Feasibility of reusable continuous thrust spacecraft for cargo resupply missions to Mars

by

C. B. Rabotin

M.S., ESIEE Paris, 2011

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Masters of Science in Aerospace Engineering Sciences

Ann and H.J. Smead Aerospace Engineering Sciences

2017

This thesis entitled:
Feasibility of reusable continuous thrust spacecraft for cargo resupply missions to Mars
written by C. B. Rabotin
has been approved for the Ann and H.J. Smead Aerospace Engineering Sciences

_____

Dr. Hanspeter Schaub

_____

Dr. Natasha Bosanac

_____

Dr. Jay McMahon

Date _____

The final copy of this thesis has been examined by the signatories, and we find that both the
content and the form meet acceptable presentation standards of scholarly work in the above
mentioned discipline.

Rabotin, C. B. (M.Sc., Aerospace)

Feasibility of reusable continuous thrust spacecraft for cargo resupply missions to Mars

Thesis directed by Dr. Hanspeter Schaub

Continuous thrust propulsion systems benefit from a much greater efficiency in vacuum than chemical rockets, at the expense of lower instantaneous thrust and high power requirements. The satellite telecommunications industry, known for greatly emphasizing heritage over innovation, now uses electric propulsion for station keeping on a number of spacecraft, and for orbit raising for some smaller satellites, such as the Boeing 702SP platform. Only a few interplanetary missions have relied on continuous thrust for most of their mission, such as ESA's 367 kg SMART-1 and NASA's 1217 kg Dawn mission.

The high specific impulse of these continuous thrust engines should make them suitable for transportation of heavy payloads to inner solar system destinations in such a way to limit the dependency on heavy rocket launches. Additionally, such spacecraft should be able to perform orbital insertions at destination in order to deliver the cargo directly in a desired orbit. An example application is designing round-trip missions to Mars to support exploration and eventually colonization.

This research investigates the feasibility of return journeys to Mars based on the performance of existing or in-development continuous thrust propulsion systems. In order to determine the business viability of such missions, an emphasis is made on the time of flight during different parts of the mission, the relative velocity with respect to the destination planet, and the fuel requirements. The study looks at the applicability for interplanetary mission design of simple control laws for efficient correction of orbital elements, and of thrusting purely in velocity or anti-velocity direction. The simulations explore different configurations of continuous thrusting technologies using a patched-conics approach. In addition, all simulation scenarios facilitate escape from planetary gravity wells as the initial spacecraft orbit is highly elliptical, both around the Earth and around

Mars. This work does not include any optimal trajectory design. For this research, a highly configurable orbit propagation software with SPICE ephemerides was developed from scratch in Go, a modern compiled computer language.

The outcome of this research is that simple orbital element control laws do not lead to more efficient or faster interplanetary transfers. In addition, spiraling out of Earth's gravity wells requires a substantial amount of time despite starting from a highly elliptical orbit, and even with clustered high thrust engines like the VASIMR VX-200. Further investigation should look into hybrid solutions with a chemical engine for departing Earth; outbound spirals from Mars take a more reasonable amount of time.

Dedication

To my father, whom I miss dearly, and to the women of my life, my mother and my sister, for their unconditional love and support;

To my friends, for the fascinating and humorous discussions;

To Naveen Jain, for his advice at TVC 2014;

To Elon Musk, a source of inspiration since 2003.

Acknowledgements

I would like to acknowledge Dr. Hanspeter Schaub, my advisor, for agreeing to guide this research, for his insights, pertinent questions and his patience throughout my thesis. I also extend my gratitude to Dr. Natasha Bosanac and Dr. Jay McMahon for their feedback and help on this manuscript.

I would also like to thank everyone of the AVS Lab, for the interesting discussions on engineering, history, or politics. It has been an honor to spent time with such a brilliant crowd.

Contents

Tables

Table

Figures

Figure

Chapter 1

Introduction

Will man ever go to Mars? I am sure he will—**Wernher von Braun, 1954** [42]

1.1   Motivation

As soon as the 19th century, science fiction made some rough attempts at designing missions to the Moon, such as in "From the Earth to the Moon" by Jules Vernes, published in 1865. Later, as spaceflight was seemingly leaving the realms of science fiction to join those of reality, visionary aerospace engineers, like Wernher von Braun and a team at Lewis Research Center, started designing endeavors to the Moon and Mars. [42, 60]

> To seize this opportunity, I'm not proposing a 10-year plan like Apollo; I'm proposing a long-range, continuing commitment. (...) And then a journey into tomorrow, a journey to another planet: a manned mission to Mars. — **Georges H. W. Bush, 1989** [11]

In 1989, Georges H. W. Bush as President of the United States set the stage for a manned mission to Mars through the Space Exploration Initiative. Proposals for such missions, such as Zubrin's "Mars Direct" [62], started to emerge as soon as the following year. However, NASA Administrator Dan Goldin, appointed in 1992, shifted the Space Exploration Initiative to a more realistic plan given the know-how: focus on robotic missions to Mars in order to better know Mars prior to a human mission. As Administrator, he also led a 10 % decrease in the human spaceflight budget due in part to a 40 billion dollar reduction from prior budget plans. [22] Since then, the budget of a human mission to Mars has been one of its key limitations. In fact, NASA's 2009 Design Reference

Mission version 5 mentions the importance of the sustainability of the NASA budget to support human missions to Mars. [2, 20] This reference mission proposes using a total of eight launches, cf. Figure 1.1.

> If one can figure out how to effectively reuse rockets just like airplanes, the cost of access to space will be reduced by as much as a factor of a hundred. A fully reusable vehicle has never been done before. That really is the fundamental breakthrough needed to revolutionize access to space. — **Elon Musk** [5]

Although made popular by Elon Musk and SpaceX starting 2011 via their Grasshopper program [5, 55], re-usability of launchers is also being investigated by several companies including Airbus Defense and Space, cf. Figure 1.2, Blue Origin and ULA. [48, 54] A reusable launcher would allow its manufacturing costs to be amortized over several missions. Re-usability of spacecraft bound to Mars is not a novel idea either. In fact, Dr. Buzz Aldrin envisioned the use of cyclic trajectories between the Earth and Mars in 1985. [3] Aldrin-cyclers present key challenges, such as the necessity to perform a hyperbolic rendezvous with the cycler, which is a hazardous operation. [4, 26] Another architecture could include splitting the journey as launch segment and space segment. Such is the vision shared by both SpaceX, with their Interplanetary Transportation System, and ULA with their cis-lunar architecture [30, 56].

1.2   Overview of propulsion

Specific impulse, I_{sp}

The specific impulse is one of the more important measures of thruster performance when operating in a vacuum. It corresponds to the total impulse of a thruster normalized over the mass of the propellant and as such is a measure of the efficiency of the thruster. [24, Chapter 18.1] Chemical propulsion can deliver low I_{sp} high thrusts for short durations. On the other hand, electric propulsion, a class of continuous thrust technologies, allows to achieve very high exhaust velocities while also reducing the total propellant mass. [49] Hence, the I_{sp} of electric propulsion thrusters is several times greater than that of chemical rockets, cf. examples in Table 1.1. The

Table 1.1: Propulsion characteristics of two high performance thrusters

	Type	Vacuum I_{sp} (s)	Nominal Thrust (N)	Mass (kg)
Safran Chemical 4N [28]	Bi-propellant	290	4	0.350
Snecma PPS NG [59]	Hall Effect thruster	1800	0.180	6

exhaust from electric propulsive thrusters is a stream of neutral atoms. Therefore, the exhaust mass is also orders of magnitude lower than that of chemical rockets and in turn, the thrust generated by electric propulsion is usually much smaller.

Chemical propulsion

The principle of chemical propulsion is to expand gases as high pressure through a converging-diverging nozzle. There are four main types of such propulsion. [24, Chapter 18.3]

- Cold gas thruster, which are operated by simply expanding a gas without combustion;

- Liquid propellant, either mono-propellant or bi-propellant where an oxidizer is used to ignite the combustion of the fuel;

- Solid propellant rockets, much simpler than their liquid propellant counterparts; and

- Hybrid thrusters, which combine a liquid fuel, usually as an oxidizer, and solid fuel.

Hall Effect Thruster

This is a class of gridded ion thrusters, which work by accelerating the ions between a source and a grid as depicted in Figure 1.3. The exhaust velocity of the charged atoms is proportional to the voltage difference between the electrodes. A stream of electrons is shot at the plume in order to avoid charging the spacecraft as these electrons depart from it. It was demonstrated in 2003 on mission to the Moon on the SMART-1 spacecraft. The thruster used on that spacecraft was a variable thrust, providing between 30 mN and 70 mN for a respective I_{sp} of 1100 s to 1600 s. [18] One of the advantages of the Hall Effect thruster compared to the broader gridded ion thrusters is its magnetic field which prevents erosion of the cathode. [50] In addition, this magnetic field is symmetric which allows for clusters of even numbers and original configurations, cf. Figure 1.4.

High powered Hall Effect thrusters, such as the Busek BHT-8000, the Safran/SNECMA PPS20k and NASA HERMeS, are currently in development specifically for exploration missions. [36, 63]

Pulsed Plasma Thruster

Pulsed Plasma Thrusters (PPTs) are the first form of electric propulsion flown: they were demonstrated on the 1964 Soviet spacecraft Zond 2. One of their main advantages is the use of an ablative solid propellant, leading to a simple thruster design. [33] NASA researchers think this is a promising technology for Cubesat orbit raising and station keeping, following a Cubesat demonstrator in 2000 which confirmed that the plasma did not interfere with any subsystem. [35]

VASIMR ®

The Variable Specific Impulse Magnetoplasma Rocket (VASIMR) is an experimental electric propulsion thruster in development by Ad-Astra whose operating principles are summarized in Figure 1.5. The VX-200 variant has produced in ground tests a thrust of 5.8 ± 0.4 N for an I_{sp} of 4900 ± 300 s when provided 200 kW of power. [53] The VASIMR engine requires a 2 Tesla magnetic field for operation, which would interact with planetary magnetic fields unless complemented with another VASIMR thruster with opposite polarization. This engine was planned to be demonstrated on the ISS, but was instead selected for NASA's NextSTEP program in the category of advanced electric propulsion. The goal of that program is to mature the thruster to a technology readiness level of 5 (out of a maximum of 9). [29]

1.3   Thesis research goals

Impulsive interplanetary missions spend the greater time of the transfer cruising to their destination, with little thrusting along the way, as discussed in chapter 2. Despite increased hazardous crew exposure to Galactic Cosmic Rays and potential Solar Proton Events, NASA's Mars Design Reference Mission, SpaceX's ITS and ULA's cis-lunar architecture all plan on using chemical propulsion. [2, Page 50] [30, 56, 61]

This research explores the feasibility of reusable semi-cyclers using continuous propulsion technologies between the Earth and Mars. A semi-cycler is herein defined as a spacecraft acting as

tug between two orbital destinations around two different planets. One of the key differences with standard interplanetary mission design is that the semi-cycler is launched without any payload. The vehicle awaits for payload docking in a parking orbit around Earth. The semi-cycler starts its journey to Mars only once all payload is docked. Similarly, upon arrival at Mars, the vehicle jettisons each payload at their respective desired parking orbit around Mars. It may also collect other spacecraft in orbit around Mars for a return trip to Earth. Finally, the semi-cycler embarks on a return continuous-thrust trajectory.

