

Yanık, Tahir
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Halil Ersin Söken

Eylül 2023 , 83 sayfa

Günümüzde küçük uyduların kullanımı maliyet ve yeteneklerindeki gelişmelerden dolayı artmaktadır. Ancak, bu uyduların arzu edilen yörüngeye fırlatılmaları genellikle ortak, paylaşılan uçuşlarla gerçekleştirilmektedir. Ana faydalı yük olarak küçük uyduları fırlatacak olan ve taşıyıcı bir uçaktan havadan fırlatılan roketler bu sorunu çözmek için incelenmektedir. Bu çalışmada havadan fırlatılan bir roketi gütmek için farklı yöntemlerin de incelendiği bir güdüm algoritması tasarlanmıştır. Bu yöntemlerin birbiriyle ve nümerik yöntemlerle elde edilen ideal sonuç ile karşılaştırılması yapılmıştır. Tasarlanan güdüm algoritmasının gürbüzlüğünü test etmek adına farazi bir fırlatma aracı modellenmiş, farklı faydalı yükler ve farklı ayrılma koşullarında güdüm sistemi test edilmiştir.

Anahtar Kelimeler: Havadan fırlatılan roket, tırmanma güdümü, uydu fırlatma aracı, küçük uydu

To my family

ACKNOWLEDGMENTS

First and foremost, I want to express my deepest gratitude to my supervisor, Assoc. Prof. Dr. Halil Ersin Söken, for his interest, endless support, and encouragement throughout my graduate study. If not for his understanding throughout this process it would not have been possible to complete this thesis.

I would like to express my deep appreciation to the members of my thesis committee for their insightful feedback, and constructive criticism.

I would like to extend my sincere gratitude to my current employer Roketsan for their unwavering support and flexibility, which has allowed me to pursue and complete this degree.

I would like to extend my heartfelt gratitude to Ahmed Sabri Yağmur for our fruitful conversations. His insights and guidance have significantly enriched my understanding, and I am truly thankful for his contributions. My appreciation goes to Gamze Mert for her invaluable help in mechanical drawing and her constant support throughout the journey. I want to express my sincere thanks to my colleagues Fatih Çalış, Atakan Süslü, and Sena Güler for their consistent support throughout this study. The collaborative spirit, encouragement, and dedication of my friends have been instrumental in the successful completion of this project, and I truly appreciate their contributions.

I would like to express my endless gratitude to my parents, Müzeyyen and Muhittin, and my sister, Zühre Nur, for their unwavering support, love, and encouragement throughout the journey of completing this thesis. Their constant belief in me and their guidance have been a source of inspiration and strength, without which this work would not have been possible. I would also like to extend my appreciation to my newborn niece, Elif Ebrar, whose laughter and infectious joy brought a sense of fun and lightness to the process of writing this thesis. I sincerely thank them for making this journey memorable and meaningful.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

DCM	Direction Cosine Matrix
DOF	Degree of Freedom
ECI	Earth Centered Inertial
ECEF	Earth Centered Earth Fixed
FPA	Flight Path Angle
GEO	Geostationary Earth Orbit
GNC	Guidance Navigation and Control
IGM	Iterative Guidance Mode
LEO	Low Earth Orbit
LEOP	Launch and Early Orbit Phase
LLA	Latitude, Longitude, Altitude
MDO	Multidisciplinary Design Optimization
MEO	Medium Earth Orbit
MPC	Model Predictive Control
NASA	National Aeronautics and Space Administration
NLP	Nonlinear Programming
OCP	Optimal Control Problem
PEG	Powered Explicit Guidance
QP	Quadratic Programming
RCS	Reaction Control System
SLS	Space Launch System
SQP	Sequential Quadratic Programming
TVC	Thrust Vector Control
TPBVP	Two Point Boundary Value Problem

TPS	Thermal Protection System
UPFG	Unified Powered Flight Guidance

CHAPTER 1

INTRODUCTION

There is a huge increase in the use of small satellites. Thanks to the improvements in material science and electronics, they can be built in smaller sizes and weigh less. The areas where the small satellites are used range from reconnaissance, remote sensing, disaster monitoring to communication.

Nowadays it is of great interest to launch small satellites with proper response time instead of launching a bigger satellite that lasts long enough to fulfill the same mission [1]. Small satellites can be used either in a single satellite mission or in a constellation. Many small satellites were used independently in the past, however nowadays there is a trend to use them as a group in space that will perform missions such as Earth observation, disaster forecasting, navigation, and communication. [1].

Satellites are usually classified by their size, application, or their operating orbits. In terms of the orbit altitude, they are mainly categorized as three types. These are Low Earth Orbit (LEO) satellites, Medium Earth Orbit (MEO) satellites, and Geostationary Earth Orbit (GEO) satellites, whose altitudes are given in Table 1.1.

Table 1.1: Satellite Classification based on Altitude [2]

Satellite Class	**Altitude (km)**
LEO satellite	$\geq$ 160 and < 1000
MEO satellite	$\geq$ 1000 and < 35786
GEO satellite	35786

The satellites are also categorized depending on their mass and this classification is

given in Table 1.2. Small satellites are defined as satellites whose mass are less than 500 kg. The majority of the small satellite missions are designed to operate in LEO.

Table 1.2: Satellite Classification based on Mass [3]

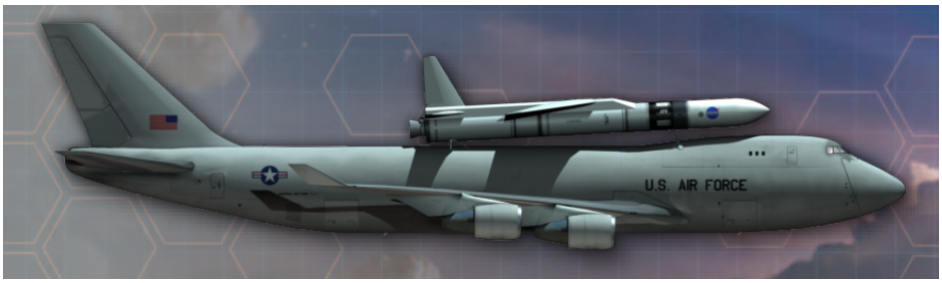
Satellite Class	**Mass (kg)**
Large satellite	$\geq$ 1000
Medium satellite	$\geq$ 500 and < 1000
Small satellite	< 500
Mini satellite	$\geq$100 and <500
Microsatellite	$\geq$10 and <100
Nanosatellite	$\geq$1 and <10
Pico satellite	$\geq$0.1 and <1

The popularity of small satellites has increased in the last decade, and this trend is going to continue. It is foreseen that there will be 2,500 small satellites on average launched each year for the next decade [4]. This resulted from the wide range of application areas of the small satellites thanks to their advantages, such as the increased rate of revisit, increase in temporal resolution, and formation flight [5]. Hence, this reveals the problem of their launch and requires an efficient solution to their access to space.

A paradigm shift in launch has been performed in the last decades. Reusable launchers were introduced and they became operational, for example Falcon 9 rocket developed by SpaceX. Today considerable amount of the launch missions are performed by reusable rockets. In this concept, some parts such as the first stage, liquid rocket engines, actuators, etc. are used in more than a mission, and their reuse makes the process cheaper compared to expendable ones.

1.1 Air Launch Method

The air launch method refers to a launch vehicle that is launched from an aircraft in the air instead of a launch pad on the ground. The rocket is released from the carrier aircraft at a certain altitude and speed. In this method, the rocket is mounted on the carrier aircraft, and it is carried to higher levels of the atmosphere and then launched from the aircraft with an initial speed compared to zero initial speed relative to the ground. This method is also called a "horizontal launch" whereas the traditional launch from the launch pad is called as "vertical launch".



(a) Example captive on top configuration [6]

(b) Example captive on bottom configuration [7]

Figure 1.1: Airlaunch Configuration Examples

There are different types of airlaunch basically due to the stowed position of the rocket. They are mainly categorized into five classes [8]:

- Captive on top
- Captive on bottom
- Towed

- Internally carried
- Aerial refueled

In the captive on top category, the large rockets can be carried by the aircraft as an advantage. However, it requires an extensive modification to the aircraft and the release in the air may be dangerous. Similarly, the rocket is stored under the plane in the captive on bottom category, which provides an easy separation and it is a proven technology. Examples of captive on top and bottom situations are given in Fig. 1.1. The rocket or the launch vehicle is towed by another aircraft in the towed category, which results in easy separation and less modification to the aircraft. In the internally carried category, the launch vehicle is carried in the inner bay of the aircraft and this provides little modification to carrier aircraft however results in strict size constraints for the carrier aircraft. In the aerial refueled scenario, the launch vehicle is refueled by another carrier aircraft at certain altitudes [8].

Figure 1.2: Launch Locations of Pegasus Rocket [9]

Airlaunch method has some advantages compared to the traditional vertical launch. The first advantage is less ΔV requirement since the rocket does not pass through the dense part of the atmosphere as experienced in ground launch. It is an important effect when the density of the atmosphere is one-fourth of the value at sea level, the fuel mass is reduced since the initial altitude is the release height [10]. In addition,

the vehicle experiences less amount of drag force. Also, the gravity loss becomes less since the time spent in ascent is shorter than the ground launch [10]. Also, there is no need for a launch pad, and quick access to space is provided by airlaunch. As an example launch points of Pegasus rocket are given in Fig. 1.2. In addition, the high cost of launch pads may be reduced. It can be concluded that the launch point at higher altitudes is better, and the optimum condition for the flight path angle is approximately 30 degrees [10]. The airlaunch also results in reduced acoustic energy, thus the design of a Thermal Protection System (TPS) and structures with less weight is possible since this process is affected by the acoustic energy at launch [8]. Moreover, thanks to lower ambient pressure, the nozzle design can be performed to be more efficient. Lastly, ground launch operations can be delayed due to the weather conditions and sea and air traffic. Unlike ground launches, airlaunch rockets are not heavily affected by them, it causes a safer launch campaign and less delayed operations for the operator and customer. It makes the airlaunch very crucial for small satellites when their fast nature in terms of build, and testing is considered.

On the other hand, there are some disadvantages associated with air launch. The first disadvantage is the size limitations of the rockets constrained by the carrier, the maximum size and the payload mass are limited in this method [11]. However, the size limitation is not significant for rockets that launch small satellites especially up to 100 kg. It also might require a modification to the aircraft, which results in cost. In addition, the boil-off for the fuel might be a problem in the case of liquid rocket engines with cryogenic propellants [12].

A real-world example to airlaunch rockets is the Pegasus XL rocket manufactured by Northrop Grumman. The carrier aircraft of the rocket is Stargazer L-1011, which has similar dimensions and spesifications with MC Donnell Douglas DC10, and it is released approximately at 12.6 km with an initial speed of 0.8 M [7]. The trajectory of Pegasus is given in Fig. 1.3 and the flight sequence is shared in Table 1.3.

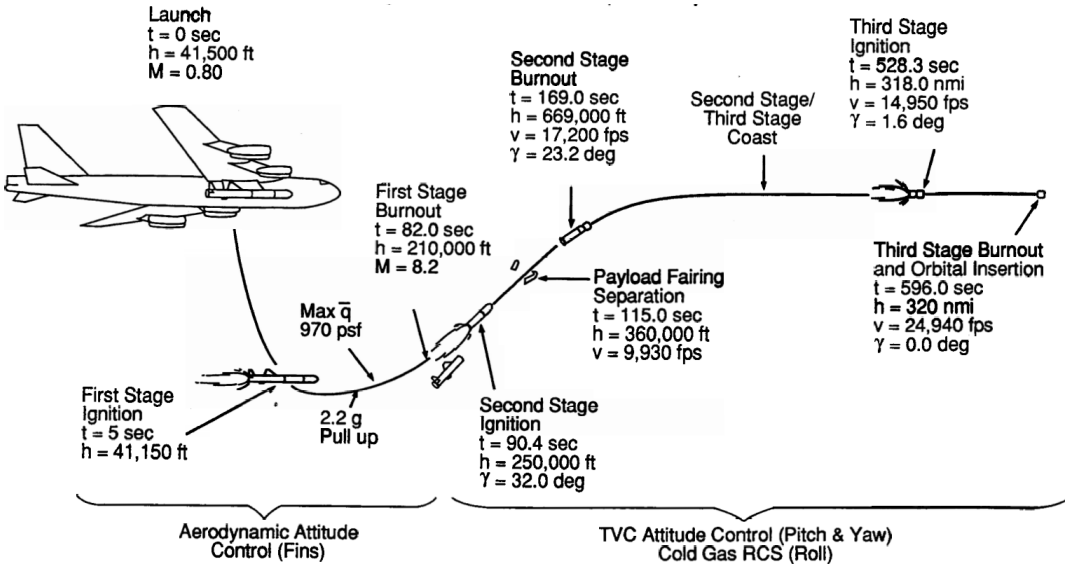


Figure 1.3: Trajectory of Pegasus XL [13]

Table 1.3: Flight Sequence of Pegasus XL [13]

Time	**Event**	**Altitude (km)**
t_0	Separation from the carrier aircraft	12.65
t_0 + 5 s	1st stage ignition	12.54
t_0 + 82 s	1st stage burnout	64
t_0 + 90.4 s	2nd stage ignition	110
t_0 + 169 s	2nd stage burnout	204
t_0 + 528.3 s	3rd stage ignition	589
t_0 + 596 s	3rd stage burnout and orbital insertion	593

1.2 Literature Survey

In this section, a brief literature survey regarding launch vehicles and spacecraft is given.

Ascent guidance is the method of guidance that steers the launch vehicle from the beginning of the flight to orbital insertion for launchers. It is desired that a successful

guidance method results in low cost, performance maximization in terms of payload, and meets the load requirements for the rocket. Usually, the flight of the rocket is divided into two groups, open-loop and closed-loop phases. In the open-loop methods, guidance commands are usually attitude commands obtained as a result of parameter optimization and are tabulated as a function of rockets' other states such as velocity, altitude, time, etc. Usually, optimal attitude angles are found using 3-DOF simulations that meet the requirements for the constraints and maximize the objective of the mission. Then those optimum commands are used in 6-DOF simulations to estimate the loads. The attitude angles are tabulated as a function of Mach in Space Shuttle [14]. It gives more performance to choose to tabulate the commands as a function of speed, and altitude rather than time since its change is linear during the flight although the others give more information about the state of the rocket. On the other hand, closed-loop algorithms use both the states of the rocket such as velocity, position, and the information regarding the desired orbit. In the closed loop part, the algorithms try to solve Two Point Boundary Value Problem (TPBVP), whose boundaries are the current state and the final state. They decide the cut-off time and calculate the necessary commands to reach the intended orbital insertion point. It is often preferred to employ open loop methods at the atmosphere effective phase i.e. endoatmospheric phase, whereas closed loop algorithms are chosen for the less effective atmosphere phase, exoatmospheric phase, or the vacuum phase [14]. The illustration of a ususal guidance scheme for a launcher is given in Fig. 1.4.

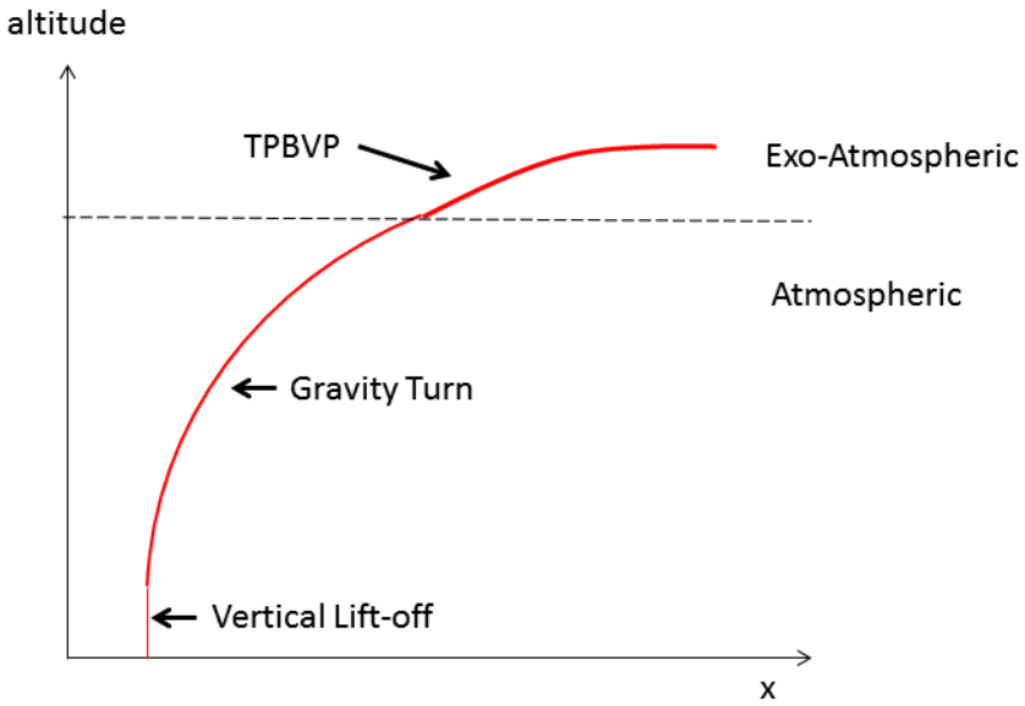


Figure 1.4: Illustration of Ascent Guidance for Ground Launch [15]

The first method developed to guide a rocket is the E guidance method, which is applied in Saturn V [16]. In this method, the acceleration of the rocket is assumed to be a polynomial of time, then the coefficients of this polynomial are found, and then thrust commands are adjusted to achieve that amount of acceleration.

It is found that the optimal guidance for rockets in the case of flat earth and constant gravity assumption is linear tangent guidance developed by [17].

The Iterative Guidance Mode (IGM) is developed at Charles Stark Draper Laboratory at Massachusetts Institute of Technology, and this algorithm is employed in Saturn V [18] [19] [20]. IGM is also based on linear tangent law. This algorithm focuses on finding the coefficients of linear tangent law considering the current and target position.

Unified Powered Flight Guidance (UPFG) is another algorithm that is developed specifically for the Space Shuttle to employ an optimal ascent for the Shuttle [21]. Later, the algorithm is renamed as Powered Explicit Guidance (PEG). This algorithm

is also used in Pegasus XL [13].

PEG is a very common algorithm, hence its modified and enhanced version is used to steer the Space Launch System (SLS) [22].

The use of techniques of machine learning is also popular for control and guidance algorithms of aerospace vehicles. An application example is to train a neural network that decides the guidance commands based on the states of the rocket to achieve the desired endpoint conditions [23].

The launch vehicle problem or the orbital insertion occurs at higher altitudes of the atmosphere where the effect of the air can be neglected, thus the solutions based on analytical techniques work well under vacuum conditions. However, the analytical methods do not perform as desired in the case of booster landing cases due to highly nonlinear effects. To overcome this problem, the trajectory optimization problem is tried to be solved onboard by employing Quadratic Programming (QP), Sequential Quadratic Programming (SQP), Model Predictive Control (MPC), etc. [24]. Online optimal control problem solver algorithms are of great interest thanks to increased computational power.

1.3 The Contribution and Outline of the Thesis

In this study, it is aimed to develop a guidance algorithm for an airlaunch rocket dedicated to deploying small satellites. The majority of launches are performed by ground-launched rockets, especially for heavier satellites. Due to the increase in the use of small satellites, it has become popular to find dedicated launch methods for them. The method of airlaunch is a promising method to realize this. Therefore, the two existing algorithms in the literature and the optimal trajectory for the launchers are investigated and applied to an airlaunch rocket. The guidance problem is an inertial guidance problem where both position and velocity constraints need to be met in an inertial frame. In this thesis, a guidance algorithm is designed to steer the launch vehicle for small satellites, whose weights are around 30-50 kg, and its performance and robustness for the different release and target conditions are presented. The airlaunched rockets are mostly carried by piloted wide-body jets however, it is

considered that the rocket in this study is launched from a UAV.

The fundamentals of rocket dynamics, orbital mechanics, and coordinate frames are given in Section 2 to clarify the concept discussed in the thesis. The methods including the optimal trajectory, Explicit guidance, and Iterative Guidance Mode are introduced and theoretically shown in Section 3. The application of the mentioned methods is applied to some example payloads and the comparison of the results is shown in Section 4. The current study is summarized and the findings are emphasized in the conclusion part of the thesis in Section 5.

CHAPTER 2

PRELIMINARIES

In this chapter, the preliminaries related to coordinate frames, orbital mechanics, rocket dynamics, and modeling of the rocket and its subsystems are given.

