Orbital Rendezvous Using an Augmented Lambert Guidance Scheme

by

Sara Jean MacLellan

B.S. Aerospace Engineering
Embry-Riddle Aeronautical University, 2003

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN AERONAUTICS AND ASTRONAUTICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2005

Signature of Author:_____
Department of Aeronautics and Astronautics
May 06, 2005

Certified by:_____
Andrew Staugler
Senior Member of the Technical Staff
The Charles Stark Draper Laboratory, Inc.
Technical Advisor

Accepted by:_____
Richard H. Battin, Ph.D.
Senior Lecturer of Aeronautics and Astronautics
Thesis Supervisor

Accepted by:_____
Jaime Peraire, Ph.D.
Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

[This page intentionally left blank.]

Orbital Rendezvous Using an Augmented Lambert Guidance Scheme

by

Sara Jean MacLellan

Submitted to the Department of Aeronautics and Astronautics
on May 06, 2005, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

The development of an Augmented Lambert Guidance Algorithm that matches the position and velocity of an orbiting target spacecraft is presented in this thesis. The Augmented Lambert Guidance Algorithm manipulates the inputs given to a preexisting Lambert guidance algorithm to control the boost of a launch vehicle, or chaser, from the surface of the Earth to a transfer trajectory enroute to the aim point. After the chaser coasts along this transfer trajectory for a time, a manoeuver is performed to match the position and velocity of the target spacecraft. A three degree-of-freedom model was created to simulate the dynamics of the chaser and target spacecraft. The simulation was used to evaluate the ability and versatility of the Augmented Lambert Guidance Algorithm. The analysis proved that the methods developed in this thesis create a feasible algorithm to perform the desired tasks.

Technical Advisor: Andrew Staugler
Title: Senior Member of the Technical Staff, C.S. Draper Laboratory

Thesis Supervisor: Richard H. Battin, Ph.D.
Title: Senior Lecturer of Aeronautics and Astronautics

[This page intentionally left blank.]

Acknowledgments

May 06, 2005

Over the past two years, I've had the opportunity to attend the best engineering school in the country and perform research for a company that has been intimately involved with advancing space technologies since the dawn of the space age. I thank The Charles Stark Draper Laboratory for supporting me throughout this time.

There is one particular person at Draper that I can't thank enough, my technical advisor Andy Staugler. Andy, it was a pleasure working with you the past two years and I thank you for all of the time you have spent working with me. I thank you for helping me work out most of the kinks in my simulation (we'll just forget about that offset position issue) and especially all of the dozen or so times you read through the rough drafts of this document.

I've had the pleasure of learning Astrodynamics from the man who literally wrote the book. Thank you Prof. Battin for all you have taught me. You truly are an inspiration.

There is no way I could have gone through this without Evan. I can't believe you were able to put up with me when I was under so much stress. I don't know how I would have survived these two years if I couldn't come home and see you every day. I also thank some other members of my family: My mom and Russ for their generosity in helping with my housing costs. My wonderful grandparents that have supported me throughout my college education and deserve a great big thank you. Jenny, thanks for picking up the phone when I call, you always do say the right things.

To my officemates: Gordon, Theresia, and Dave thanks for being there to help me pass the time when I just didn't want to work. You guys made coming to that white, windowless cube actually bearable. I'd also like to thank Anil Rao who is kind of like our unofficial office mate. Thanks for helping me figure out where I wanted to work, and thanks for all the advice and words of wisdom. I will truly miss you when I go to JPL.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under contract N00030-05-C-0008, sponsored by the U.S. Navy.

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

Sara Jean MacLellan

Contents

[This page intentionally left blank.]

List of Figures

[This page intentionally left blank.]

List of Tables

[This page intentionally left blank.]

Chapter 1

Introduction

The purpose for the research conducted in this thesis is to develop a new guidance algorithm. This new algorithm can be used for missions that launch from Earth, or any other solid-surfaced celestial body, and match the position and velocity of a spacecraft in orbit. This chapter provides the motivation behind pursuing such a guidance scheme, the concept and scope of the research, and the objectives to be accomplished. Finally, the topics to be covered are outlined by chapter.

1.1 Motivation

In the area of mission design, two particular topics that are currently of public and scientific interest involve the launching of a spacecraft into orbit to eventually inspect and/or dock with another craft in space.

The first topic pertains to humans living and working in space. With the tragic loss of Space Shuttle Columbia and her seven crew on February 01, 2003, NASA is now requiring the inspection of the exterior of the shuttle prior to returning to Earth. In the event of another incident of ceramic tile damage or the occurrence of any other factor that would impair the shuttle from reentering the Earth's atmosphere, the Orbiter and its crew are now required to manoeuver to the ISS and wait for another Shuttle to be readied and

launch a rescue mission.

In November 2004, the ISS was hit by a crisis that almost led to the evacuation of its three crew members. The previous crew was given permission to eat some of the food stored on the station intended for the next crew. However, by the time the new crew arrived, the food reserves had not been replenished. This oversight led to the crew rationing its food until a cargo ship could dock with the ISS weeks later.

Because the ISS is now going to be used as a life-raft of sorts for the crew of the Orbiter, there is an even greater need to be able to get such resources as food, water, oxygen and/or medical supplies on the ISS in a short period of time. Hence, new guidance algorithms must be developed to accommodate the new demands of being able to launch from the Earth and rendezvous with the orbiting ISS in a short period of time.

The creation of a guidance algorithm that conforms to the rapid ascent requirements can also be applied to other missions where time may not be as critical. This secondary use could help in the management of the aging satellite fleet.

Since the advent of commercial satellites in the 1960's, companies have been launching equipment into orbit to facilitate such common services as cellular telephones, GPS, radio, and the internet. With this ever-growing and aging satellite fleet, questions have been raised about ways in which the mature satellites can be inspected, refurbished, refueled, or removed from orbit.

Several missions have be constructed to support the upkeep or demise of these old satellites. In particular, there are three missions that have either been recently launched or are planning to launch in the near future: DART, XSS-11, and Orbital Express. These three missions are all geared to demonstrate the ability of a spacecraft to autonomously rendezvous, inspect, refuel, and/or service another satellite. Extending the lives of the current fleet of orbiting satellites by way of smaller maintenance missions carries with it the potential to saving money, reduce 'space junk', and change the way satellites are designed.

From fly-by visual inspections and/or rendezvousing with an ailing craft to sending up an ambulance of sorts to the ISS, the possible missions that use the new guidance algorithm are as diverse as the hundreds of satellites in orbit. However, all the missions have one component in common: the launch of another craft to reach the vicinity of the already orbiting satellite.

One option for the launch from Earth is a direct-ascent trajectory. In the direct-ascent scenario, the payload would be able to immediately connect to, or inspect, the malfunctioning satellite instead of the more traditional approach of launching the payload into a parking orbit and performing manoeuvres to reach the target satellite.

The advantage of the direct-ascent trajectory is that the inspection and/or rendezvous can occur in a matter of minutes after the launch of the spacecraft instead of the hours it may take using the traditional method. In the situation where the crew of the ISS is in dire need of oxygen or supplies, the speediness with which a craft can get from the surface of the Earth to the spacecraft is the most important aspect of the mission.

In this thesis, the attention is centered on a guidance algorithm capable of accomplishing the direct-ascent trajectory in a relatively short time span. Focusing on the guidance algorithm leads to easier implementation in current launch vehicles by making simple software changes.

1.2 Concept, Scope, & Objectives

When planning a direct-ascent, quick-response mission, one way to minimize the time needed to get the relatively small amount of supplies into orbit is to use a launch vehicle that is abundant and readily available. Hence, the use of small commercially available launch vehicles (or surplus missiles) make the most sense when planning missions of this sort.

These types of vehicles have standard guidance algorithms that may not be capable of

getting the payload to rendezvous with another spacecraft. To remedy this, these vehicles could add functionality to their existing guidance algorithms by simply adding, either into the existing processor or to another connected computer, an algorithm that modifies the inputs to the existing guidance algorithm. These modified inputs would alow the vehicle to match the position and velocity of the spacecraft in orbit.

One common guidance algorithm used in these type of small launch vehicles is the Lambert Guidance Algorithm. There have been a multitude of papers published examining the characteristics of the Lambert Problem, which provide the framework of the Lambert Guidance Algorithm. Most notably are the many papers written by Battin in References [3]-[5]. More recently, Burns and Scherock in Reference [6] looked at how a Lambert Guidance Algorithm can be modified to match the position and velocity of a ballistic target. The research in this thesis goes one step beyond the work done by Burns and Scherock and creates an Augmented Lambert Guidance Algorithm that modifies the inputs to a Lambert Guidance Algorithm with the objective of matching the position and velocity of an orbiting spacecraft.

The purpose of creating the Augmented Lambert Guidance Algorithm is to investigate the possibility of constructing such an algorithm, not to create a final version to be implemented on an actual launch vehicle. Therefore, only an initial version of the Augmented Lambert Guidance Algorithm was developed. The newly-developed algorithm will then be evaluated to quantify how well both the position and velocity of the launch vehicle payload match those of an orbiting spacecraft.

The first-order evaluation consists of creating a computer simulation, which utilizes various assumptions to simplify a range of factors. This simulation is created using a three degree-of-freedom model of the launch vehicle dynamics. As a result, the launch vehicle is essentially a point mass and is not concerned with the orientation, or attitude, of the craft. Additionally, the simulation only manages the guidance system and not the other systems of the launch vehicle, meaning that the intricacies of such items as the sensors,

attitude control, stage separation control, and thrust vector control mechanisms will be severely simplified or eliminated.

The focus of this thesis is solely on the ability of the Augmented Lambert Guidance Algorithm to control the launch vehicle while thrusting; therefore, not all of the aspects of the mission can or will be analyzed. Because such tasks as docking with an orbiting spacecraft and/or controlling the manoeuvres of one spacecraft around another for inspection require a higher fidelity model, they will not be considered in this thesis. For an example of possible inspection manoeuvres, and the fidelity needed to analyze them, see Woffinden's Masters thesis (Reference [12]). Because the docking manoeuvres are neglected, the payload of the launch vehicle will only match the position and velocity of the spacecraft in orbit instead of fully rendezvousing. However, the term 'rendezvous', as used in the rest of this thesis, refers to this matching of the position and velocity between the launch vehicle payload, also referred to as the chaser, and the orbiting spacecraft, or target spacecraft.

1.3 Thesis Overview

One of the main requirements in designing the Augmented Lambert Guidance Algorithm is that it must work with a preexisting Lambert Guidance Algorithm to guide the chaser. Therefore, the first task is to explore the theory behind the Lambert Guidance Algorithm. Chapter 2 discusses the fundamental problem that defines the Lambert Guidance Algorithm, which is known as Lambert's Problem. The properties of the problem along with the governing equations and solution techniques are described. Then, the implementation of Lambert's problem into a guidance algorithm is shown to provide a background for the development of the Augmented Lambert Guidance Algorithm.

After the principles of the Lambert Guidance Algorithm have been discussed, Chapter 3 defines the rendezvous problem in the context of the missions described in Section 1.1.

The assumptions used to better characterize the problem are then outlined. From these assumptions, the requirements of the launch vehicle are specified along with the necessary constraints on the orbit of the spacecraft.

Chapter 4 describes the development of the Augmented Lambert Guidance Algorithm by adhering to the assumptions given in Chapter 3. After the construction of the Augmented Lambert Guidance Algorithm was completed, a simulation was made to provide a first-order evaluation of the abilities of the newly-developed Augmented Lambert Guidance Algorithm. Chapter 5 depicts the four functional models used in the creation of a simulation and discusses how the functional models are implemented in the simulation software and executed over time.

Following the creation of the simulation, Chapter 6 provides an analysis of how well the Augmented Lambert Guidance Algorithm succeeds in rendezvousing with an orbiting spacecraft. The analysis covers a specified operational area based on the missions that will use this new algorithm. Finally, Chapter 7 summarizes the results of the evaluation, offers conclusions and describes what future work may be done in this area.

Chapter 2

Guidance Algorithm

For the missile programs of the 1950's through the newest missions headed to Saturn and beyond, engineers have been tasked to develop robust and versatile guidance systems that ensure success in spite of the unpredictability of many variables. Inconsistencies in such quantities as gravity, thrust, stage separation forces, and a host of others, all have an impact on the position and velocity of a spacecraft. Because of these variabilities, an onboard guidance system is needed to adapt to the changing conditions and control the spacecraft to ensure it satisfies the mission objectives.

One main component of the guidance system is the guidance algorithm. The guidance algorithm defines how the vehicle is steered while traveling from one point to another. For a spacecraft outside any discernable atmosphere, thereby eliminating any aerodynamic control, the only method of steering is to ignite an engine and thrust. Figure 2-1 outlines how the components of the onboard guidance system interact with the guidance algorithm to adjust the thrust and steer the spacecraft. The guidance algorithm takes inputs from Navigation and Targeting and outputs the velocity-to-be-gained to Steering. Then Steering and Flight Control collaborate to actually alter the thrust direction. After the thrust direction is changed, Navigation updates the spacecraft's position and velocity and the flow of information begins again; therefore, the interactions between the elements

occur in a closed-loop manner.

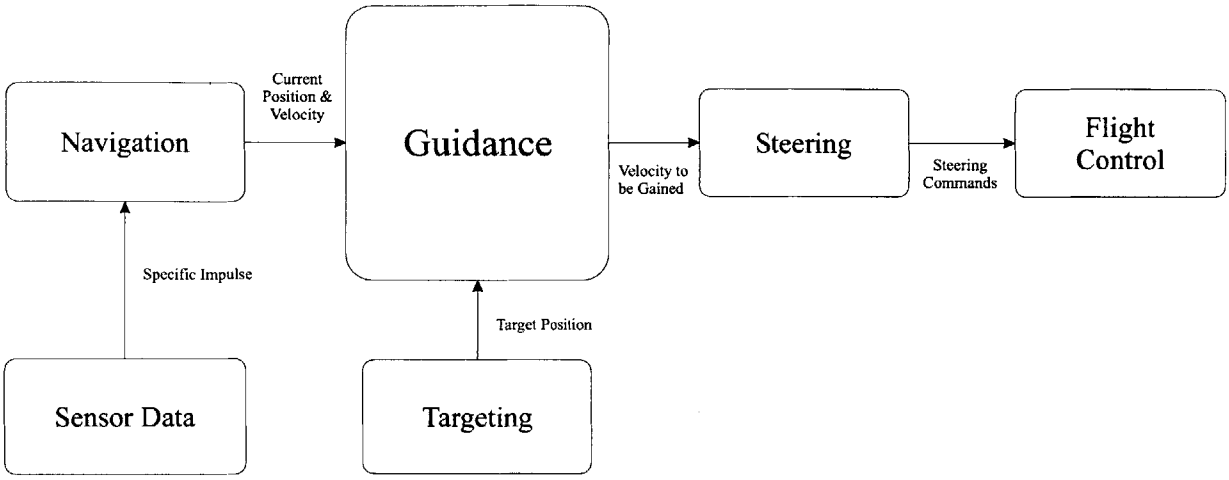

Figure 2-1: Guidance System Flow Chart

Given the spacecraft's current position from Navigation and an aim point from Targeting, there exists an infinite number of trajectories that will get the craft from its existing location to its destination. To limit the number of possible solutions, another constraint must be implemented. While there are many constraints from which to choose, fixing the time of flight between the two points, which corresponds to a Lambert Guidance Algorithm, is a reasonable and common choice for time-sensitive applications such as intercept and rendezvous.

A Lambert Guidance Algorithm solves for the velocity-to-be-gained by first focusing on a terminal state where the vehicle can shut off its engines and coast to the target. By focusing on this terminal state, the problem transforms into determining a transfer orbit that intersects both the current position and final destination with a transfer time between the two points equal to the remaining flight time; this is known as the Lambert Problem. The solution to Lambert's Problem is the correlated velocity, which is the instantaneous velocity needed at the current position for the spacecraft to be on the transfer orbit to the target. The Lambert Guidance Algorithm then takes the correlated velocity from Lambert's Problem and calculates the velocity-to-be-gained, which measures the difference in velocity between the current velocity and correlated velocity at the current time. The

spacecraft will then use the velocity-to-be-gained to control its thrust direction over time through Steering and Flight Control to a point where the spacecraft velocity matches the correlated velocity. At this point, the spacecraft terminates its thrust and coasts predictably to the target.

From determining the orbital elements of celestial bodies to guiding spacecraft and rockets, the solution to Lambert's Problem has been essential to the study of Astrodynamics. In this chapter, the properties of Lambert's Problem are explained, the governing equations are listed, and methods to solve the problem are discussed. The implementation of the Lambert Problem solution into a guidance algorithm will then be detailed. This chapter gives a general understanding of Lambert Guidance, which provides a basis for the development of the Augmented Lambert Guidance Algorithm discussed in Chapter 4.

2.1 Lambert's Problem

Given an initial position ($\vec{r_1}$), final position ($\vec{r_2}$), and a time of flight ($\mathbb{T}$), the solution to Lambert's Problem gives the instantaneous velocity ($\vec{v_L}$) needed for a craft to coast on a unique orbit from $\vec{r_1}$ to $\vec{r_2}$ in the time ($\mathbb{T}$) with gravity being the only force acting on the body.

2.1.1 Properties of the Problem

With the invariant time of flight, the number of possible solution trajectories between the initial and final destinations is limited to two. By defining the transfer angle, θ, as the angle between $\vec{r_1}$ and $\vec{r_2}$, the dot product theorem stated in Equation 2.1 is used to find the specific value of θ.

$$\theta = \arccos\left(\frac{\vec{r_1} \cdot \vec{r_2}}{r_1 r_2}\right) \tag{2.1}$$

On the interval from 0° to 360° there are two solutions, one in which $0° \leq \theta < 180°$ and another when $180° \leq \theta < 360°$. The two values of θ represent the two possible directions

of motion for a spacecraft orbiting around a body. To overcome this ambiguity and define a unique solution trajectory for Lambert's Problem, a direction of travel is simply chosen. In practice, the "short" ($\theta < 180°$) route is usually preferred. An example of two possible solutions for a given value of $\mathbb{T}$ are shown in Figure 2-2.

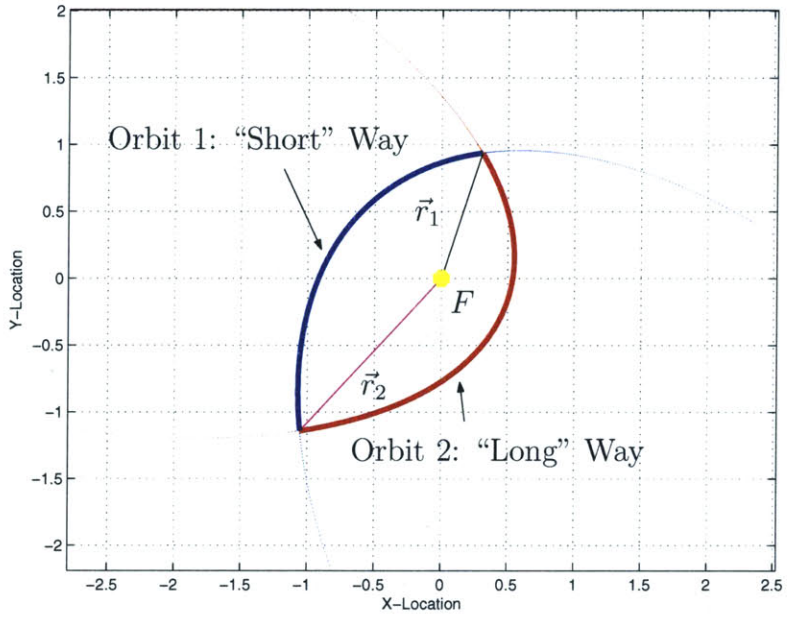

Figure 2-2: Two Possible Paths

No matter what combination of travel direction and time of flight is chosen, the resulting paths are all in the same orbital plane. The orbital plane is defined by the three points: the initial and final positions, $\vec{r}_1$ and $\vec{r}_2$, and the center of gravitational force, or focus, F. Although these three points are always known, there exists two situations where the orbital plane cannot be defined: when $\theta = 180°$ or $\theta = 0°$. In both of these cases, the three points defining the orbital plane are collinear and a unique orbital plane cannot be specified; therefore, many solutions exist. Some algorithms have developed special subroutines to handle these two cases; however, they will not be discussed here.

Before there can be any more discussion of Lambert's Problem, the geometry and associated variables must be illustrated. In the case of an elliptical transfer, the geometry is shown in Figure 2-3 while the symbol definitions are found in Table 2.1.

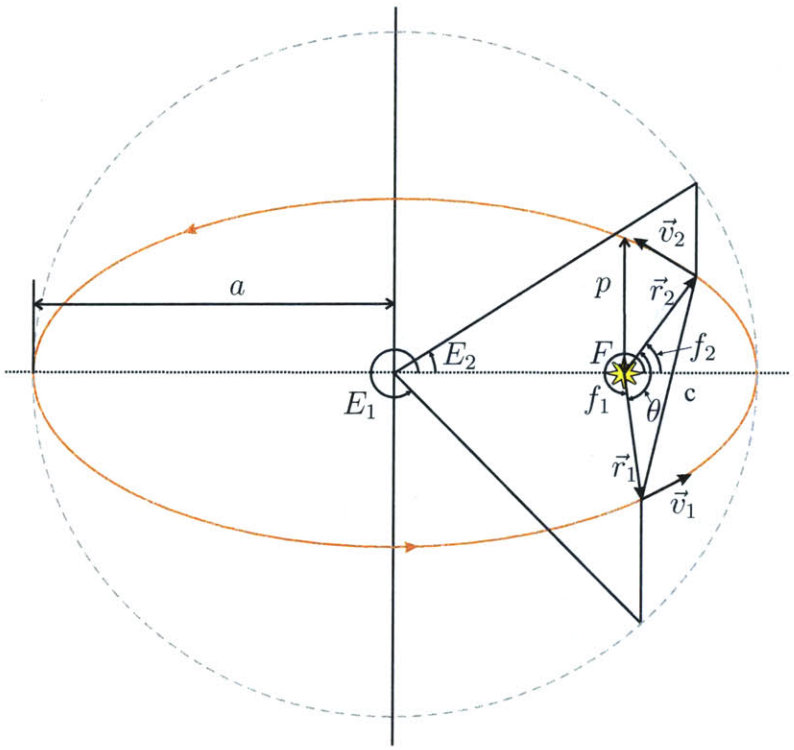

Figure 2-3: Elliptical Transfer

Table 2.1: Symbol Definition

Symbol	**Definition**
$\vec{r}_1$	Starting Position Vector
$\vec{r}_2$	Destination Position Vector
θ	Transfer angle
a	Semi-major axis
c	Chord
p	Semiparameter
s	Semiperimeter
e	Eccentricity
E_1	Eccentric Anomaly of $\vec{r}_1$
E_2	Eccentric Anomaly of $\vec{r}_2$
f_1	True Anomaly of $\vec{r}_1$
f_2	True Anomaly of $\vec{r}_2$
$\vec{v}_1$	Velocity at $\vec{r}_1$
$\vec{v}_2$	Velocity at $\vec{r}_2$

In addition, similar drawings can be made for parabolic and hyperbolic transfers, which can be found in Appendix A. More information about the derivation of these quantities and their meanings can be found in several references, but most notably in Reference [2].

2.1.2 Governing Equations

To solve Lambert's Problem, all of the values in Table 2.1 must be found to fully describe the orbit and find the correlated velocity. After the path, either "short" or "long", has been chosen, some quantities are found simply from the geometry, i.e.

$$\theta = \arccos\left(\frac{\vec{r_1}\cdot\vec{r_2}}{r_1 r_2}\right) \tag{2.2}$$

$$c = \sqrt{r_1^2 + r_2^2 - 2r_1 r_2\cos\theta} \tag{2.3}$$

$$s = \frac{r_1 + r_2 + c}{2} \tag{2.4}$$

The other quantities must be described in terms of the given quantities, $\vec{r_1}$, $\vec{r_2}$, $\mathbb{T}$, and those found from the geometry, θ, s, and c. The equations that describe the rest of the quantities were developed by Lagrange after Lambert theorized that, "the orbital transfer time ($\mathbb{T}$) depends only upon the semimajor axis (a), the sum of the distances of the initial and final points of the arc from the center of force ($r_1 + r_2$), and the length of the chord joining these points (c)"[2]. Lagrange was the first to supply the analytic formulas to prove Lambert's theories; therefore, these formulas are called Lagrange's Equations and are stated in Equations 2.5-2.9 for an elliptic transfer. The Lagrange Equations for parabolic and hyperbolic orbits are, again, listed in Appendix A. The two quantities ψ and $\cos\phi$ were defined to simplify the set of equations (2.7-2.9).

$$= \frac{1}{2}(E_2 - E_1) \tag{2.5}$$

$$\cos\phi = e\cos\frac{1}{2}(E_1 + E_2) \tag{2.6}$$

$$\mathbb{T}\sqrt{\mu} = 2a^{\frac{3}{2}}\left(\psi - \sin\psi\cos\phi\right) \tag{2.7}$$

$$r_1 + r_2 = 2a(1 - \cos\psi\cos\phi) \tag{2.8}$$

$$\sqrt{r_1 r_2}\cos\frac{1}{2}\theta = a(\cos\psi - \cos\phi) \tag{2.9}$$

As can be seen by this set of equations, there are three equations and three unknowns: a, ψ, and ϕ. The methods of solving this set of equations is the subject of section 2.1.3; however, once a solution is found, the other quantities listed in Table 2.1 can be ascertained from Equations 2.10 and 2.11 for the elliptic transfer.

$$p = \frac{r_1 r_2 \sin^2 \frac{1}{2}\theta}{a \sin^2 \psi} \tag{2.10}$$

$$e = \sqrt{1 - \frac{p}{a}} \tag{2.11}$$

By finding ψ, ϕ, a, p, and e, the unique orbit is completely defined, but the correlated velocity still needs to be found. This is where another part of Lagrange's work is used.

Lagrange Coefficients, or F and G functions, are usually used in the form shown in Equations 2.12 and 2.13. In this form, the position and velocity, $\vec{r}_2$ and $\vec{v}_2$, of a point anywhere on the orbit can be found by knowing the position and velocity of another point on the same orbit, where F, G, $\dot{F}$, and $\dot{G}$ are constants defined by the unique transfer orbit. These equations are mainly used to propagate the position and velocity of a spacecraft on an orbit.

$$\vec{r}_2 = F\vec{r}_1 + G\vec{v}_1 \tag{2.12}$$

$$\vec{v}_2 = \dot{F}\vec{r}_1 + \dot{G}\vec{v}_1 \tag{2.13}$$

With some careful manipulation, the velocities at two points on an orbit, $\vec{v}_1$ and $\vec{v}_2$, can be found given the two corresponding positions on that orbit, $\vec{r}_1$ and $\vec{r}_2$.

$$\vec{v}_1 = \frac{\vec{r}_1 - F\vec{r}_2}{G} \tag{2.14}$$

$$\vec{v}_2 = \frac{\dot{G}\vec{r}_1 - \vec{r}_1}{G} \tag{2.15}$$

In the case of Lambert's Problem, the correlated velocity is found by solving for $\vec{v}_1$ in

Equation 2.14. The quantities F, G, $\dot{F}$, and $\dot{G}$ for elliptic orbits are found from Equations 2.16-2.19. Appendix A lists the Lagrange Coefficients for the Parabolic and Hyperbolic cases.

$$F = 1 - \frac{r_2}{p}(1 - \cos\theta) = 1 - \frac{a}{r_1}(1 - \cos\psi) \tag{2.16}$$

$$G = \frac{r_2 r_1}{\sqrt{\mu p}} \sin\theta = \mathbb{T} - \sqrt{\frac{a^3}{\mu}}(\psi - \sin\psi) \tag{2.17}$$

$$\dot{F} = \frac{\sqrt{\mu}}{p} \tan\frac{1}{2}\theta \left(\frac{1 - \cos\theta}{p} - \frac{1}{r_1} - \frac{1}{r_2} \right) = -\frac{\sqrt{\mu a}}{r_1 r_2} \sin\psi \tag{2.18}$$

$$\dot{G} = 1 - \frac{r_1}{p}(1 - \cos\theta) = 1 - \frac{a}{r_2}(1 - \cos\psi) \tag{2.19}$$

The Lagrange Coefficients can also be used to solve Lambert's Problem without first using Lagrange's Equations. As it turns out, F, G, $\dot{F}$, and $\dot{G}$, are not independent, because they are described by only three unknowns: p, a, and ψ. Therefore, three of the equations can be solved for the three unknowns and fully describe the unique orbit between $\vec{r}_1$ and $\vec{r}_2$.

The derivation of the Lagrange Equations can be found in Reference [2] while the derivation of the Lagrange Coefficients is shown in Reference [1].