This research emphasizes the relative velocity at destination, the fuel usage and the duration of flight for each segment of the patched-conics. The relative velocity provides an indication concerning whether the vehicle can use its continuous thrust propulsion system in order to perform an orbital injection. A low relative velocity is desired due to the limited thrust of this kind of propulsion. Additionally, the thrusters analyzed in this research are more efficient in vacuum than chemical propulsion. Hence the interest in the fuel usage of for each trajectory segment analyzed. Finally, the latter parameter is related to the usefulness of continuous thrust propulsion to limit radiation exposure for crewed mission during interplanetary flight. First is analyzed the trade space of parking and departure orbits around both Earth and Mars. Subsequently, the planetary escape and orbital injection using continuous thrusting technologies is compared to using on chemical thrusting for these segments of the mission. An additional comparison is performed between an in-plane thrust in the velocity direction and a continuous efficient correction of orbital elements using two different control laws. This research does not provide any detailed mission design with planetary ephemerides, and excludes thrusting technologies which rely on experimental sources of power, such as space-ready nuclear power. However, different thruster configurations providing different thrust levels are compared on the basis of the aforementioned metrics for each segment of the mission. All the simulations were ran on custom software which was initially developed to better understand the implementation of control laws for efficient correction of orbital elements.
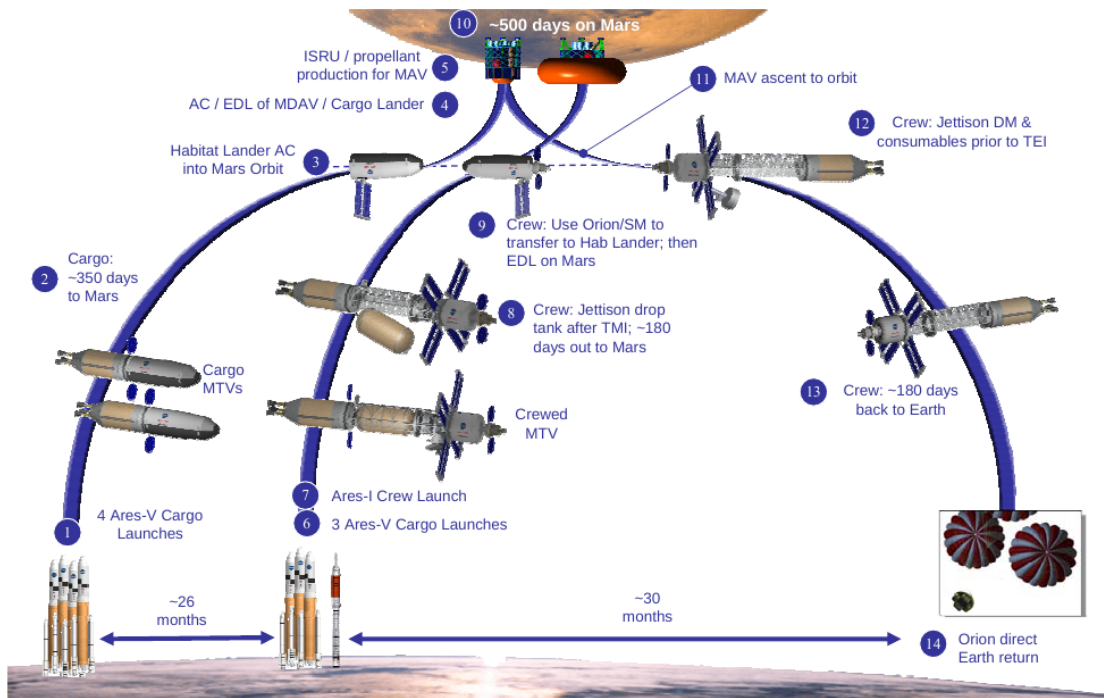
Figure 1.1: NASA Mars Design Reference Mission summary [20]
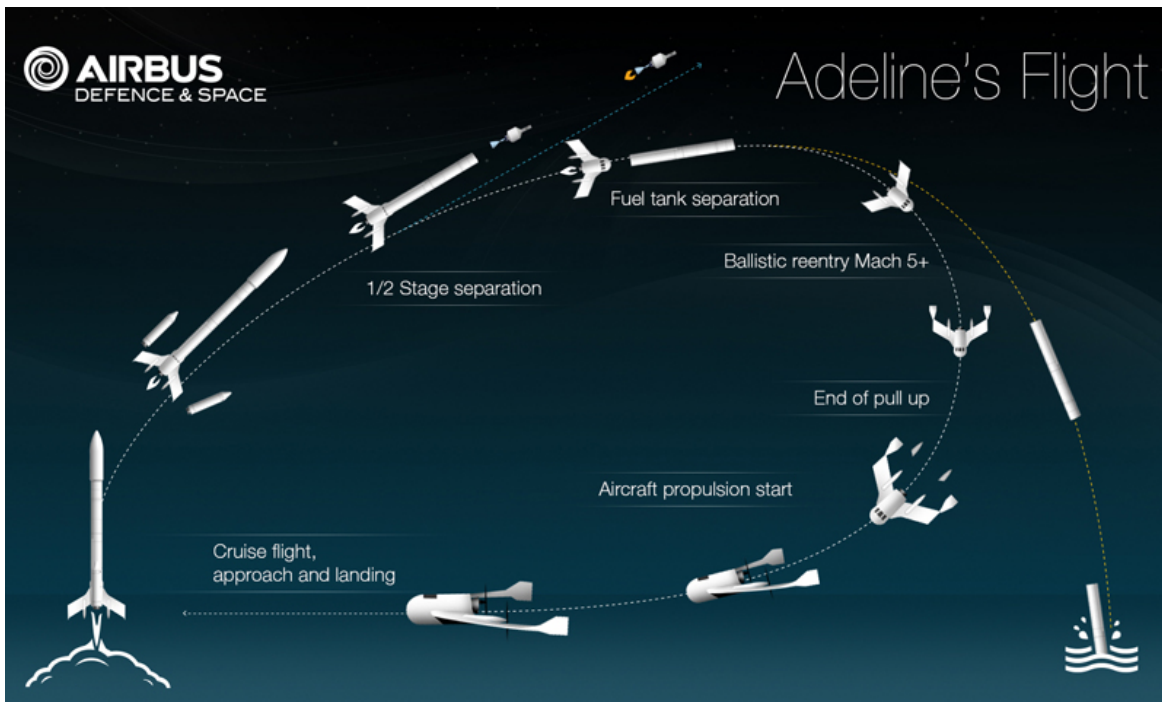
Figure 1.2: Airbus Defense and Space - Adeline's flight, an add-on to Ariane 6[17]

electron

neutral propellant atom

positive ion
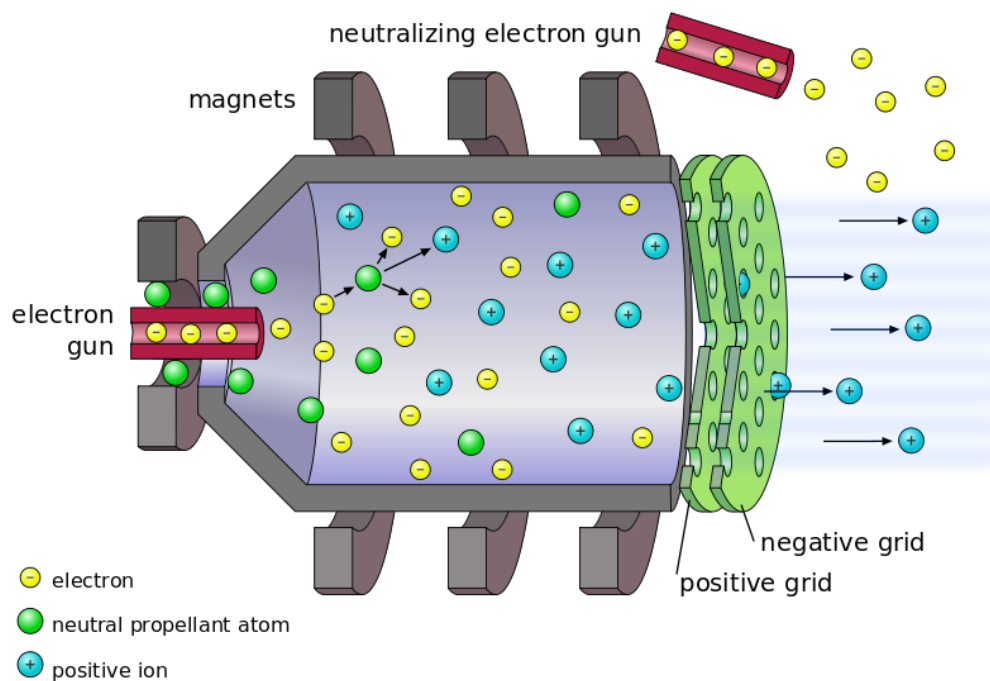
Figure 1.3: Electrostatic Ion Thruster [47]

Figure 1.4: Busek multiple cluster configurations [1]
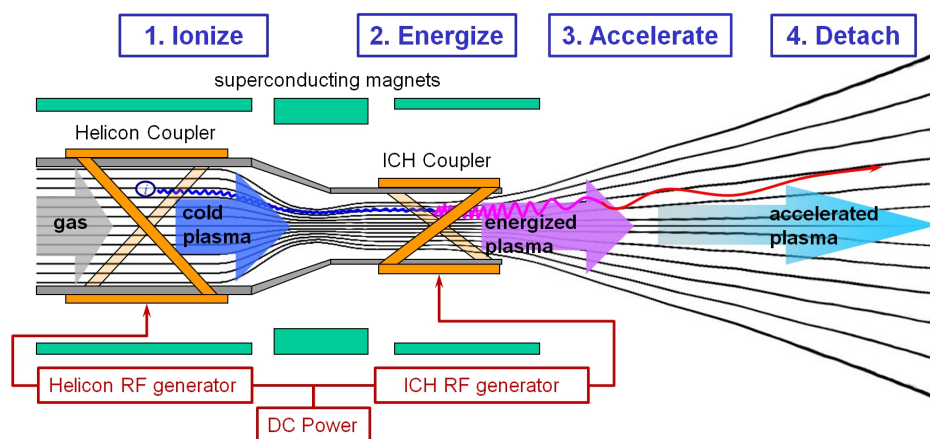
Figure 1.5: VASIMR ® operating principle [1]

Chapter 2

Overview of Interplanetary Trajectory Design with Impulsive Maneuvers

2.1 Introduction

Compared to Earth bound trajectories, interplanetary trajectory design must account for planetary ephemerides, making their design more complex. In addition to departing Earth's sphere of influence, another key challenge is accounting for celestial body perturbations, whose effects are proportional to the inverse square of the distance to the given celestial object. These perturbations vary over time based on the relative position of the spacecraft.

Apart from a few missions like NASA Dawn [25], the majority of spacecraft leaving the Earth's vicinity has relied on chemical propulsion. As discussed in the introductory chapter, chemical propulsion allows for large Δv changes with short duration burns. This enables mission designers to plan for the spacecraft to cruise during most of its mission and to perform a burn as specific points in time and space throughout the mission.

This chapter provides an overview of interplanetary trajectory design. First is discussed the relative two-body problem, with a definition of Keplerian orbital elements. Then, the equations of motion used in this research are detailed. Finally, two common approximations in broad trajectory design are explained: the sphere of influence and patched conics.

2.2 Relative two-body problem

According to Newton's universal law of gravitation, two masses m_1 and m_2, such that m_1 is the greater of both masses, separated by a distance r exert a force on each other of equal norm and

opposite direction. It is expressed as follows, where G is the gravitational constant, and $\boldsymbol{F_{i \to j}}$ the force exerted by object i on object j. Throughout this document, bold symbols represent vectors and normal typeface scalars: $|\boldsymbol{A}| = A$.

$$|\boldsymbol{F}_{1 \to 2}| = |\boldsymbol{F}_{2 \to 1}| = \frac{m_1 \, m_2}{r^2} G \tag{2.1}$$

$$\boldsymbol{F}_{1 \to 2} = -\boldsymbol{F}_{2 \to 1} \tag{2.2}$$

In the case where m_2 is negligible compared to m_1, and for unperturbed dynamics, the orbital motion is described by the following differential equation where $\boldsymbol{a}$ corresponds to the acceleration, $\boldsymbol{r}$ the vector from the planet to the object of negligible mass, and μ to the gravitational parameter of the central body ($\mu = G \, m_1$):

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3} \boldsymbol{r} \tag{2.3}$$

2.2.1   Keplerian orbital elements

Kepler discovered that the orbital motion of one body around another is an ellipse. Ellipses are defined as the intersection of the surface of a cone with a plane. Keplerian orbital elements allow to uniquely define an orbit. Figure 2.1 shows five of the six orbital elements, the semi major axis is missing. The semi major axis is the mean value of the furthest and closest points from a focus of the ellipse. Assuming unperturbed motion, the five of the six orbital elements of a point mass are time invariant: only the true anomaly varies with time. The orbital parameter notation used throughout this paper is the following:

- semi major axis: a

- inclination: i

- eccentricity: e

- right ascension of the ascending node: Ω

- argument of periapsis: ω

- true anomaly: ν

2.2.2 Equations of motion

The equations of motion are integrated as follows, where $T_{x,y,z}$ is the components of the thrust vector $\boldsymbol{T} = T\hat{\boldsymbol{u}}$ applied at each instance when applicable, and $\dot{m}$ is the mass flow rate. Moreover, throughout this research, a constant I_{sp} and constant thrust is assumed. In addition, I_{sp} corresponds to the specific impulse of the thrusters in use and g_0 is the standard acceleration due to gravity. The state vector $\boldsymbol{X}$, and its time derivative $\dot{\boldsymbol{X}}$, are expressed in an inertial frame.

$$
\boldsymbol{X} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ m \end{bmatrix}
\tag{2.4}
$$

$$
\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ -\frac{\mu}{r^3}x \\ -\frac{\mu}{r^3}y \\ -\frac{\mu}{r^3}z \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ T_x \\ T_y \\ T_z \\ -\frac{|\boldsymbol{T}|}{g_0 \times I_{sp}} \end{bmatrix}
\tag{2.5}
$$

Table 2.1: Sphere of influence of selected planets

Celestial body	Earth	Mars	Jupiter	Saturn
r_{SOI} (km)	$9.24 \cdot 10^5$	$5.76 \cdot 10^5$	$4.82 \cdot 10^7$	$5.46 \cdot 10^7$

2.3   Interplanetary trajectory design approximations

2.3.1   Sphere of influence

In the case of a spacecraft of negligible mass compared to that of any celestial body, the sphere of influence centered on a body 1 represents a region in space where the dominant force exerted on the spacecraft is the gravity of object 1. The radius of the sphere of influence (SOI), noted r_{SOI}, is approximated to the following:

$$r_{\mathrm{SOI}} = a \left(\frac{m}{M} \right)^{2/5} \tag{2.6}$$

Where a and m are the semi major axis and the mass of the least massive body respectively, and M the mass of the more massive object. Table 2.1 presents the radius of the sphere of influence of some celestial objects. The SOI provides a convenient approximation for interplanetary flight design. In fact, a common approach to preliminary interplanetary trajectory design involves assuming the spacecraft is strictly orbiting the departure planet up until its distance from that planet reaches the boundary of the sphere of influence. Then the spacecraft is assumed to be in a heliocentric orbit where only the Sun is exerting a perturbing force. Finally, as the vehicle enters the sphere of influence of the destination planet, it is assumed to only be orbiting that planet. This method is depicted in Figure 2.2.