2.1 Coordinate Frames

In this section, coordinate frames used in launch and satellite applications are given. The most common frames are Earth Centered Earth Fixed Frame (ECEF) and Earth Centered Inertial Frame (ECI).

2.1.1 Earth Centered Inertial (ECI) Frame

The inertial frame is a frame that is fixed in space or moving at constant velocity with no acceleration in space. It is well known that the sensors that are used in satellites or rockets are measuring related quantities relative to an inertial frame. For the satellites that orbit the Earth, Earth Centered Inertial Frame is chosen as the inertial frame to define their movement. The origin of this frame is at the center of mass, the x-axis is towards the vernal equinox and it lies in the equatorial plane, the z-axis is pointing the direction of the geographical North Pole, and the z-axis is in the direction such that it completes the right-hand rule. This frame is denoted with the letter i in the superscript of the vectors. The illustration of this frame is given in Fig. 2.1.

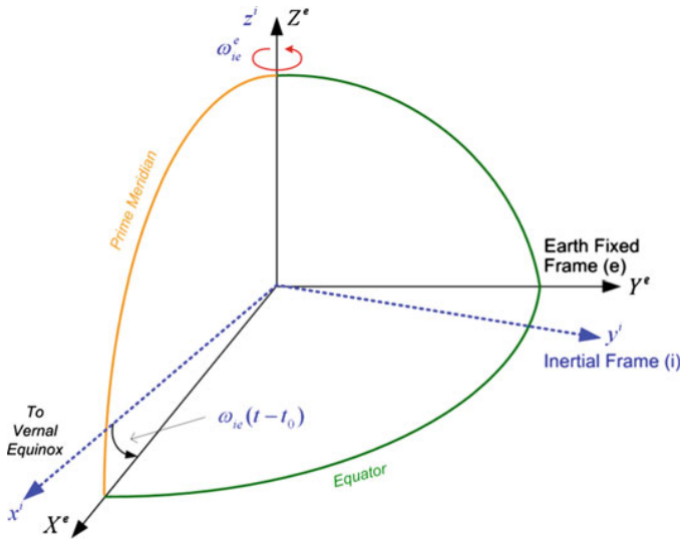


Figure 2.1: ECI and ECEF coordinate frames [25]

2.1.2 Earth Centered Earth Fixed (ECEF) Frame

The Earth Centered Earth Fixed Frame is a frame that is fixed in the Earth. It does rotate as the globe rotates unlike the inertial frame, and it is usually used for navigation purposes. The given latitude, longitude, and altitude (LLA) for the position of an object define a unique point in ECEF. The origin of this frame is at the center of mass of the Earth, the x-axis is through the intersection of the Greenwich meridian and equatorial plane, the z-axis is through the direction of the geographical North Pole and the z-axis completes the right-hand rule. This frame is denoted with the letter e in the superscript of the vectors. The illustration of this frame is given in Fig. 2.1.

2.1.3 Coordinate Transformations

The movement of aerospace vehicles is usually described by resolving frame. It is important to define a vector using its resolving frame, which states the vector is expressed in which frame. For example, the position vector of a satellite is a constant vector in space whereas the same vector can be expressed in either ECI frame or ECEF frame, both representations are different although they point the same vector. The resolving frame of a vector is given as a superscript of the vector notation. For

example, the position vector of a vehicle ($\mathbf{r}$) in the ECI frame is denoted as a bold lowercase letter with superscript i, ($\mathbf{r^i}$).

The same vector can be resolved in different frames as mentioned, and coordinate transformation is the operation to express a vector in a different frame using the representation in another frame. The coordinate transformations are made using Direction Cosine Matrix (DCM, represented with $\mathbf{C}$ in this thesis), which is a 2x2 or 3x3 matrix in 2D and 3D, respectively. Any DCM used is denoted with a bold capital letter where the subscript implies the frame in which the vector is expressed and the superscript implies the frame into the vector will be expressed. DCM expresses a vector in the ECI frame using its expression resolved in the ECEF frame is given in equation (2.1).

$$\mathbf{r^i} = \mathbf{C^i_e r^e} \tag{2.1}$$

There are some important features of DCM that are given in equations (2.2) - (2.3).

$$\det \mathbf{C} = 1 \tag{2.2}$$

$$\mathbf{C_i^{e-1}} = \mathbf{C_e^i} = \mathbf{C_i^{e^T}} \tag{2.3}$$

Those properties are valid for any DCM used for coordinate transformation. There are rotation matrices that help to rotate a vector around x, y, and z axes, which are given as follows.

$$\mathbf{C(x)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \tag{2.4}$$

$$\mathbf{C(y)} = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix} \tag{2.5}$$

$$\mathbf{C}(\mathbf{z}) = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

Using the elementary rotation matrices, vectors can be expressed in other frames as a result of successive rotations around three axes.

2.2 Orbital Mechanics

The types of orbits and their examples with an emphasis on small satellites will be given. Fundamentals of the orbital mechanics will be mentioned here. In this study, the motion of two bodies due to their gravitational attraction i.e., the two-body problem is considered. Usually, smaller objects whose mass is m_2 in terms of mass are affected by the bigger object whose mass is m_1 although the reverse effect is valid but usually ignored. ECI frame is preferred in orbital equations. The position of the heavy body is denoted with $\mathbf{r_1}$ whereas the position vector of the smaller mass is denoted with $\mathbf{r_2}$. The relative position of the orbiting object with respect to a bigger object is denoted with $\mathbf{r}$ and it is found by subtracting the position vector of each object.

$$\mathbf{r} = \mathbf{r_2} - \mathbf{r_1} \tag{2.7}$$

$$r = |\mathbf{r}| \tag{2.8}$$

The gravitational acceleration of the second body due to only gravitational force exerted by the main body is

$$\ddot{\mathbf{r}} = -\frac{G(m_1 + m_2)}{r^2}\hat{\mathbf{r}} \tag{2.9}$$

where G is the universal gravitational constant and $\hat{\mathbf{r}}$ is the unit vector of relative position vector $\mathbf{r}$. At this point, the gravitational parameter μ is defined for the simplification. The mass of the orbiting body can be neglected when the order of magnitude difference is present between two body masses.

$$\mu = G(m_1 + m_2) \approx G(m_1) \tag{2.10}$$

The body m_1 becomes the Earth in case an Earth-orbiting spacecraft is considered. Therefore, μ becomes a constant parameter and it is given as

$$\mu = Gm_{earth} \tag{2.11}$$

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^2}\hat{\mathbf{r}} \tag{2.12}$$

Relative angular momentum per unit mass (specific angular momentum) of the body m_2 with respect to m_1 is denoted with $\mathbf{h}$, and it is

$$\mathbf{h} = \mathbf{r} \times \dot{\mathbf{r}} \tag{2.13}$$

When the derivative of the specific angular momentum vector with respect to time, it is observed that the angular momentum is constant and not varying with time.

$$\dot{\mathbf{h}} = 0 \tag{2.14}$$

The magnitude of the specific angular momentum is equal to the product of the magnitude of the relative position vector and horizontal (perpendicular) velocity v_r.

$$h = rv_h \tag{2.15}$$

After a brief introduction of the two-body problem, the orbit equation that relates the position of the spacecraft with the orbital radius for the path of body 2 around body 1 is given as

$$r = \frac{h^2}{\mu}\frac{1}{1 + e\cos\theta} \tag{2.16}$$

where e is the orbit eccentricity and constant parameter similar to specific angular momentum (h) and parameter (μ), and θ is the true anomaly of the body. The radial velocity of the body m_2 is given by the following expression

$$v_r = \frac{h^2}{\mu}e\sin\theta \tag{2.17}$$

Eccentricity is a parameter that shows the type of the orbit. The orbit types depending on eccentricity are listed.

- Circular orbit: $e = 0$
- Elliptical orbit: $0 < e < 1$
- Parabolic orbit: $e = 1$

- Hyperbolic orbit: $e > 1$

Most small spacecraft missions aim to insert the satellite into a circular or elliptical orbit. If required the spacecraft can be transferred to a parabolic or hyperbolic orbit once after the Launch and Early Orbit Phase (LEOP) operations, specifically for interplanetary spacecraft. Therefore, only circular and elliptical orbits are of interest in this thesis. The position and velocity relations in those orbits are given in subsections of this section. The rocket must satisfy the required pair of position and velocity for the desired orbit of the spacecraft.

2.2.1 Circular Orbit

The eccentricity value is 0 for circular orbits and its description is given in Fig. 2.2. Thus, the orbital equation becomes

$$r = \frac{h^2}{\mu} \tag{2.18}$$

The object has no radial velocity if it is moving in a circular orbit, there is only a horizontal component of the velocity.

$$v_h = \sqrt{\frac{\mu}{r}} \tag{2.19}$$

The period for a circular orbit is given as

$$T = \frac{2\pi}{\sqrt{\mu}}\sqrt{r^3} \tag{2.20}$$

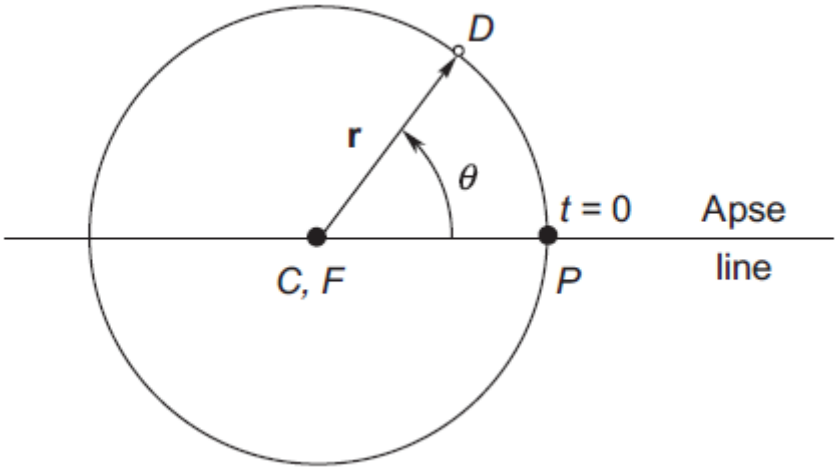


Figure 2.2: Illustration of Circular Orbit [26]

2.2.2 Elliptical Orbit

The eccentricity is between 0 and 1 for an elliptical orbit and its description is given in Fig. 2.3. The orbital equation is the same as defined before.

$$r = \frac{h^2}{\mu}\frac{1}{1 + e\cos\theta} \tag{2.21}$$

It is important to note that r reaches its maximum value when θ is π whereas reaches the minimum value when θ is 0, and those points are called apogee and perigee, respectively. Distance of apogee and perigee points are given as

$$r_a = \frac{h^2}{\mu}\frac{1}{1 - e} \tag{2.22a}$$

$$r_p = \frac{h^2}{\mu}\frac{1}{1 + e} \tag{2.22b}$$

The eccentricity of an elliptical orbit can be found using r_p and r_a.

$$e = \frac{r_a - r_p}{r_a + r_p} \tag{2.23}$$

For the velocity of an object orbiting in an elliptical orbit at any distance r, the equation might be helpful.

$$\frac{v^2}{2} - \frac{\mu}{r} = \frac{-\mu}{2a} \tag{2.24}$$

where a is the length of the semimajor axis and it is given by

$$a = \frac{h^2}{\mu}\frac{1}{1 - e^2} \tag{2.25}$$

Lastly, the period for one orbit can be found using

$$T = \frac{2\pi}{\sqrt{\mu}}\sqrt{a^3} \tag{2.26}$$

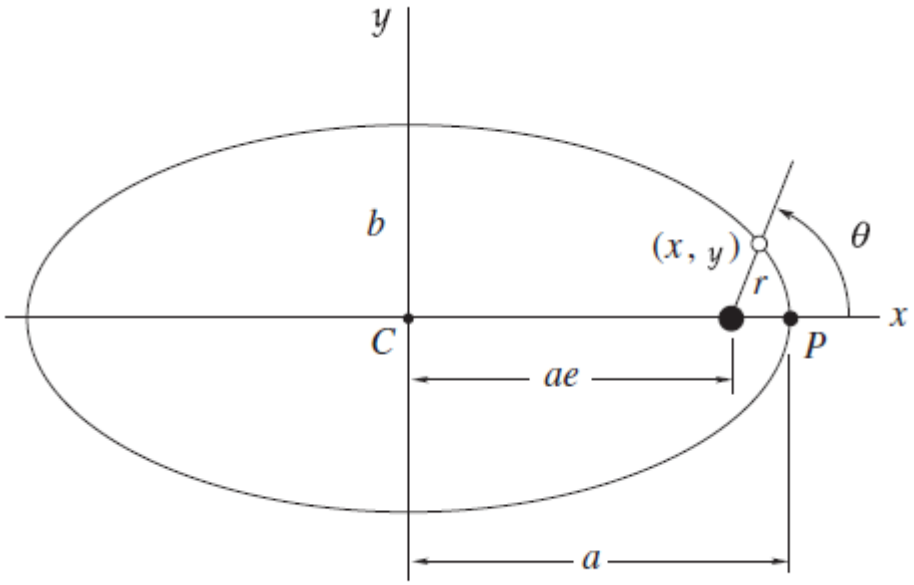


Figure 2.3: Illustration of Elliptical Orbit [26]

It is important to note that again the rocket must satisfy the position and velocity constraints required by the orbit within an acceptable error range.

2.3 Rocket Dynamics

The equations of motion of a rocket moves in the atmosphere are given in equation (2.27) and (2.28).

$$\dot{\mathbf{r}} = \mathbf{v} \tag{2.27}$$

$$\dot{\mathbf{v}} = -\frac{\mu}{|\mathbf{r}|^3}\mathbf{r} + \frac{T}{m}\mathbf{u_T} - \frac{D}{m}\mathbf{u_D} \tag{2.28}$$

The change in mass of the rocket is given in (2.29) [27].

$$\dot{m} = -\frac{T}{I_{sp}g_0} \tag{2.29}$$

$\mathbf{r}, \mathbf{v}$ represent position vector, and velocity vector, respectively. m, T stands for the mass of the rocket and the magnitude of thrust force. I_{sp} is the specific impulse and g_0 stands for the standard gravity. Lastly, $\mathbf{u_T}, \mathbf{u_D}$ denotes the unit vector of thrust and the aerodynamic drag force. The equation (2.28) is valid for the atmospheric part of the flight. At some point, the atmospheric effects can be neglected by removing the drag effect in equation (2.28) for the exo-atmospheric part of the flight. μ is a constant parameter which is the multiplication of universal gravitational constant G and mass of the earth m_E which was mentioned in section 2.2.

$$\mu = Gm_E \tag{2.30}$$

Aerodynamic drag force is expressed as the dynamic pressure ($\frac{1}{2}\rho|\mathbf{v}|^2$) multiplied by reference area (S) and drag coefficient (C_D).

$$D = \frac{1}{2}\rho|\mathbf{v}|^2 SC_D \tag{2.31}$$

The drag force acts on the vehicle in the opposite direction to its velocity.

$$\mathbf{u_D} = \frac{-\mathbf{v}}{|\mathbf{v}|} \tag{2.32}$$

The equations (2.27), (2.28) and (2.29) are the dynamic constraints that are considered in the trajectory optimization problem mentioned in section 3.1.

2.4 Modeling

In this section, the rocket model used in this study is given. The overview characteristics, aerodynamic, and thrust properties are given briefly in corresponding sections. In addition, environmental models such as gravity and atmosphere are given.

2.4.1 Rocket Model

In this study, a hypothetical launch vehicle is considered to employ the guidance algorithms developed and tested under various conditions that arise from the nature of the air-launch method. It is aimed to consider a rocket that aims to insert a small satellite whose mass ranges from 10-50 kg into LEO orbit. A configuration of the

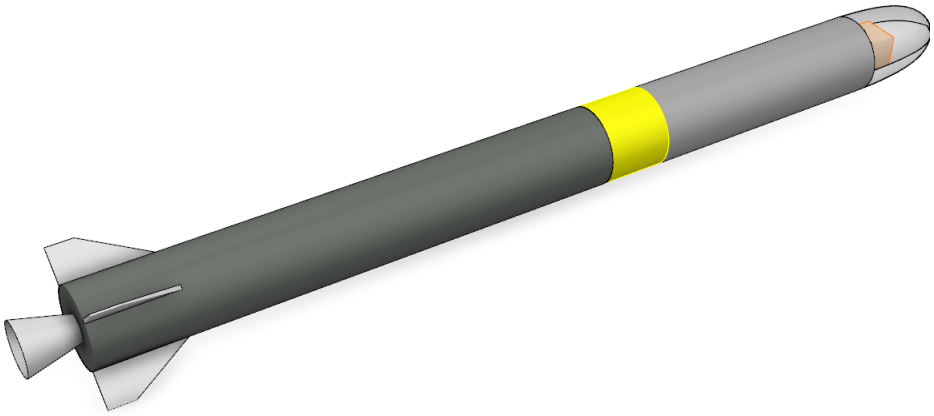


Figure 2.4: Overview of the Hypothetical Airlaunch Vehicle

two-stage rocket is selected. Both stages have a liquid rocket engine, whose details are given in section 2.4.1.2. For the first stage of the rocket, 4 aerodynamic fins and Thrust Vector Control (TVC) are used the control the rocket. It is desired to have aerodynamic fins since the atmosphere is effective at the release conditions, thus aerodynamic control surfaces are used to control the roll maneuver of the rocket and help the TVC subsystem to realize the commands generated by the control algorithm. On the other hand, it is assumed that there will be a Reaction Control System (RCS) that controls the roll channel and a TVC subsystem for the pitch and yaw commands for the second stage of the rocket. The overview of the rocket is given in Fig. 2.4.

Table 2.1: Rocket Properties

Property	**Value**
Diameter (1^{st} stage)	0.8 m
Diameter (2^{st} stage)	0.8 m
Length (1^{st} stage)	10 m
Length (2^{nd} stage)	3.5 m
Mass	6500 kg

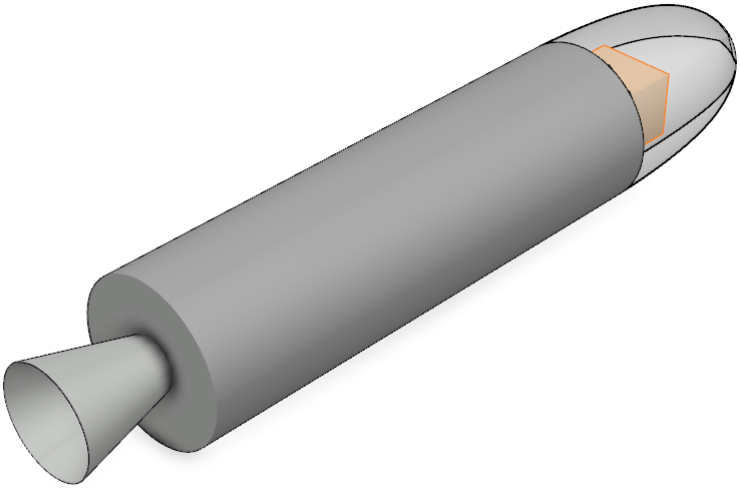


Figure 2.5: Overview of the Upper Stage

Figure 2.6: Fairing and Payload

2.4.1.1 Aerodynamic Model

It is important to consider the aerodynamic drag forces for the endoatmospheric flight of the rocket as mentioned in section 2.3. Since the trajectory optimization is done using 3-DOF simulations, only the drag coefficient is found for the rocket, the moment coefficients are not investigated for this study. For the aerodynamic coefficients calculations, a Panel method is used to obtain the drag coefficient of the rocket. The external shape of the rocket is defined the shape is given in Fig. 2.7.