2.1.3 Solution Techniques

Upon closer inspection of these two sets of three equations (2.7-2.9 and 2.16-2.18) no amount of algebraic operations can analytically solve either set of equations for all three unknowns, which means the they are transcendental and must be solve iteratively. This property of the problem along with the importance of solving Lambert's Problem to the study of Astrodynamics is what has spurred the development of several algorithms to solve Lambert's Problem. Creating algorithms to iteratively find a solution has commanded attention from scholars since Lambert first published his solution almost 250 years ago. From Lambert's original solution to many algorithms developed by commercial companies

and the military, there are a plethora of papers and documentation on how to solve this problem.

While these algorithms were developed for varying reasons, their methods are all similar: they choose an independent iteration parameter and either Lagrange's Equations or Lagrange's Coefficients to solve for the unique solution orbit, then solve for the correlated velocity.

The basic procedure used in these many algorithms can be shown as a flow chart in Figure 2-4.

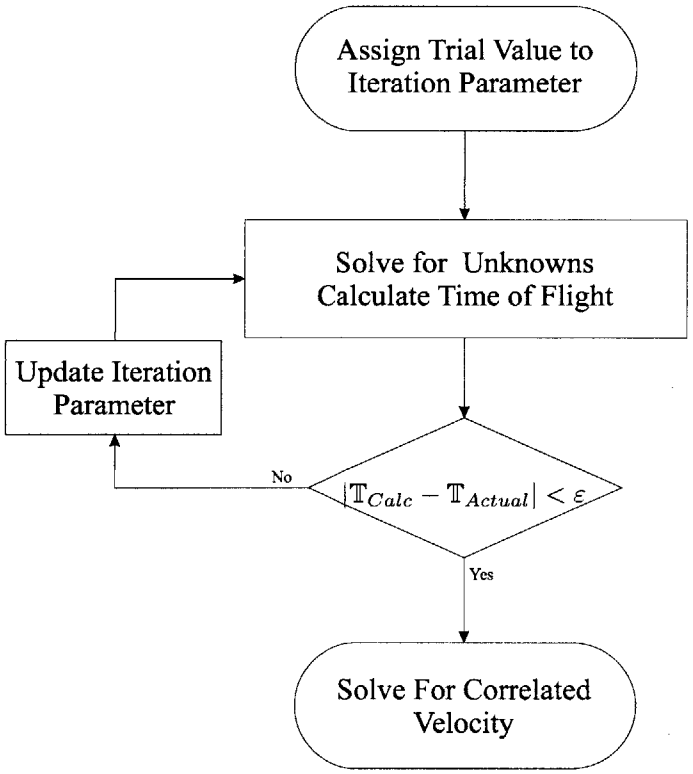

Figure 2-4: Lambert Algorithm Flow Chart

Iteration Parameter Characteristics

One of the most important decisions in developing or selecting an algorithm is the choice for the iteration parameter. While iteration parameters vary from algorithm to algorithm, inspecting both Lagrange's Equations and Lagrange's Coefficients reveals some likely

candidates for the iteration parameter, i.e. p, a, ϕ, and ψ. In addition, other quantities have been utilized such as f_1, the true anomaly and e, the eccentricity. These quantities have also been transformed into other forms such as $\frac{1}{2}\psi^2$ or $\sin^2\frac{1}{2}\psi$. Essentially any quantity can be used as long as it either directly or indirectly involves one of the three unknowns in the set of chosen equations.

Although any of these quantities can be chosen and used to solve Lambert's Problem, some have characteristics that make them less desirable while others have advantageous qualities, which must be taken into consideration when looking at an algorithm. For instance, a has many characteristics associated with it that make it an unattractive iteration parameter candidate. The time of flight equation (2.7) is a double-value function of a, so a goes to infinity for a parabolic orbit, which may cause computational errors in an algorithm. Furthermore, there are issues with other parameters such as p because according to Battin in Reference [2] the semiparameter has the same value for all orbits in a 180° transfer orbit, which in some cases may cause erroneous results. By choosing the true anomaly as an iteration parameter, a singularity or multiple solutions when $\theta = 180°$ can be avoided, yet the orbital plane can still not be defined as stated previously in Section 2.1.1.

An important characteristic of an iteration parameter is its versatility. Throughout the discussion of solving Lambert's Problem, it has been noted that there are three different variations of both the Lagrange Equations and Lagrange Coefficients, one for each of the elliptic, parabolic, and hyperbolic transfer orbit cases. It would be useful to have an iteration parameter that is versatile and would encompass all possible orbits.

Some candidate iteration parameters that are defined for all orbits are the eccentricity, true anomaly, and semiparameter. Furthermore, there are a host of other iteration parameters that are defined in terms of the eccentric, parabolic, and hyperbolic anomalies. Based on the algorithms developed in References [2] and [8], the iteration parameters that use the anomalies are developed using the difference between the anomaly values at the

two positions. Therefore the actual iteration parameter ends up being ΔE, $\Delta\beta$, and ΔH. Where ΔE is the change in eccentric anomaly, $\Delta\beta$ is the change in parabolic anomaly, and ΔH is the change in hyperbolic anomaly.

$$\Delta E = (E_2 - E_1) \tag{2.20}$$

$$\Delta\beta = (\beta_2 - \beta_1) \tag{2.21}$$

$$\Delta H = (H_2 - H_1) \tag{2.22}$$

One iteration parameter that uses the anomaly differences is χ as seen in Equation 2.23.

$$\chi = \begin{cases} \dfrac{\Delta E}{2} & \text{elliptic} \\ 0 & \text{parabolic} \\ -\dfrac{\Delta H}{2} & \text{hyberbolic} \end{cases} \tag{2.23}$$

By definition $E_2 > E_1$ and $H_2 > H_1$, hence the range of values for χ can vary from positive to negative with no discontinuities. Furthermore, values of χ greater than zero correspond to ellipses, while values less than zero are hyperbolas and a value equal to zero represents a parabolic orbit.

Other sets of iteration parameters that also have this property are X and x defined in Equations 2.24 and 2.25.

$$X = \begin{cases} \sin^2 \frac{1}{4}(\Delta E) & \text{elliptic} \\ 0 & \text{parabolic} \\ -\sinh^2 \frac{1}{4}(\Delta H) & \text{hyperbolic} \end{cases} \tag{2.24}$$

$$x = \begin{cases} \tan^2 \frac{1}{4}(\Delta E) & \text{elliptic} \\ 0 & \text{parabolic} \\ -\tanh^2 \frac{1}{4}(\Delta H) & \text{hyperbolic} \end{cases} \tag{2.25}$$

2.2 Lambert Guidance

For any moment in time, the spacecraft's position and velocity, $\vec{r}_R$ and $\vec{v}_R$, are known and the correlated velocity, $\vec{v}_L$, that satisfies the Lambert Problem can be found as described in the previous section. Therefore, to get the spacecraft on the transfer trajectory found from Lambert's Problem, a change in velocity ($\Delta\vec{v}$) is needed. This $\Delta\vec{v}$ will be referred to as the velocity-to-be-gained ($\vec{v}_G$) and can be found from Equation 2.26.

$$\vec{v}_G = \vec{v}_L - \vec{v}_R \tag{2.26}$$

A vector diagram of the relationship between $\vec{v}_R$, $\vec{v}_L$, and $\vec{v}_G$ is shown in Figure 2-5.

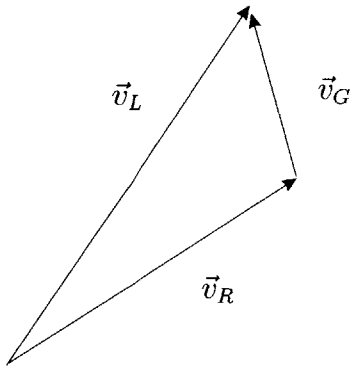

Figure 2-5: Lambert Guidance Vector Diagram

Once $\vec{v}_G$ is calculated, it is fed into a steering algorithm, which has its own logic to manipulate the thrust to reduce $\vec{v}_G$ over time.

Chapter 3

Rendezvous Problem

From the discussion in Chapter 2, a thrusting spacecraft can be steered, using Lambert Guidance, from an initial position to a point where the engines can be shut off and the spacecraft will coast to its final destination. One particular application where Lambert Guidance can be used is that of the intercept problem. Whether it is an Earth-launched ballistic missile intercepting a target on the ground, having one spacecraft performing a fly-by inspection of another, or any of the other possible missions, vehicles using Lambert Guidance for intercept missions have been successful.

To get a better understanding of an intercept problem, the fly-by inspection scenario is outlined and a diagram of the mission is shown in Figure 3-1. For this mission, the "Inspector" spacecraft, or chaser, on an initial orbit would execute a single manoeuver burn using a Lambert Guidance Algorithm to get onto a transfer trajectory that reaches the target point. Once $\vec{v}_G$ has been reduced to the cut-off value, the thrust terminates and the chaser coasts the rest of the way to target point on the transfer trajectory. Once at the target point, the fly-by inspection occurs and the chaser continues on the transfer orbit while the "Target" continues on its original orbit. Although the two spacecraft get in close proximity, their velocities at the target point are such that their positions diverge soon after their closest encounter.

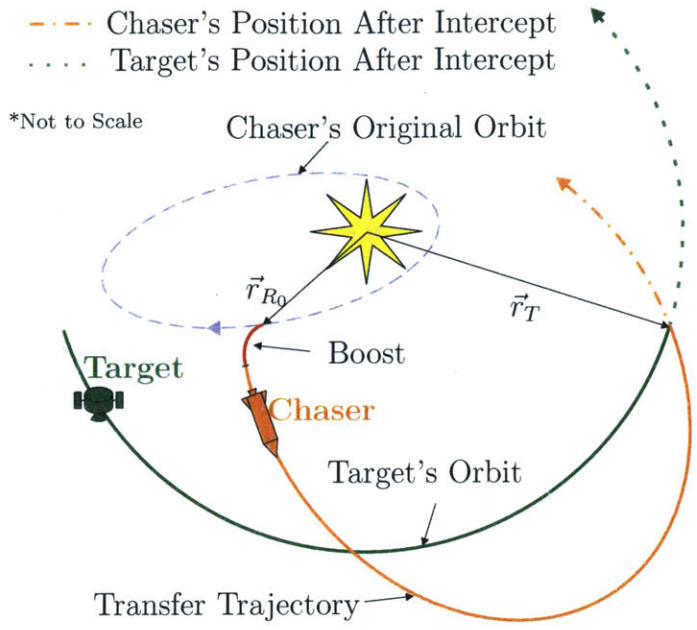

Figure 3-1: Intercept Diagram

While a Lambert Guidance Algorithm is well-suited for intercept problems, it does not have direct control over the spacecraft's velocity when it gets to the target point; therefore, applications such as rendezvous, which require a specific velocity at the target point, must either use a different guidance algorithm, or use a modified Lambert Guidance Algorithm.

The development of a particular method to modify a Lambert Guidance Algorithm (LGA) to create an Augmented Lambert Guidance Algorithm (ALGA) is discussed in detail in Chapter 4; however, the rendezvous problem description, including assumptions and success criteria, is the subject of this chapter.

3.1 Problem Definition

Figure 3-2 shows a typical trajectory for the rendezvous missions discussed in this thesis. A launch vehicle, initially at rest on the surface of the Earth, launches off its platform and performs the first burn, which is referred to as the boost phase. The engine is then cut off and the vehicle coasts on the transfer trajectory. Some time later, the chaser performs

a second burn manoeuver to match both the position and velocity of the orbiting target spacecraft. The target and chaser follow the same orbit after the second burn until a time when the terminal manoeuvres for inspection and/or docking are performed.

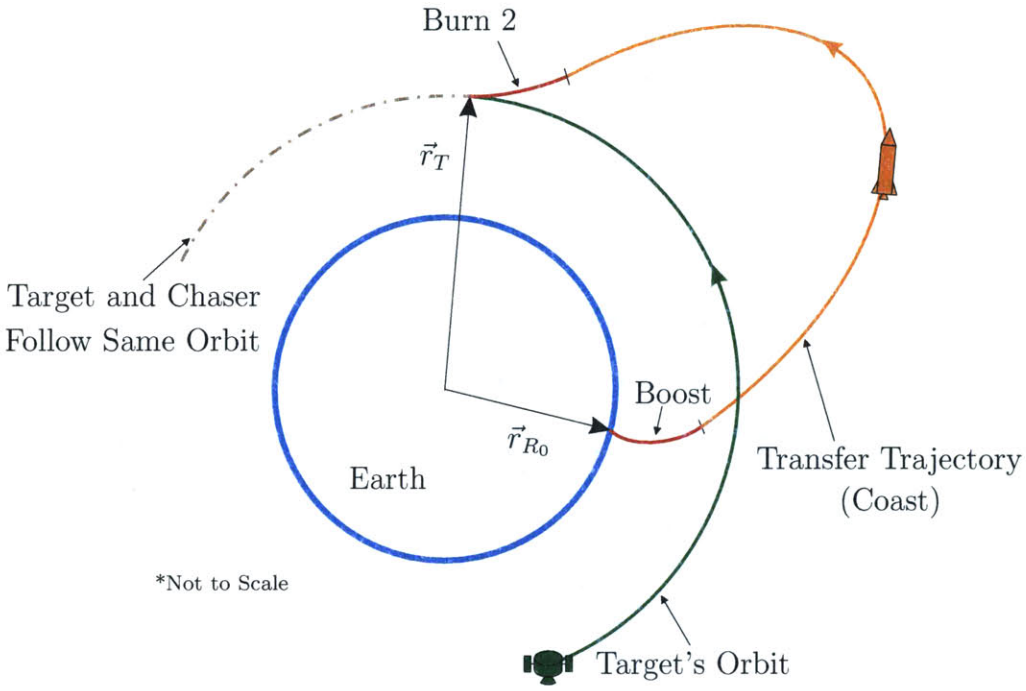

Figure 3-2: Rendezvous Diagram

3.2 Assumptions

Before the process of designing the ALGA could commence, assumptions were made to provide a framework from which to develop the algorithm. These assumptions dictate the dynamic equations used along with the initial conditions and vehicle configuration used to further characterize how to create the ALGA.

3.2.1 Initial and Target Conditions

At the launch time, the chaser knows all of its physical characteristics, i.e. weight, number of stages, etc. along with its position on the globe perfectly and is at rest at the surface

of the Earth.

$$\vec{r}_R(0) = \vec{r}_{R_0} \tag{3.1}$$

$$\vec{v}_R(0) = \vec{0} \tag{3.2}$$

Additionally, the chaser is given the exact position and velocity of the target spacecraft at the launch time along with the time of flight to the target, $\mathbb{T}$.

$$\vec{r}_{s/c}(0) = \vec{r}_{s/c_0} \tag{3.3}$$

$$\vec{v}_{s/c}(0) = \vec{v}_{s/c_0} \tag{3.4}$$

The target spacecraft is designated as a non-manoeuvering target; therefore, its path is an orbit around the Earth. Given the initial conditions and the specification that it moves in a predictable orbit, the position and velocity ($\vec{r}_{s/c}$ and $\vec{v}_{s/c}$) of the target spacecraft can be found by propagating the initial conditions along the orbit over a time interval using Lagrange Coefficients. The target conditions are then found from Equations 3.5 and 3.6.

$$\vec{r}_T = \vec{r}_{s/c}(\mathbb{T}) \tag{3.5}$$

$$\vec{v}_T = \vec{v}_{s/c}(\mathbb{T}) \tag{3.6}$$

3.2.2 Dynamics

One major assumption is that all of the motion is governed by the two-body approximation. This assumption provides sufficient accuracy to perform a first-order evaluation of the ALGA and significantly simplifies the necessary equations.

In combination with the two-body approximation, the target spacecraft and chaser's motion will be found with respect to the Earth, which will be modeled as a non-rotating, perfect sphere with gravity being proportional to the square of the distance from the

center of the Earth as described by Newton's Second Law.

Although effects from the oblateness of the earth, atmospheric drag, solar radiation pressure and third-body gravitation would perturb the actual trajectory of the vehicles, they will be neglected for this analysis.

3.2.3 Vehicle

For the rendezvous mission scenarios to be investigated in this thesis, the target spacecraft was elected to be launched from the surface of the Earth on either a land- or sea-based platform. The vehicle is assumed to have multiple stages to bring a payload to orbit; therefore, the chaser will consist of three boost stages and the payload.

As stated before, the LGA has no direct control over the velocity of a spacecraft when it gets to its destination. A second burn provides the necessary velocity change needed to match both the position and velocity of the target spacecraft. To execute this manoeuver, the spacecraft payload must have a motor and fuel available to achieve the second manoeuver. Hence, the payload will consist of an additional 4^{th} stage engine attached to the 100 kg structure and equipment (i.e., cameras, sensors, etc.) essential to the mission.

The second burn, also known as the 4^{th} stage manoeuver, was chosen to have a constant thrust magnitude and a fixed thrust direction with respect to the Earth for the entire burn time, $\mathbb{T}_b$. The 4^{th} stage ignites at a time, $\mathbb{T}_{4^{\text{th}}}$, so that after the 4^{th} stage burn the chaser can terminate its thrust and arrive at the target point at the given time, $\mathbb{T}$.

$$\mathbb{T}_{4^{th}} = \mathbb{T} - \mathbb{T}_b \tag{3.7}$$

Dictated by the amount of velocity change needed to rendezvous with the target, $\mathbb{T}_b$, and therefore $\mathbb{T}_{4^{th}}$, are both variable quantities. Consequently, the 4^{th} stage is assumed to be able to initiate and terminate its thrust according to these values. The chaser can begin

its terminal guidance either immediately after the 4^{th} stage cut-off or at any future time.

3.2.4 Boost Control

The development of the ALGA described in Chapter 4 is based on modifying the inputs into the LGA. Because the LGA is designed for intercept missions, a comparison of the intercept and rendezvous trajectories is illustrated in Figure 3-3 to provide more insight into the differences between the two. The trajectories were given the same initial conditions: $\vec{r}_{R_0}$, $\vec{r}_{s/c_0}$, $\vec{v}_{s/c_0}$ and, $\mathbb{T}$.

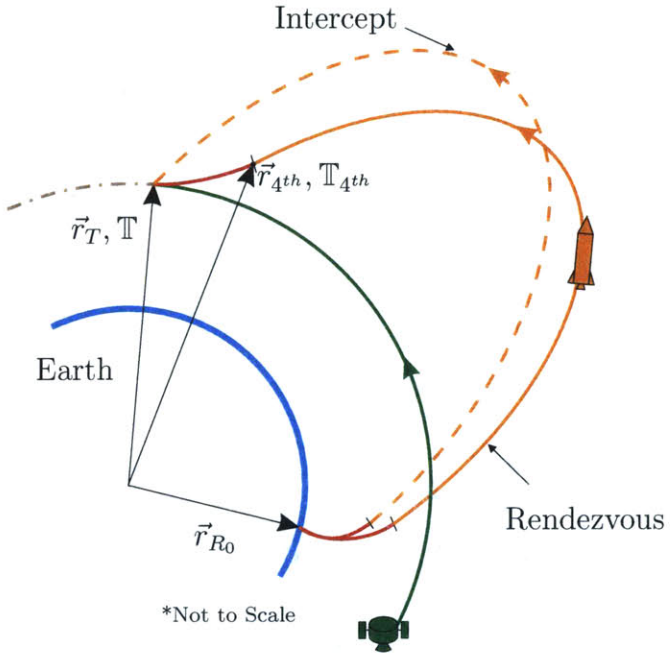

Figure 3-3: Trajectory Comparison

In the intercept problem, the LGA controls the boost phase in such a way that the vehicle coasts to the target point, $\vec{r}_T$, in the specified time, $\mathbb{T}$, as detailed in Chapter 2. For the rendezvous problem, as defined in this chapter, the ALGA is assumed to control the boost phase of the chaser to coast to a different point, $\vec{r}_{4^{th}}$, at a different time, $\mathbb{T}_{4^{th}}$, in order to perform the 4^{th} stage burn of length $\mathbb{T}_b$ to match the position and velocity of the target at the given time, $\mathbb{T}$.

3.3 Requirements

After the ALGA is developed, a simulation will be created to test the versatility and effectiveness of the algorithm by running through multiple test cases. But, before the ALGA can be put through a rigorous evaluation, several requirements must be specified to generate consistent and meaningful test cases. To this end, the chaser vehicle specifications along with limitations on the target orbits are described. Following this, the success criteria are outlined.

3.3.1 Vehicle Specifications

To provide an accurate representation of the chaser vehicle, the vehicle specifications are based on current US Space Launch Systems and will adhere to the assumptions outlined previously in Section 3.2.3.

The boost stages will have solid fuel motors with no restart or throttling capabilities; however, the 4^{th} stage engine will be capable of restarting multiple times, but it will not be able to throttle. These requirements lead to a search for the configuration and capabilities of current launch vehicles. The search will be limited to vehicles that launch small payloads (< 2000 kg) and those that are widely available. The results of this search will be the basis by which the chaser vehicle specifications will be defined.

The following is a description of a selection of current US Space Launch Systems. These summaries are based off information in Reference [7]:

Athena II - Developed by Lockheed Martin to launch small spacecraft into LEO orbits, the Athena II consists of four stages with the first three boost stages having solid fuel and the fourth having a liquid engine.

Atlas IIIA/B - Initially designed to be an ICBM and then redesigned by Lockheed Martin, the Atlas IIIA/B is a two stage liquid-fueled vehicle designed to carry medium payloads into GTO.

Delta II - Developed mainly to launch US Air Force payloads into GPS or LEO orbits, the Delta II is available in two- and three-stage versions (with a zeroth solid rocket booster stage) to accommodate both small and medium payloads. The first and second stages are liquid fueled and the optional third stage is solid fueled.

Minotaur - Used by the US Air Force, this launch vehicle combines parts from the Minuteman II ICBM and Pegasus XL launch vehicle (Made by The Orbital Sciences Corporation) to launch small payloads into LEO orbits. Its four stages are solid fueled.

Taurus - This solid fueled four-stage launch vehicle originally used a Peacekeeper missile for the first stage and a Pegasus as an upper stage to launch military payloads. Then The Orbital Sciences Corporation used a commercial first stage to create a commercial version and launch small payloads into LEO and GTO.

Boost Sizing

The vehicle design specification values for the Space Launch Systems described previously, again gathered from Reference [7], were tabulated in a spreadsheet and then averaged. The tabulated data from Reference [7] can be found in Appendix B. Using engineering judgement, values were then chosen for the specific quantities needed for the boost stages. The following table describes the specifications of the chosen chaser to be used in the simulation.

Table 3.1: Boost Stage Specifications

	Engine Statistics			**Mass**		
Stage	**Thrust** *Newtons*	**Burn Time** *Seconds*	**ISP** *Seconds*	**Structural** *Kg*	**Propellant** *Kg*	**Fraction** *unitless*
1	1,409,250	78.05	258.10	4276.50	45221.25	0.91
2	746,600	70.46	278.50	2141.00	20496.80	0.91
3	186,200	95.52	289.24	794.25	6143.00	0.89

4th Stage Sizing

In this research, the abilities of the 4th stage are very important. It has to have enough thrusting capability to change the velocity of the craft by values of up to 5000 m/sec. To make sure the 4th stage would be able to complete the manoeuvres it will be tasked to do, some planning had to go into its design.

The mass of the payload is comprised of the mass of the propellant (m_{Prop}), the structural mass for the engine m_{struct}, and the mass of the equipment (m_{equip}).

$$M_{4^{th}} = m_{prop} + m_{struct} + m_{equip} \tag{3.8}$$

As stated in Section 3.2.3, the equipment mass is assumed to be 100 kg. However, the configuration of the 4th stage engine, which includes the propellant mass and structural mass has yet to be determined.

Having looked at the third or fourth stages of the launch vehicles on which the boost stages of the chaser were based, preliminary values for the thrust (T), approximate mass fraction (mf), and specific impulse (I_{sp}) were chosen for the 4th stage engine. Based on these values and estimating that the 4th stage engine will have to provide approximately 5000 m/sec of velocity change (ΔV) to the entire payload, sufficient information has been acquired to size the 4th stage engine.

The contributions of mass for the 4th stage engine from both the propellant and structure were found from the equations developed in Reference [13]. First, the propellant mass was calculated from Equation 3.9. Note that g_0 is the gravitational acceleration at sea-level.

$$m_{prop} = \frac{m_{equip}\left[\exp\left(\frac{\Delta V}{g_o I_{sp}}\right)\right]}{\frac{1}{mf} - \frac{1-mf}{mf}\exp\left(\frac{\Delta V}{g_o I_{sp}}\right)} \tag{3.9}$$

After the propellant mass was calculated, it was rounded up to the nearest integer and

Equation 3.10 is then used to find the structural mass.

$$m_{struct} = \frac{m_{prop}(1 - mf)}{mf} \tag{3.10}$$

As a consequence of having a limited amount of fuel, there is a finite amount of time in which the 4th stage engine can burn fuel and produce thrust. Given the propellant mass, the maximum burn time was calculated from Equation 3.11.

$$\mathbb{T}_{b_{max}} = \frac{I_{sp} g_o m_{prop}}{T} \tag{3.11}$$

After all of the calculations, the 4th stage engine specifications are now known and are shown in Table 3.2.

Table 3.2: 4th Stage Engine Specifications

	Engine Statistics			**Mass**		
Stage	**Thrust** *Newtons*	**Burn Time** *Seconds*	**ISP** *Seconds*	**Structural** *Kg*	**Propellant** *Kg*	**Fraction** *unitless*
4	10,000	264.87	300	100	900	0.90

3.3.2 Target Orbits

Based on the capabilities of the US Space Launch Systems, it is impractical to try and rendezvous with a target spacecraft in such a high altitude orbit as a geostationary communications satellite, because the Space Launch Systems simply do not have enough fuel to reach those high altitude orbits. Therefore, the possible orbits for the target spacecraft are limited to Low Earth Orbits (LEO), which are defined as orbits whose semimajor axis is between 100km and 500km greater than the radius of the Earth.

Because of the limited fuel, the chaser launch vehicle is presumed to not have enough fuel to reach the parabolic and hyperbolic escape velocities; consequently, the focus is primarily on circular and elliptic target orbits.

3.4 Success Criteria

After the development of the ALGA, it must be tested for its effectiveness. To this end, computer models are created to replicate the conditions of an actual test flight. Test cases are constructed that utilize various initial conditions for the chaser and an assortment of target orbits, which adhere to the requirements outlined previously. The information generated by the test cases is used as inputs to the Simulink simulation software, which executes the logic dictated by the models over the given time interval, $\mathbb{T}$. The models used in the simulation to test the ALGA are described in Chapter 5.

At the completion of the simulation, the effectiveness of the ALGA in guiding the chaser to rendezvous with the target can be evaluated for that particular test case. The evaluation is based on four factors. Two of the factors are based on calculating the deviation in the position and velocity of the chaser with respect to the target. The position and velocity errors, R_{err} and V_{err} respectively, are defined as the magnitude difference between the value for the chaser and the value for the target spacecraft at the rendezvous time, $\mathbb{T}$.

$$R_{err} = \|\vec{r}_R - \vec{r}_{s/c}\| \tag{3.12}$$

$$V_{err} = \|\vec{v}_R - \vec{v}_{s/c}\| \tag{3.13}$$

A test case is deemed successful if the following four criteria are met:

1. The chaser does not impact the Earth - The magnitude of the chaser's position vector ($\vec{r}_R$) is never less than the radius of the Earth ($r_\oplus$).

$$r_R > r_\oplus \tag{3.14}$$

2. The 4^{th} stage does not run out of fuel - The burn time needed to perform the

rendezvous does not exceed the maximum burn time of the 4^{th} stage.

$$\mathbb{T}_b \leq \mathbb{T}_{b_{max}} \tag{3.15}$$

3. The Position Error is within tolerable limits - The value is less than 100 meters.

$$R_{err} \leq 100 \text{ m} \tag{3.16}$$

4. The Velocity Error is within tolerable limits - The value is less than 0.5 meters/second.

$$V_{err} \leq 0.5 \text{ m/sec} \tag{3.17}$$

Chapter 4

Augmenting Lambert

Starting with a preexisting Lambert Guidance Algorithm (LGA) in conjunction with the assumptions outlined in Chapter 3, a method was devised to create the Augmented Lambert Guidance Algorithm (ALGA) for use in rendezvous missions. This method entails the creation of an adjunct algorithm that modifies the inputs into the original LGA as opposed to explicitly altering it. The inspiration for creating an adjunct algorithm that utilizes the given LGA stems from the implementation in an actual launch vehicle. The adjunct algorithm could be placed on a supplementary processor or as a separate function within the existing guidance algorithm processor. Essentially, the adjunct algorithm could be used as an optional module that could be added based on the mission requirements.

The general procedure used to generate the ALGA is derived from the work done by Burns and Scherock as described in Reference [6]. In their analysis, they ignored the effects of the changing gravity vector during the 4^{th} stage burn because their chaser and target were both traveling on ballistic trajectories and their burn times were relatively short (<50 seconds). In contrast, the missions discussed in this thesis have the target in an orbital trajectory and the burn times are anticipated to be long (100-300 seconds) due to the large $\Delta\vec{v}$ changes (maximum of 5000 m/sec). Therefore, ignoring the shifting gravity vector could not be upheld for the analysis in this thesis. Consequently, the method

derived in this chapter builds upon the technique created by Burns and Scherock. This new method provides insight into how to modify the inputs to the LGA while including compensation for the varying gravity vector during the 4^{th} stage burn.

