Abstract

Title of Thesis: A COUPLED INTERPLANETARY – ENTRY, DESCENT AND LANDING TARGETING PROCEDURE

Jeremy David Shidner, M.S., 2006

Thesis Directed by: Dr. Robert Tolson
National Institute of Aerospace

In NASA's past, targeting to a Mars landing site has required iteration between the Entry, Descent and Landing (EDL) analysts at NASA Langley Research Center and the interplanetary navigators at the Jet Propulsion Laboratory (JPL). JPL would develop thousands of arrival states at Mars based on an assumed, constant entry flight path angle and down range angle from entry to landing. Langley would perform the EDL analysis using the arrival states from JPL. Feasible trajectories developed by Langley had varying flight path and down range angles over the entire launch/arrival window forcing an iteration of the trajectory design between JPL and Langley. The iteration process was inefficient, stretching out the design phase while introducing the possibility for error. The purpose of this study is to develop a method that calculates an interplanetary trajectory from Earth to Mars and seamlessly couples the tool with Langley's EDL analysis. The method will show a novel extension to current Lambert solution methods while incorporating a multiple revolution capability.

A COUPLED INTERPLANETARY – ENTRY, DESCENT AND LANDING TARGETING PROCEDURE

by

Jeremy David Shidner

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2006

Advisory Committee:

Professor Robert H. Tolson, Chair
Professor Robert Sanner
Professor David L. Akin

Table of Contents

Page

Table of Figures iv

1 Introduction 1

2 Fundamental Astrodynamics in Interplanetary Travel 4

2.1 Trajectory Types 4

2.2 Spherical Trigonometry 6

2.3 Time 8

2.4 Coordinate Systems 10

2.4.1 Perifocal 10

2.4.2 Earth Mean Equator and Mean Equinox of J2000 (EME) 11

2.4.3 IAU Physical Ephemerides 12

2.4.4 B-plane Coordinate System 14

2.5 Hohmann Transfer 17

3 Lambert's Problem 19

3.1 Selection of Method to Solve Lambert's Problem 20

3.2 Lambert Solution Method 24

4 Interplanetary Trajectories 38

4.1 Launch and Arrival Date Considerations 39

4.2 Ephemerides 41

4.3 Lambert Solution 44

4.4 Calculation of the Interplanetary Trajectory 45

5 Launch and Arrival Space Synthesis 48

5.1 Launch Space 48
5.2 Arrival Space Synthesis 49
5.2.1 B-plane 50
5.2.2 Maximum Latitudes 51
5.2.3 Targeting Latitudes 57
5.2.4 Arrival State 58
6 Results 63
6.1 Planetary Ephemeris Comparison 65
6.2 Position Comparison 69
7 Linking the Entry, Descent, and Landing Analysis to the Interplanetary Calculations 72
7.1 Selecting the Optimal Mission 79

Table of Figures

Page

Figure 2.1: Celestial Sphere 7

Figure 2.2: Perifocal Coordinate Systems 10

Figure 2.3: Earth Mean Equator and Earth Mean Ecliptic Coordinate Systems 12

Figure 2.4: IAU Coordinate System 13

Figure 2.5: B-plane Coordinate System 14

Figure 2.6: Hohmann Transfer 17

Figure 3.1: Multiple Revolution Orbits 21

Figure 3.2: Two Point Time of Flight Representation 22

Figure 3.3: Behavior of Universal Variables Time of Flight 26

Figure 3.4: Dual Solutions of the Same Transfer Time 28

Figure 3.5: Lambert Solution Computational Cost 32

Figure 3.6: Regions of Suitable Newton Iterations 33

Figure 3.7: Lambert Solutions Incorporating Minimum-Energy 35

Figure 4.1: Lead Angle 39

Figure 4.2: Arrival Geometry 45

Figure 4.3: Ballistic Trajectory 46

Figure 4.4: Porkchop Plot 47

Figure 5.1: Launch Asymptote Declination 49

Figure 5.2: B-plane Targeting System 50

Figure 5.3: Inclination for Varying θ and δ 51

Figure 5.4: Latitude Targeting 52

Figure 5.5: Spherical Side f´ 52

Figure 5.6: Oblate Modeling....55

Figure 5.7: Achievable Latitudes for 2009'-10' Earth-Mars Missions....56

Figure 5.8: 1st Rotation 58

Figure 5.9: 2nd Rotation....59

Figure 5.10: 3rd & 4th Rotation....59

Figure 6.1: C_3 and V_∞ Contours....64

Figure 6.2: Van Flandern Error in V_∞....66

Figure 6.3: DE200 Error in V_∞ 67

Figure 6.4: DE405 Error in V_∞ 68

Figure 6.5: Van Flandern Position Error....69

Figure 6.6: DE200 Position Error 70

Figure 6.7: DE405 Position Error 71

Figure 7.1: Parachute and Sky-Crane 72

Figure 7.2: Inertial Entry Velocity at Atmospheric Entry 75

Figure 7.3: Altitude at Parachute Deploy 75

Figure 7.4: Max g's....76

Figure 7.5: Max Heat Rate....77

Figure 7.6: Interplanetary/EDL Combined Porkchop Plot 78

Figure 7.7: Finding the Optimum Mission 81

1 Introduction

With President Bush's new 'Vision for Space Exploration', Mars should no longer be a fantasy of exploration. For aspiring students and engineers that wish to develop bold new ideas for human travel to Mars, new systems need to be researched in the realms of thermal, power, communication, propulsion, and most importantly, entry, descent and landing (EDL) technologies. All these systems are critically linked to the interplanetary trajectory traveled from Earth to Mars by variables such as atmospheric entry velocity, flight path angle, Sun angle, Earth angle, time of arrival, etc. The linking of these interplanetary trajectories to the required systems, particularly EDL, is often the most overlooked aspect of the design.

The interplanetary trajectory is commonly conceptualized by a patched conic method utilizing physical and planetary ephemerides in the early mission design phases. During actual mission operations, a reference, integrated trajectory is used for navigational purposes. However, the patched conic interplanetary trajectory provides an acceptable reference for the preliminary mission studies in EDL systems. The question is how to seamlessly link the EDL system to the patched conic interplanetary trajectory.

From the Mars Science Laboratory (MSL) mission study performed at NASA, the Jet Propulsion Laboratory (JPL) would generate the interplanetary trajectories to reach Mars. The results were then given to NASA Langley Research Center (LaRC) to perform the EDL analysis required to accomplish the mission. Because MSL is introducing new and revolutionary technology in the realm of entry, descent, and landing at Mars, a large subspace of arrival dates had to be analyzed by LaRC, and then relayed back to JPL for fine-tuning of the interplanetary trajectory design. This costly relay of

information between JPL and LaRC is inefficient and introduces the possibility for error. Having the capability to perform the interplanetary analysis along with the EDL analysis would allow for much greater insight to the mission window constraints than the current iterative method.

The purpose of this research is to develop a simplified, two-body, patched conic trajectory with B-plane targeting that will yield results that will be adequate enough to define and explore the EDL capability upon arrival. The method developed by this research will be shown to closely correspond to data from JPL for the MSL mission study. A variety of planetary and physical ephemerides are introduced by the method and weighted with respect to their accuracy to the JPL data.

The fundamental ideas used in this research are Lambert's Problem, Physical and Planetary Ephemerides, Patched Conic Analysis, and B-plane Targeting. Substantial discussion of each will be presented, explaining the advantages and disadvantages of each and the overall effect on precision. Special attention will be given to Lambert's problem where a novel solution method will be presented that covers some of the difficulties often encountered by traditional Lambert solutions. A method will also be shown for optimizing the solution according to set constraints to achieve a 'best guess' interplanetary trajectory according to mission specifications. The method developed will be utilized for future launch dates and mission scenarios using the same entry, descent, and landing analysis from MSL. The results will show that a simplified, patched-conic method of interplanetary trajectory development linked with an EDL analysis can reproduce comparable results in entry speed, heat rate, maximum acceleration and

altitude at parachute deploy to that of a high fidelity precision analysis by the Mars Science Laboratory study.

2 Fundamental Astrodynamics in Interplanetary Travel

Along an interplanetary trajectory from one point to another, a spacecraft will spend the majority of the flight time moving under the gravitational influence of the sun, and only brief periods under the influence of a planet. The assumption can be made that the planet perturbations to the spacecraft heliocentric trajectory are negligible. This allows the heliocentric trajectory to be considered as a separate, two-body problem about the sun without the inclusion of the gravitational attraction of the planets. However, the heliocentric trajectory will only yield the necessary velocities required by the spacecraft to complete the interplanetary trajectory. A planetocentric trajectory is required to define the launch and arrival characteristics of the spacecraft trajectory. A patch assumption is made to connect the heliocentric trajectory to two separate, planetocentric trajectories at the point in time when the planetocentric and heliocentric trajectories coincide, thus assuring continuity in time of position and velocity, but not acceleration. This is the fundamental idea in using the patched conic method for interplanetary trajectory design.

The first step in utilizing the patched conic method is introducing the necessary astrodynamic fundamentals: trajectory types, spherical trigonometry, time, and coordinate systems.

2.1 Trajectory Types

The trajectories designed for interplanetary analysis may all contain varying degrees of precision. For mission design studies, two-body motion combined with the patched conic method is adequate. For mission operations that will require precise orbit determination, numerical integration of the equations of motion is recommended. When

the highest degree of precision is required, general relativity is used as the underlying model. For purposes of developing the interplanetary trajectories to link with the EDL analysis, the patched conic method is adequate.

The two-body motion in patched conic analysis can theoretically fall into 5 different categories of conic sections: ellipses, circles, parabolas, hyperbolas and degenerate sections. For the heliocentric trajectory, the ellipse is used. Although heliocentric trajectories are possible using hyperbolic motion, elliptical motion is much more feasible as energy costs in launching a spacecraft along a hyperbolic trajectory is impractical. However, hyperbolic motion has been achieved by a planetary gravity assist flyby.

During interplanetary transfer, planetocentric motion is usually hyperbolic. The reason is explained by the energy of a spacecraft's motion. If the spacecraft energy (E) is negative, the motion will be bounded to the spherical region of radius, $\mu/(-E)$. Only when the energy is positive, does the orbit retain an excess velocity, also known as v_∞, beyond the spherical region mentioned. If a spacecraft is to leave the Earth along an interplanetary trajectory, the spacecraft must have positive energy so that an adequate v_∞ is maintained for injection along the interplanetary trajectory. Two-body motion with positive energy is defined by a hyperbola, thereby confirming its use for planetocentric motion.

The patching of the hyperbola to the ellipse in the patched conic method will use the sphere of influence to help define the geometry of the planetocentric hyperbola. For purposes of discussion, the method will be related to the arrival hyperbola at Mars, but is equally valid for defining the hyperbola at the departure planet, Earth. First, the sphere of

influence is defined as the location where the ratio of perturbation due to Mars with respect to the central attraction of the sun, equals the ratio of perturbation exerted by the sun with respect to the central attraction of Mars. Equating these two ratios leads to the approximation for the sphere of influence radius [1]:

$$r_{SOI} = r_{1/2}\left(\frac{\mu_1}{\mu_2}\right)^{\frac{2}{5}} \qquad 2.01$$

In the case of Mars, the variable $r_{1/2}$ would be the distance from the sun to Mars. The variables μ_1 and μ_2 are the gravitational constants of Mars and the sun respectively. Assuming Mars is in a circular orbit of 1.5 AU, the radius of the sphere of influence would be approximately 5.7×10^5 kilometers. To define the patch, the $\vec{v}_\infty$ vector is first calculated as the vector difference between the spacecraft heliocentric velocity and the Mars heliocentric velocity at the time of arrival where the two orbits intersect. The patch then places the $\vec{v}_\infty$ vector at a position, $\vec{R}$ that is equal in distance to the radius of the sphere of influence from the center of Mars, along the negative direction of the $\vec{v}_\infty$ vector. This position $\vec{R}$ can then be slightly displaced perpendicular to the $\vec{v}_\infty$ vector to tailor the hyperbolic orbit about Mars. The sphere of influence is useful as it helps to define the position ($\vec{R}$) around Mars, while the patched conic method defines the spacecraft velocity ($\vec{v}_\infty$) at $\vec{R}$, thereby defining the planetocentric hyperbola.

2.2 Spherical Trigonometry

Spherical trigonometry will be used to relate the planetocentric hyperbola to specified targeting constraints. These constraints are typically latitude and longitude, but

can also apply to the central angle of an entry, descent, and landing atmospheric entry point to impact.

The celestial sphere is a fundamental concept in spherical trigonometry. The celestial sphere is defined as a spherical surface of infinite radius. The center of the sphere is arbitrary. Figure 2.1 shows a representation of the celestial sphere where the symbols denote angles. All angles are measured along great circles in the range of 0 to π. Any two points on a plane that intersects the center of the celestial sphere defines a great circle. The distance between two points on the surface of the sphere, also known as a 'side', is the central angle subtended by the two points, i.e. γ in Figure 2.1. The points can be representative of a planet pole, or target latitude and longitude. Three great circles create a spherical triangle as denoted by the bold solid lines in Figure 2.1. The angles (α,β,γ) between the planes of the great circles and the sides (a,b,c) of the great circles define the spherical triangle. The spherical triangle has the following properties.

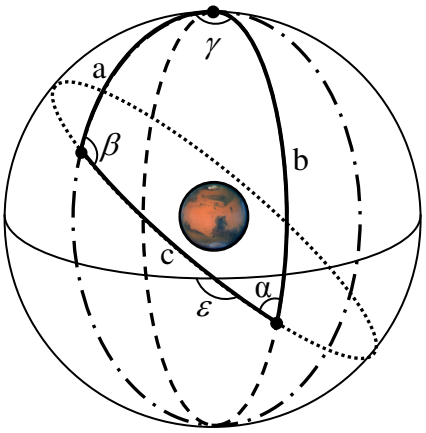

Figure 2.1: Celestial Sphere

$$\pi < \alpha + \beta + \gamma < 3\pi$$

$$0 < a + b + c < 2\pi$$

$$a + b > c, \quad \text{etc.}$$

Given three parts of a spherical triangle, spherical trigonometry is used to determine the other parts. As in plane trigonometry, there is the law of sines.

$$\frac{\sin a}{\sin \alpha} = \frac{\sin b}{\sin \beta} = \frac{\sin c}{\sin \gamma} \qquad 2.02$$

There are two laws of cosines. The first law relates one angle to three sides. The second law relates one side to three angles.

$$\cos a = \cos b \cos c + \sin b \sin c \cos \alpha \qquad 2.03$$

$$\cos \alpha = -\cos \gamma \cos \beta + \sin \gamma \sin \beta \cos a \qquad 2.04$$

When there are 4 adjacent parts on the spherical triangle, the four-part formula is useful.

$$\cos(IS)\cos(IA) = \sin(IS)\cot(OS) - \sin(IA)\cot(OA) \qquad 2.05$$

The variables of the four-part formula are the inner side, inner angle, outer side, and outer angle. For example, these variables are c, β, a, and α respectively in Figure 2.1. There are other relations, but with these formulae most spherical triangle problems can be solved.

The primary benefit of spherical trigonometry to the planetocentric hyperbola is the ability to relate the angular locations on the celestial sphere to the hyperbolic orbital elements. These elements will include true anomaly, inclination, and argument of perigee. These orbital elements will be used in developing an arrival targeting method for the planetocentric hyperbola.

2.3 Time

Ephemeris time is the independent variable in the motion of a spacecraft, planets, and other heavenly bodies. Ephemeris time is the most common time system used throughout literature when referencing interplanetary travel and coordinate systems. Ephemeris time is precisely defined according to a standard of measurement that defines a time system with respect to the coordinate system of the body included in the ephemeris. For the JPL planetary ephemeris DE405, barycentric coordinate time (TCB) is mathematically equivalent to the independent variable in the equations of motion of the

ephemeris, only differing by a constant offset and constant rate to the actual time used [2]. However, because TCB is beyond the scope of this document to calculate, the closest approximation is barycentric dynamical time (TDB), which will be used instead for querying of the planetary ephemeris. Note that even though ephemeris time is not the time system used in ephemeris creation, it will be used here for all references to time as ephemeris time is commonly used throughout literature. Using units of days, an approximate value [3] for barycentric dynamical time can be calculated from ephemeris time by:

$$\begin{aligned} c &= 357.53° + 0.9856003°/\text{day}\left(ET - 2451545.0\right) \\ TDB &= ET + 0.001658\sin c + 0.000014\sin 2c \end{aligned} \qquad 2.06$$

For querying physical and planetary ephemerides, ephemeris time (as used in equation 2.06) must be in units of Julian days. Julian days is simply a means of calculating the number of days past the epoch in time, noon, November 24, 4713 BC as defined by the Bureau International des Poids et Measures in Sèveres, France. For physical ephemerides, time must be in Julian days past noon, January 1, 2000, also known as the epoch of J2000. Since time of departure and arrival is typically given as a Gregorian date, a simple transformation is needed to convert to Julian days. An excellent algorithm is given by Jean Meeus [4],

$$\begin{aligned} A &= \langle Y/100 \rangle \qquad B = 2 - A + \langle A/4 \rangle \\ JD &= \langle 365.25(Y + 4716) \rangle + \langle 30.6001(M + 1) \rangle + D + B - 1524.4 \end{aligned} \qquad 2.07$$

where Y is the integer Gregorian year, M is the integer month and D is the number of days, including any fractional part. The carrot-brackets are used to denote integer part of

the argument. The term JD is the Julian day of date. For example, the epoch of J2000, January 1.5, 2000, is 2,451,545.0 Julian days using Meeus's algorithm.

2.4 Coordinate Systems

The coordinate systems used in interplanetary mission design are the perifocal, heliocentric, IAU, B-plane, Earth mean equator and mean equinox of J2000 (EME) and Earth mean ecliptic and mean equinox of J2000. The primary conventions used when defining and referencing coordinate systems are the location of the origin, orientation of the fundamental plane, and orientation of the fundamental direction.

2.4.1 Perifocal

The perifocal coordinate system is used commonly to define the motion of a spacecraft orbit. The origin is at the center of gravitational attraction located at the primary focus of the respective two-body conic section. The fundamental plane is the plane of motion. The fundamental direction is towards periapse measured from the origin of the perifocal system. An example of the perifocal system is shown in Figure 2.2.

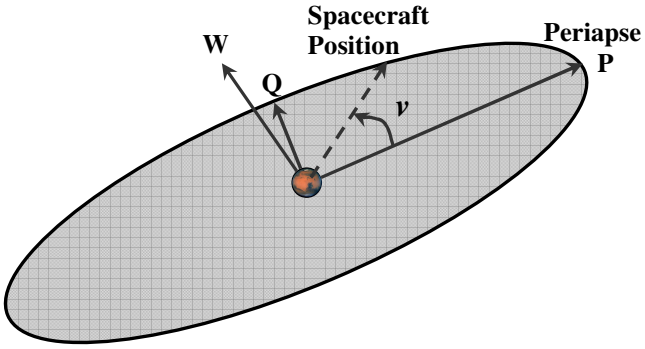

Figure 2.2: Perifocal Coordinate System

The variable v is the true anomaly of the spacecraft's position, measured counterclockwise from the fundamental direction (P) along the fundamental plane. The

axis W points in the direction of the spacecraft's angular momentum. The axis Q completes the right-handed system. The perifocal system's relationship to the orbital parameters true anomaly and angular momentum, among other parameters, make it useful when targeting the heliocentric and planetocentric trajectories to specified locations.

2.4.2 Earth Mean Equator and Mean Equinox of J2000 (EME)

The International Astronomical Union (IAU) has recommended a common coordinate system from which all precision coordinates are measured, known as the Earth mean equator and mean equinox of J2000 (EME). The reference to a 'mean' value is used to define a system that ignores the small variations of short period in the motions of the equator and is only affected by the average precession of the equator [3]. For purposes here, the origin of the EME of J2000 system is at the Earth center of mass. The fundamental plane is the Earth mean equator of J2000. The Earth mean equator is the plane perpendicular to the Earth axis of rotation at the epoch of J2000. Note that the equator is not fixed in space, but moves over time. The fundamental direction is the vernal equinox of J2000. The vernal equinox is the point where the direction of motion of the sun along the ecliptic moves from the southern hemisphere to the northern hemisphere. The unit vector defining this point in the EME system is from the Earth center of mass to the Sun center of mass at the time of the vernal equinox. The EME coordinate system is shown in Figure 2.3.

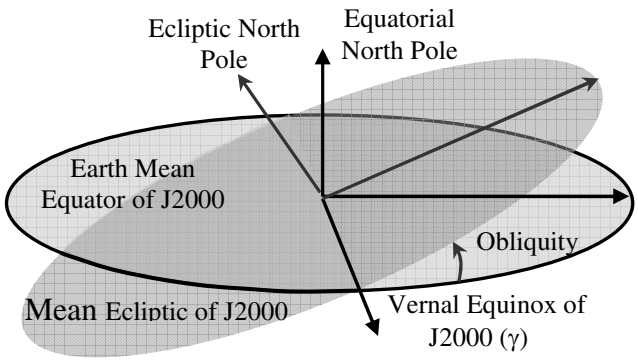

Figure 2.3: Earth Mean Equator and Earth Mean Ecliptic Coordinate Systems

The EME system is closely related to the Earth mean ecliptic system of J2000 as shown in Figure 2.3. For purposes here, the Earth mean ecliptic system shares the same origin and fundamental direction as the EME system, but has a different fundamental plane. The fundamental plane is the Earth mean ecliptic at the epoch of J2000. The EME and Earth mean ecliptic system can be related by one angle, the obliquity. The Earth mean ecliptic system can however be considered as a heliocentric coordinate system. The Earth mean ecliptic system will be the coordinate system used for the calculation of the interplanetary trajectory from Earth to Mars.

2.4.3 IAU Physical Ephemerides

The IAU physical ephemeris system defines the planet mean equator and prime meridian of date. Hence, the fundamental plane is the mean planet equator. The fundamental direction is the prime meridian. The origin is located at the center of mass of the planet. The IAU provides the physical ephemeris parameters by cataloguing the motion of the planets and planetary features with respect to the epoch January 1.5, 2000.

The IAU coordinate system is defined relative to the EME coordinate system. To determine the IAU coordinate system's orientation, a 3-1-3 Euler angle rotation is used

beginning at the orientation of the EME system. The first rotation is about the z-axis or North Pole of the EME system. This rotates the coordinate system by $(\alpha + 0.5\pi)$ where α is the right ascension of the planet pole of date, bringing the new x axis to the ascending node of the planet equator with respect to the EME system. The second rotation of $(0.5\pi - \delta)$ is about the new x-axis where δ is the declination of the planet pole of date. At this point fundamental plane of the coordinate system is the planet mean equator of date ($\perp$ to z-axis), and the fundamental direction (x-axis) is the *IAU vector of date*. The IAU vector lies in both the Earth and planet equators. The third rotation is about the new z-axis by the number of revolutions the planet makes about the planet north pole at the specified time. Specifically, the third rotation is the angle traveled from the IAU vector to the position of the prime meridian of date. The rotation matrix used to transform from the EME to the IAU frame is given by

$$[\mathrm{R}]_{EME/IAU} = \begin{bmatrix} \cos W & \sin W & 0 \\ -\sin W & \cos W & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(0.5\pi - \delta) & \sin(0.5\pi - \delta) \\ 0 & -\sin(0.5\pi - \delta) & \cos(0.5\pi - \delta) \end{bmatrix} \begin{bmatrix} \cos(\alpha + 0.5\pi) & \sin(\alpha + 0.5\pi) & 0 \\ -\sin(\alpha + 0.5\pi) & \cos(\alpha + 0.5\pi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad 2.08$$

The variable *W* represents the angle to the prime meridian from the IAU vector. Figure 2.4 shows the IAU coordinate system along with angles α, δ, and W in equation 2.06.

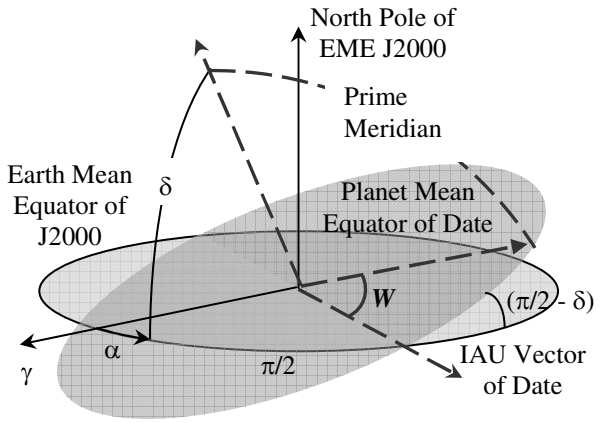

Figure 2.4: IAU Coordinate System

2.4.4 B-plane Coordinate System

The B-plane coordinate system origin is often assumed to be in one of two locations. For purposes here, the origin is located along the incoming/outgoing asymptote at a distance equal to the radius of the sphere of influence from the attracting body. The other location is the center of mass of the attracting body. Recall that the $\bar{v}_\infty$ vector defines the direction of the radius vector to the sphere of influence in the patched conic method. Hence, the fundamental plane, also called the B-plane, is perpendicular to the $\bar{v}_\infty$ vector. The fundamental direction is in the plane of the planet equator, oriented perpendicular to the $\bar{v}_\infty$ vector.

The B-plane coordinate system is shown in Figure 2.5. The incoming hyperbolic trajectory, denoted 'Trajectory', has asymptote perpendicular to the fundamental plane (B-plane). The point of intersection of the asymptote (parallel to $\hat{S}$) with the B-plane is the location of entry measured with respect to the origin by B and θ. The orientation of the trajectory plane with respect to the planet equator defines the B-plane angle, θ. The fundamental direction $\hat{T}$ is in the same plane as the reference equator. The direction $\hat{R}$ completes the right hand system.

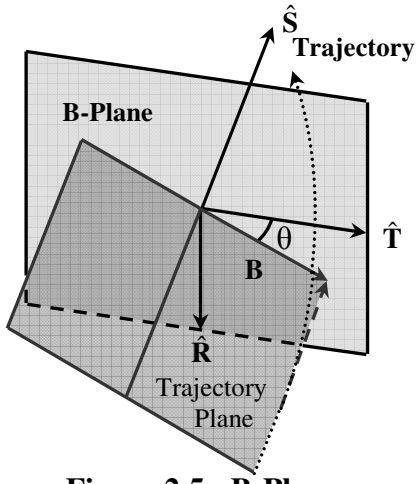

Figure 2.5: B-Plane Coordinate System

The B-plane coordinate system is used to target the planetocentric hyperbola at arrival. Since the fundamental plane is defined by the orientation of the $\bar{v}_\infty$ vector, the B-plane orientation is a function of the velocity at arrival on the heliocentric ellipse, which

is a function of the launch and arrival dates. Hence, the B-plane coordinate system provides an important link to the interplanetary trajectories and planetocentric hyperbolas on any given launch and arrival date.

The hyperbola is defined in the B-plane coordinate system by B, the $\vec{v}_\infty$ vector and B-plane angle θ. The value B is the small displacement perpendicular to the $\vec{v}_\infty$ vector at the sphere of influence described by the patched conic method. Since the displacement is perpendicular to the $\vec{v}_\infty$ vector, only θ is needed to define the entry location on the B-plane. Furthermore, because the $\vec{v}_\infty$ vector is perpendicular to the B-plane, the angular momentum of the hyperbola is the product, $h = Bv_\infty$. The magnitude of B can therefore be related to position and velocity of the hyperbola by $Bv_\infty = rv\cos\gamma$. If the flight path angle is set to zero, the magnitude of B can be related to periapsis by

$$B = \frac{\mu}{v_\infty^2}\sqrt{\left(1+\frac{v_\infty^2 r_p}{\mu}\right)^2 - 1} \qquad 2.09$$

where the variable μ is the planet gravitational constant. Therefore, by knowing just the magnitude of $\vec{v}_\infty$, one can target the planetocentric hyperbola to a specified periapse radius by displacing the entry location on the sphere of influence by B.