2.3.2   Patched conics

The patched conics approximation ignores the celestial departure and arrival trajectories by assuming the spacecraft departs precisely at the position of the planet at the departure date and time, and arrives at the position of the arrival planet at the desired epoch. Hence, the patched

conics approximation reduces the trajectory design to solving a boundary value problem solely based on the positions of the planets at the expected departure and arrival dates and times.

Given two position vectors and a time of flight between these two points in space, Lambert showed in 1791 that there exists, in two-body dynamics, an elliptical orbit which connects both points. [57, Chapter 7.6] The solutions to this problem vary based on the initial conditions, the number of revolutions the spacecraft is allowed to make and whether to use a prograde or retrograde orbit. Lambert's boundary value problem is used in numerous mission design situations, including rendezvous.

In interplanetary mission design, given the initial and final radial vectors with respect to a celestial object and the transfer type, Lambert problem solvers enable calculation of the initial and final velocities needed to define a transfer orbit between both points in 3D space (i.e. completing the six degrees of freedom defining both points on the transfer orbit). These velocity vectors are then used to compute the hyperbolic excess velocity, or V_∞. It is defined as the velocity attained by a spacecraft relative to the velocity of the planet at the time of departure.

$$\boldsymbol{V}_\infty = \boldsymbol{V}_{\text{spacecraft/Sun}} - \boldsymbol{V}_{\text{Planet/Sun}} \tag{2.7}$$

For example, if a solution to Lambert's problem requires a departing velocity norm of $25\,\text{km/s}$ and the planet's velocity norm is $23\,\text{km/s}$, and both vectors are in the same direction, then the norm of the hyperbolic excess velocity is $2\,\text{km/s}$. The hyperbolic excess velocity corresponds to the desired spacecraft velocity at the planet's SOI boundary of specifically $2\,\text{km/s}$. The square of $|V_\infty|$, called C_3, is a key parameter when analyzing Earth's departure trajectory: most interplanetary missions rely on the launcher to provide that Earth departure velocity. In this research, the outward spiral trajectories out of gravity wells do not lead to hyperbolic orbits. Instead, the analysis is performed on the relative velocity with respect to the planet, as explained in further detail in section 5.3.
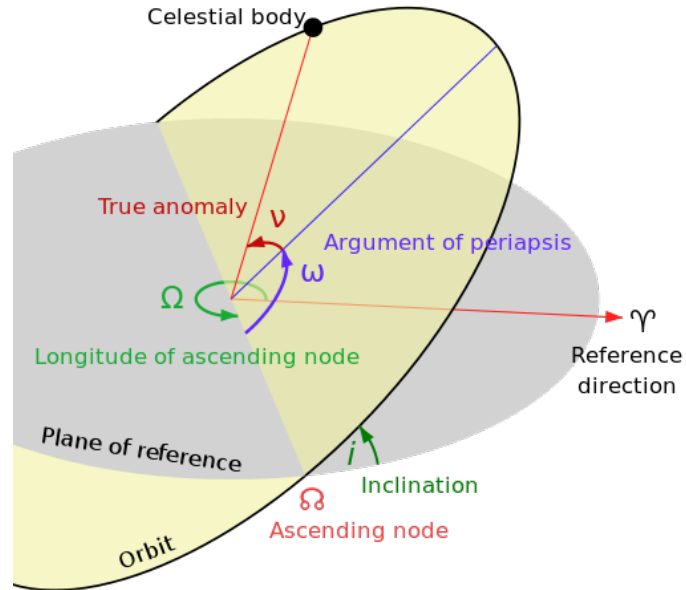
Figure 2.1: Keplerian orbital elements [27]
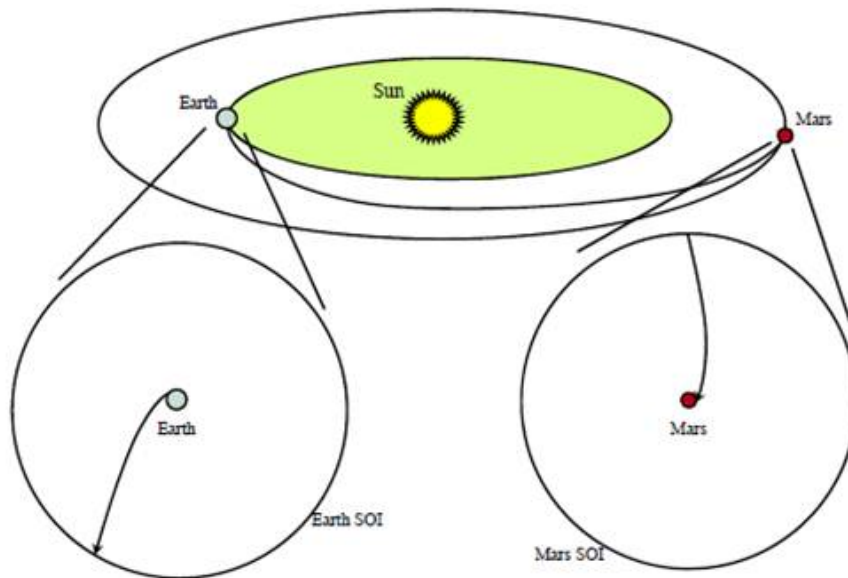
Figure 2.2: Example patched-conics Earth-Mars transfer [57, Chapter 12.2]

Chapter 3

Continuous thrust control laws

3.1   Introduction

As discussed in chapter 2, the standard approach of interplanetary missions involves a direct escape from Earth using the thrusting power of the rocket. Such methods are not applicable in the context of cyclers using continuous thrust. In fact, a more likely scenario is the docking of one or multiple spacecraft to the cycler in a conveniently located Earth orbit. The semi-cycler would only subsequently depart on its interplanetary mission. Therefore, the interest of studying control laws for continuous thrusting is relevant to the scenarios when the vehicle is orbiting a planet, which includes picking up or separating from cargo at specific orbits and preparing the outward spiral to optimize the relative velocity with respect to the given planet at the edge of the sphere of influence.

Hence, this chapter presents some control laws applicable to continuous thrusting for the efficient correction of orbital elements. A comparison of the application of each control law for orbits about Earth and Mars is also presented. All control laws presented rely on the same following principles, and differ only on the summation methods for correcting multiple orbital elements at the same time.

(1) All laws are applied starting an initial reference orbit;

(2) All laws propose a combination of in-plane and out-of-plane thrusting with respect to the orbital plane;

(3) All laws may share the same equations for efficient correction of each orbital element, and all of which are defined in the following section;

(4) For all laws, the thrust is applied until the difference between all osculating orbital element which needs correction is within a given tolerance of the desired orbital element;

(5) During the thrusting arc, some orbital elements may initially overshoot the desired orbital element prior to subsequently converging to the desired orbital element, herein referred to as a "targeted" orbital element, or "targeted" orbit.

Only the control laws implemented in the "space mission design" software library are detailed below. This library is detailed in Chapter 4. The Petropoulos control law, or Q-law, also handles efficient correction of orbital elements from the osculating elements. [39] It was not implemented due to its complexity. However, the Q-law test cases were used to validate the implementation of the other laws.

3.2   Efficient correction of orbital elements

The instantaneously optimal correction of orbital elements relies on in-plane or out-of-plane thrusting with respect to the orbital plane, as summarized in [51]. In the following, α and β respectively correspond to the in-plane and out-of-plane thrusting angles. Theses thrusting angles are converted to the radial-circumferential-normal (RNC) frame to determine the thrusting direction $\hat{\boldsymbol{u}}$ as follows:

$$
{}^{\mathcal{RNC}}\hat{\boldsymbol{u}} = \begin{bmatrix} \sin\alpha\cos\beta \\ \cos\alpha\cos\beta \\ \sin\beta \end{bmatrix} \tag{3.1}
$$

Herein, $\hat{\boldsymbol{u}}_\gamma$ will represent the thrust direction for the instantaneous correction of orbital element γ.

The laws, summarized in Table 3.1, rely on Keplerian orbital elements. These laws encounter singularities if the initial or target orbits are not elliptical about the main celestial object. In addi-

Table 3.1: Instantaneously optimal in-plane and out-of-plane thrust angles for maximum change of each orbital element [51]

Orbital element	In-plane angle (α)	Out-of-plane angle (β)
Semi major axis	$\alpha = \mathrm{atan}\, \frac{e\sin\nu}{1+e\cos\nu}$	$\beta = 0$
Eccentricity	$\alpha = \mathrm{atan}\, \frac{\sin\nu}{\cos\nu+\cos E}$	$\beta = 0$
Inclination	$\alpha = 0$	$\beta = \frac{\pi}{2}\,\mathrm{sgn}\left\{\cos\left(u\right)\right\}$
Right ascension of the ascending node	$\alpha = 0$	$\beta = \frac{\pi}{2}\,\mathrm{sgn}\left\{\sin\left(u\right)\right\}$

tion, no analysis was performed on the time variation of thrust vector direction: it is possible that the resulting thrust vector direction varies to the extend of rendering such control laws impractical in real world operations.

In Table 3.1, a corresponds to the semi major axis, e to the eccentricity, i to the inclination, Ω to the RAAN, ω to the argument of periapsis, and ν to the true anomaly. Additionally, r is the norm of the radius vector, E the eccentric anomaly, p the semi-parameter. The argument of latitude is also used for conciseness: $u = \omega + \nu$. Moreover, atan corresponds to the arctan function.

The argument of periapse may be changed via different formulations, as summarized in 3.2. The Ruggerio formulation performs simultaneously an in-plane and out-of-plane thrust. The Petropoulos formulation [39] performs either an in-plane or an out-of-plane thrust, depending on whichever is the most efficient. In equations 3.2 and 3.3, ν_α and ν_β correspond to optimum true anomaly ν to perform respectively an in-plane or an out-of-plane maneuver to correct the argument of periapsis to lead to the greatest change in that orbital parameter. As discussed in the implementation details in Chapter 4, the software developed for the thesis determines whether an in-plane or an out-of-plane maneuver to correct the argument of periapsis is the most optimal given the osculating orbital elements. If the osculating true anomaly is closer to the optimal true anomaly to perform an in-plane thrust, then an in-plane thrusting direction is applied. Conversely, if the

Table 3.2: Instantaneously optimal in-plane and out-of-plane thrust angles for maximum change of the argument of periapsis [39, 51]

Orbital element	In-plane angle (α)	Out-of-plane angle (β)
Argument of periapsis (Ruggiero)	$\alpha = \text{atan} \left\{ \frac{1+e\cos\nu}{2+e\cos\nu} \cot\nu \right\}$	$\beta = \text{atan} \left\{ \frac{e\cot i \sin(u)}{\sin(\alpha-\nu)(1+e\cos\nu)-\cos\alpha\sin\nu} \right\}$
Argument of periapsis (Petropoulos in-plane)	$\alpha = \text{atan} \frac{-p\cos\nu}{\sin\nu(p+r)}$	$\beta = 0$
Argument of periapsis (Petropoulos out-of-plane)	$\alpha = 0$	$\beta = \frac{\pi}{2} \text{sgn} \left\{ -\sin(u) \right\} \cos i$

true anomaly is closer to the optimal true anomaly for an out-of-plane thrust, then such a thrust direction is selected.

$$\cos\nu_\alpha = \left[\frac{1-e^2}{2e^3} + \sqrt{\frac{1}{4}\left(\frac{1-e^2}{e^3}\right)^2 + \frac{1}{27}} \right]^{1/3} - \left[-\frac{1-e^2}{2e^3} + \sqrt{\frac{1}{4}\left(\frac{1-e^2}{e^3}\right)^2 + \frac{1}{27}} \right]^{1/3} - \frac{1}{e} \quad (3.2)$$

$$\nu_\beta = \cos^{-1}\left\{ -e\cos\omega \right\} - \omega \quad (3.3)$$

3.3   Control laws

The Naasz and Ruggiero laws presented only differ on the summation method used to efficiently correct multiple orbital elements at the same time.

3.3.1   Naasz control

Naasz [21, 31] proposes an algorithm to simultaneously correct multiple orbital elements. The change in each orbital element is weighed and added to the overall instantaneous thrust. Correction of orbital elements are zeroed when computing the overall thrust if the osculating element is within an acceptable vicinity to its target.

It is important to note that the original formulation of these laws only allowed for an increase of each orbital parameter. For example, these laws would fail to efficiently correct an orbit from an eccentricity of 0.5 down to 0.2. The laws are corrected here by adding the $\text{sgn}\,\Delta\gamma$ to each weight.