4.1 General Procedure

As stated in Chapter 3, the objective of the ALGA is to get the chaser on a transfer trajectory that reaches $\vec{r}_{4^{th}}$ at the time $\mathbb{T}_{4^{th}}$. It is then the adjunct algorithm's task to modify the inputs given to the LGA to produce the expected transfer trajectory. Considering what values the LGA needs to compute $\vec{v}_G$, the adjunct algorithm is only able to modify the target position, $\vec{r}_T$, the current position, $\vec{r}_R$, and/or the time of flight, $\mathbb{T}$, given to the LGA. Consequently, there are seven possible combinations of modified inputs that could be given to the LGA and produce the same transfer trajectory.

1. Modify only the target position. INPUTS: $\vec{r}_{Tmod}$, $\mathbb{T}$, and $\vec{r}_R$
2. Modify only the time of flight. INPUTS: $\vec{r}$, $\mathbb{T}_{mod}$, and $\vec{r}$
3. Modify only the current position. INPUTS: $\vec{r}$, $\mathbb{T}$, and $\vec{r}_{Rmod}$
4. Modify the target position and the time of flight. INPUTS: $\vec{r}_{Tmod}$, $\mathbb{T}_{mod}$, and $\vec{r}_R$
5. Modify the target position and the current position. INPUTS: $\vec{r}_{Tmod}$, $\mathbb{T}$, and $\vec{r}_{Rmod}$
6. Modify the time of flight and the current position. INPUTS: $\vec{r}$, $\mathbb{T}_{mod}$, and $\vec{r}_{Rmod}$
7. Modify all three inputs. INPUTS: $\vec{r}_{Tmod}$, $\mathbb{T}_{mod}$, and $\vec{r}_{Rmod}$

In this thesis, the choice was made to implement option one, thereby only modifying one input into the LGA; the target position, $\vec{r}_T$. Figure 4-1 shows the relationship between the actual target point, modified target position, and the ignition point; $\vec{r}_T$, $\vec{r}_{Tmod}$, and $\vec{r}_{4^{th}}$, respectively.

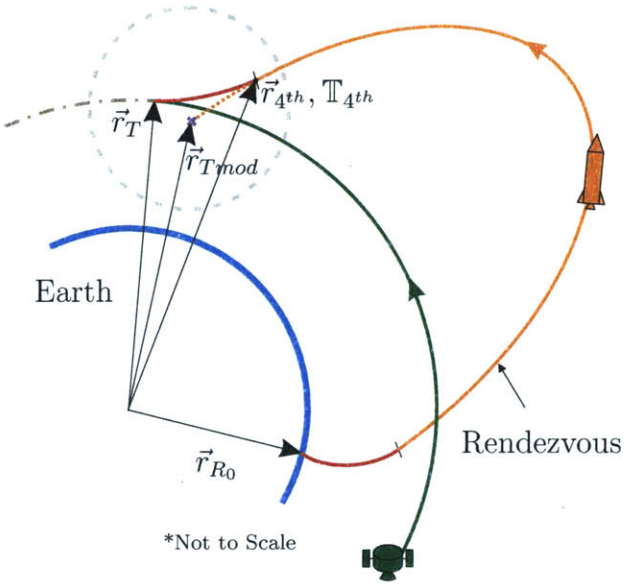

Figure 4-1: False Target Diagram

By fixing the time of flight, several properties of the modified target position become evident. First, as a result of the 4$^{\text{th}}$ stage burn, the chaser never reaches $\vec{r}_{Tmod}$ (providing the 4$^{\text{th}}$ stage engine ignites); so, an important characteristic of $\vec{r}_{Tmod}$ is that it is a false target. Another feature of this point is that the time to get from $\vec{r}_{4th}$ to $\vec{r}_{Tmod}$ is equal to the time to get from $\vec{r}_{4th}$ to $\vec{r}_T$, which is the burn time, $\mathbb{T}_b$.

Calculating $\vec{r}_{Tmod}$ requires that the chaser reach $\vec{r}_{4th}$ at $\mathbb{T}_{4th}$; hence, it would be helpful to know those values. However, $\vec{r}_{4th}$ and $\mathbb{T}_{4th}$ are dependant on the values of the thrust direction vector, $\vec{B}_{4th}$, and the burn time, $\mathbb{T}_b$. Both of these values are initially unknown and are, coincidentally, needed by the other systems on the chaser to successfully execute the 4$^{\text{th}}$ stage manoeuver. Therefore, in addition to calculating $\vec{r}_{Tmod}$, the adjunct algorithm is also responsible for calculating $\mathbb{T}_b$ and $\vec{B}_{4th}$.

The goal of the adjunct algorithm as part of the ALGA is to exploit the properties of the LGA and the methods used by Burns and Scherock to calculate the three unknown values: $\vec{r}_{Tmod}$, $\vec{B}_{4th}$, and $\mathbb{T}_b$. The modified technique is outlined in Figure 4-2. As can be seen from the flow chart, there are five main steps (shown by bold outline): Initialization, State Propagation, Match Velocity, Match Position, and Iteration.

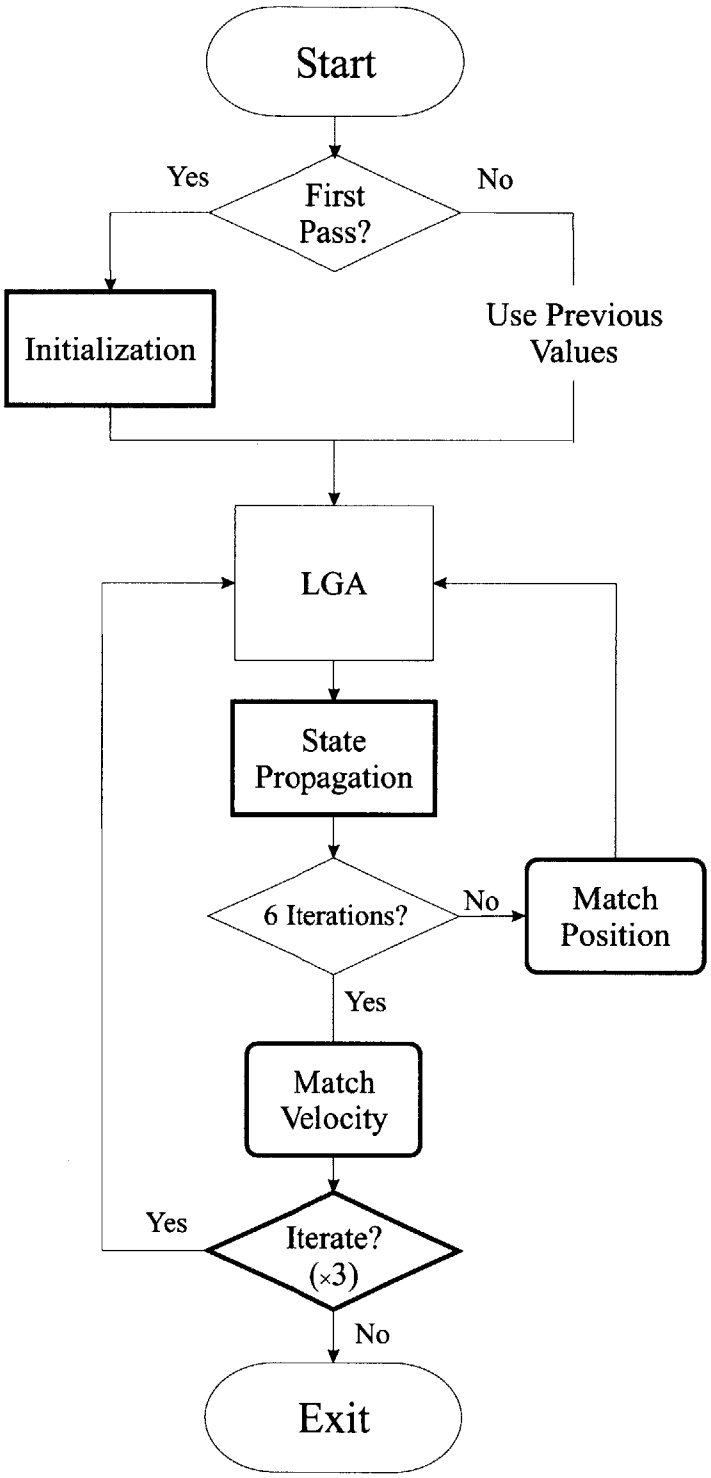

Figure 4-2: ALGA Flow Diagram

The Initialization block employs the method used by Burns and Scherock to find initial approximations for the three unknowns. Therefore, the initialization values are found by

neglecting the change in the acceleration due to gravity while the 4^{th} stage engine is thrusting. The LGA is run using either the quantities from the Initialization block or the values from the previous time step depending on whether this is the first time the ALGA is called. The state produced by the LGA is propagated forward to get values at the rendezvous time in the State Propagation block. These terminal conditions are then used by the Match Position and Match Velocity blocks to generate updated values for the unknowns. The Matching Position block produces a more accurate value of $\vec{r}_{Tmod}$ while the Matching Velocity block updates the values for $\vec{B}_{4th}$ and $\mathbb{T}_b$. The outputs of these two blocks collectively compensate for the effects due to the variation in gravity during the 4^{th} stage burn. Finally, an outer iteration loop is used to produce more accurate values for $\vec{r}_{Tmod}$, $\vec{B}_{4th}$, and $\mathbb{T}_b$.

4.2 Initialization

On the first call to the ALGA, the only information given is the initial conditions as described in Section 3.2.1. The goal of the initialization block is to take the initial conditions and produce reasonable approximations for the three unknowns. This is done by making a first guess for the three unknowns, propagating the chaser's state forward to find the terminal conditions, then updating the values for the three unknowns based on the assumption that the gravity vector during the 4^{th} stage burn doesn't change.

4.2.1 First Guess

The first-guess step entails setting the 4^{th} stage burn time to zero and the modified target position equal to the actual target position and running these values through the LGA.

$$\mathbb{T}_b = 0 \tag{4.1}$$

$$\vec{r}_{Tmod} = \vec{r}_T \tag{4.2}$$

By setting these initial conditions, the resulting output from the LGA is essentially the correlated velocity needed to be on an intercept trajectory to the target.

4.2.2 Propagation

Using the current chaser position, and the correlated velocity computed by the LGA, the chaser's position and velocity are propagated forward in time by $\mathbb{T}$ to define the terminal conditions of the initialization trajectory. The propagation is accomplished using the Lagrange Coefficients method mentioned in Section 2.1.1. To find more information on using the Lagrange Coefficients to propagate a spacecraft's trajectory, see References [2] and [11].

The terminal conditions defined by the propagation of the initialization trajectory gives the chaser's velocity, $\vec{v}_R$, at the actual target point, $\vec{r}_T$, and provides the adjunct algorithm with a starting point from which to begin its calculations for the three unknowns.

Figure 4-3 is the area highlighted in Figure 4-1 and illustrates the initialization trajectory along with the chaser's velocity and the target spacecraft's velocity at the actual target point.

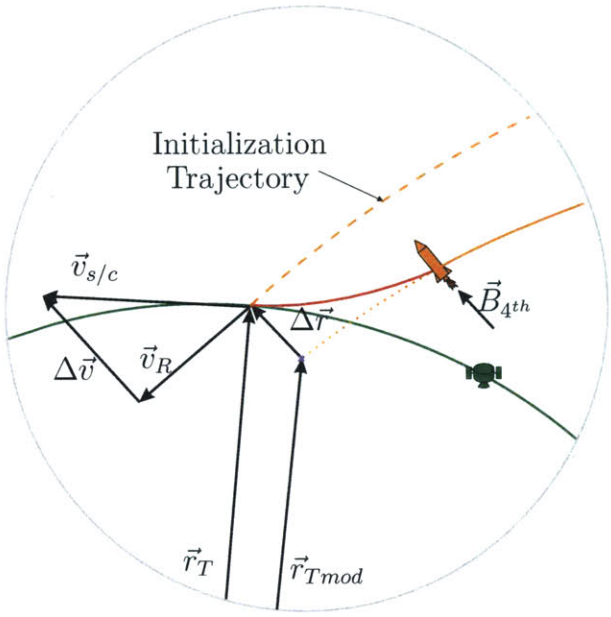

Figure 4-3: Rendezvous Velocity Vectors

4.2.3 Thrust Direction

As can be seen from the figure, $\Delta\vec{v}$ is the relative change in velocity that needs to occur to make the velocities of the chaser and target match, which is the reason for the 4^{th} stage burn. The initial value of $\Delta\vec{v}$ is found from Equation 4.3.

$$\Delta\vec{v} = \vec{v}_{s/c} - \vec{v}_R \tag{4.3}$$

Once a value for $\Delta\vec{v}$ has been found, the next step in the initialization is to find the thrust direction required to reduce $\Delta\vec{v}$ to zero by the rendezvous time, $\mathbb{T}$. To do this, the chaser initiates a 4^{th} stage manoeuver at $\mathbb{T}_{4th}$ in the direction of $\vec{B}_{4th}$. By aligning $\vec{B}_{4th}$ with $\Delta\vec{v}$, $\vec{v}_R$ changes over time and constantly reduces $\Delta\vec{v}$. Reducing $\Delta\vec{v}$ to zero accomplishes the goal of matching the velocity. The 4^{th} stage thrust direction vector $\vec{B}_{4th}$ is then defined by a unit vector in the direction of $\Delta\vec{v}$. As it turns out, the magnitude of $\Delta\vec{v}$ as well as its direction is needed in future calculations; therefore, $\Delta\vec{v}$ will be passed out of the Initialization block and $\vec{B}_{4th}$ will be extracted from that value when needed.

4.2.4 Burn Time

Remembering that $\mathbb{T}_{4th}$ is calculated by subtracting $\mathbb{T}_b$ from $\mathbb{T}$ (Equation 3.7), the value of $\mathbb{T}_b$ is a reasonable choice for the next unknown to be calculated.

The thrust provided by the 4^{th} stage engine is used to reduce $\Delta\vec{v}$ to zero over the burn time, $\mathbb{T}_b$. Therefore, $\Delta\vec{v}$ can be calculated by integrating the acceleration experienced by the chaser over the same interval.

$$\Delta\vec{v} = \int_0^{\mathbb{T}_b} \vec{a}_R(t)\ \mathrm{d}t \tag{4.4}$$

Given that the 4^{th} stage provides a constant thrust and burns its fuel at a constant rate, $\dot{M}$, where M_{4th} is the initial mass of the 4^{th}stage, the acceleration due to thrust can be

expressed as a function of time as stated in Equation 4.5.

$$\vec{a}_T(t) = \frac{I_{sp} g_0 \dot{M}}{M_{4th} - \dot{M}t} \tag{4.5}$$

For the initialization, the acceleration of the chaser during the 4$^{\text{th}}$ stage burn is just the acceleration produced by the thrust because the variation of the gravity vector is ignored.

Now that an equation for the acceleration with respect to time has been found, the integration is performed to get $\Delta\vec{v}$ expressed as a function of time. It should be noted that this is simply the standard rocket equation. This equation is then evaluated at the two limits using the Fundamental Theorem of Calculus, which results in an equation for $\Delta\vec{v}$ in terms of $\mathbb{T}_b$.

$$\begin{aligned} \Delta\vec{v} &= I_{sp} g_0 \int_0^{\mathbb{T}_b} \left[\frac{\dot{M}}{M_{4th} - \dot{M}t} \right] \vec{B}_{4^{th}} \, \mathrm{d}t \\ &= I_{sp} g_0 \left[\ln\left(\frac{M_{4th}}{M_{4th} - \dot{M}\mathbb{T}_b} \right) \right] \vec{B}_{4^{th}} \end{aligned} \tag{4.6}$$

By rearranging Equation 4.6 and noting that $\Delta\vec{v}$ is in the same direction as $\vec{B}_{4^{th}}$, the 4$^{\text{th}}$ stage burn time is found by Equation 4.7.

$$\mathbb{T}_b = \frac{M_{4th}}{\dot{M}} \left[1 - \exp\left(-\frac{\Delta v}{g_0 I_{sp}} \right) \right] \tag{4.7}$$

4.2.5 Modified Target Position

Now that initial values for $\vec{B}_{4^{th}}$ and $\mathbb{T}_b$ have been calculated, the last unknown, $\vec{r}_{Tmod}$, is ready to be determined. Equation 4.6 is integrated to find the position offset.

$$\begin{aligned} \Delta\vec{r} &= I_{sp} g_0 \int_0^{\mathbb{T}_b} \left[\ln\left(\frac{M_{4th}}{M_{4th} - \dot{M}t} \right) \right] \vec{B}_{4^{th}} \, \mathrm{d}t \\ &= I_{sp} g_0 \left[\left(\frac{M_{4th}}{\dot{M}} - \mathbb{T}_b \right) \ln\left(\frac{M_{4th} - \dot{M}\mathbb{T}_b}{M_{4th}} \right) + \mathbb{T}_b \right] \vec{B}_{4^{th}} \end{aligned} \tag{4.8}$$

When the position offset is found, the modified target position can finally be calculated from Equation 4.9

$$\vec{r}_{Tmod} = \vec{r}_T - \Delta\vec{r} \tag{4.9}$$

4.3 State Propagation

Using the burn time, modified target position, and thrust direction vector calculated from Equations 4.7, 4.9, and 4.3 will result in the chaser not matching the position and velocity of the target spacecraft due to the change in gravity vector during the burn. Figure 4-4 shows the gravity vectors when the 4$^{\text{th}}$ stage ignites and at the target point along with its effect on the chaser during the burn.

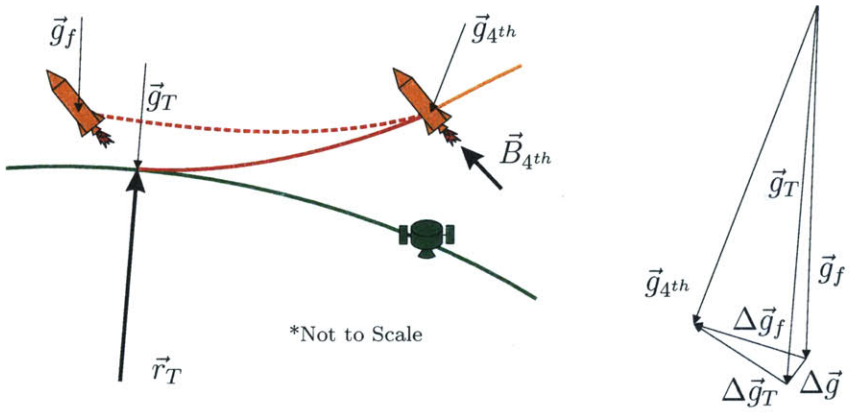

Figure 4-4: Gravity Vectors

Not accounting for the variation in gravity results in the acceleration experienced by the chaser, $\vec{a}_R$, being rotated. The amount of rotation is governed by how much the gravity vector changes from the ignition point to the final position after the burn. This change is denoted by $\Delta\vec{g}$ in the figure. As a consequence of the rotating acceleration vector, the position and velocity of the chaser also change direction and the chaser is driven away from the desired target location as shown by the dotted trajectory.

The objective of the State Propagation block in conjunction with the Match Position and Match Velocity blocks is to calculate values of $\vec{B}_{4th}$, $\vec{r}_{Tmod}$, and $\mathbb{T}_b$ that account for

the change in gravity over the burn time.

To more accurately represent the dynamics, gravity is now included in the acceleration equation experienced by the chaser and is shown in Equation 4.10.

$$\vec{a}_R = \left[\frac{I_{sp} g_0 \dot{M}}{M_{4th} - \dot{M}t}\right] \vec{B}_{4^{th}} - \frac{\mu}{r_R^3}\vec{r}_R \tag{4.10}$$

Unfortunately, this equation cannot be analytically integrated with respect to time like Burns and Scherock were able to do for the acceleration when ignoring gravity. To remedy this, a method was devised to account for the gravity variation.

First, the current chaser position along with the correlated velocity produced from the call to the LGA is propagated forward using Lagrange Coefficients. This is identical to the propagation done in the Initialization block in Section 4.2.2, except it is propagated forward to the ignition time, $\mathbb{T}_{4^{th}}$. The value of $\mathbb{T}_{4^{th}}$ is given by either the Initialization block or the value from the previous iteration.

The second step involves using a separate fourth-order Runge-Kutta integrator to numerically integrate Equation 4.10 over the burn time, $\mathbb{T}_b$. The initial conditions given to the Runge-Kutta integrator are the position and velocity output from the Lagrange Coefficient propagation. The result of this second propagation is the predicted position, $\vec{r}_p$, and velocity, $\vec{v}_p$, of the chaser after coasting, executing the 4^{th} stage burn and accounting for the effects of gravity. From these predicted values, the position and velocity of the target spacecraft are matched using the following techniques.

4.3.1 Matching Position

Figure 4-5 shows a possible miss-trajectory as predicted by the State Propagation block using values for $\mathbb{T}_{4^{th}}$, $\vec{r}_{Tmod}$, and $\Delta\vec{v}$ from either the Initialization block or from the previous iteration.

To find a better value of $\vec{r}_{Tmod}$, the difference in the predicted position and target

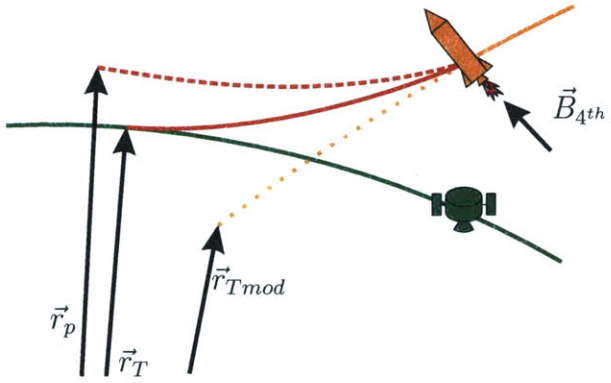

Figure 4-5: Matching Position

spacecraft position is found.

$$\vec{r}_{miss} = \vec{r}_p - \vec{r}_T \tag{4.11}$$

This value for the miss distance is then subtracted from target position given to the LGA.

$$\vec{r}_{T_{new}} = \vec{r}_{Tmod} - \vec{r}_{miss} \tag{4.12}$$

A diagram of the vector addition is shown in Figure 4-6.

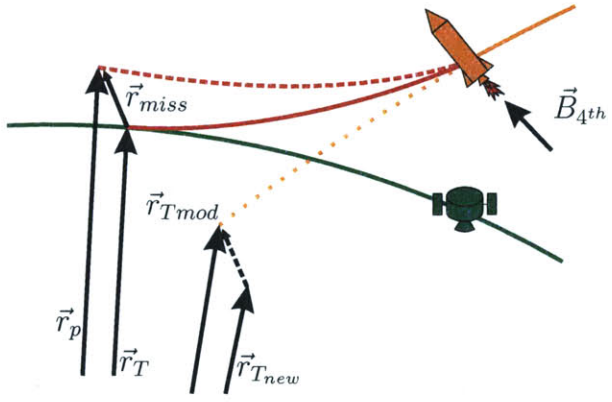

Figure 4-6: Matching Position Vector Addition

Once an updated value of $\vec{r}_{T_{new}}$ is calculated it is used as an input to the LGA to get a new value of $\vec{v}_G$. The State Propagation block is then executed again to acquire an updated predicted position and miss distance. This procedure is repeated until the value of $\vec{r}_{T_{new}}$, changes minimally over successive iterations. To find the appropriate number of

iterations, the magnitude of the miss distance versus the number of iterations was charted for one call to ALGA and one set of initial conditions. Figure 4-7 shows a logarithmic decrease in the magnitude of the miss distance over successive iterations. As can be seen by this particular example, the predicted miss distance for the sixth iteration is on the order of 10^{-6}. Several other trajectories were examined and it was established that six iterations leads to sufficient accuracy. Additionally, the three lines represent the three outer loop iterations as shown in Figure 4-2 and discussed further in Section 4.4.

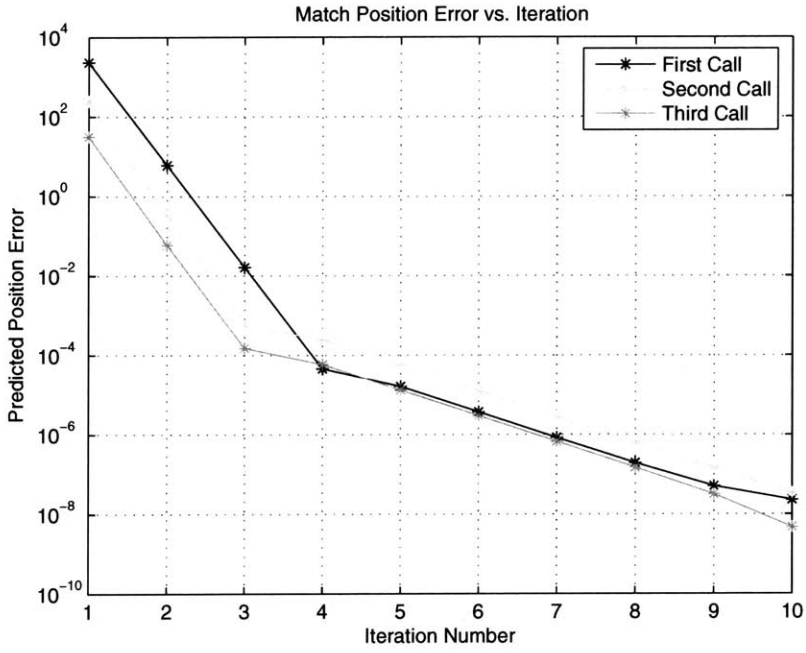

Figure 4-7: Matching Position Error

4.3.2 Matching Velocity

Using the predicted velocity, $\vec{v}_p$, from the Runge-Kutta integrator and Equations 4.6 and 4.7, an updated value for the thrust direction vector, $\vec{B}_{4^{th}}$, and burn time, $\mathbb{T}_b$, can be calculated. This updated value, again, takes into account the velocity difference due to gravity.

Figure 4-8 shows the predicted velocity with respect to the target spacecraft velocity

and the value of $\Delta\vec{v}$ passed from the Initialization block or the previous time step, which is parallel to the thrust direction vector, $\vec{B}_{4th}$.

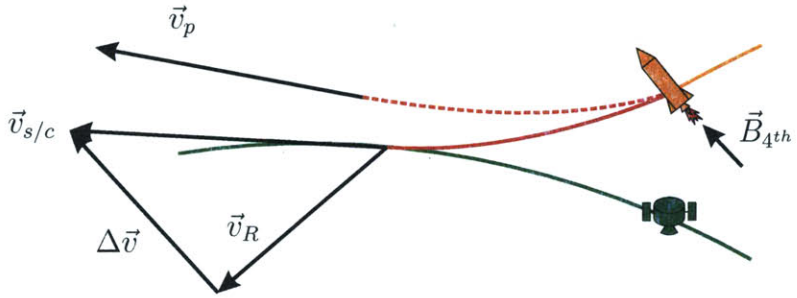

Figure 4-8: Matching Velocity

The first calculation to find improved values for the thrust direction vector and burn time is to take the difference between the predicted chaser velocity and target velocity.

$$\vec{v}_{miss} = \vec{v}_{s/c} - \vec{v}_p \tag{4.13}$$

This miss velocity is an additional change in velocity that the 4$^{\text{th}}$ stage needs to provide to the chaser to match the velocity of the target spacecraft. Therefore, the miss velocity is added to the initial value of the velocity change, $\Delta\vec{v}$.

$$\Delta\vec{v}_{new} = \Delta v + \vec{v}_{miss} \tag{4.14}$$

Figure 4-9 shows a diagram of the vector addition used to calculate the new velocity change.

The updated velocity change, $\Delta\vec{v}_{new}$, is now a closer approximation to how much velocity change the 4$^{\text{th}}$ stage needs to provide. Given that the thrust supplied by the 4$^{\text{th}}$ stage is constant, the burn time must be adjusted to accommodate the new velocity change. Equation 4.15 is used to calculate the new burn time.

$$\mathbb{T}_{b\ new} = \frac{M_{4th}}{\dot{M}}\left[1 - \exp\left(-\frac{\Delta v_{new}}{g_0 I_{sp}}\right)\right] \tag{4.15}$$

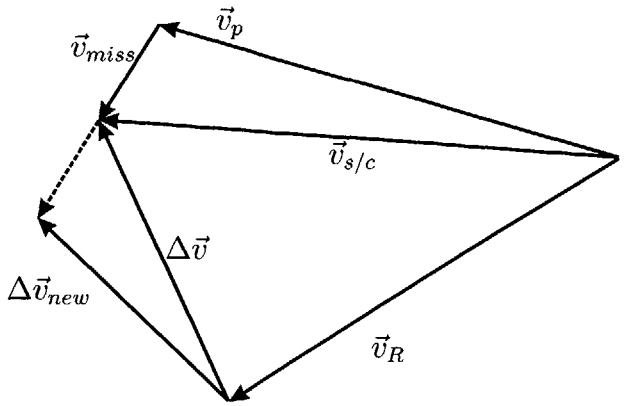

Figure 4-9: Matching Velocity Vector Addition

4.4 Iteration

The techniques employed to update the values of $\vec{r}_{Tmod}$, $\vec{B}_{4^{th}}$ and $\mathbb{T}_b$ did so by matching the position and velocity separately. Because the position and velocity of the chaser are coupled, an iteration scheme needs to be implemented to get increased accuracy for the values of $\vec{r}_{Tmod}$, $\vec{B}_{4^{th}}$ and $\mathbb{T}_b$.