By knowing the B-plane coordinate system orientation with respect to the planet equator, the B-plane parameters θ and B can be used to target the desired hyperbolic trajectory about the planet. B is used to target the radius of periapse of the hyperbola. To target inclination, the B-plane orientation with respect to the planet equator is utilized. Since the fundamental direction $\hat{T}$ is in the plane of the equator, the B-plane coordinate system can be rotated about $\hat{T}$ by the declination of the $\vec{v}_\infty$ vector (δ) plus 0.5π to align

the B-plane fundamental plane with the planet equator. The alignment of the B-plane fundamental plane with the equator is an arbitrary planet equator coordinate system and is only used for the derivation of the equation for inclination. The B vector is defined in the B-plane coordinate system by $\vec{B} = B[\cos\theta \quad \sin\theta \quad 0]$. By rotating about $\hat{T}$, the B vector becomes

$$\vec{B} = B[\cos\theta \quad \sin\theta\cos(\delta + \pi/2) \quad -\sin\theta\sin(\delta + \pi/2)]$$

in the arbitrary planet equator system. From the 2-body equations, inclination is $\cos i = \frac{h_z}{h}$, where the angular momentum is the cross product of the $\vec{v}_\infty$ vector and the B vector in the arbitrary planet equator system.

$$h = \begin{bmatrix} B\cos\theta \\ B\sin\theta\cos(\delta + \pi/2) \\ -B\sin\theta\sin(\delta + \pi/2) \end{bmatrix} \times \begin{bmatrix} 0 \\ v_\infty\cos\delta \\ v_\infty\sin\delta \end{bmatrix} = \begin{bmatrix} \sin\delta\sin\theta\cos(\delta + \pi/2) - \cos\delta\sin\theta\sin(\delta + \pi/2) \\ -\cos\theta\sin\delta \\ \cos\theta\cos\delta \end{bmatrix}$$

By substituting the z component of the angular momentum vector and the angular momentum magnitude into the 2-body equation for inclination, the B-plane angle θ becomes a function of only the inclination and the declination of the $\vec{v}_\infty$ vector.

$$\cos\theta = \frac{\cos i}{\cos\delta} \tag{2.10}$$

Note that since the absolute value of $\cos(\theta)$ must be less than 1, the lowest inclination is the declination of the launch asymptote, δ.

Equations 2.09 and 2.10 can be used in conjunction with the B-plane coordinate system to target the planetocentric hyperbola to a specified inclination and periapse radius. With the B-plane coordinate system being used to provide the link between the

heliocentric and planetocentric trajectories, equations 2.09 and 2.10 provide a realization of how the two trajectories are linked.

2.5 Hohmann Transfer

The Hohmann transfer is a minimum fuel transfer between two, circular, coplanar orbits. The transfer is elliptical and utilizes two impulsive maneuvers, one at periapsis and one at apoapsis of the transfer orbit (Figure 2.6). Since many of the planets in the solar system exhibit orbits with low eccentricity, the Hohmann transfer can be a good starting point in defining an interplanetary trajectory. The Hohmann transfer places strict conditions on time of flight and the orientation of the planes of the orbits. Therefore, application of the transfer is rarely feasible in the real world. Regardless, the Hohmann transfer will be utilized to approximate optimal times for interplanetary trajectories.

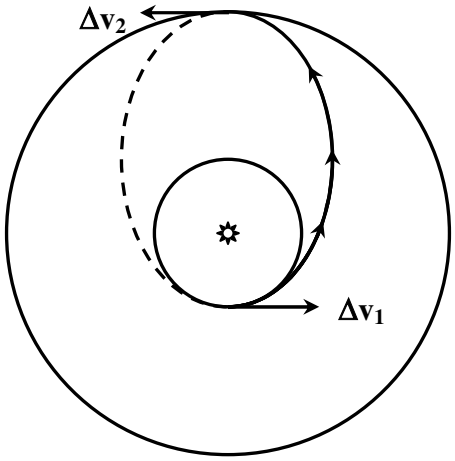

Figure 2.6: Hohmann Transfer

The primary focus in interplanetary travel is the trajectory taken from Earth to Mars over the specified time interval. The patched conic method utilizes a heliocentric trajectory calculated in the Earth mean ecliptic system of J2000 to describe the motion taken from Earth to Mars with the Hohmann transfer being the ideal trajectory. The difference of the heliocentric trajectory velocity and the heliocentric planet velocity provides the $\vec{v}_\infty$ vector to be used in the B-plane coordinate system. The B-plane coordinate system can then be used to determine the properties of a planetocentric hyperbola about the target planet. From the planetocentric hyperbola, the perifocal,

EME, and IAU coordinate systems are used in conjunction with spherical trigonometry to target to the desired location on Mars. The details of these calculations will be described next, starting with Lambert's problem.

3 Lambert's Problem

In interplanetary calculations, trajectories from planet to planet are determined by selecting a launch (departure) and arrival date. From a planetary ephemeris, the positions of the launch and arrival planets are determined for the selected launch and arrival date. With position and time known, the determination of the trajectory connecting the two points is commonly referred to as a two-point boundary value problem. The solution to this problem is covered by Lambert's theorem.

Lambert's theorem, initially formulated in 1761, states that the time of flight between two points, P_1 and P_2, is only a function of the semi-major axis, sum of the distances from the primary focus to each of the two points, and the chord length between P_1 and P_2:

$$\sqrt{\mu}(t_2 - t_1) = f(a, r_1 + r_2, c) \tag{3.01}$$

In 1802, Carl Friedrich Gauss reformulated the problem to practical applications, stating that only two position vectors, the direction of motion, and the time of flight between the two points are required for a solution. The term 'direction of motion' refers to selecting the transfer trajectory that traverses an angular distance in the same plane either less than or greater than π radians for the same two points. Ever since Gauss's restatement of Lambert's problem, solutions have proliferated throughout literature. Since there has been no direct solution of Lambert's problem, iterative methods and series expansions have become the accepted solutions.

3.1 Selection of Method to Solve Lambert's Problem

Bate, Mueller, White[5], and Battin[1] have provided some of the more popular techniques for solving Lambert's problem because of their elegance and robustness for a large variety of transfer orbits. Thorne [6] has derived a series solution to Lambert's problem using series reversion and inversion that works for hyperbolic and elliptic orbits. Prussing and Conway [7] present the classical Lambert solution by iterating on the semi-major axis. Other Lambert solutions have been derived that iterate on the orbital elements including, but not limited to, semi-latus rectum, true anomaly, and eccentricity.

Each of these solutions possesses different advantages and disadvantages. The classical solution to Lambert's problem, by Prussing and Conway [7], develop a method that introduces two auxiliary angles, α and β that represent the change in eccentric anomaly and allow multiple revolutions. The auxiliary angles α and β introduce difficulty though in the determination of the quadrants. Gedeon [8] has presented a method for determining the solution using a Newton iteration scheme that also takes care of the quadrant ambiguities using α and β. Gedeon's method however utilizes numerous checkpoints in the solution causing unneeded complexity. Battin's solution has been shown to converge quickly and efficiently but becomes troublesome when implementing multiple revolutions. The Bate, Mueller and White universal variables solution provides an ideal method for multiple-revolution transfers. For the purposes of interplanetary trajectory design, the Bate, Mueller and White universal variables solution is chosen as it provides an elegant and robust algorithm that accommodates multiple revolution transfers.

The advantage of the multiple revolution capability is that the solution may be used to reduce the eccentricity of an elliptic transfer while still achieving the final destination in the desired flight time, in effect reducing the propulsive Δv required at the expense of longer flight times. This is illustrated in Figure 3.1, where the 2-revolution transfer is advantageous as the v_∞ required at injection is one-third of the 0-revolution transfer.

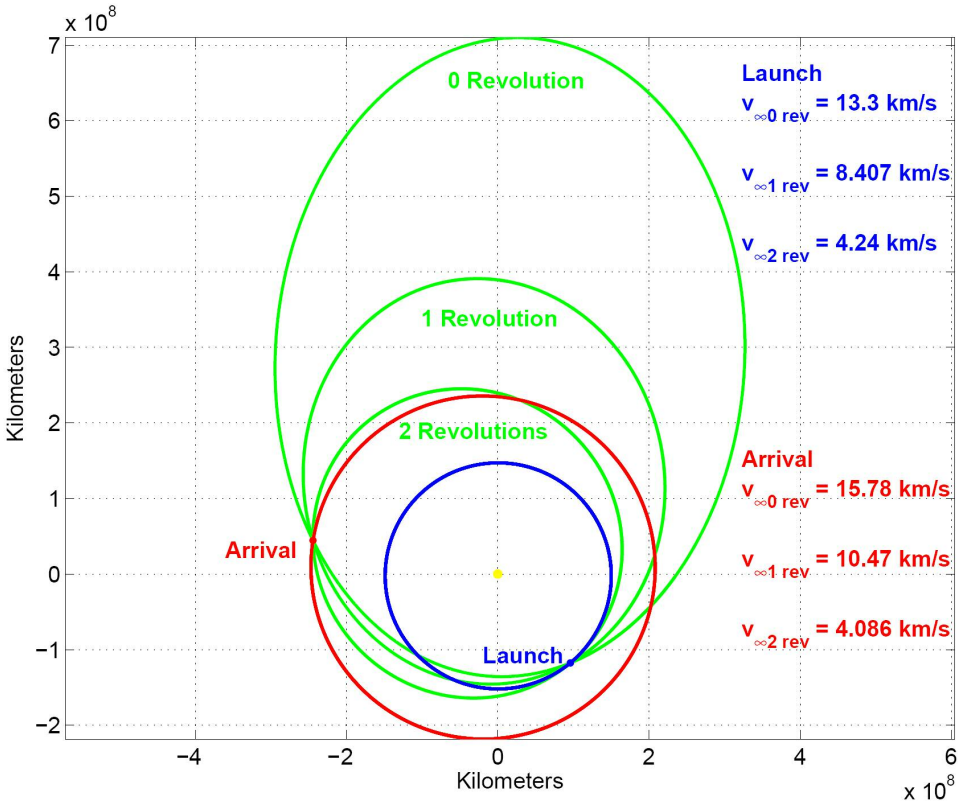

Figure 3.1: Multiple Revolution Orbits

The examples shown in Figure 3.1 are unrealistic for missions to Mars. However, the multiple revolution capability is important as the interplanetary trajectories considered can occur near 360 degrees true anomaly. For the trajectories that slightly exceed a true anomaly of 360 degrees, the solution needs to account for one revolution plus the extra bit required to reach Mars. If the solution method does not include multiple revolutions, the resulting trajectory would likely be a highly eccentric ellipse, like the zero-revolution and one-revolution trajectories seen in Figure 3.1.

Another advantage to multiple revolutions can be noted in cycler trajectories. For a cycler trajectory, a supply craft is left in a multiple revolution orbit about the sun that is utilized by other spacecraft en-route. The multiple revolutions Lambert capability permits targeting nearby planets while remaining in a multiple revolution cycler trajectory.

The universal variables Lambert solution accommodates multiple revolutions because the variable of iteration is the difference in eccentric anomaly. This allows a simple approach to determine a multiple revolution Lambert solution because the desired number of revolutions explicitly defines the bounds of the feasible regions of the variable of iteration, i.e. a 1 revolution trajectory is bounded by a difference in eccentric anomaly of 2π to 4π. This is a substantial advantage over the other Lambert solution methods, as a multiple revolution capability becomes difficult to implement in the other methods due to the variables of iteration being in terms of orbital parameters that do not account for an angle that is greater 2π. To explain this advantage over the other methods, begin with Kepler's equation.

$$M = n(t - \tau) = E - e\sin E \qquad 3.02$$

With Kepler's equation, time from periapse is defined by $t-\tau$. The variable n is the mean

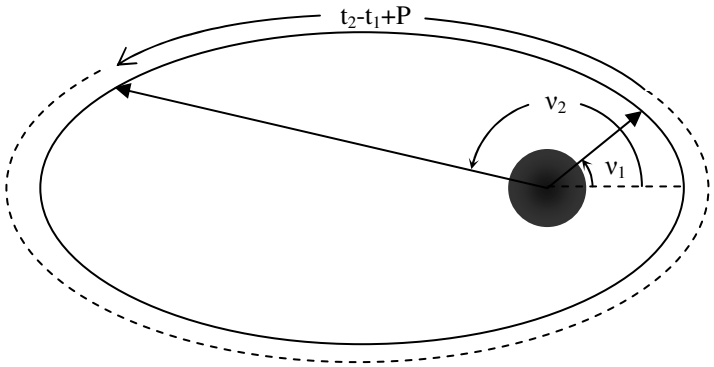

Figure 3.2: Two Point Time of Flight Representation

motion, e is the eccentricity, and E is the eccentric anomaly. From Figure 3.2, the time of flight spent during multiple revolutions can be represented as:

$$t_2 - t_1 = kP + (t_2 - \tau) - (t_1 - \tau) \tag{3.03}$$

P is the orbital period, k is the number of revolutions taken, and τ is the time of periapse. Orbital period is given by

$$P = 2\pi\sqrt{\frac{a^3}{\mu}} \tag{3.04}$$

where a is the semi-major axis and μ is the gravitational constant. By substituting Kepler's equation (3.02) for each of the time from periapse terms in equation 3.03, and including equation 3.04, the time of flight between two points is:

$$t_2 - t_1 = \sqrt{\frac{a^3}{\mu}}\left[2k\pi + (E_2 - e\sin E_2) - (E_1 - e\sin E_1)\right] \tag{3.05}$$

Equation 3.05 represents the start of the classic formulation of Lambert's problem. If 4 of the 5 variables, i.e. semi-major axis, eccentricity, time of flight and eccentric anomaly of the two points are known, the remaining variable may be determined. The universal variables formulation uses equation 3.05 to accommodate multiple revolutions by using the difference of the two eccentric anomaly terms and the $2k\pi$ term. The other variables, as used by the other Lambert methods, become much more difficult to incorporate the $2k\pi$ term due to quadrant ambiguities. This makes the universal variables multiple revolution solution simple and effective in comparison to the other Lambert methods.

Since equation 3.05 is transcendental, it is impossible to solve directly for the change in eccentric anomaly between the two points. Thus an iterative solution is required.

3.2 Lambert Solution Method

The derivation of the universal variable time of flight equation is given in Appendix A and is shown here as

$$\sqrt{\mu}t = \chi^3 S + A\sqrt{y} \tag{3.06}$$

where t is the time of flight between the two points. The variables S, A, y, and χ are derived in Appendix A as equations 9.14, 9.17, 9.18, and 9.19 respectively.

$$\begin{aligned}
&\text{For Elliptical Motion :}\\
&S(z) = \frac{\sqrt{z} - \sin\sqrt{z}}{\sqrt{z^3}}\\
&z = (E_2 - E_1)^2\\
&\text{For Hyperbolic Motion :}\\
&S(z) = \frac{\sinh\sqrt{-z} - \sqrt{-z}}{\sqrt{-z^3}}\\
&z = -(F_2 - F_1)^2
\end{aligned} \tag{9.14}$$

$$A = \frac{\sqrt{r_1 r_2}\sin\Delta v}{\sqrt{1-\cos\Delta v}} \qquad 0^\circ < \Delta v < 180^\circ \tag{9.17}$$

$$y = r_1 + r_2 - A\frac{(1 - zS)}{\sqrt{C}} \tag{9.18}$$

$$\chi = \sqrt{\frac{y}{C}} \tag{9.19}$$

The variable C is derived in Appendix A as equation 9.13.

For Elliptical Motion :

$$C(z) = \frac{1 - \cos\sqrt{z}}{z}$$

For Hyperbolic Motion :

$$C(z) = \frac{1 - \cosh\sqrt{-z}}{z} \qquad 9.13$$

The variable A is a function of the given constants in Lambert's problem: the radius of the two points and angle between the two position vectors. The variable A is very powerful as it defines the quadrant of the ellipse by inclusion of the sine and cosine terms for a right handed coordinate system, thus removing the quadrant ambiguity seen in Prussing and Conway's Lambert solution. If the transfer between the two points is to be retrograde, A is negative. The remaining variables are only functions of the difference in the square of the eccentric anomalies denoted by $z = (E_2 - E_1)^2$. The variable z is equally valid for hyperbolic orbits as the change in hyperbolic anomaly, as is utilized in equations 9.13 and 9.14 if z is less than zero. From Bate, Mueller, and White [5], it is shown that the eccentric and hyperbolic anomalies are related by $\cosh F = \cos E$. By using the identity $\cosh\theta = \cos i\theta$, it is seen that $E = \pm iF$. Therefore, the change in hyperbolic anomalies can be denoted by $z = -(F_2 - F_1)^2$, and is therefore always negative.

To solve for the change in eccentric anomaly (z) a Newton iteration scheme is implemented. When implementing the Newton iteration scheme, the guess generated by the Newton iteration can possibly result in a negative y, creating an imaginary solution. Care must be taken so that the guess does not place y below zero. If y is found to be negative, the solution method should step back towards the previous z in the Newton iteration until y becomes positive.

Before implementing the Newton iteration solution, properties of the bounds of the variable of iteration need to be considered. Multiple revolution orbits can be calculated for the same time of flight, as shown in Figure 3.1. This means that there are multiple solutions to equation 3.06. This forces a requirement to place bounds on the variable of iteration, z. The bounds on the variable of iteration for a specified number of revolutions is the square of the difference in eccentric anomaly for the specified number of revolutions, i.e. if 1 revolution is specified, $z_{\text{lower}} = (2\pi)^2$ and $z_{\text{upper}} = (4\pi)^2$. The multiple revolution bounds will be incorporated into the Newton iteration as a convergence check for the correct solution.

The behavior of equation 3.06 is shown in Figure 3.3 for two positions utilizing solutions across a range of z and time of flight. Although the interplanetary transfers are

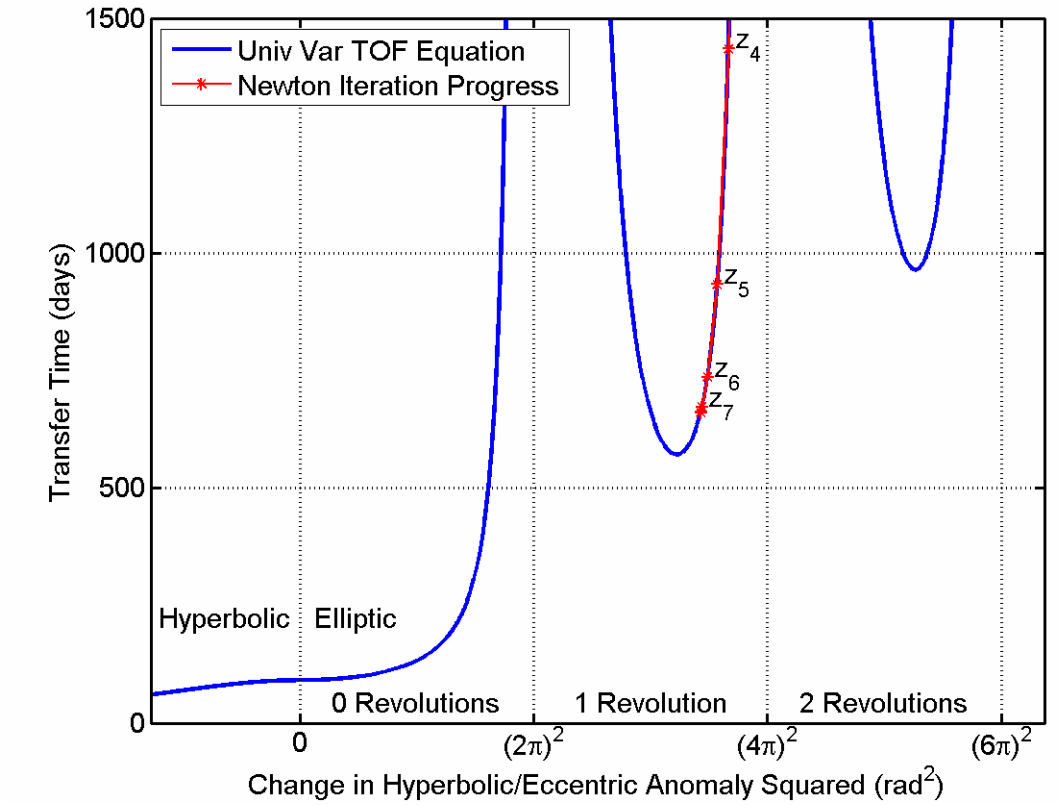

Figure 3.3: Behavior of Universal Variables Time of Flight

nominally elliptical, hyperbolic transfers are possible, as denoted in Figure 3.3 for negative change in hyperbolic anomaly squared. The x-axis in Figure 3.3 is the variable of iteration (z), the y-axis is the time of flight from position 1 to position 2, and the blue line represents equation 3.06 evaluated for the range of z and time of flight.

Figure 3.3 shows that a starting point anywhere in the 0 Revolutions region will be located in a region of only one possible change in eccentric anomaly for a specified transfer time. A starting point in the 1 Revolution region has the possibility to converge to either of two changes in eccentric anomalies. This leads to the need of setting a starting point in the multiple revolutions cases to converge to the desired change in eccentric anomaly. For Newton iteration, convergence will occur along the side of the blue line in the multiple revolution cases if a starting point is selected near the bounds of $2\pi \times n_{orbit}$; the red solid line in Figure 3.3 demonstrates this convergence where starting point was placed near the bound at $(4\pi)^2$. Therefore, the starting point should be designated at a small offset from the bounds for multiple revolution cases to converge on the desired solution.

The difference between the two, equal time of flight solutions on the multiple revolution curves, is the eccentricity of the transfer orbit. For clarification, the difference between the two solutions is shown as the dotted and dash-dot lines in Figure 3.4. The dash-dot orbit is a solution that converges from the left of the multiple revolution curves in Figure 3.3, while the dotted orbit is the solution that converges from the right of the multiple revolution curves. The solid black orbit represents the minimum time of flight solution of equation 3.06 in the 1 revolution region. The blue orbit represents the orbit of

the launch planet (Earth) and the red orbit represents the orbit of the arrival planet (Mars). The direction of motion of each orbit is counter clockwise.

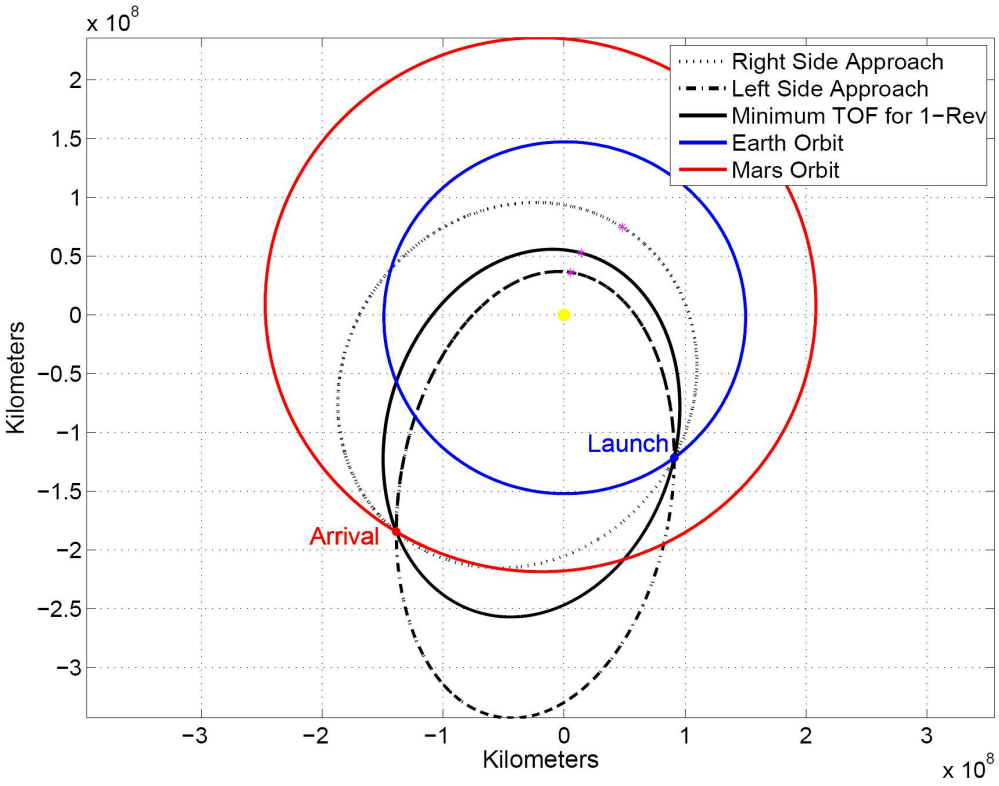

Figure 3.4: Dual Solutions of the Same Transfer Time

It is interesting to note the arrival vector for the solutions have different angles of approach at the target planet. This will cause a different time of day at arrival. In interplanetary mission planning, time-of-day at arrival is often an important factor. Designing a multiple revolution trajectory is one way to tailor the time of day at arrival according to the mission specifications.

Now that the bounds and regions of convergence are identified, the iterative solution to equation 3.06 can be explained. The first derivative of equation 3.06 is required for the Newton iteration. From Appendix A, the first derivative is:

$$\frac{dt}{dz} = \left(\chi^3 \left(S' - \frac{3SC'}{2C} \right) + \frac{A}{8} \left(\frac{3S\sqrt{y}}{C} + \frac{A}{\chi} \right) \right) \Big/ \sqrt{\mu} \qquad 3.07$$

The variables C´ and S´ are the derivatives of C and S with respect to the variable z, derived as equations 9.31 and 9.32 in Appendix A.

$$\frac{dC}{dz} = \frac{1}{2z}\left(1 - zS - 2C\right) \tag{9.31}$$

$$\frac{dS}{dz} = \frac{1}{2z}\left(C - 3S\right) \tag{9.32}$$

The Newton iteration for solving Lambert's problem is set up as:

$$z_{k+1} = z_k + \frac{\left(t_{target} - t_k\right)}{\left(\frac{dt}{dz}\right)_{z=z_k}} \tag{3.08}$$

As seen from equation 3.08, the Newton iteration will converge by a series of linear iteration steps. This presents a problem for convergence because of the presence of multiple solutions. In Figure 3.3, if the desired time of flight exceeds 570 days the Newton iteration can place the solution in the 1 revolution region, yielding an undesirable answer. Since the solution is bounded, the algorithm should recognize that this is not a correct answer. There are many remedies to this problem, however, only two will be presented. The first remedy is to use a bisection iterative method. The second remedy is to modify the algorithm to select a new point when the bounds are violated from which to continue the Newton iteration.

The bisection solution method is very stable as it guarantees convergence in the 0 revolution region. Bisection is performed by setting initial bounds on the solution, as has already been done for the variable z according to the number of revolutions, and taking the average of the two bounds. If the average value yields a transfer time that is larger than the desired time, the lower bound is retained, and the upper bound is set to the previous average. If the transfer time is smaller than the desired time, the upper bound is

retained and the lower bound is set to the previous average. This continues in a loop until the solution is bounded to within a specified convergence criterion. This method is costly in terms of computation (over 40 iterations are typically required), but does guarantee convergence for the 0 Revolution elliptical case. The difficulty of using a bisection technique for all cases is in the multiple revolution cases where the variable z has multiple solutions for the same time of flight. Also, hyperbolic cases have difficulty because the solution may not be captured by the initial bounds of the bisection method.

The multiple revolution curves require knowledge of the minimum time of flight on the selected multiple revolution curve so that the bisection technique may be bounded correctly. From Kepler's equal area in equal time law, the minimum time of flight for multiple revolutions is related to the minimum amount of area swept out with respect to the variable of iteration.

$$\frac{\mathrm{d}(\mathrm{Area})}{\mathrm{d}(z)} = 0 \quad , \quad \sqrt{z} > 2\pi$$

Finding the minimum area with respect to orbital parameters semi-major axis, semi-latus rectum, and eccentricity is non-applicable since such a solution will always assume zero revolutions. The solution will need to be derived in terms of difference in eccentric anomaly that is greater than 2π to accommodate multiple revolutions (difference in true anomaly is also applicable). Since Kepler's time of flight equation is derived from Kepler's equal area law and is transcendental for eccentric anomaly, it follows that the solution to differentiating the area with respect to the eccentric anomaly would be transcendental. Two possible methods for finding the minimum time of flight can be curve fitting the multiple revolution region of the curve in Figure 3.3 and solving for the minimum, or finding the point at which the first derivative of equation 3.06 is equal to

zero by implementing another Newton iteration method using the second derivative of equation 3.06. This is costly and ineffective though, giving reason to return to the original Newton algorithm while making the appropriate changes to fix the convergence problem for multiple revolution regions.

The proposed solution for the universal variables Lambert algorithm is to incorporate both the Newton iteration method and the bisection method so that when the bounds are violated by the Newton iteration, bisection iteration is used to select a new point from which to continue. As stated earlier, the Newton iteration will always converge along the sides of the blue curves in Figure 3.3 and therefore on the desired multiple revolution solution if the starting point is placed at a small offset from the multiple revolution bounds. Therefore, the bisection iteration that is incorporated will only need to address zero revolution transfers. Furthermore, because the interplanetary transfer is assumed to be elliptical, the bisection iteration will only have bounds from zero to $(2\pi)^2$. Just as the bisection algorithm explained above replaces either the lower or upper bound after each iteration, so will the bounds be changed in the same manner for the combined Newton-bisection algorithm.

$$\begin{aligned} &\text{if } z_{k+1} > (2\pi)^2 && \rightarrow \quad z_{k+1} = \frac{z_{lower} + z_{upper}}{2} \\ &\quad \text{if } t(z_{k+1}) > t_{desired} && \rightarrow \quad z_{upper} = z_{k+1} \\ &\quad \text{else} && \rightarrow \quad z_{lower} = z_{k+1} \end{aligned} \tag{3.09}$$

Using the same positions by the Lambert solution in Figure 3.3, Figure 3.5 details the computational efficiency of the bisection Lambert algorithm (red) and the combined Newton-bisection Lambert algorithm (black). The time of flight is plotted along the x-axis, number of iterations is plotted along the primary y-axis, and change in eccentric

anomaly (green) is plotted along the secondary y-axis. The dotted green line separates the hyperbolic solutions from the elliptic solutions. The variable of iteration was started at a value of zero. The solutions are all located in the 0 revolution region of Figure 3.3.