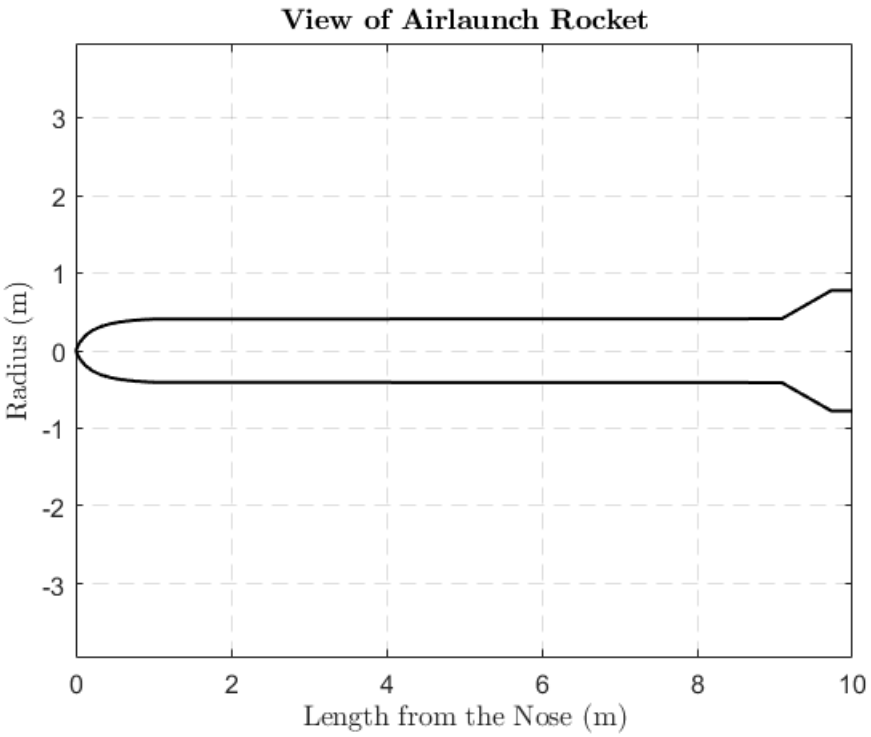


Figure 2.7: Rocket Drawing for Drag Coefficient Calculation

The drag and lift coefficients are calculated for the following conditions.

$$M = \{1.2, 1.5, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \tag{2.33a}$$

$$h = \{10, 20, 30, 40, 50, 60, 70, 80\} km \tag{2.33b}$$

$$\alpha = \{0\}^{\circ} \tag{2.33c}$$

The aerodynamic coefficients are found for each condition mentioned above. Aerodynamic coefficients are interpolated using the information gained from the analysis in the simulation and optimal trajectory.

The drag force acting on the rocket is calculated using the equation

$$D = \frac{1}{2}\rho V^2 S C_D \tag{2.34}$$

where S is the reference area and it is equal to the crossectional area of the rocket.

2.4.1.2 Thrust Model

The engines are selected to be using liquid fuel. There are mainly two types of rocket engines that are used in launchers, which are solid rocket motors and liquid rocket engines. Also, there exists a hybrid version of them, however, hybrid engines are not that much preferred in current applications. The liquid rocket engines can provide thrust termination and re-ignition and they can be throttled if necessary. There must be a liquid rocket engine for the second stage of the rocket when the necessity of thrust termination is considered after reaching the desired orbit to deploy the satellite and re-ignition of the engine for the de-orbiting of the rocket after deployment. A similar rocket engine is selected to power the first stage of the rocket for simplicity. As a result, both engines are selected to be liquid rocket engines and use the same type of fuel. The properties of the engines are given in the following table.

Table 2.2: Liquid Rocket Engine Properties

Property	**Value**
I_{sp}	400 s
V_{ex}	3924 m/s
$\dot{m}_{fuel}$	30 kg/s

The thrust (T) generated by the engine is calculated using the fuel mass flow rate $(\dot{m}_{fuel})$ and the exit velocity (V_{ex}).

$$V_{ex} = g_0 I_{sp} \tag{2.35}$$

$$T = \dot{m}_{fuel} V_{ex} \tag{2.36}$$

2.4.2 Gravity Model

The inverse-square gravity model which basically state that the gravitational force is changing with the square of the distance is used in this study.

$$\mathbf{g} = -\frac{\mu}{r^3}\mathbf{r} \tag{2.37}$$

2.5 Atmosphere Model

The U.S. Standard Atmosphere model which is a static atmosphere model is used in this study. The properties of the atmosphere including air density, pressure, and temperature at different altitudes can be found in [28]. The parameters denoted with subscript 0 are the values at sea level conditions. The values at sea level conditions are given in the Table 2.3.

Table 2.3: Properties of the Atmosphere at Sea Level

Property	**Value**
ρ_0	1.225 m^3/kg
T_0	288.15 K
P_0	101325 Pa
γ	1.4
R	287.053 $\frac{J}{kgK}$

The speed of sound of the air at an altitude can be found by multiplying the specific heat ratio (γ), specific gas constant (R) and the temperature (T).

$$a = \gamma RT \tag{2.38}$$

The density of the air ρ at an altitude can be found after pressure and temperature values are obtained at that altitude using the equation (2.39).

$$\rho = \rho_0 \frac{P}{P_0}\frac{T_0}{T} \tag{2.39}$$

Therefore, first temperature and pressure values at the desired altitude are calculated in order to calculate density. The calculation of pressure and temperature changes for different sections of the atmosphere [28].

- If $0 < h < 11000$

$$T = 288.15 - 0.0065h \tag{2.40}$$

$$P = 101325 \left[\frac{T}{288.15}\right]^{5.2559} \tag{2.41}$$

- If $11000 < h < 20000$

$$T = 216.65 \tag{2.42}$$

$$P = 22632.064e^{(1.73457 - 0.0001577h)} \tag{2.43}$$

- If $20000 < h < 32000$

$$T = 196.65 + 0.001h \tag{2.44}$$

$$P = 5474.889 \left[\frac{T}{216.65}\right]^{-34.163} \tag{2.45}$$

- If $32000 < h < 47000$

$$T = 139.05 + 0.0028h \tag{2.46}$$

$$P = 868.03 \left[\frac{T}{228.65}\right]^{-12.2011} \tag{2.47}$$

- If $47000 < h < 51000$

$$T = 270.65K \tag{2.48}$$

$$P = 110.9063e^{(5.932644 - 0.00012623h)} \tag{2.49}$$

- If $51000 < h < 71000$

$$T = 413.45 - 0.0028h \tag{2.50}$$

$$P = 66.9389 \left[\frac{T}{270.65}\right]^{12.2011} \tag{2.51}$$

- If $71000 < h < 84852$

$$T = 356.65 + 0.002h \tag{2.52}$$

$$P = 3.95642 \left[\frac{T}{228.65} \right]^{17.0816} \tag{2.53}$$

- If $84852 < h < 86000$

$$T = 186.946K \tag{2.54}$$

$$P = 0.3734e^{(15.5062 - 0.000182744h)} \tag{2.55}$$

The density is assumed to be zero for altitudes higher than 86 km since the effect of air resistance can be neglected beyond that point for a launcher.

CHAPTER 3

METHODS

In this chapter, the trajectory optimization problem is introduced and the solution methods are briefly mentioned. Then, two of these methods are employed and the results are discussed.

3.1 Trajectory Optimization Problem

In this section, the Optimal Control Problem (OCP) for the launch vehicle is introduced.

The problem of launching a satellite into a desired orbit can be considered a trajectory optimization problem i.e., an optimal control problem. The initial states are the states that belong to the release condition whereas the final states are the states that meet the required orbit. States are defined as the position and velocity of the rocket.

$$\mathbf{r_i} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{3.1}$$

$$\mathbf{v_i} = \begin{bmatrix} v_{x_i} \\ v_{y_i} \\ v_{z_i} \end{bmatrix} \tag{3.2}$$

Terminal conditions can be defined as either position and velocity as in equations (3.3) and (3.4). The final condition which is actually the same as defining the inertial position and velocity since Kepler orbital elements can be easily obtained from the

inertial position and velocity vectors might also be expressed in (3.5).

$$\mathbf{r_f} = \begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix} \tag{3.3}$$

$$\mathbf{v_f} = \begin{bmatrix} v_{x_f} \\ v_{y_f} \\ v_{z_f} \end{bmatrix} \tag{3.4}$$

$$\mathbf{o_f} = \begin{bmatrix} a_f & e_f & i_f & \Omega_f & \omega_f & \theta_f \end{bmatrix}^T \tag{3.5}$$

The beginning point is indicated by the subscript i, and the final point, which is after T_{go} seconds from the initial point, is indicated by the subscript f. For the final conditions, a is the semi-major axis, e is the orbit's eccentricity, i is its inclination, Ω is the right ascension of the ascending node, ω is the argument of perigee of the orbit, and θ is the orbit's actual anomaly [29]. It is desired for the rocket's trajectory to insert the payload with the least amount of fuel expenditure or with the largest payload mass possible. As a result, the fuel mass specified in (3.6) can be used to define the cost function (J), and its minimization is the goal of the problem [29].

$$J = \int_{t_0}^{t_f} \dot{m}_{fuel}\, dt = m_{fuel} \tag{3.6}$$

The terms m_{fuel} and $\dot{m}_{fuel}$ refer to the mass of the fuel consumed and the fuel mass flow rate, respectively. assuming only the exo-atmospheric portion of the flight is taken into account, the issue can be expressed as a minimum time problem because, assuming a constant mass flow rate for the fuel is assumed, limiting the time results in minimizing fuel consumption [29].

$$J = \int_{t_0}^{t_f} dt \tag{3.7}$$

The ignition and cut-off start times are denoted by t_0 and t_f, respectively. The problem is an optimal control problem with free final time and fixed final conditions but fixed initial time and conditions[29].

For the dynamic constraints, the rocket dynamics given in equations (2.27) - (2.28) are considered. They define the dynamic constraints that the system subjects to.

There are mainly two methods that solve optimal control problems; analytical and numerical methods. For the analytical methods, the problem is solved analytically, however, it is not usually possible to solve aerospace trajectory problems due to its complex and nonlinear nature. Thus, numerical methods are generally preferred for the solution of trajectory optimization problems. The numerical methods are divided into two subgroups, indirect and direct methods. In indirect methods, Two Point Boundary Value Problem (TPBVP) that contains the necessary conditions of the optimality is solved using numerical methods such as the shooting method, etc. On the other hand, in direct methods, the states and controls are parameterized into finite-dimensional Nonlinear Programming (NLP) problem by applying certain discretization methods. Then, this nonlinear programming problem is solved using complex numerical methods. At the end, the optimal trajectory and the control input set are found. In the literature, there are developed software that solve optimal control problems and some examples of them are DIO, PSOPT, GPOPS-II [24] and FALCON.m [30], which is used in this study. The FALCON.m is a MATLAB-based optimal control problem solver using direct discretization methods and it interfaces to common NLP solvers such as Ipopt, etc. It uses collocation methods and the time is normalized during the solution procedure. When the defined problem is solved, the states and controls that minimize the cost under given constraints are found and they are called optimal trajectory (x^*) and optimal control (u^*) [23].

3.2 E Guidance Method

In this section, the E-guidance method introduced by Cherry [16] is discussed and its implementation for an airlaunch rocket is given.

The E guidance method is an explicit method that is developed and used in Saturn V [3]. An explicit algorithm is the algorithm that aims to find the steering command by considering the current position, velocity, and desired position, and velocity for the vehicle. This method considers two different approaches for the rockets; the first one

is the rocket with throttleable engines presented in this study.

3.2.1 Varying Thrust Case

In this subsection, the derivation for the throttleable engine i.e. varying thrust case is given.

The E-guidance method, for varying thrust cases, aims to find a solution for two-point boundary value problem defined in the section 3.1. It is intended to solve the problem whose initial and final conditions are given as

$$\mathbf{r}(t_0) = \mathbf{r_i} \tag{3.8}$$

$$\mathbf{v}(t_0) = \mathbf{v_i} \tag{3.9}$$

$$\mathbf{r}(t_f) = \mathbf{r_f} \tag{3.10}$$

$$\mathbf{v}(t_f) = \mathbf{v_f} \tag{3.11}$$

T_{go} is time-to-go, i.e. time between initial and the final time

$$T_{go} = t_f - t_0 \tag{3.12}$$

The equations of motion can be written as the following

$$\ddot{\mathbf{r}} = \mathbf{g} + \mathbf{a_T} \tag{3.13}$$

$\mathbf{g}$ is the gravitational acceleration vector and $\mathbf{a_T}$ is the acceleration vector due to the thrust of the engine.

$$\mathbf{g} = \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} \tag{3.14}$$

$$\mathbf{a_T} = \begin{bmatrix} a_{T_x} \\ a_{T_y} \\ a_{T_z} \end{bmatrix} \tag{3.15}$$

g_x, g_y, g_z are gravitational accelerations, and $a_{T_x}, a_{T_y}, a_{T_z}$ are the thrust accelerations in respective axis of the inertial coordinate system. The motion can be expressed in scalar form for each element of the position vector.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} g_x + a_{T_x} \\ g_y + a_{T_y} \\ g_z + a_{T_z} \end{bmatrix} \tag{3.16}$$

The E guidance method aims to find a solution for the thrust acceleration $(\mathbf{a_T})(t)$ that will satisfy the boundary conditions that are given in equations (3.8) - (3.11) and subjected to dynamics of the motion given in equation (3.13). At the same time, it can be intended to minimize the fuel expenditure which reflects the variational aspect of the problem [16].

$$J = \int_{t_0}^{t_f} \sqrt{\mathbf{a_T}.\mathbf{a_T}}\, dt \tag{3.17}$$

E guidance method aims to find the solution for thrust acceleration vector $(\mathbf{a_T})$ that will satisfy the boundary conditions while minimizing the mass burned [16]. The analysis can be decomposed into three components, the process for one axis is explained and the same procedure is applied for the remaining axes. The motion in x-axis can be considered which is given in equation (3.16).

$$\ddot{x}(t) = g_x(t) + a_{T_x}(t) \tag{3.18}$$

The component $a_{T_x}(t)$ is desired to be found, and then the other components of the thrust vector are found, as a result, the thrust vector in the inertial frame is obtained. In this method, the solution for $\ddot{x}(t)$ is found instead of finding $a_{T_x}(t)$ directly [16]. When $\ddot{x}(t)$ is found, $a_{T_x}(t)$ can be found by simply subtracting the gravity component [16].

$$a_{T_x}(t) = \ddot{x}(t) - g_x(t) \tag{3.19}$$

If the equation (3.18) is integrated once the following expression is obtained.

$$\int_{t_0}^{t_f} \ddot{x}(t)dt = \dot{x}(t_f) - \dot{x}(t_0) \tag{3.20}$$

When the equation (3.20) is integrated between t_0 and t_f, the following expression is obtained.

$$\int_{t_0}^{t_f} \int_{t_0}^{t_f} \ddot{x}(t)dt = \dot{x}(t_f) - \dot{x}(t_0)dt = x_f - x_i - v_{x_i}(t_f - t_0) \tag{3.21}$$

Now, the unknown acceleration function $\ddot{x}(t)$ can be expressed as a series expansion, that only two coefficients can be found from equations (3.20) and (3.21) [16]. Assume an acceleration function in the form of the following

$$\ddot{x}(t) = c_1 y_1(t) + c_2 y_2(t) \tag{3.22}$$

where $y_1(t)$ and $y_2(t)$ are linearly independent and integrable functions of time (t), that are chosen by the user. They are in the form of:

$$y_1(t) = a_0 + a_1(t_f - t)^1 + a_2(t_f - t)^2 + a_3(t_f - t)^3 + ... \tag{3.23}$$

$$y_2(t) = b_0 + b_1(t_f - t)^1 + b_2(t_f - t)^2 + b_3(t_f - t)^3 + ... \tag{3.24}$$

The coefficients a_i, b_i in equations (3.23) and (3.24) are selected by user. In addition selection of the sub-functions of time are also decided by the user. It is mentioned that the selection of the polynomial does not considerably affect the optimization i.e., burned fuel mass. Instead of that, the coefficients (a_i, b_i) of the terms affect mostly. However, this process can be done via brute force, trial-error, and optimal values for coefficients can be found [16]. In this study, different solutions that utilize distinct values for a_i, b_i are obtained for the same problem. Then the coefficient set that gives the minimum amount of fuel is selected. The whole process is performed using MATLAB fminsearch command [31].

Since the polynomials are defined, c_1 and c_2 are the coefficients that need to be found. The assumed solution for $\ddot{x}(t)$ is plugged into the equations (3.20) and (3.21), and the following expressions are obtained.

$$v_{x_f} - v_{x_i} = c_1 \int_{t_0}^{t_f} y_1(t)dt + c_2 \int_{t_0}^{t_f} y_2(t)dt \tag{3.25}$$

$$x_f - x_i - v_{x_i}(t_f - t_0) = c_1 \int_{t_0}^{t_f} \left(\int_{t_0}^{t_f} y_1(t)dt \right) dt + c_2 \int_{t_0}^{t_f} \left(\int_{t_0}^{t_f} y_2(t)dt \right) dt \tag{3.26}$$

In equations (3.25) and (3.26), only unknown terms are c_1 and c_2. If that system of equations is solved for unknowns, the solution for the required coefficients is obtained. This expression can be written in matrix form as follows

$$\begin{bmatrix} v_{x_f} - v_{x_i} \\ x_f - x_i - v_{x_i}(t_f - t_0) \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{3.27}$$

where f_{ij}'s are the integral terms listed below.

$$f_{11} = \int_{t_0}^{t_f} y_1(t)dt \tag{3.28a}$$

$$f_{12} = \int_{t_0}^{t_f} y_2(t) \tag{3.28b}$$

$$f_{21} = \int_{t_0}^{t_f} \left(\int_{t_0}^{t_f} y_1(t)dt \right) dt \tag{3.28c}$$

$$f_{22} = \int_{t_0}^{t_f} \left(\int_{t_0}^{t_f} y_2(t)dt \right) dt \tag{3.28d}$$

The matrix whose components are $f_{11}, f_{12}, f_{21}, f_{22}$ is called as F matrix. If this system of equations is rewritten to find out coefficients, it obtained that

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} \begin{bmatrix} v_{x_f} - v_{x_i} \\ x_f - x_i - v_{x_i}(t_f - t_0) \end{bmatrix} \tag{3.29}$$

The matrix whose components are $e_{11}, e_{12}, e_{21}, e_{22}$ is called as E matrix, and it is inverse of the F matrix.

$$E = F^{-1} \tag{3.30}$$

The solution procedure is to calculate the E matrix first, then calculate values for coefficients c_1 and c_2. The algorithm for the varying thrust case is summarized in Algorithm 1.

Algorithm 1 Varying Thrust E Guidance Algorithm

1: $T_{go} \leftarrow T_{go}$ ▷ User Input
2: $y_1(t)$ and $y_2(t)$ are defined
3: E matrix is found
4: c_1 and c_2 are found
5: $\ddot{x}$ is obtained
6: a_{T_x} is found using $a_{T_x}(t) = \ddot{x} - g_x(t)$

The same procedure is applied for the remaining two axes, the y and z axes. The acceleration in the y and z axis of the inertial system is also assumed to be a function

of time given in equation (3.31)

$$\ddot{x} = c_1 y_1(t) + c_2 y_2(t) \tag{3.31a}$$

$$\ddot{y} = c_3 y_3(t) + c_4 y_4(t) \tag{3.31b}$$

$$\ddot{z} = c_5 y_5(t) + c_6 y_6(t) \tag{3.31c}$$

where $y_1(t), y_2(t), y_3(t), y_4(t), y_5(t), y_6(t)$ are linearly independent, integrable and operator-defined functions of time (t). It is important to note that the pairs $y_1(t) - y_2(t)$, $y_3(t) - y_4(t)$, and $y_5(t) - y_6(t)$ must be linearly independent. Polynomials in individual axes can be the same. The selection of y_i's is similar to that done in the x-axis of the problem.

$$y_3(t) = c_0 + c_1(t_f - t)^1 + c_2(t_f - t)^2 + c_3(t_f - t)^3 + \tag{3.32}$$

$$y_4(t) = d_0 + d_1(t_f - t)^1 + d_2(t_f - t)^2 + d_3(t_f - t)^3 + ... \tag{3.33}$$

$$y_5(t) = e_0 + e_1(t_f - t)^1 + e_2(t_f - t)^2 + e_3(t_f - t)^3 + \tag{3.34}$$

$$y_6(t) = f_0 + f_1(t_f - t)^1 + f_2(t_f - t)^2 + f_3(t_f - t)^3 + ... \tag{3.35}$$

c_1 and c_2 were found beforehand. The same procedure is applied to obtain the values of c_3, c_4, c_5, c_6. The thrust acceleration required in other axes is found easily since the whole accelerations are already found.

$$a_{T_y}(t) = \ddot{y} - g_y(t) \tag{3.36}$$

$$a_{T_z}(t) = \ddot{z} - g_z(t) \tag{3.37}$$

The resultant thrust vector in the inertial frame is represented as

$$\mathbf{a_T} = a_{T_x}\hat{i} + a_{T_y}\hat{j} + a_{T_z}\hat{k} \tag{3.38}$$

Since this method is applicable to varying thrust rocket engine cases, the thrust acceleration magnitude is given as

$$|\mathbf{a_T}| = \sqrt{a_{T_x}^2 + a_{T_y}^2 + a_{T_z}^2} \tag{3.39}$$

Consequently, the thrust level of the engine becomes

$$T(t) = |\mathbf{a_T}| m(t) \tag{3.40}$$

The throttle command for the liquid rocket engine is adjusted such that the required thrust is supplied to the rocket. It should be noted that the thrust accelerations are expressed in the inertial frame. This procedure can be repeated during flight for a better result. It is important to note that this method relies on the mathematical model and assumes perfect subsystem, thus it is vital to perform the calculations during the flight to overcome the problems that arise from imperfections and uncertainties.