This can be seen by running the same initial condition, but only changing the number of iterations used to calculate the three unknown values. Table 4.1 shows the difference in R_{err} and V_{err} when the number of iterations is increased from 2 to 3 for one particular set of initial conditions.

Table 4.1: R_{err} and V_{err} vs. Iterations

Iteration	R_{err} **(m)**	V_{err} **(m/s)**
1	30.5858	0.2686
2	51.5083	0.1422
3	10.0722	0.0798

Consequently, the number of iterations chosen for the ALGA used in this thesis is three. This provides sufficient accuracy given the success criteria outlined in Section 3.4.

Once the three iterations are performed, the ALGA outputs $\vec{r}_{Tmod}$, $\mathbb{T}_b$, and $\Delta\vec{v}$ (remembering that $\vec{B}_{4^{th}}$ is the unit vector of $\Delta\vec{v}$) to be saved and used in the next call to ALGA at a future time step.

Chapter 5

Models & Simulation

Modeling and Simulation of physical systems has become an indispensable tool in evaluating and developing new algorithms, software, and hardware in not only the space industry, but in every engineering field. Using such computer software as MATLAB, Simulink, and functions written in C-code, several models were created to embody the physical characteristics of the chaser, target spacecraft, and the environment in which they both operate. The implementation of the logic contained in the models over time creates the computer simulation, which will be used as a first-order assessment of the Augmented Lambert Guidance Algorithm (ALGA) developed in Chapter 4.

After the logic and equations dictated by the models are integrated into the simulation software, the robustness and versatility of the ALGA can be tested by running several simulations with various initial conditions. The results of running multiple simulations are used to gauge how successful the ALGA is at manoeuvering the chaser to rendezvous with the target spacecraft. The success of the algorithm will be determined by how well it meets the criteria laid out in Chapter 3 and will be discussed comprehensively in Chapter 6. But first, this chapter focuses on the logic and equations used to mimic the physical properties of the two vehicles and the dynamics of motion. These equations define the models, which are implemented using simulation software to create the computer simulation.

5.1 Models

The simulation used to evaluate the ALGA includes four separate functional models whose interactions are depicted in Figure 5-1. The current time measured from launch and the remaining flight time is available in all of the models. The four models are described in this chapter in conjunction with diagrams outlining their operations performed. These models implement the assumptions outlined in Chapter 3.

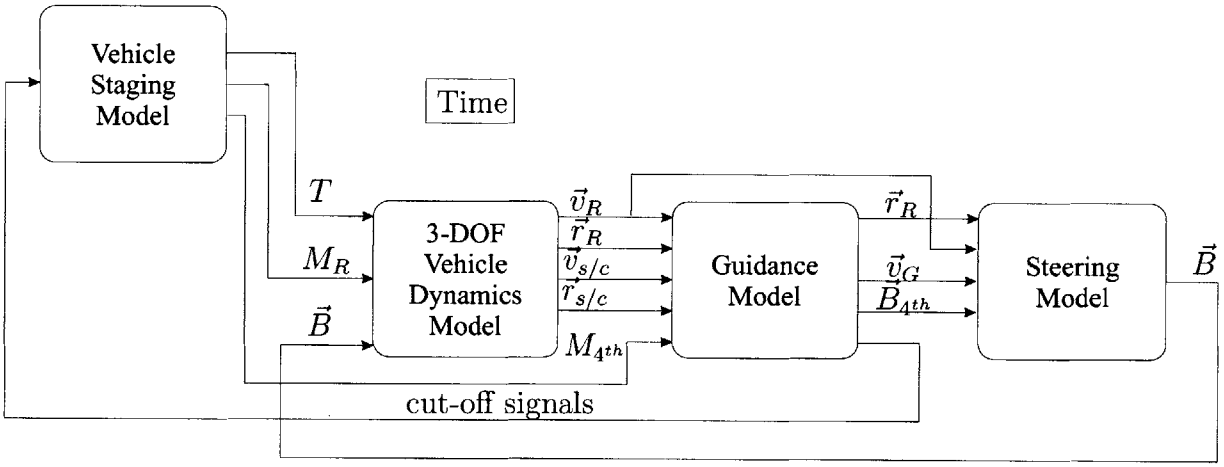

Figure 5-1: Model Diagram

The Vehicle Staging Model encompasses the physical characteristics of the chaser, from the mass depletion as a result of the burning of fuel to the execution of the stage separation. The 3-DOF Vehicle Dynamics Model doubly integrates the accelerations experienced by both the chaser and target spacecraft to provide the position and velocity of both vehicles. The execution of the ALGA is contained in the Guidance Model, while the Steering Model manages how the thrust of the chaser is directed throughout the trajectory from launch to rendezvous.

5.1.1 Symbol Definition

Although most of the symbols used in the four functional models have been defined and used throughout the preceding chapters, a table is included here as a refresher.

Table 5.1: Model Symbols

Symbol	Definition
M_R	Mass - Chaser
T	Thrust - Chaser
$\vec{B}$	Steering Vector
$\vec{r}_{s/c}$	Position - Target
$\vec{v}_{s/c}$	Velocity - Target
$\vec{r}_R$	Position - Chaser
$\vec{v}_R$	Velocity - Chaser
$\vec{v}_G$	Velocity-to-be-Gained
$\vec{B}_{4^{th}}$	Thrust Direction Vector - 4^{th} Stage
$M_{4^{th}}$	Mass - 4^{th} Stage
$\mathbb{T}_b$	Burn Time - 4^{th} Stage
T_{si}	Thrust - i^{th} Stage
M_{si}	Mass - i^{th} Stage
M_{pay}	Mass - Payload
$\vec{a}_T$	Acceleration - Thrust
$\vec{a}_R$	Acceleration - Chaser
$\vec{a}_{s/c}$	Acceleration - Target
$\vec{g}_R$	Gravity - Chaser
$\vec{g}_{s/c}$	Gravity - Target

5.1.2 Vehicle Staging Model

A generic small launch vehicle (GSLV), as discussed in Chapter 3, was chosen to be used as the chaser in the simulation. The staging conditions for the GSLV have been implemented to follow the vehicle model flow chart shown in Figure 5-2. A stage controller receives cut-off signals from the Guidance Model, which it uses along with its own logic to give signals to each of the stage motor subsystems. These signals indicate when it is time to either ignite the motor and begin expending fuel or to separate the remaining fuel and structural mass from the vehicle. This controller is where the physical limitations of the rocket stages are implemented. For instance, no two stages can be burning at the same time and the current stage cannot ignite without having first deployed the previous stages.

Each of these stage motor subsystems controls a mechanism that computes the propellant remaining in the storage tanks after the motor is given the signal to ignite. While

the propellant mass is depleting, the model also outputs the thrust value produced by the particular engine. When one motor is ignited, the thrust produced by the other stages is zero. Furthermore, once a stage is separated it contributes values of zero for the mass and thrust into the summations at the right of the diagram. To accurately represent the chaser's thrust and mass characteristics at any time during the simulation, the thrust and mass contributions from all of the stages is summed and output from the Vehicle Staging Model.

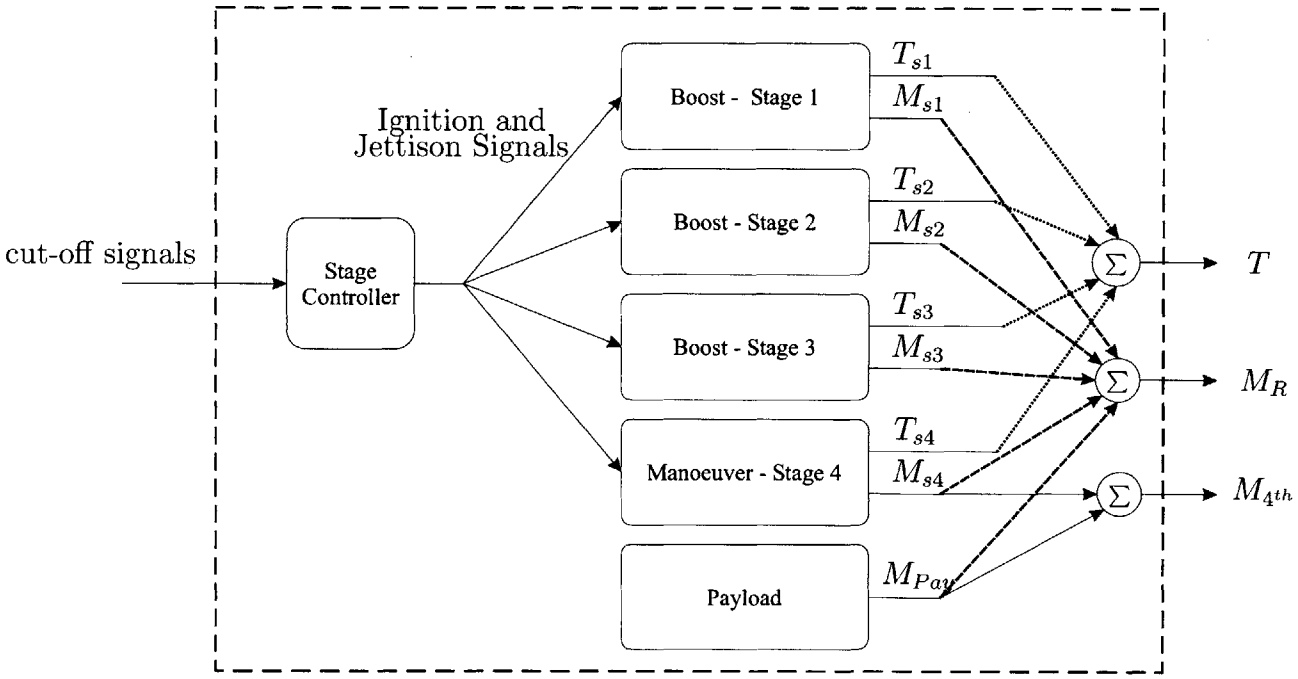

Figure 5-2: Vehicle Model Diagram

5.1.3 3-DOF Dynamics Model

Accurately describing the motion of the chaser relative to the Earth is critical to creating a useful simulation. Based on the assumptions outlined in Chapter 3, the equation of motion for the chaser and target are described by Equation 5.1.

$$\frac{d^2\vec{r}}{dt^2} + \frac{\mu}{r^3}\vec{r} - \vec{a}_T = \vec{0} \tag{5.1}$$

With this equation, the only information needed to calculate the state of the chaser or target at any time are initial values for their positions and velocities, $\vec{r}_R(0)$ and $\vec{r}_{s/c}(0)$, and $\vec{v}_R(0)$ and $\vec{v}_{s/c}(0)$, respectively and any external acceleration caused by the engine thrust, $\vec{a}_T$. To determine a future state, the equation needs to be numerically integrated over the specified time interval for each of the spacecraft.

Figure 5-3 shows the 3-DOF Dynamics model used to perform the numerical integration to find the current position and velocity of the two spacecraft in the simulation.

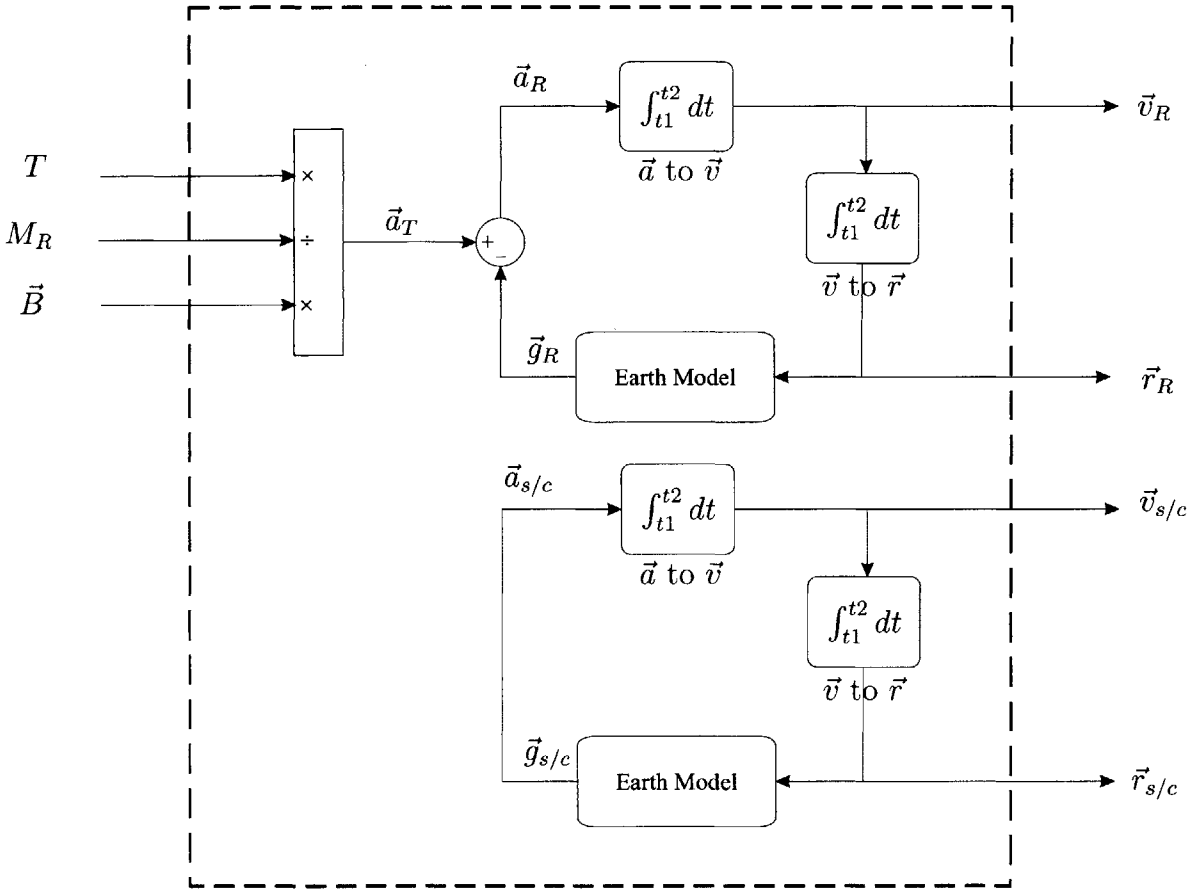

Figure 5-3: 3-DOF Dynamics Model Diagram

This model inputs the thrust and mass given by the Vehicle Staging Model and the thrust direction vector supplied by the Steering Model. From these input values, the specific force, or acceleration due to thrust, of the Launch Vehicle is calculated from

Equation 5.2.

$$\vec{a}_T = \frac{T}{M_R}\vec{B} \tag{5.2}$$

Meanwhile, the Earth model calculates the acceleration due to gravity for each of the spacecraft from Equation 5.2 using the previous value of their respective positions.

$$\vec{g}_R = \frac{\mu}{r_R^3}\vec{r}_R \tag{5.3}$$

$$\vec{g}_{s/c} = \frac{\mu}{r_{s/c}^3}\vec{r}_{s/c} \tag{5.4}$$

Equation 5.1 can now be rearranged and simplified by using the values of $\vec{g}_R$ and $\vec{g}_{s/c}$ from Equations 5.3 and 5.4 and substituting $\vec{a}_R$ and $\vec{a}_{s/c}$ for the second-order differentials. The resulting equations represent the accelerations of the chaser and target, respectively.

$$\vec{a}_R = \vec{a}_T - \vec{g}_R \tag{5.5}$$

$$\vec{a}_{s/c} = -\vec{g}_{s/c} \tag{5.6}$$

The 3-DOF Dynamics Model outputs the current velocities and positions of the chaser and target spacecraft by twice numerically integrating the two differential equations (Equations 5.5 and 5.6). The numerical integration performed in this simulation is governed by a variable step-size Runge-Kutta fourth- and fifth-order integration scheme.

5.1.4 Guidance Model

Once the current position and velocity for the chaser and target spacecraft have been calculated, the task of guiding the chaser on to the rendezvous trajectory can be undertaken. The guidance model is depicted in Figure 5-4 where the Augmented Lambert Guidance Algorithm is detailed in Chapter 4. Per the discussion in the previous chapter, the Guidance Model takes inputs of the current position and velocity as computed by the 3-DOF Dynamics Model. From these values and $M_{4^{th}}$ from the Vehicle Staging Model,

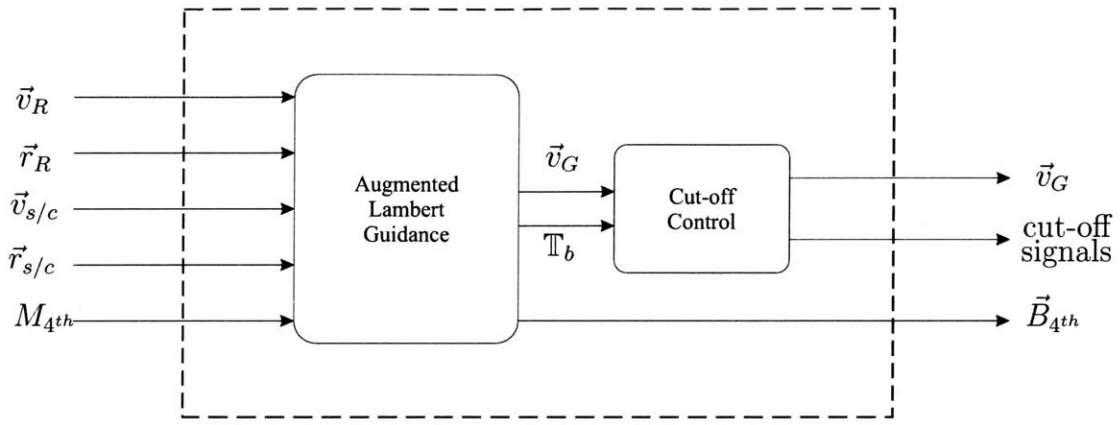

Figure 5-4: Guidance Model Diagram

the ALGA outputs the values for $\vec{v}_R$, $\vec{v}_G$, $\mathbb{T}_b$, and $\vec{B}_{4th}$ to be used in the cut-off controller and the Steering Model. In addition, quantities such as $\vec{r}_{Tmod}$ and $\Delta\vec{v}$ are saved to be used the next time ALGA is called.

To limit the amount of computing power needed to run the ALGA, it is called at a rate of 2 Hz. Consequently, the $\vec{v}_G$ is updated every half-second. The ALGA is called at this rate until $\vec{v}_G$ is less than 100 m/sec. After this benchmark, the ALGA is called at a rate of 100 Hz. Figure 5-5 shows an example of how $\vec{v}_G$ is affected by the change in the rate at which the ALGA is called.

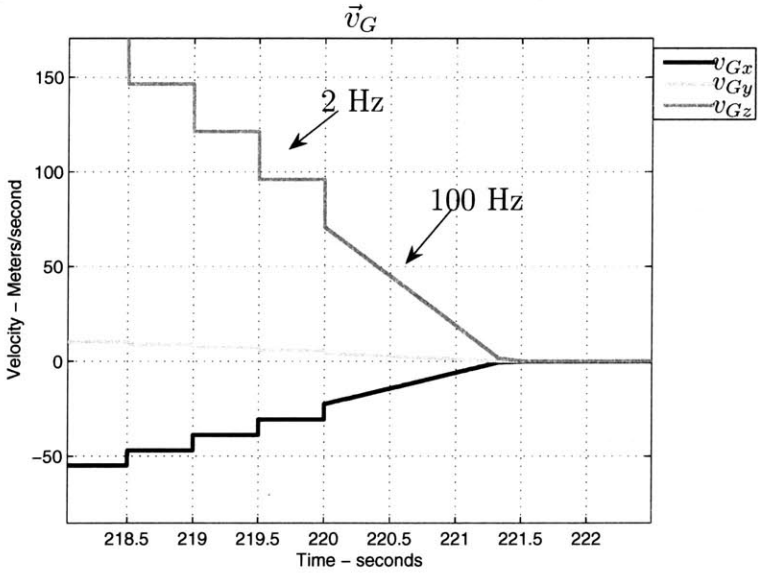

Figure 5-5: Transition of Execution Rate

This increase in rate is done to get a more accurate value of $\vec{v}_G$ around the time that the boost stages are cut-off by the cut-off controller and the coast phase begins. A perfect cut-off controller would be able to terminate thrusting when $\vec{v}_G$ is exactly zero; in practice, this is very difficult to do. Therefore, the boost is cut-off when $\vec{v}_G$ falls below a predetermined value. The cut-off value is dependant on how often $\vec{v}_G$ is computed and the amount of thrust being provided by the engine. Given the specifications tabulated in Section 3.3.1, the 3rd stage is jettisoned when $\vec{v}_G$ is reduced to a value below 2.0 m/sec. The importance of reducing the velocity error due to cut-off is discussed in Section 5.2.3.

Immediately after the 3rd stage jettison, the 4th stage ignites to reduce $\vec{v}_G$ at a slower rate due to significantly lower thrust. When $\vec{v}_G$ is reduced to a value below 0.10 m/sec, the direction of $\vec{v}_G$ is held fixed, or frozen. From this point on, the controller runs at a rate of 1000 Hz, but the ALGA stops running and constantly outputs the last calculated values for $\vec{v}_G$, $\mathbb{T}_b$, and $\vec{B}_{4^{th}}$. The remaining magnitude of $\vec{v}_G$ is reduced further by subtracting the change in chaser velocity over the time interval between runs from the magnitude of $\vec{v}_G$. This is done until the magnitude of $\vec{v}_G$ is below 0.02 m/s. A plot of $\vec{v}_G$ versus time through the 3rd stage jettison and 4th stage correction burn is shown in Figure 5-6.

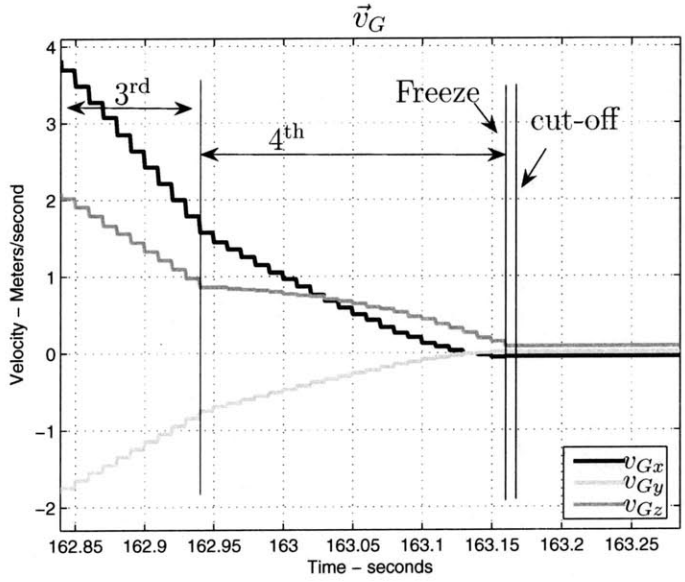

Figure 5-6: Execution of 3rd Stage Cut-off and 4th Stage Correction

Meanwhile, the cut-off controller is sending signals to the stage controller in the Vehicle Staging Model. The cut-off controller signals to jettison the 3rd stage and ignite the 4th stage and again signals to cut-off the 4th stage. After the 4th stage correction burn and the subsequent coast phase, the initiation of the 4th stage manoeuver at time $\mathbb{T}_{4th}$ is also signaled by the cut-off controller by using the value of $\mathbb{T}_b$ from the ALGA.

5.1.5 Steering Model

For the first-order evaluation being carried out in this simulation, the physical characteristics and limitations of the thrust vector control mechanisms for each of the four rocket engines is not being modeled. Therefore, instead of having separate steering and flight control subsystems, they are both combined into one model. Hence, the output of the steering model is simply a unit vector defining the thrust direction, $\vec{B}$. The simulation assumes that the actual thrust changes instantaneously to match the direction output by the steering model. Figure 5-7 shows the steering model layout.

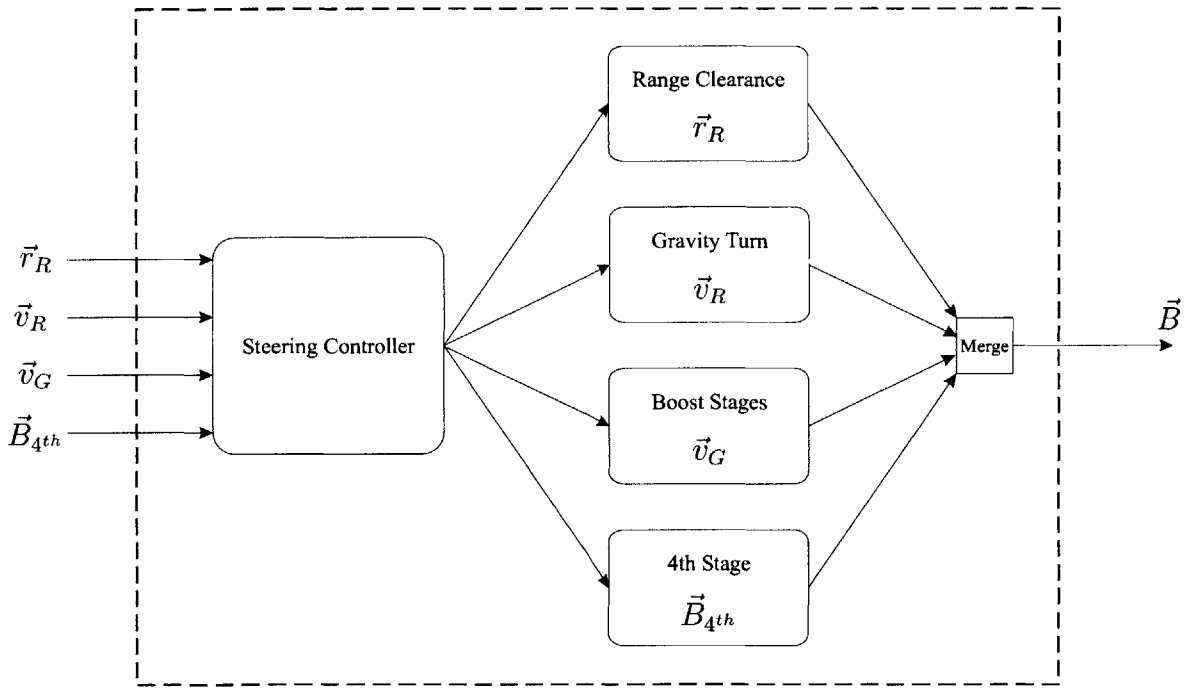

Figure 5-7: Steering Model Diagram

The Steering Model is comprised of the steering controller and four subsystems that dictate which value is used for the steering direction, $\vec{B}$. The steering controller outputs an execution signal to the particular subsystem based on the current time of flight and also includes the necessary input value: $\vec{r}_R$, $\vec{v}_R$, $\vec{v}_G$, or $\vec{B}_{4^{th}}$. The value for $\vec{B}$ is determined by the logic contained in the particular subsystem signaled by the steering controller.

Range Clearance & Gravity Turn

Even though the specific range safety requirements imposed at any Earth-based launch site are not modeled in this simulation, the steering controller implements an initial phase that mimics a possible range safety trajectory.

Whether the chaser is launched from either a land- or sea-based platform, the rocket is configured to launch vertically. From ignition, the steering controller positions the thrust vector perpendicular to the launch platform making $\vec{B}$ align with $\vec{r}_R$. Therefore, $\vec{B}$ is defined by a unit vector in the direction of $\vec{r}_R$.

After 10 seconds, the chaser performs a gravity turn. A gravity turn is performed by aligning the thrust vector with the velocity vector of the rocket; hence, $\vec{B}$ is defined by the unit vector in the direction of $\vec{v}_R$. The gravity turn is performed for an additional 10 seconds after the vertical portion. The mock range clearance manoeuver profile is shown in Figure 5-8.

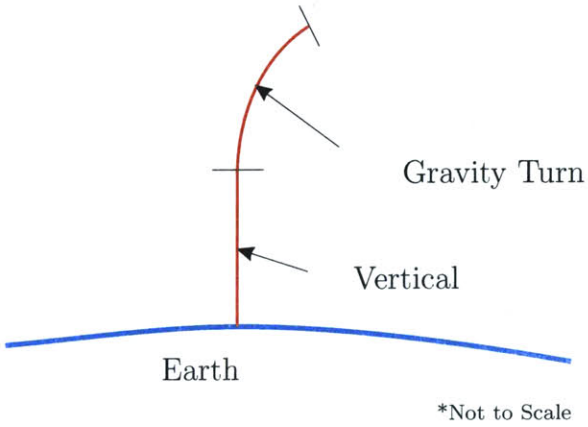

Figure 5-8: Mock Range Clearance Manoeuver

Boost Steering

After the 20 second clearance manoeuver, the chaser begins to steer as guided by the output of the guidance system. Recalling from Sections 2.2 and 5.1.4, the Guidance Model outputs the velocity-to-be-gained, $\vec{v}_G$, which is the instantaneous change in velocity needed to get the chaser on the correct transfer trajectory. In practice, however, the spacecraft's velocity cannot instantaneously change by $\vec{v}_G$ to match the correlated velocity; therefore, the spacecraft will have to thrust to change it's velocity.