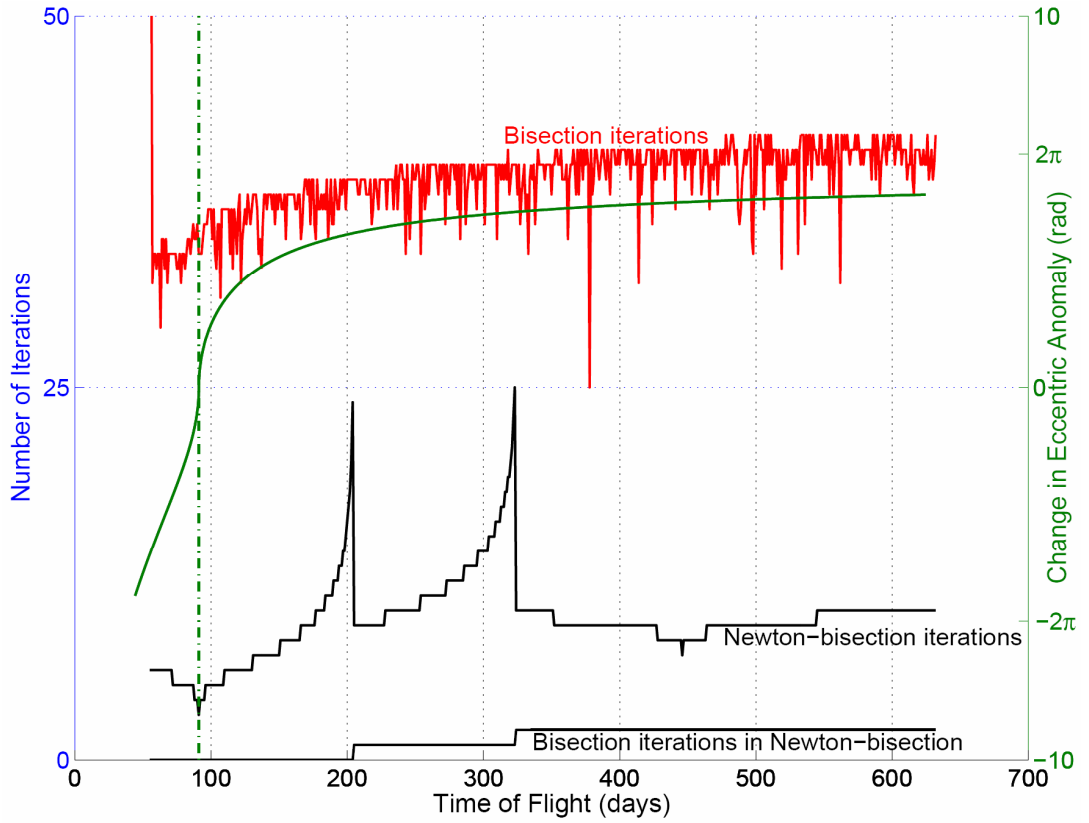

Figure 3.5: Lambert Solution Computational Cost

By analyzing Figure 3.5, a direct rise in computational effort is seen for both algorithms as the change in eccentric anomaly rises from zero. The performance of the Newton-bisection solution of equation 3.06 can be seen to spike in the 200 to 300 day time of flight region before bisection is required. This is inefficient, especially in the calculations required by interplanetary transfers from Earth to Mars, as transfer times are typically on the order of 180 to 270 days. The lack of performance of the solution can be attributed to the starting point of the iteration. The current starting point for z has been arbitrarily selected as zero. Since the bisection corrects the iteration from 'jumping' to

multiple revolution curves, as is done at the spikes by the black lines in Figure 3.5, the ideal starting point would be where the solution will have the least likelihood of 'jumping' to the multiple revolution curves. Figure 3.6 explains this jump, where the red lines represent the region of the curve where the next iteration will be placed on the multiple revolution curve and the blue lines represent where the next iteration will be placed on the zero revolution curve. Figure 3.6 is for a 0 revolution solution with desired time of flight equal to 850 days, where the green line shows the solution progress. The

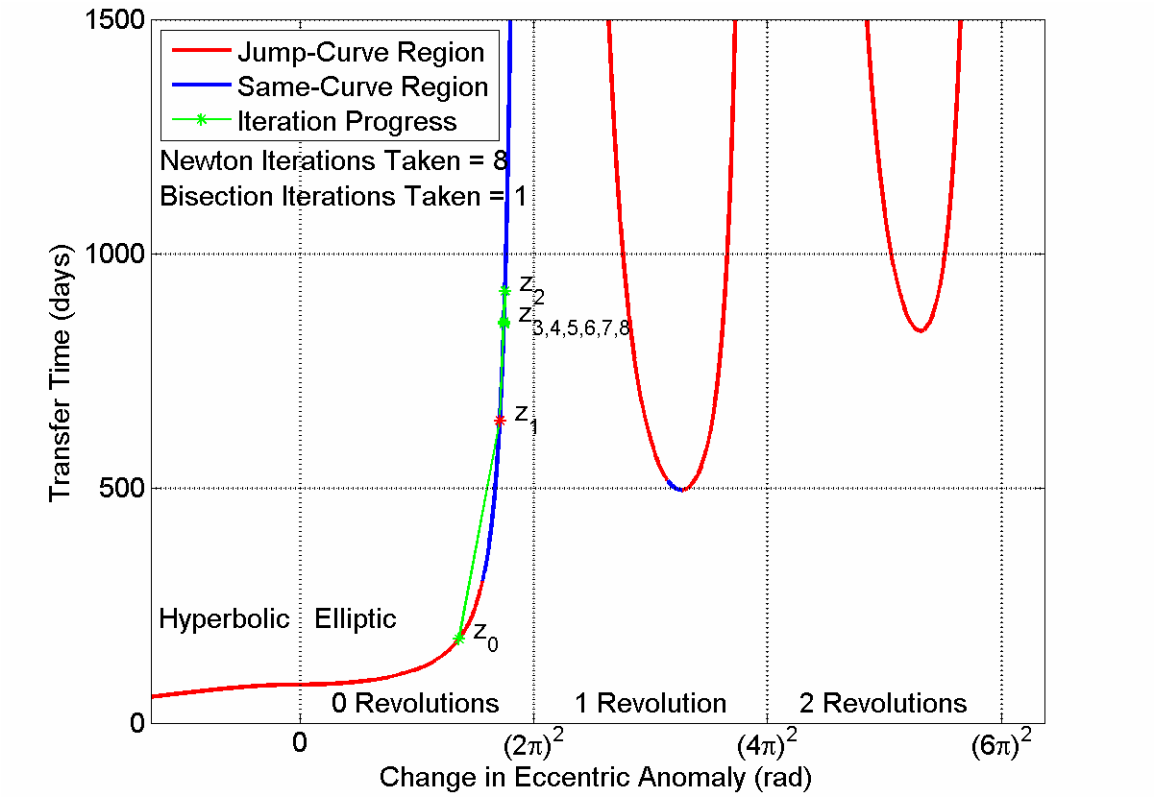

Figure 3.6: Regions of Suitable Newton Iterations

green stars are the steps in the Newton iteration while the red star represents a step taken by the bisection iteration. The large time of flight is impractical but is selected to plainly show the possibility of the iteration jumping to the multiple revolution curves from the zero revolution curve. Note that the iteration can return to the zero revolution curve from

the multiple revolution curve because the desired-time-of flight is greater than the minimum time of flight for multiple revolutions.

To reduce the number of bisection iterations necessary, a valid starting point is desired that will not place the iteration on a multiple revolution curve. The point where the solution switches from the correct curve to the one revolution curve is represented by the following equation:

$$t_{tof} - t = \frac{dt}{dz}\left((2\pi)^2 - z\right)$$

The variable t_{tof} is the desired time of flight, t is the universal variables time of flight (equation 3.06), dt/dz is the first derivative of the time of flight (equation 3.07), and z is the square of the change in eccentric anomaly. The variables t and z represent the location of the blue-red transition point in Figure 3.6. By direct substitution of equations 3.06 and 3.07 into the above equation, a transcendental equation in z is determined:

$$t_{tof}\sqrt{\mu} = \frac{x^3 S}{(2\pi)^2 - z} + \frac{A\sqrt{y}}{(2\pi)^2 - z} + 3x^2\frac{dx}{dz}S + x^3\frac{dS}{dz} + \frac{A}{2\sqrt{y}}$$

Since the above equation is transcendental for z, an iterative procedure is required. Since it is inefficient to iteratively solve for the variable z at the blue-red transition point, other methods should be explored to find an effective starting point.

Instead of trying to formulate the location of the transition point, the problem's relation to interplanetary trajectories will instead be considered. For interplanetary calculations, the trajectories considered are typically idealized by the Hohmann transfer, as showed in section 2.5. The minimum energy transfer is well defined by two position vectors, as shown by Prussing and Conway [7]. This minimum energy transfer, when evaluated at a 180-degree separation for coplanar positions, are the same as the Hohmann

transfer. This commonality offers an excellent recommendation for the initial starting point in interplanetary calculations as the minimum energy transfer represents a solution that is close to the ideal interplanetary transfer, and therefore close to the solutions of interest.

$$z_{start} = \left(\pi - 2\sin^{-1}\sqrt{1 - \frac{2c}{r_1 + r_2 + c}} \right)^2 \qquad 3.10$$

$$\text{where,} \quad c = \sqrt{r_1^2 + r_2^2 - 2r_2 r_1 \cos\Delta v}$$

The variables r, r_o, and $\Delta\nu$ are the two position magnitudes and the angle between the two positions. By implementing the new starting position, the numbers of iterations are reduced for the 200 to 300 day transfer time solutions, as seen in Figure 3.7.

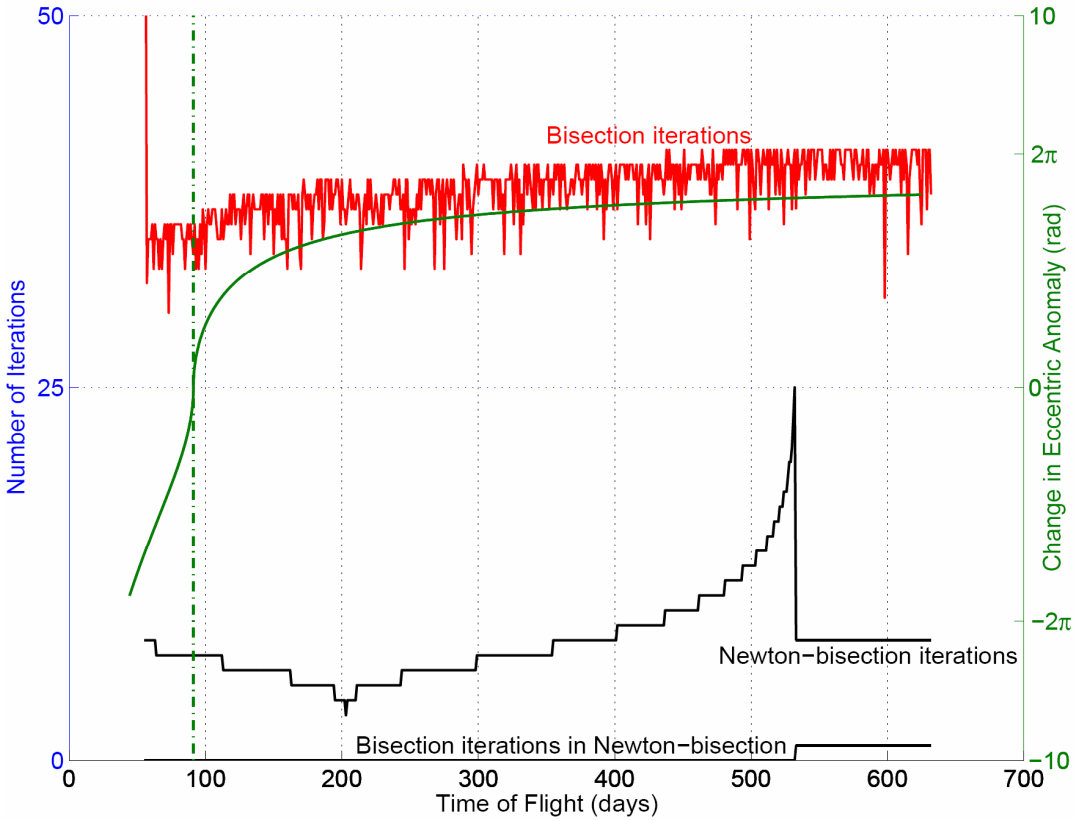

Figure 3.7: Lambert Solutions Incorporating Minimum-Energy

By reducing the number of iterations for solutions in the regions of common transfer times for Earth-Mars interplanetary trajectories, the overall computation time is reduced greatly. This will aid the overall performance of the algorithm when calculating thousands of interplanetary trajectories for mission analyses.

Now that the time of flight equation has been satisfied for the convergence and performance criterion, the interplanetary trajectory can be calculated. For this, the f and g expressions are used, as derived in Appendix A:

$$f = 1 - \frac{y}{r_1} \tag{9.22}$$

$$g = A\sqrt{\frac{y}{\mu}} \tag{9.23}$$

$$\dot{g} = 1 - \frac{y}{r_2} \tag{9.24}$$

Since the f and g functions define planar motion from a starting point, the velocity of the first and second points can be determined from:

$$r_2 = fr_1 + gv_1 \Rightarrow v_1 = \frac{r_2 - fr_1}{g} \tag{3.11}$$

$$v_2 = \dot{f}r_1 + \dot{g}v_1 \Rightarrow v_2 = \frac{\dot{g}r_2 - r_1}{g} \tag{3.12}$$

An important subtlety in equations 3.11 and 3.12 needs to be addressed when the transfer between the two points is 180 degrees, causing A in equation 9.17 to be zero. For a zero A, the g function in equation 9.23 becomes zero, causing equations 3.11 and 3.12 to be undefined for a 180-degree transfer. Therefore, the universal variables algorithm does not handle 180-degree transfers. This subtlety is acceptable though, as 180-degree transfers will not be considered by the interplanetary trajectories calculated.

From equations 9.22 through 9.24, 3.12 and 3.13, the interplanetary trajectory is determined. The following steps outline the universal variables Lambert solution:

1. Determine a starting point, z_o, using equation 3.09.
2. Determine z using the Newton iteration algorithm (3.08) combined with the bisection algorithm (3.09).
3. Determine A from equation 9.17.
4. Determine *y* from equation 9.18.
5. Determine the f, g, and $\dot{g}$ expressions using equations 9.22 through 9.24.
6. Determine the velocities v_1 and v_2 from equations 3.12 and 3.13.

The Lambert solution that has been derived here provides a robust algorithm for interplanetary calculations. The multiple revolution capability has been presented along with a Newton-bisection iterative method that provides efficient convergence. In the next section, the application of the Newton-bisection Lambert solution to targeting will be shown.

4 Interplanetary Trajectories

In the patched conic method, the assumption is made that the heliocentric conic is only influenced by the sun's gravity. Such interplanetary trajectories are not characteristic of a real spacecraft, but the overall error in doing such an approximation will be shown to be minimal.

In calculating the interplanetary trajectory, inputs of launch and arrival dates are first required. The final output of interest in the trajectory is the $\bar{v}_\infty$ vectors of the planetocentric hyperbolas in the IAU frame of date. The following interim steps are used in calculating the interplanetary trajectory:

1. A launch and arrival date is selected.
2. The planetary ephemeris of choice is called to retrieve the positions and velocities of the respective planets.
3. The coordinate frame of the planet position and velocity is rotated to the Earth mean ecliptic frame.
4. The Lambert algorithm is used to obtain the f and g functions of the transfer trajectory using the planet position from the planetary ephemeris and difference in time between the launch and arrival dates.
5. The orbital elements are evaluated at the two known positions to yield the transfer velocity vector.
6. The transfer velocity vector is subtracted from the planet velocity vector to yield the $\bar{v}_\infty$ vectors of the planetocentric hyperbolas.

7. A final coordinate transformation is performed to rotate the $\bar{v}_\infty$ vectors of the planetocentric hyperbolas from the Earth mean ecliptic frame to the IAU frame of date.

4.1 Launch and Arrival Date Considerations

Assuming coplanar circular planetary orbits, the minimum Δv transfer between two planets is the Hohmann transfer. For real planetary orbits, the 180° Hohmann transfer requires the launch and arrival to occur on the line of intersection of the orbit planes of the two planets, which is not feasible. However, near 180° transfers are feasible due to the small angles between most of the planet orbit planes and the ecliptic plane, requiring only small plane changes out of the ecliptic by the interplanetary trajectory. A region of dates are therefore selected about the Hohmann transfer, allowing exploration of the trajectories in the vicinity of the minimum Δv.

To determine the times for a potential Hohmann transfer, some orbital geometry needs to be introduced. Since Earth and Mars are in elliptic orbits with low eccentricity and inclination, the orbits are assumed to be circular-coplanar, as shown by the solid (circular) and dotted (elliptical) lines in Figure 4.1. Therefore, the mean motion is the constant angular rate of the planet, $n = \sqrt{\frac{\mu}{a^3}}$

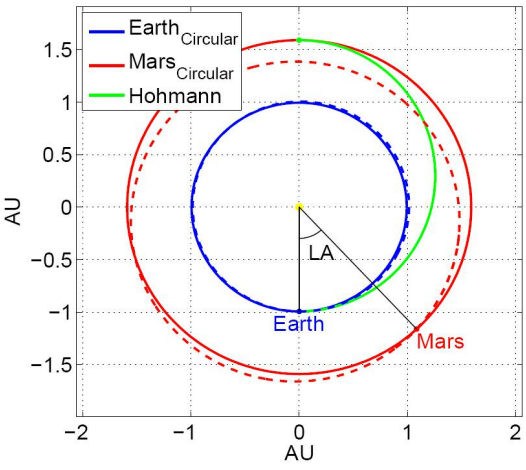

Figure 4.1: Lead Angle

where μ is the gravitational constant of the sun and a is the semi-major axis. Using the

orbital period, $P = \frac{2\pi}{n}$, and the geometry from Figure 4.1, an expression relating the lead angle (LA) required for a launch and arrival that is 180° apart can be derived as, $\pi = LA + n_2 T$, where T is the time of flight for the Hohmann transfer, or half the transfer orbit period and n_2 is the constant mean motion of Mars. Since only the semi-major axis of the transfer needs to be known for the time of flight of the Hohmann transfer, the orbital period can be represented as a function of the two semi-major axes, a_1 and a_2 of Earth and Mars. From the above equations, the expression for the lead angle is

$$LA = \pi\left(1 - \sqrt{\left(\frac{a_1 + a_2}{2a_2}\right)^3}\right) \qquad 4.01$$

For Earth and Mars, the lead angle is about 43°. To find an appropriate launch and arrival date, the planetary ephemeris needs to be called successively until Mars leads Earth by the lead angle. At this time, the Hohmann transfer is possible under the circular, coplanar assumption for Mars and Earth.

When a suitable launch and arrival date combination has been found, a synodic period may be added to the launch/arrival date combination so that the planets are aligned in the correct lead angle position for subsequent mission windows. The proper lead angle alignment period is moving at a rate equal to the difference in mean motion of the two planets, $n_{synodic} = \left(n_{Mars} - n_{Earth}\right)$. The resultant mean motion can be used to calculate the synodic period.

$$P_{synodic} = \frac{2\pi}{n_{synodic}} \qquad 4.02$$

Expected regimes that are possible for interplanetary trajectories can be explored by multiplying the synodic period by an integer number of future mission windows and adding to the time of the lead angle transfer alignment for the circular, coplanar Mars and Earth. The synodic period for Earth-Mars is about 26 months.

Although the launch and arrival dates have been predefined for the comparison case of this study, it is important to know how to predict future mission windows. The process of selecting a mission window using the described method above has been validated versus the 2009-2010 dates in the JPL comparison data. Note that the method does not actually predict a mission window, but rather a point in time from which the mission window is selected. With the method developed for mission window time selection, an example case will be shown for a 2013-2014 Earth-Mars mission window in chapter 7. This completes step 1 in the interplanetary trajectory calculation, selecting a launch and arrival date. The next step is calculating the planetary positions in time from the planetary ephemeris.

4.2 Ephemerides

There are numerous types of ephemerides that may be used in retrieving planetary positions. High precision ephemerides will use Einstein's theory of relativity as the underlying premise in calculating planetary positions. Many ephemerides are tabulated, while others have been fit to a curve, or are calculated using specialized formulae. Currently, NASA's Jet Propulsion Laboratory maintains the most accurate ephemerides.

The simplest ephemeris used is a least squares fit of the JPL DE200. The ephemeris is given in orbital elements and includes a constant centennial rate for each element. The ephemeris is stated to be accurate to within 25 arc-seconds over the interval

of 1800 AD to 2050 AD. The ephemeris' coordinate system is in the Earth mean ecliptic system of J2000. This ephemeris offers great flexibility and use in the astrodynamic community due to its simple model and reasonable precision. The DE200 least squares fit ephemeris can be downloaded from the JPL website at http://ssd.jpl.nasa.gov/txt/p_elem_t1.txt.

A second ephemeris is Van Flandern's Low Precision Formulae for Planetary Positions [9]. This ephemeris was generated for such applications as automatic telescope pointing, tidal theory, and planetarium projector settings, where overall accuracy is not as important as computational time. The planets are calculated using a set of trigonometric series that include the mean longitude, mean anomaly, argument of latitude, and their time derivatives. The data tables for these series have been tabulated which reduces computation time greatly. The overall accuracy of Van Flandern has been shown to be within one arc-minute of the actual planetary position over a range of ± 300 years from the year 1978 [9]. All coordinates are given with respect to the Earth mean equator system.

The final ephemeris is the DE405 planetary ephemeris from JPL. At the time of writing, DE405 is the most accurate ephemeris that is freely available to the public. JPL does generate other ephemerides for mission specific needs and long-term planetary estimation, but this is the best ephemeris for general-purpose applications making it ideal for the purposes of this research. The equations of motion for DE405 were integrated forward from 1600 AD to 2200 AD offering a generous range of data to sample from. The ephemeris is tabulated using Chebyshev polynomials and coefficients to interpolate the planetary positions. All coordinates are given with respect to the International

Celestial Reference frame (ICRF). The ICRF frame only differs from the Earth mean equator of J2000 system by 78 ± 10 micro arc seconds [20]. Therefore, the position and velocity from the DE405 ephemeris is assumed to be in the Earth mean equator system of J2000. The DE405 ephemeris may be downloaded from the anonymous ftp server: ftp://ssd.jpl.nasa.gov/pub/eph/export/DE405.

For consistency, the coordinates of each ephemeris are transformed to the Earth mean ecliptic system of J2000. To perform the necessary conversion, the following rotation matrix is used.

$$R_{EME/ECL} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varepsilon & -\sin\varepsilon \\ 0 & \sin\varepsilon & \cos\varepsilon \end{bmatrix} \qquad 4.03$$

The variable ε is the obliquity of the Earth equator and ecliptic. The obliquity is 23^0 26' 21.448" at J2000 [3].

The ephemerides described above will be compared to each other when calculating step 2 of the interplanetary trajectory, the determination of the position and velocity of the planets. The transformation to the ecliptic given by equation 4.03 completes step 3 in the interplanetary trajectory calculation, rotating from the EME system to the Earth mean ecliptic frame.

4.3 Lambert Solution

As described in chapter 3, the Lambert solution is utilized to solve the interplanetary transfer from planet to planet. Interplanetary transfer terminology needs to address three main points: multiple revolution orbits, orbit geometry and the angle traveled between the two points. These main points distinguish 4 separate transfers, Type I, Type II, Type III and Type IV transfers.

As explained in reference [10], a Type I trajectory has the true anomaly at arrival bounded by 0° to 180°. A Type II trajectory has the true anomaly at arrival bounded by 180° to 360°. If the transfer takes multiple revolutions about the central attracting body the trajectory is called a Type III and Type IV trajectory. Type III, multiple revolution trajectories have the true anomaly at arrival bounded by 0° to 180°. Type IV, multiple revolution trajectories have the true anomaly at arrival bounded by 180° to 360°. Typically, the multiple revolution transfer trajectories are denoted with a "–" and "+" to designate the left and right side approach in the Lambert solution. For example, a highly elliptic, multiple revolution transfer with true anomaly at arrival less than 180° is called a Type –III. It should be noted that these terms are not uniformly accepted definitions. However, for purposes in this report, the definitions presented will be used.

The Lambert solution developed in chapter 3 is valid for all types of transfers. From equations 3.12 and 3.13, the appropriate velocity vectors of the interplanetary transfer are determined. This completes steps 4 and 5 of the interplanetary trajectory calculation, the Lambert solution and launch and arrival velocity calculation.

4.4 Calculation of the Interplanetary Trajectory

The inputs generated so far have been the launch and arrival dates, and the planetary position and velocity at those times. The Lambert solution was utilized to determine the transfer between the two planet positions. As stated by step 6, the transfer velocity vectors are subtracted from the planetary velocity vector to yield the direction and magnitude of $\vec{v}_\infty$.

$$\vec{v}_\infty = \vec{v}_{lambert} - \vec{v}_{planet} \qquad 4.04$$

The result of equation 4.04 will later be shown to be the $\hat{S}$ axis of the B-plane coordinate system. Figure 4.2 depicts the resultant $\vec{v}_\infty$ vector that is in the Earth mean ecliptic coordinate system.

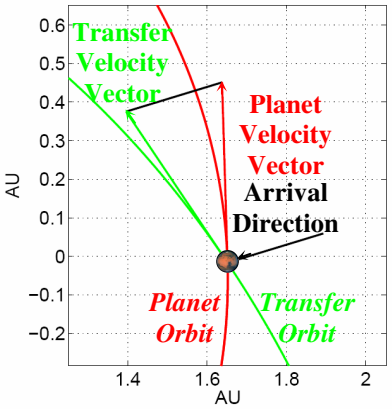

Figure 4.2: Arrival Geometry

The final phase in calculating the interplanetary trajectory, step 7, is rotating the $\vec{v}_\infty$ vector to the planet IAU coordinate frame of date. The transformation to the IAU coordinate frame is first accomplished by rotating the Earth mean ecliptic coordinate system to the Earth mean equator system of J2000 by using the transpose of the rotation matrix in equation 4.03. Then, by using the physical ephemeris of date for the planet, the rotation matrix in equation 2.06 can be utilized to rotate the $\vec{v}_\infty$ vector from the Earth mean equator system of J2000 to the IAU coordinate system of date. This transformation is done for both the launch and arrival $\vec{v}_\infty$ vectors.

The interplanetary transfer trajectory is depicted in Figure 4.3 for a launch date of November 1st 2011, and arrival date of September 1st 2012. The steps given at the beginning of chapter 4 have been included for reference. The x and y axes are in the plane of the ecliptic of J2000 in units of kilometers. The blue curve is the orbit of the Earth, the red curve is the orbit of Mars and the green curve is the transfer trajectory with the dashed magenta lines connecting the location of Earth and Mars at the given points in time.