Table 3.3: Naasz control weight computation

Orbital element	Weight
Semi major axis	$w_a = \operatorname{sgn}\left(\Delta a\right) \frac{h^2}{4a^2(1+e)^2}$
Eccentricity	$w_e = \operatorname{sgn}\left(\Delta e\right) \frac{h^2}{4p^2}$
Inclination	$w_i = \operatorname{sgn}\left(\Delta i\right) \left[\frac{h+eh\cos\left(\omega+\sin^{-1}(e\sin\omega)\right)}{p\left[(e\sin\omega)^2-1\right]}\right]^2$
Argument of periapsis	$w_\omega = \operatorname{sgn}\left(\Delta\omega\right) \frac{(eh)^2}{4p^2}\left(1-\frac{e^2}{4}\right)$
Right ascension of the ascending node	$w_\Omega = \operatorname{sgn}\left(\Delta\Omega\right) \left[\frac{h\sin(i)\left(e\sin\left(\omega+\sin^{-1}(e\cos\omega)\right)-1\right)}{p\left(1-(e\cos\omega)^2\right)}\right]^2$

In the following equation, w_γ represents the weight for the orbital parameter γ, $\delta_{i,j}$ the Kronecker delta function, γ_{osc} the osculating orbital element, γ_{tgt} the target orbital element. The direction of change in the orbital element is noted and defined as $\Delta\gamma = \gamma_{\mathrm{tgt}} - \gamma_{\mathrm{osc}}$.

$$\hat{\boldsymbol{u}}_{\mathrm{total}} = \sum_\gamma \left\{\left(1 - \delta_{\gamma_{\mathrm{osc}},\gamma_{\mathrm{tgt}}}\right)\left(\gamma_{\mathrm{osc}} - \gamma_{\mathrm{tgt}}\right)\right\} w_\gamma \left(\Delta\gamma\right)^2 \hat{\boldsymbol{u}}_\gamma \tag{3.4}$$

3.3.2 Ruggiero control

The Ruggiero control laws [51] allow for a simple algorithm based on the the generic correction of orbital element described in section 3.2.

In the following equation, γ represents one of the five relevant Keplerian orbital elements, $\delta_{i,j}$ the Kronecker delta function, γ_{ini} the initial orbital element, γ_{osc} the osculating orbital element and γ_{tgt} the target orbital element. $\hat{\boldsymbol{u}}_\gamma$ corresponds to the instantaneously optimal thrust direction needed for an efficient correction of orbital element γ. The Kronecker delta function allows to cancel the contribution of any $\hat{\boldsymbol{u}}_\gamma$ if the osculating value of the orbital element γ is equal to the desired value.

$$\hat{\boldsymbol{u}}_{\mathrm{total}} = \sum_\gamma \left(1 - \delta_{\gamma_{\mathrm{osc}},\gamma_{\mathrm{tgt}}}\right) \frac{\gamma_{\mathrm{tgt}} - \gamma_{\mathrm{osc}}}{\gamma_{\mathrm{tgt}} - \gamma_{\mathrm{ini}}} \hat{\boldsymbol{u}}_\gamma \tag{3.5}$$

3.4   Analysis

3.4.1   Implementation and validation

Validation of implementation was done by running the single and multiple orbital element correction tests cases from the papers by [37, 38, 40, 51]. These correspond to all the test cases available in these papers in which the spacecraft was orbiting Earth. Selecting all these test cases ensures thorough validation. In fact, these test cases are part of the test suite of the "space mission design" software library, as explained in Chapter 4. The equations of motion used for this validation are those described in section 2.2.2. In addition, also as explained in section 2.2.2, a constant thrust and constant I_{sp} is assumed. Unless specified otherwise, all work presented in this research uses a fixed time step of 10 seconds for the propagation. This time step was chosen after confirming that the variation in state propagation between a fixed one second time step and a fixed ten second time step was negligible.

In tables 3.5 and 3.6, cases A through D originate from the Petropoulos Q-Law paper [37]. Cases E through G originate from [51]: success in these test cases served as the primary validation method for the Ruggiero control laws. In these validation scenarios, a "free" parameters is defined as not having any constraint. For example, if only the eccentricity is set as a "free" parameters and the other orbital elements have an assigned numerical value, then the control law is requested to perform a correction of all the parameters at the same time, apart for the eccentricity. An "unknown" initial parameter means that the test case is only partially defined in the original paper. Case B' is a novel case similar to case B but the eccentricity is set as a free parameter. Cases E and F only perform a correction on one orbital element: I obtain strictly the same results for these cases, as expected since the Naasz and Ruggiero laws differ only in their summation for multiple orbital element correction. The convergence criteria for the test cases in this section is defined in table 3.4.

It is interesting to note that the Naasz and Ruggerio control laws lead to very similar results expected by the Q-Law since they lead to the same results as the validation cases of the Q-Law

Table 3.4: Convergence criteria for Earth centered orbit targeting

distance	eccentricity	angles
$20\,km$	0.00005	0.005 deg

papers, i.e. cases A through D. However, in test case B, which is also from the Q-Law publications, and where the semi major axis, eccentricity and inclination are to be corrected simultaneously, both the Naasz control and the Ruggiero fail to reach the target orbit.

3.4.2   Comparison

The simulations below were ran in the "space mission design" software library which is detailed in Chapter 4.

3.4.2.1   Earth orbits

Figures 3.1 to 3.3 show differences in the variation of orbital elements over time for different validation cases where several orbital parameters are corrected at the same time. In blue is the variation using the Ruggiero control laws, and in black is the same variation using the Naasz control laws.

Figure 3.4 compares how both control laws vary the orbital elements over time of a hypothetical scenario of a 3755 kg spacecraft from a GTO orbit to a GEO orbit. The initial parking orbit is that of ViaSat-1 launched by Proton Briz-M Enhanced in 2011: perigee altitude of 2363 km and an apogee altitude of 35786 km with an inclination of 30.4 deg. The scenario also sets the initial orbit with $\Omega = 1$ deg, $\omega = 1$ deg and $\nu = 15$ deg. The target orbit is a co-located geostationary orbit with a radius of 42164.0 km, an eccentricity of 0.01 and an inclination of 0.03. Other orbital elements are free. One may notice that in all these scenarios the Ruggiero control laws lead to a quicker increase of the semi-major axis than in the Naasz control laws.
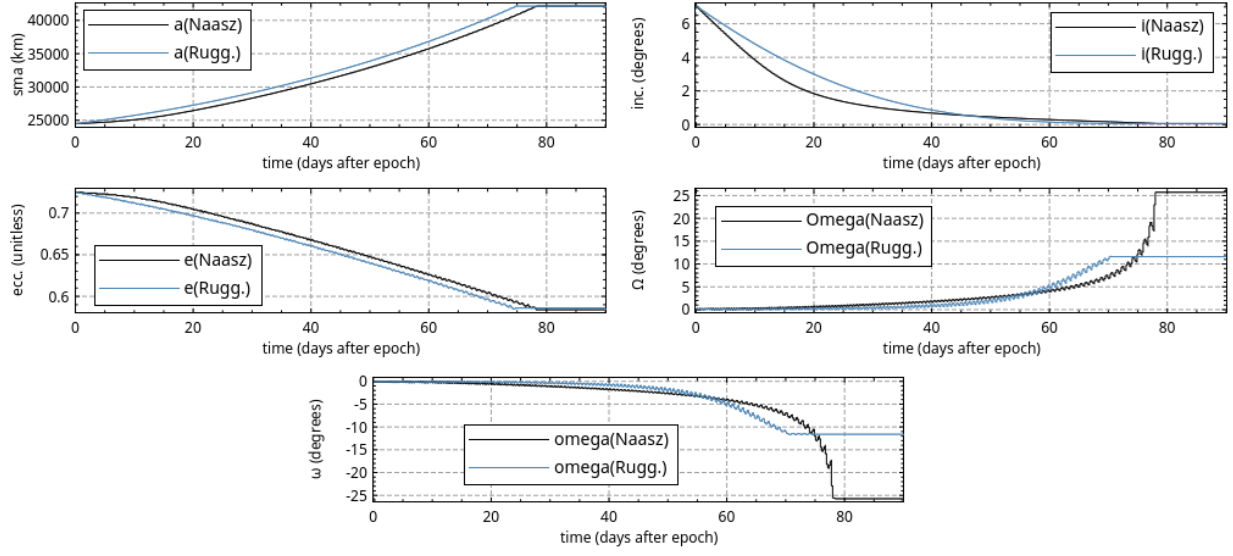
Figure 3.1: Comparison of Naasz and Ruggiero control laws: test case B'
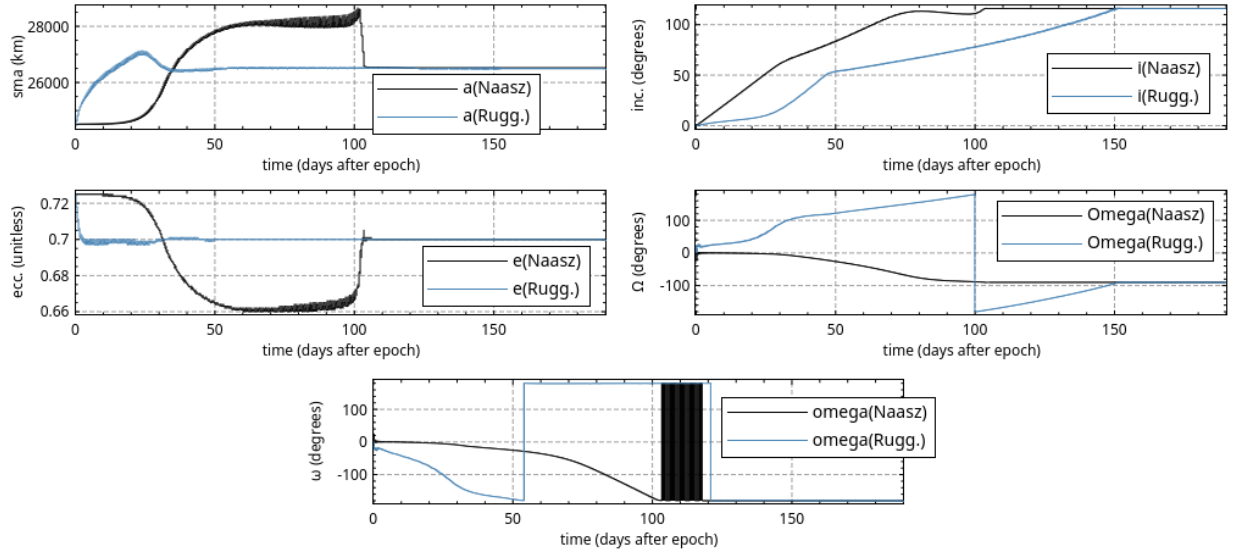
Figure 3.2: Comparison of Naasz and Ruggiero control laws: test case D
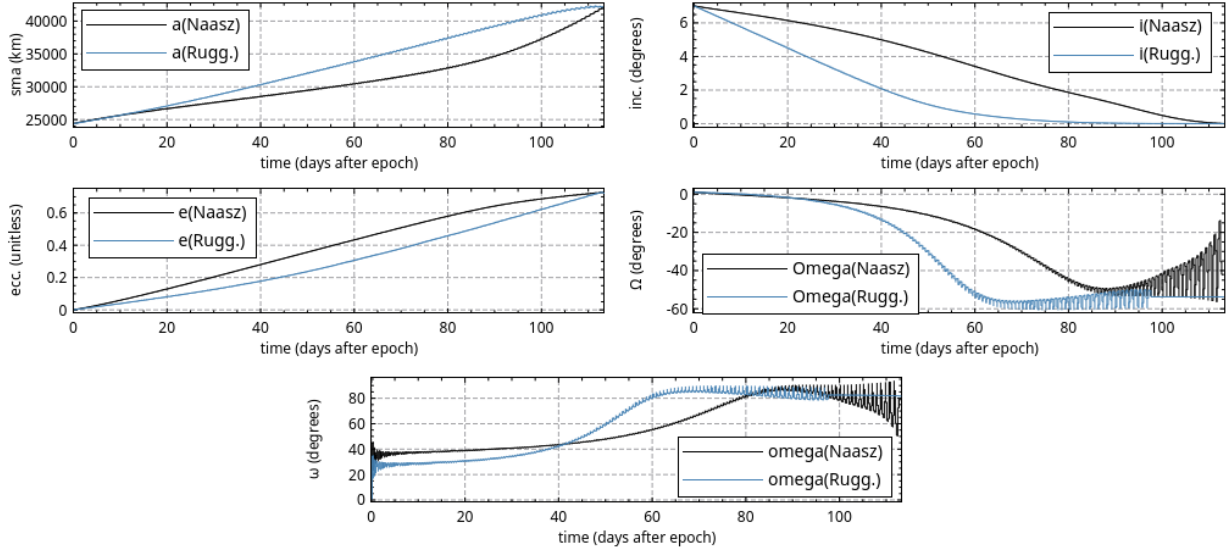
Figure 3.3: Comparison of Naasz and Ruggiero control laws: test case G
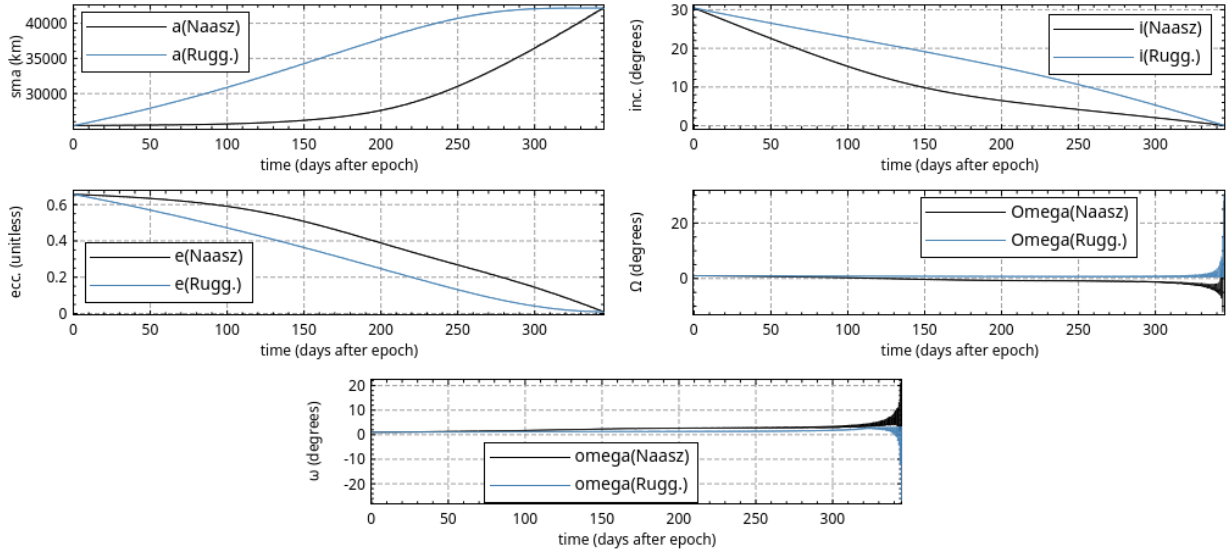
Figure 3.4: Comparison of Naasz and Ruggiero control laws: Hypothetical GTO to GEO scenario
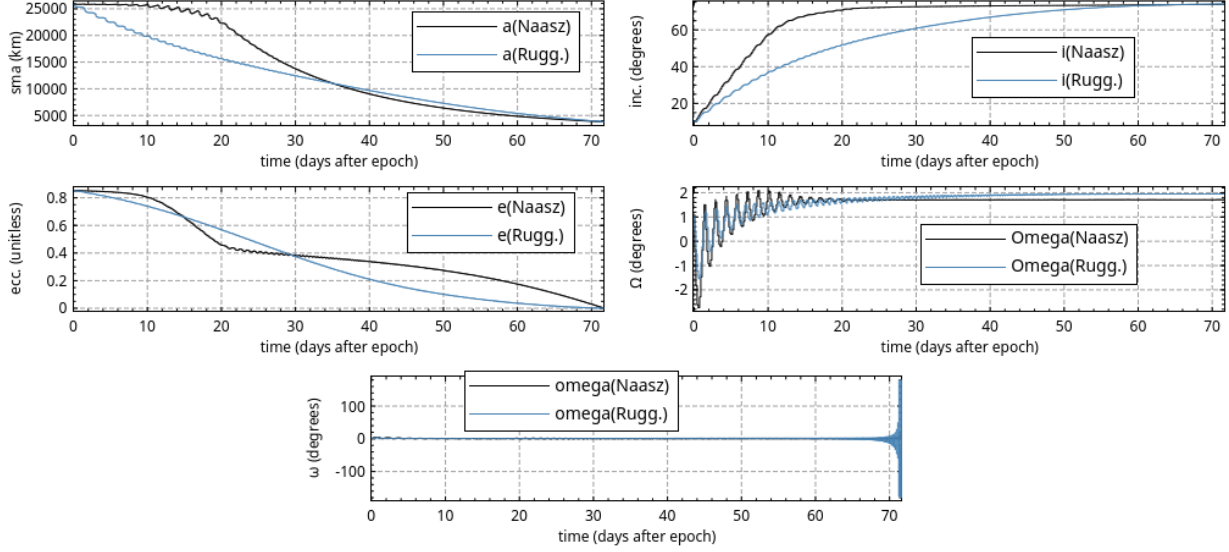
Figure 3.5: Comparison of Naasz and Ruggiero control laws: MRO