3.3 Iterative Guidance Mode (IGM)

In this section, the Iterative Guidance Method is discussed and its implementation in an airlaunch rocket is given. Iterative Guidance Method is an iterative algorithm that is designed to steer the spacecraft in space where atmosphere effects can be neglected. The algorithm is used in many spacecraft including Saturn V and Space Shuttle [21], and it is a guidance algorithm that is based on the well-known linear tangent steering law applied to rockets. In this method, the thrust angle regime of the rocket is calculated using the position of the rocket and the target orbit. The algorithm is developed based on Smith's work [18].

3.3.1 2-D Flat Earth Case

Equations (3.41) and (3.42), which represent acceleration in both the x and y axes, provide the equations of motion of an object under the assumptions of a flat earth and uniform constant gravity. In Fig. 3.1, the thrust attitude angle [18] is given in 2-D coordinate axes.

$$\ddot{x} = \frac{T}{m}\cos\gamma \tag{3.41}$$

$$\ddot{y} = \frac{T}{m}\sin\gamma - g \tag{3.42}$$

In equations (3.41) and (3.42), T is the magnitude of the thrust, m is the mass of the rocket, and g is the gravitational acceleration, which is constant in this case. The solution for thrust attitude γ is bilinear tangent law if flat earth and constant uniform gravity are assumed [18].

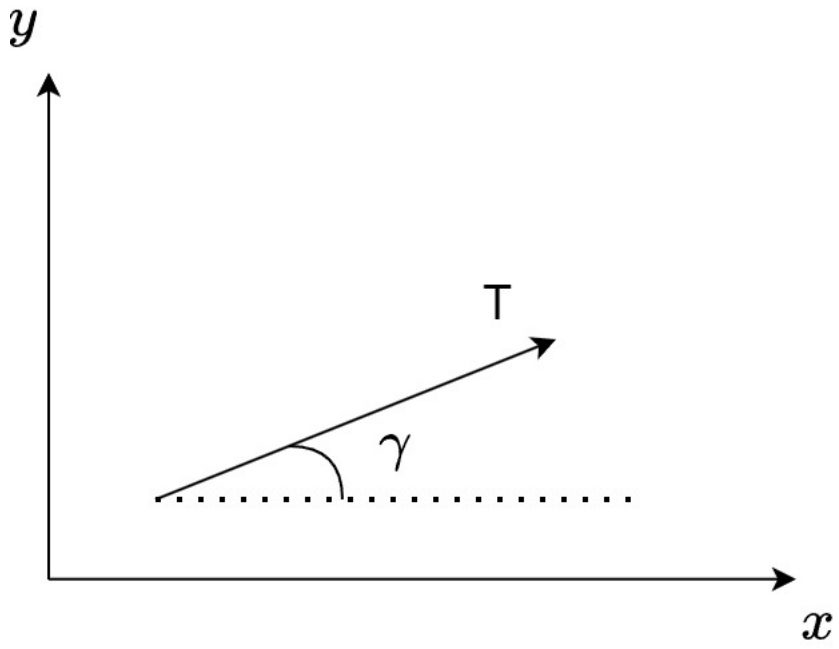


Figure 3.1: Thrust attitude angle (γ)

$$\tan\gamma = \frac{k_1 + k_2 t}{k_3 + k_4 t} \tag{3.43}$$

k_1, k_2, k_3 and k_4 are all constants, and t is the time.

For terminal constraints, if only velocity and height constraints are considered, the solution to optimal solution for the thrust attitude angle yields to linear tangent law [32].

$$\tan\gamma = c + dt \tag{3.44}$$

In equation 3.44, c and d are both constants and the aim is to obtain the values for them that will satisfy the terminal velocity and the position constraints desired. It is obvious that it is a linear function of time.

The thrust attitude angle is a constant parameter, not a function of time and it is denoted with γ_0 if only terminal velocity constraints (v_{x_f}, v_{y_f}), not height constraint (h_f) are considered. The required ΔV calculation and time required to get that amount of

ΔV are given in equations (3.45) and (3.46), respectively. T_{go} is the required flight time from initial point (i) to terminal point (f) and is calculated iteratively. Its iteration scheme is given in algorithm 2.

$$\Delta V = (v_{x_f} \times T - v_{x_i}^2 + (v_{y_f} \times T - v_{y_i} - g \times T)^2 \tag{3.45}$$

$$T_{go} = \tau \times (1 - e^{\frac{-\Delta V}{v_{ex}}}) \tag{3.46}$$

Algorithm 2 Iterative T_{go} calculation algorithm

$T \leftarrow T_{guess}$ ▷ Begin with a guess for time-to-go

$\varepsilon \leftarrow$ 1E8 ▷ Define error as a large number

$\Delta V \leftarrow (v_x \times T - v_{x_i}^2 + (v_y \times T - v_{y_i} - g \times T)^2$

$T \leftarrow \tau \times (1 - e^{\frac{-\Delta V}{v_{ex}}})$

while $\varepsilon > \varepsilon_{tol}$ **do** ▷ Perform calculation until acceptable tolerance achieved

  $\Delta V \leftarrow (v_x \times T - v_{x_i}^2 + (v_y \times T - v_{y_i} - g \times T)^2$

  $T_n \leftarrow \tau \times (1 - e^{\frac{-\Delta V}{v_{ex}}})$

  $\varepsilon \leftarrow T - T_n$

  $T \leftarrow T_n$

end while

$T_{go} = T$

Finding the correct value for T_{go}, if the equations of motion are integrated once from 0 to T_{go}, the expressions in equations (3.47) and (3.48) are obtained.

$$\dot{x}_{0 \to T_{go}} = \dot{x}_1 + V_{ex} L \cos \gamma_0 \tag{3.47}$$

$$\dot{y}_{0 \to T_{go}} = \dot{y}_1 + V_{ex} L \sin \gamma + g T_{go} \tag{3.48}$$

When equations (3.47) and (3.48) are solved for γ_0, the expression in equation (3.49) is obtained.

$$\tan \gamma_0 = \frac{v_{y_f} + v_{y_i - g T_{go}}}{v_{x_f} - v_{x_i}} \tag{3.49}$$

The thrust attitude is a function of both initial and terminal velocity terms and terminal time as seen in equation (3.49). The small angle assumption can be made for γ_0.

$$\gamma_0 \approx \tan\gamma_0 \tag{3.50}$$

In addition, if the height y_f (altitude) is also needed to be satisfied, the optimum thrust attitude becomes a linear function of time that consists of the previous solution for only velocity constraints and given in (3.51) [18].

$$\tan\gamma = \tan\gamma_0 - (K_1 - K_2 t) \tag{3.51}$$

It is assumed that both γ_0 and $(K_1 - K_2 t)$ are small. Therefore the expression can be rewritten as given in equation (3.52) by assuming they are small.

$$\gamma = \gamma_0 - (K_1 - K_2 t) \tag{3.52}$$

If the expression for γ in equation (3.52) is inserted into equations of motion given in equations (3.41) and (3.42) and they are integrated from initial time to terminal time T_{go}, the expressions given in equations (3.53) and (3.54) are obtained. In equations (3.53) and (3.54), V_{ex} is the exit velocity, S, J, Q are thrust integrals which are given in Table 3.1.

Table 3.1: Thrust Integrals

Expression
$S = -\int_0^{T_{go}} \frac{T}{m(t)}\, dt$
$J = \int_0^{T_{go}} \frac{T}{m(t)} t\, dt$
$Q = -\iint_0^{T_{go}} \frac{T}{m(t)} t\, dt^2$

$$-K_1 V_{ex}\cos\gamma_0 + K_2 J\cos\gamma_0 = 0 \tag{3.53}$$

$$y_i + v_{y_i} T_{go} + \frac{1}{2} g T_{go}^2 - S \sin\gamma_0 + K_1 S \cos\gamma_0 - K_2 Q \cos\gamma_0 - y_f = 0 \qquad (3.54)$$

In equations (3.53) and (3.54), each term is known except for K_1 and K_2. If the system of equations is solved for those two unknowns, the values for K_1 and K_2 are obtained.

3.3.2 3-D Spherical Earth Case

Moreover, this solution can be extended to spherical earth case and three-dimensional space [19]. The spherical assumption for the shape of the Earth is made in this study, and the gravitational acceleration is changing inversely proportional to the distance from the Earth's center. When equations (3.41) and (3.42) are considered, the acceleration term should be added to both equations not only one of them. However, the effect of gravity changes as the rocket moves in space.

$$\mathbf{g} = -\frac{\mu}{|\mathbf{r}|^3}\mathbf{r} \qquad (3.55)$$

The equations of motion for 3-D and varying Earth gravity case are given in equation (3.56)

$$\ddot{k} = \frac{T}{m}\cos\gamma_p \cos\gamma_y + g_k \qquad (3.56a)$$

$$\ddot{l} = \frac{T}{m}\sin\gamma_p \cos\gamma_y + g_l \qquad (3.56b)$$

$$\ddot{m} = \frac{T}{m}\sin\gamma_y + g_m \qquad (3.56c)$$

This leads to the employment of an average term derived from the gravitational acceleration $(\bar{g})$. A new coordinate frame, known as the target frame, is defined at the insertion point for three-dimensional coordinates. The position vector resolved in the target frame is shown as $\mathbf{r}^t$ for example, and this new frame is marked by t as superscripts. Fig. 3.2 shows the defined target frame. The three axes of the target frame are k,l, and m. The kl plane is in the original xy plane, the l is perpendicular to the earth's surface and faces away from the gravitational pull, and the m and k, l axes are

positioned in such a way as to satisfy the right-hand rule. The xyz frame and the klm frame are illustrated in Fig. 3.2. Additionally, xyz frame can be rotated at an angle Γ around z to produce a klm frame, according to [19]. Another thrust attitude angle, γ_y, is defined for the third axis and is determined using the same steps as in the pitch plane.

The angle Γ is unknown as stated beforehand. It is calculated in an iterative way such that, an initial guess for Γ is made and the estimation for the position at the orbital insertion is calculated, the target frame is selected using the estimated position, and Γ is recalculated, and finally the error between the guess and the calculated value the guess is updated with the previous calculation for Γ until the error becomes within the acceptable range.

The coordinate transformation between the two frames mentioned can be done using Direction Cosine Matrix (DCM) which is denoted by $\mathbf{C}$. The matrix that transforms a vector resolved in xyz frame $\mathbf{r^x}$ into the vector resolved in klm frame ($\mathbf{r^t}$) is a 3x3 matrix whose determinant is equal to one and denoted with $\mathbf{C_x^t}$ and its calculation is given in equation (3.57). Examples of similar transformations are given also in .

$$\mathbf{C_x^t} = \begin{bmatrix} \cos\Gamma & -\sin\Gamma & 0 \\ \sin\Gamma & \cos\Gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.57}$$

$$\mathbf{r^t} = \mathbf{C_x^t}\mathbf{r^x} \tag{3.58a}$$

$$\mathbf{r^x} = \mathbf{C_x^t}^{-1}\mathbf{r^t} \tag{3.58b}$$

After defining the klm frame, the initial and final conditions now must be revisited. The initial condition of the rocket is given in the xyz frame whereas the final conditions are given usually in the klm frame. The expression for the final position condition in the target frame is given in equation (3.59). Also, the satellite will have a position component only in the l axis in the new frame since its k and m component must be zero because the new frame is selected. In addition, the satellite will have only a velocity component in the k axis at the orbital insertion considering a circular

orbit. The magnitude of the velocity must be equal to the requirement of that orbit. Therefore, the final condition for the velocity now may be written as given in equation (3.60).

$$\mathbf{r_f^t} = \begin{bmatrix} r_{k_f} \\ r_{l_f} \\ r_{m_f} \end{bmatrix} = \begin{bmatrix} 0 \\ r_{l_f} \\ 0 \end{bmatrix} \tag{3.59}$$

$$\mathbf{v_f^t} = \begin{bmatrix} v_{k_i} \\ v_{l_i} \\ v_{m_i} \end{bmatrix} = \begin{bmatrix} v_{k_i} \\ 0 \\ 0 \end{bmatrix} \tag{3.60}$$

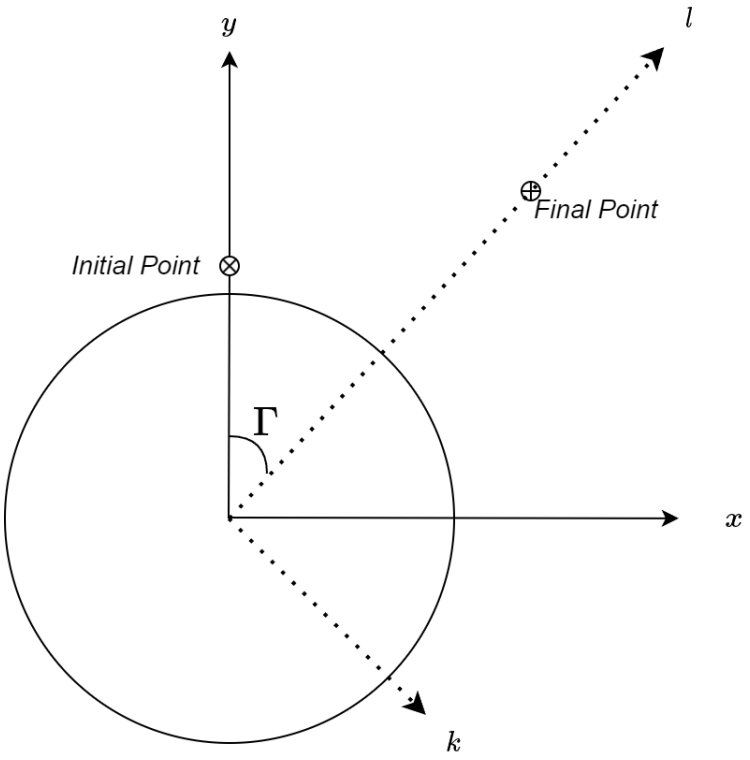


Figure 3.2: xyz and klm Frames

For the solution methodology, the equations of motion are expressed in order to consider the variation in the gravitational acceleration by assuming average gravitational acceleration.

$$\ddot{k} \approx \frac{T}{m}\cos\gamma_p \cos\gamma_y + \bar{g}_k \tag{3.61a}$$

$$\ddot{l} \approx \frac{T}{m}\sin\gamma_p \cos\gamma_y + \bar{g}_l \tag{3.61b}$$

$$\ddot{m} \approx \frac{T}{m}\sin\gamma_y + \bar{g}_m \tag{3.61c}$$

$\bar{g}$ is the gravitational acceleration that varies depending on the position of the rocket as indicated in equations (3.55) in the expressions (3.61a) - (3.61c). The average gravitational acceleration is denoted by the symbols $\bar{g}_k$, $\bar{g}_l$ and $\bar{g}_m$, and its computation is given in the equations (3.62a) - (3.62c).

$$\bar{g}_k = \frac{1}{2}(g_{k_i} + g_{k_f}) \tag{3.62a}$$

$$\bar{g}_l = \frac{1}{2}(g_{l_i} + g_{l_f}) \tag{3.62b}$$

$$\bar{g}_m = \frac{1}{2}(g_{m_i} + g_{m_f}) \tag{3.62c}$$

In the 3-D case, the motion of the rocket can be analyzed by dividing it into two planes, pitch and yaw planes. The motion in the pitch plane will refer to the movement of the rocket in the kl plane whereas the yaw motion will refer to the movement in the km plane defined in the klm frame. Similar to the 2-D case, the thrust attitude angles can be specified as the linear function of time with the unknowns K_1, K_2, K_3, and K_4, and they are represented with γ_p and γ_y, respectively.

$$\tan\gamma_p = \tan\gamma_{p_0} - (K_1 - K_2 t) \tag{3.63}$$

$$\tan\gamma_y = \tan\gamma_{y_0} - (K_3 - K_4 t) \tag{3.64}$$

γ_{0_p} and γ_{0_y} are the constant thrust regime angles for only the velocity constraint case for both planes. K_1, K_2, K_3, and K_4 are to be determined considering both velocity and position constraints in 3-D. First, the values γ_{0_p} and γ_{0_y} are obtained using the same procedure in 2-D case. Then, the equations of motion integrated and corresponding components are equated to desired values the equations (3.65) - (3.68) are

obtained.

$$-K_1 V_{ex} \cos\gamma_{0_p} + K_2 J \cos\gamma_{0_p} = 0 \tag{3.65}$$

$$y_i + v_{y_i} T_{go} + \frac{1}{2}\bar{g}_l T_{go}^2 - S\sin\gamma_0 + K_1 S\cos\gamma_{0_p} - K_2 Q\cos\gamma_{0_p} - y_f = 0 \tag{3.66}$$

$$-K_3 V_{ex} \cos\gamma_{0_y} + K_4 J \cos\gamma_{0_y} = 0 \tag{3.67}$$

$$z_i + v_{z_i} T_{go} + \frac{1}{2}\bar{g}_m T_{go}^2 - S\sin\gamma_{0_y} + K_1 S\cos\gamma_{0_y} - K_2 Q\cos\gamma_{0_y} - z_f = 0 \tag{3.68}$$

The values for K_1, K_2, K_3, and K_4 found by solving the system of equations given in equations (3.65) - (3.68)

The calculation for γ_p and γ_y can be updated during the flight. While updating, the i condition becomes the current state of the rocket. It is important to note that as T_{go} gets smaller the algorithm may cause a problem, thus it is recommended to stop updating the solution after certain time remains to insertion.

CHAPTER 4

RESULTS AND DISCUSSIONS

In this chapter, the application of the methods mentioned is given and the comparison between methods and the optimal solution is given in the first section. In the second section, a further examination of the IGM method for different scenarios is presented. Moreover, a Monte Carlo analysis for the robustness of the IGM methods is performed and its results are presented. There are some scenarios created to test the algorithms and compare their results and in all cases, the target orbit is a circular one. In all scenarios, the start point of the simulation is the same and its latitude (ϕ) and longitude (λ) are given as follows.

$$\phi_0 = 42.3^\circ \tag{4.1a}$$

$$\lambda_0 = 29.5^\circ \tag{4.1b}$$

It is important to note that the first stage burnout occurs at 160 s after the ignition which is followed by 5 s of coast phase. Then, the second stage engine is ignited. The thrust is terminated when desired conditions for the payload are achieved and this is determined by the algorithm itself. In the scenarios, the release altitude of the rocket from the carrier aircraft, release speed, flight path angle (FPA), and heading angle at release differ.

4.1 Comparison of Methods

In this section, the solution of IGM, E guidance, and the optimal solution are given and compared for a circular orbit. The conditions for the test case are given in Table

4.1. In this case, the flight path angle values are taken to be zero which implies that the carrier has no vertical velocity and it lies in the same plane as the orbit.

Table 4.1: Comparison Case Conditions

	Values
Release Altitude, (km)	10
Release Speed, (m/s)	300
Release Flight Path Angle , $(^\circ)$	0
Release Heading Angle, $(^\circ)$	0
Target Altitude, (km)	270
Payload Mass, (kg)	50

4.1.1 Iterative Guidance Mode Application

In the IGM method, the aim is to obtain expressions for the thrust attitude angles under the assumption of vacuum conditions, therefore thrust attitude angles are calculated by disregarding the atmospheric effects whereas simulations are performed considering the atmosphere.

The expression for both thrust attitude angles γ_p, γ_y is in the form of a first-order linear function of time, that consists of a constant and a coefficient term. As a result, the four unknowns namely K_1, K_2, K_3, K_4 are obtained at each solution step. It should be noted that the values for the unknowns are updated each 5 s.