The Steering Model is required to change the thrust direction in a way that will reduce $\vec{v}_G$ over time without specific knowledge of the thrust produced by the rocket engines due to the potential variabilities. To this end, the Steering Model controls the rocket during boost by producing a steering command that will align the thrust vector with $\vec{v}_G$, subsequently reducing it. Figure 5-9 shows the results of aligning the thrust vector with $\vec{v}_G$ for a particular trajectory.

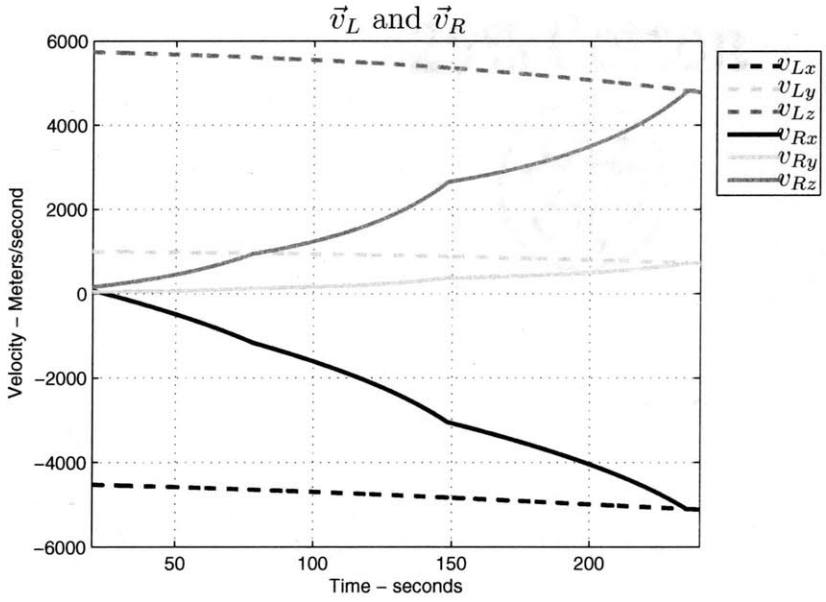

Figure 5-9: Matching $\vec{v}_L$ by Reduction of $\vec{v}_G$ Over Time

In this example, the velocity for this spacecraft starts at zero and increases due to the engines thrusting. Even though the value of $\vec{v}_L$ changes over time, the thrust provides

more than enough acceleration to steadily decrease the difference between the Lambert solution and the spacecraft velocity over successive time steps.

Once $\vec{v}_G$ is below a predetermined threshold, the thrust is terminated and the,spacecraft coasts on the orbit found by the augmented Lambert Guidance Algorithm until it is time to perform the 4^{th} stage burn.

4^{th} Stage Manoeuver

Subsequent to the boost and the coast periods, the chaser begins the 4^{th} stage manoeuver to match the position and velocity of target spacecraft. During this manoeuver, the steering controller aims the thrust vector in the direction dictated by the ALGA, $\vec{B}_{4th}$. The steering vector remains constant throughout the entire 4^{th} stage manoeuver until the rendezvous time is reached and the 4^{th} stage manoeuver is complete.

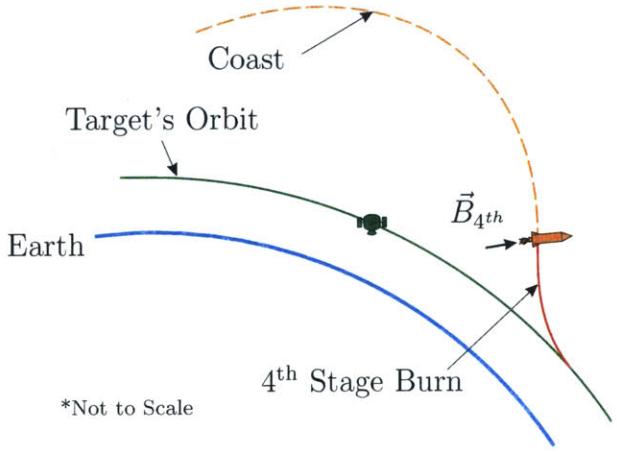

Figure 5-10: 4^{th} Stage Manoeuver

5.2 Simulation

After the development of the equations and logic contained in the four functional models, the next step is to integrate all of the components together into a cohesive unit known as the simulation. The software used to implement the logic and equations defined by

the four functional models is the same software used to execute the interactions between those models over time. This software made by MathWorks, Inc. is called Simulink and it works with another MathWorks program, MATLAB to implement the logic and produce useable results. These two programs work in concert to pass information back and forth performing the calculations outlined in the models over time while saving information necessary to evaluate the ALGA. The technical specifications of these two programs are:

Simulink: Version 6.1 (Release 14) with Service Pack 1

MATLAB: Version 7.0.1.24704 (Release 14) with Service Pack 1

These programs were executed on a computer using the LINUX operating system.

Using this combination of hardware and software, a first-order, 3-DOF simulation was created. This simulation consists of a four-stage launch vehicle that, using the ALGA developed in Chapter 4, rendezvous's with a spacecraft orbiting the Earth. Included in this section is a brief discussion of how the models were implemented using a combination of Simulink blocks and functions written in C-code. In addition, the method of interaction between Simulink and MATLAB is presented. After the implementation, the simulation was put through several tests to validate the accuracy of the entire simulation by looking at the models and the simulation as a whole. After the validation, the portions of the simulation that significantly contribute to the terminal state deviation of the chaser from the nominal trajectory are identified and discussed.

5.2.1 Implementation

The first step in creating the simulation used to evaluate the ALGA is to take the equations and logic contained in the model descriptions and translate them into a form that Simulink can understand. Once this is done, MATLAB is used to interface with Simulink by supplying the inputs and collecting the outputs.

Model Translation

The earlier model discussion simply outlined the equations or logic used to manipulate the given inputs and produce the given outputs. It is in Simulink where these equations and logic are executed by using a combination of pre-defined blocks, which are built into Simulink, and defining new blocks that use functions written in C-code in a form Simulink can understand.

Figure 5-11 shows the 3-DOF Dynamics Model as a representation of how the models are implemented in Simulink. The inputs and outputs are represented by ovals as shown in the diagram. The 'Multiplication & Division' block computes the value of $\vec{a}_T$ as given in Equation 5.2. This value is then given to the GSLV Dynamics Subsystem. Subsystem blocks are used to simplify the diagrams in Simulink. In this model, there are two subsystems, one to calculate the position and velocity of the chaser and the other to calculate the position and velocity of the target spacecraft.

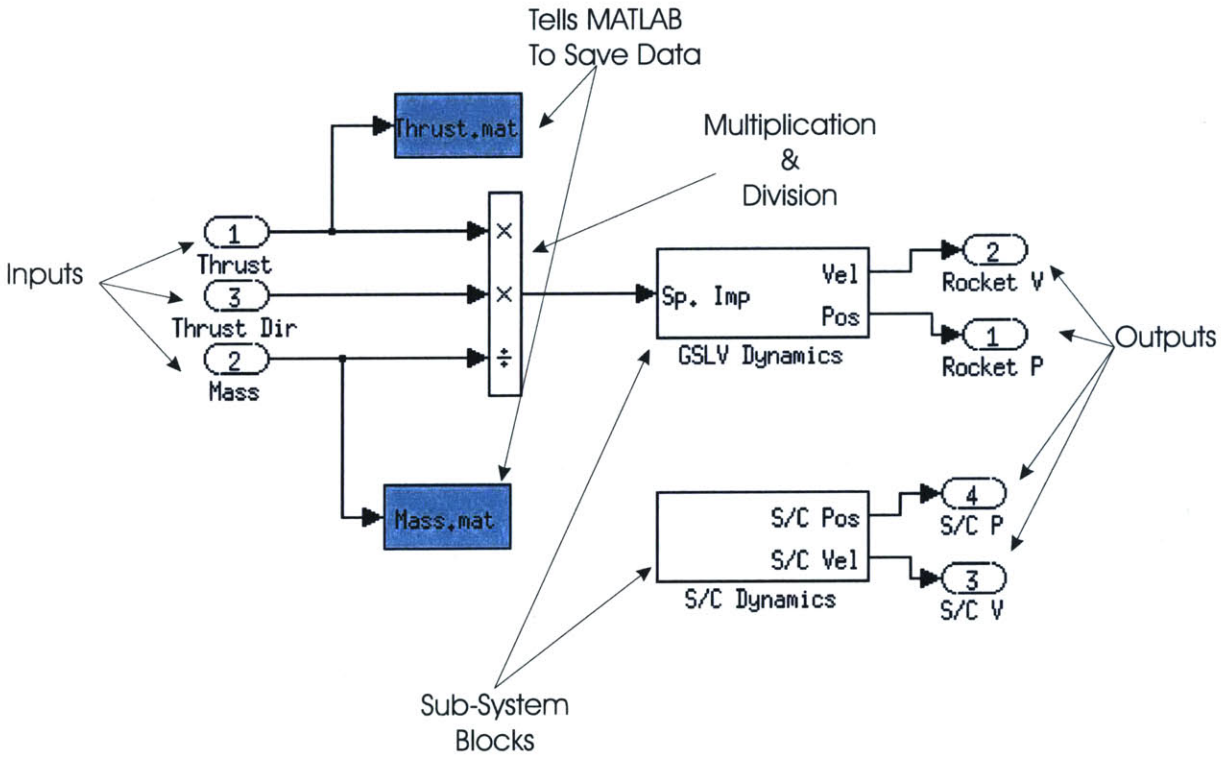

Figure 5-11: Simulink: 3-DOF Dynamics Model

The 'GSLV Dynamics' subsystem is shown in Figure 5-12. The 'Spacecraft Dynamics' subsystem is similar to the 'GSLV Dynamics' Subsystem and is not shown. It is in these subsystem blocks where the double integration is carried out. First the 'Addition & Subtraction' block computes $\vec{a}_R$ from Equation 5.5, then the integration is completed using Simulink's 'Integration' blocks as shown in the diagram. The 'Earth Model' subsystem contains the blocks necessary to calculate $\vec{g}_R$ from Equation 5.3. Once all of the calculations in the 'GSLV Dynamics' subsystem are completed, the position and velocity of the chaser are output back to the 3-DOF Dynamics Model, which is then passed to the Guidance Model. The 'Spacecraft Dynamics' subsystem similarly finds $\vec{a}_R$ and $\vec{g}_R$ and does the integration to find the position and velocity of the target spacecraft.

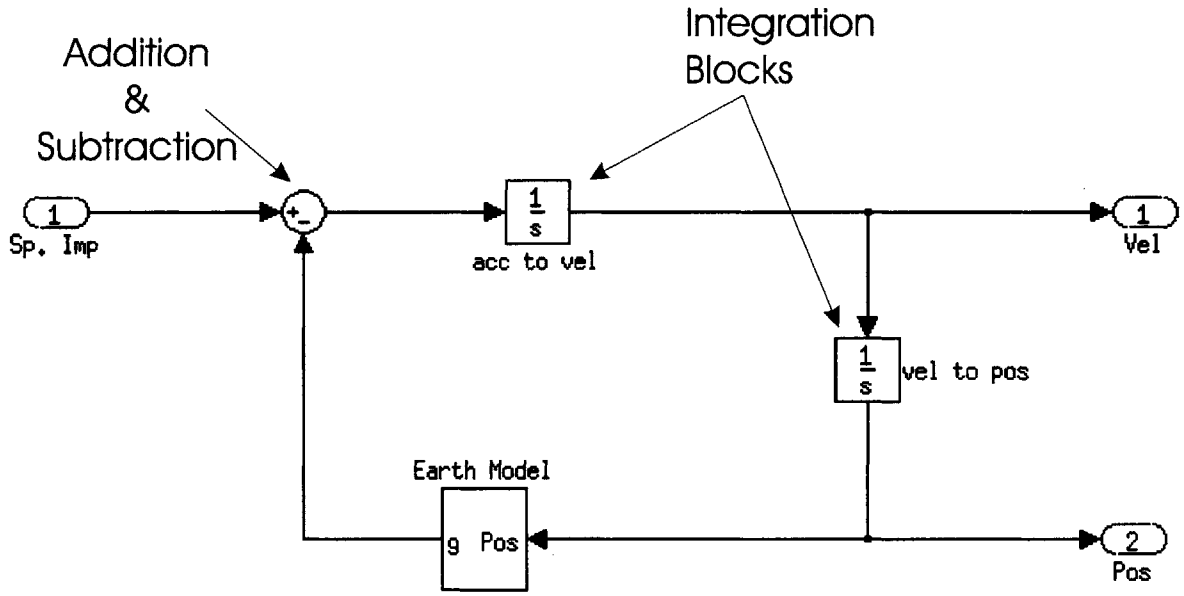

Figure 5-12: Simulink: GSLV Dynamics Subsystem

The remaining models are implemented in a similar fashion. Most of the components of the models were implemented using native Simulink blocks; however, the LGA, stage controller, and cut-off controller were written in C-code and new Simulink blocks were created to integrate these three functions into the simulation. After the individual models were created in Simulink, they were connected together to complete the simulation.

As stated before, data is passed back and forth between MATLAB and Simulink using a combination of MATLAB scripts and specific blocks in Simulink. In this simulation, the initial conditions, as described in Section 3.2.1, were given to the simulation via a MATLAB script and variables on the MATLAB workspace. A MATLAB script generated the initial conditions for the many test cases used to evaluate the ALGA, which were then delivered to the MATLAB workspace. Simulink can then grab this data, i.e. the target position, from the workspace by using a 'From Workspace' block. The target position is then used as dictated by the models. Similarly, Simulink has several ways to extract data from the simulation and give it back to MATLAB. There are two blocks that convert the data from Simulink into a form that MATLAB can use. Once the data is converted it can be loaded back into the MATLAB workspace and other MATLAB scripts can be used to create plots to view the data.

An example of one method to exchange information is shown in Figure 5-11. The 'Thrust.mat' and 'Mass.mat' Simulink blocks convert the thrust and mass data into MATLAB data files and saves them to the current directory under the names 'Thrust.mat' and 'Mass.mat', respectively. The other method of converting data is to use a 'To Workspace' block, which is not pictured. This block saves the data directly to the MATLAB workspace with the variable name given in the block.

Subsequent to implementing the models into Simulink, creating MATLAB scripts to pass necessary information to Simulink, and providing a means to get data out of Simulink, the task of validating the simulation can begin.

5.2.2 Validation

The validation of the models used in the simulation is carried out to confirm that the implementation of the model into Simulink accurately represents the equations derived in the model descriptions as given in Section 5.1. The execution of the model validation is the most crucial step in assuring that the evaluation of the ALGA is valuable and

accurate.

There are many possible validation methods, i.e., comparing results to flight data, comparing results to cases which can also be solved analytically, etc. The method used to validate this simulation is to run several trajectories using a Lambert Guidance Algorithm, which has already been verified to produce sufficiently accurate outputs. Then, the LGA is replaced by the ALGA and more trajectories are simulated to validate the parts of the model associated with the ALGA. Several simulations are run and the data is then plotted by MATLAB so that it may be analyzed. The model is considered validated if the logic in the model description is adhered to or the data plotted by running multiple simulations can be checked against known values or values that can be calculated analytically.

Running intercept trajectories using an LGA will test the Vehicle Staging Model, the 3-DOF Dynamics model, the cut-off controller in the Guidance Model, and the Steering Model except the 4^{th} stage subsystem; essentially, all of the simulation parts are tested besides those dealing with the 4^{th} stage manoeuver and the ALGA.

The next step is then to validate the parts of the model associated with the 4^{th} stage manoeuver and the ALGA including the execution of the 4^{th} stage burn at the appropriate time, $\mathbb{T}_{4^{th}}$, and in the correct direction, $\vec{B}_{4^{th}}$. This involves replacing the LGA with the ALGA and again running several trajectories to validate the rest of the simulation. The steps to validate each model are briefly discussed here.

A sample intercept trajectory is shown in Figure 5-13 as a representative of the several trajectories used to validate the Vehicle Staging Model, 3-DOF Dynamics model, the cut-off controller in the Guidance Model, and the Steering Model as discussed previously.

For completeness, the initial and target conditions are also listed here.

$$\begin{aligned} \vec{r}_T &= [2097576\ -2717648\ 5375425]\ \text{m} \\ \vec{r}_R(0) &= [-2668983\ -4510008\ 3635517]\ \text{m} \\ \vec{v}_R(0) &= [0\ 0\ 0]\ \text{m/sec} \\ \mathbb{T} &= 1623\ \text{sec} \end{aligned}$$

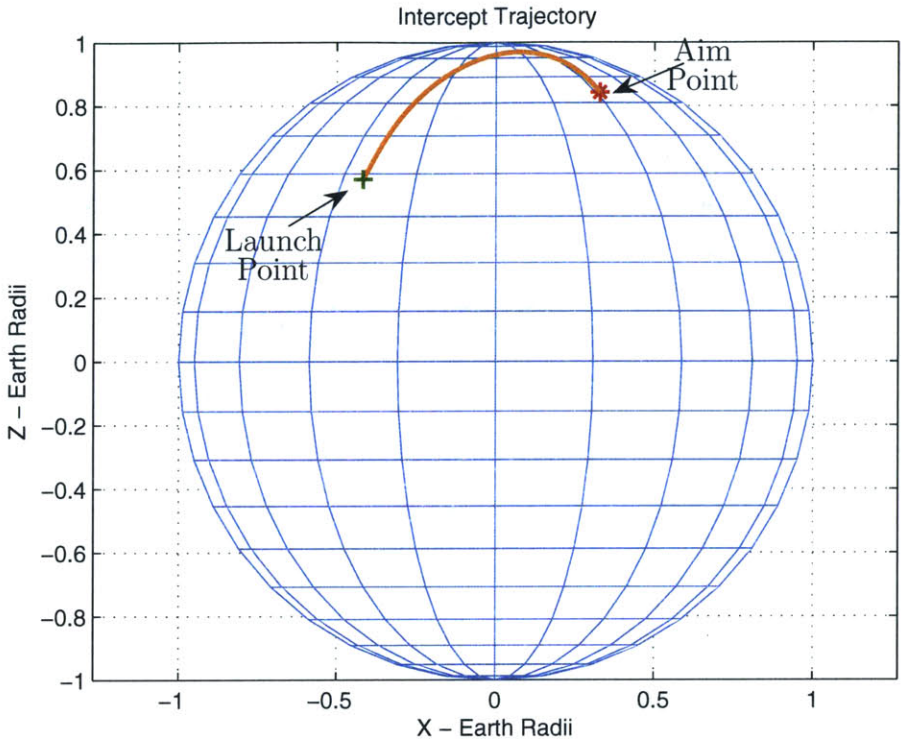

Figure 5-13: Intercept Trajectory

Vehicle Staging Model

The quantities T and M_R are the two outputs from the Vehicle Staging Model. From the vehicle specifications outlined in Section 3.3.1, the values of T and M_R can be analytically determined over time and compared against the output of the model.

Because the quantities can be analytically computed, a plot of T and M_R versus time is a simple and accurate method of validating this model. Furthermore, as can be seen from the plots in Figure 5-14, the stage controller executed its logic successively, i.e., the previous stage mass is jettisoned before the next engine ignites, etc.

3-DOF Dynamics Model

The motion of a non-thrusting vehicle over time when acted upon only by gravity is well known, when using the two-body approximation, and can be analytically calculated. Using this information and recognizing that the LGA used has already been verified to

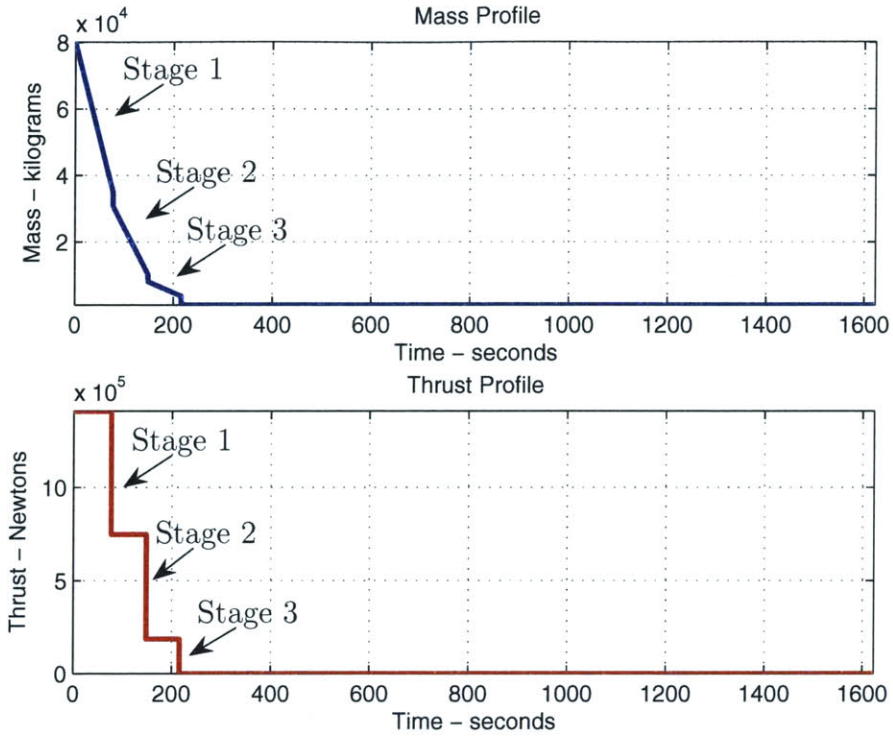

Figure 5-14: Mass and Thrust Profiles for Intercept Sample

produce accurate results, enough information is known to begin to validate the 3-DOF Dynamics Model for both the coast and boost phases.

Per the discussion in Chapter 2, given the chaser's current position and velocity, the LGA outputs the correlated velocity needed to have the chaser be on an intercept trajectory with the target. Hence, the model was validated by taking the position and velocity of the chaser as calculated by the 3-DOF Dynamics Model and feeding it into the LGA. The LGA then produces a correlated velocity which is used in conjunction with the position and the remaining $\mathbb{T}$ to predict the final position and velocity of the chaser using Lagrange Coefficients. The 3-DOF Dynamics model is considered validated if the predicted final position is the same as the target position.

The result of executing this procedure over the entire $\mathbb{T}$ and for several different trajectories showed that the position and velocity were calculated by the 3-DOF Dynamics Model accurately to the third decimal place. This small difference in the position and

velocity of the chaser compared with the nominal values does not significantly contribute to the terminal state deviation, which will be discussed in more detail in Section 5.2.3.

Cut-off Controller & Boost Steering

The job of the cut-off controller in conjunction with the boost steering is to control the thrust direction of the chaser during boost to reduce the difference between the correlated velocity and the chaser's velocity over the course of the boost phase. This is done by using the logic as described in Sections 5.1.4 and 5.1.5.

Considering the values of $\vec{v}_L$ and $\vec{v}_G$ produced by the LGA are accurate, and after validating the 3-DOF Dynamics model, simple plots can be used to validate the execution of the cut-off controller and boost steering. A comparison of $\vec{v}_L$ and $\vec{v}_R$ versus time is shown in Figure 5-15, while the reduction in $\vec{v}_G$ over time is shown in Figure 5-16.

These figures show that the velocity of the chaser reaches the correlated velocity given by the LGA and, accordingly, $\vec{v}_G$ is reduced. Then, the boost phase is terminated, resulting in the velocity of the chaser closely following $\vec{v}_L$ for the rest of the coast phase.

A closer inspection of the cut-off time for the several trajectories show that, indeed, the cut-off controller and boost steering execute as expected to reduce $\vec{v}_G$ below the cut-off value of 0.02 m/sec.

A plot of the components of the steering vector, $\vec{B}$, during the boost phase also verify that the boost steering is executing as expected. For the first 20 seconds, the boost vector steers vertically and performs a gravity turn. For the rest of the boost phase, the boost vector aligns itself with the velocity-to-be-gained vector, where v_{Gx}, v_{Gy}, and v_{Gz}, denote the components of the unit vector of $\vec{v}_G$ as shown in Figure 5-17.

Now the parts of the simulation not dealing with the 4^{th} stage manoeuver have been sufficiently tested and deemed adequately accurate. The task is now to test the parts of the simulation associated with the 4^{th} Stage Manoeuver.

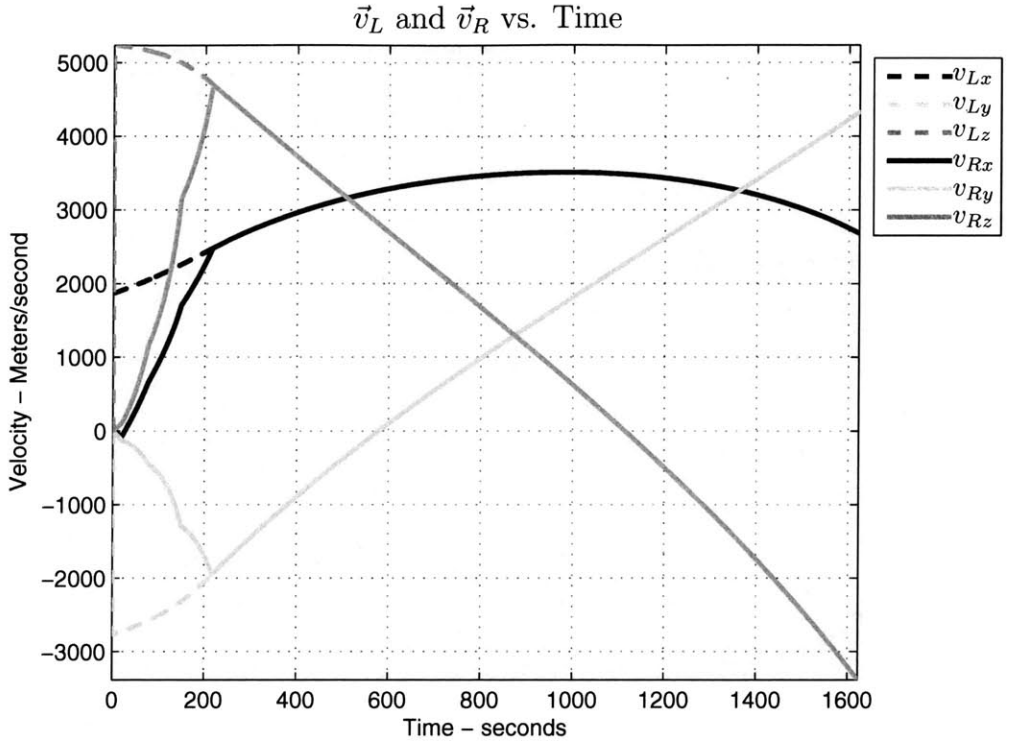

Figure 5-15: $\vec{v}_L$ vs. $\vec{v}_R$ for Intercept Sample

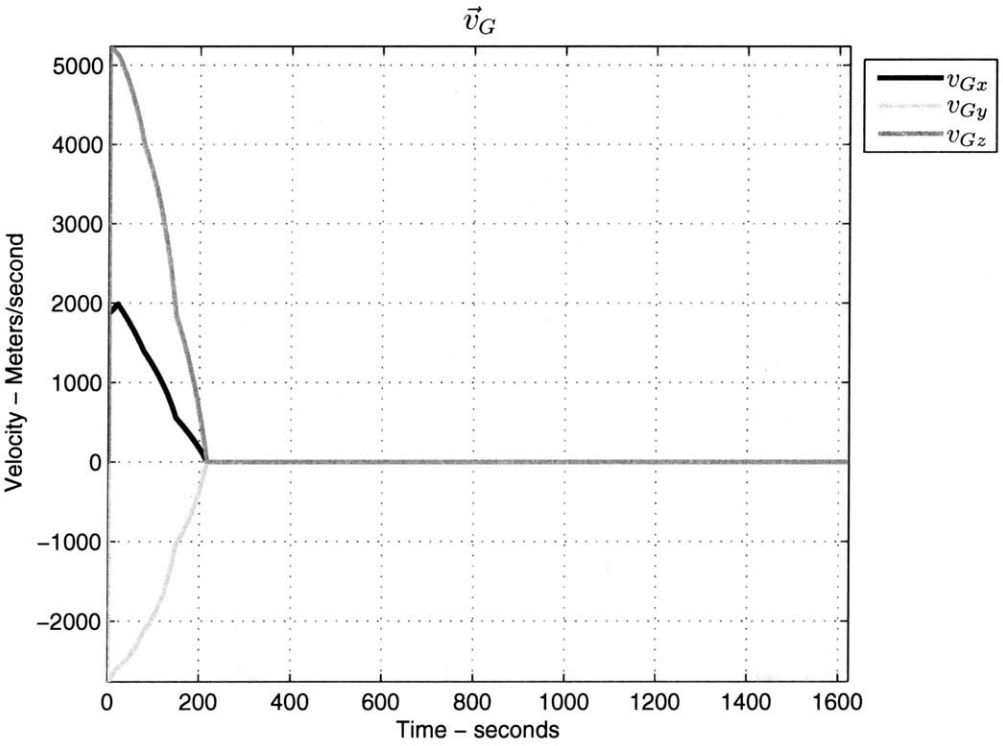

Figure 5-16: $\vec{v}_G$ for Intercept Sample

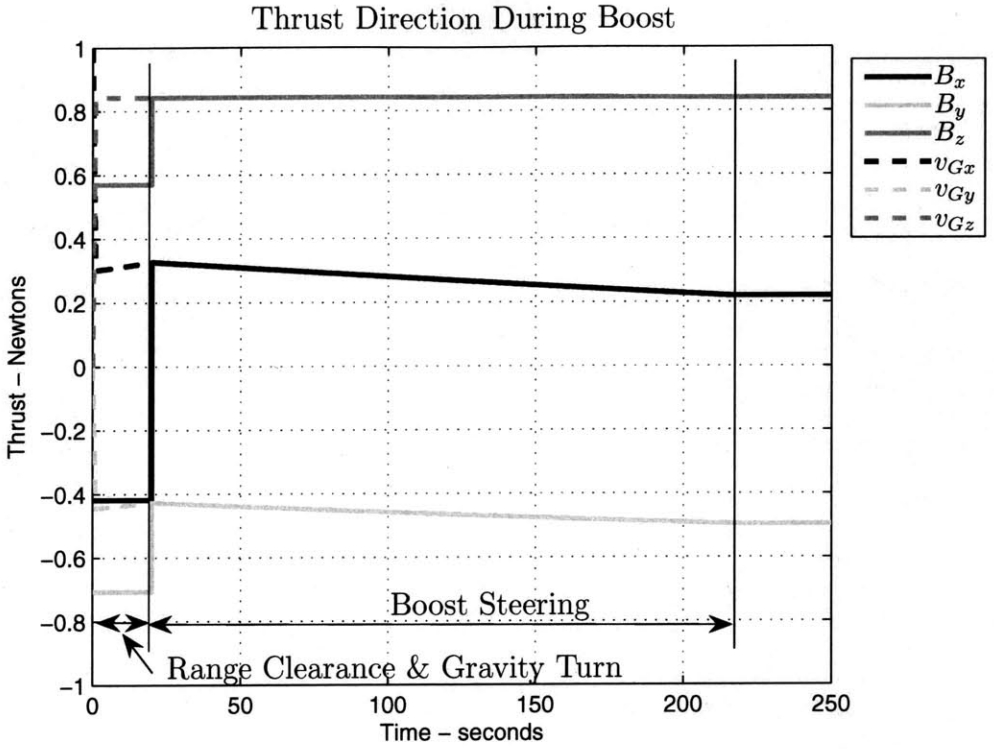

Figure 5-17: $\vec{B}$ for Intercept Sample Trajectory

Stage Controller & 4th Stage Steering

It is the job of the ALGA to compute both the 4th stage ignition time, $\mathbb{T}_{4th}$, and steering vector for the 4th stage manoeuver, $\vec{B}_{4th}$. Therefore, the validation of the 4th stage components of the Steering Model and the stage controller can only be accomplished after the LGA is replaced by the ALGA in the simulation.