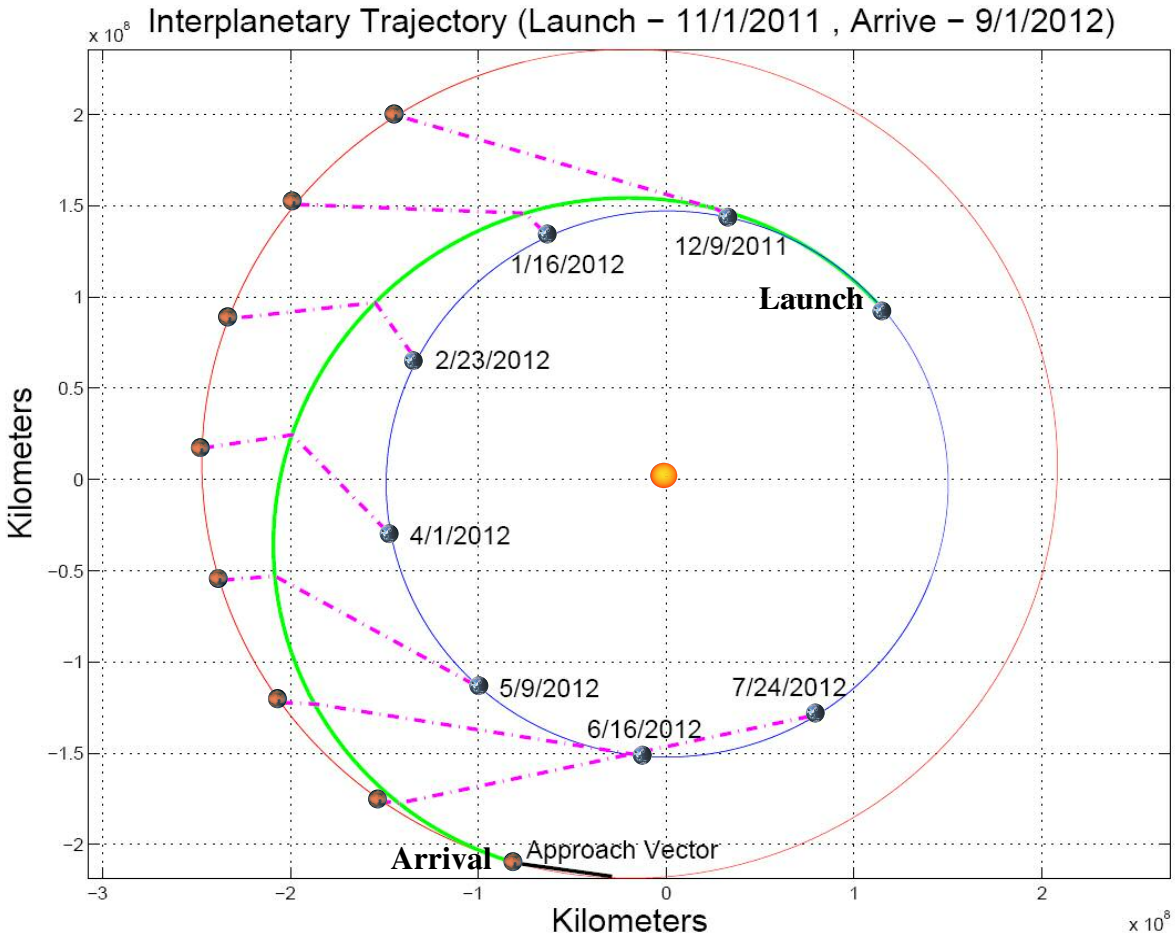

Figure 4.3: Ballistic Trajectory

Following the steps outlined in this chapter, the interplanetary trajectory may be calculated between two planets. From just the transfer velocities at launch and arrival, the v_∞ may be determined for the selected window of launch and arrival dates. The

launch v_∞ is commonly referenced by C_3, which is the velocity at infinity squared (equation 4.04)[2]. By plotting the necessary launch C_3 and arrival v_∞, the necessary transfer velocities can be explored for the mission window. Such plots are commonly called porkchop plots because of their shape. A porkchop plot for launch C_3 and arrival v_∞ for a 2009-2010 Earth-Mars mission window is shown in Figure 4.4.

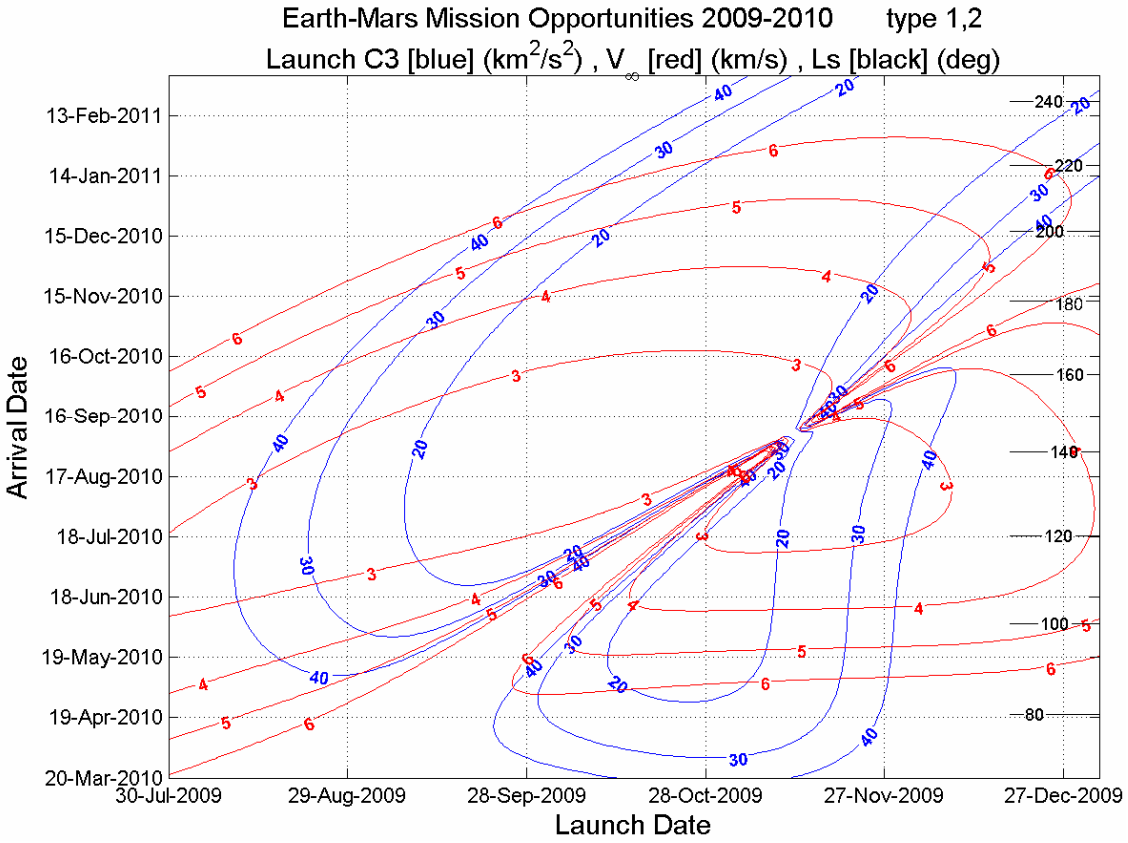

Figure 4.4: Porkchop Plot

The next goal is to couple the interplanetary trajectory with the EDL analysis. To do this, the patched conic approximation is used to link the EDL analysis to the interplanetary analysis. Although patching the two trajectories will inherently introduce errors, the end result is acceptable for mission design purposes, and will be shown to correlate well with the JPL data.

5 Launch and Arrival Space Synthesis

The window of available launch and arrival dates about the optimum time of launch will affect the shape of the interplanetary trajectory that can in turn affect the EDL analysis. The planetocentric hyperbola that is defined by the patched conic approximation will be used to link the EDL analysis to the interplanetary trajectory. The synthesis of closed form, analytical solutions to the planetocentric hyperbola from the interplanetary trajectory are therefore useful when targeting to specified constraints.

5.1 Launch Space

The launch space is critical to mission designers for two reasons. The first reason is the launch energy required to get from planet to planet. Second is the launch vector orientation with respect to the Earth tracking stations that will determine the trajectory of the departing spacecraft. The derivation of the launch energy and orientation will be presented and showed for application to interplanetary trajectories.

Launch vehicle companies such as Lockheed Martin and Boeing define the capability of their launch vehicle by the launch energy (C_3) versus the amount of payload. For the impulsive Δv required by the launch vehicle from a parking orbit, the energy integral can be used to calculate the velocity requirement at any radius from the planet.

$$v^2 = v_\infty^2 + \frac{2\mu}{r} \qquad 5.01$$

This is the velocity the spacecraft must attain at the specified radius, r. By knowing a specified circular parking orbit radius, the tangential Δv to the orbit may be calculated by subtracting the velocity of the parking orbit from the velocity calculated in equation 5.01.

The orbital geometry must also be considered by the launch space. The declination of the launch asymptote, or DLA, is very important to mission navigators. For example, an initial orbit determination of the interplanetary trajectory depends strongly on DLA. This is because of the location of tracking stations on Earth having the ability to *accurately* track spacecraft within a specific band of DLA. A DLA that falls outside of the band could result in a poor orbital determination by the various tracking stations causing misalignment in the interplanetary trajectory injection.

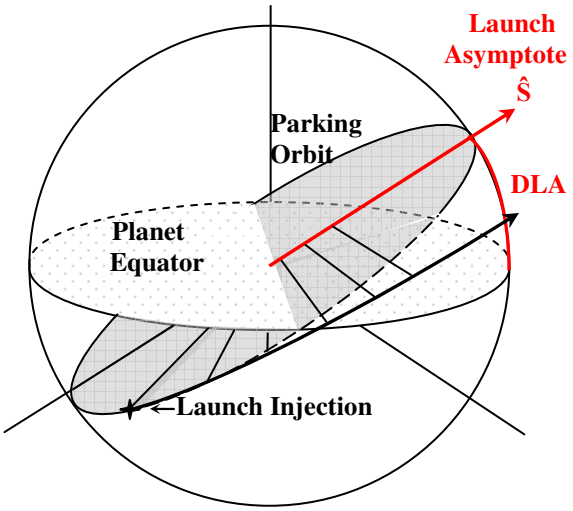

Figure 5.1: Launch Asymptote Declination

DLA (δ) can be computed from the direction of the launch asymptote, Ŝ, which is the unit vector of the velocity at infinity in the IAU coordinate frame, as determined from equation 4.04.

$$\delta = \sin^{-1}\left(\frac{\hat{S}_z}{\sqrt{\hat{S}_x^2 + \hat{S}_y^2 + \hat{S}_z^2}}\right) \qquad 5.02$$

5.2 Arrival Space Synthesis

The arrival space is another critical regime. Orbit inclination, latitude, argument of periapse, declination, time of day, and planet season are all parameters of interest. Each parameter is in part responsible for the performance of the EDL system. Therefore,

the calculation of each parameter is necessary in developing a linked interplanetary and EDL analysis.

5.2.1 B-plane

The coordinate system that is of primary concern in the arrival space synthesis is the B-plane. B-plane coordinates are very useful in targeting scenarios, and are used here to determine injection properties at Mars. Targeting in the B-plane system can be done from only three input variables: the magnitude of the $\vec{v}_\infty$ vector (equation 4.04), right ascension of the approach asymptote and declination of the approach asymptote (equation 5.02). The right ascension of the approach asymptote is calculated as:

$$\alpha = \text{atan2}\left(\hat{S}_y, \hat{S}_x\right) \qquad 5.03$$

Note that the right ascension and declination are both calculated with respect to the IAU coordinate frame of date. The B-plane coordinate system relation to targeting is shown in Figure 5.2.

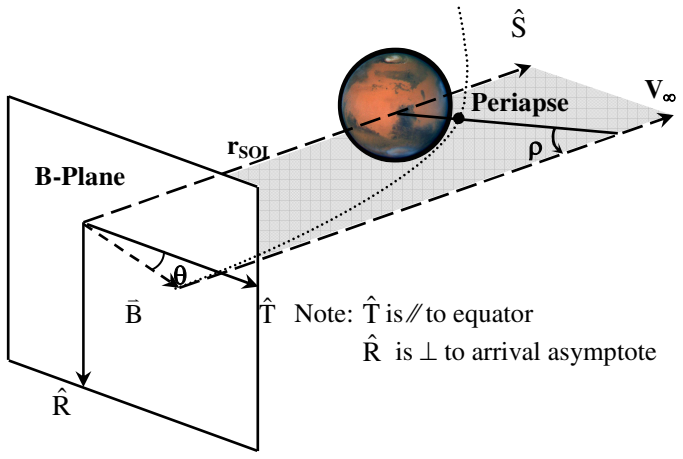

Figure 5.2: B-Plane Targeting System

The variables B and θ are used to target the approach hyperbola. B represents the magnitude of the miss distance with respect to the center of Mars perpendicular to the

arrival asymptote. The variable ρ is the turn angle of the approach hyperbola, $\rho = \cos^{-1}\frac{1}{e}$. The variable θ is the B-plane angle, measured clockwise in Figure 5.2.

Note by rearranging equation 2.10 to

$$\cos i = \cos\theta\cos\delta$$

inclination will have a minimum bound equal to the asymptote declination. Figure 5.3 shows the achievable inclination with respect to the declination of the approach asymptote (x-axis) and B-plane angle (y-axis). For a B-plane angle of -90° or 90°, the approach hyperbola will pass over the North or South Pole of Mars as long as the two-body assumptions are maintained. The benefits of B-plane targeting can be seen from the simplicity of equation 5.04, as orbit inclination and B-plane angle are easily related.

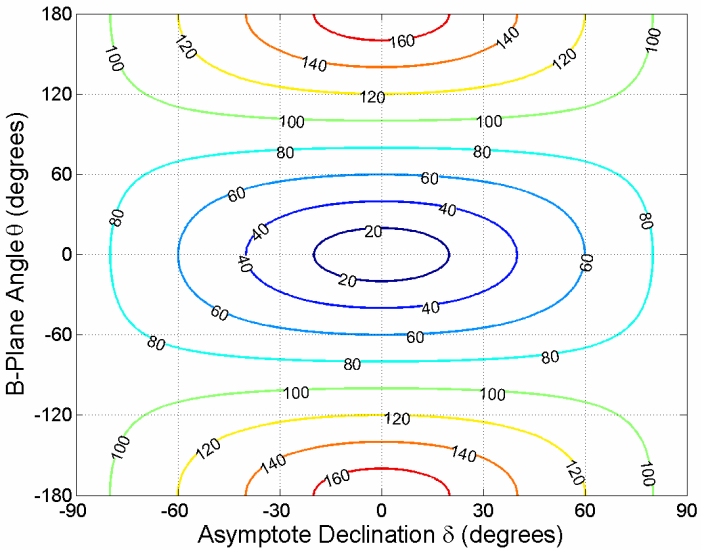

Figure 5.3: Inclination for Varying θ and δ

The interplanetary trajectory has been patched to the B-plane coordinates, a targeting method can begin to be developed. The next section will detail how B-plane targeting is employed in determining the maximum latitude capability.

5.2.2 Maximum Latitudes

The variables used to determine the maximum achievable latitudes, λ_{North} and λ_{South}, are the declination of the launch asymptote (δ) and the spherical side subtending the point of impact to the point where the arrival asymptote intersects the sphere of

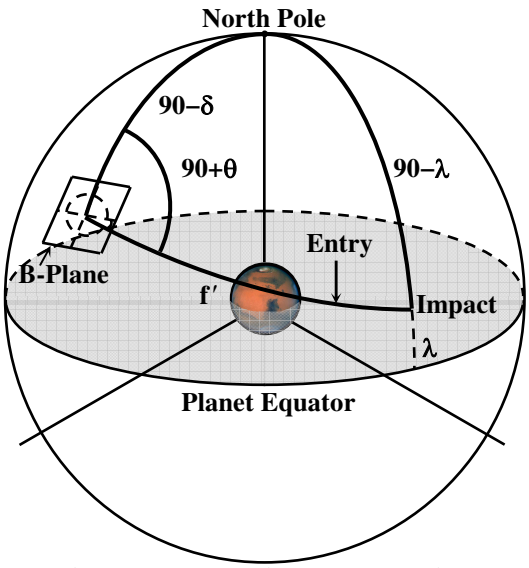

Figure 5.4: Latitude Targeting

influence at arrival (f´). From Figure 5.4 and spherical trigonometry, the equation for latitude at impact is:

$$\sin\lambda = \sin\delta\cos f' + \cos\delta\sin f'\cos\left(\theta + 90^\circ\right) \qquad 5.05$$

The maximum latitudes, assuming constant f´ and δ, occurs when B-plane angle θ is $\pm 90^\circ$, i.e. when the approach trajectory passes over one of the poles. Thus, the expressions for the maximum achievable northern and southern latitudes are given by:

$$\begin{aligned} \sin\lambda_{North} &= \sin\left(\delta + f'\right) \\ \sin\lambda_{South} &= \sin\left(\delta - f'\right) \end{aligned} \qquad 5.06$$

The next step is determining the variable f´, which consists of three components, the angle (ψ) subtended by B and the arrival asymptote, change in true anomaly from SOI to entry (Δf) and the central angle from entry to impact (ξ), all shown in Figure

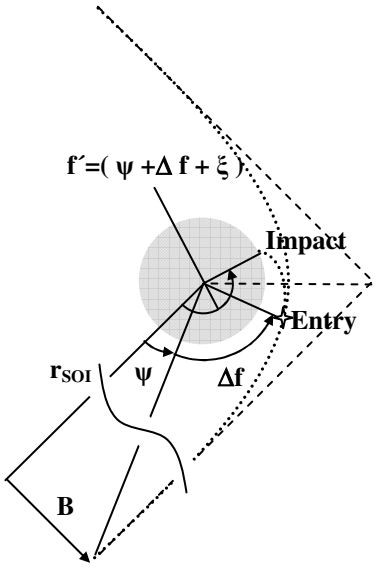

Figure 5.5: Spherical Side f´

5.5. The central angle variable ξ is a user defined first guess that is to be refined by the entry, descent and landing analysis. The variable ψ represents the angle between the approach asymptote and point of entry on the sphere of influence as measured from the center of the target planet. To calculate this angle, plane trigonometry will be used as opposed to spherical. Therefore, the radius of the sphere of influence and the magnitude of B are needed.

The radius of the sphere of influence is calculated from equation 2.01. To calculate B, the angular momentum of the arrival hyperbola must be defined according to the targeting parameters, target radius (r_{target}) and flight path angle (γ).

$$h_{target} = r_{target} v_{target} \cos \gamma_{target} \tag{5.08}$$

Typically, the target radius is defined at atmospheric entry, as this is the location that the EDL analysis will start. For Mars, atmospheric entry is defined at 3,522.2 kilometers. The flight path angle is initially a first guess that is to be later refined by the EDL analysis.

The approach hyperbola angular momentum can also be defined by the position at entry on the B-plane and the $\bar{v}_\infty$ vector. As explained in section 2.4.4, the angular momentum is $h = Bv_\infty$. By rearranging, the variable B is

$$B = \frac{h_{target}}{v_\infty} \tag{5.09}$$

By using the radius of the sphere of influence and B, ψ is determined from Figure 5.5 as,

$$\sin \psi = \frac{B}{r_{SOI}} \tag{5.10}$$

Now that ψ is determined, the only variable remaining is the true anomaly (f) from entry on the sphere of influence to atmospheric entry. True anomaly is calculated from the hyperbolic eccentricity, semi-latus rectum, semi-major axis and eccentric anomaly. The semi-latus rectum is $p = \frac{h^2}{\mu}$. The semi-major axis comes from the vis-viva integral evaluated at the target radius conditions, $a = \mu \Big/ \left(\frac{2\mu}{r_{target}} - v_{target}^2 \right)$. Finally, eccentricity is $e = \sqrt{1 - p/a}$. The true anomaly from the sphere of influence to atmospheric entry is defined by the hyperbolic eccentric anomaly.

$$F = \cosh^{-1}\left(\frac{1 - r/a}{e} \right) \qquad \begin{matrix} -\infty < F < 0 & \text{For Inbound} \\ 0 < F < \infty & \text{For Outbound} \end{matrix} \tag{5.11}$$

The sign of the hyperbolic eccentric anomaly at the sphere of influence will always be negative because dr/dt is assumed to always be less than zero. From the hyperbolic eccentric anomaly, true anomaly can be calculated.

$$f = \tan^{-1}\left(\frac{\sinh F \sqrt{e^2 - 1}}{e - \cosh F} \right) \tag{5.12}$$

Equation 5.12 will be used to calculate the change in true anomaly from the sphere of influence encounter to atmospheric entry.

$$\Delta f = f_{entry} - f_{SOI} \tag{5.13}$$

Equation 5.06 can now be used to solve for the extremes of latitude for various arrival conditions and targeting parameters. To be accurate when calculating the maximum latitudes, an oblate planet model should be used. The IAU defines the oblateness of the planets by their flattening, a parameter that specifies the degree by

which a planet's figure differs from that of a sphere. The IAU definition is known as the reference ellipsoid. An iterative approach is used when calculating the maximum latitudes on the reference ellipsoid. First, the problem is defined by using the equation of an ellipse to represent the ellipsoid in 2-D coordinates:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

From Figure 5.4, the variable y for the ellipse is $x\tan\lambda$ and the semi-minor axis, b, is $a(1-f)$. Substituting both into the equation for the ellipse, and solving for x yields,

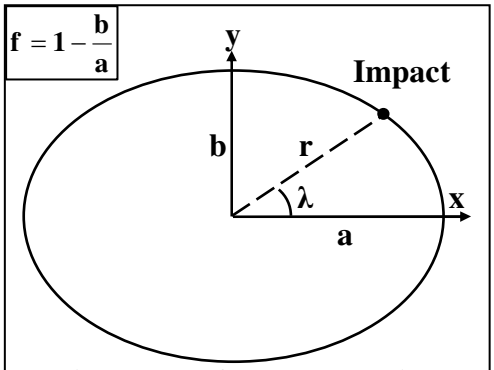

Figure 5.6: Oblate Modeling

$$x = a\left(1 + \frac{\tan^2\lambda}{(1-f)^2}\right)^{-\frac{1}{2}}$$

The radius of the ellipse is $r = x/\cos\lambda$. By substituting x into the equation for r, the equation for the radius of an oblate planet is determined as a function of the equatorial radius (a), latitude (λ) and flattening (f). .

$$r = a\left(\cos^2\lambda + \frac{\sin^2\lambda}{(1-f)^2}\right)^{-\frac{1}{2}} \qquad 5.14$$

Iteration is now used because the impact radius that has been redefined by the oblate model will yield a different angle, f´. The iteration then proceeds as follows:

1. Select a target entry radius, flight path angle, and central angle ψ.

2. Determine the angle f´ using equations 5.07 through 5.13.

3. Determine the latitude λ by equation 5.05.

4. Determine the oblate radius by equation 5.14.

5. Check the difference between the previous impact radius, and the oblate radius. For the first iteration, use an impact radius equal to the equatorial radius. For subsequent iterations, reset impact radius to the calculated oblate radius. When the difference falls below a specified tolerance, the correct impact radius and latitude has converged for the oblate model.

In mission planning, determining the maximum achievable latitudes is an important part of the interplanetary analysis. For example, if a 2009 Earth to Mars mission targets a latitude of 60N, only a certain set of launch and arrival combinations will be viable solutions, as shown by the shaded blue regions in Figure 5.7. The red

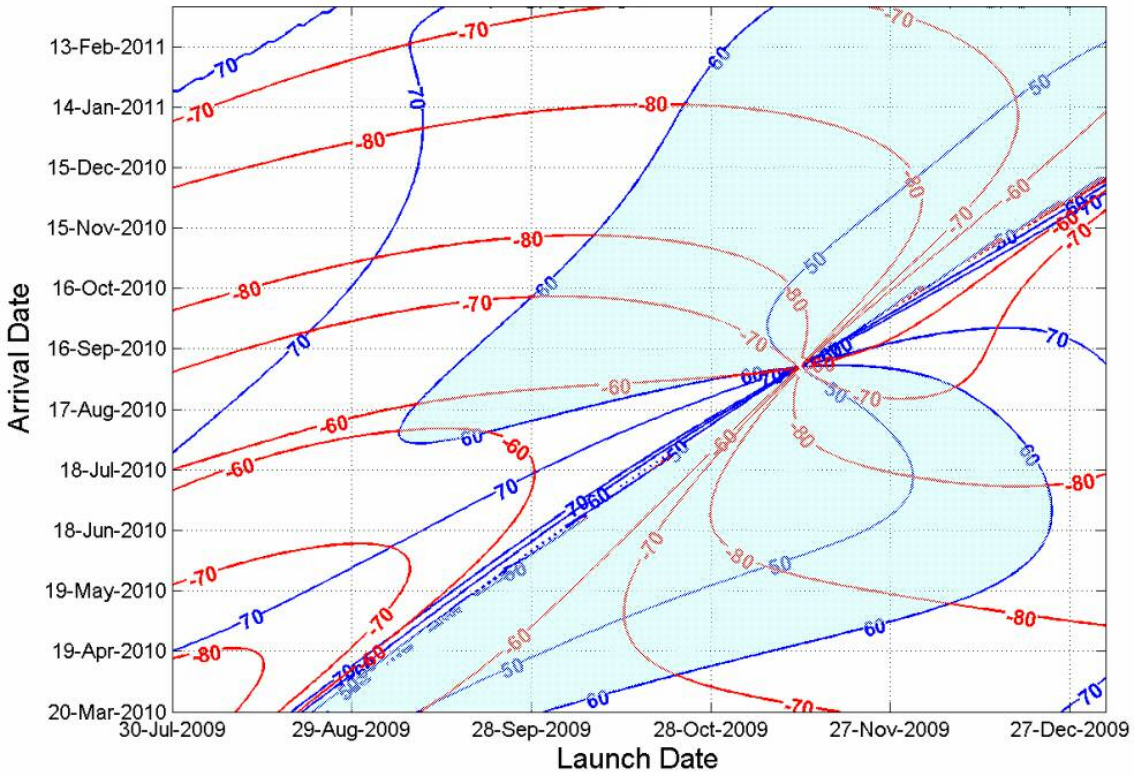

Figure 5.7: Achievable Latitudes for 2009'-10' Earth-Mars Missions

contours show the maximum southern latitude. Note that Figure 5.7 assumes a central angle from entry to impact of 12.6 degrees, meaning that the shaded blue region may expand or contract if the central angle assumed for the EDL system is either increased or

decreased from 12.6 degrees. This type of insight allows a preliminary view, before any EDL analysis is required, of what capabilities the mission may have for the given launch and arrival dates and EDL system.

5.2.3 Targeting Latitudes

Targeting to specified latitudes can be accomplished in the same way as done for the maximum latitudes, but will now include the EDL analysis using POST 2 [21]. The POST 2 analysis will be shown in chapter 7. The POST 2 analysis begins at atmospheric interface, thus defining the target radius. The variables flight path angle, B-plane angle and B magnitude will be used to target the desired latitude.

Since flight path angle is initially guessed, B-plane angle and B are the only inputs remaining to target to the defined requirements. By rearranging equation 5.05, the solution for B-plane angle (θ) as a function of target latitude becomes:

$$\theta = \cos^{-1}\left(\frac{\sin\lambda - \sin\delta\cos f'}{\cos\delta\sin f'}\right) - 90^{\circ} \qquad 5.15$$

$$\lambda_{South} < \lambda < \lambda_{North}$$

Equation 5.15 is only valid for the maximum target latitudes determined by equation 5.05. The magnitude of B is calculated from equation 5.09. Although B-plane angle and B have been developed to target the landing site, position and velocity are still needed for comparison to the JPL data. The conversion from B-plane coordinates to position and velocity will be covered in the next section.

5.2.4 Arrival State

Now that the necessary targeting variables have been determined, the conversion from the targeting parameters to position and velocity must be addressed. The position and velocity unit vectors are first defined in the perifocal system of chapter 2.3.1 using the true anomaly of entry (f_{entry}), eccentricity and flight path angle. The unit vectors for position and velocity in the perifocal system are:

$$\hat{r}_{PQW} = \begin{Bmatrix} \cos f \\ \sin f \\ 0 \end{Bmatrix} \tag{5.16}$$

$$\hat{v}_{PQW} = \begin{Bmatrix} -\dfrac{\cos\gamma \sin f}{1+e\cos f} \\ \dfrac{\cos\gamma\left(e+\cos f\right)}{1+e\cos f} \\ 0 \end{Bmatrix} \tag{5.17}$$

The next step is to rotate the perifocal coordinate system to the B-plane coordinate system. This is performed by first rotating about the perifocal W-axis by the turn angle of the hyperbola as shown in Figure 5.8. This will place the P-axis along the approach asymptote, $\hat{S}$ in the B-plane coordinate system.

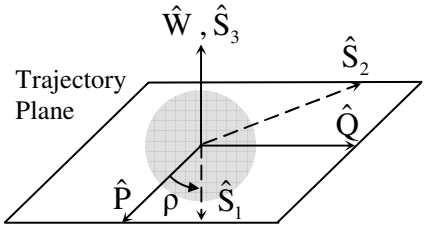

Figure 5.8: 1st Rotation

A rotation about the intermediate x-axis is then made by the negative of the B-plane angle θ to align the axes with the B-plane system and the equator of the planet, Figure 5.9. Note that the B-plane angle rotation does not place the axes of the coordinate system directly on the B-plane system, but only aligns the axes. The next rotation is about the y-axis by the declination of the approach asymptote (δ) to align the z-axis with the north pole of the planet equator and x-axis with the planet equator, shown in Figure 5.10. The notation for this intermediate rotation denotes the axes as the unit vectors b_1, b_2 and b_3. The final rotation is about the z-axis by the negative of the right ascension of the approach asymptote, shown in Figure 5.10. A vector undergoing these four rotations will be in the IAU coordinate system of date. These four rotations yield the following rotation matrix,

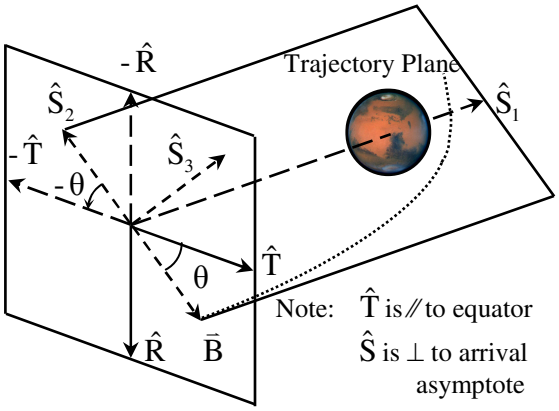

Figure 5.9: 2nd Rotation

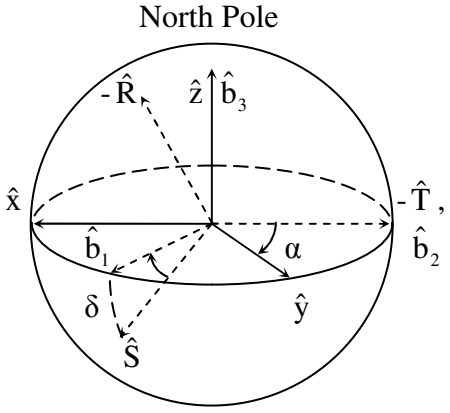

Figure 5.10: 3rd & 4th Rotation

$$
\begin{aligned}
R_{11} &= \sin\alpha\cos\theta\sin\rho + \cos\alpha\left(\cos\delta\cos\rho + \sin\delta\sin\theta\sin\rho\right) \\
R_{12} &= -\sin\alpha\cos\theta\cos\rho + \cos\alpha\left(\cos\delta\sin\rho - \sin\delta\sin\theta\cos\rho\right) \\
R_{13} &= -\cos\alpha\sin\delta\cos\theta + \sin\alpha\sin\theta \\
R_{21} &= -\cos\alpha\cos\theta\sin\rho + \sin\alpha\left(\cos\delta\cos\rho + \sin\delta\sin\theta\sin\rho\right) \\
R_{22} &= \cos\alpha\cos\theta\cos\rho + \sin\alpha\left(\cos\delta\sin\rho - \sin\delta\sin\theta\cos\rho\right) \\
R_{23} &= -\sin\alpha\sin\delta\cos\theta - \cos\alpha\sin\theta \\
R_{31} &= \sin\delta\cos\rho - \cos\delta\sin\theta\sin\rho \\
R_{32} &= \sin\delta\sin\rho + \cos\delta\sin\theta\cos\rho \\
R_{33} &= \cos\delta\cos\theta
\end{aligned}
\qquad 5.18
$$

Applying this rotation matrix to equations 5.16 and 5.17 will yield the proper position and velocity vectors in the IAU coordinate system of date.