3.4.2.2   Mars orbits

In this example, I look at a hypothetical efficient orbital element correction of the NASA Mars Reconnaissance Orbiter from its initial injection orbit to its mission orbit. The initial injection orbit has an altitude of apoapsis of 44500 km, an altitude of periapsis of 426 km, both above Mars, and an inclination of 10 deg [34]. The desired orbit is a circular orbit with an altitude of 400 km and a 74 deg inclination. In this scenario, the argument of periapsis and the right ascension of the ascending node are kept as free parameters. In addition, the initial orbit is set to $\Omega = 1$ deg, $\omega = 1$ deg and $\nu = 15$ deg. The spacecraft is simulated as having a 755 kg dry mass and 3000 kg of propellant. It is hypothesized to be equipped with a single thruster delivering a constant 2 N at an $I_{SP} = 2000$ s, also constant.

Figure 3.5 show the variation in orbital elements over time when using respectively the Naasz and Ruggiero control laws. Figures 3.6 and 3.7 display these orbits in NASA's Cosmographia tool.

These simulations confirm that the control laws are applicable to other celestial bodies, as was expected.
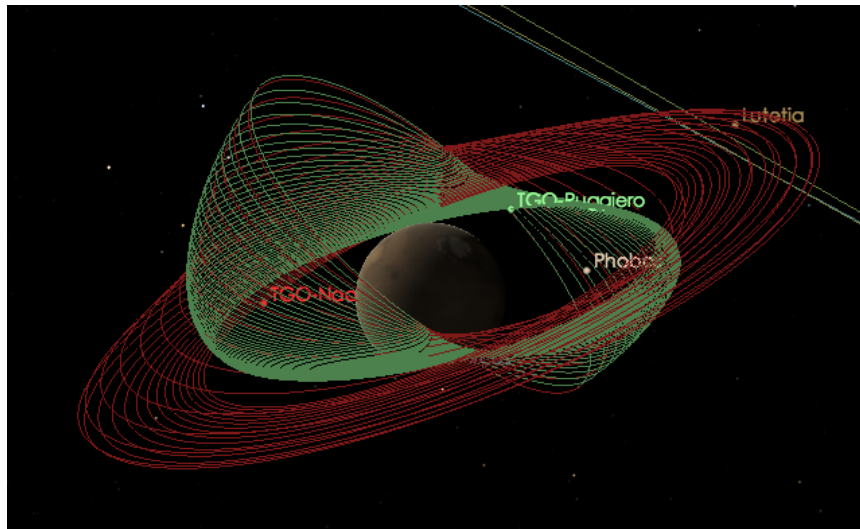
Figure 3.6: MRO comparison plots in Cosmographia - One month before objective - Naasz laws trajectory in red, Ruggiero in green
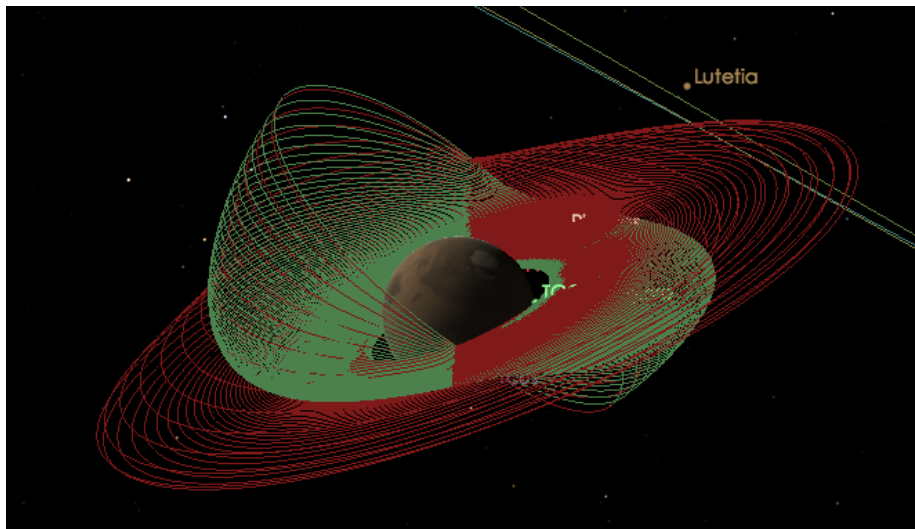
Figure 3.7: MRO comparison plots in Cosmographia - One day before objective - Naasz laws trajectory in red, Ruggiero in green

Table 3.5: Naasz and Ruggiero orbit validation examples

Case	Orbit	a (km)	e	i (deg)	Ω (deg)	ω (deg)	Result
A	Init.	7000	0.01	0.05	0	0	
	Targ.	42000	0.01	free	free	free	
	Naasz	42006.694	0.0100	0.050	0.000	22.129	ok
	Ruggiero	41980.3	0.0376	0.050	0.000	345.283	fail
B	Init.	24505.9	0.725	7.05	0	0	
	Targ.	42165	0.001	0.05	free	free	
	Naasz	28354.4	0.3220	2.834	0.045	0.022	fail
	Ruggiero	34332.5	0.3594	0.284	0.338	359.603	fail
B'	Init.	24505.9	0.725	7.05	0	0	
	Targ.	42165	free	0.05	free	free	
	Naasz	42145.002	0.584	0.055	25.724	-25.736	ok
	Ruggiero	42145.028	0.586	0.055	11.604	-11.609	ok
C	Init.	9222.7	0.2	0.573	0	0	
	Targ.	30000	0.7	free	free	free	
	Naasz	29980.065	0.700	0.573	0.000	2.304	ok
	Ruggiero	29980.015	0.700	0.573	0.000	16.233	ok
D	Init.	24505.9	0.725	0.06	0	0	
	Targ.	26500	0.7	116	-90	180	
	Naasz	26519.7792	0.700	116.005	-90.004	-179.998	ok
	Ruggiero	26495.279	0.700	116.005	-90.005	-179.996	ok
E	Init.	24396	unknown	unknown	unknown	unknown	
	Targ.	42164	free	free	free	free	
	Naasz	42144.039	0.003	0.005	0.088	5.352	ok
	Ruggiero	42144.039	0.003	0.005	0.088	5.352	ok
F	Init.	15378	0.01	98.7	unknown	unknown	
	Targ.	free	0.15	free	free	free	
	Naasz	15399.585	0.150	98.700	0.000	1.817	ok
	Ruggiero	15399.585	0.150	98.700	0.000	1.817	ok
G	Init.	24396	0.7283	7	unknown	unknown	
	Targ.	42164	0	0	free	free	
	Naasz	42146.549	0.728	0.010	-42.079	78.782	ok
	Ruggiero	42144.041	0.728	0.010	-53.765	81.633	ok

Table 3.6: Naasz and Ruggiero duration and fuel validation examples

Case	Thrust (N)	Initial Mass (kg)	Specific Impulse (s)	Control law	Flight time (days)	Fuel usage (kg)	Result
A	1	300	3100	Expected	14.600	41.4953	
				Naasz	14.547	41.3417	ok
				Ruggiero	14.414	40.964	ok
D	2	2000	2000	Expected	81.61	719.012	
				Naasz	117.93	1039.04	fail
				Ruggiero	152.6	1344.668	fail
F	89×10^{-3}	367	1650	Expected	~ 30.5	14	
				Naasz	30.00	10.37	fuel usage issue
				Ruggiero	30.00	10.37	fuel usage issue
G	89×10^{-3}	367	1650	Expected	103.9	49	
				Naasz	113.24	53.8	ok
				Ruggiero	109.0	51.8	ok

Chapter 4

Simulation software and environment

4.1 Introduction

The topic of this research made it evident that a very large number of missions were to be simulated, and the parameters of these missions would have to be easily amenable. This led to a reflection concerning the tools and environment in which to run these scenarios.

Both NASA GMAT and AGI STK support some level of scripting, either via Matlab, or Python. However, both of these mission design tools are heavy to run, require a front-end environment, i.e. cannot be ran on virtual machines on cloud computing platforms, and their high fidelity models cause each simulation to run over relatively long periods of time. Hence both pieces of software were disqualified. It initially seemed wise to leverage my thorough programming experience and interest in new compiled languages to develop a new tool which would support this research. The desire to use a recent compiled language was driven especially by the ease of implementing real multi-threading capabilities while also limiting data race conditions.

This chapter discusses the simulation software language and its architecture, along with some hindsight relevant to the choice of language for prototyping scientific applications. Additionally is discussed the visualization software for orbits and plots, and the use of commercial cloud computing platforms for running CPU intensive simulations over several days.
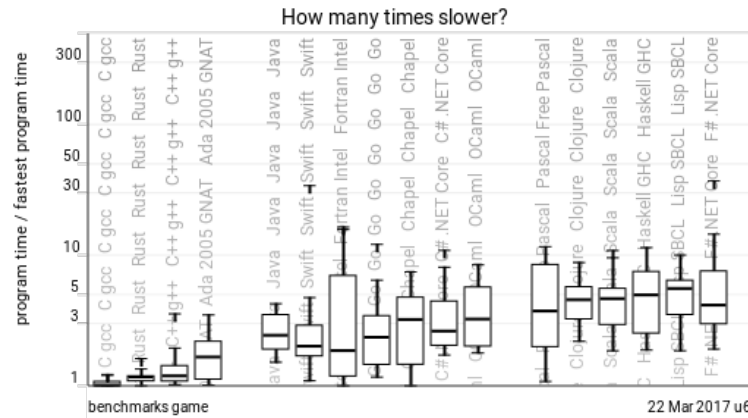
Figure 4.1: Go performance compared to other compiled languages [58]

4.2   Simulation software

4.2.1   The Go programming language

Go is an open source statically typed compiled programming language. It was initially developed at Google Research by Robert Griesemer, Rob Pike, and Ken Thompson. It was made open source 2009. The goal of this programming language is to provide close to C++ performance while increasing the speed of development, easing dependency management and especially taking advantage of modern computer hardware and its many processing cores. [41] Figure 4.1 compares the execution speed of Go with other compiled languages. Overall, it is designed to be concise yet expressive, and concurrent yet memory safe.