A plot of the steering vector, $\vec{B}$, versus time shows whether or not the 4th stage manoeuver was executed and if it was executed at the correct time and proper direction as dictated by the values given by the ALGA. Figure 5-18 shows $\vec{B}$ over the entire mission time for one particular rendezvous trajectory.

By comparing the plot in Figure 5-18 with the thrust profile (not shown) it can be confirmed that the simulation executed the 4th stage manoeuver after a time of coasting. In addition, $\vec{B}_{4th}$ remained constant during the entire 4th stage burn. Checking the plotted values against those given by the ALGA verifies that the values of $\mathbb{T}_{4th}$ and $\vec{B}_{4th}$ are

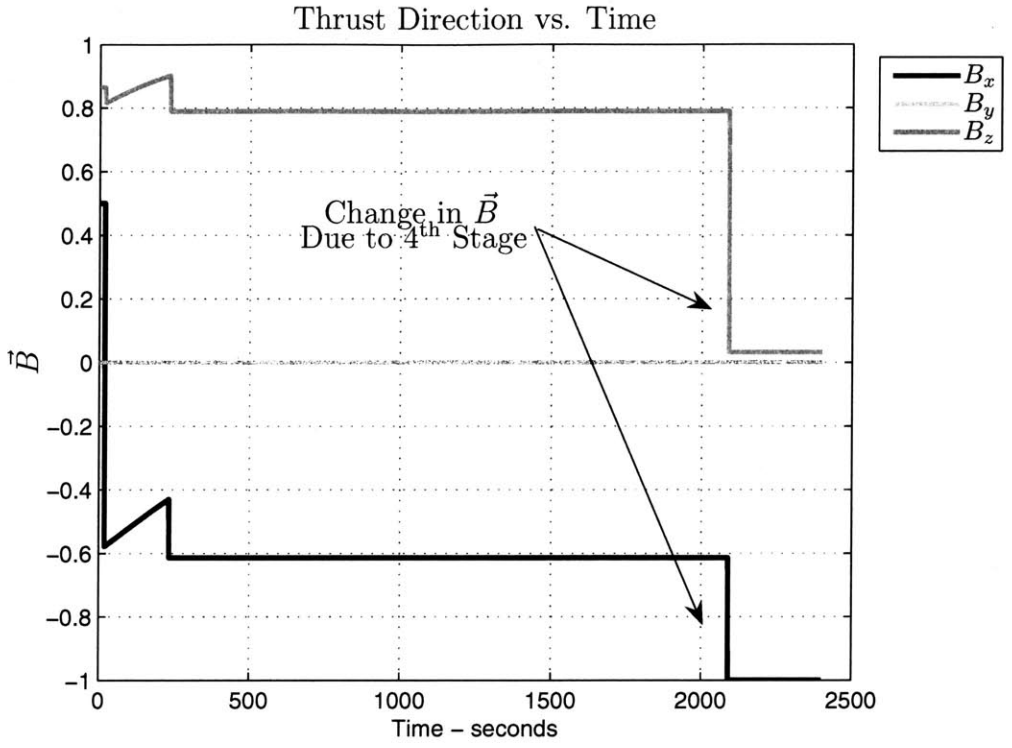

Figure 5-18: Change in $\vec{B}$ for 4^{th} Stage Burn

executed according to the values output from the ALGA. Consequently, the interactions between the 4^{th} stage components of the Steering Model and the stage controller work as expected.

5.2.3 Sources of Terminal State Deviation

The terminal state deviation is the difference between the positions and velocities of the chaser and the target spacecraft at the rendezvous time, $\mathbb{T}$, and is quantified by the terms R_{err} and V_{err}. While some of the terminal state deviation is due to the inaccuracy of the logic in the ALGA, any contribution from other sources must be identified to better understand the results as they relate to the computations in the ALGA.

For any algorithm or simulation, there is assured to be some amount of calculation inaccuracy in the output. In fact, the LGA used in this thesis has its own inherent inadequacies when calculating the values of $\vec{v}_G$ and $\vec{v}_L$, although they are small and

contribute to the terminal state deviation in the third decimal place as stated in Section 5.2.2.

During the course of the model validation, potential sources that significantly contribute to the terminal state deviation were identified and are listed here and discussed in more detail below.

1. The inability to cut-off the boost thrust at the exact value of $\vec{v}_L$ as dictated by the ALGA.

2. The inability to initiate the 4^{th} stage burn at the exact ignition time, $\mathbb{T}_{4^{th}}$.

3. The discrepancy in the 4^{th} stage mass due to the correction burn near cut-off.

Cut-Off Sources

The value of $\vec{v}_G$ at cut-off is a measure of how accurately the path of the chaser follows the transfer trajectory dictated by the ALGA. Hence, the smaller the value of $\vec{v}_G$, the more accurately the chaser will follow the correct transfer trajectory. Having any amount of inaccuracy in the velocity at cut-off results in the chaser coasting on a different transfer trajectory where the position and velocity of the chaser constantly diverge from the nominal values over time. During the course of the coast phase, the difference in position of the chaser compared to the nominal trajectory may grow to be on the order of hundreds of meters while the difference in velocity reaches values on the order of tens of meters per second, depending on the cut-off value of $\vec{v}_G$. Therefore reducing $\vec{v}_G$ to a small value before the termination of the boost phase is very important to the success of the trajectory. This is the reasoning behind the logic of the cut-off controller as described in Section 5.1.4.

Using the several trajectories used to validate these models, it was found that an acceptable cut-off value for $\vec{v}_G$ is below 0.02 m/sec, as stated in Section 5.1.4. By reducing $\vec{v}_G$ to below this value, it was found that the resulting terminal state deviation for the

range of coast times used in this thesis are on the order of 10 m and less than 0.1 m/sec, respectively.

4^{th} Stage Ignition Timing

Along with the velocity-to-be-gained, the ALGA outputs the burn time, and thrust direction vector of the 4^{th} stage manoeuver. This burn time, $\mathbb{T}_b$, is used by the cut-off controller to calculate the ignition time of the 4^{th} stage engine, which is passed to the stage controller by way of the cut-off signals and is used to tell the engine when to start burning fuel.

As stated in Section 5.1.4, the maximum rate at which the simulation is called is 1000 Hz. Therefore, the minimum lapse in time between when the cut-off signals are updated and when these updated signals are read by the stage controller is 0.001 seconds. Assuming that the burn time is calculated to more than three decimal places of accuracy, there is an associated velocity discrepancy of at most 0.0009 m/sec by the time the 4^{th} stage burn is executed. This may seem like a trivial amount of time, but with speeds ranging from 3 to 7 km/sec, this discrepancy translates in to a deviation of approximately 3 to 7 meters for terminal position alone.

4^{th} Stage Mass Difference

During the first three boost phases, the Vehicle Staging Model does not know how much fuel is going to be expended for the short 4^{th} stage correction burn as described in Section 5.1.4. Therefore, the 4^{th} stage mass ($M_{4^{th}}$) given by the Vehicle Staging Model does not account for the reduced mass due to the fuel burn. The calculation of $\mathbb{T}_b$ and $\vec{B}_{4^{th}}$ in the ALGA along with the state propagation by the Runge-Kutta integrator depend on $M_{4^{th}}$, which is given as 1100 kg until a more accurate value is calculated. A more accurate calculation of $M_{4^{th}}$ isn't achieved until the end of the boost phase when the 4^{th} stage correction burn is executed. Consequently, the values of $\mathbb{T}_b$ and $\vec{B}_{4^{th}}$ have a

computational inaccuracy proportional to the mass of the fuel expelled during the 4^{th} stage correction burn, which is approximately 1 to 2 kg. This mass difference is small compared to the total mass of the 4^{th} stage, but this difference in mass still effects the terminal state deviation of the chaser.

Chapter 6

ALGA Performance Analysis

After the development of the Augmented Lambert Guidance Algorithm, as described in Chapter 4, the task of creating a computer simulation to evaluate the robustness, versatility, and overall ability of the ALGA was undertaken. Chapter 5 illustrated the models created to mimic the physical characteristics of the chaser, target spacecraft, and the environment in which they both operate. These models were then implemented over time through the use of simulation software. Once the models and the simulation were completed, the evaluation of the ALGA was able to commence.

The evaluation consists of quantifying how well the ALGA completes the task of getting the chaser to rendezvous with a target spacecraft in orbit around the Earth. To this end, a multitude of test cases were created to survey the expected operational region of the rendezvous missions. But first, the boundaries of this region must be defined. The boundaries are characterized by several parameters, including the range of mission times, the likely target orbits, and the possible approach paths that the chaser may take to get to the target spacecraft. Once the range of parameters have been found, the test cases are created.

After each of these test cases is run, the positions and velocities of both the chaser and target spacecraft are found at the rendezvous time to calculate R_{err} and V_{err}. The

performance of the ALGA is then measured with respect to the success criteria outlined in Chapter 3. The results of running these various test cases are then plotted and discussed.

6.1 Operational Region

The definition of the operational region is primarily based on the discussion of the possible types of missions that would use the ALGA, as described in Section 1.1. Conforming to these mission descriptions, the mission timing and the characterization of the target orbits are the two major contributors to the boundaries of the operational region.

Because the 4^{th} stage engine has a finite amount of fuel, it is the most influential factor in determining whether the chaser can rendezvous with the target spacecraft. No matter how accurate the ALGA computes the values of $\vec{B}_{4^{th}}$, $\mathbb{T}_b$, and $\vec{r}_{Tmod}$, if the chaser does not have enough fuel to burn for the entire predicted burn time, the chaser will never rendezvous with the target spacecraft.

Because the success of the rendezvous mission relies heavily on the amount of $\Delta\vec{v}$, any factor that may increase the required amount of $\Delta\vec{v}$ must be examined. Therefore, the two possible approach paths that the chaser is able to take to reach the target will also be discussed in this section.

6.1.1 Mission Timing

Recall from Section 1.1 that the expected missions employing this algorithm have the desire to reach the target expeditiously; therefore, the segment of the mission time from launch to rendezvous, $\mathbb{T}$, is chosen to be less than an hour.

Furthermore, the LGA used in this thesis has an inherent limitation that restricts the value of $\mathbb{T}$. The LGA yields inaccurate results when the coast phase time, $\mathbb{T}_{CP}$, of the transfer trajectory is less than 200 seconds. With this constraint and considering that the possible maximum time for the boost phase, $\mathbb{T}_{BP_{max}}$, is 244 seconds, the lower bound for

$\mathbb{T}$ is calculated from Equation 6.1.

$$\mathbb{T}_{min} = \mathbb{T}_{BP_{max}} + \mathbb{T}_{CP_{min}} \tag{6.1}$$

To assure that there are no calculation errors due to a coast phase time being close to the minimum, the lower bound of $\mathbb{T}$ is chosen to be greater than 444 seconds. Consequently, the evaluation of the ALGA is limited to values of $\mathbb{T}$ between 700 and 3600 seconds.

6.1.2 Target Orbits

Section 3.3.2 stated that the target orbits for the missions that use the ALGA are required to be either circular or elliptical and to have a semimajor axis between 6478 and 6878 kilometers. The resulting set of target orbits are either circular or very near circular as shown in Figure 6-1.

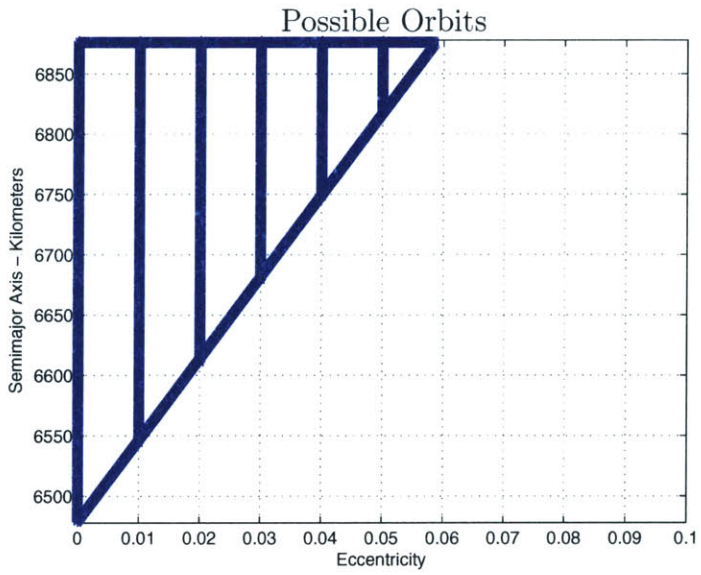

Figure 6-1: Target Trajectories That Meet Requirements

This graph was created by plotting the semimajor axis versus eccentricity of orbits who didn't intersect the surface of the Earth and had an altitude of perigee (the height

above the Earth at closest approach) greater than 100km. To reduce the number of test cases, and the ensuing number of results plots, only circular orbits were used during the evaluation of the ALGA.

Now that the target orbits have been limited to circular trajectories, the next step is to investigate the possibility of another orbital element effecting the ability of the chaser to rendezvous with the target spacecraft. There is a property of the target orbit that has a significant effect on the amount of fuel required by the chaser to rendezvous with the target spacecraft. This property is the difference in inclination of the target orbit with respect to the inclination of the chaser's transfer trajectory. As the difference in inclination between the two spacecraft grows, the required amount of $\Delta\vec{v}$ grows as well.

The primary evaluation will focus on the test cases where the chaser is launched in such a way that it does not have to make a plane change to rendezvous with the target. By eliminating the plane change, the evaluation of the ALGA will show the maximum range of successful test cases. Section 6.2.2 illustrates the difference between test cases that differ by just the inclination change.

6.1.3 Approach Paths

To achieve the rendezvous point from the launch position, there are two possible approach paths: Overtaking and Head-On. The importance in differentiating the two approach paths is derived from the amount of $\Delta\vec{v}$ that the 4^{th} stage engine must provide to the chaser. The velocity characteristics of the two approach paths are briefly discussed here.

Overtaking Trajectory

The Overtaking trajectory is shown in Figure 6-2. For the Overtaking trajectory, the chaser and target spacecraft are traveling in the same direction before the 4^{th} stage manoeuver.

In this case, the target spacecraft has a higher speed than the chaser before the 4^{th}

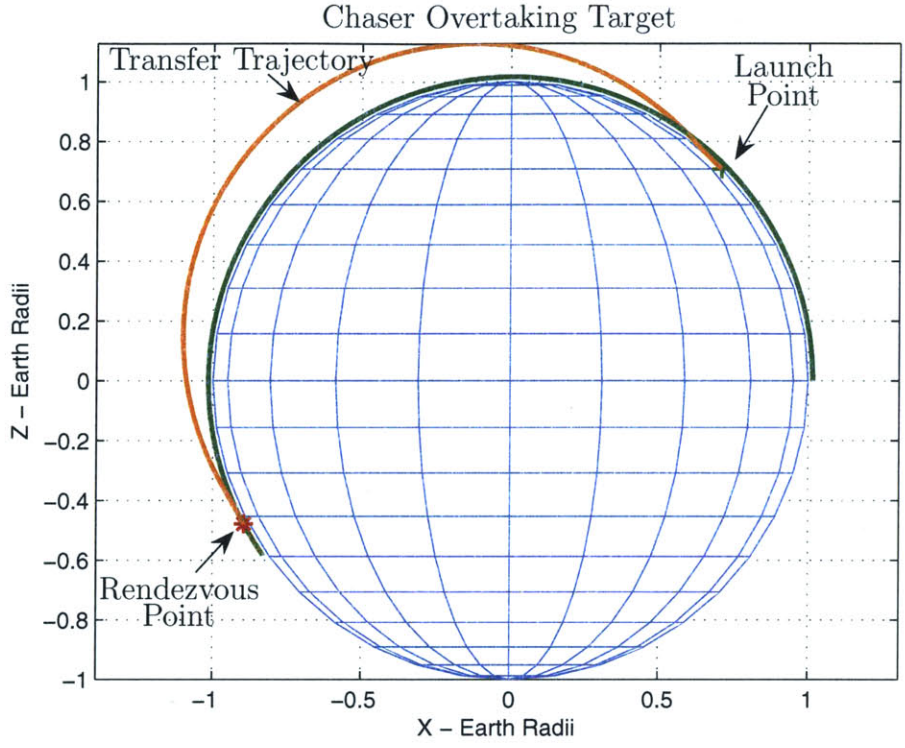

Figure 6-2: Overtaking Trajectory

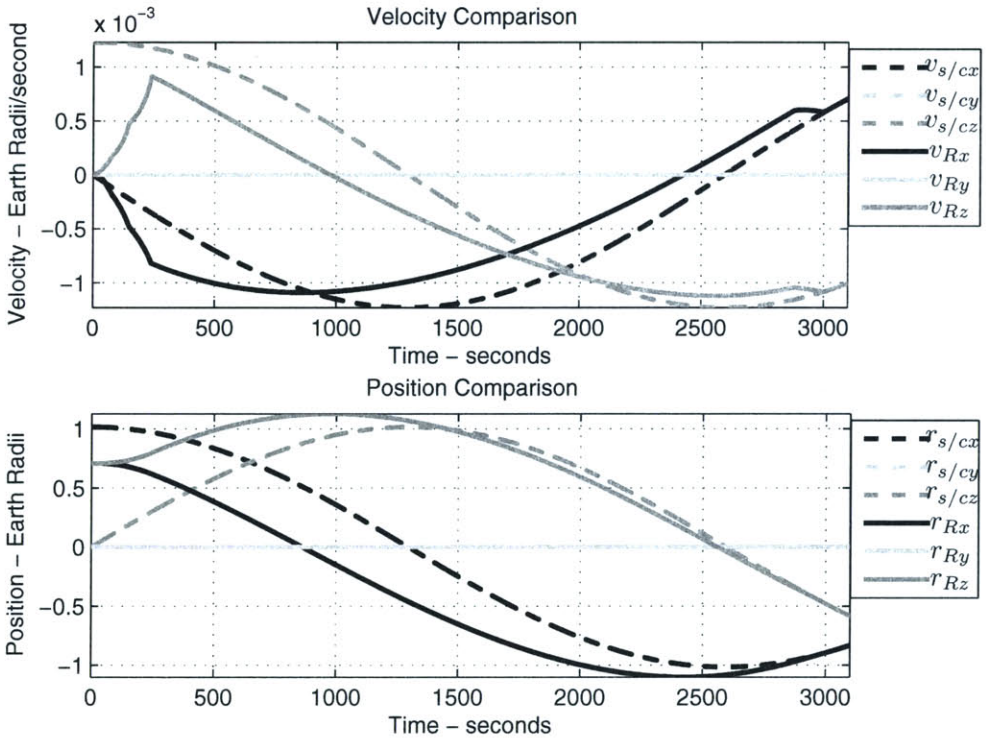

Figure 6-3: Overtaking Trajectory - Position and Velocity Comparison

stage. Therefore, the chore of the chaser's 4^{th} stage engine is to provide enough thrust to increase its speed. Figure 6-3 shows the x, y, and z components of both the velocity and position for the target spacecraft versus the chaser.

The top graph compares the velocity of the two spacecraft. For this particular set of initial conditions, the chaser ignites its 4^{th} stage engine at approximately 2800 seconds. At this time the chaser's approximate speed is roughly 7.67 km/sec while the target's is 7.84 km/sec. Additionally, the relative speed between the two spacecraft, since they are traveling in nearly the same direction, is estimated by simply finding the magnitude difference between the chaser and target spacecraft. Beginning with a relative velocity difference of 0.17 km/sec, the chaser executes the 4^{th} stage burn and both the chaser and target are traveling at the 7.84 km/sec.

Head-On Trajectory

Figure 6-4 shows the Head-On trajectory. In this situation, the chaser and target spacecraft are traveling in opposite directions toward the same rendezvous point.

Again, the target spacecraft has a higher speed than the chaser before the 4^{th} stage manoeuver and needs the thrust from the 4^{th} stage burn to match the velocity of the target. However, because the two spacecraft are traveling in opposite directions, the relative velocity difference between the two cannot be computed by simply finding the magnitude difference between the two quantities.

The velocity and position comparisons for a Head-On trajectory is shown in Figure 6-5. The particular Head-On trajectory shown in these graphs requires that the chaser's 4^{th} stage engine ignites at roughly 1100 seconds, when the chaser's speed is approximately 2.19 km/sec. Meanwhile, the target spacecraft is traveling in roughly the opposite direction at 7.61 km/sec. Because of the approach path, the relative speed between the two craft is estimated by adding their two respective speeds. Therefore, the relative velocity difference is roughly 9.8 km/sec.

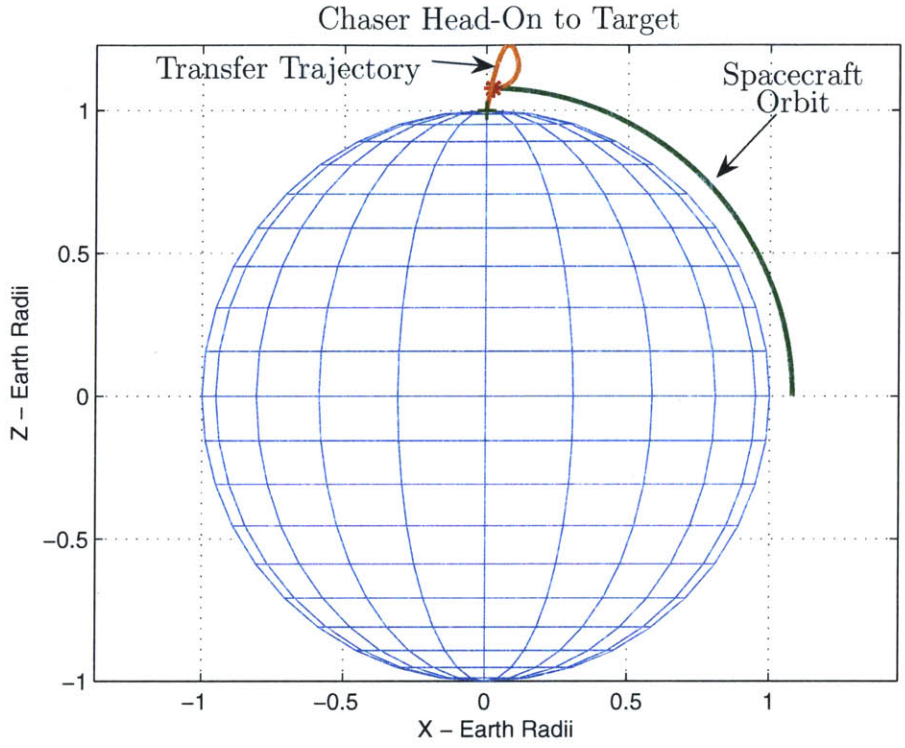

Figure 6-4: Head-On Trajectory

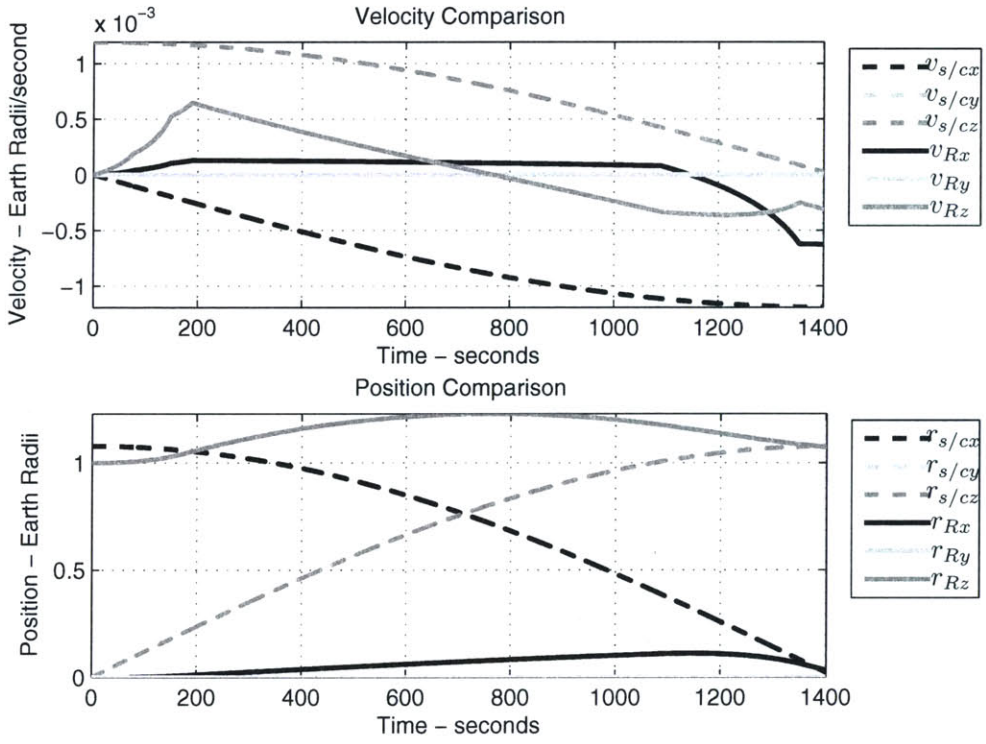

Figure 6-5: Head-On Trajectory - Position and Velocity Comparison

It can be seen that for this particular set of initial conditions, the 4^{th} stage engine simply doesn't have enough fuel and thrust capability to match the position and velocity of the target spacecraft. Recall that the 4^{th} stage engine was designed to provide a maximum velocity change of 5000 m/sec and that the orbital speed of a spacecraft in a 100 or 500 km orbit is 7844 or 7612 m/sec, respectively; therefore, for any head-on trajectory, the chaser's 4^{th} stage engine would have to provide well over 5000 m/sec of velocity change to rendezvous with the orbiting target spacecraft. So, for the 4^{th} stage engine and operational regions defined previously, none of the head-on approach trajectories will result in a successful rendezvous. To verify this, the head-on trajectories will still be run through the simulation.