The method for targeting the arrival latitude has been developed for position and velocity. Other variables that are of specific interest in arrival space targeting are the time of day and solar longitude. These variables are calculated by use of the positions of the planets around the sun from the planetary ephemeris.

Solar longitude (Ls) is defined as the celestial longitude of the planet measured from the planet vernal equinox of date. To determine Ls, first calculate the location of the north pole of the planet equator in the Earth mean ecliptic system of J2000. This is accomplished by rotating the north pole from the IAU coordinate system of date to the Earth mean ecliptic system. The rotation matrix operates on a unit z vector that represents the north pole of the planet.

$$\hat{u}_{equ} = R_{IAU/ECL}\vec{e}_z \qquad 5.19$$

Next, the inclination and node of the planet about the sun is determined using the position and velocity vector of the planet in the Earth mean ecliptic system.

$$\vec{h} = \vec{r} \times \vec{v}$$

$$\cos i = \frac{h_z}{h} \qquad 0 \le i \le \pi$$

$$\sin\Omega = \frac{h_x}{\sqrt{h_x^2 + h_y^2}} \qquad \cos\Omega = -\frac{h_y}{\sqrt{h_x^2 + h_y^2}} \qquad 0 \le \Omega < 2\pi$$

The unit vector representing the pole of the planet orbit is:

$$\hat{u}_{ecl} = \begin{Bmatrix} \sin\Omega \sin i \\ -\cos\Omega \sin i \\ \cos i \end{Bmatrix} \qquad 5.20$$

The cross product of unit vectors along the ecliptic north pole (equation 5.20) and the planet equator north pole (5.19) will yield the vernal equinox of date. From the vernal equinox, solar longitude is measured along the direction of rotation of the planet about the sun to the planet position vector in the Earth mean ecliptic system.

Just as the vernal equinox symbolizes spring on Earth, the solar longitude (Ls) of other planets represent the season. This is especially important on Mars, as history has shown the Mars atmosphere to be more active in the northern or southern hemispheres depending on the season.

The time of day is also important if a nighttime landing is not desired. Type I trajectories will arrive on the sun lit side of the planet and vice versa for Type II trajectories. Type III and IV trajectories can provide extremes on time of day if desired. Therefore, because of the advantage in time of day a Type II trajectory may be favored over a Type I trajectory.

To determine the local solar time, the position vector of the sun and position vector at the target and flight path angle are used. Both vectors are in the IAU coordinate system of date. The sun position vector is determined by using the rotation matrices from

equations 4.03 and 2.06 to rotate the negative of the planet position vector, as retrieved from the planetary ephemeris, from the Earth mean ecliptic system to the IAU coordinate system of date. The position vector at the targeted radius is calculated from equation 5.16 and rotated to the IAU coordinate system by equation 5.18. The longitude of the position vector subtracted from the longitude of the sun vector is the local solar time.

Now that the launch and arrival space characteristics have been defined, the entry, descent, and landing analysis may be performed. But first, the comparison with the JPL data needs to be considered to verify the method. The next section will use the methods described by chapters 4 and 5 to retrieve the results for the comparison data.

6 Results

By comparing the simplified interplanetary calculation results to the JPL data, accuracy will be seen to greatly depend on the accuracy of the ephemeris used. To show this, porkchop plot data sets have been generated for the JPL mission window using the method outlined thus far for 3 different ephemerides. There are two significant parts to this method, the first being outlined in chapter 4: the interplanetary trajectory. The second has been outlined in chapter 5: the patched conic, arrival space synthesis. The data sets representing the two portions of the method will be differenced with the JPL data set and the percent difference will be plotted for the mission window range. The launch dates covered by the JPL data are from July 30th, 2009 to December 27th, 2009. The arrival dates covered are from March 20th, 2010 to February 13th, 2011. The interplanetary trajectory error results will be shown first. The patched conic, arrival space error will be shown second.

The first step in validating the process described herein is to solve for the targeting conditions of the JPL data. This includes the target radius, flight path angle, latitude, and central angle. The JPL data was found to have a target radius of 3,522.2 kilometers, a flight path angle of –14.5 degrees, target latitude of 0 degrees, and a central angle of 12.6 degrees. Using these targeting variables, a common targeting scenario for the analysis of the launch and arrival dates may be considered.

A grid of 157 launch dates and 176 arrival dates were generated using the targeting variables listed above. Three variables were selected for comparison, the arrival v_∞, solar longitude and launch C_3, and plotted with respect to the launch and arrival dates, as shown in Figure 6.1.

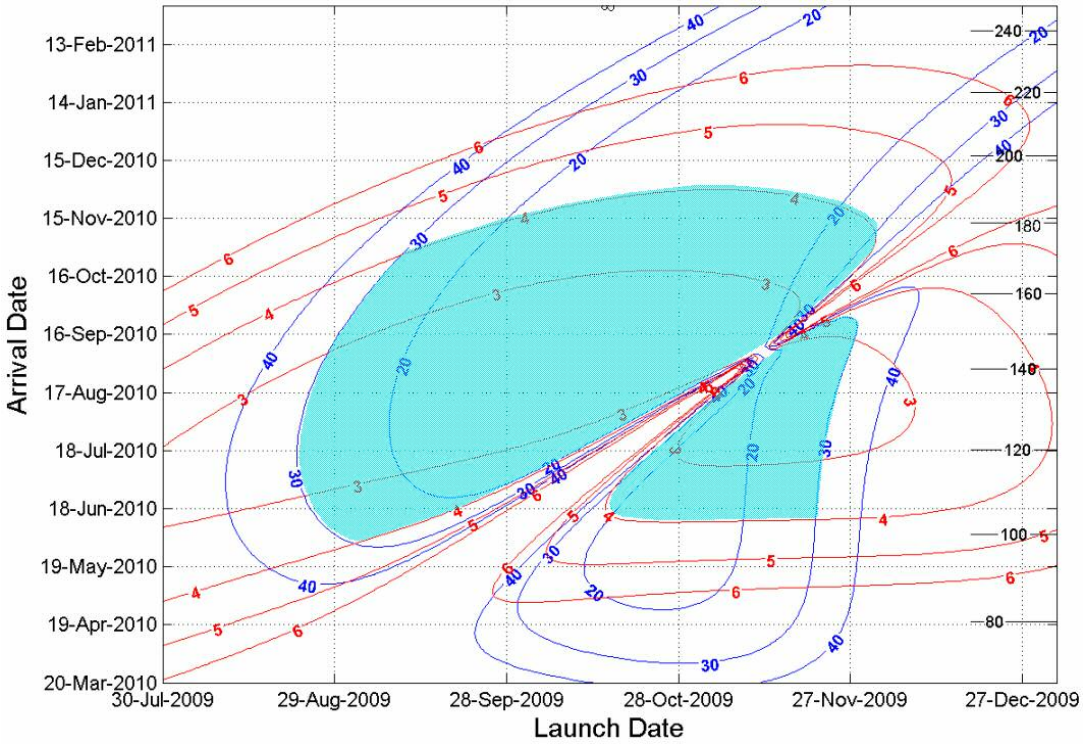

Figure 6.1: C3 and V_∞ Contours for Earth-Mars 2009-2010 Transfers

To understand the effects by the velocity at infinity, C_3 and solar longitude at arrival, a set of constraints were imposed on the mission window considered in Figure 6.1. Although lower velocities are desirable, the C_3 begins to grow where v_∞ is low. Since launch vehicles for MSL had a maximum C_3 around 30 km^2/s^2, a mission designer can bound the launch space by the contour curves of C_3 and the arrival space by the contour curves of v_∞. In Figure 6.1, the blue shading represents the bounded regions.

The reason for using arrival v_∞ is that it is a test of the accuracy of the planetary ephemeris used, as $\vec{v}_\infty$ is calculated directly from the Lambert solution that only uses the positions and times of the planets. The arrival position vectors are useful because they test the accuracy of the physical ephemeris, as the position vectors from JPL were given in the Mars prime meridian coordinate system, which is determined from the physical ephemeris.

6.1 Planetary Ephemeris Comparison

The planetary ephemeris used is directly correlated to the accuracy of the interplanetary trajectory calculated from the Lambert algorithm because it provides the position of the planets. The $\vec{v}_\infty$ magnitude is related by equation 4.04 to the interplanetary trajectory and will therefore be considered the variable of comparison to the JPL data. At the time of this research, it was unknown what planetary ephemeris JPL utilized to generate the data. The velocity at infinity for the JPL position and velocity data is calculated from equation 5.01. Using three different planetary ephemerides for the generation of the planetary positions will show the interplanetary trajectory accuracy. These ephemerides were presented in section 4.3.

The first ephemeris used is Van Flandern's Low Precision Formulae. This is considered to be the most inaccurate ephemeris of the three used, as accuracy is stated to only be reliable to one arc-minute. Figure 6.2 shows the percent difference in v_∞ for Van Flandern with respect to the JPL data. Since the difference is so small, the contour plots of v_∞ are all visually identical. Therefore, a contour plot of v_∞ difference is more illustrative for these purposes. The plot shows that v_∞ difference varies widely across the spectrum of dates, and is not biased towards any regimes. Even with such a low precision ephemeris, the v_∞ difference is nearly negligible from the JPL data and is acceptable for preliminary mission design. There is no advantage that the Van Flandern ephemeris holds over the other ephemerides. It is simply used to show that even for low precision ephemerides, interplanetary trajectories can still be calculated with relatively high accuracy.

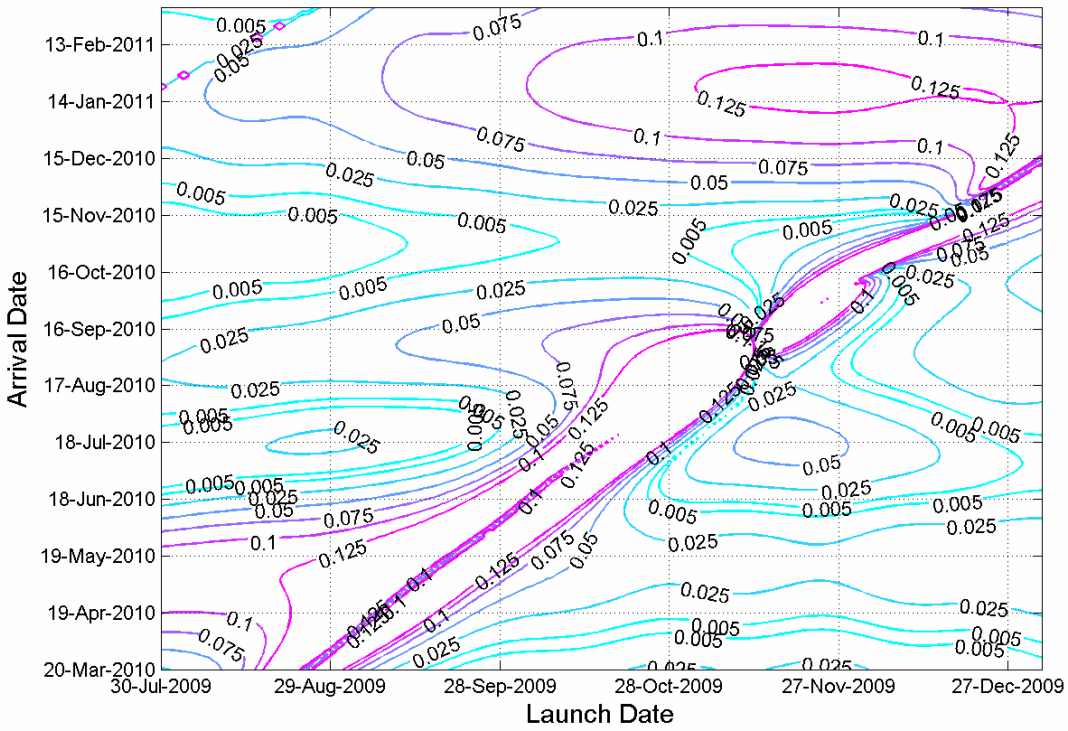

Figure 6.2: Van Flandern Difference in V_∞

The next ephemeris used is the DE200 least squares fit. The DE200 ephemeris is very fast to compute and has a precision more than twice that of Van Flandern. The percent difference in v_∞ is shown in Figure 6.3. A ridge of improved accuracy is seen to appear along the green line. The dispersion of v_∞ difference is smoother than Van Flandern, but is still large at the extremities.

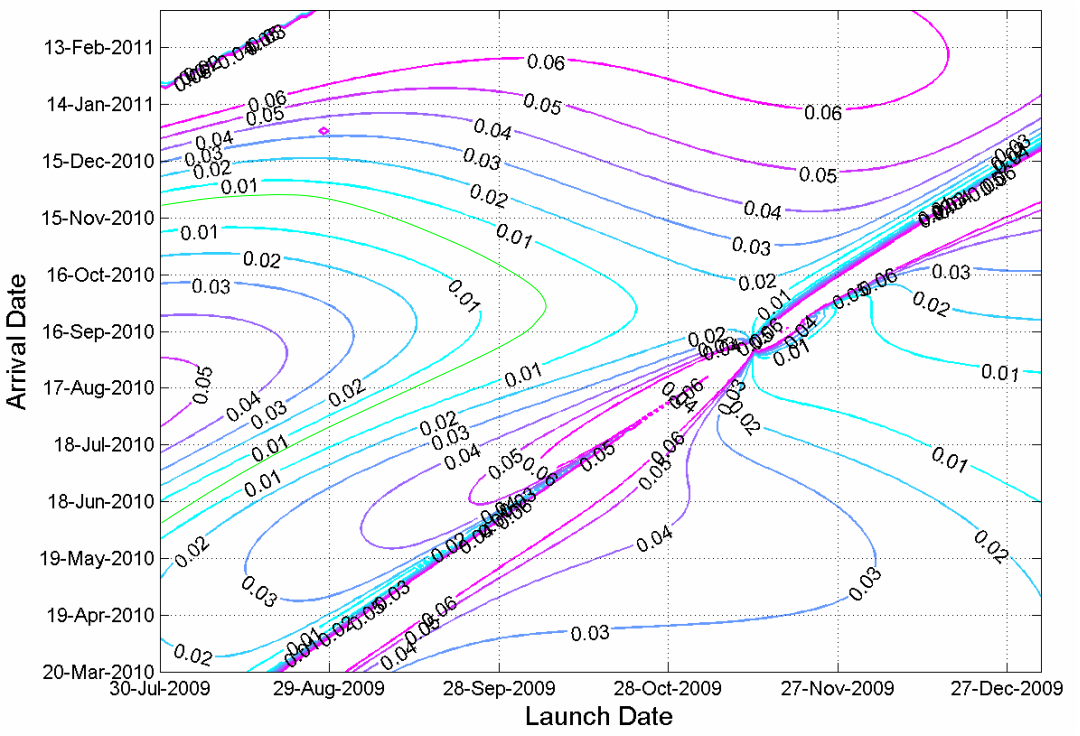

Figure 6.3: DE200 Difference in V_∞

The final ephemeris used is the DE405 ephemeris. JPL releases this ephemeris to the public for general use and is considered to be a very accurate prediction of planetary positions and velocities. DE405 is applicable from the years 1600 to 2200 AD. JPL has recently released DE410, which holds more accurate positions for Mars and Saturn due to the Mars Exploration Rover and Cassini missions respectively. The improved accuracy is not discernable though for this analysis. Figure 6.4 shows the percent difference in v_∞ for the DE405 ephemeris versus the JPL data. The dispersion of v_∞ difference is seen to vary periodically along the x-axis with a frequency of one month. The periodicity is likely due to the motion about the Earth-Moon barycenter is accounted for in the DE405. The Earth-Moon motion is less apparent in the other ephemerides as they are curve fits to the planetary positions and thus smooth out the period of motion seen by the Earth-Moon motion.

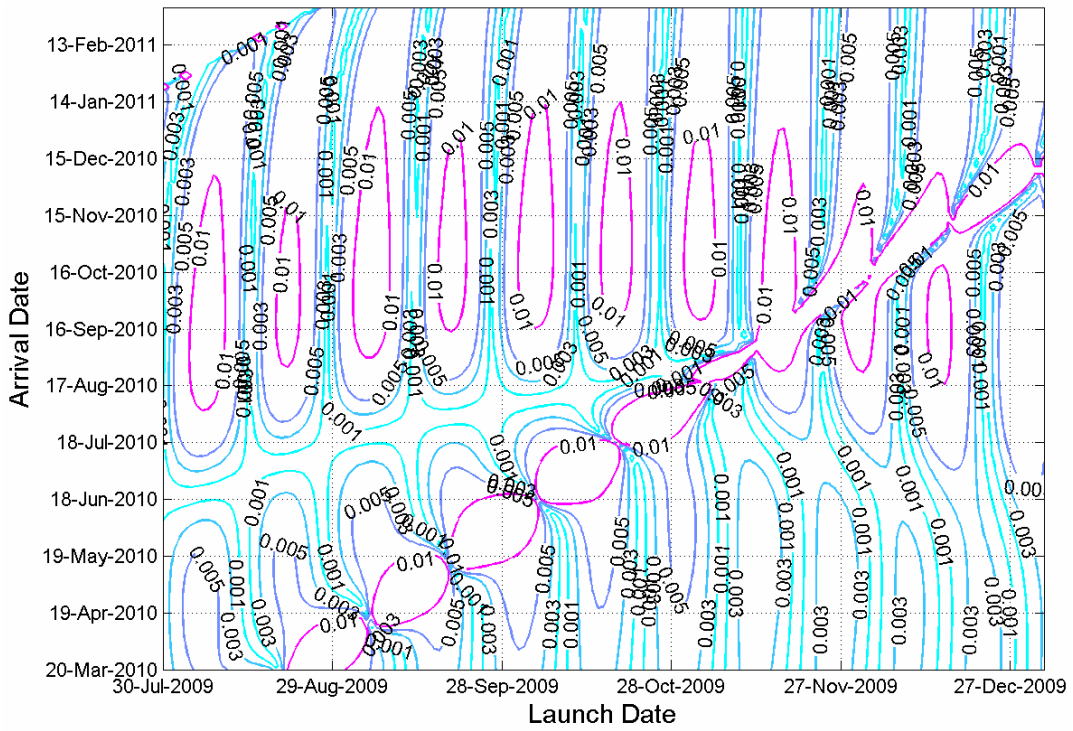

Figure 6.4: DE405 Difference in V_∞

The method used in the synthesis of the interplanetary trajectory from the planetary ephemeris has now been validated by the comparison of v_∞. This validates the assumptions used in the interplanetary calculation method. Furthermore, the results show that the selection of the ephemeris is not a large driver in retrieving accurate results. For a simplified interplanetary analysis, the DE200 ephemeris provides more than enough resolution of the planetary positions for determining the interplanetary trajectory.

The next step is to validate the position at arrival with respect to the JPL data. Remember that a patched conic method was used to retrieve the orbital elements about the planet and error will be inherent due to the patch. The next section will show the validity of the patch assumption along with the respective differences due to the planetary ephemeris used.

6.2 Position Comparison

To determine the accuracy of the targeted position, the method detailed in chapter 5 will be used to calculate position at arrival according to the targeting constraints of radius, central angle, flight path angle and latitude. For this analysis, the position vector from the JPL data is subtracted from the calculated position vector, and the magnitude of the difference is evaluated is plotted.

The position difference for the Van Flandern ephemeris is shown in Figure 6.5, with contours given in units of kilometers difference. Accuracy is seen to be limited in the shaded region of interest in Figure 6.1. Though accuracy is validated for the physical ephemeris as the position difference is relatively close to the form exhibited by Figure 6.2, implying that the inaccuracy is translated from the planetary ephemeris differences and not the position differences.

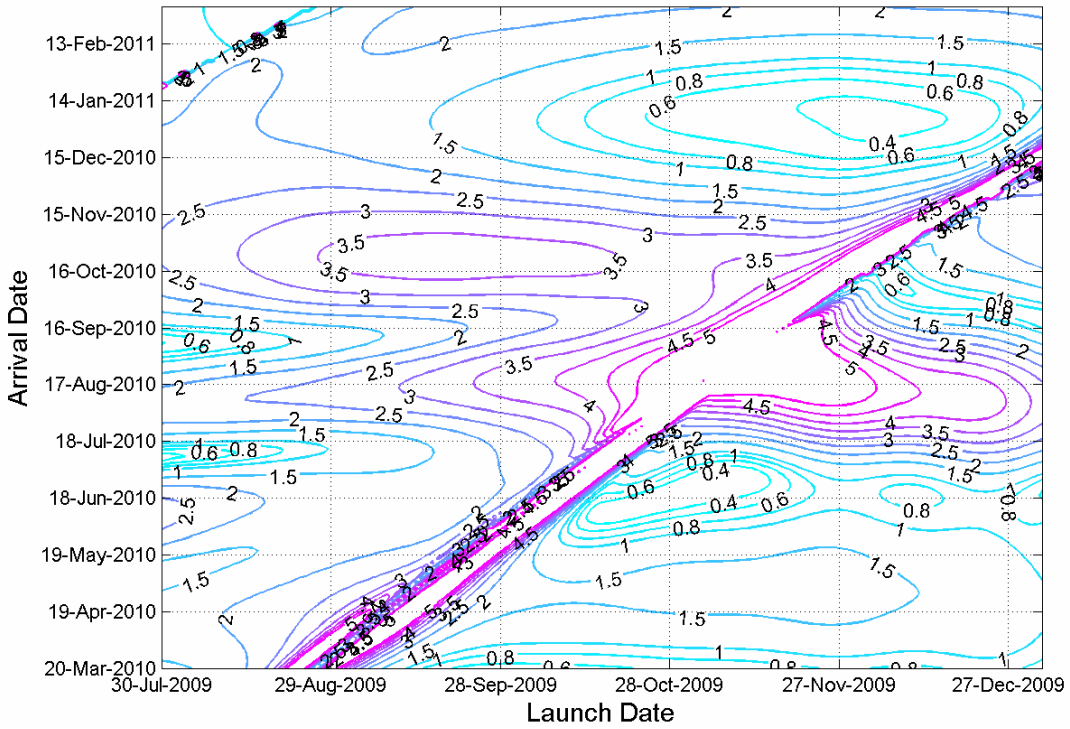

Figure 6.5: Van Flandern Position Difference

The DE200 ephemeris is evaluated next for position difference at arrival, Figure 6.6. Relative difference does not improve greatly over the differences encountered by Van Flandern. Although the DE200 ephemeris may have improved accuracy over the Van Flandern ephemeris for the interplanetary leg, target position difference is found to not increase as would be expected.

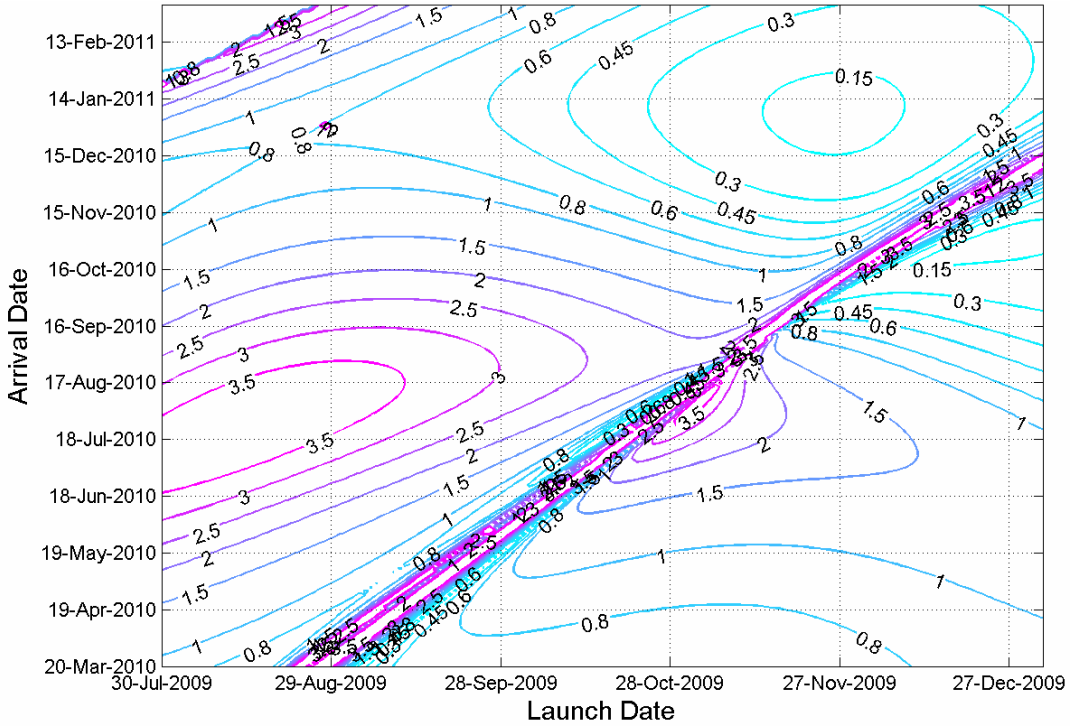

Figure 6.6: DE200 Position Difference

The position difference of the DE405 ephemeris is shown in Figure 6.7. The periodic variation is once again seen. Furthermore, since the variation remained periodic, the physical ephemeris used is also validated. If the physical ephemeris were inaccurate, contours would not have been similar to the periodic contours of the v_∞ difference seen in Figure 6.4. Position difference is also reduced considerably from the DE200 and Van Flandern ephemeris.