Memory management

In C++, the programmer must ensure that the destruction of an object leads to the release of memory and that no memory is used after being freed. Another paradigm, which is used by Go, is that of garbage collection. The programming language will regularly "stop the world" (i.e. stop the execution of every single thread), analyze the memory and free memory wherever possible. Hence, Go is not applicable for hard real time applications. However, this delay is usually very small. Specifically, as of 2015 and Go version 1.5 [23], the garbage collection pauses the execution of the program for a typical worst case of $2\,ms$. There is at least one design change proposal [6]

by A. Clements and R. Hudson which should lead to a typical worst case garbage collection time of less than $1\,ms$.

Concurrency handling

True multi-threaded programming leads to many complications when different thread must communicate between each other. For example, if two threads A and B are running in parallel, and both attempt to modify a variable D, then which thread of A or B gets the upper hand? This situation is called a data-race condition, since both threads are racing to modify the variable. The usual approach to solve this problem is to have specific parts of the software which are not concurrent. For example, in Java, the keyword `sync` will virtually make a method single threaded by decorating it call a mutex lock. In Go, the communicating sequential processing paradigm is employed. Although allowed by the language, concurrent context sharing is highly discouraged and the programmer may be warned at testing time of such issues. Instead, the language provides communication channels, which work just like UNIX pipes. One thread will wait for data to be written to a given channel, and execute its chunk of logic based on that data, while another N threads may be writing data to the channel. Channel operations are unidirectional, preventing any data race between them. [7]

Dependency management

Managing library dependencies in any large project may quickly become complicated. In C++, it is not unusual to keep obsolete versions of the same library to ensure compilation of a project which requires that specific version of the library. In Go, dependency management is also a big issue, but is handled differently. Libraries, usually referred to as packages, are usually installed per user or per project. They may be fetched directly from Git repositories, but Go does not trivially support versioning of said packages. Hence, it is not seldom that a package update will lead to API breaking changes, and therefore require the developer to update their project code in response. However, the Go maintainers and contributors are aware of such issues, and there is currently a major initiative to provide a versioned dependency management tool similar to Python's `virtualenv` or Javascript's `package.json`. [9, 52]

Use cases and scientific application

The Go programming language is especially suited for highly concurrent systems and as such is used extensively in high throughput web servers, messaging queues, or high availability databases. Apart from a few comprehensive mathematical packages like gonum [10], which relies on OpenBLAS for linear algebra computations, there are only very few scientific pieces of software which primarily use Go. In fact, as far as I know, I was the first to release a mutli-variate integrator of any flavor [45] and a Kalman filtering library [44].

4.2.2   The "Space Mission Design" package

The "space mission design" Go package, herein `smd` [46], was developed specifically for simulations in my thesis. The library also includes additional developments done to support classes of orbital determination and interplanetary mission design.

Architecture

The package is organized as a library in the sense that each simulation run independently of the package itself and must import the library. The main components of `smd` are the spacecraft, the mission design and the mission propagation segments. A spacecraft is defined by a name, a dry mass and fuel mass, a electrical power subsystem (EPS), a set of electric propulsion thrusters, and a set of cargo (themselves instances of spacecraft). Each thruster is defined as thrust level and specific impulse (in second) given a voltage and power: this information is used by the EPS to determine whether the spacecraft can perform the desired thrust.

In addition, the spacecraft includes the mission design information. A given mission is defined by a set of way points, each of which may also include an action to be taken at the given waypoint, such as switching reference frame, or adding and dropping cargo and eventually spawning a new mission propagation instance. Way points are quite general to allow for flexibility. For example, `smd` currently defines way points for spiraling out of a planet's sphere of influence, as well as for attempting to make the spacecraft's orbit elliptical from a hyperbolic orbit. The more complex waypoints allow for orbit targeting: a mission design may specify which Keplerian orbital elements

to target using either the Naasz or Ruggiero control laws. The orbit targeting way point is specif-
ically useful for simulating a whole mission from a parking orbit to an interplanetary destination,
including the orbital injection and a parking orbit at destination. Leveraging one key aspect of
the Go programming language, the spacecraft state vector and its orbital elements are streamed in
near real-time to output files: this allows for low memory consumption of the simulation while also
allowing for analysis of a mission if the propagation must be stopped before completion.

Figure 4.2 shows the `smd` propagation flowchart. It is important to note that the only EPS
available as of writing provides unlimited power at all times. Similarly, only one fuel tank is
available and only a warning is displayed once it is empty. Moreover only fixed I_{sp} and maximum
thrusting is tested, although variable specific impulse is supported.
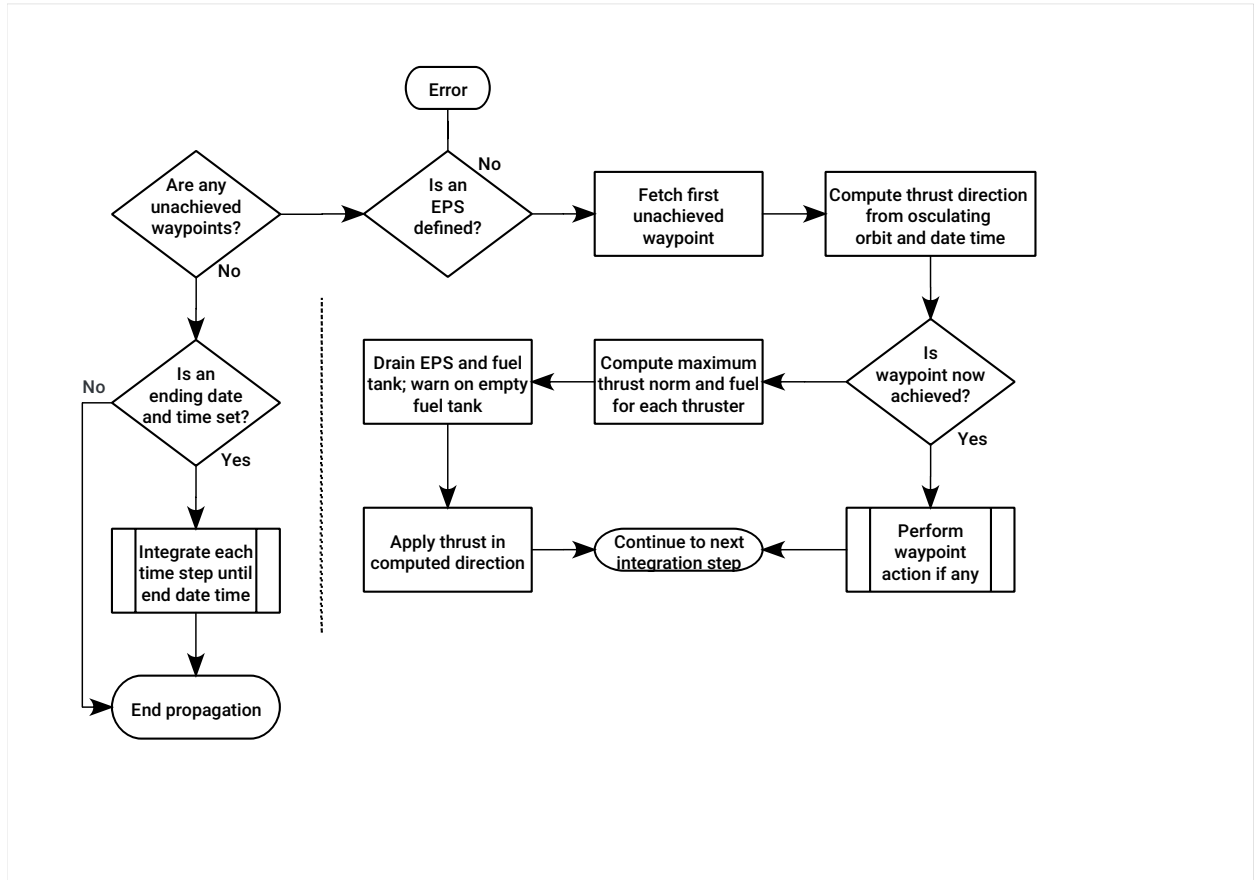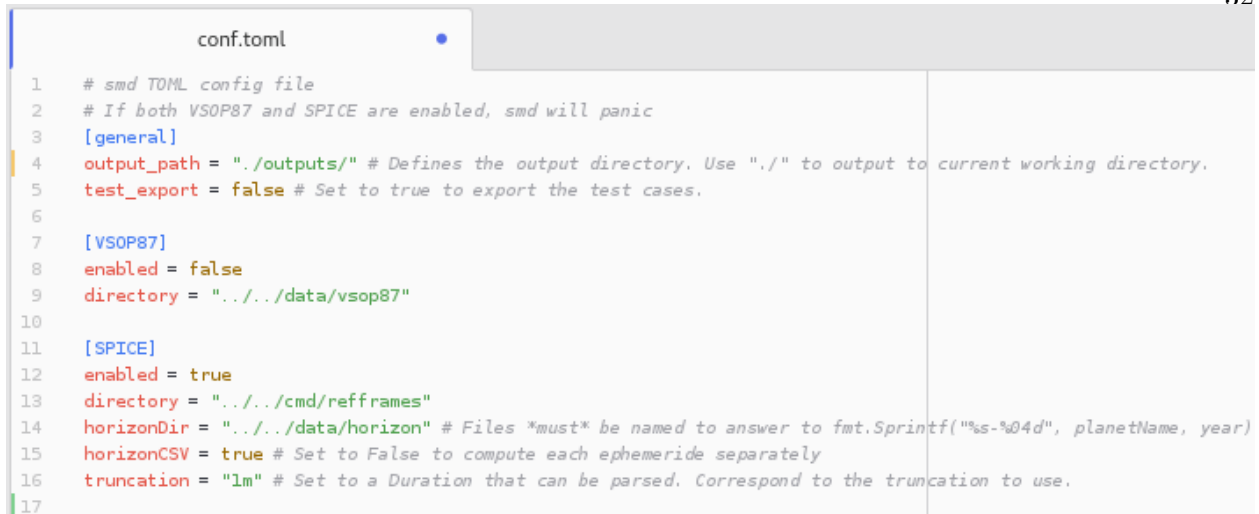
Figure 4.2: Integration flowchart

Fidelity

```
conf.toml                    ●
1    # smd TOML config file
2    # If both VSOP87 and SPICE are enabled, smd will panic
3    [general]
4    output_path = "./outputs/" # Defines the output directory. Use "./" to output to current working directory.
5    test_export = false # Set to true to export the test cases.
6
7    [VSOP87]
8    enabled = false
9    directory = "../../data/vsop87"
10
11   [SPICE]
12   enabled = true
13   directory = "../../cmd/refframes"
14   horizonDir = "../../data/horizon" # Files *must* be named to answer to fmt.Sprintf("%s-%04d", planetName, year)
15   horizonCSV = true # Set to False to compute each ephemeride separately
16   truncation = "1m" # Set to a Duration that can be parsed. Correspond to the truncation to use.
17
```

Figure 4.3: Example ephemerides configuration file for `smd`

The mission propagation segment currently relies on a fixed-step $4th$ order Runge-Kutta integrator, although there are plans to support a variable step Dormant Prince of higher order in the future. The default fixed-time step in the mission propagation segment is set to 10 seconds, but this can be specified on instantiation of a mission propagator. Go being a compiled language, such a low time step still allows for multi-year simulations with continuous thrusting to be numerically integrated in a few minutes. Initially for the purposes of orbital determination, `smd` also supports J_2, J_3 and third body perturbations. Currently the perturbing third body must be specified at the start of the scenario, but automatic multi-body perturbations are planned. Planetary ephemerides may be calculated via one of three methods: either using the VSOP87 algorithms, which are computed directly in Go, or via NASA' SPICE Toolkit through a Python interpreter or an import of JPL HORIZON comma separated values files. The ephemerides configuration is determined in a TOML configuration files, an example of which is shown in Figure 4.3. The ephemeride computation process is summarized in Figure 4.4.