Consequently, the evaluation will be performed for both the Overtaking approach and the Head-On approach over the entire operational region. By looking at trajectories following both approach paths, the evaluation will better quantify the ability of this specific 4^{th} stage engine while using the ALGA.

6.1.4 Distances

Previously, Section 2.1.1 discussed the possibility of two solution trajectories between the launch point and rendezvous point. Because the direction of travel was chosen to only traverse the "short" route, the transfer angle, θ, does not exceed 180°. Therefore, the linear distance (d) measured from the launch point to the rendezvous point, for any given set of initial conditions, should not exceed a value of twice the radius of the Earth plus the altitude of the target spacecraft ($2r_{\oplus} + h_{s/c}$). However, when the value of θ gets in the vicinity of 180°, the LGA may not be able to distinguish which direction is the "short" way. Therefore, the actual value of d may be slightly greater than $2r_{\oplus}$.

$$d = \|\vec{r}_T - \vec{r}_R\| < (2r_{\oplus} + 1\%) \tag{6.2}$$

6.1.5 Summary of Boundaries and Approach Paths

The boundaries of the Operational region consist of mission times ranging from 700 to 3600 seconds, with linear distances ranging from 0 to roughly $2r_{\oplus}$. The target orbits will be circular ranging in altitude from 100 to 500 km while the difference in inclination between the transfer trajectory and target orbit varies anywhere between 0° and 90°. The chaser spacecraft is also able to take either the overtaking or head-on approach path to the target.

6.2 Evaluation Results

Now that the boundaries of the operational region have been defined, and the differences between the two types of approach paths have been explored, the evaluation of the ALGA over this entire operational range can now be carried out. The evaluation will first focus on the cases where the chaser is launched in the same plane as the target spacecraft to get a baseline performance envelope. Then, two successful test cases will be singled out and examined more thoroughly: one with a plane change of 0° and the other with a plane change of 15°

6.2.1 Baseline Performance Envelope

Considering the boundaries of the operational region as summarized in Section 6.1.5, the evaluation begins with creating test cases that cover the entire mission time span and possible linear distances, but are limited to the lower altitude threshold of 100 km for the circular orbits and no plane change necessary. Then, similar test cases were created for the upper altitude bound of 500 km.

For each test case the linear distance from the initial launch point, $\vec{r}_R(0)$, to the target point, $\vec{r}_T$, was plotted versus the time of flight, $\mathbb{T}$. Then, a marker was assigned to classify the results. There are three categories to describe the possible finding for each test case:

1. The values for R_{err} and V_{err} meet the success criteria ($*$).

2. The chaser runs out of fuel and success criteria are not met. ($+$).

3. The chaser impacts the surface of the Earth ($\circ$).

Figures 6-6 and 6-7 show the results for the 100 km head-on and overtaking approach paths, respectively. Figures 6-8 and 6-9 show the results for the upper altitude threshold.

Head-On Approach Trajectories

By comparing the results plotted in Figures 6-7 and 6-9 to those in Figures 6-6 and 6-8, it is apparent that the 4^{th} stage engine is simply not capable of rendezvousing with the target when taking a head-on approach as expected. The two plots for the head-on approach path clearly show two zones, one marked by ($\circ$) and the other by ($+$). For the semi-circular zone marked by ($+$), the 4^{th} stage engine, as outlined in Section 3.3.1, cannot produce a sufficient amount of thrust to provide the necessary velocity change to rendezvous with the target. Yet, the data suggests that these types of head-on trajectories are possible if given a different 4^{th} stage engine capable of producing either a higher level of thrust or a longer burn time.

The other zone, marked by ($\circ$), indicates the test cases where the chaser impacted the Earth. For the head-on approach path, the chaser may impact the ground for one of two reasons. First, the combination of a long distance and short time of flight produces a transfer trajectory whose semimajor axis is less than the radius of the Earth resulting in an impact as the spacecraft follows this trajectory. The next explanation for why the chaser crashes into the Earth results from the 4^{th} stage engine, again, not having enough fuel. As stated before, the head-on approach requires the chaser to essentially make a U-turn. During this reversal manoeuver, the 4^{th} stage engine runs out of fuel changing the chaser's trajectory enough to run into the Earth. Regardless of the reason, all of the

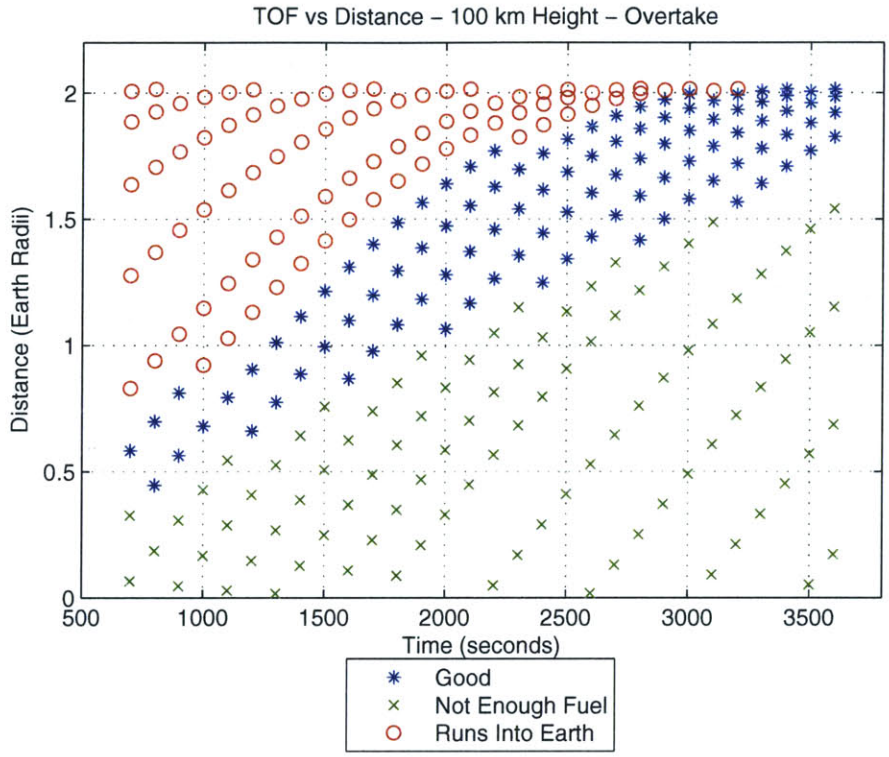

Figure 6-6: $\mathbb{T}$ vs. Distance - 100km - Overtaking Target

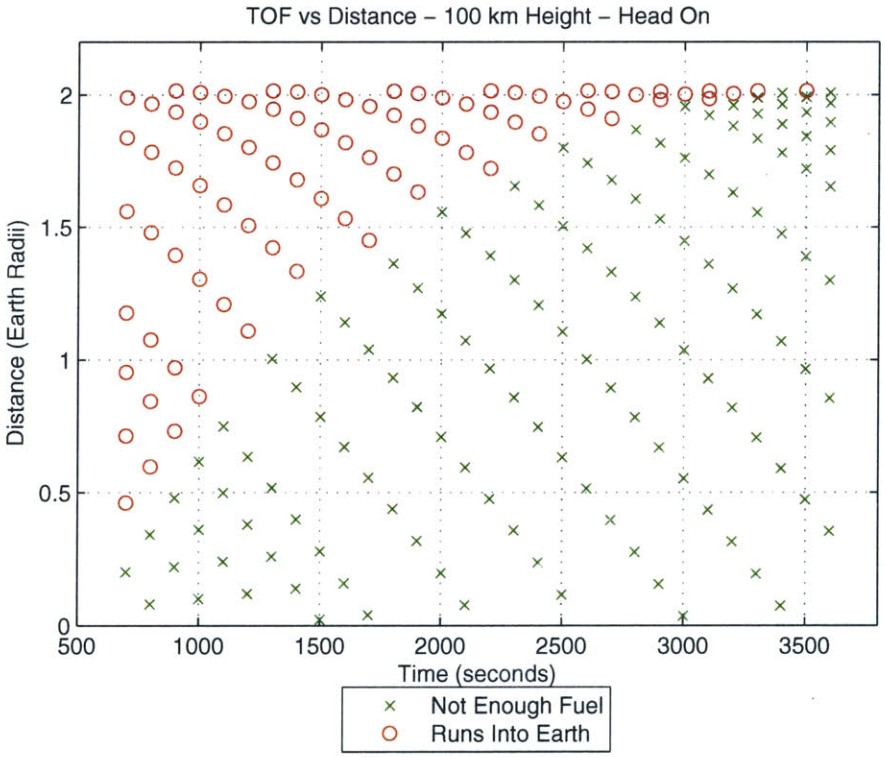

Figure 6-7: $\mathbb{T}$ vs. Distance - 100km - Heading to Target

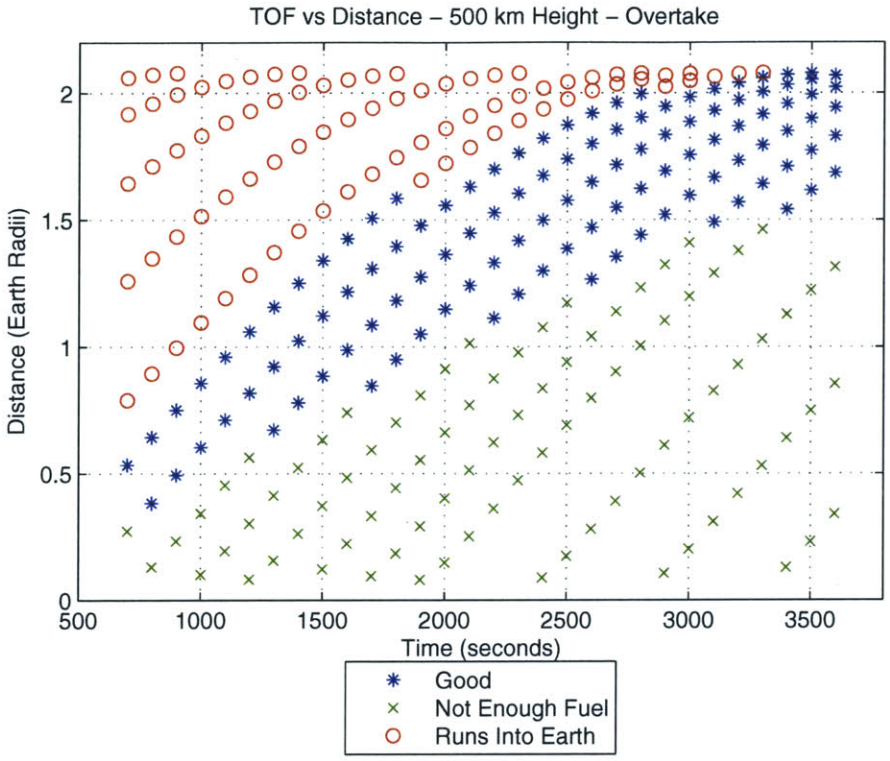

Figure 6-8: $\mathbb{T}$ vs. Distance - 500km - Overtaking Target

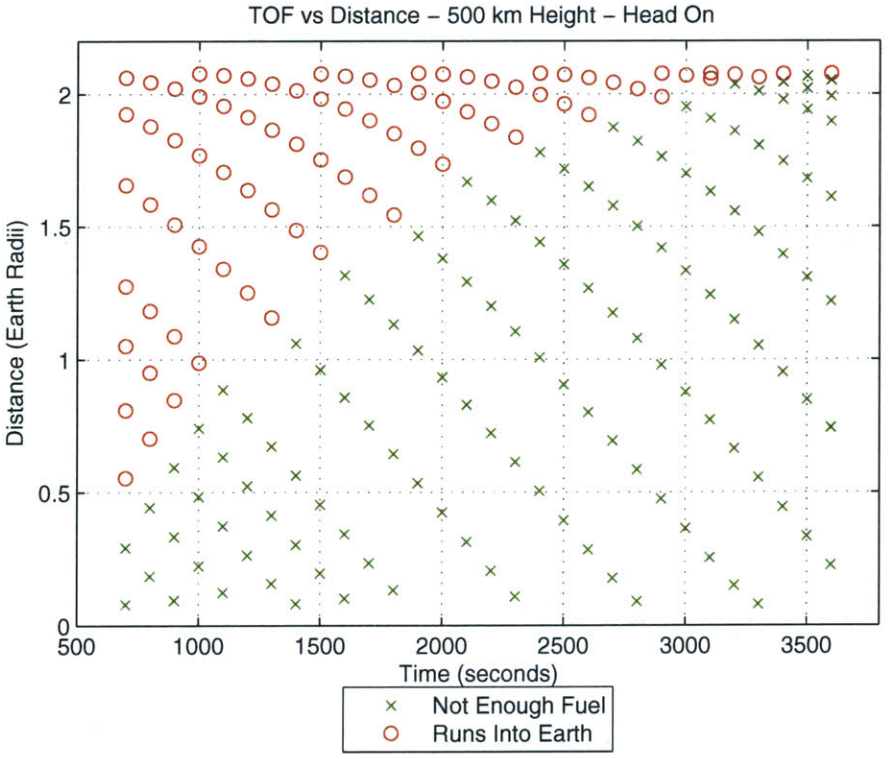

Figure 6-9: $\mathbb{T}$ vs. Distance - 500km - Heading to Target

head-on trajectories that impacted the Earth, produced a value of $\mathbb{T}_b$ greater than the maximum burn time of the 4$^{\text{th}}$ stage engine, $\mathbb{T}_{b_{max}}$, just like the trajectories that didn't crash. This confirms the expectation stated in Section 6.1.3.

Overtaking Approach Trajectories

Focusing on Figures 6-6 and 6-8, the first and most important observation about these plots is that the ALGA is indeed able to compute values for $\vec{r}_{Tmod}$, $\vec{B}_{4^{th}}$, and $\mathbb{T}_b$ that, when executed, result in the chaser meeting the success criteria. Accordingly, there are three visible zones. Like the head-on approach paths, the overtaking approach for both the 100 km and 500 km circular target orbits have the two zones marked by ($\circ$) and ($+$) with an additional ($*$) zone. This third zone conveys that the 4$^{\text{th}}$ stage engine has more than enough fuel to burn for the entire burn time, $\mathbb{T}_b$. Fortuitously, this zone also indicates that the values for R_{err} and V_{err} also meet the success criteria.

Zone Predictability

The occurrence of these three distinct zones raises the question: given some initial conditions, is it possible to predict the outcome without running the entire simulation?

For the overtaking trajectory, the difference between rendezvousing with a craft depends, almost solely, on how much $\Delta\vec{v}$ the 4$^{\text{th}}$ stage engine can provide to the chaser by means of burning fuel. As stated before, the 4$^{\text{th}}$ stage engine used in this thesis has the ability to provide a limited amount of $\Delta\vec{v}$ totalling 5000 m/s. Therefore, one major part of answering the predictability question is dependant on whether the amount of $\Delta\vec{v}$ needed to rendezvous with the target can be predicted given a set of initial conditions.

Section 4.2 explains how the ALGA calculates a first guess for the needed $\Delta\vec{v}$ by using the methods developed by Burns and Scherock. These equations are based on the given initial conditions, are simple to calculate, and give an approximation of how much $\Delta\vec{v}$ is necessary for the 4$^{\text{th}}$ stage engine to provide. Looking at several test cases, the difference

between this initial approximation and the final $\Delta\vec{v}$ for the test cases that landed in the (+) and (∗) zones was approximately 100 to 400 m/s. For instance, one of the test cases that landed in the (∗) zone had an initial prediction from the Initialization block of 3264.8 m/sec of $\Delta\vec{v}$, which is much less than the maximum value of 5000 m/sec. By the end of the boost phase, the calculated value of $\Delta\vec{v}$ for this same trajectory was 3428.8 m/sec, which is a difference of 164 m/sec. For the test cases that land in the (+) zone, the Initialization block similarly predicts that the required $\Delta\vec{v}$ is substantially greater than the limiting value of 5000 m/sec.

The conclusion is that the Initialization block can be a useful tool in predicting whether or not the 4^{th} stage engine is capable of providing the required amount of $\Delta\vec{v}$ for a given set of initial conditions. However, there is a region of ambiguity when using this prediction method, which is due to the 300 m/sec variation of $\Delta\vec{v}$ over the course of the boost phase. Therefore, this prediction method cannot be relied on when the predicted value of $\Delta\vec{v}$ is within 400 m/sec of the 5000 m/sec limit, which is the region near the boundary between the (+) and (∗) zones.

On a side note, the boundary between the "Good" zone and the "Not Enough Fuel" zone appears to be nearly a straight line. However, using the 'add trendline' feature in Excel, it was found that the data fit a parabolic polynomial as stated in Equations 6.3 and 6.4 for the 100 km and 500km orbits, respectively.

$$d_{100} = -4 \times 10^{-08}\mathbb{T}^2 + 0.0006\mathbb{T} - 0.037 \tag{6.3}$$

$$d_{500} = -3 \times 10^{-08}\mathbb{T}^2 + 0.0006\mathbb{T} - 0.0785 \tag{6.4}$$

This Initialization block prediction technique only applies to those trajectories that fall into either the (+) or (∗) zones and does not account for the transfer trajectories that impact the Earth (the (○) zone). A similar solution for predicting this zone is likely possible, but has not yet been established.

6.2.2 Sample Trajectories

After the creation of the baseline performance envelope, two specific test cases were selected to take a comprehensive look at the trajectories produced by using the ALGA. This section will focus on the values of $\vec{r}_{Tmod}$, $\mathbb{T}_b$, and $\vec{B}_{4^{th}}$ calculated by the ALGA over the course of the boost phase. The differences in these values when a plane change is involved will also be explored.

The two test cases were chosen based on their initial conditions. It was desired to have two similar trajectories, which could be compared and contrasted more easily. Table 6.1 lists the initial conditions of the two sample trajectories for both the chaser and target spacecraft. Both of the selected test cases have the same time of flight, are targeting a spacecraft with the same position and velocity, and the linear distance between the launch and rendezvous points are almost identical. The only difference between these two test cases, is that one causes the chaser trajectory to make no plane change while in the other, the chaser must make a plane change of 15°.

Table 6.1: Plane Change Comparison

	0° **Plane Change**	15° **Plane Change**
Target		
Position (m)	$[-6299284, 0, 151171]$	$[-6299284, 0, 151171]$
Velocity (m/s)	$[-1830.47, 0, -7627.55]$	$[-1830.47, 0, -7627.55]$
Chaser		
Position (m)	$[3189068, 0, 5523629]$	$[2761814, 1594534, 5523628]$
$\mathbb{T}$ (s)	2400	2400
d (m)	10301666	10037008

Trajectories and $\vec{r}_{Tmod}$

Figure 6-10 shows the profile of the test case that does not have to make a plane change. The vector $\vec{r}_{Tmod}$ describes the modified target position, which ends up being a point beneath the surface of the Earth. Per the discussion in Chapter 4, $\vec{r}_{Tmod}$ is expected to have a value that would put it beneath the surface. It should be noted that this figure

displays a bisected Earth to better view the aim point. This is the point that the ALGA uses to calculate the correct transfer trajectory for the chaser to follow. Due to the chaser not having to make a plane change, it is expected that the aim point lies in the same plane as the both the transfer trajectory and target spacecraft trajectory. Through inspection of the data, the aim point was found to be in the same orbital plane as the chaser and target spacecraft.

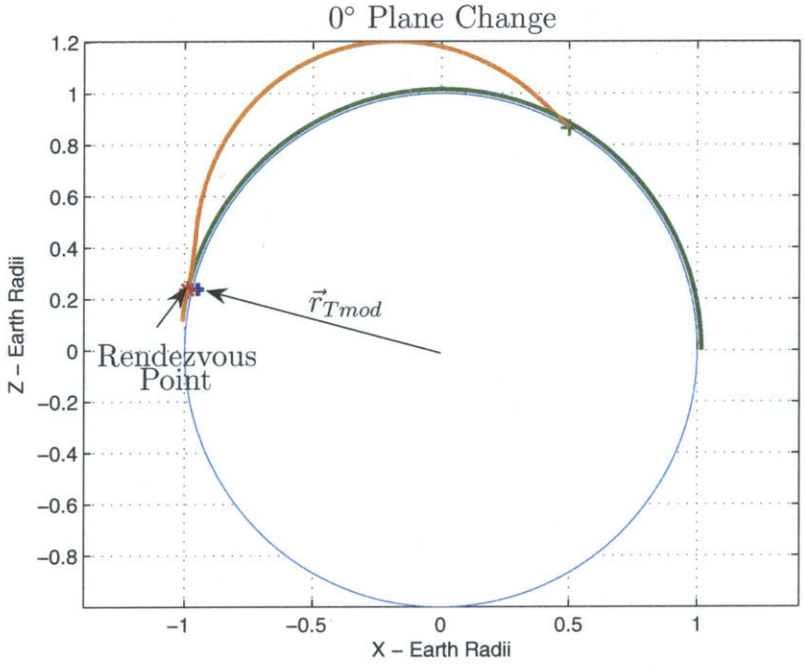

Figure 6-10: Profile of 0° Plane Change Trajectory

The 15° plane change trajectory is shown in Figure 6-11. The important thing to recognize in this picture is that the transfer trajectory does indeed have a different orbital inclination compared to the target spacecraft.

Figures 6-12 and 6-13 present two different views of the 15° plane change trajectory. First is a zoomed-in profile view of the 15° plane change trajectory, which is similar to that shown in Figure 6-10. The dotted line in the figure represents the path that the chaser would follow if the 4^{th} stage engine did not ignite (and if the Earth was not there). The aim point coincides with where the chaser would be at time, $\mathbb{T}$. This confirms the

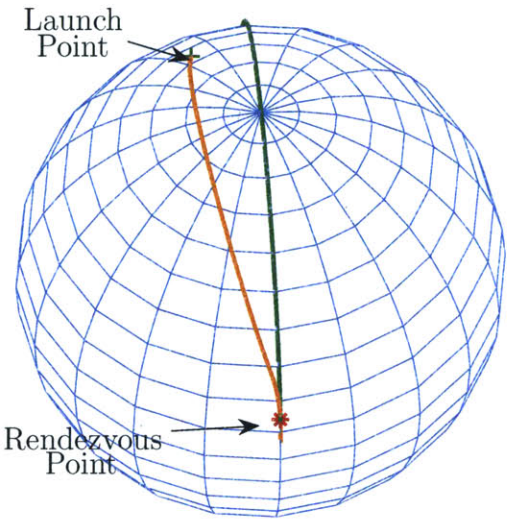

Figure 6-11: View of 15° Plane Change Trajectory

derivation of the ALGA as dictated in Chapter 4.

The next interesting aspect of $\vec{r}_{Tmod}$ comes from looking at an edge-on view of the trajectory. Figure 6-13 is a view that is perpendicular to the target spacecraft's trajectory. The path of the chaser obviously changes as the chaser gets closer to the rendezvous point. This change is due to the thrust produced by the 4^{th} stage engine. However, the aim point does not follow this motion. The aim point, again, is at the termination of the chaser's projected path had the 4^{th} stage engines not ignited.

Table 6.2 shows the position of both the rendezvous point and aim point for the two sample trajectories. The position difference between the aim points and the rendezvous point is represented by $\Delta\vec{r}_T$ and is shown in the table. From the values of $\Delta\vec{r}_T$, the position of the aim point is found to change more for the 15° case than for the 0° case.

Table 6.2: Comparison of $\vec{r}_{Tmod}$

	0°			15°		
Rendezvous	[-6299285	0	151171]	[-6299285	0	151171]
Aim	[-6050556	0	1523913]	[-6001514	-151650	1547332]
$\Delta\vec{r}_T$	[248729	0	12202]	[297770	-151650	35620]
Δr_T	249029			336056		

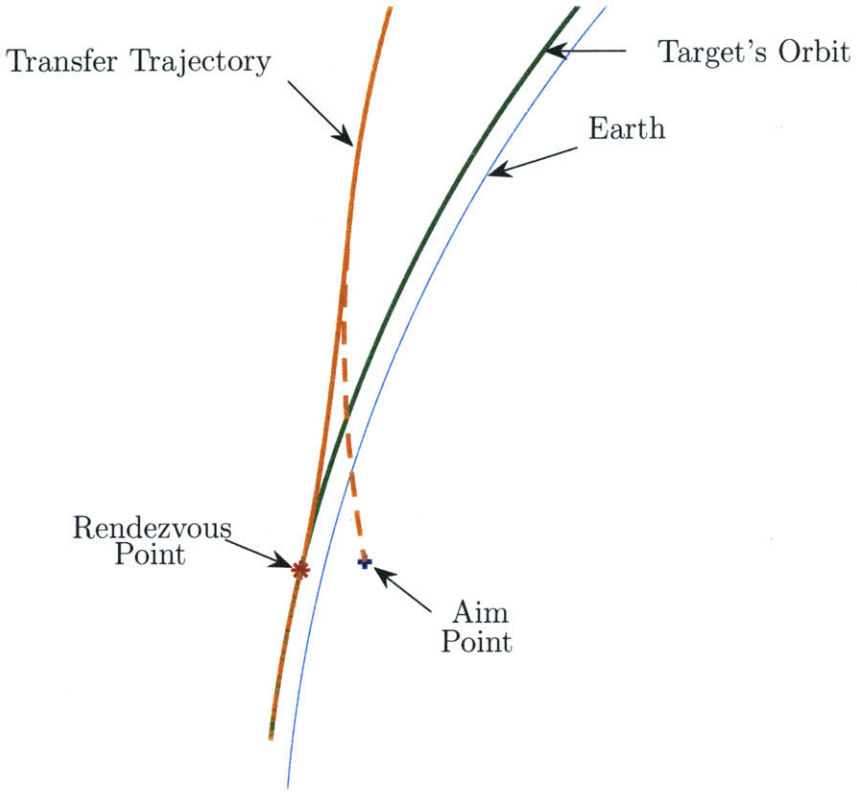

Figure 6-12: Profile View of 15° Plane Change Trajectory with $\vec{r}_{Tmod}$

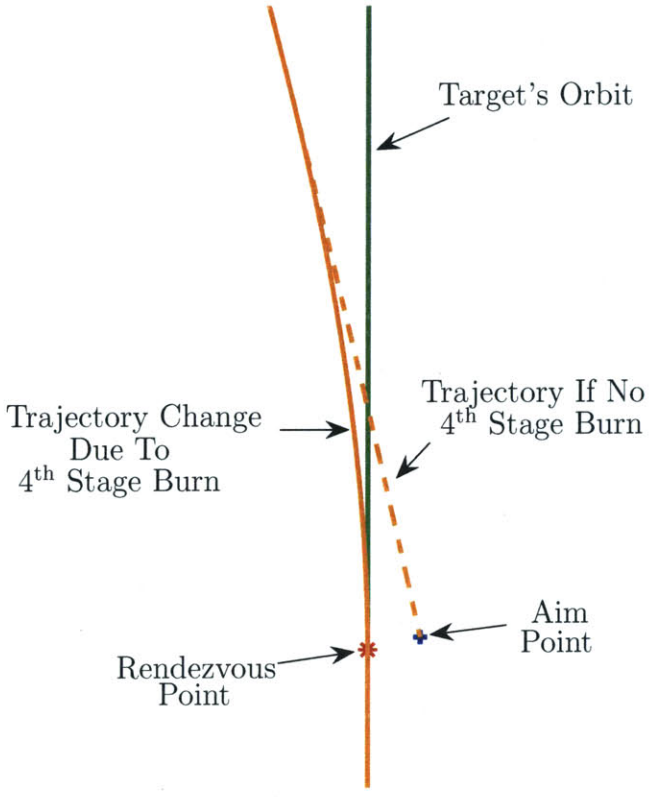

Figure 6-13: Edge-On View of 15° Plane Change Trajectory with $\vec{r}_{Tmod}$

$\mathbb{T}_b$ & $\vec{B}_{4^{th}}$

Figures 6-14 to 6-15 both show a similar trend in the calculation of $\mathbb{T}_b$, and $\vec{B}_{4^{th}}$: the values start with the Initialization block value and exponentially reach their final value over the course of the boost phase. The final values of $\mathbb{T}_b$, and $\vec{B}_{4^{th}}$ are then used to execute the 4$^{\text{th}}$ stage manoeuver. Per the discussion of zone predictability in Section 6.2.1, the value of Δv is expected to change over the course of the boost phase. This is confirmed by Figure 6-14 showing approximately 315 m/sec and 265 m/sec decrease in the predicted Δv versus the ending Δv for the 0° plane change and 15° plane change, respectively. These figures also show that the amount of Δv is substantially greater for the 15° plane change trajectory than the 0° plane change trajectory, which was expected.

With the decreasing value of Δv as time progresses, the value of the burn time decreases accordingly, due to the dependance of the burn time on the required amount of Δv. Hence, the burn time is greater for the 15° plane change trajectory than the 0° plane change trajectory.