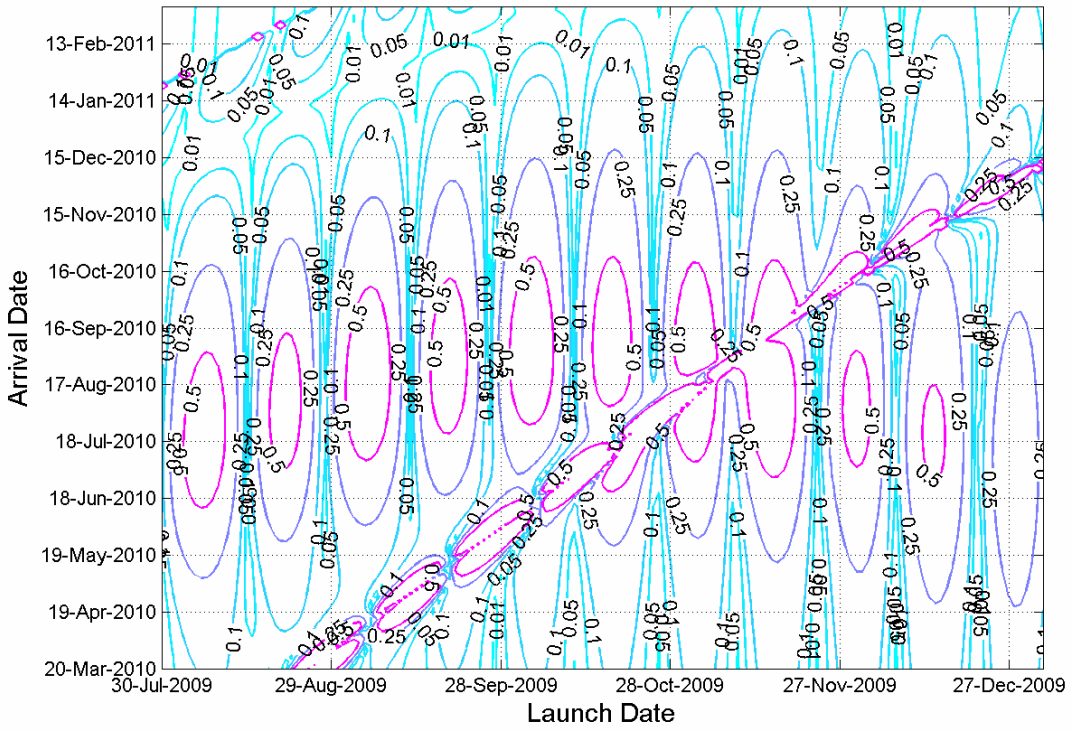

Figure 6.7: DE405 Position Difference

7 Linking the Entry, Descent, and Landing Analysis to the Interplanetary Calculations

The analysis performed up to this point has all been based upon the assumption of a constant target flight path angle and central angle. To refine these parameters, an entry, descent, and landing analysis is performed. The software generated up to this point is excellent for general cases, as it provides a simple understanding of the requirements of interplanetary travel and the ability to reach a designated landing site. When the entry, descent, and landing phase of the mission must be considered to a greater extent, a program with greater accuracy must be employed. The program POST 2 (Program to Optimize Simulated Trajectories II) is therefore utilized for the entry, descent and landing analysis. POST 2 provides a flexible amount of input to the trajectory analysis and can be used for a variety of missions. The mission that was analyzed for purposes here is the Mars Science Laboratory, which will utilize new technology in the areas of supersonic parachutes and a sky-crane landing system, as shown in Figure 7.1. For such a mission like this, various parameters such as heat rate, max g's,

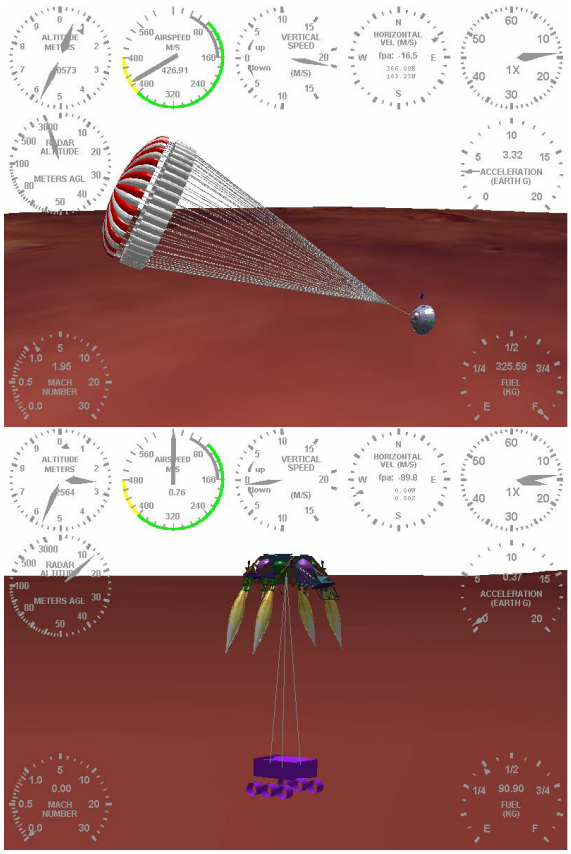

Figure 7.1: Parachute and Sky-Crane

and altitude at parachute deploy are of obvious interest. For instance the maximum heat rate is important for thermal protection system consideration. Max g's is important due to mission constraints on Mars Science Laboratory structural capabilities. The altitude at parachute deploy is of concern as well, as the altitude is directly linked to the ability of the vehicle to effectively slow itself for landing. The input deck for the POST 2 analysis has been included in Appendix C.

Since POST 2 can vary any parameters to optimize the trajectory, sets of four independent parameters were selected to reach the dependent conditions of the simulation. The independent variables were B-plane angle, flight path angle, time on parachute, and initial time. The initial time is used to target the longitude by artificially rotating the planet before entry, in turn causing the simulation to not brute force the entry to either inaccurately extend or decrease the central angle to reach the target. This assumption is acceptable because it was assumed that the actual time of arrival would be targeted well before arrival of the spacecraft. This change in time will only have a maximum variance of one Mars day, which is negligible to the change in declination and right ascension of the arrival asymptote for different arrival dates. The simulation dependent variables were latitude, longitude, landing radius, and a specified bank angle profile at entry. The input variables to the POST 2 simulation are the right ascension and declination of the approach asymptote, Julian day of arrival, and magnitude of velocity at infinity, all derived from the simple interplanetary analysis developed herein. The POST 2 optimizer would then optimize the flight path angle, B-plane angle, and central angle for the trajectory analysis.

As with any optimization method, the better the initial guess is, the quicker the optimization will proceed. For the EDL analysis, over 14,000 optimization cases were necessary to run. By supplying a decent first guess, the processing time could be significantly reduced. Therefore, the simulation initially ran a single case at 6 km/s entry velocity to yield a first guess for two of the independent variables, flight path angle and time on parachute. The entry velocity was chosen to 6 km/s because the majority of the cases of interest were in the range of 5.5 to 6.5 km/s. For this case, the initial flight path angle was optimized to –13.43 degrees and the time on parachute was optimized to 68.27 seconds.

Entry velocities above 7 km/s were disregarded, as they would not likely be considered due to extreme g's and heat loads. A total of 14,484 cases were then selected out of the 27,632 total cases generated from the launch/arrival space grid. The cases were run on a 64-node, 3.2 GHz Intel Xeon processors, Linux cluster taking approximately 14 hours to complete. If such computing power and time is not available, the grid used does not have to be as fine to yield similar results. A grid containing $1/8^{th}$ of the 14,484 cases that is equally spaced will yield nearly the same results and reduce computation time greatly. However, at the time of this research, the aforementioned cluster was undergoing testing as it had just been completed. Therefore, the 14,484 cases were all run as they provided an excellent test suite.

From the EDL analysis, performance characteristics were retrieved at specified events along the trajectory from each case. These events included the conditions at atmospheric entry and parachute deploy. In Figures 7.2 and 7.3, the inertial entry

velocity at atmospheric entry and altitude at parachute deploy is plotted using the same axes in Figure 6.1.

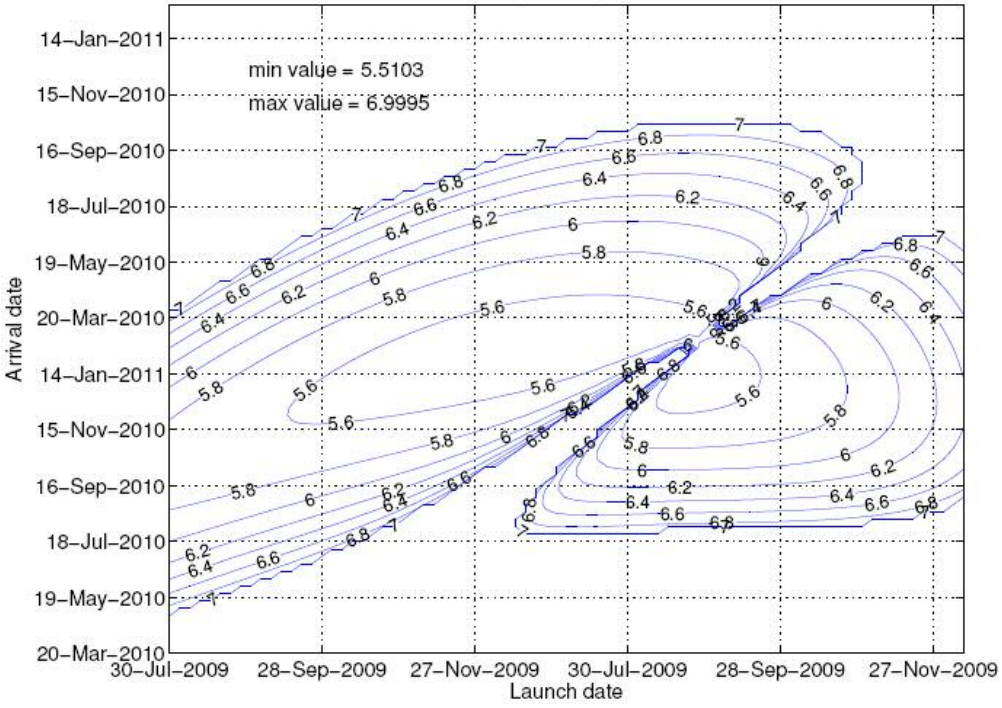

Figure 7.2: Inertial Entry Velocity at Atmospheric Entry

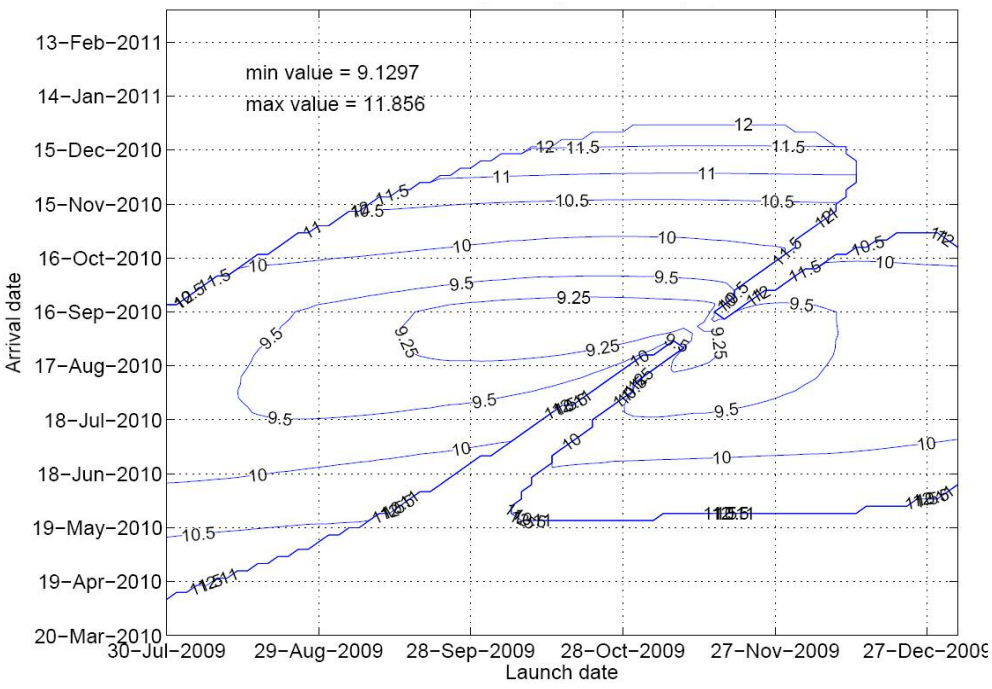

Figure 7.3: Altitude at Parachute Deploy

The parachute deploy trigger activates when the entry capsule slows to Mach 2.1. This trigger location will depend on the atmospheric density used. For these cases, the Mars Global Reference Atmospheric Model (MarsGRAM) is used. The atmospheric density in MarsGRAM, depends on time of day and solar longitude. One can note the nearly horizontal lines that run along Figure 7.3, just as solar longitude runs in Figure 6.1. Such correlation implies that solar longitude can have a great impact on the atmospheric density. This produces targeting inaccuracy of the entry, descent and landing system due to the increased drag. If the mission requires an altitude greater than 9.25 kilometers at parachute deploy, the targeting region is bounded even tighter than that of Figure 6.1. Only by running the EDL optimization analysis has further insight been obtained for the solar longitude or time of day effect.

As in all missions there are further constraints that need to be considered for mission designer. Max g's and heat rate are important when characterizing the physical limits of the entry vehicle. Max g's is plotted in Figure 7.4 and heat rate in Figure 7.5.

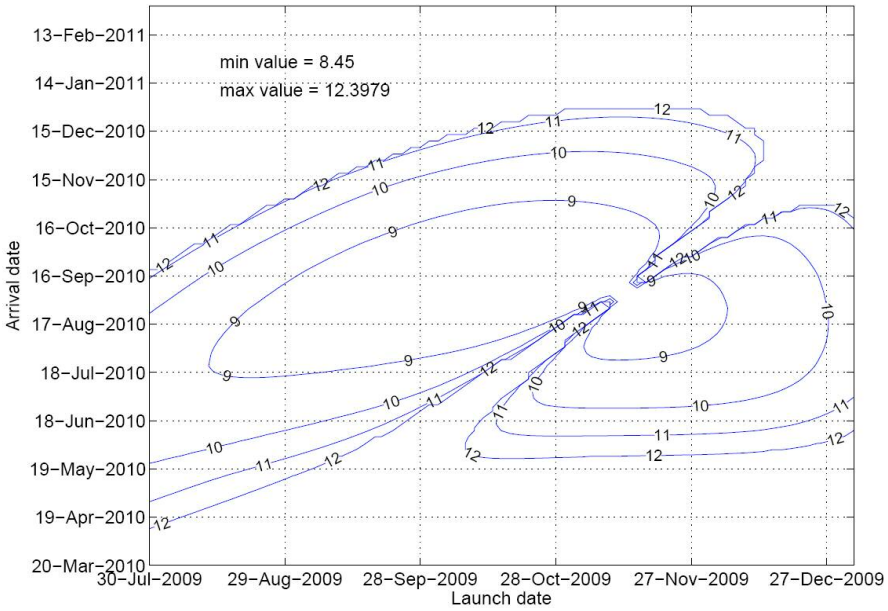

Figure 7.4: Max g's

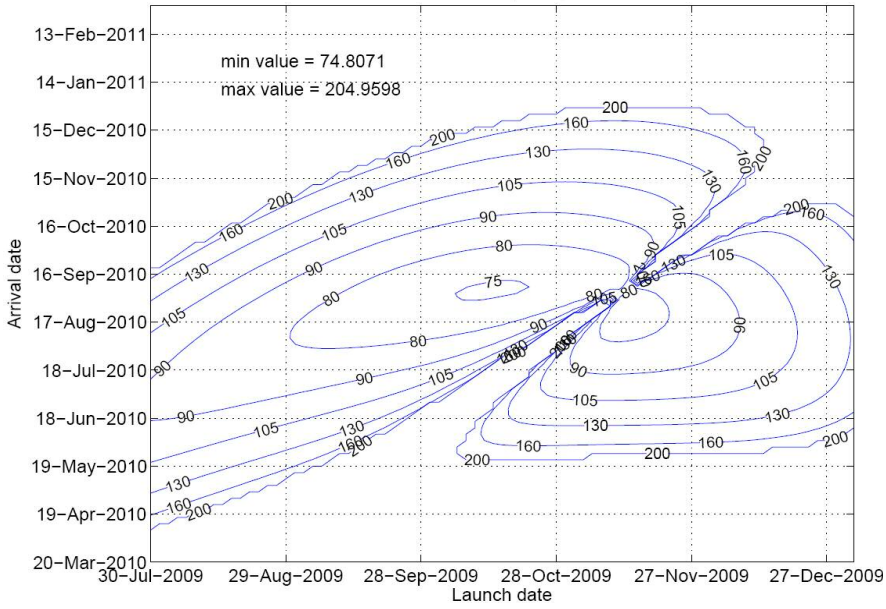

Figure 7.5: Max Heat Rate

Figures 7.4 and 7.5 both show a direct correlation to entry velocity. By running a sweep of entry velocities, one can determine the max capability of the mission for such variables as g-load and heat rate. Using such knowledge, one can effectively reduce the number of cases while still achieving the required results. This was done for the cases analyzed, as 12 g's was the max limit, therefore excluding all results over ~7 km/s entry velocity. The direct correlation of the EDL analysis to the entry velocity leads to the conclusion that entry conditions can greatly affect the mission requirements and need to be known to a fair degree of accuracy if the EDL analysis is to be accurate. The comparison of the generated entry conditions to the JPL entry conditions has verified the validity of the method and hence the validity of the entry results.

The applicability of this tool will now be shown for other dates. A launch grid ranging from September, 2013 to February, 2014 and arrival grid ranging from May, 2014 to March, 2015 was generated and plotted in Figure 7.6. The results from the simplified interplanetary method were then input to the EDL simulation. The results for

C_3, v_∞, and altitude at parachute deploy are overlaid on Figure 7.6. Altitude at parachute deploy is a necessary constraint, as altitude affects the time spent on the parachute. Time on chute is crucial for the navigation software and its ability to analyze the landing environment. Without adequate altitude, the entry system will be unable to accurately navigate to the desired landing site. For this case, the altitude at parachute deploy constraint has been set to 11.5 kilometers. The velocities at infinity are constrained to 7 km/s and lower due to its relation to max g's and heat rate (Figures 7.4 and 7.5 are directly correlated to Figure 6.1). The C_3 is constrained to 10 km^2/s^2 to simulate a small launch vehicle.

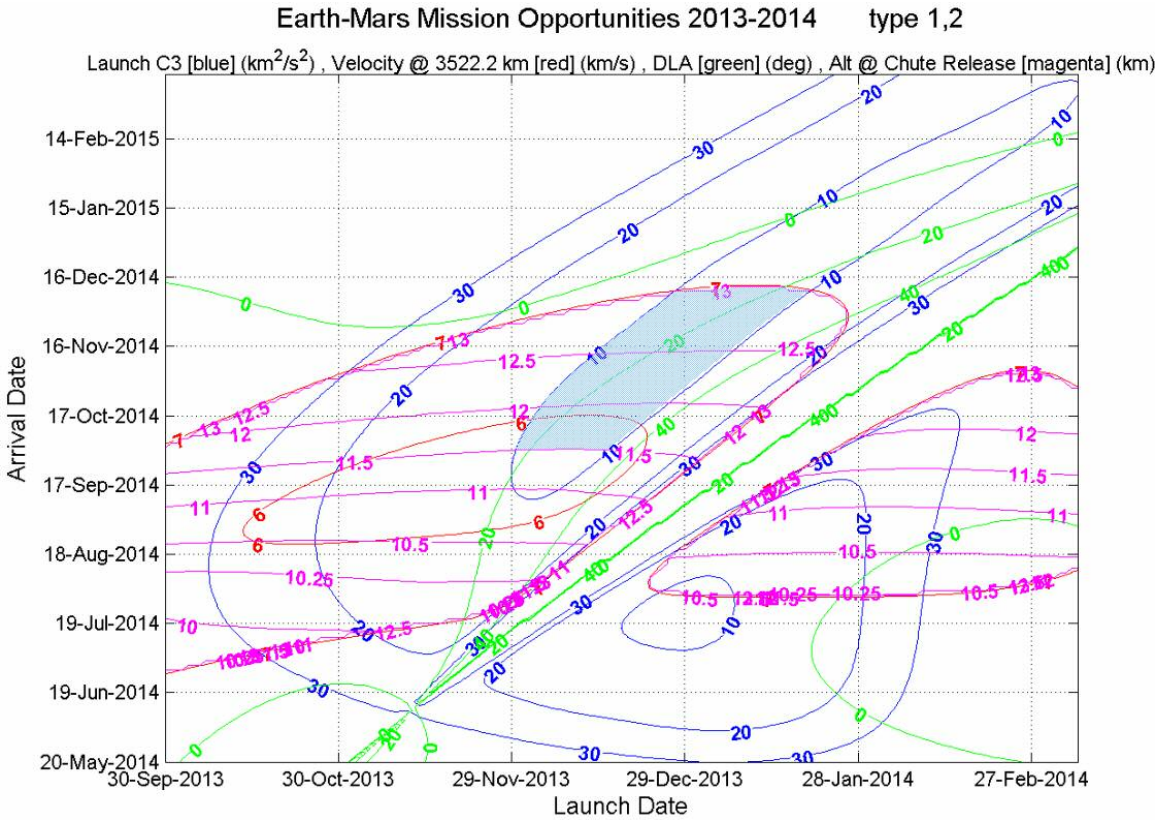

Figure 7.6: Interplanetary/EDL Combined Porkchop Plot

The importance of doing an EDL analysis is seen to be important when selecting a mission window. Without such insight, the development of an interplanetary mission can be greatly hindered and in some cases nullified by the overlooked conditions imposed by entry, descent and landing, i.e. if the proposed mission cannot reach an adequate altitude at parachute deploy, it will be impossible to safely land on the surface. The conclusion presented by this research is that a greater degree of understanding in interplanetary mission planning is possible through EDL analysis combined with a simplified interplanetary analysis method. Another pertinent conclusion is the direct transformation of entry velocity to specific variables such as max g's and heat rate. Since entry velocity is derived from the interplanetary analysis, the results must be held to a high degree of accuracy for the EDL analysis to be correct. The verification of the interplanetary analysis in chapter 6 proves the validity of the EDL results.

7.1 Selecting the Optimal Mission

For any interplanetary mission proposal, an optimal mission according to specific mission parameters is necessary. As mentioned earlier, the altitude at parachute deploy can create a constraint. By selecting such bounds and an objective function to minimize, a constrained optimization problem can be created that will converge on the optimum mission according to the design specifications. To perform this optimization, the constraints and objective function to minimize must first be selected.

The objective function to minimize is max g's. From Figure 7.4, entry velocity is a surrogate of max g's, and will thus be used. The constraints will be C_3, declination of the launch asymptote, and the altitude at parachute deploy. The objective function and constraints are shown on the next page.

Design Variables: $\vec{x} = \begin{Bmatrix} \text{Launch.date} \\ \text{Arrival.date} \end{Bmatrix}$

Objective Function: $f(\vec{x}) = \max\ g\text{'s}\ (\text{surrogate is velocity}) \Rightarrow \text{minimum}$

Constraint 1: $g_1(\vec{x}) = \text{altitude at parachute deploy} \geq 11.5\text{km}$

Constraint 2: $g_2(\vec{x}) = C3 \leq 10\ {}^{km^2}/_{s^2}$

Constraint 3: $g_3(\vec{x}) = \text{declination} \leq 40^\circ$

The optimization uses a sequential quadratic programming method to find the minimum velocity subject to the constraints. Data were generated from the interplanetary method for the C_3, entry velocity and declination of the approach asymptote for the launch and arrival date combination used in Figure 7.6. The data for altitude at parachute deploy were generated previously by POST 2 and was tabulated and linearly interpolated for the dates desired.

The optimization proceeds as shown in Figure 7.7 along the black line, starting at the red star, ending at the blue star. A minimum entry velocity converged at 5.88 km/s in 19 iterations for a launch date of December 4th, 2013 and an arrival date of October 2, 2014. The optimum from this type of analysis is seen in Figure 7.7, although the shaded region represents all the dates that may be considered according to the given constraints.

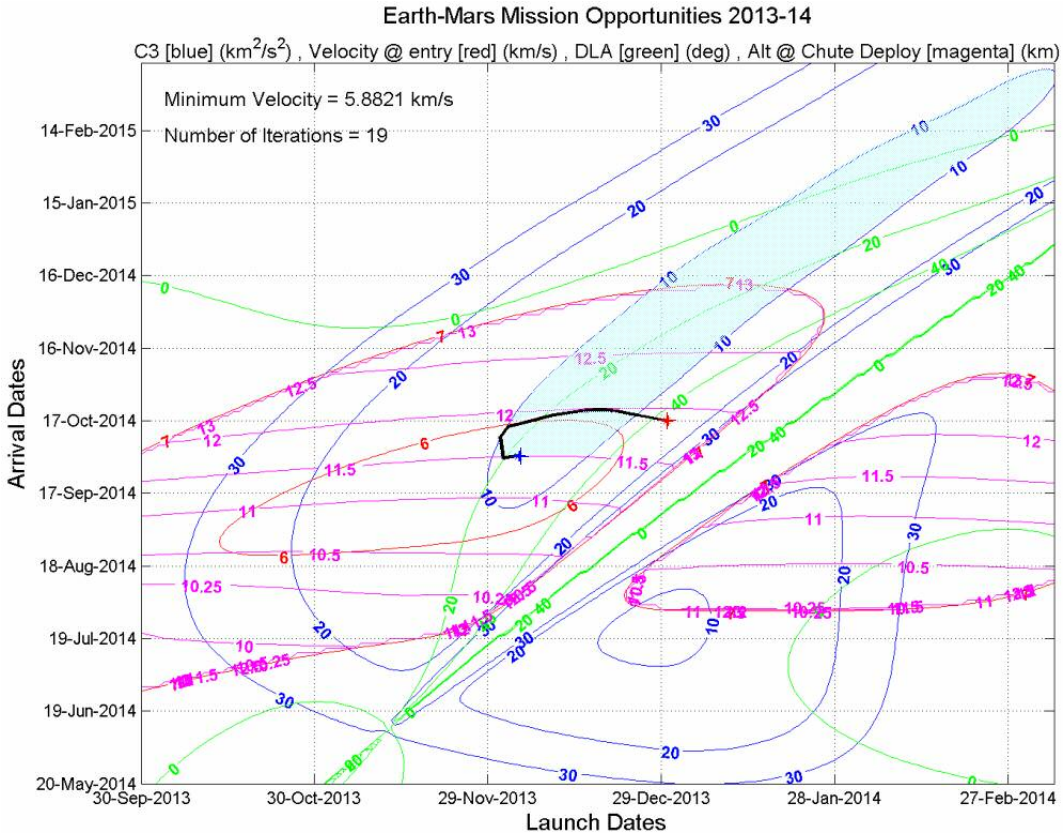

Figure 7.7: Finding the Optimum Mission

The importance of finding the optimum is obvious from a mission perspective. The optimum is the trajectory that satisfies the specified mission constraints while minimizing the objective function (max g's). Often, the mission analysis will use a nominal trajectory in designing the various components of the mission. With the presented method, the relay of interplanetary trajectories and EDL analysis between two teams is no longer necessary and the analysis can find the nominal trajectory without an EDL-interplanetary iteration.

8 Conclusion

A link that removes the iteration between interplanetary trajectory design and EDL design has been the goal of this research. An effective tool has been presented that couples both of these analyses. The tool resolves a planetary landing site location from the interplanetary trajectory using patched-conic analysis and B-plane methodology. From this tool, results were generated and validated versus JPL data for a 2009-2010 Mars mission opportunity. An EDL simulation was then integrated to analyze the constraints placed on the mission by the EDL analysis. By coupling these analyses, conclusions can be made about the effectiveness of coupling the interplanetary calculations to the EDL analysis.

For the interplanetary calculations, a variety of ephemerides were investigated and reviewed for accuracy. Lower precision planetary ephemerides were found to yield comparable results to higher precision ephemerides when calculating the velocity at infinity upon arrival. Thus, a lower precision planetary ephemeris is adequate for the scope of interplanetary analysis presented. For the physical ephemeris, differences were found to neither grow nor shrink dramatically.

An in-depth review was given to the solution of Lambert's problem, resulting in an improved solution method that is ideal for interplanetary calculations. Furthermore, the Lambert solution was extended beyond normal textbook solutions, which neglects the possibility of the Newton iteration jumping to the multiple revolution curves. This Lambert extension led to a combined Newton-bisection algorithm that removed the possibility of convergence on a multiple revolution solution.

The primary conclusion from this research is that top-level interplanetary and EDL analysis should be performed together in an effort to reduce analysis time. For example, the interplanetary trajectory analysis might recommend the minimum entry velocity cases. From the EDL analysis, it may be discovered that even though heat rate and g loads were minimized at the minimum entry velocity, the altitude at parachute deploy could be unacceptable depending on the constraints of the mission. An iterative process would then have to be performed between the two analyses until an interplanetary trajectory is found that satisfies the EDL constraints. By the coupling of both analyses, the EDL analysis is overlaid to the interplanetary analysis seamlessly, yielding the necessary capability in defining the mission window while reducing the number of iterations between the two. When the interplanetary calculations are integrated with the EDL analysis, only one iteration is effectively needed to determine the ideal trajectory, versus the frequent iterations encountered by traditional analysis.

The next step to be taken in this research is to generalize the EDL analysis for a variety of missions. Specific parameters should be commonplace in this kind of analysis. For example, the only inputs for the entry capsule parameters could be the ballistic coefficient, mass, and cross sectional area assuming a ballistic entry trajectory. Using these inputs, a ballistic trajectory could be analyzed that would allow one to generate porkchop plots including max g loads and heat rate. Verification and validation would need to be performed on such a generalized analysis with respect to a specialized analysis like the Mars Science Laboratory.

Interplanetary mission analysis is a very important regime of study. The simplifications presented herein and the assumptions made have shown that even a basic

understanding of the principles can yield a solution that is adequate for top-level mission analysis and planning. Furthermore, the method detailed by this research substantially reduces the time that has been previously required between interplanetary and EDL analysis. This reduction in time will be a key element in developing exciting and new concepts in support of NASA's new "Vision for Space Exploration".