In this method, thrust termination is performed by introducing an error term that is the velocity difference between rocket velocity and the orbit requirements in the klm frame given in equation (4.3).

$$\varepsilon_{v_k} = v_{k_i} - v_{k_f} \tag{4.2a}$$

$$\varepsilon_{v_l} = v_{l_i} - v_{l_f} \tag{4.2b}$$

$$\varepsilon_{v_m} = v_{k_i} - v_{m_f} \tag{4.2c}$$

$$\varepsilon_v = \sqrt{\varepsilon_{v_k}^2 + \varepsilon_{v_l}^2 + \varepsilon_{v_m}^2} \tag{4.3}$$

The thrust is terminated if there is an increase in the velocity difference. Thus, the condition checked for the termination is as follows.

$$\dot{\varepsilon}_v > 0 \tag{4.4}$$

The position and the velocity of the rocket in the klm frame are given in Fig. 4.1 and Fig. 4.2, respectively. It is preferred to present the result in klm frame first since the final conditions are defined in that frame for IGM method.

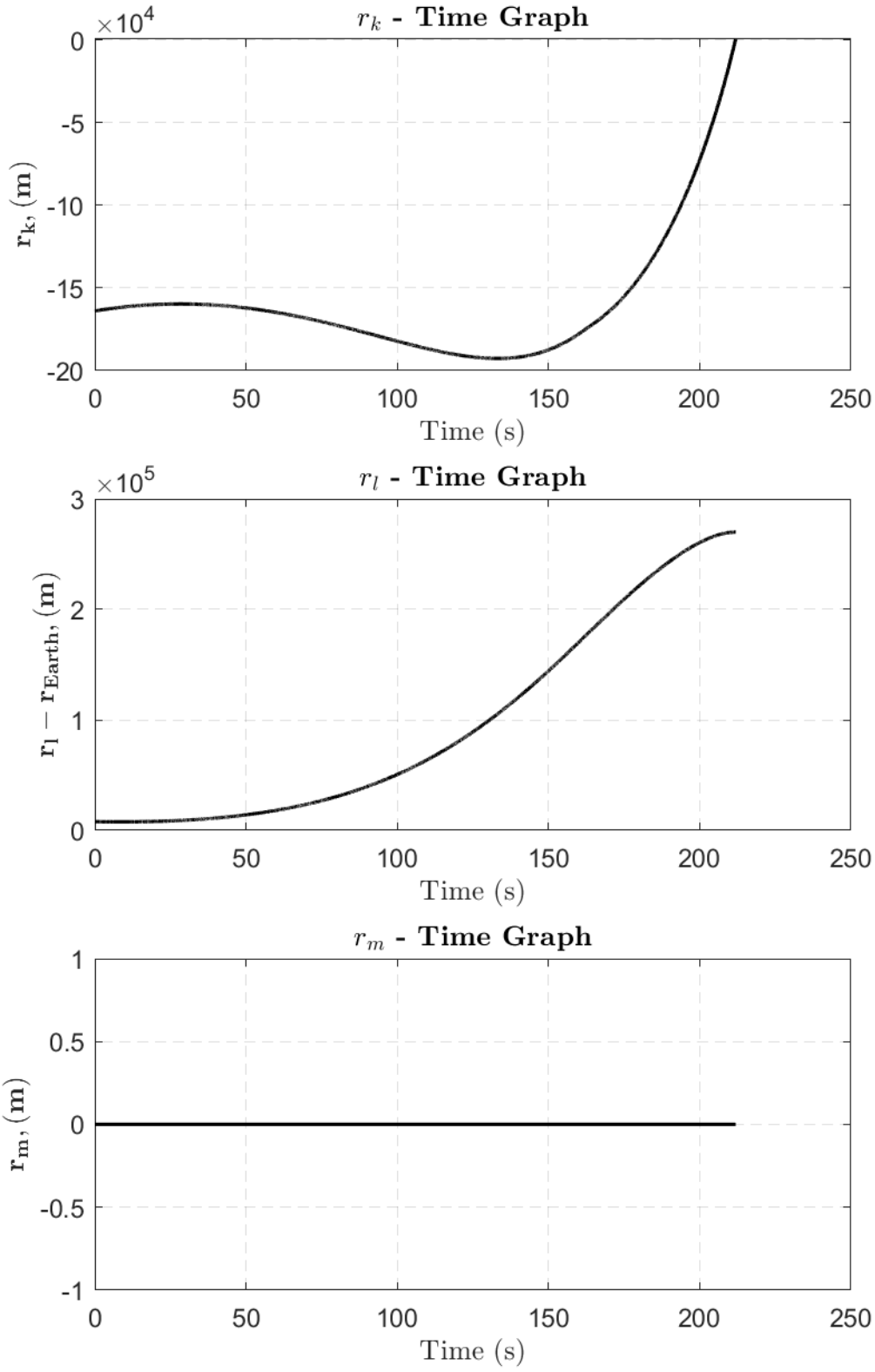


Figure 4.1: Position - Time Graph in the klm Frame, IGM Method

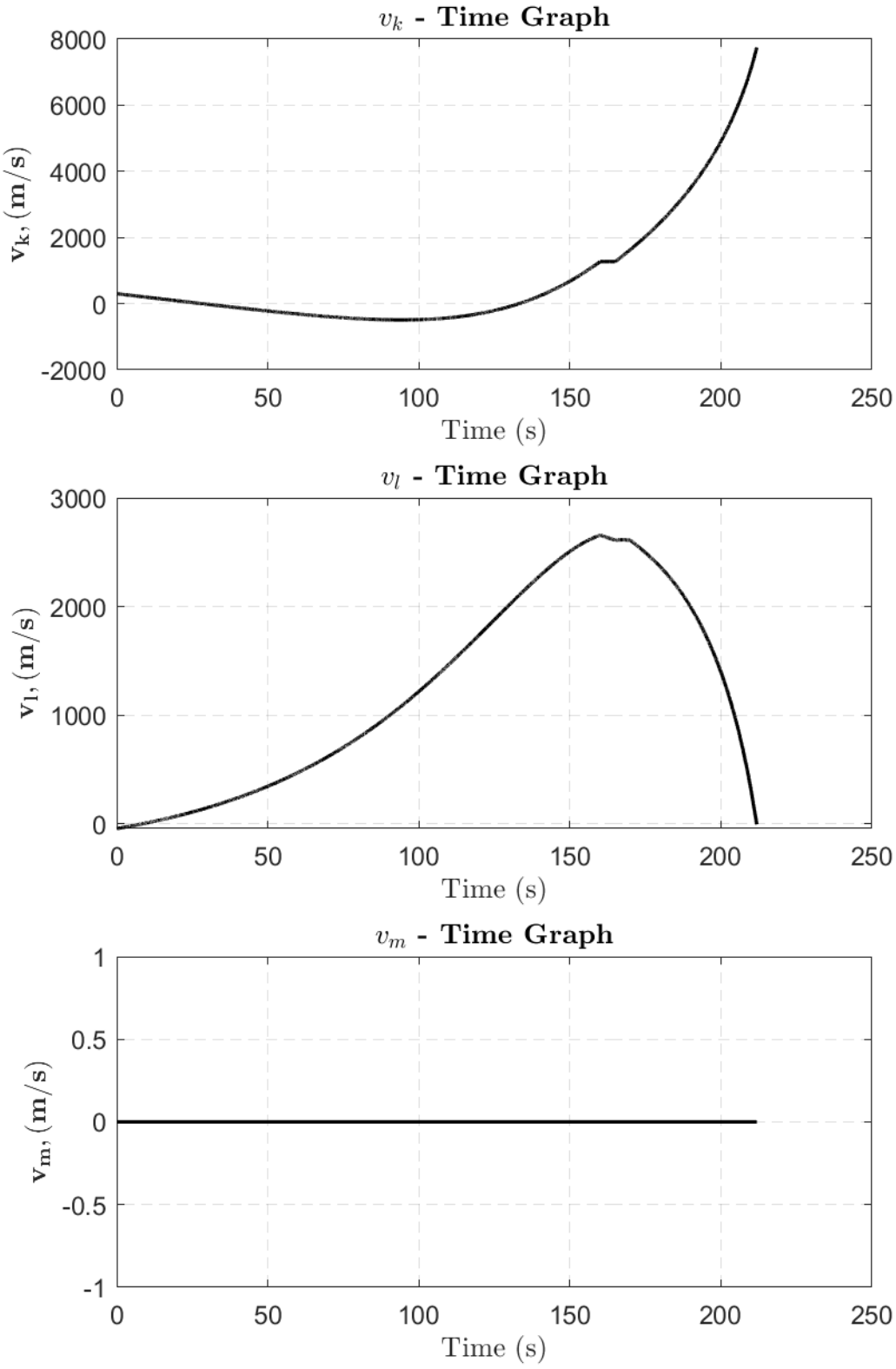


Figure 4.2: Velocity - Time Graph in the klm Frame, IGM Method

It is seen that the position value in the l axis reaches 270 km which is the desired orbit altitude. Also, the position in the m axis is zero during the flight and at the end. It is expected since there is no initial and final value other than zero in the velocity or position. The position in the k axis is not constrained as stated before, and the value of the position in the k axis must be zero since the klm frame is defined in the insertion point. Thus, its value is near zero as can be seen in the figure. It is not equal to exactly zero since there occurs some error in the calculation of Γ and the nature of the process. The error values in klm for both position and velocity are presented in Table 4.2.

Table 4.2: Error Values in the klm Frame, IGM Method

Error	**Value**
ε_{r_l}	-0.23 m
ε_{r_m}	0.0 m
ε_{v_k}	-1.99 m/s
ε_{v_l}	1.42 m/s
ε_{v_m}	0.0 m/s

The position and the velocity of the rocket in the xyz frame are given in Fig. 4.3 and Fig. 4.4, respectively.

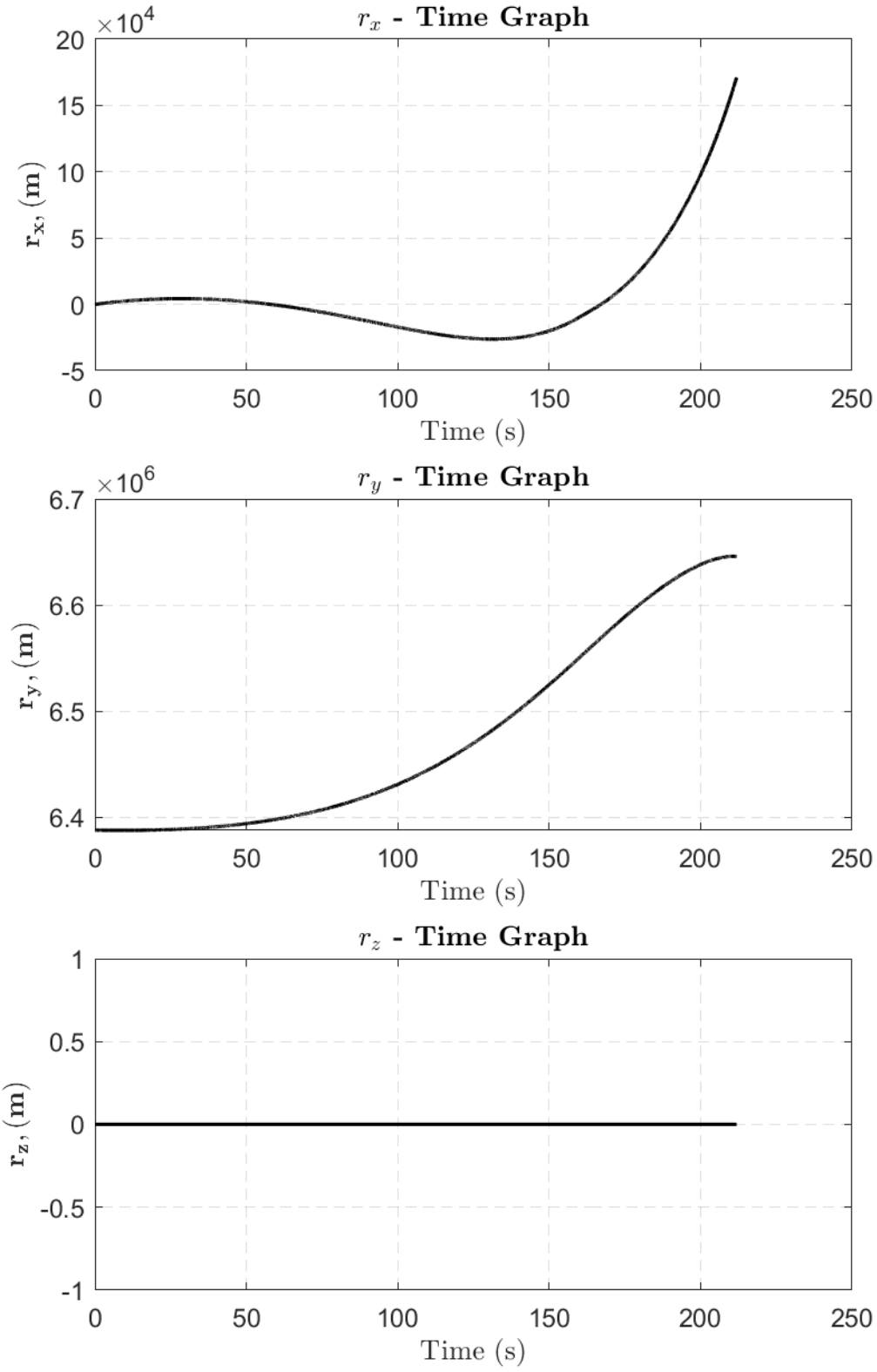


Figure 4.3: Position - Time Graph, IGM Method

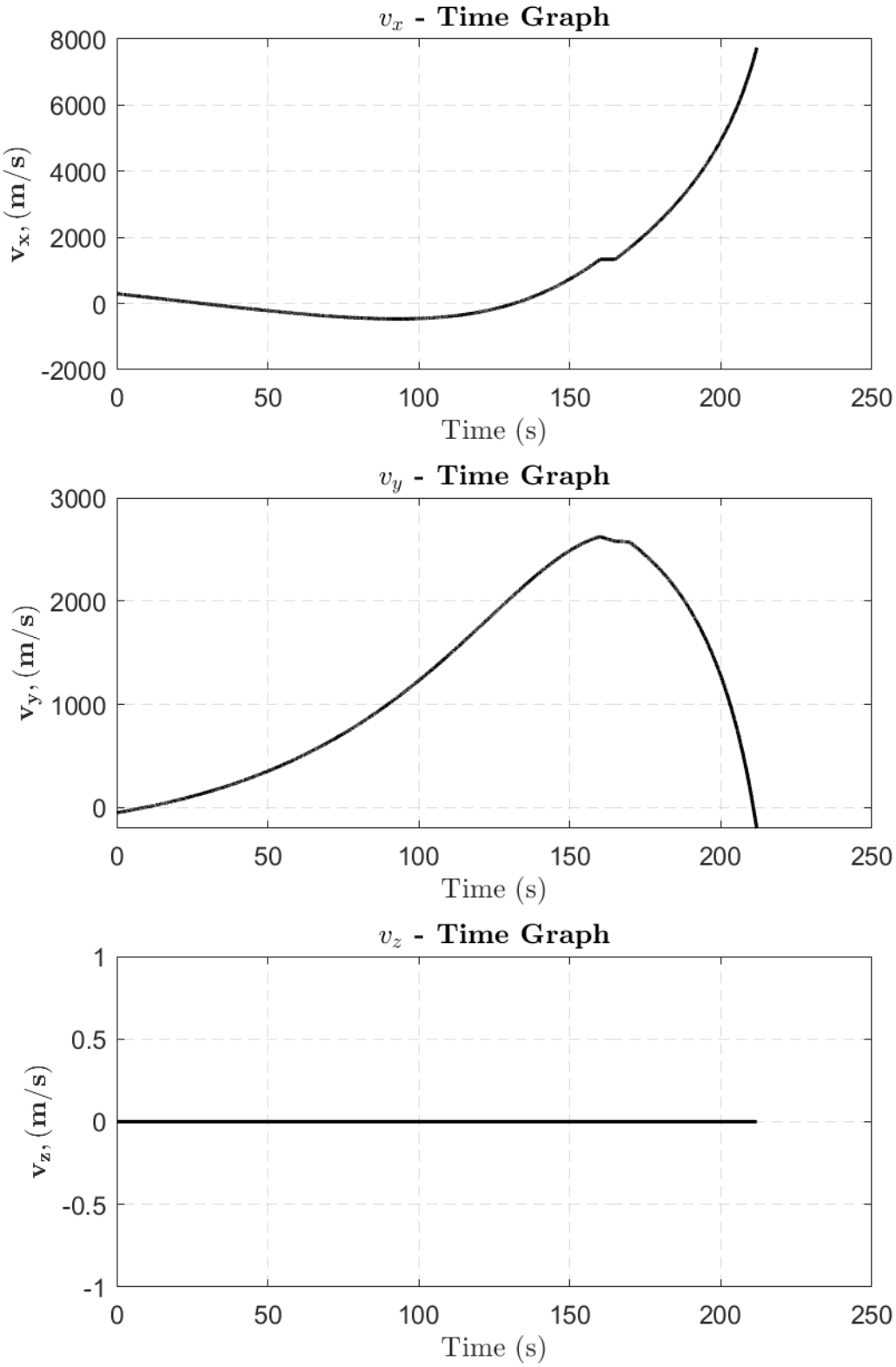


Figure 4.4: Velocity - Time Graph, IGM Method

The values are the ones that went under coordinate transformation between the klm results, therefore they are presented to make the comparison with other algorithms. It can be observed that the satellite inserted the desired orbit successfully using the IGM method roughly in 212 s, which is a fair quick response time. Consequently, the solution using IGM is considered successful since both the position requirements in height and out-of-plane and the velocity requirements in radial, horizontal, and the out of plane directions are met with an acceptable small error.

4.1.2 E Guidance Application

For the E guidance method, the same initial and final conditions are defined as in the IGM case. This algorithm requires a desired time, thus the time of flight in the IGM method is selected to obtain the performance of the E guidance method and compare it to others. In this method, the acceleration is assumed to be a polynomial of time. The polynomials are selected to be the same for each axis and the second polynomial is selected to include the first polynomial for simplicity [16]and they are given as follows.

$$y_1(t) = a_0 + a_1(t_f - t)^1 + a_2(t_f - t)^2 \tag{4.5}$$

$$y_2(t) = y_1(t)(t_f - t) \tag{4.6}$$

Hence, there are three coefficients to be determined. The coefficients of these polynomials are the parameters that affect the optimality of the method, and they can be calculated using brute force [16]. Therefore, the values for them that provide the minimum fuel expenditure (given in equation (3.17)) are found using MATLAB [31], and their value is presented in the Table 4.3.

Table 4.3: Optimal Values of the Coefficients

Coefficient	**Value**
a_0	352.56
a_1	-4.02
a_2	0.117

The values for the coefficients c_i's are found using the optimal values of a_i's and the

boundary conditions. Similar to the IGM method, the calculation for c_i's is repeated through the flight at 1 Hz to achieve better accuracy. Then, the thrust acceleration commands are found and they are applied to the rocket, and simulation is performed. The position and the velocity of the rocket in the xyz frame for the E Guidance method are given in Fig. 4.5 and Fig. 4.6, respectively.