Hindsights

The main advantage of using the Go programming language was the ease of development, multi-thread handling and deployment. However, Go is not usually used for scientific applications.
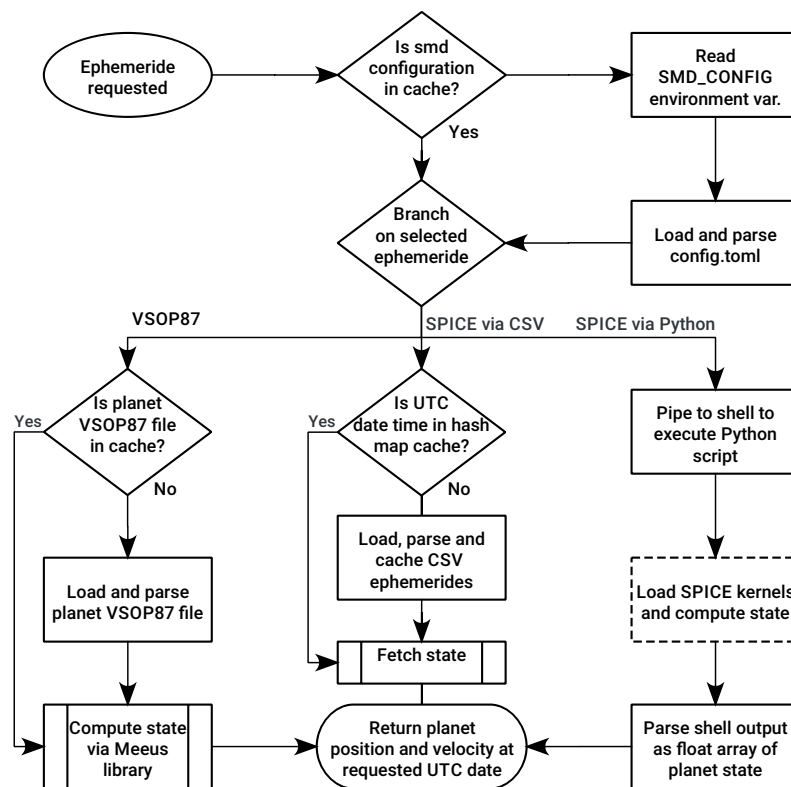
Figure 4.4: Planetary ephemerides computations in `smd`

This required developing a number of features and packages which are more commonly available in other languages such as C, C++ or Python. For example, prior to completing the `ode` package [45], there was no usable multivariate RK4 integrator in Go neither in the official libraries, nor on Github, the largest host of open source code in the world [19].

Availability

The space mission design package is open sourced on Github [43] in hopes of getting some feedback from people not involved in this work. Installing `smd` requires Go version 1.6 or higher and Python 2.7 or higher. The deployment process is as follows.

(1) Clone the Git repository to a local machine

(2) Navigate to one of the examples, such as `examples/thesis` directory

(3) Compile the code with `go build`

(4) Set the `SMD_CONFIG` environment variable to point to a directory containing the `conf.toml` file. This configuration file determines whether to use the Meeus algorithms or SPICE for planetary ephemerides, and where to store the output files generated by the simulation. An example of such file is found in the root directory of the project.

(5) If using the Meeus and VSOP87 algorithms for planetary ephemerides, the relevant VSOP87 files must be downloaded (e.g. from [ftp://ftp.imcce.fr/pub/ephem/planets/vsop87]) and placed in the directory referenced in the configuration file

(6) If using the SPICE kernels, the following actions must also be taken:

    (a) Navigate to the `cmd/refframes` directory

    (b) Create a new Python 3 virtual environment and install the requirements from `reqs.txt`

    (c) Source into the new virtual environment

    (d) Download the following SPICE kernels from NAIF: `de430.bsp, naif0012.tls, pck00010.tpc`

    (e) Place these files in the directory referenced in the configuration file

(f) Optionally, test the installation is valid by running `python tests.py`, which should print `OK`.

(7) Finally, run the example by invoking the executable file directly

4.3   Visualizations

4.3.1   Cosmographia

Trajectories are plotted in Cosmographia, a NASA NAIF mission visualization tool. [32] Planet and available spacecraft trajectories are plotted from their respective SPICE kernel data. In this interactive tool, users can manipulate time and 3D space, and import any pre-computed trajectory which ought to be plotted in a given frame. In addition, spacecraft CAD models may also be displayed, each possibly defining any number of additional reference frames relative to the center of the mass of the spacecraft.

The `smd` suite and Cosmographia are integrated simply through export files. Specifically, the Space Mission Design library exports two Cosmographia files as it propagates a mission: an `xyzv` file and a `json` file. The former is a tab separated values file where each line corresponds to a propagated state: the first column is the Julian date, the three subsequent ones are the X, Y and Z position coordinates and the three final columns are the velocity coordinates. The `json` file describes the list of spacecraft to plot, the starting and ending dates of each, the number of interpolations Cosmographia should compute between each row of the `xyzv` file, the reference frame in which the states are expressed, along with trajectory and label colors. A given `json` file may contain several spacecraft plots: in fact Cosmographia does not support frame switching of a given spacecraft during its propagation. Hence, the smd package will create as many spacecraft in this description file and as many xyzv files as there are reference frame changes. For example, an Earth to Mars mission starting in the ECI frame and ending the IAU Mars barycenter frame will generate four files: three `xyzv` files for the ECI, Sun-centered J2000 Ecliptic frame and IAU Mars barycenter

frame, and one `json` file which contains three spacecraft (numbered zero to two) each with its own `xyzv` trajectory file.

4.3.2   Kst

Kst is a self-described as "the fastest real-time large-dataset viewing and plotting tool" [8]. It is a freely available open-source tool designed to run on Linux, macOS and Windows. Figure 4.5 shows the interface with data from one of the plots used in this document. Kst saves the plot layout and data source in an XML file. An XML file is human-readable and editable in any text editing software. This eases the modifying of kst files to generate identical plots from different datasets. Unless specified otherwise, all two-dimensional plots in this report have been generated in Kst.

4.4   Cloud computing

Some simulations of Chapter 5 had to be ran over several days. Powerful computers were rented on a hourly basis in remote datacenters (could computing) in order to avoid interruptions in such simulation campaigns. Specifically, an eight-core machine with 32 GB of RAM was rented on the Google Compute Engine (GCE) platform. In terms of main user experience and to the look and feel of the interfaces, the use of GCE was quite similar to Amazon Web Services (AWS)

As a statically linked compiled language, a Go executable file is portable between any platform of the same architecture. Hence, an easy deployment process would have been to perform a secure copy of the file, via `scp`, from a local Linux machine on an `x86_64` platform. However, as some simulation campaigns requires initial condition alterations with respect to the source code on the repository, the GCE server had its own local clone of the repository, and was deployed as described in the availability paragraph of 4.2.2.
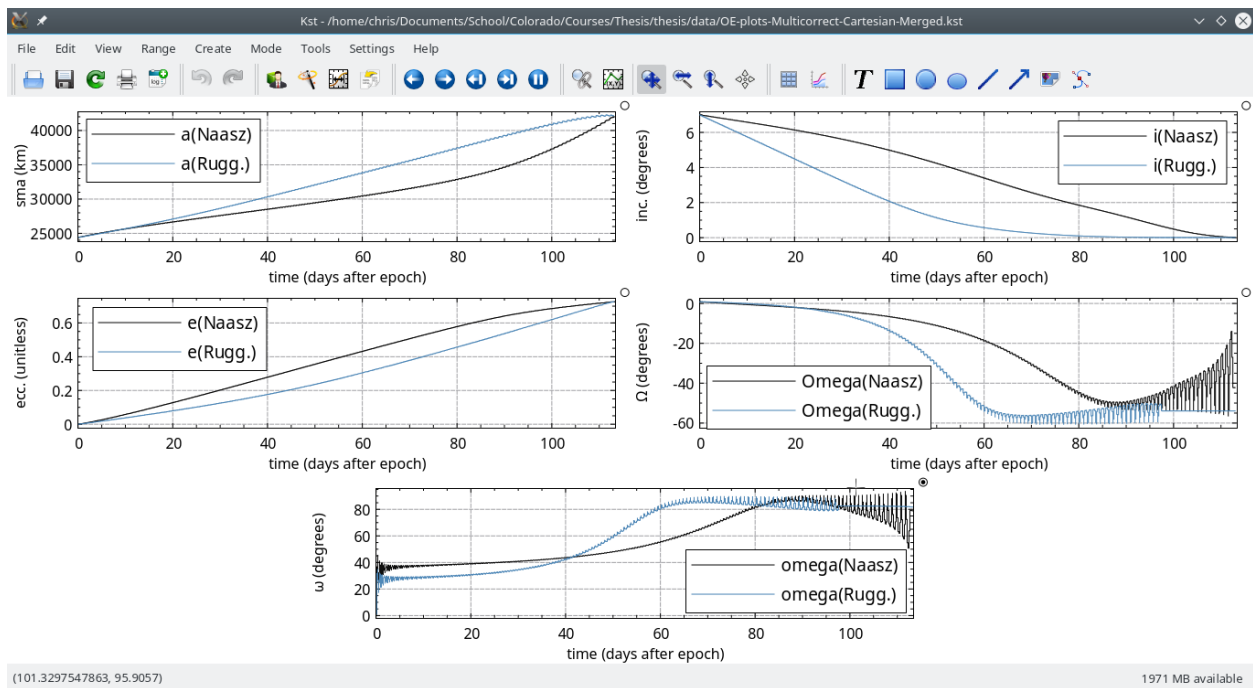
Figure 4.5: The Kst interface

Chapter 5

Continuous thrust for Earth to Mars interplanetary missions with return

5.1 Introduction

This chapter investigates the selection of a convenient parking orbit for the semi-cycler based on a specific thruster configuration, described in Table 5.1. A simple trade study of each patched conic segment is presented, comparing the time of flight of three different hypothetical missions using a cluster of thrusters of the same type. All these scenarios provide a constant thrust and constant specific impulse equivalent to their maximum thrust level. Simulations are stopped when the spacecraft runs out of fuel.

5.2 Simulation scenarios

The following scenarios are summarized in Tables 5.2 through 5.4. The fuel mass was arbitrarily selected as a ratio of the overall payload mass. These scenarios are reset and simulated independently between each test run. This allows for each segment of the patched conics to be analyzed independently and in the most optimistic scenario. The independent analysis also provides insight into ways each section of mission and the overall mission architecture itself can be improved upon. Specifically, the final state of the spacecraft is not carried over from the first segment, i.e. the Earth GTO to Earth SOI boundary, to the second segment, i.e. the Earth to Mars journey. The same methodology was applied for the inbound trip. Moreover, the simulation was stopped if the spacecraft ran out of fuel during its mission. Such instances are flagged below.

Mission 1

Table 5.1: Thruster characteristics

Name	I_{sp} (s)	Thrust (N)	Power (kW)	Thrust to power (mN/kW)
Ad Astra VX-200 [53]	4900	5.800	200	29.0
Busek BHT-1500 [14]	1865	0.179	2.7	66.3
Busek BHT-8000 [15]	2210	0.449	8.0	56.1
NASA HERMeS [36]	2960	0.680	12.5	54.4
Snecma PPS-1350 [59]	1650	0.089	1.35	65.9
Snecma PPS-5000 [13]	1800	0.310	5.0	62.0

Basic example of a spacecraft of 5 metric tons of fuel and 1 kg of dry mass: the vehicle is simulated to have the same mass during both the inbound and outbound journeys. The simulations were ran based on a clustering of up to two of each of the thrusters described in Table 5.1. In the following, mission 1A corresponds to a spacecraft equipped with only one thruster and 1B equipped with two thrusters.

Mission 2

A semi-cycler would be especially useful for delivering more mass than what larger rockets can propel into an escape trajectory. Hence, this hypothetical scenario describes the possibility of delivering multiple payload to Mars, and returning to Earth with light payload. More specifically, the outbound journey corresponds to the delivery of the Mars Science Laboratory, 900 kg, the Mars Reconnaissance Orbiter, 1000 kg, and ESA's Schiaparelli lander, 577 kg. It also supposes the vehicle has a 1000 kg bus and begins its mission with 3000 kg of fuel. The fuel mass at departure, i.e. for the outbound trip, corresponds to 46% of the overall vehicle mass, and 66% on inbound journey. The spacecraft has a clustering of up to two of each of the thrusters described in Table 5.1. In the following, mission 2A corresponds to a semi-cycler equipped with only one thruster and 2B equipped with a cluster of two thrusters.

Mission 3

This scenario investigates using continuous thrust instead of impulse thrusting for the crewed launch of the Mars Design Reference Mission version 5. [20, Table 4-2] Both trips suppose a 52 ton payload and 24 tons of fuel. The analysis was performed supposing a clustering of either eight

Table 5.2: Instantaneous power requirements for each mission

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Power (kW)					
1A, 2A	200.0	2.700	8.000	12.50	1.350	5.000
1B, 2B	400.0	5.400	16.00	25.00	2.700	10.00
3A	1600	N/A	64.00	100.0	N/A	40.00
3B	24.00	N/A	32.40	96.00	N/A	60.00

(mission 3A) or twelve (mission 3B) of the Snecma PPS-5000, Busek BHT-8000, NASA HERMeS or Ad Astra VX-200 thrusters. A semi-cycler vehicle could be especially useful if the time of flight is shorter, as it would lead to less radiation exposure for the crew. In addition, this scenario looks at the possibility of using such an architecture for very large mass transportation for which the Space Launch System is currently being developed. The fuel mass at departure corresponds to 45% of the overall vehicle mass.

5.3   Semi-cycler parking orbit to escape

5.3.1   Around Earth

The main considerations when selecting an Earth parking orbit for a semi-cycler is its ease of accessibility by the payload and the velocity the vehicle can reach relative to Earth at the boundary of the sphere of influence. The geostationary transfer orbit is served by several rockets including Ariane 5, Flacon 9, and Proton with a Briz-M third stage. It's a highly elliptical orbit which reduces the number of spirals of the spacecraft has to perform to depart the sphere of influence of Earth, $9.24 \cdot 10^5$ km. In the following simulations, the GTO is defined by an altitude of periapsis of 290 km and an altitude of apoapsis of 39300 km, and an inclination of 0.0 degrees. Additionally, the right ascension of the ascending node was set to 0.0 degrees and the true anomaly to -150 degrees. For comparison, an Hohmann transfer from an Earth circular orbit of 290 km altitude to the Earth SOI boundary takes 18.3 days; from a circular geostationary orbit, the Hohmann transfer to that same distance would take 19.5 days.