Accuracy

With the values of $\vec{r}_{Tmod}$, $\mathbb{T}_b$, and $\vec{B}_{4^{th}}$ calculated for these two trajectories, the next logical step is to look at the accuracy as measured by the quantities R_{err} and V_{err} described in Section 3.4. The values of R_{err} and V_{err} illustrate how well the ALGA was able to guide the chaser during boost, calculate the proper ignition time, and determine thrust direction of the 4$^{\text{th}}$ stage manoeuver, which drives the chaser to match the position and velocity of the target spacecraft.

Before looking at the exact values of R_{err} and V_{err}, the position and velocity of both the 0° plane change test case and 15° plane change test case are compared to the position and velocity of the target spacecraft. Figures 6-16 and 6-17 show the comparisons for each of the test cases. These figures confirm that the position of the chaser in both test cases reaches the vicinity of the target spacecraft with comparable velocities.

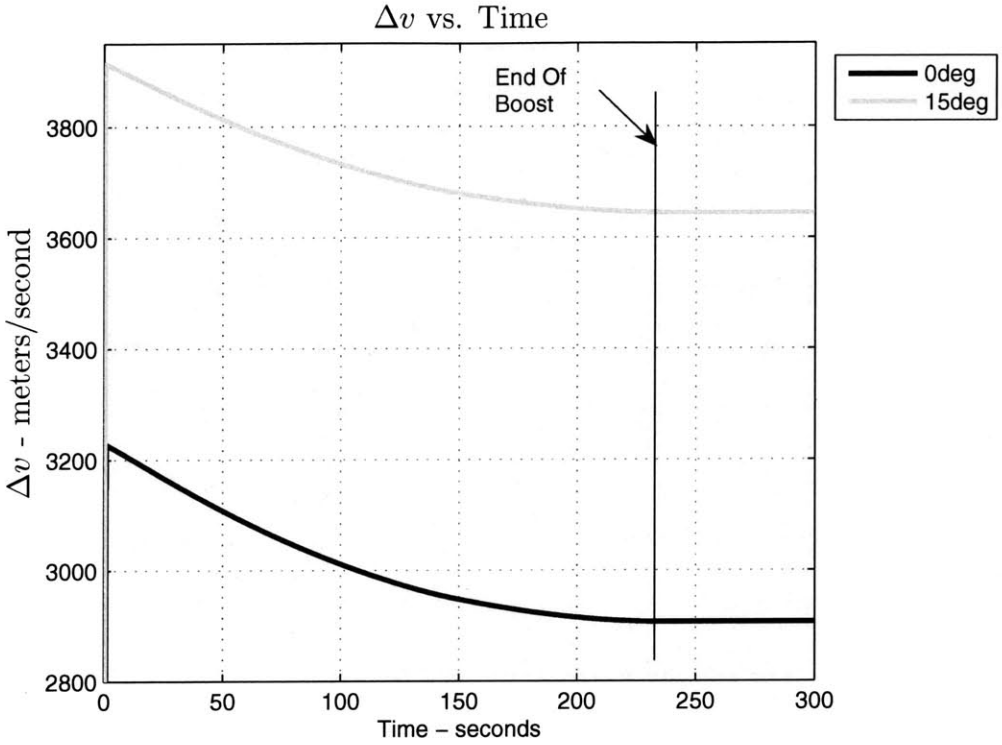

Figure 6-14: Δv vs. Time for 0° and 15° Plane Change

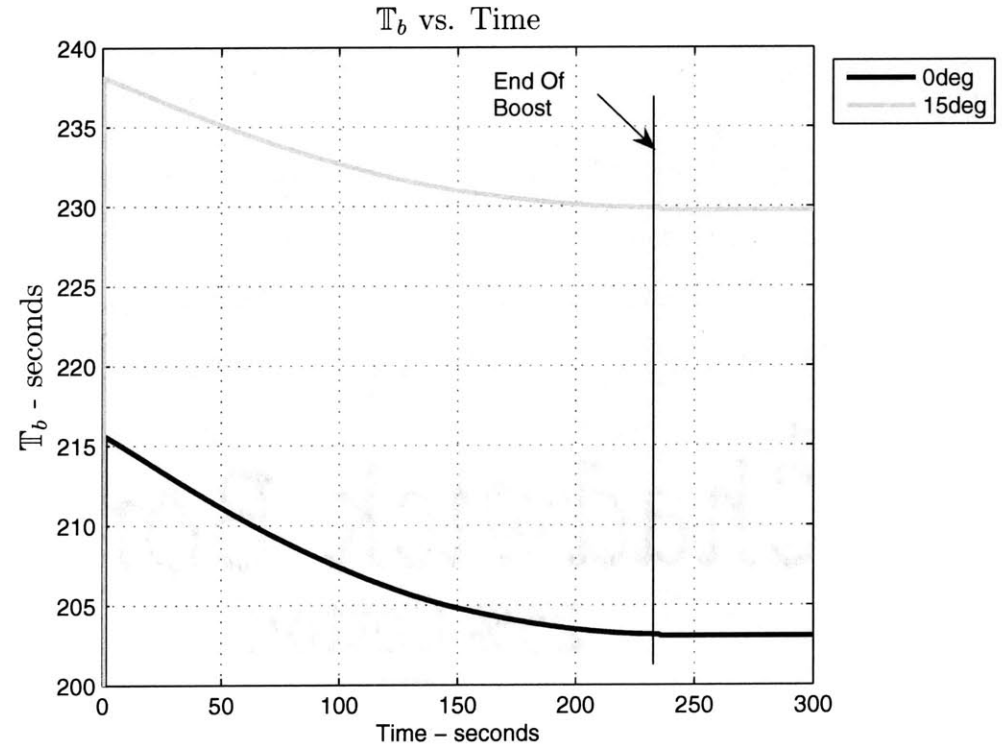

Figure 6-15: $\mathbb{T}_b$ vs. Time for 0° and 15° Plane Change

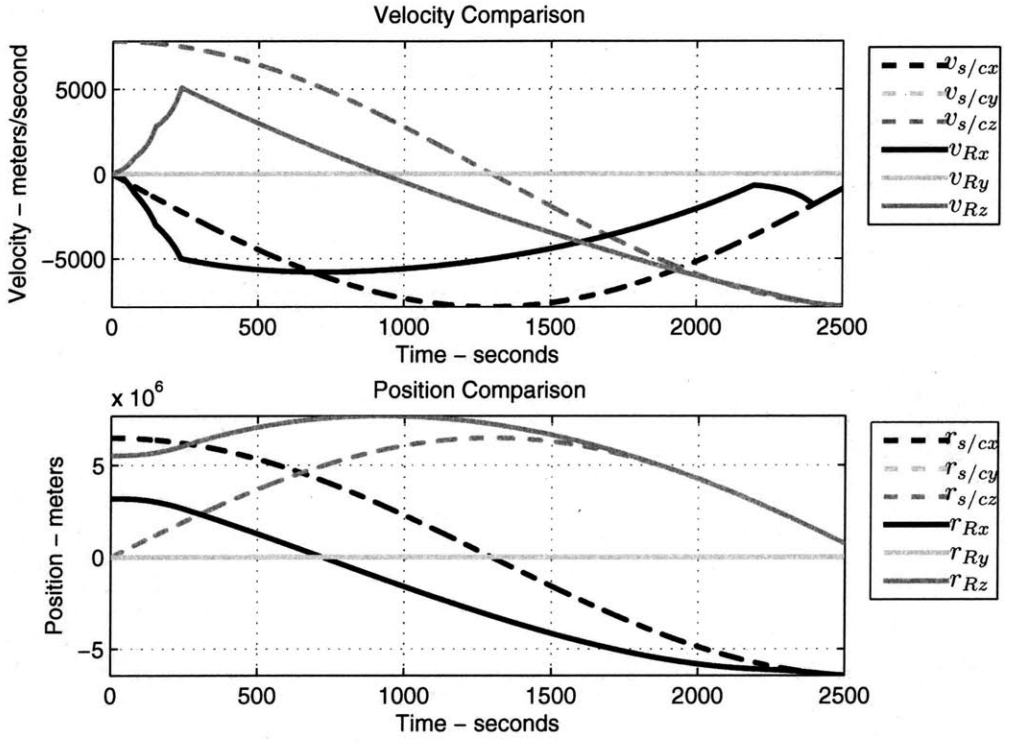

Figure 6-16: $\vec{r}$ and $\vec{v}$ Comparison for 0° Plane Change

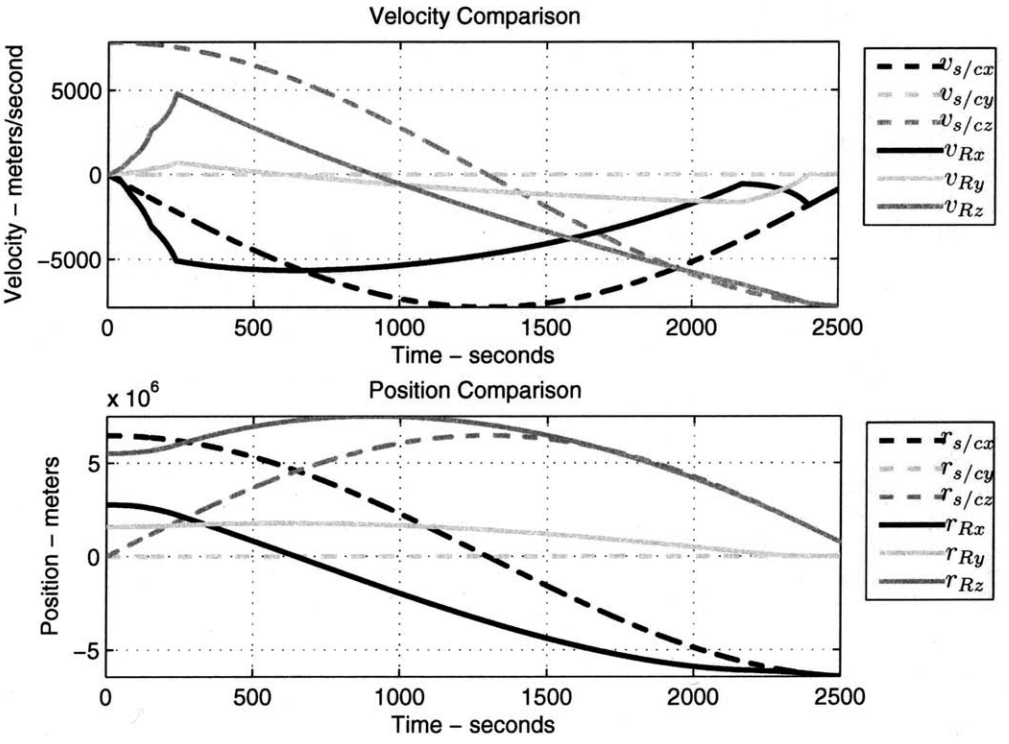

Figure 6-17: $\vec{r}$ and $\vec{v}$ Comparison for 0° Plane Change

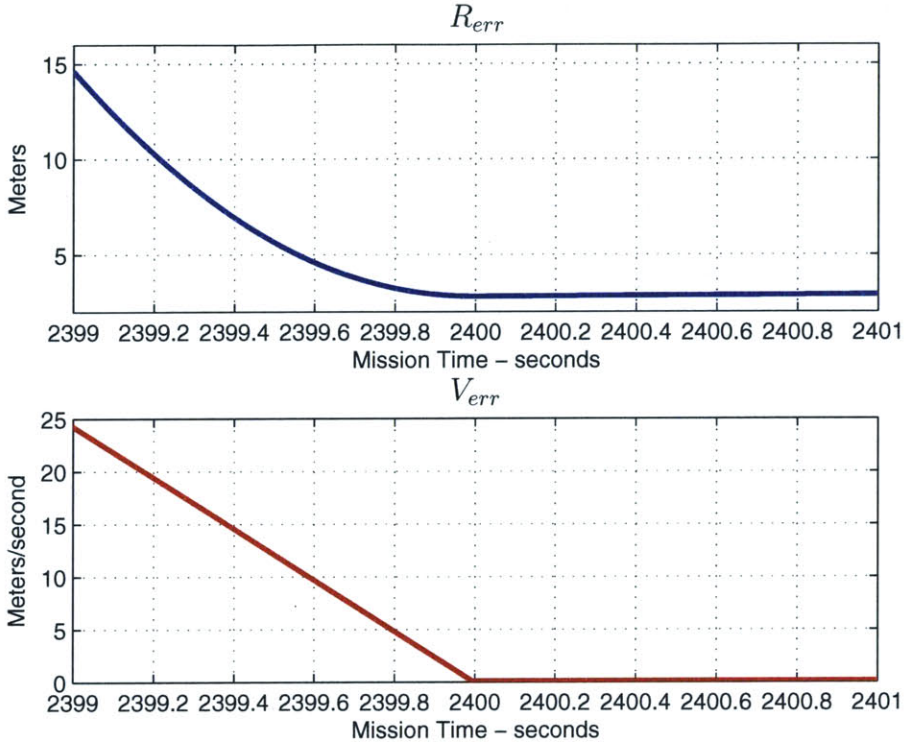

Figure 6-18: R_{err} and V_{err} for 0° Plane Change

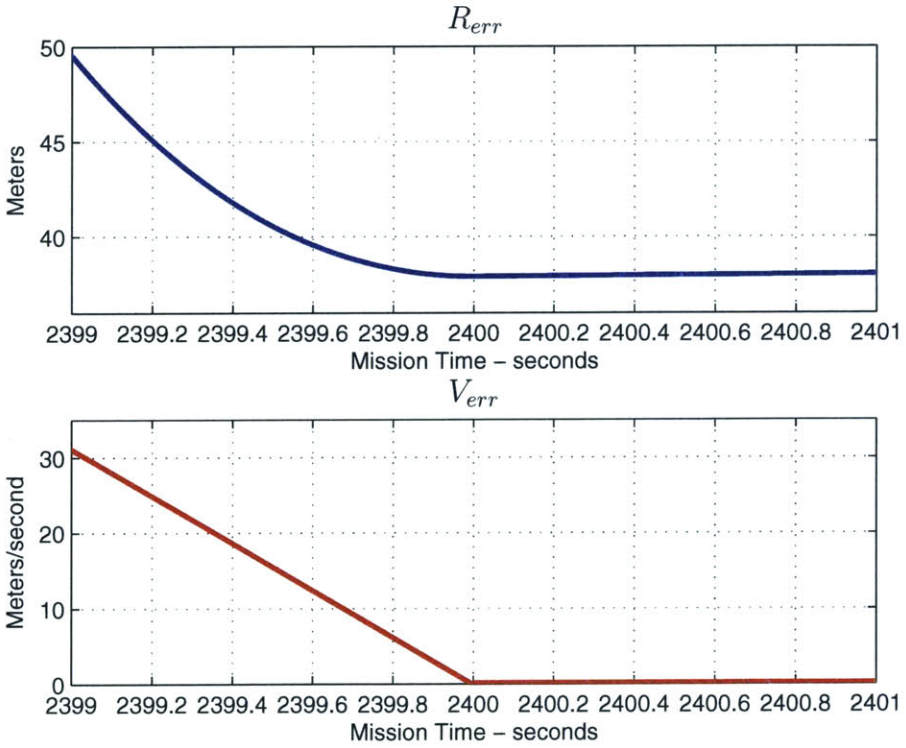

Figure 6-19: R_{err} and V_{err} for 15° Plane Change

Focusing on the range of time of one second before and one second after the rendezvous time, $\mathbb{T}$, Figures 6-18 and 6-19 prove that these two test cases belong to the $(*)$ zone, meaning the values of R_{err} and V_{err} meet the success criteria outlined in Section 3.4. By examining these four graphs, it is apparent that R_{err} and V_{err} are much less for the 0° plane change trajectory than the 15° plane change trajectory. The 0° plane change case has an approximate position error of 1 m and velocity error of less than .1 m/sec while the 15° plane change case has an approximate position and velocity error of 38 m and less than .5 m/s, respectively.

A summary of the findings for the two selected test cases is presented in Table 6.3.

Table 6.3: Plane Change Comparison

	0° **Plane Change**	15° **Plane Change**
Target		
Position (m)	$[-6299284, 0, 151171]$	$[-6299284, 0, 151171]$
Velocity (m/s)	$[-1830.47, 0, -7627.55]$	$[-1830.47, 0, -7627.55]$
IC's		
Position (m)	$[3189068, 0, 5523629]$	$[2761814, 1594534, 5523628]$
$\mathbb{T}$ (s)	2400	2400
d (m)	10301666	10037008
Result		
$\mathbb{T}_b$ (s)	203.065	229.753
$\vec{B}_{4^{th}}$	$[-0.9986, 0, -0.0535]$	$[-0.8830, 0.4559, -0.1113]$
$\vec{r}_{Tmod}$	$[-6061354, 0, 3201670]$	$[-6087747, 236237, 3164915]$
R_{err} (m)	2.801	37.919
V_{err} (m/s)	0.0773	0.132

[This page intentionally left blank.]

Chapter 7

Conclusions

The objective of the research conducted in this thesis was to generate an initial version of an Augmented Lambert Guidance Algorithm (ALGA) that could be used in conjunction with a preexisting Lambert Guidance Algorithm to guide a launch vehicle from the surface of the Earth and rendezvous with a spacecraft in orbit. After the completion of this first version, a simulation was created to evaluate the viability of the ALGA. The simulation consisted of a 3 degree-of-freedom model implementing the two-body approximation for the motion of the chaser and the target spacecraft. The simulation included models for the thrust and mass characteristics of the chaser along with a simple mechanism for steering so that the focus was mainly on the performance of the ALGA and not on other aspects of the direct-ascent rendezvous missions.

7.1 Summary of Results

The simulation generated to mimic the physical characteristics of the chaser and target spacecraft was used to evaluate how well the ALGA was able guide the chaser to rendezvous with the target. The evaluation consisted of creating a multitude of test cases that conformed to the defined operational region for the immediate-response direct-assent missions.

The data gathered from running these many test cases produced several plots that described the ability of the ALGA to meet the defined success criteria. These plots confirmed that the ALGA was, indeed, able to guide the chaser to rendezvous with the target spacecraft. There were two main factors that contributed to the success or failure of a particular test case: the relative direction of travel between the chaser and the target spacecraft, and the amount of fuel carried by the chaser.

During the evaluation process, two types of approach paths were defined: overtaking and head-on. For the overtaking approach the chaser and target spacecraft travel in the same relative direction. Conversely, the head-on approach involves the chaser and target spacecraft moving in opposite directions, headed towards one another. All of the head-on trajectories required much more fuel than the maximum amount of fuel carried by the chaser; consequently, there were no successful rendezvous test cases for the head-on approach paths.

The data gathered for the overtaking approach paths showed three regions. One where the chaser impacted the Earth, a region of successful test cases, and another unsuccessful region. The difference between the successful region and the unsuccessful region was found to be simply a difference in the amount of fuel required to rendezvous. As previously stated, the chaser carries a limited amount of fuel. The successful test cases required an amount of fuel that was less than this limit while the unsuccessful cases (not including the ones that impacted the Earth) required more than the limit. A prediction method was devised that can be used to predict the success of a particular test case based on how much fuel is required compared to the available amount.

Overall, the simulations conducted for the various test cases proved that the methods developed in this thesis created a feasible Augmented Lambert Guidance Algorithm for use in the direct-ascent, quick-response missions.

7.2 Future Work

Considering the ALGA created in this thesis is a first edition, much more work has to be completed before the algorithm will ever be implemented in an actual launch vehicle. To this end, several avenues of future work have been identified.

For the first-order evaluation of the ALGA completed in this thesis, a three degree-of-freedom simulation is sufficiently adequate. However, much more detailed models will be needed to test the ALGA further. A higher-fidelity six degree-of-freedom model would be a logical next step in the evaluation process. This high-fidelity simulation would incorporate more components of the launch vehicle, including attitude control and possibly the guidance logic to control the final terminal manoeuvres to dock with the target spacecraft.

In its current state, the ALGA uses much more processing power than would potentially be available on an actual launch vehicle. Therefore, more research needs to be conducted to make the ALGA more efficient. Another area to investigate is that of the region of successful test cases. As it stands now, the successful region is limited by the amount of fuel available to the payload of the chaser. It would be advantageous for the payload to be able to tap into the excess fuel in the boost stages of the launch vehicle, which could possibly increase the range of successful test cases.

[This page intentionally left blank.]

Appendix A

Other Conic Sections

A.1 Parabola

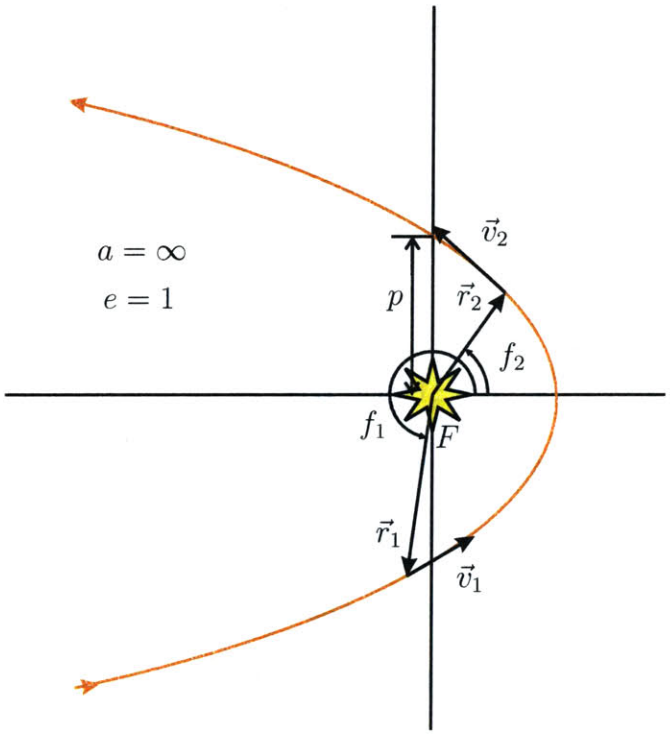

Figure A-1: Parabolic Transfer

$$\beta_{1,2} = \sqrt{p}\tan\frac{1}{2}f_{1,2} \tag{A.1}$$

$$\mathbb{T}\sqrt{\mu} = \frac{\beta_2-\beta_1}{6}\left[3(p+\beta_1\beta_2)+(\beta_2-\beta_1)^2\right] \tag{A.2}$$

$$r_1+r_2 = \frac{1}{2}(2p+\beta_1^2+\beta_2^2) \tag{A.3}$$

$$\sqrt{r_1r_2}\cos\frac{1}{2}\theta = p+\beta_1\beta_2 \tag{A.4}$$

$$F = 1-\frac{(\beta_1-\beta_2)^2}{2r_1} \tag{A.5}$$

$$G = \frac{(\beta_1-\beta_2)}{2\sqrt{\mu}}(2r_1+\beta_1(\beta_1-\beta_2)) \tag{A.6}$$

$$\dot{F} = -\frac{\sqrt{\mu}(\beta_1-\beta_2)}{r_1r_2} \tag{A.7}$$

$$\dot{G} = 1-\frac{(\beta_1-\beta_2)^2}{2r_1} \tag{A.8}$$

A.2 Hyperbola

$$\psi = \frac{1}{2}(H_2-H_1) \tag{A.9}$$

$$\cosh\phi = e\cosh\frac{1}{2}(H_1+H_2) \tag{A.10}$$

$$\mathbb{T}\sqrt{\mu} = 2(-a)^{\frac{3}{2}}(\psi-\sinh\psi\cosh\phi) \tag{A.11}$$

$$r_1+r_2 = 2a(1-\cosh\psi\cosh\phi) \tag{A.12}$$

$$\sqrt{r_1r_2}\cos\frac{1}{2}\theta = a(\cosh\psi-\cosh\phi) \tag{A.13}$$

$$p = -\frac{r_1r_2\sin^2\frac{1}{2}\theta}{a\sinh^2\psi} \tag{A.14}$$

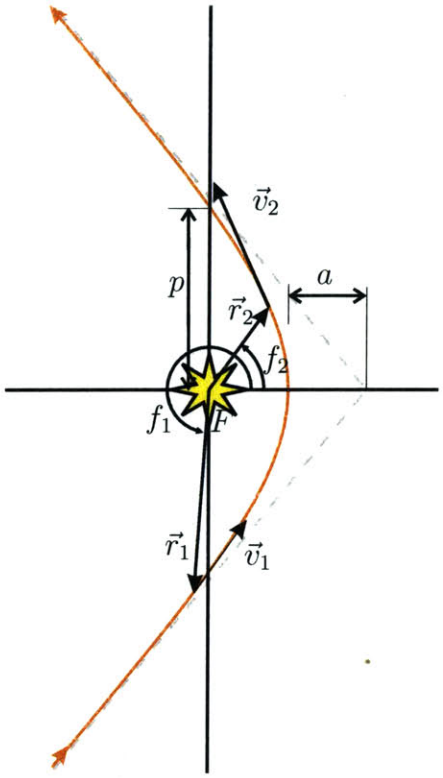

Figure A-2: Hyperbolic Transfer

$$e = \sqrt{1 - \frac{p}{a}} \tag{A.15}$$

$$F = 1 - \frac{a}{r_1}(1 - \cosh 2\psi) \tag{A.16}$$

$$G = \mathbb{T} - \sqrt{\frac{(-a)^3}{\mu}}(\sinh 2\psi - 2\psi) \tag{A.17}$$

$$\dot{F} = -\frac{\sqrt{-\mu a}}{r_1 r_2} \sinh 2\psi \tag{A.18}$$

$$\dot{G} = 1 - \frac{a}{r_2}(1 - \cosh 2\psi) \tag{A.19}$$

[This page intentionally left blank.]

Appendix B

Boost Sizing Spreadsheet

The values expressed in Table B.1 were taken from Reference [7].

Table B.1: US Space Launch Systems

Rocket	**Stage**	T *Newtons*	$\mathbb{T}_{b_{max}}$ *seconds*	M_{tot} *kg*	M_{struct} *kg*	M_{prop} *kg*	I_{sp} *seconds*
ATHENA 2	1	1604000	83.4	531500	4450	48700	253
	2	1604000	83.4	531500	4510	48700	253
	3	187000	150	10715	1030	9780	293
	4	890	1500	596	360	236	220
MINOTAUR	1	792000	60.8	23077	2248	20785	237
	2	268000	65.54	7032	691	6237	287.5
	3	154000	72.5	4332	416	3915	289
	4	32000	69.6	897	126	771	290.1
START	1	980000	63	26000	3000	23000	263
	2	490000	60	13000	1500	11500	280
	3	245000	63	6000	1000	5000	280
	4	100000	53	1000	300	700	295
Delta II	0	1497600	63.3	13080	1315	11765	274
	1	1085800	261	101800	5680	96100	301.7
	2	43657	431	6954	950	6004	319.2
	3	66400	87.1	2217	208	2009	292.2
	4	45800	66.4	1147	82	1065	291.8
TAURUS 1110	1	1615000	82.5	53100	4400	48.7	277.9
	2	471000	72.4	13242	1088	12154	285
	3	115000	75.1	3379	352	3027	290.2
	4	32000	68.5	875	104	771	286.7

[This page intentionally left blank.]

Bibliography

[1] Bate, R. R., Mueller, D. D., and White, J. E., *Fundamentals of Astrodynamics*, Dover Publications, Inc., New York, 1971.

[2] Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics*, AIAA, Washington, DC, 1999.

[3] Battin, R. H. and Vaughan, R. M., "An Elegant Lambert Algorithm," Vol. 7, No. 6, 1983. pp. 662-670.

[4] Battin, R. H., Fill, T. J., and Shepperd, S. W., "A New Transformation Invariant in the Orbital Boundary-Value Problem," *Journal of Guidance and Control*, Vol. 1, Jan.-Feb. 1978, pp. 50-55. Vol. 1, Feb. 1978, pp. 50–55.

[5] Battin, R. H., "Lambert's Problem Revisited," *AIAA Journal*, Vol. 15, May 1977, pp. 707-713.

[6] Burns, S. P. and Scherock, J. J., "Lambert Guidance Routine Designed to Match Position and Velocity of Ballistic Target," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, Nov-Dec 2004, pp. 989–996.

[7] Isakowitz, S. J., Hopkins, J. B., and Hopkins, J. P., *International Reference Guide to Space Launch Systems*, AIAA, Washington, DC, 2004.

[8] Moulton, F. R., *Introduction to Celestial Mechanics*, Macmillan, New York, 1914.

[9] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, New York, 1992.

[10] Sellers, J. J., *Understanding Space: An Introduction to Astronautics*, The McGraw-Hill Companies, Inc., New York, 2004.

[11] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Microcosm Press, California, 2001.

[12] Woffinden, D. C., *On-Orbit Satellite Inspection Navigation and Δv Analysis*, Master of Science Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, August 2004.

[13] Zarchan, P., *Tactical and Strategic Missile Guidance*, Vol. 157, Progress in Aeronautics and Astronautics, AIAA, Washington, DC, 1994.