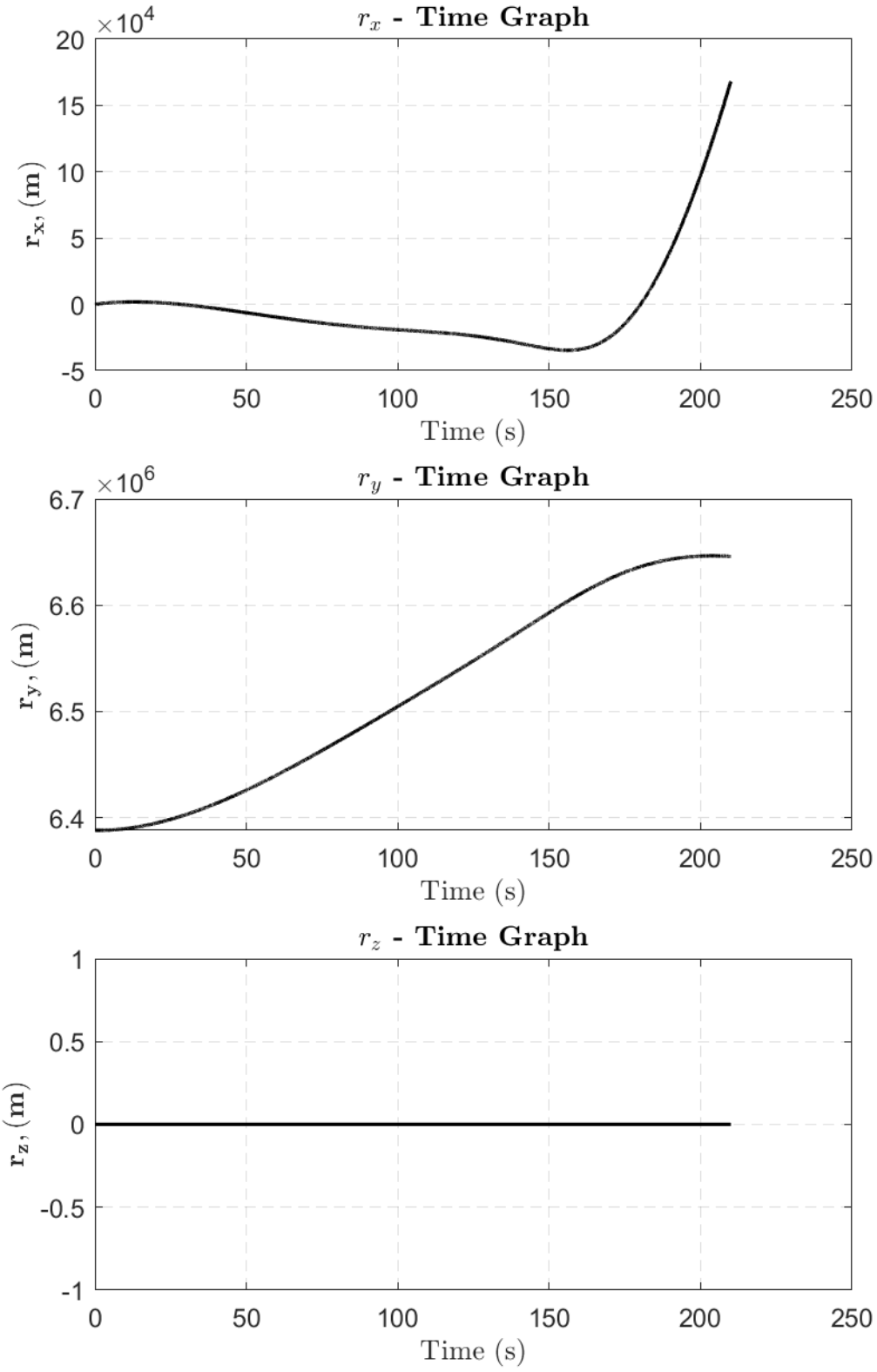


Figure 4.5: Position - Time Graph, E Guidance Method

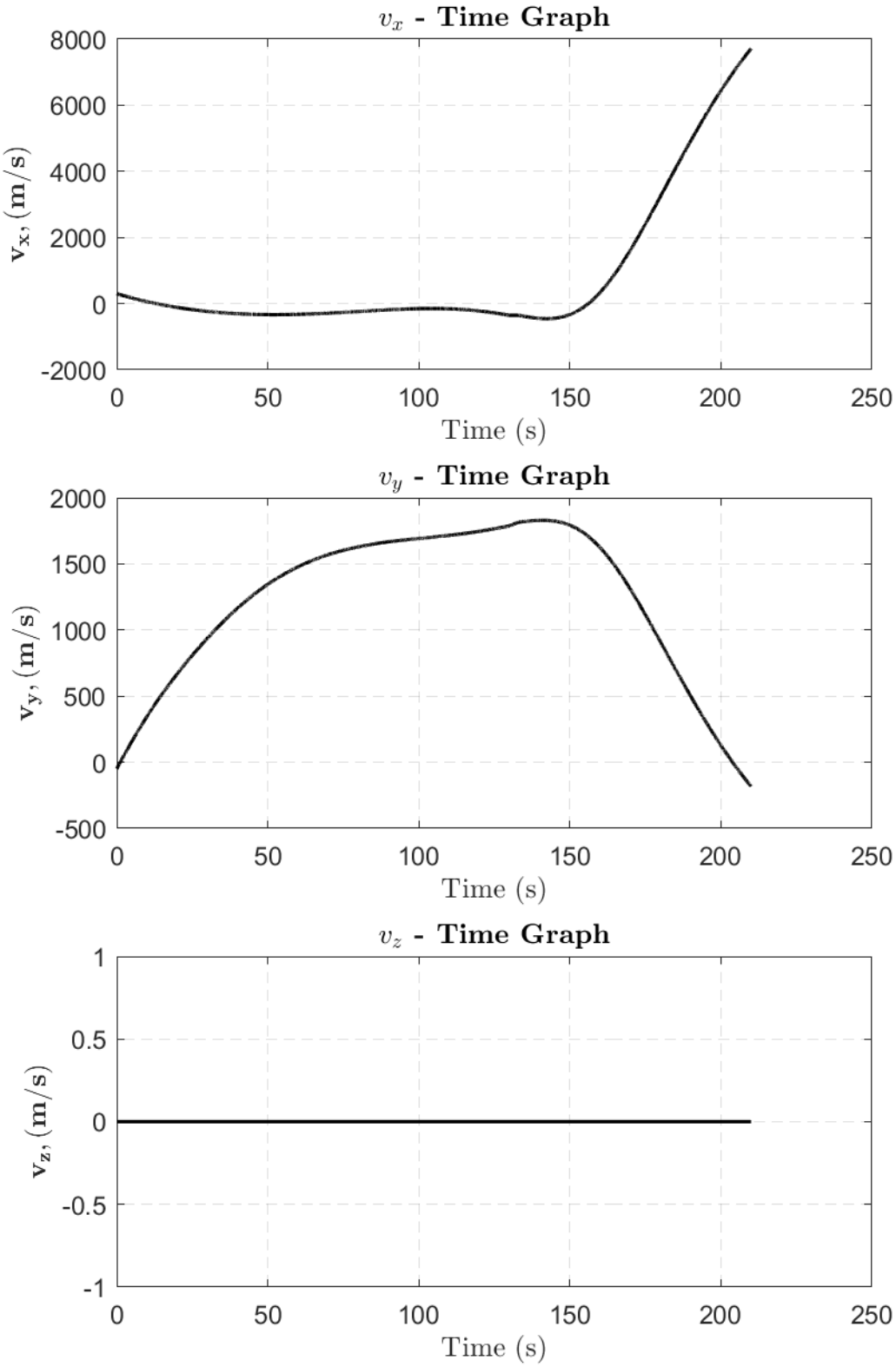


Figure 4.6: Velocity - Time Graph, E Guidance Method

It can be seen that the position and velocity values in each axis reach to desired point in the specified time. It is important to note the behavior is a bit different than the IGM solution. The error value in position and velocity are given in Table 4.4.

Table 4.4: Error Values in Position and Velocity, E Guidance Method

Error	**Value**
ε_{r_x}	2.35 m
ε_{r_y}	-2.62 m
ε_{r_z}	0.0 m
ε_{v_x}	0.02 m/s
ε_{v_y}	-0.41 m/s
ε_{v_z}	0.00 m/s

The error values are small to conclude that the E guidance method works well for the given conditions and successfully guides the rocket to the desired point with the desired velocity in the specified time of flight.

4.1.3 Optimal Trajectory

In this study, the optimal trajectory for the orbital insertion case of the satellite is obtained using FALCON.m software [30], which is an optimal control solver via direct discretization. The flight of the rocket is divided into three phases and each phase is modeled inside the software to find the optimal states and inputs. The first phase of the flight, called Phase 1, is the first stage engine active section. The second phase describes the flight section between the first engine burnout and the second stage of engine ignition. The last phase is Phase 2, where the second stage engine is ignited at the beginning and the orbital insertion occurs at the end of this phase.

States of the dynamical system were mentioned in the 3.1, and they are the position, velocity, and mass of the rocket. The control inputs are γ_p and γ_p angles, that are thrust attitude angles. The aim is to find both optimal states and optimal control

inputs that minimize the fuel spent. The initial velocity of the rocket is not zero in y-axes due to ignition occurring 5s after the release from the carrier aircraft free fall. The optimal states belonging to the rocket with the minimum time are given in Fig. 4.7 and Fig. 4.8.

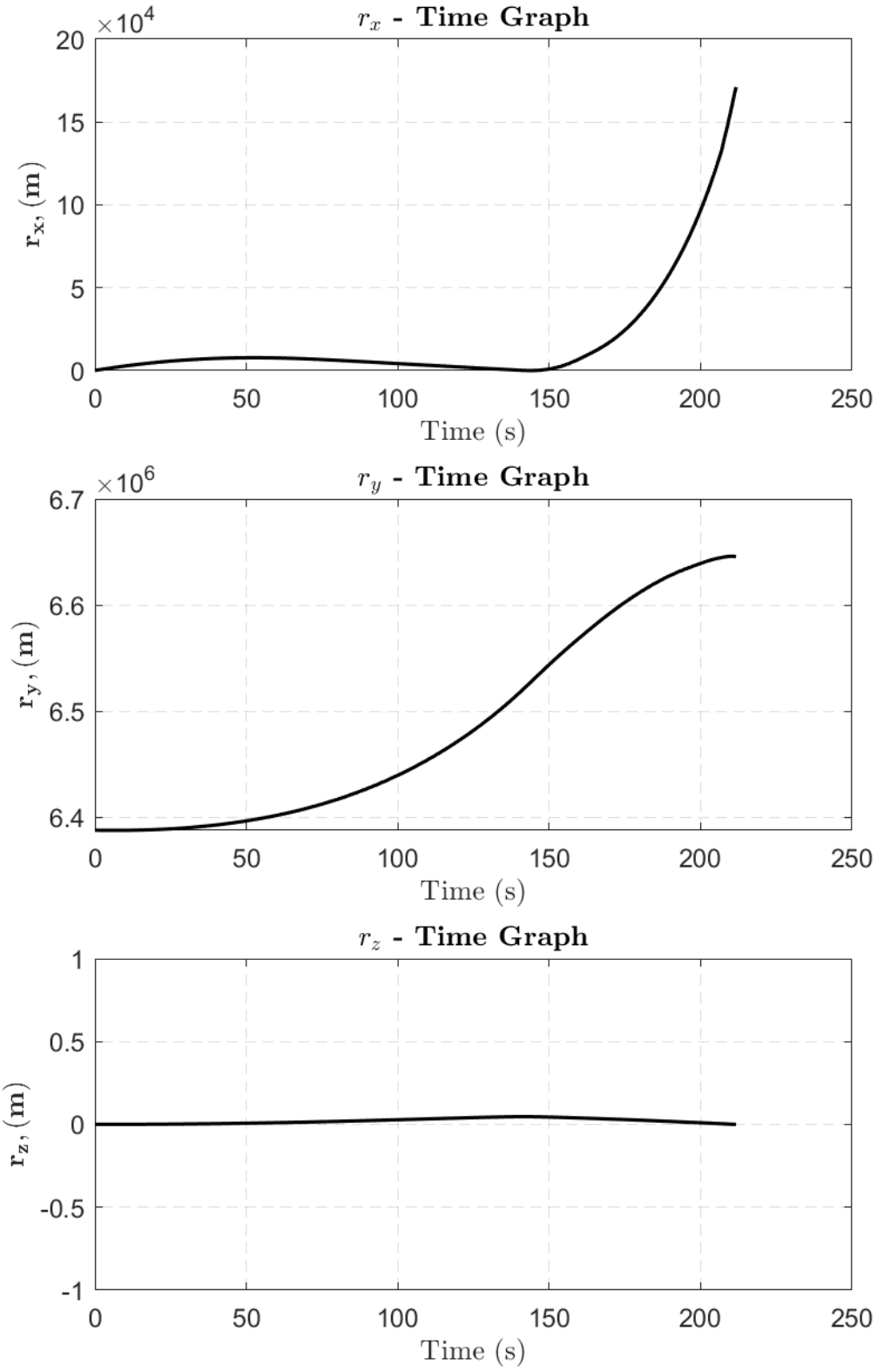


Figure 4.7: Position - Time Graph, Optimal Solution

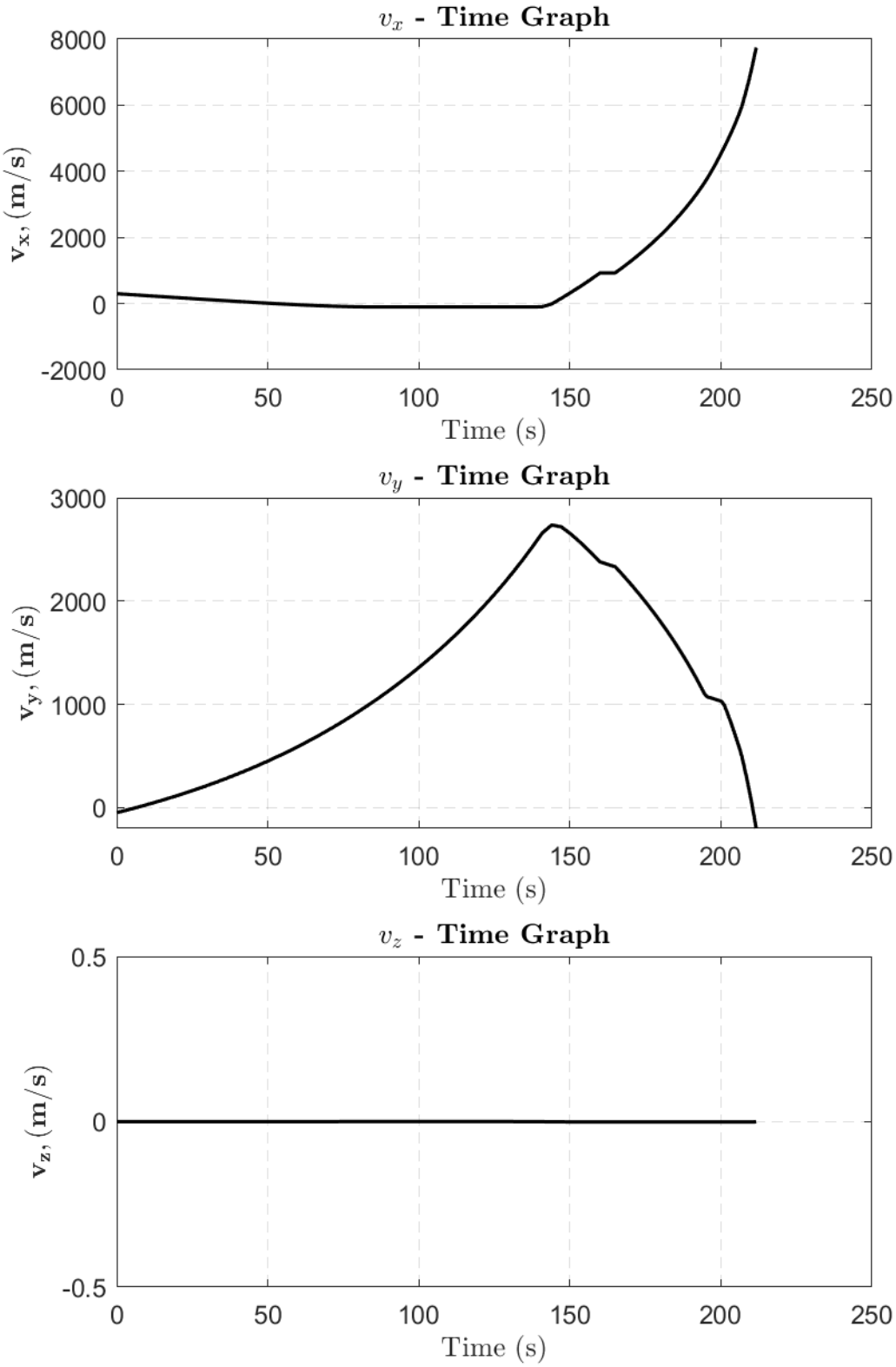


Figure 4.8: Velocity - Time Graph, Optimal Solution

It is seen that the position and velocity in each component in inertial axes reach the desired value at 212 seconds. Since the solution is optimal, it is expected to reach almost to exact desired points, and it is as expected. Thus, no error values are given for the optimal solution. As a result, the optimal trajectory of the rocket for the same initial and end conditions as the IGM and E guidance method is obtained.

4.1.4 Comparison of Methods

There are three trajectories obtained for the same end condition, and each of them shows a difference in their path in addition to a small time difference between the optimal trajectory and IGM, where the optimal solution reaches in a slightly shorter time. Therefore, their components in position and velocity vectors in the inertial axes are compared, and this comparison is given in Fig. 4.9 and 4.10, respectively.