9 Appendices

9.1 Appendix A – Derivation of the Time of Flight Equation via Universal Variables

The derivation of the universal variable z starts with the angular momentum and orbital energy equations:

$$h = r^2\dot{v} = \sqrt{\mu p}$$

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

where $\dot{v}$ is the rate of change of the true anomaly with respect to time, r the radius, v the velocity, a the semi-major axis, p the semi-latus rectum, and μ the gravitational constant. First, introduce the radial and transverse components of velocity,

$$v^2 = \dot{r}^2 + r^2\dot{v}^2$$

and eliminate $\dot{v}$ using a relationship derived from the angular momentum, $r\dot{v} = \frac{\sqrt{\mu p}}{r}$.

Substitute the expression for v^2 into the energy integral to get,

$$\dot{r}^2 = -\frac{\mu p}{r^2} + \frac{2\mu}{r} - \frac{\mu}{a} \qquad 9.01$$

Note that if r is unbounded, then a must be less than zero.

The goal now becomes to develop a general solution for r. To accomplish this, a new independent variable, χ (also known as the universal variable), is introduced by,

$$\dot{\chi} = \frac{d\chi}{dt} = \frac{\sqrt{\mu}}{r} \qquad 9.02$$

Dividing equation 9.01 by $(9.02)^2$ and simplifying yields,

$$d\chi = \frac{dr}{\sqrt{-p + 2r - r^2/a}}$$

Integrating the above equation yields,

$$\chi + c_o = \sqrt{a}\sin^{-1}\left(\frac{r/a - 1}{\sqrt{1 - p/a}}\right) \qquad 9.03$$

where the variable c_o is the constant of integration and the variable a must always be greater than zero. For an ellipse, the eccentricity is $e = \sqrt{1 - p/a}$, so that when substituted into equation 9.03 and simplified, a solution for r is obtained.

$$r = a\left(1 + e\sin\frac{\chi + c_o}{\sqrt{a}}\right) \qquad 9.04$$

With the solution of r, the physical importance of χ is defined by comparing equation 9.04, to the classical representation of radius, $r = a(1 - e\cos E)$, yielding,

$$\sin\frac{\chi + c_o}{\sqrt{a}} = -\cos E$$

The sine term can be transformed to a cosine term,

$$\sin\frac{\chi + c_o}{\sqrt{a}} = -\cos\left(\frac{\pi}{2} + \frac{\chi + c_o}{\sqrt{a}}\right)$$

Combining the above two equations yields the eccentric anomaly in terms of the universal variable χ,

$$E = \frac{\pi}{2} + \frac{\chi + c_o}{\sqrt{a}} \qquad 9.05$$

To evaluate the constant of integration, an expression for the initial eccentric anomaly is developed. To do this, the expression for time is derived by substituting equation 9.04 into equation 9.02, and separating out the derivatives,

$$\sqrt{\mu}dt = a\left(1 + e\sin\frac{\chi + c_o}{\sqrt{a}}\right)d\chi$$

Then by integration,

$$t\sqrt{\mu} = a\chi - ae\sqrt{a}\left(\cos\frac{\chi + c_o}{\sqrt{a}} - \cos\frac{c_o}{\sqrt{a}}\right) \qquad 9.06$$

As can be seen from equation 9.06, at time t=0, the variable χ must equal zero as well because semi-major axis (a) must be greater than zero. Equation 9.05 at time t=0 for initial E_o must then be,

$$E_o = \frac{\pi}{2} + \frac{c_o}{\sqrt{a}}$$

By subtracting the two equations, the definition of the variable χ is,

$$\chi = \sqrt{a}\left(E - E_o\right) \qquad 9.07$$

From equation 9.07, the variable χ is the difference in eccentric anomaly. Generalizing the variable χ to be only in terms of eccentric anomaly, the new variable z is introduced,

$$z = \frac{\chi^2}{a} = \left(E - E_o\right)^2 \qquad 9.08$$

The variable *z* is equally valid for hyperbolic orbits as the change in hyperbolic anomaly. From Bate, Mueller, and White [5], it is shown that the eccentric and hyperbolic anomalies are related by $\cosh F = \cos E$. By using the identity $\cosh\theta = \cos i\theta$, it is seen that $E = \pm iF$. Therefore, the change in hyperbolic anomalies can be denoted by $z = -\left(F_2 - F_1\right)^2$, and is therefore always negative.

Now that the universal variable χ has been defined, the definition of the f and g functions via universal variables needs to be determined. These expressions have been developed in Bate, Mueller, and White [5],

$$\mathrm{f} = 1 - \frac{\mathrm{r}_2}{p}\left(1 - \cos\Delta v\right) = 1 - \frac{\chi^2}{\mathrm{r}_1} C \tag{9.09}$$

$$\mathrm{g} = \frac{\mathrm{r}_1 \mathrm{r}_2 \sin\Delta v}{\sqrt{\mu p}} = \mathrm{t} - \frac{\chi^3}{\sqrt{\mu}} \mathrm{S} \tag{9.10}$$

$$\dot{\mathrm{f}} = \sqrt{\frac{\mu}{p}} \frac{\left(1 - \cos\Delta v\right)}{\sin\Delta v}\left(\frac{1 - \cos\Delta v}{p} - \frac{1}{\mathrm{r}_1} - \frac{1}{\mathrm{r}_2}\right) = \frac{-\sqrt{\mu}}{\mathrm{r}_1 \mathrm{r}_2} \chi\left(1 - z\mathrm{S}\right) \tag{9.11}$$

$$\dot{\mathrm{g}} = 1 - \frac{\mathrm{r}_1}{p}\left(1 - \cos\Delta v\right) = 1 - \frac{\chi^2}{\mathrm{r}_2} C \tag{9.12}$$

Where the variables S and C are both functions of the variable z,

$$\begin{aligned} &\text{For Elliptical Motion :} \\ &C\left(z\right) = \frac{1 - \cos\sqrt{z}}{z} \\ &\text{For Hyperbolic Motion :} \\ &C\left(z\right) = \frac{1 - \cosh\sqrt{-z}}{z} \end{aligned} \tag{9.13}$$

$$\begin{aligned} &\text{For Elliptical Motion :} \\ &\mathrm{S}\left(\mathrm{z}\right) = \frac{\sqrt{z} - \sin\sqrt{z}}{\sqrt{z^3}} \\ &\text{For Hyperbolic Motion :} \\ &\mathrm{S}(\mathrm{z}) = \frac{\sinh\sqrt{-z} - \sqrt{-z}}{\sqrt{-z^3}} \end{aligned} \tag{9.14}$$

The goal now is to reduce these equations to a function of time of flight and change in eccentric anomaly. First, start by solving for χ in equation 9.09,

$$\chi = \sqrt{\frac{\mathrm{r}_1 \mathrm{r}_2 \left(1 - \cos\Delta v\right)}{p\mathrm{C}}} \tag{9.15}$$

Substitute equation 9.15 into equation 9.11 to yield,

$$\frac{\left(1 - \cos\Delta v\right)}{\sin\Delta v}\left(\frac{1 - \cos\Delta v}{p} - \frac{1}{\mathrm{r}_1} - \frac{1}{\mathrm{r}_2}\right) = -\sqrt{\frac{1 - \cos\Delta v}{\mathrm{r}_1 \mathrm{r}_2}} \frac{\left(1 - z\mathrm{S}\right)}{\sqrt{\mathrm{C}}}$$

Multiplying by $r_1 r_2$ on both sides, and simplifying yields,

$$r_1 r_2 \frac{(1-\cos\Delta v)}{p} = r_1 + r_2 - \frac{\sqrt{r_1 r_2}\sin\Delta v}{\sqrt{1-\cos\Delta v}} \frac{(1-zS)}{\sqrt{C}} \tag{9.16}$$

To simplify things further, the variable A is introduced as a function of r_1, r_2, and Δv,

$$A = \frac{\sqrt{r_1 r_2}\sin\Delta v}{\sqrt{1-\cos\Delta v}} \tag{9.17}$$

The variable y is also introduced as,

$$y = r_1 r_2 \frac{(1-\cos\Delta v)}{p} = r_1 + r_2 - A\frac{(1-zS)}{\sqrt{C}} \tag{9.18}$$

As the left side of equation 9.16 is identical to part of equation 9.15, a new expression for χ becomes,

$$\chi = \sqrt{\frac{y}{C}} \tag{9.19}$$

Now, solving for t in equation 9.10,

$$\sqrt{\mu}t = \chi^3 S + \frac{r_1 r_2 \sin\Delta v}{\sqrt{p}} \tag{9.20}$$

To simplify equation 9.20 into functions of only z, the square root of y is determined from equation 9.19 and substituted into equation 9.18 to yield,

$$\sqrt{y} = \sqrt{\frac{r_1 r_2 (1-\cos\Delta v)}{p}}$$

Then multiply both sides by equation 9.17 and simplify to get,

$$A\sqrt{y} = \frac{r_1 r_2 \sin\Delta v}{\sqrt{p}}$$

The above equation is equal to the last term of equation 9.20, so that the time of flight equation is,

$$\sqrt{\mu}\mathrm{t} = \chi^3 \mathrm{S} + \mathrm{A}\sqrt{y} \tag{9.21}$$

By using equations 9.19 and 9.21, the f, g and g-dot expressions are simplified to,

$$\mathrm{f} = 1 - \frac{y}{\mathrm{r}_1} \tag{9.22}$$

$$\mathrm{g} = \mathrm{A}\sqrt{\frac{y}{\mu}} \tag{9.23}$$

$$\dot{\mathrm{g}} = 1 - \frac{y}{\mathrm{r}_2} \tag{9.24}$$

To solve the f and g expressions, the unknown variable z in equation 9.21 must be determined. To do this, a Newton Iteration scheme is implemented that requires the derivative of equation 9.21 with respect to z. To do this, first differentiate equation 9.21 with respect to z to get,

$$\sqrt{\mu}\frac{\mathrm{dt}}{\mathrm{d}z} = 3\chi^2 \frac{\mathrm{d}\chi}{\mathrm{d}z}\mathrm{S} + \chi^3 \frac{\mathrm{dS}}{\mathrm{d}z} + \frac{\mathrm{A}}{2\sqrt{y}}\frac{\mathrm{d}y}{\mathrm{d}z} \tag{9.25}$$

The derivatives with respect to z are obtained from equations 9.18 and 9.19 to give,

$$\frac{\mathrm{d}y}{\mathrm{d}z} = -\mathrm{A}\left[\sqrt{\mathrm{C}}\left(-\mathrm{S} - z\frac{\mathrm{dS}}{\mathrm{d}z}\right) - \frac{(1 - z\mathrm{s})}{2\sqrt{\mathrm{C}}}\frac{\mathrm{dC}}{\mathrm{d}z}\right] \tag{9.28}$$

$$\frac{\mathrm{d}\chi}{\mathrm{d}z} = \frac{1}{2\chi\chi}\left(\frac{\mathrm{d}y}{\mathrm{d}z} - \chi^2 \frac{\mathrm{dC}}{\mathrm{d}z}\right) \tag{9.29}$$

Substituting equations 9.28 and 9.29 into equation 9.25 and simplifying yields,

$$\sqrt{\mu}\frac{\mathrm{dt}}{\mathrm{d}z} = \chi^3\left(\frac{\mathrm{dS}}{\mathrm{d}z} - \frac{3\mathrm{S}\dfrac{\mathrm{dC}}{\mathrm{d}z}}{2\mathrm{C}}\right) + \frac{\mathrm{A}}{8}\left(\frac{3\mathrm{S}\sqrt{y}}{\mathrm{C}} + \frac{\mathrm{A}}{\chi}\right) \tag{9.30}$$

The derivatives with respect to C and S are given by,

$$\frac{dC}{dz} = \frac{1}{2z}\left(1 - zS - 2C\right) \tag{9.31}$$

$$\frac{dS}{dz} = \frac{1}{2z}\left(C - 3S\right) \tag{9.32}$$

Equation 9.30 can then be implemented in the Newton iteration scheme for the solution of z.

9.2 Appendix B – Simplified Interplanetary Targeting Tool: Manual

A Simplified Interplanetary Targeting Tool/Porkchop Plotter Using MATLAB

Jeremy Shidner
NIA / NASA Langley Research Center, Hampton, Virginia

Table of Contents

Page

1 Introduction 1

2 Method 1

3 Step-by-Step Instructions 2

3.1 Step 1: Startup 2

3.2 Step 2: Input Selections 2

3.2.1 Launch/Arrival Space 2

3.2.2 Targeting 3

3.2.3 Options 4

3.3 Step 3: Calculations 5

3.4 Step 4: Plotting 6

3.4.1 Plot Options 6

3.4.1.1 Launch Variables 6

3.4.1.2 Arrival Variables 7

3.4.2 User Definitions 8

3.5 Step 5: Retrieving the Output 8

3.5.1 Data Output 9

3.5.2 Plot Output 9

References 10

A Simplified Interplanetary Targeting Tool /Porkchop Plotter Using MATLAB

1 Introduction

Interplanetary mission planning is dictated by the mission year selected, allowing different trajectories to be considered in designing an effective mission. These trajectories offer specific advantages in such mission requirements as target latitude, season selection, and propulsive requirements. For the entry, descent, and landing investigator this data is crucial for completion of the analysis. 'Porkchop Plots' are an effective way for the investigator to retrieve the necessary data and for the mission designer to select the ideal launch/arrival space. The purpose of this tool is to offer the investigator an easy-to-use interface for retrieving the 'Porkchop' data. The data has also been formatted for use with Langley's Program to Optimize Simulated Trajectories (POSTII).

2 Method

The tool is designed to have a direct user interface. All data variables that are calculated by the program will be described later in this document. The only inputs required by the user will be the launch/arrival planets, mission timeline, targeting parameters, and program specified options. Plots of the results are available via the plotting tool that is incorporated in the primary interface for ease-of-use.

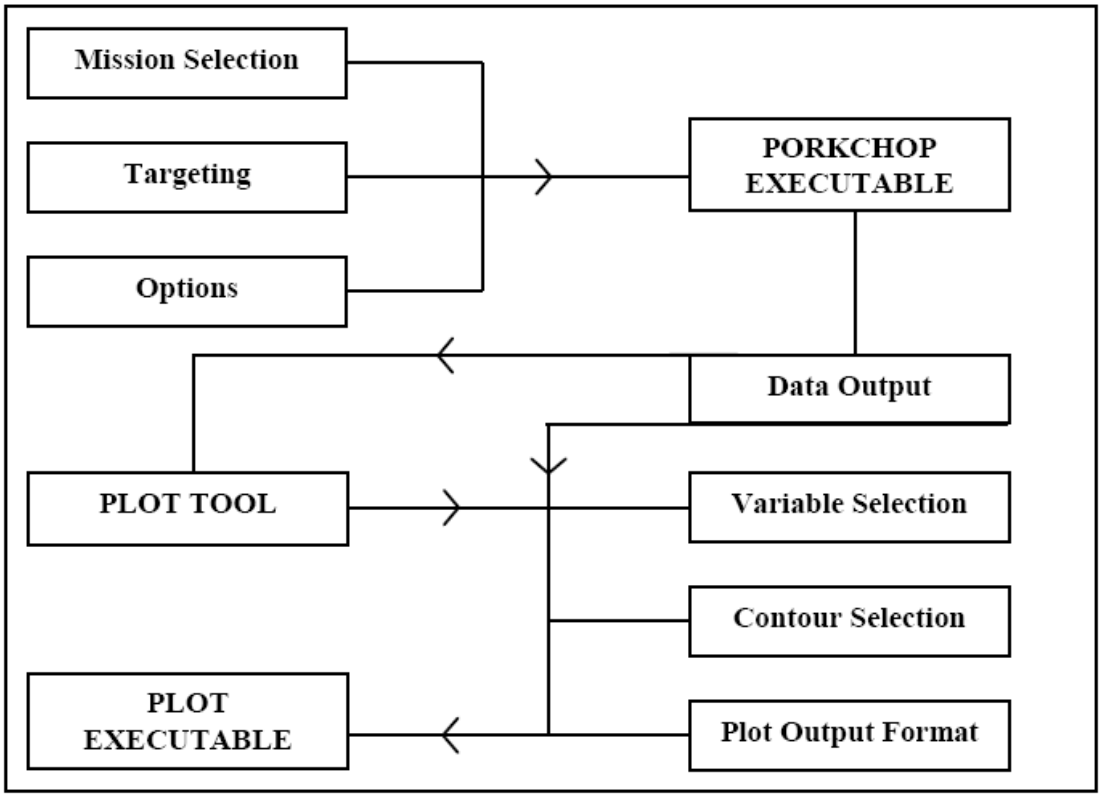

Figure 1 Porkchop Tool Flow Diagram

3 Step-by-Step Instructions

This section will outline the procedures for obtaining the Porkchop data and plots. All functions have been designed and run using MATLAB. Programming was performed on MATLAB Version 6.5. It is assumed that the user will have MATLAB version 6.5 or later installed on their computer in order to run the tool. The tool has also been tested and validated for MATLAB Version 7.0.

3.1 Step 1: Startup

The tool is provided as a set of MATLAB (.m) files located in the 'PC_GUI.zip' file. The files need to be unzipped to a directory of the user's choice. The primary interface is opened by execution of the 'MainGui.m' function from the MATLAB command prompt.

3.2 Step 2: Input Selections

The user is prompted in the main gui window to input the various parameters of interest. Inputs are performed by using the mouse and keyboard. There are two types of inputs used in the window, drop-down selections and user-keyed inputs. To input the user-keyed inputs, the user needs to click the white box corresponding to the variable of interest. A cursor will then flash inside the box. Input can then be accepted from the keyboard. The main gui window is shown in Figure 2.

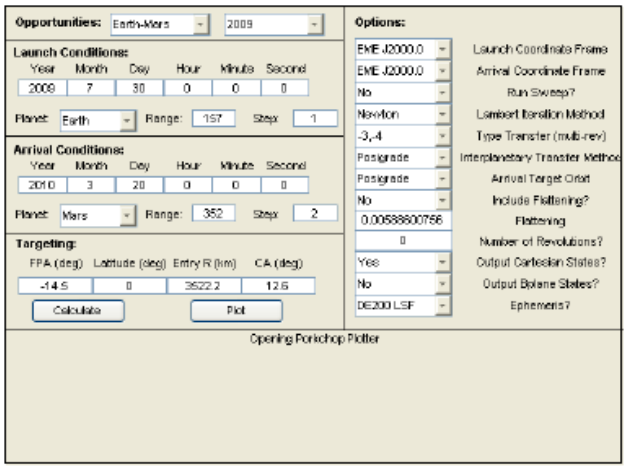

Figure 2 Porkchop Executable Window

3.2.1 Launch/Arrival Space

The first set of options discussed will be the Launch/Arrival Space selection. This includes the 'Opportunities', 'Launch Conditions', and 'Arrival Conditions' shown in Figure 2.

The 'Opportunities' options, allow the user to select a pre-defined mission window for the desired planet-to-planet combination. The default combination is Earth to Mars. If an opportunity has been selected, the Launch and Arrival Conditions will automatically be updated from the pre-defined mission-window database.

The 'Launch Conditions' options allow the user to fine tune the settings for the mission window desired. The specific launch date can be input by the user through the

year, month, day, hour, minute and second inputs. The 'Planet' selection defines the solar system body from which the mission is set to launch. The default is Earth. The 'Range' input allows the user to define the launch mission window, by declaring the range of dates that are extended from the user defined launch date. The units for 'Range' are days. 'Step' defines the sampling rate for the launch mission window specified. If 'Step' is set to 3, the tool will calculate a trajectory every 3 days for the launch mission window specified.

The 'Arrival Conditions' are displayed and used in the same manner as the 'Launch Conditions' options. The only difference is that the inputs and selections specified here are for the solar system body selected for arrival.

3.2.2 Targeting

The second set of options is the Targeting selections. This includes flight path angle ('FPA'), target latitude ('Latitude'), target radius for atmospheric entry ('Entry R') and initial estimate of the angle traveled from atmospheric entry to planet touchdown ('CA'), as shown in Figure 3.

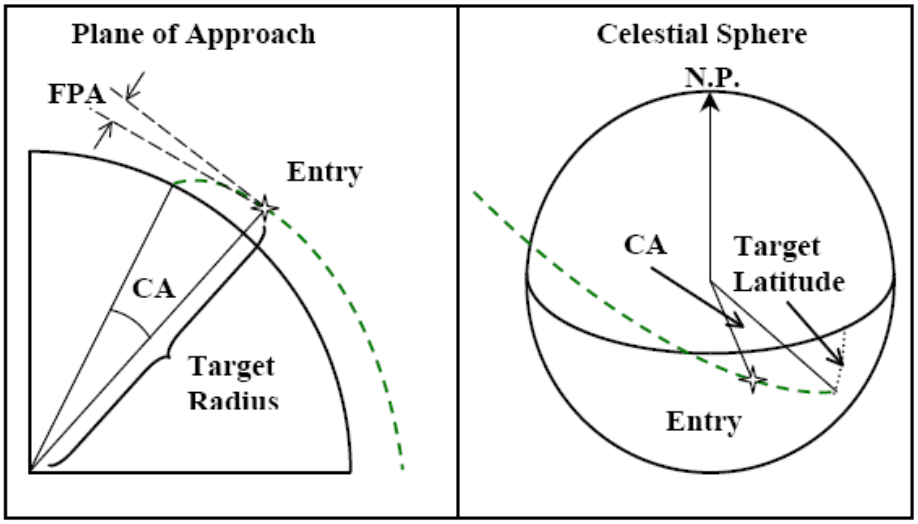

Figure 3 Targeting Variables

Target flight path angle will often be a mission specific variable that is studied by the EDL analysis team. Target flight path angle units are in degrees. The program will take into account whether or not the flight path angle is negative, meaning the target position is before periapse, or positive, meaning the target position is after periapse. The default value is –14.5 degrees.

Target latitude is generally a mission requirement. Target latitude units are in degrees. Negative valued target latitudes correspond to the southern hemisphere, while positive values correspond to the northern hemisphere. The default value is zero.

Target radius for atmospheric entry is also a general mission requirement. Target radius units are in kilometers. The program will calculate this variable in a geocentric coordinate system. The actual targeted radius, when flattening is included, will not be the

user-defined radius if the target location is at a non-zero degree latitude. The default value is 3,522.2 kilometers.

The angle traveled from atmospheric entry to planet touchdown is also known as the central angle or 'CA'. This is often the variable that is determined by the EDL analysis team once a feasible simulation has been developed. Central angle units are in degrees. The default value is 12.6 degrees, which was taken from a ballistic, unguided trajectory through the Mars atmosphere.

3.2.3 Options

The 'Options' set of selections allows the user to fine-tune the output data format. The selections relate to coordinate frames, variable sweeps, calculation methods, trajectory options, and planetary flattening. There are two options included that allow the user to output an ASCII state file at arrival in two separate coordinate frames.

The first options selection includes the selection of the coordinate frames used in the launch and arrival trajectory outputs, 'EME J2000', 'IAU Vector' and 'Prime Meridian'. The last two coordinate frames share the planet equator as the fundamental plane. The first selection's fundamental plane and direction is the Earth mean equator and vernal equinox of J2000. The second selection's fundamental plane and direction is the mean planet equator of date and the ascending node of the mean planet equator of date and earth mean equator of J2000.0 (EME J2000), otherwise known as the IAU vector. The third selection's fundamental plane and direction is the mean planet equator of date and the Prime Meridian of date. The third selection is commonly known as a cartographic coordinate frame.

The 'Run Sweep?' option allows the user to generate targeting sweeps over a range of inputs for either flight path angle, latitude or central angle for one launch-arrival date pair. Only one of these variables may be used when the sweep option is selected. To input the sweep values, the user needs to click the white box corresponding to the sweep variable of interest. The format to input the sweep variables is to either list the variables with spaces between (e.g. – '1 2 3 4'), or to define a start and end for the variable with a step size defined (e.g. – '-15:-0.5:-11'). The tool will save the results in the ASCII state file.

There are three points addressed in calculation of the interplanetary trajectory, the lamber solver, transfer method, and solar system body positions. The 'Lambert Iteration Method' allows the user to change the lambert solver calculation method in order to optimize either speed or accuracy. This can be useful when attempting to calculate an extremely hyperbolic trajectory because the Newton technique will have difficulty in converging to an answer, while Bisection will converge much more rapidly. The 'Type Transfer (multi-rev)' option dictates one of the two types of transfers possible in multiple revolution trajectories about the sun. Specifically, the plus and minus three and four options refer to the direction of approach for convergence used in the Lambert solver. The 'Interplanetary Transfer Method' allows the user to select either a posigrade or retrograde trajectory with respect to the direction of orbital rotation by the planets selected. The 'Ephemeris?' option allows the user to select a specific ephemeris to use when calculating the initial and final positions of the planets for the dates selected on the interplanetary trajectory. The 'DE405' ephemeris is considered to be the most accurate ephemeris publicly released by JPL. The 'DE200 LSF' is a least squares fit of JPL's old

ephemeris, DE200. This selection is said to be accurate to within twenty-five arc-seconds for the terrestrial planets. The 'VanFlandern' ephemeris is computed from Van Flandern's low precision formulae for the nine planets in our solar system.

The 'Include Flattening?' and 'Flattening' options will switch the arrival targeting from a spherical planet to an elliptical planet defined by the flattening parameter. The tool will automatically include the flattening value for whatever the target planet selected is. The user can change the flattening value if desired. Flattening is calculated with respect to a geocentric coordinate system of the arrival planet.

Finally, the arrival conditions can either be output as cartesian states or bplane states to a .txt file depending on the selection of 'Output Cartesian States?' and 'Output Bplane States?'. The two files both share a common first word, 'pc' for porkchop, or if a sweep is run, 'sweep' will replace 'pc'. The output format for the cartesian states is ten variables listed as columns in the 'pc_cartes_*year*.txt' ASCII output file, where *year* will be replaced by the corresponding launch years. The tenth column in the file is the magnitude of velocity at atmospheric entry, or one of the sweep variables if a sweep is performed. The format of pc_cartes_year.txt is shown in table 1.

pc_cartes_year.txt									
Case#	**LJD**	**AJD**	**x**	**y**	**z**	**vx**	**vy**	**vz**	**variable**
Case number	Launch Julian Date	Arrival Julian Date	Position vectors at atmospheric entry			Velocity vectors at atmospheric entry			Defined by sweep

Table 1 Cartesian ASCII File Description

The output format for the bplane states is nine variables listed as columns in the 'pc_bplane_sort_*year*.txt' ASCII output file, where *year* will be replaced by the corresponding launch years. The ninth variable is the magnitude of velocity at atmospheric entry, or the sweep variable if a sweep is performed. The format of pc_bplane_sort_year.txt is shown in table 2.

pc_bplane_sort_year.txt								
Case#	**LJD**	**AJD**	**bpang**	**vinf**	**DAP**	**RAP**	**v_entry**	**variable**
Case Number	Launch Julian Date	Arrival Julian Date	B-plane Angle	Velocity at infinity	Declination of vinf	Right Ascension of vinf	Velocity at entry	Defined by sweep

Table 2 Bplane ASCII File Description

3.3 Step 3: Calculations

Once the input selections have been made, the calculations are ready to be performed. The 'Calculate' button located in the Targeting section will execute the main driver when selected. When the calculations are being performed, the tool will update the user in the main gui window with the tool's current operations. Once the tool has finished, output data will be saved to either of the two ASCII text files if they are

selected, and plot data will be saved to a matlab data file that is labeled 'TrajData*year*.mat', where *year* is the actual launch years of the trajectory calculations.

3.4 Step 4: Plotting

Selecting 'Plot' in the main gui window activates the plotting tool. If calculations have been performed, the plotting tool will open. If calculations have not been performed, a window will open asking for the file location of the plot data. This is the 'TrajData*year*.mat' file mentioned above. Select the corresponding file and press 'Ok' to load the plot data and open the plotting tool.

The user is then prompted in the plot window to input the various parameters of interest. There are three types of inputs used in the window, drop-down selections, check boxes, and user-keyed inputs. The plot window is shown in Figure 4.

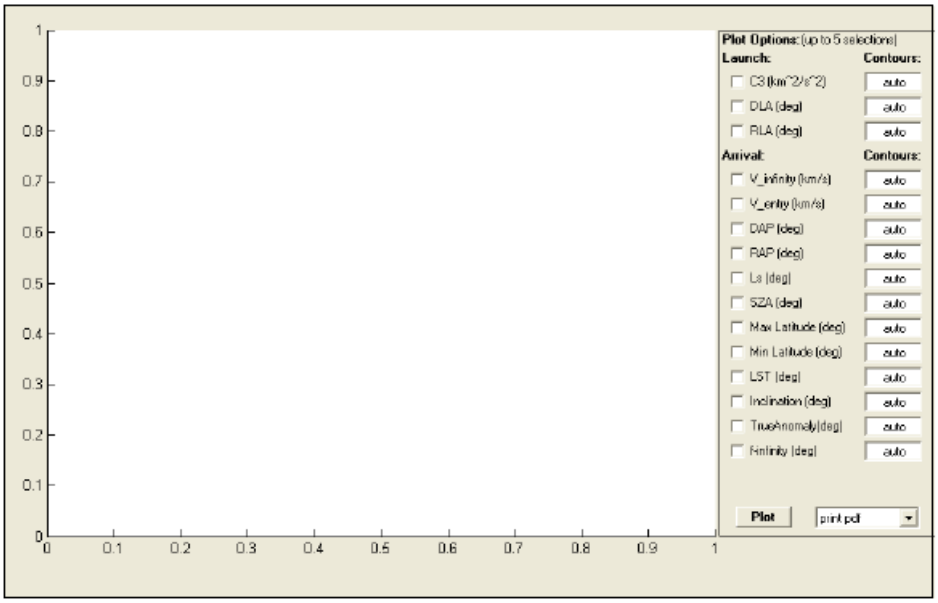

Figure 4 Plot Executable Window

3.4.1 Plot Options

There are a total of fifteen possible variables to plot in the window. Only a maximum of 5 variables may be plotted at one time. These variables are displayed directly underneath 'Launch:' and 'Arrival:'. The variable name and variable unit definition is displayed next to its checkbox. To select the variable for plotting, use the mouse to click the corresponding check box. The variables are described below in sections 3.4.1.1 and 3.4.1.2.