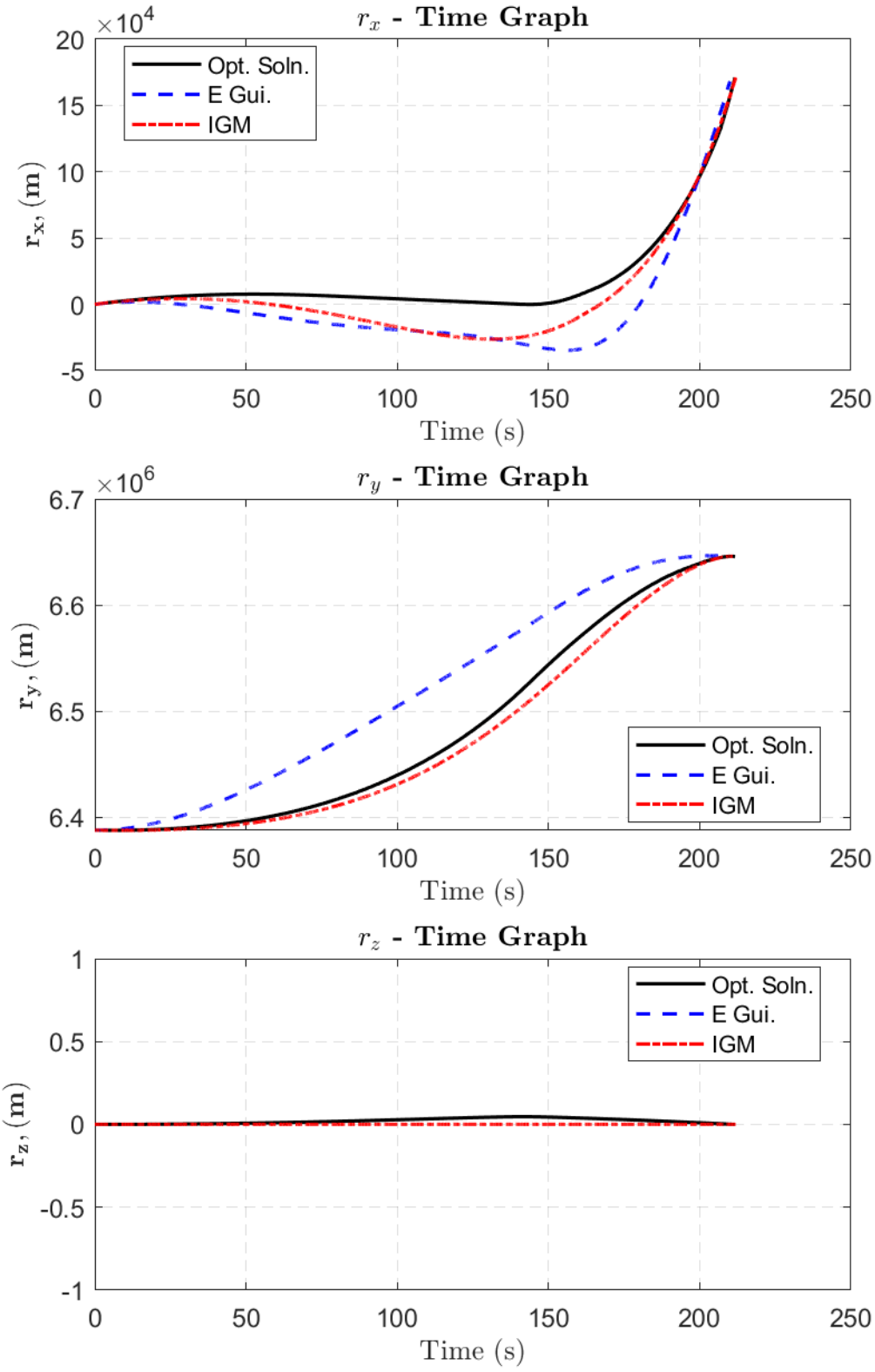


Figure 4.9: Position - Time Comparison

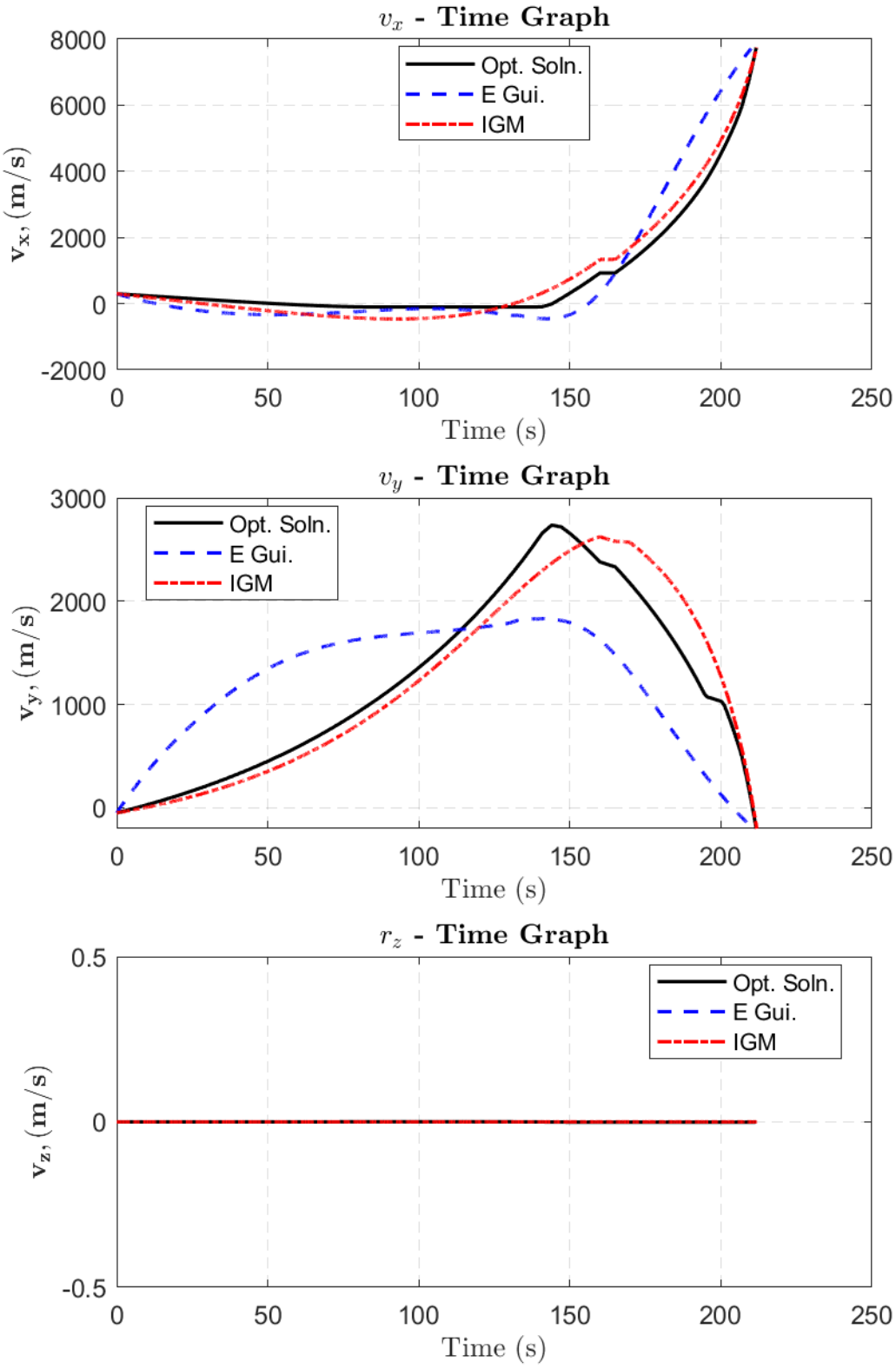


Figure 4.10: Velocity - Time Comparison

It can be accepted that all methods reach the same point with the same velocity although there exists a small difference in their final values. However, the path followed is different for each method, especially for the E Guidance method. The E guidance follows a very different path than the others, whereas the IGM solution and optimal solution seem to follow the same way in the y-axis. On the other hand, in the x-axis, all methods follow different paths, and since there is no movement in the z-axis it is the same for three of them. The E Guidance method gains speed in the y-axis aggressively compared to the IGM solution and the optimal one, however, its maximum value is lower than the other methods. In all methods, the coast phase between $160.s - 165.s$ is obvious in velocities although it is not possible to observe it in positions. Since there is no thrust coming from the engine at that period, no considerable change in velocity is observed. After the second engine ignition, the velocity starts to change and the rocket makes its maneuver the insert the payload.

Table 4.5: Cost Comparison of the Methods

Method	**Fuel Expenditure (kg)**	**Energy Exerted by the Engine (MJ)**
IGM	6206.4	24.354
E Guidance	5975.6	23.448
Optimal Solution	6200.3	24.330

The considerable difference between the E guidance methods and the others is due to the fact that thrust varies. In the applied method of E guidance, thrust is varied as stated beforehand. In the others, the amount of thrust is fixed and it cannot be throttled, thus the thrust acceleration changes only due to the change of mass. The costs of each method are presented in Table 4.5. It is seen that minimal fuel consumption and energy spent occurs in the E guidance method, whereas the values are similar for IGM and the optimal solution. Although the E guidance provides the minimum fuel expenditure, its thrust requirement may violate the engine operating range however, this is not possible in the IGM method. The IGM method is close to the optimal solution compared to the E Guidance method, and their flight time is reasonably close to each other, which implies that amount of the fuel burned is similar. This fact makes

the IGM a potential algorithm for an airlaunch rocket.

4.2 Further Examination of the Iterative Guidance Method

In this section, the IGM algorithm is tested in two other scenarios to investigate its performance under different release and target conditions. In addition, a Monte Carlo analysis is performed to examine its robustness.

4.2.1 Examination of IGM for Different Conditions

In section 4.1, the IGM algorithm is tested under a scenario whose target altitude is 270 km. To evaluate its performance, two more scenarios are considered. The details of them are given in Table 4.6.

Table 4.6: IGM Test Scenarios

	Scenario-1	**Scenario-2**
Release Altitude, (km)	10	12
Release Speed, (m/s)	300	640
Release Flight Path Angle , $(^\circ)$	0	10
Release Heading Angle , $(^\circ)$	9.5	5
Target Altitude, (km)	230	250
Payload Mass, (kg)	40	30

4.2.1.1 Scenario-1

This scenario is created to test the algorithm in case of an out-of-plane velocity error and for a lower altitude with a smaller payload when the release speed is subsonic. Out-of-plane component means that the aircraft hence the rocket's initial velocity plane is not the same as the desired orbital plane. It is aimed that the IGM steer the rocket into a lower altitude orbit in case of an out-of-plane velocity term. The position

and the velocity of the rocket in the klm frame are given in Fig. 4.11 and Fig. 4.12, respectively.

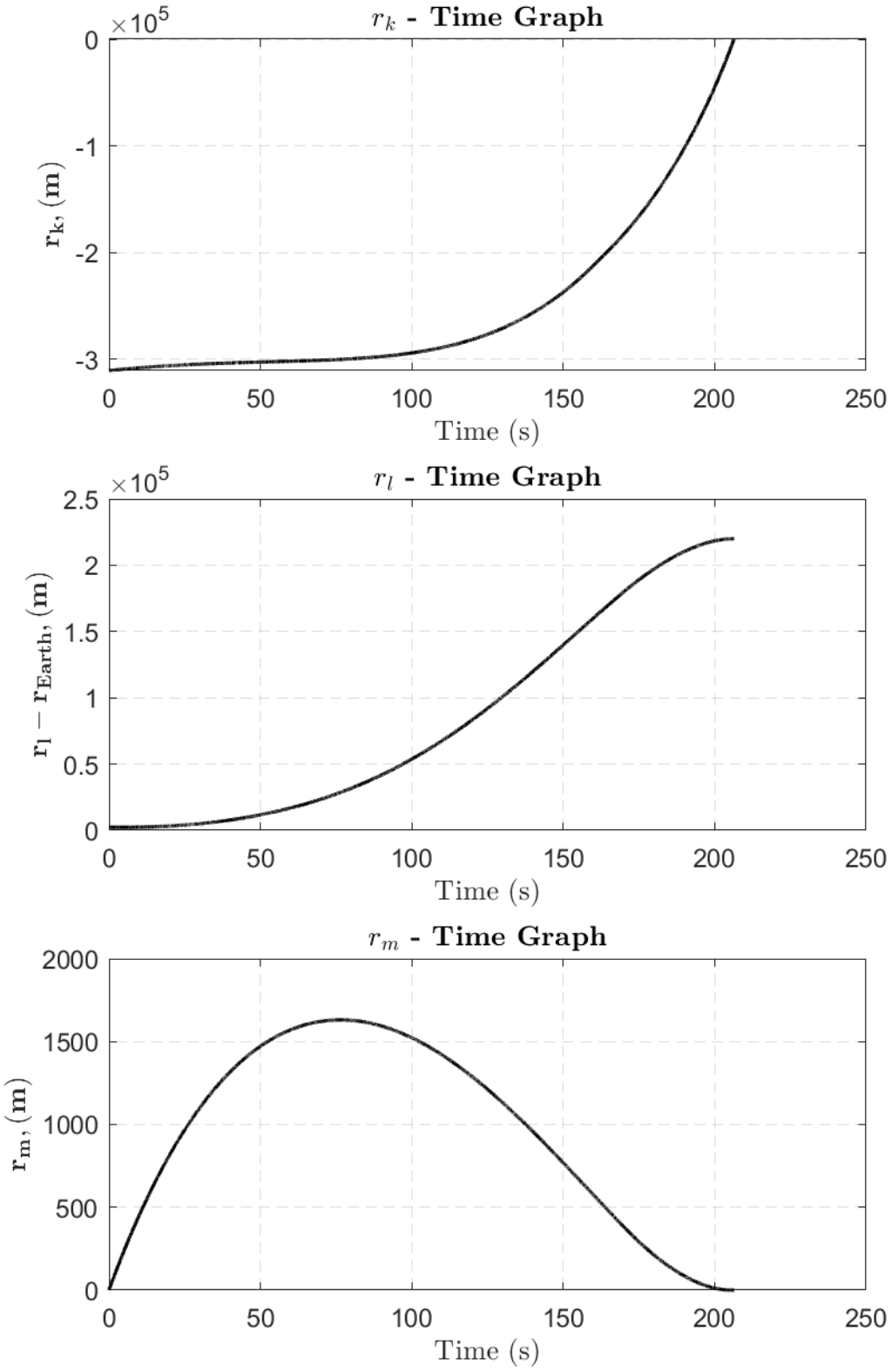


Figure 4.11: Position - Time Graph in the klm Frame, IGM Method

It can be seen that the height in the l axis reaches the desired value of 230 km at the

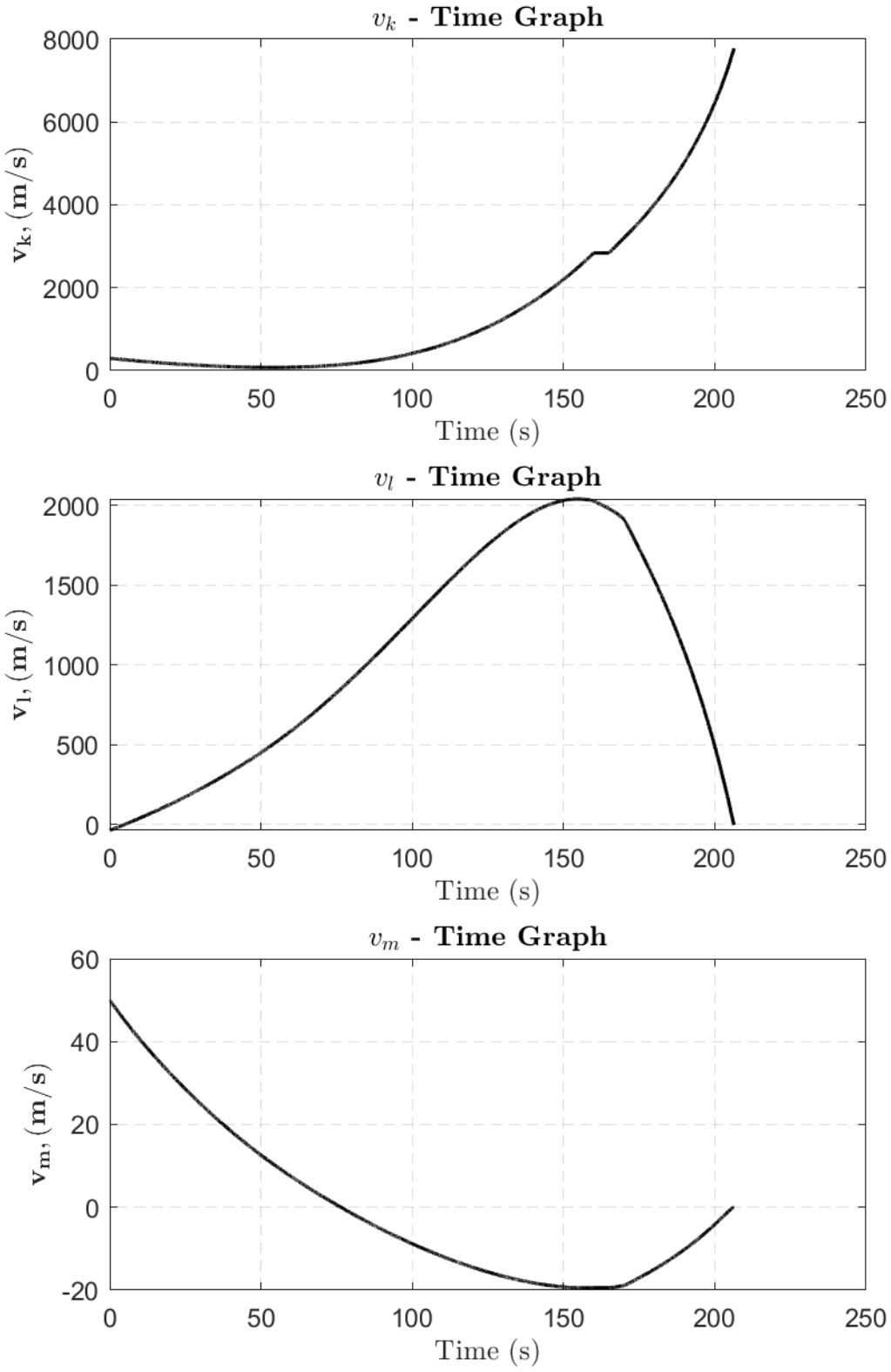


Figure 4.12: Velocity - Time Graph in the klm Frame, IGM Method

end of the flight, and the position in the k axis is close to zero as expected. In this scenario, the initial velocity in the z-axis is not zero however, it is required to have zero velocity at the insertion point (note that z and m axes are the same. From the $v_m - Time$ plot, the motion starts with 50 m/s and it reaches -20 m/s at the coast phase. Then, the second stage engine is fired and the velocity is corrected to zero. The same behavior can be seen in the $r_m - Time$ plot, which increases to approximately 1.5 km due to the initial velocity component and then goes back to zero as desired. Moreover, the error values at the end of the flight are presented in Table 4.7.

Table 4.7: Scenario-1 Error Values in the klm Frame

Error	**Value**
ε_{r_l}	-0.4 m
ε_{r_m}	0.1 m
ε_{v_k}	-1.61 m/s
ε_{v_l}	3.13 m/s
ε_{v_m}	0.02 m/s

The error term in both position and velocity in the m axis is not zero anymore but close to zero. All remaining error terms are close to zero and within an acceptable range, thus it can be concluded that the IGM works well in a subsonic release where the flight path angle is not zero in the yaw axis.

4.2.1.2 Scenario-2

This scenario is created to test the algorithm in a supersonic release condition with a flight path angle in both pitch and yaw axes and the release occurs at a supersonic speed. In this case, a carrier aircraft with the ability to release the rocket in inclined attitude ($\approx 10^\circ$) at 12 km is assumed, and the performance of the IGM is investigated. It is aimed that the IGM steer the rocket into an orbit at 250 km altitude in case of supersonic release conditions whose flight path angle differs from zero in both axes. The position and the velocity of the rocket in the klm frame are given in Fig. 4.13

and Fig. 4.14, respectively.

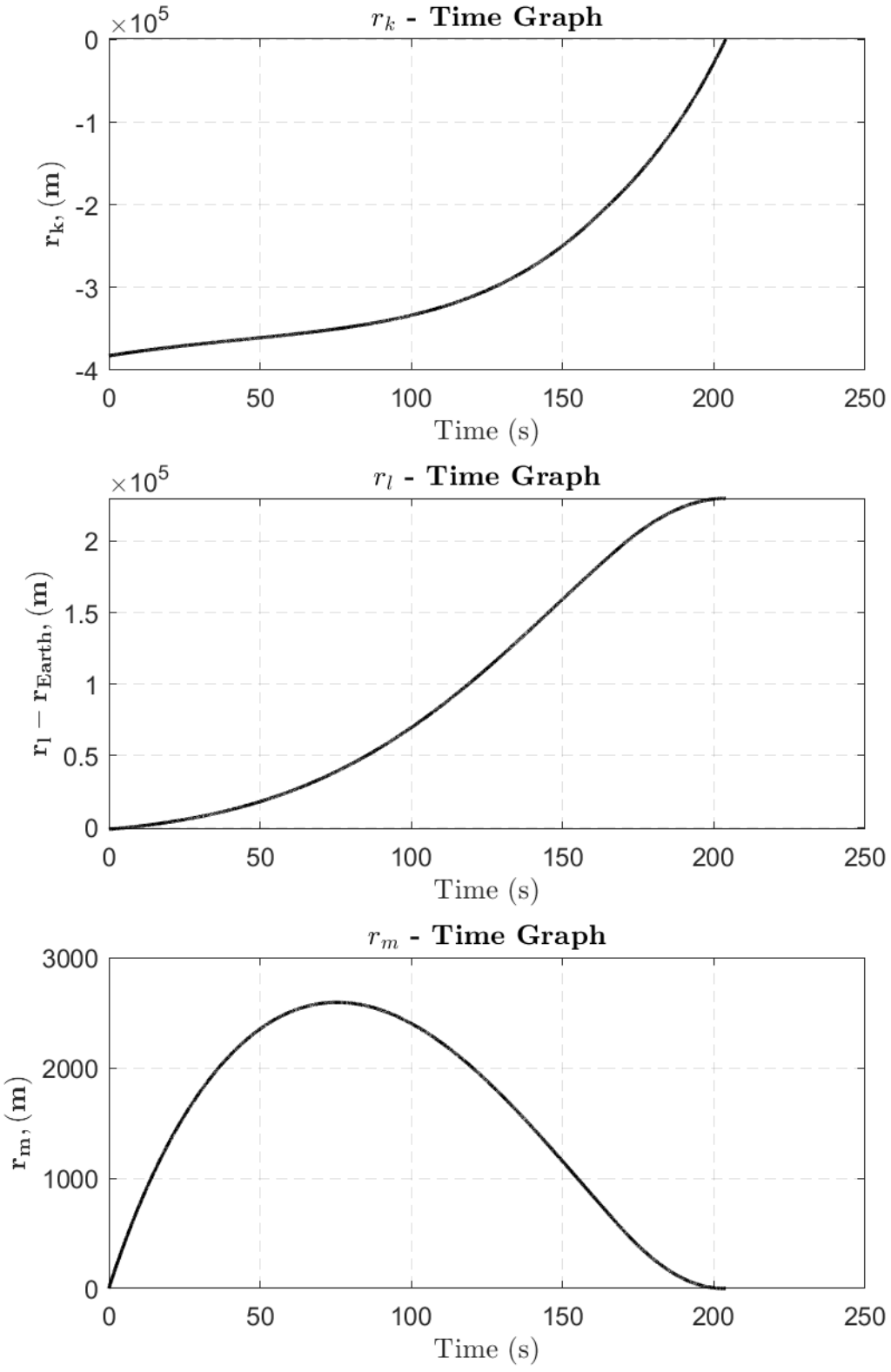


Figure 4.13: Position - Time Graph in the klm Frame, IGM Method

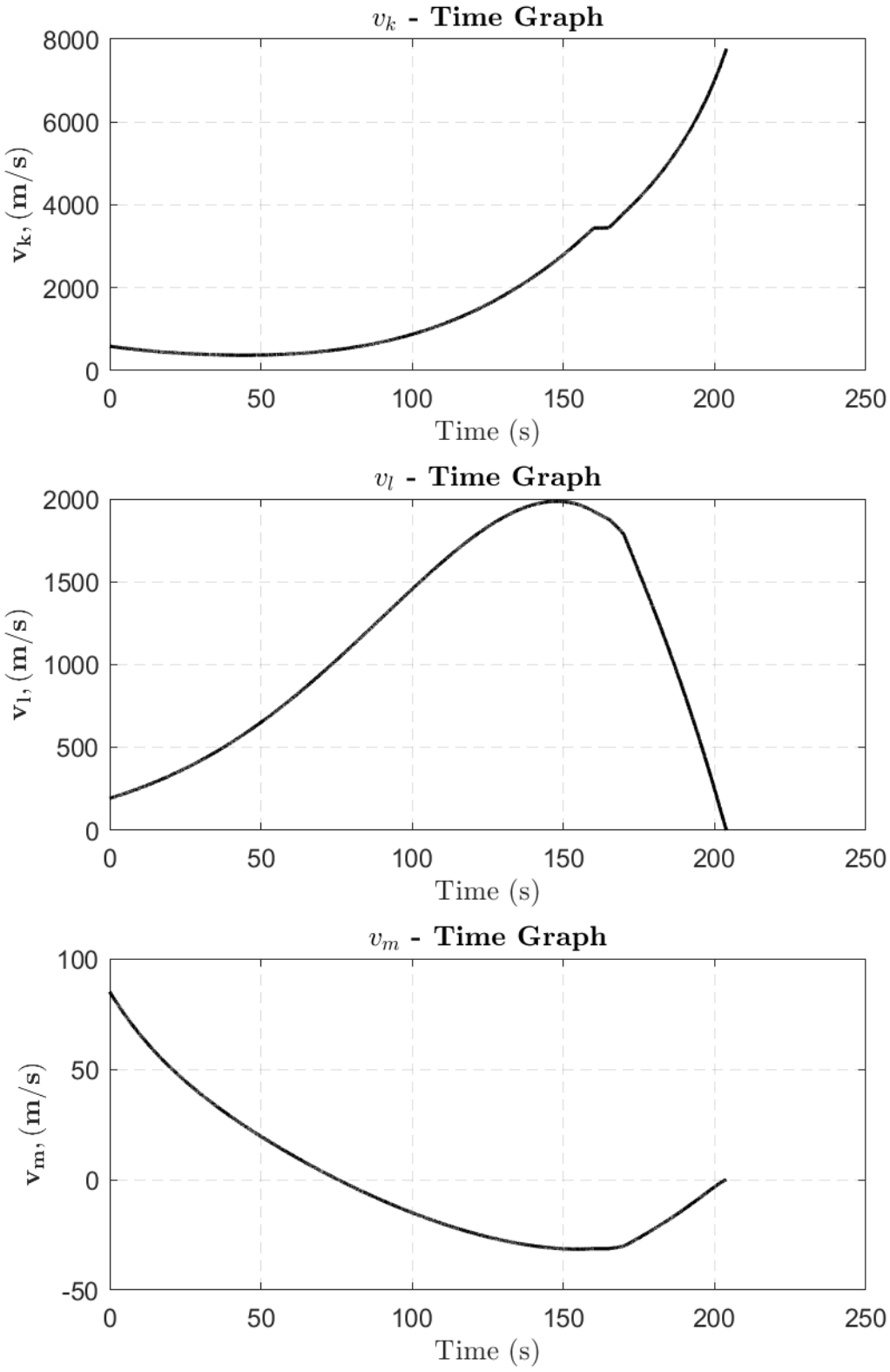


Figure 4.14: Velocity - Time Graph in the klm Frame, IGM Method

It can be seen that the height in the l axis reaches the desired value of 250 km at the end of the flight, and the position in the k axis is close to zero as expected. The position component in the m axis also reaches its required value. The initial velocities in both l and m axes are not zero although both of them must be equal to zero at the insertion. It is seen that both terms reach zero, and the velocity in the k axis reaches the desired value as the orbit requires. Moreover, the error values at the end of the flight are presented in Table 4.8.

Table 4.8: Scenario-2 Error Values in the klm Frame

Error	**Value**
ε_{r_l}	-0.26 m
ε_{r_m}	0.02 m
ε_{v_k}	-2.74 m/s
ε_{v_l}	2.48 m/s
ε_{v_m}	0.09 m/s

The error terms in both position and velocity in all axes are very close to zero. The order of magnitude is higher in the k and l axes since the real challenge is in the pitch plane. Hence, it can be concluded that the IGM works well in a supersonic release where the flight path angle in the yaw and pith plane is not zero.

4.2.2 Monte Carlo Analysis

The IGM methods work well under the knowledge of rocket parameters exactly. However, there are some uncertainties and measurement errors in the rocket and its subsystems and there might be small differences in release conditions due to the operation of the aircraft. A Monte Carlo analysis, which is a statistical method to search for the robustness of the algorithm in the presence of uncertainties in the system, is performed to test the algorithm's robustness. The scenario conditions are given in 4.9, and the probability distributions are given to the rocket parameters and release conditions, and they are given in table 4.10.

Table 4.9: Monte Carlo Analysis Scenario

	Values
Release Altitude, (km)	10
Release Speed, (m/s)	300
Release Flight Path Angle , $(^\circ)$	0
Release Heading Angle , $(^\circ)$	0
Target Altitude, (km)	220
Payload Mass, (kg)	50

Table 4.10: Uncertainty Values at Initial Conditions

Parameter	**Distribution Type (Uniform / Normal)**	**Mean Value**	**Interval or Standard Deviation**
$m(kg)$	N	6500	10
$I_{sp}(m/s)$	N	405	5
$\dot{m}(kg/s)$	U	30	[-2, +2]
$x(m)$	U	0	[-200, +200]
$y(km)$	U	10	[-0.2, +0.2]
$z(m)$	U	0	[-200, +200]
$v_x(m/s)$	U	300	[-30, +30]
$v_y(m/s)$	U	-50	[-30, +30]
$v_z(m/s)$	U	0	[-30, +30]

Using the distribution for the uncertainties, an analysis with 500 runs is performed. In each run, the values in the simulation are replaced with the values in the uncertainty distribution. It is important to note that there is no feedback for the rocket parameters in the simulations, i.e., the rocket mass or mass flow rate is not estimated and fed to the algorithm. As a result, the flight time, and error values in both position and velocity are recorded, and their distribution is given in Fig. 4.15.

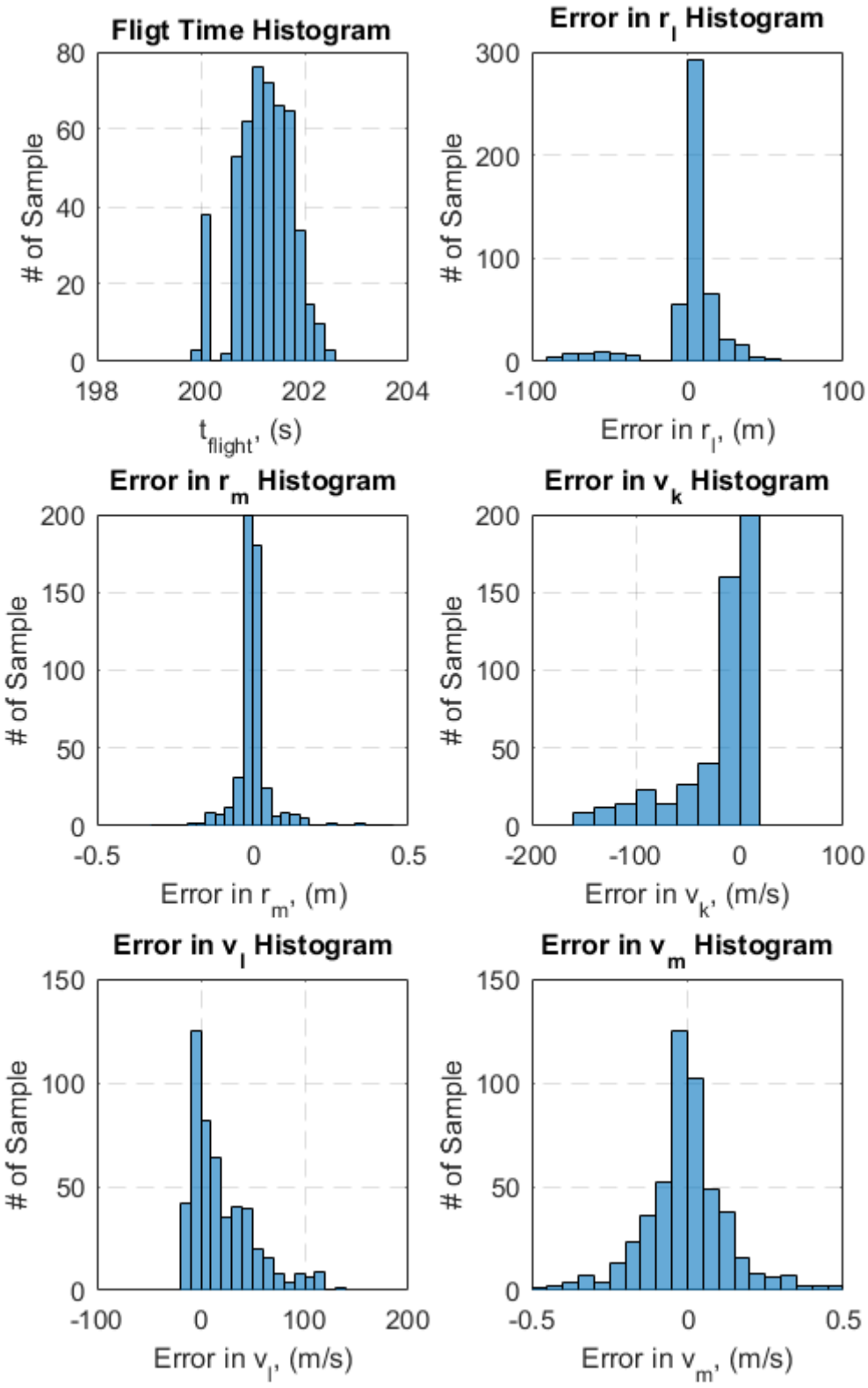


Figure 4.15: Monte Carlo Analysis Results

When the results are analyzed, the time of flight ranges between 200 - 203 s, which is a quite good interval. The error terms in the m axis are close to zero between -0.5 and 0.5 m for the position, whereas -0.5 m/s and 0.5 m/s for the velocity. Since the major challenge is not in the yaw plane, the small values are obtained as expected. The velocity error in the k axis (radial velocity) is quite high and between -180 m/s and 20 m/s. The effect of the uncertainty is shown in that, its value was below 5 m/s for the nominal cases. It is due to the algorithm using the values of specific impulse, mass, and the mass flow rate. However, its interval is quite acceptable and bounded. For the height plane, the l frame, the position errors range from -100 m to 100 m whereas the velocity errors range from -30 m/s to 140 m/s. Both intervals show that the error values in the presence of uncertainties differ from the nominal case and reach to order of hundreds of meters. Although it is higher than the nominal case, the interval can be considered acceptable and bounded.

A correlation analysis between the uncertainties in the parameters at release conditions and the outputs of the Monte Carlo analysis, the correlation of each term is given in Table 4.11. It is seen that the time of flight is affected by the initial mass and the initial horizontal speed. The flight time increases as the mass of the rocket at release increases whereas it decreases if the initial horizontal speed is increased. It is seen that the error in position in the radial direction or in the l axis and the error in the velocity in the k axis from Fig. 4.15. When those parameters are considered, both of them are heavily related to the initial mass value at the release. Lastly, there is no correlation is observed between the release condition values and the velocity error in the radial direction or the l axis.

Table 4.11: Correlation Matrix

	Time of Flight	ε_l	ε_m	ε_{v_k}	ε_{v_l}	ε_{v_m}
m_0	0.76	0.52	0.09	0.64	-0.08	0.09
I_{sp}	-0.20	-0.08	0.05	-0.01	-0.03	0.04
$\dot{m}$	0.04	0.04	-0.03	0.00	-0.03	-0.02
x_0	-0.10	-0.05	0.06	0.02	-0.07	0.06
y_0	-0.01	0.06	-0.05	0.04	-0.05	-0.04
z_0	0.13	0.05	0.03	0.09	-0.06	0.04
v_{x_0}	-0.60	-0.20	0.08	-0.02	0.08	0.08
v_{y_0}	-0.02	-0.02	0.05	0.02	-0.04	0.05
v_{z_0}	0.01	-0.05	0.05	0.03	-0.01	-0.06

There is a trend increase in error in the presence of uncertainties, and this is because the IGM method uses some rocket parameters while calculating the thrust attitude angles. Therefore, any deviation from reality in those parameters causes some error in the final position and velocity. It should be noted that there will be additional sources of uncertainty and some dynamics for the subsystems. Therefore, a solution to that might be the estimate those parameters during the flight by using other algorithms. Consequently, the IGM method handles the uncertainties with an acceptable performance as given in the Monte Carlo analysis.

CHAPTER 5

CONCLUSION

This thesis has dealt with the problem of orbital insertion for small satellites using the concept of airlaunch. The flexibility and quick nature of the airlaunch makes it desirable for especially small satellites.

Firstly, the preliminaries related to orbital mechanics, coordinate frames, rockets, and their subsystems have been mentioned. A hypothetical airlaunch vehicle is considered and its properties such as mass, aerodynamic drag coefficient, thrust values, etc. are presented.

Then, the trajectory optimization problem of an airlaunched rocket is introduced as an optimal control problem, and third-party software is used to obtain optimal trajectory. After this, the methods that have been developed and readily exist in the literature such as E guidance and Iterative Guidance Mode are explained and their application to the rocket is performed, after which the evaluation of these algorithms is performed by comparing them to the optimal solution obtained. Since the developed algorithm based on the E Guidance method is a varying thrust algorithm, it may create a problem when the thrust requirement is above or below what the liquid rocket engine can deliver despite its throttling ability. It is observed that the IGM method works well and its results are similar to optimal ones in terms of fuel consumption. Also, it is a fixed thrust algorithm that does not require any ability to alter the thrust of the engine. It also handles the change in release conditions and target conditions. However, errors in horizontal and radial velocity arise due to uncertainties in the rocket parameters such as initial mass, mass flow rate, and specific impulse since this method directly uses those values to calculate time-to-go and the thrust attitude angle. Although the error magnitudes are not large, it might be a problem when the other sources of error

that are not modeled in this study but exist in real life are considered.

In conclusion, two different guidance algorithms have been implemented in an air-launch rocket to deploy small satellites into LEO orbit, and compared to the optimal trajectory. It has been concluded that the guidance system based on the Iterative Guidance Mode is a potential candidate algorithm for an airlaunch rocket.

REFERENCES

[1]P. Murugan and Y. Agrawal, "Small satellites applications, classification and technologies", *International Journal of Science and Research (IJSR)*, vol. 9, no. 7, pp. 1682–1687, 2020. DOI: 10.21275/SR20723213825.

[2]"Types of orbits". (2020), [Online]. Available: https://www.esa.int/Enabling_Support/Space_Transportation/Types_of_orbits.

[3]G. D. Mauro, M. Lawn, and R. Bevilacqua, "Survey on guidance navigation and control requirements for spacecraft formation-flying missions", *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 3, pp. 581–602, 2018. DOI: 10.2514/1.G002868.

[4]G. Willmer, *In booming satellite market, micro-rockets are the next big thing*, 2023. [Online]. Available: https://ec.europa.eu/research-and-innovation/en/horizon-magazine/booming-satellite-market-micro-rockets-are-next-big-thing.

[5]M. Borowitz, "The military use if small satellites in orbit", *Briefings de l'Ifri*, Mar. 4, 2022.

[6]"Horizontal launch: A versatile concept for assured space access", NASA - DARPA, Tech. Rep. NASA SP 2011-215994, 2011.

[7]"Pegasus rocket". (), [Online]. Available: https://www.northropgrumman.com/space/pegasus-rocket/.

[8]M. Sarigul-Klijn and N. Sarigul-Klijn, "A stufy of air launch methods for rlvs", AIAA 2001-4619, AIAA Space 2001 - Conference and Exposition, Albuquerque: AIAA, Aug. 2001.

[9]N. Grumman, *Pegasus Payload User's Guide*, Northrop Grumman, Sep. 2020.

[10]M. W. van Kesteren, "Air launch versus ground launch: A muldisciplinary design optimization study of expendable launch vehicles on cost and performance", Delft University of Technology, 2013.

[11]M. W. van Kesteren, B. T. C. Zandbergen, M. C. Naeije, and A. van Kleef, "Design and analysis of an airborne, solid propelled, nanosatellite launch vehicle using multidisciplinary design optimization", 6th EUROPEAN CONFERENCE FOR AERONAUTICS and SPACE SCIENCES (EUCASS), Jun. 2015.

[12]M. M. Sarigul-Klijn, N. Sarigul-Klijn, G. C. Hudson, and C. Brown, "A new air launch concept: Vertical air launch sled (vals)", AIAA SPACE 2012 Conference Exposition, Pasadena, Calofirnia: AIAA, Sep. 2012. DOI: https://doi.org/10.2514/6.2012-5156.

[13]D. Rovner, "Gnc for pegasus®air-launched space booster: Design and first flight results", 1991 SAE Aerospace Atlantic, Dayton, Ohio, Apr. 1991.

[14]J. M. Hanson, M. W. Shareder, and C. A. Cruzen, "Ascent guidance comparisons", *The Journal of the Astronautical Sciences*, vol. 43, no. 3, pp. 307–326, 1995.

[15]D. M. Gaspar, "A tool for preliminary design of rockets", Tecnico Lisboa, 2014.

[16]G. W. Cherry, "A general, ecplicit optimizing guidance law for rocket-propelled spaceflight", AIAA/ION Astrodynamics Guidance and Control Conference, Aug. 1964.

[17]F. M. Perkins, "Derivation of linear-tangent steering laws", Aerospace Corporation, El Segundo, California, Tech. Rep. SSD-TR-66-211, 1966.

[18]I. E. Smith, "General formulation of the iterative guidance mode", NASA, George C. Marshall Space Flight Center, Huntsville, Alabama, Tech. Rep. NASA TM X-53414, 1966.

[19]D. C. Chandler and I. E. Smith, "Development of the iterative guidance mode with its application to various vehicles and missions", *Journal of Spacecraft*, vol. 4, no. 7, pp. 898–903, 1967.

[20]H. J. Horn, D. C. Chandler, and V. L. Buckelew, "Iterative guidance applied to generalized missions", *Journal of Spacecraft*, vol. 6, pp. 898–903, 1967. DOI: 10.2514/3.29523.

[21]T. J. Brand, D. W. Brown, and J. P. Higgins, "Space shuttle gnc equation document", Charles Stark Draper Laboratory, Cambridge, Massachusetts, 02139, Tech. Rep. NAS9-10268, 1973.

[22]P. V. der Porten, N. Ahmad, M. Hawkins, and T. Fill, "Powered explicit guidance modifications enhancements for space launch system block-1 and block-1b

vehicles", AIAA 2001-4619, AIAA Space 2001 - Conference and Exposition, Albuquerque: AIAA, Aug. 2001.

[23]U. Celik, "Robust booster landing guidance/control", KTH Royal Institute of Technology, 2020.

[24]D. Malyuta, Y. Yu, and B. Açıkmeşe, "Advances in trajectory optimization for space vehicle control", *Annual Reviews in Control*, 2021.

[25]A. Noureldin, J. Georgy, and T. B. Karamat, *Fundamentals of Inertial Navigation, Satellite-based Positioning and their Integration*. Springer, 2013.

[26]H. D. Curtis, *Orbital Mechanics for Engineering Student*. Buttherwoth–Heinemann, 2014.

[27]P. Hill and C. Peterson, *Mechanics and Thermodynamics of Propulsion*. Pearson Education, 1992.

[28]U.S. Committee on Extension to the Standard Atmosphere, "U.S. Standard Atmosphere, 1976", U.S. Government Printing Office, Tech. Rep., 1976.

[29]T. Yanik and H. E. Söken, "Design of a guidance system for an airlaunch rocket using iterative guidance mode", in *2023 10th International Conference on Recent Advances in Air and Space Technologies (RAST)*, 2023, pp. 1–6. DOI: 10.1109/RAST57548.2023.10197935.

[30]M. Rieck, M. Bittner, B. Grüter, *et al.*, *Falcon.m user guide, Version 1.30*, 2023. [Online]. Available: http://www.falcon-m.com.

[31]The MathWorks, Inc., *MATLAB*, [Online]. Available: https://www.mathworks.com/products/matlab.html, Accessed: August 31, 2023, 2023.

[32]A. E. Bryson and Y.-C. Ho, *Applied Optimal Control Optimization, Estimation, and Control*. Taylor Francis, 1975.