3.4.1.1 Launch Variables

C3: The variable C3 is the launch energy required for the spacecraft heliocentric trajectory from the launch planet to the arrival planet selected. Technically, C3 is the same as the square of departure hyperbolic excess velocity (V_∞^2). Contours are automatically plotted for values less than or equal to 40 in intervals of 10. (km^2/s^2)

DLA : DLA is the planetocentric declination of the departure V_∞ vector versus the mean equator of date. Contours are automatically plotted in multiples of 10. (deg)

RLA : RLA is the planetocentric right ascension of the departure V_∞ vector versus the mean equator of date and coordinate frame selected in the main gui window. Contours are automatically plotted in multiples of 10. (deg)

3.4.1.2 Arrival Variables

V_infinity: V_infinity, represents the planetocentric arrival hyperbolic excess velocity. V_infinity is determined by the magnitude of the difference of the planetary heliocentric orbital velocity and the spacecraft heliocentric velocity. Contours are automatically plotted for values less than or equal to 6 in intervals of 1. (km/s)

V_entry: V_entry is the velocity of the spacecraft at the targeted radius and flight path angle. Contours are automatically plotted for values less than or equal to 8 in intervals of 1. (km/s)

DAP: DAP is the planetocentric declination of the arrival V_∞ vector versus the mean equator of date. Contours are automatically plotted in multiples of 10. (deg)

RAP: RAP is the planetocentric right ascension of the arrival V_∞ vector versus the mean equator of date and coordinate frame selected in the main gui window. Contours are automatically plotted in multiples of 10. (deg)

Ls: Ls is the solar longitude of date. Solar longitude is measured from the intersection of the ascending node of the planet's orbital plane with the planet mean equator of date, along the planet orbital plane, to the planet-sun vector of date. It is important to note that solar longitude is always calculated with respect to the DE200 LSF planetary ephemeris in this tool. Contours are automatically plotted in multiples of 20 along the right edge. (deg)

SZA: SZA is the solar zenith angle at the targeted radius of date. Solar zenith angle is the angle between the planet-sun vector of date and the position vector of the spacecraft at the targeted radius and flight path angle. Contours are automatically plotted in multiples of 20. (deg)

Max Latitude: Max Latitude is the maximum achievable Northern latitude of date. Max Latitude uses the central angle input from the main gui window to

determine what the touchdown location is from the targeted entry point. Contours are automatically plotted in multiples of 10. (deg)

Min Latitude: Min Latitude is the maximum achievable Southern latitude of date. Min Latitude uses the central angle input in the same fashion as Max Latitude to calculate the touchdown position. Contours are automatically plotted in multiples of 10. (deg)

LST: LST is the Local Solar Time at the targeted spacecraft position. LST is measured from the planet-sun vector parallel to the mean equator of date, to the position vector parallel to the mean equator of date. Contours are automatically plotted in multiples of 20. (deg)

Inclination: Inclination is the inclination of the hyperbolic approach orbit with respect to the planet mean equator of date. Contours are automatically plotted in multiples of 10. (deg)

True Anomaly: True Anomaly is the true anomaly of the targeted spacecraft position in the perifocal coordinate system. Contours are automatically plotted in multiples of 2. (deg)

f-infinity: The variable f-infinity is the angle traveled from entry at the planet sphere of influence of date to the spacecraft targeted position. Contours are automatically plotted in multiples of 10. (deg)

3.4.2 User Definitions

If the user does not desire the contours that are automatically plotted, the user may define specific contour values in the 'Contours:' column. To input the values, the user needs to click the white box corresponding to the variable of interest. The format to input the contour values is to either list the values with spaces between (e.g.–'10 20 30 40'), or to define a start and end for the value with a step size defined (e.g.–'-10:10:30').

3.5 Step 5: Retrieving the Output

Once the plot options have been selected, the user may select from the drop down menu labeled 'print pdf' to save the porkchop plot to either pdf format, jpg format ('print jpg'), or to have nothing saved when the porkchop plot is generated ('print none'). Once all plotting options have been selected, the user may execute the plotting function by pressing the 'Plot' button. The data is now ready for analysis and exporting, as seen in a sample case in Figure 5 for Earth-Mars Opportunities in 2009.

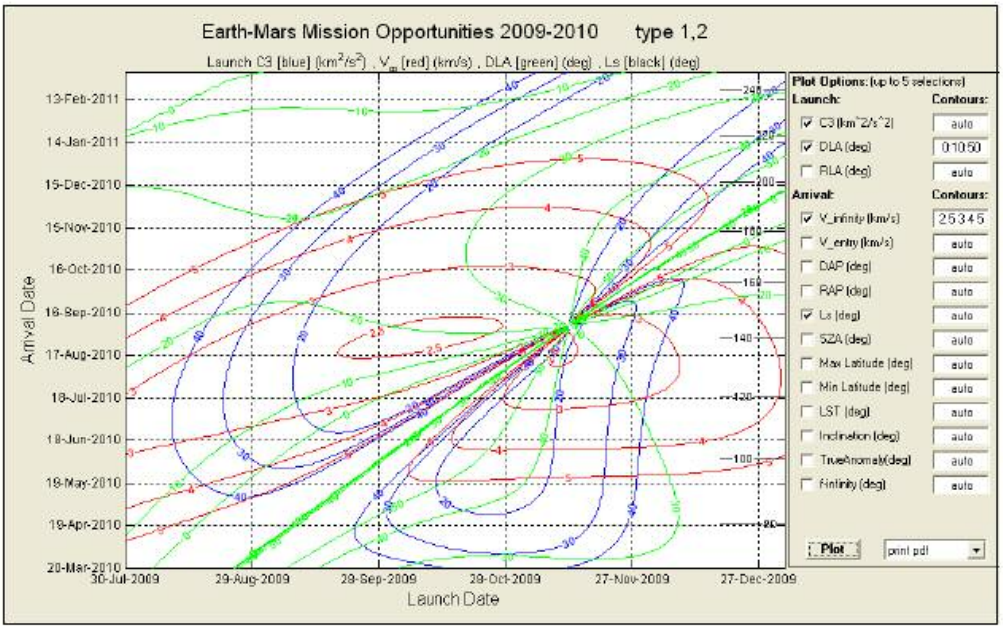

Figure 5 Sample Porkchop Plot

3.5.1 Data Output

Data files are output to the Current Working Directory. The three data files output is the ASCII bplane text file, ASCII Cartesian text file, and MATLAB data file. All three files are made readily available for use and editing by the user.

3.5.2 Plot Output

Figures are output to graphics files located in the Current Working Directory. For each plot produced in the plotting tool, a corresponding file will be saved as figure#.*format* where # is the corresponding successive number of plots being generated (i.e. – 1, 2, 3, ... , etc.) and *format* is the type of graphics file selected for output (i.e. – jpg or pdf).

References

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

[2] Vallado, D.A., **Fundamentals of Astrodynamics and Applications: Second Edition**, Space Technology Series, The McGraw-Hill Companies, Inc., 2001

[3] Seidelman, K. **Explanatory Supplement to the Astronomical Almanac**, University Science Books, Sausalito, CA, 1992.

[4] Highsmith, D. "Mars Reconnaissance Orbiter, Planetary Constants and Models," JPL D-22685. July 7, 2005.

[5] Heafner, P.J., **Fundamental Ephemeris Computations**, Willmann-Bell, Richmond, VA, 1999.

[6] Sergeyevsky, A.B., Snyder, G.C., and Cunniff, R.A., "Interplanetary Mission Design Handbook, Volume I, Part 2," JPL Publication 82-43. September 15, 1983.

[7] Tolson, R.H., "Interplanetary Navigation and Guidance," University of Maryland Textbook. Spring, 2004.

[8] Battin, R.H., **An Introduction to the Mathematics and Methods of Astronautics**, AAA Education Series, 1987.

[9] Duxbury, T.C., Kirk, R.L., Archinal, B.A., and Neumann, G.A., "Mars Geodesy/Cartography Working Group Recommendations on mars Cartographic Constants and Coordinate Systems," Symposium on Geospatial Theory, Processing and Applications, Ottawa, 2002.

9.3 Appendix C – Mars Science Laboratory POST 2 Input Deck

```
*no seq

problem = 1

c ------------------------------

c  Macro Names

c ------------------------------

*declare GC     9.8066

*declare LANDSITERAD 3394987.7972110      /m

*declare DIAM 3.75                  /m

*declare TARGLAT 0.33               /deg

*declare TARGLON 46.19;             /deg

*declare HEATSHIELD 290             /kg

*declare BACKSHELL 122.85           /kg

*declare SUPCTDIA 16.15             /m

*declare SUPCTMASS 55               /kg


 ndepv  = 4 /2

 indxi  = 1,2,3,4,

 depvr( 1) = gclat         long             gcrad            dgenv2

 depph( 1) =   840          840.0            840                200

 deptl( 1) =   0.2            0.2            0.5             0.00001

 depval(1) = {TARGLAT}     {TARGLON}     {LANDSITERAD+22}        0

 nindv     = 4

 indxi     = 1,2,3,4

 indvr( 1) =   bpang   timeo     gamei         critr

 indph( 1) =     1.0     1.0       1.0         800.0

 indvmn(1) =    -180  -50000     -20.0          25.0

 indvmx(1) =     180   50000      -8.5         150.0
```

```
 pert(  1) =     1.0   500.0      -0.5          -1.0

 indval(1) =     -1 {-25000+240}   -14            80

  vehicle = 1

cdww ------------------------------------------

cdww   ###### EVENT 1 #######

cdww   Initial Event

cdww   Cruise Stage Separation (EI-10 min)

cdww ------------------------------------------

  event = 1,

      title(50:99) = 'Exo-atmospheric',

      dof     =  3,

      ioflag  =  3,                 // Metric-Metric units

      npc(1)  =  1,                 // Calculate Conics and print

      npc(2)  =  1,                 // Runge Kutta integration

      fesn    =  999,               // final event number

      pinc    =  10,                // print interval

      npc(40) =  3,                 // Calculate 'RV' angles

      genvlim =  12                 // need this if genv table go past2

      maxtim  =  1500,

      altmax  =  1.0e+09,

      altmin  = -4.0e+3,

      lsflag    = 1

      lstflag   = 2

      nspc(21)  = 2


      // ----------------------

      //  Mass Properties

      // ----------------------

      npc(30) = 0,                  // N-stage weight model
```

```
      npc(9)  = 0,                  // Rocket engines
*     include '../includes/mass_props.dat' // MSL-0407, mass = 1883
  pi1t = "pi1t" constant 0 180.0
  tvc1m = {6*(cosd(25))}, 'one'
  tvc1t = 'TD_thr' constant,0, 3000.00,
      // -----------------------
      //  Planet Model
      // -----------------------
      npc(16) = 5,            // 85x85 Mars gravity model (MGS85F2)
      gravdata='/app/production/mars/grav_data/jpl85x85'
*       include '../includes/mars_constants.dat'
      // -----------------------
      //  Atmosphere inputs
      // -----------------------
*       include '../includes/mars_gram.dat'//Mars Gram Inputs for later
      npc(5)  = 0,                  // No Atmos -- exoatmospheric
      npc(6)  = 0,                  // No winds -- exoatmospheric
      npc(15) = 0,                  // No aeroheating -- exoatmospheric

      // -----------------------
      //  Aerodynamic inputs
      // -----------------------
      npc(8)  = 0,                  // No Aero -- exoatmospheric
      itrim = 0,              // No trim
      lref  = DIAM,           // Reference length (m)
      drefr = DIAM,           // Reference length (m)
      drefp = DIAM,           // Reference length (m)
      drefy = DIAM,           // Reference length (m)
      sref  ={DIAM*DIAM*pi/4},      // Reference area (m^2)
```

```
// ------------------------------------
//   Initial Conditions
//   NAV Filter & State Initialization
// ------------------------------------
npc(3)    =  1,1,       // Cartesian States
xi      =  0,0,0,       // Ensure milestone bombs
iguid(12) =  1          // Initialize attitude with aero angles
   alpha  = -15,
   beta   =  0.0,
   bnkang =  0.0,
   alppc(1) = -15,
   betpc(1) =  0.0,
   bnkpc(1) =  0.0,
iguid(22) =  0          // Intialize with body attitude rates
   rolbd  =  0.0,
   pitbd  =  0.0,
   yawbd  =  0.0,
inav    =  4,           // Using FSW NAV filter

azl=0,latl=0,lonl=0,// Align I- and L-frames
// ------------------------------------
//   Guidance
// ------------------------------------
  iguid(1)  =  0,0,1,         // 3-DoF:  Aero angle polynomials
  iguid(14) =  0,// No POST guidance -- Using FSW Apollo Guidance
  nspc(20)  =  0, // don't use velocity trigger for parachute
  spcv(4)   =  3.e6// Effectively NO supersonic parachute floor
  npc(12)   = 1,    // use relative great circles for xrng & drng
```

```
        nspc(88)  = 1         //Used as flag for output vars at epoch
*       include '../includes/trim_alpha.dat'// Alpha Command -> genv6
*       include '../includes/targ.dat'    // parachute deploy target
      // ------------------------------------
      //   GN&C FSW BUS
      // ------------------------------------
        hz_fsw_bus = 500,
           sf_frame_mode = 0,             // hold current frame
           sf_targ_mode = 0,              // hold current target
           tercorr_mode = 0,  // no terrain correction
        hz_sens = 0,
           imuavg_mode = 0,               // use IMU averaging
        hz_nav  =  50,
            navigation_mode = 0,          // JPL Kalman Filter
        hz_guid =  0,
           guidance_mode = 0,             // unguided exoatmospheric
           bank_cmd = {-70.0*3.14159265/180}, // pre-bank command
        hz_ctrl =  0,
           controller_mode = 0,           // 6-DoF LQR Entry Controller
           pitratefdbk_flag = 0,
           yawratefdbk_flag = 2,
           alpest_flag = 0,
        hz_act  = 0,
      // ------------------------------------
      //  Set Integration Variables
      // ------------------------------------
            npc(24)   = 1          // Use general integration
        gderv='gvrc26', gint =0,   // aeroheating indicator
      // ------------------------------------
```

```
      //  Gamma table
      // ------------------------------------
 genv11t = "initial inertial vel" monovar veli 0 lin_inp xtrap
         0,  0
        1e6 1e6

 genv3t = "genv3t" monovar gamei 0 lin_inp noxt    /current gamirv
       -1e6    -1e6
        1e6     1e6

 genv4t = "genv4t" monovar genv11 0 lin_inp noxt   /entry gamma(veli)
         5500                  -13.45
         6000                  -13.40
         6500                  -13.35
         7000                  -13.30

cdww ------------------------------------------
cdww   ###### EVENT 100 #######
cdww    holds initial inertial velocity
cdww ------------------------------------------

   event = 100,
     critr = tdurp, value = 0, mdl = 1,
 genv11t = "genv11" monovar genv11 0 lin_inp xtrap
    0,    0
   1e6    1e6

cdww ------------------------------------------
cdww   ###### EVENT 200 #######
```

```
cdww       Entry Interface
cdww       Activate Atmosphere
cdww       Turn on Apollo Guidance
cdww ---------------------------------------
  event = 200,
    critr = 'gcrad', value = 3522200, mdl = 1,
      title(50:99) = 'Hypersonic Entry'
      npc(8)      = 5,               // user aero subroutine
      npc(15)     = 1,               // turn on Aeroheating calc
      heatsg  = 1.90270e-04,         // Sutton-Graves constant
      rn    = 1.112,           // M. K. Lockwood, 13 Feb 2003
      npc(6)      = 0,               // No winds
      npc(5)      = 13,              // use Mars-GRAM 2001
      itrim = 3,               // 3-DoF: trim
      npc(10)     = 5,               // 3-DoF: calc c.g. for trim
      guidance_mode = 0,             // Apollo Entry Guidance

      // ----------------------------------------
      //  set monitor variables
      // ----------------------------------------
         monx( 1)  = 'dynp',
         monx( 2)  = 'asmg',
         monx( 3)  = 'heatrt',
         monx( 4)  = 'rdot',
 monx( 8)='gvrc26', /monitor max LAURA heat rate indicator
 mony( 8)= time,time
 monx( 9)='gvrc27', /monitor max REtheta indicator
```

```
 genv2t = "genv2t" multi genv11 0 lin_inp noxt   / bank angle profile
vs velr
         6000                        1
         7000                        2
 genv2t = "entry vel=6000" monovar velr 1 lin_inp xtrap
         0.0                      0.939692621   /20 deg
         900.0                    0.939692621
         900.1                    0.707106781   /45 deg
         2000.0                   0.707106781
         5500.0                   0.258819045   /75 deg
         11000                    0.258819045
 genv2t = "entry vel=7000" monovar velr 2 lin_inp xtrap
         0.0                      0.939692621   /20 deg
         900.0                    0.939692621
         900.1                    0.707106781   /45 deg
         3500.0                   0.573576436   /55 deg
         3900.0                   0.173648178   /80 deg
         11000                    0.173648178


  igf(3) = 1            /nspc(90) = 1


cdww ------------------------------------------
cdww   ###### EVENT 300 #######
cdww      Supersonic Parachute Deploy
cdww         Fire Parachute Mortar Pyro
cdww ------------------------------------------


   event = 300,
     critr = mach, value = 2, mdl = 1,
```

```
 monx( 7)='mach',
 mony( 7)='dynp',
 npc(6)=2,
 dtimr(3) = 1
 timrf(3) = 0
 igf(4) = 1              /nspc(91) = 1


cdww ------------------------------------------
cdww   ###### EVENT 400 #######
cdww      Supersonic Parachute Deploy
cdww      Parachute Line Stretch Phase
cdww ------------------------------------------
   event = 400,
     critr = 'times', value = 0.0, mdl = 1,
  npc(32)=1,
     parif=0,
     drgpk(1)=1.0,
     diamp(1)=SUPCTDIA,
 spci(81)=30.73                          / combined line length
(ls+lb+lr)
 spci(82)={SUPCTDIA*1.7}                    / suspension line length
(ls)
*include '../includes/chuteaero_msl_v1.dat'
 timrf(1)=0,
 dtimr(1)=1,
    itrim=0,
 nspc(22)=1   /turn on inflation model-time btwn mortar fire and line
stretch
 spci(80)=0
```

```
 alppc(1)=0.0,betpc(1)=0.0,bnkpc(1)=0.0,
 dt = 0.05,

cdww ------------------------------------------
c      ###### EVENT 410 #######
c          Supersonic Parachute Deploy
c *        Line Stretch Complete
cdww ------------------------------------------
 event=410,critr='time',valnam=gvrc36,mdl=8,
  nspc(22)=2   / inflation model - time btwn line stretch and peak load
  dt=0.005,

cdww ------------------------------------------
c      ###### EVENT 420 #######
c          Supersonic Parachute Deploy
c *        parachute Fully Inflated
cdww ------------------------------------------
 event=420,critr='time',valnam=gvrc34,mdl=8,
c full deploy after peak load
 dt = 1  /0.1
 nspc(22)=-1               / turn off inflation model and calculate
peak load


cdww ------------------------------------------
cdww   ###### EVENT 500 #######
cdww      Heatshield Jettison
cdww      Fire Heatshield Pyro
cdww ------------------------------------------
```

```
cinp ---- for input deck control of event:
 heatshield_active = 0,                    // de-activate CinC event
monitor
 event = 500,                              // prior to this event
   critr = mach, value = 0.8, mdl = 1,
     wjett = {HEATSHIELD * GC},     // OptionM2-System_Parameter 1.xls
     nspc(51)=1,                    // use terminal descent aero
     dtimr(2)=1,
     timrf(2)=0.0,

   igf(5) = 1                /nspc(93) = 1

cdww ---------------------------------------------
cdww   ###### EVENT 800 #######
cdww     Terminal Descent Engine Start
cdww     Begin 2-second Warm-up     at 25%
cdww     Fire Engine Pyro
cdww ---------------------------------------------
  event = 800,
    critr = tdurp, value = 90, mdl = 1,
      title(50:99) = 'Descent Engine Warm-up'
      weicon= 0.0
      etapc =  0.25       / set engines to 25% throttle setting
      iwdf      = 3
      neng      = 1
      npc(  9)  = 1
      npc( 22)  = 2
      dtimr(4)=1, timrf(4)=0,
      monx( 6)='hgtagl',
```

```
      monx(7)= '', '', '',
      monx(10)='etapc1', /monitor average throttle setting
      npc(9)=1,
      nspc(1)=1, /flag to indicate engine started
      spci(51)=0,
      spci(52)=0,
      spci(53)=0,
      iguid=9,0,1,
      alppc(1) = 0, betpc(1) = 0,bnkpc(1) = 0
      gvri(2)=1,  / initialize prop (with or without trim)
      gvri(3)=0,  / don't use prop trim
      sf_frame_mode = 0,              // hold current frame
      tercorr_mode = 0,               // no terrain correction

cdww ----------------------------------------
cdww   ###### EVENT 810 #######
cdww     Jettison Supersonic parachute and Backshell
cdww     Fire Backshell Pyro
cdww ----------------------------------------
cinp ---- for input deck control of event:
  land_sep_active = 0,                 // de-activate CinC event
monitor
  event = 810,                    // prior to this event
      critr = 'timrf4', value = 1.999, mdl = 8,
        backshell_pyro = 3,           // set pyro to appropriate value
        land_sep_event = 3,           // set pyro to appropriate value
   dt = 0.1
    title(50:99) = 'Terminal Descent'
      npc(32)=0,
```

```
      wjett = {(SUPCTMASS+BACKSHELL)*GC}    / {177.85 * GC},
      nspc(1)=2, /flag to indicate parachute sep, unlock throttle
      controller_mode = 0,    // POST uncontrolled
      etapc  = 0.8
    depvrs    = velrdt
    igsai     = 2
    indvrs    = etapc1
    maxits    = 10
    ndepvs    = 1
    us        = 0.5
    perts=.2
    deptls=.001
      dtimr(3) = 0            //time on supersonic parachute
      timrf(3) = 0
   nspc(41)=1      / Use max throttle limit in GSA
   gvrc(93)=0.8    / Max throttle limit for eta in GSA
   depvls1t = "depvls1t" monovar velr 0 lin_inp xtrap
         0.0                      -6.0
         10.0                     -6.0
         10.1                     -6.0
         300                    -6.0

cdww ------------------------------------------
cdww   ###### EVENT 840 #######
cdww     Start of Skycrane Phase,
cdww     Disable Center Engines
cdww     Begin GSA Descent
cdww ------------------------------------------
  event = 840,
```

```
     critr = 'wr', value = 3.0, mdl = 1, tol = 0.01
    dt = 0.1
    title(50:99) = 'Skycrane Phase'
       guidance_mode = 0,      // Turn off JSC Terminal Guidance
       controller_mode = 0,// Turn off POST Terminal Descent Controller
       rover_pyro=3,
       rover_sep_event=3,
      etapc     = 0.25
     tvc1m = {4*(cosd(25))}, 'one'
c    General Data for Vehicle 1 in Problem 1 and Event  85.000:
     depvrs    = velrdt
     igsai     = 2
     indvrs    = etapc1
     maxits    = 10
     ndepvs    = 1
     us        = 0.5
     perts=.2
     deptls=.001
    nspc(40)=1      / Initialize Skycrane descent phase
    nspc(41)=1      / Use max throttle limit in GSA
    gvrc(93)=0.8    / Max throttle limit for eta in GSA
    gvrc(90)=0.75  / target relative velocity in constant velocity phase
    gvrc(91)=10     / altitude to start constant velocity phase
    gvri(30)= 12     / delta altitude for skycrane approach phase
     iguid(1)  =  9,0,1, // 3-DoF:  Relative Aero angle polynomials
     alppc(1)=0, betpc(1)=0,bnkpc(1)=0,

cdww ----------------------------------------
cdww   ###### EVENT 845 #######
```

```
cdww    Constant Vel phase
cdww ---------------------------------------------
  event = 845,
   critr = dgenv4, value = 0, mdl = 1, tol = 0.01
  nspc(40) = 0
  depvls1t = "depvls1t" monovar velr 0 lin_inp xtrap
         0.0                      1.0
         0.75                     0.0
         1.75                    -1.0


cdww ---------------------------------------------
cdww   ###### EVENT 850 #######
cdww     Rover Touchdown
cdww     First Contact
cdww ---------------------------------------------
  event = 850,
   critr = dgenv4, value = 5, mdl = 1, tol = 0.01


   igf(6) = 1                /nspc(95) = 1


cdww ---------------------------------------------
cdww   ###### EVENT 999 #######
cdww
cdww ---------------------------------------------
  event = 999,
   critr = tdurp, value = 0.0, mdl = 1
```

References

[1] Battin, R.H., **An Introduction to the Mathematics and Methods of Astronautics**, AIAA Education Series, 1987.

[2] Standish, E.M. "Time scales in the JPL and CfA ephemerides", Astronomy and Astrophysics. 336, 381-384 (1998).

[3] Seidelman, K. **Explanatory Supplement to the Astronomical Almanac**, University Science Books, Sausalito, CA, 1992.

[4] Meeus, Jean. **Astronomical Algorithms**, Willmann-Bell, Richmond, VA, 1999.

[5] Bate, R.R., Mueller, D.D., and White, J.E., **Fundamentals of Astrodynamics**. Dover Publications, 1971.

[6] Thorne, James D., and R. D. Bain. "Series Reversion/Inversion of Lambert's Time Function." Journal of the Astonautical Sciences. 1995.

[7] Prussing, J.E. and Conway, B.A. **Orbital Mechanics,** Oxford University Press, New York, New York, 1993.

[8] Gedeon, G.S. "A Practical Note on the Use of Lambert's Equation," AIAA Journal, January 1965.

[9] T.C. Van Flandern and K.F. Pulkkinen. "Low precision formulae for planetary positions", ApJ Supp. 41, 391-411 (1979).

[10] George, L.E. and Kos, L.D. "Earth-to-Mars Mission Opportunities and Mars-to-Earth Return Opportunities 2009-2024," NASA TM/-1998-208533. July, 1998.

[11] Tolson, R.H., "Interplanetary Navigation and Guidance," University of Maryland Textbook. Spring, 2004.

[12] Cianciolo, A., Powell, R., and Lockwood, M.K. "Mars Science Laboratory Launch-Arrival Space Study: A Pork Chop Plot Analysis," IEEE paper, 2006.

[13] Heafner, P.J., **Fundamental Ephemeris Computations**, Willmann-Bell, Richmond, VA, 1999.

[14] Highsmith, D. "Mars Reconnaissance Orbiter, Planetary Constants and Models," JPL D-22685. July 7, 2005.

[15] Sergeyevsky, A.B., Snyder, G.C., and Cunniff, R.A., "Interplanetary Mission Design Handbook, Volume I, Part 2," JPL Publication 82-43. September 15, 1983.

[16] Gambis, Daniel. *International Earth Rotation and Reference Systems Service (IERS) Bulletin C*. http://hpiers.obspm.fr/iers/bul/bulc/bulletinc.dat

[17] Duxbury, T.C., Kirk, R.L., Archinal, B.A., and Neumann, G.A., "Mars Geodesy/Cartography Working Group Recommendations on mars Cartographic Constants and Coordinate Systems," Symposium on Geospatial Theory, Processing and Applications, Ottawa, 2002.

[18] Vallado, D.A., **Fundamentals of Astrodynamics and Applications: Second Edition**, Space Technology Series, The McGraw-Hill Companies, Inc., 2001

[19] IAU 2006 Resolution 3. *Re-definition of Barycentric Dynamical Time, TDB.* http://www.iau.org/fileadmin/content/pdfs/IAU2006_Resol3.pdf

[20] Central Bureau. *The International Celestial Reference System (ICRS).*. http://www.iers.org/MainDisp.csl?pid=96-107

[21] Powell, R. W., Striepe, S. A., Desai, P. N., Queen, E. M., Tartabini, P. V., Brauer, G.L., Cornick, D. E., Olson, D. W., Petersen, F. M., Stevenson, R., Engle, M. C., Marsh, S. M., "Program to Optimize Simulated Trajectories (POST2), Vol. II Utilization Manual." Version 1.1.6G, January 2004, NASA Langley Research Center, Hampton VA.