Table 5.3: Total thrusting level for each mission

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Thrust (N)					
1A, 2A	5.8	0.179	0.449	0.680	0.089	0.310
1B, 2B	1.16	0.358	0.898	1.36	0.178	0.620
3A	4.64	N/A	3.59	5.44	N/A	2.48
3B	69.6	N/A	5.39	8.16	N/A	3.72

The goal is to have the highest Earth relative velocity when leaving the sphere of influence. Mars is further from the Sun than Earth is, and therefore any vehicle leaving Earth to reach Mars must increase its orbital radius, which requires an increase in velocity. Preliminary test runs showed that the inclination played no role in the norm of the final velocity at SOI relative to Earth, herein referred to as V_{relative}. Hence, Figures 5.1 through 5.6 show the effects of the argument of perigee of the departure GTO on the V_{relative} for each mission using a 10 deg argument of perigee step. The thrust direction is purely in the velocity direction in order to increase the semi major axis. As mentioned previously, the thrust magnitude and I_{sp} are both constant. It is interesting to note that the velocity at the boundary does not depends on the thrust level in all cases but the VX-200. However, this does not seem to be related to the thrust to mass ratio, as shown in Table 5.4. Moreover, the maximum V_{relative} is in the vicinity of $0.7\,\text{km/s}$, apart for the VX-200 engine in the baseline idealized scenario 1A and 1B. As a comparison, NASA Galileo, launched in 1989, could attain a hyperbolic excess velocity of $4.1\,\text{km/s}$ after being launched on a hyperbolic trajectory via solid rocket motors. [16]

Figures 5.7 and 5.8 show the time of flight in days from the chosen GTO to the sphere of influence of Earth. A negative V_{relative} means that the spacecraft is going slower than Earth is around the Sun at the SOI boundary.

5.3.2   Around Mars

The selection of the departure orbit around Mars is not limited by its ease of access of the surface. Instead, the focus was on the ease of departure from the sphere of influence. In an attempt

Figure 5.1: Argument of perigee versus V_{relative} - Mission 1A

Figure 5.2: Argument of perigee versus V_{relative} - Mission 1B

Figure 5.3: Argument of perigee versus V_{relative} - Mission 2A

Figure 5.4: Argument of perigee versus V_{relative} - Mission 2B
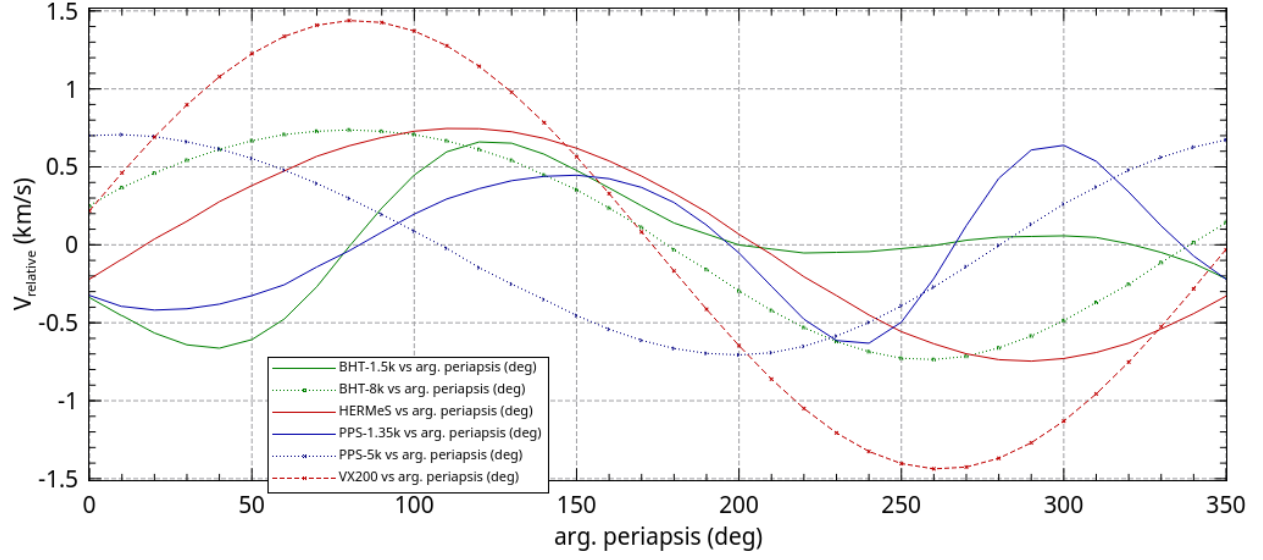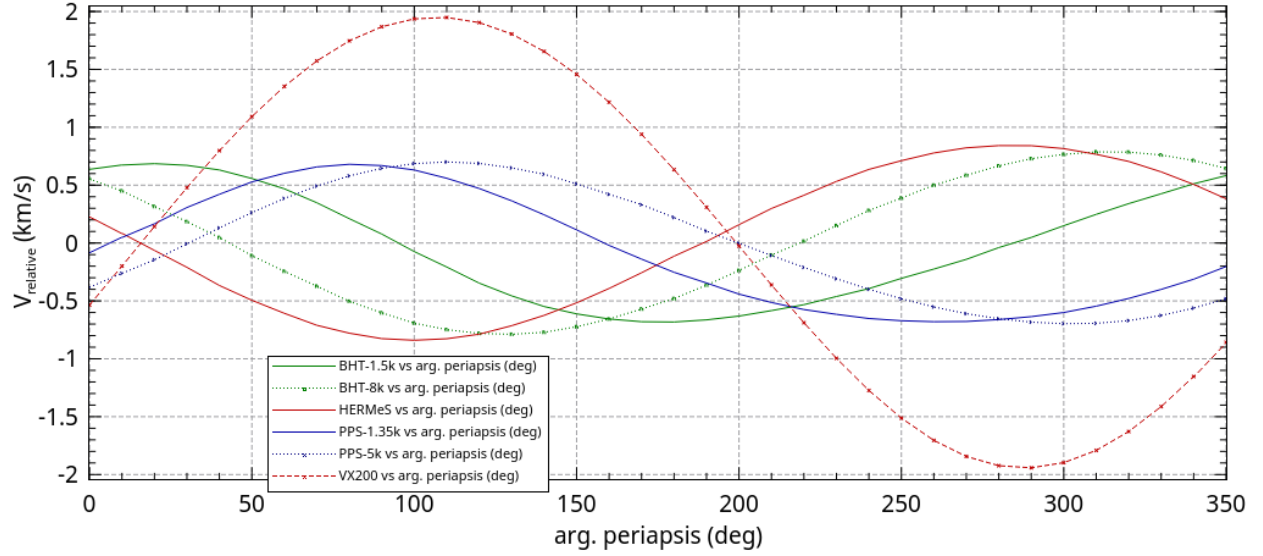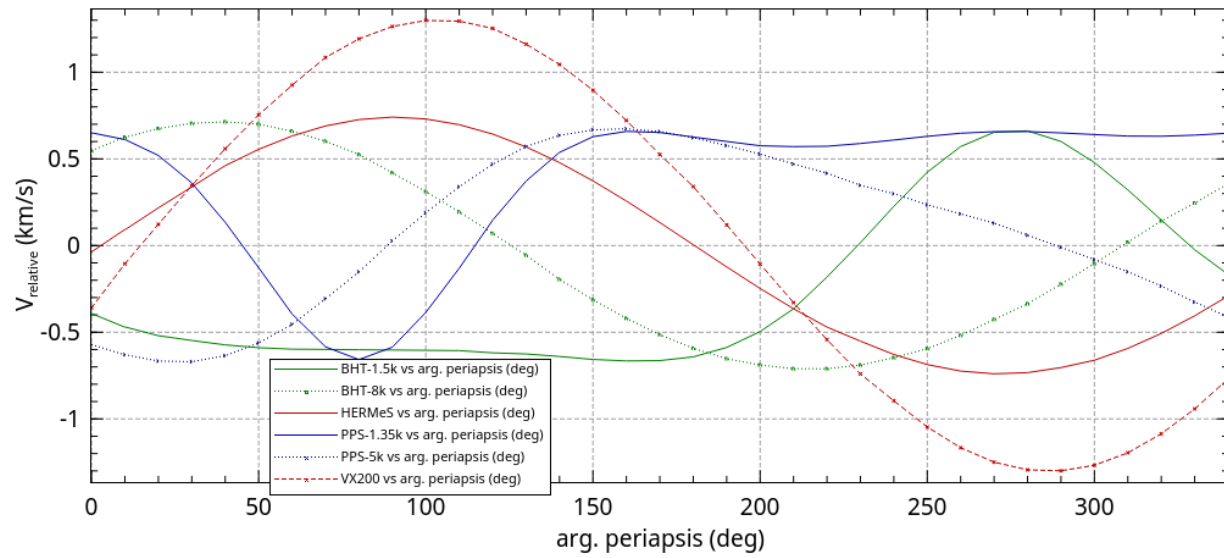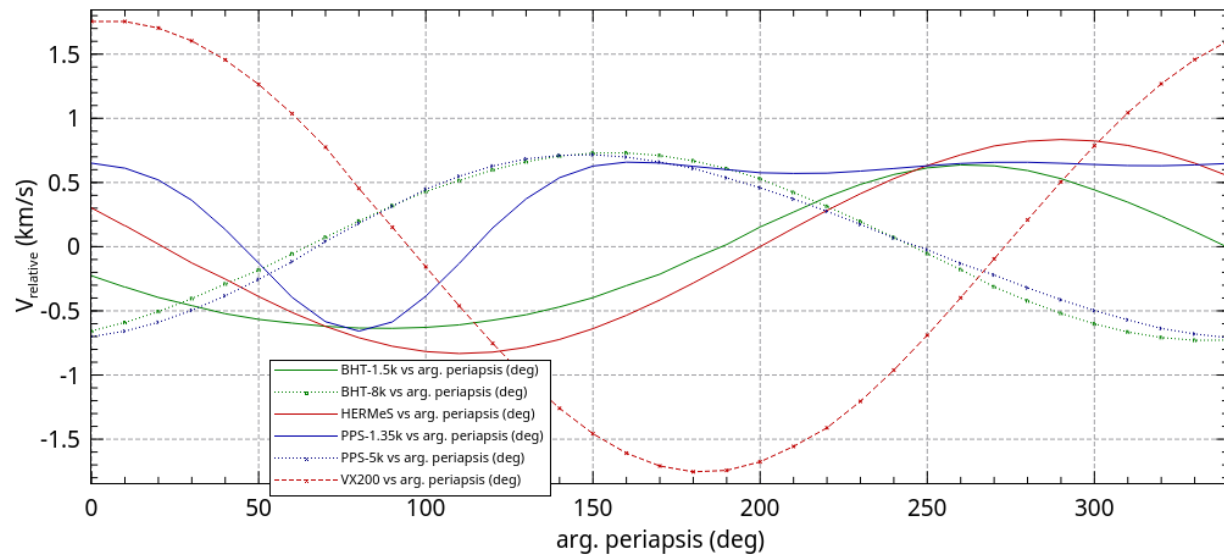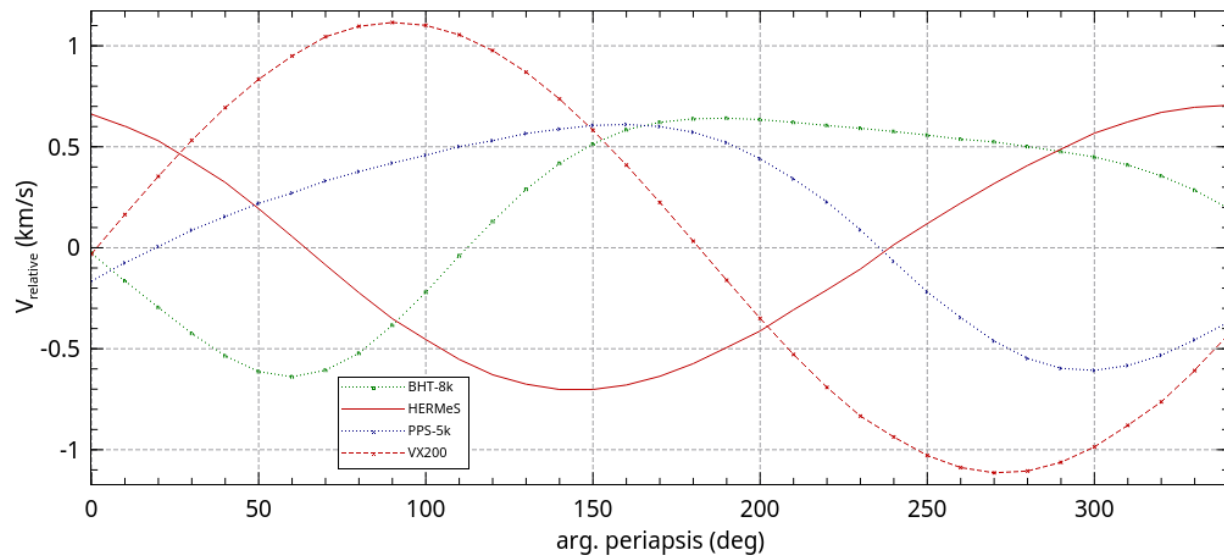
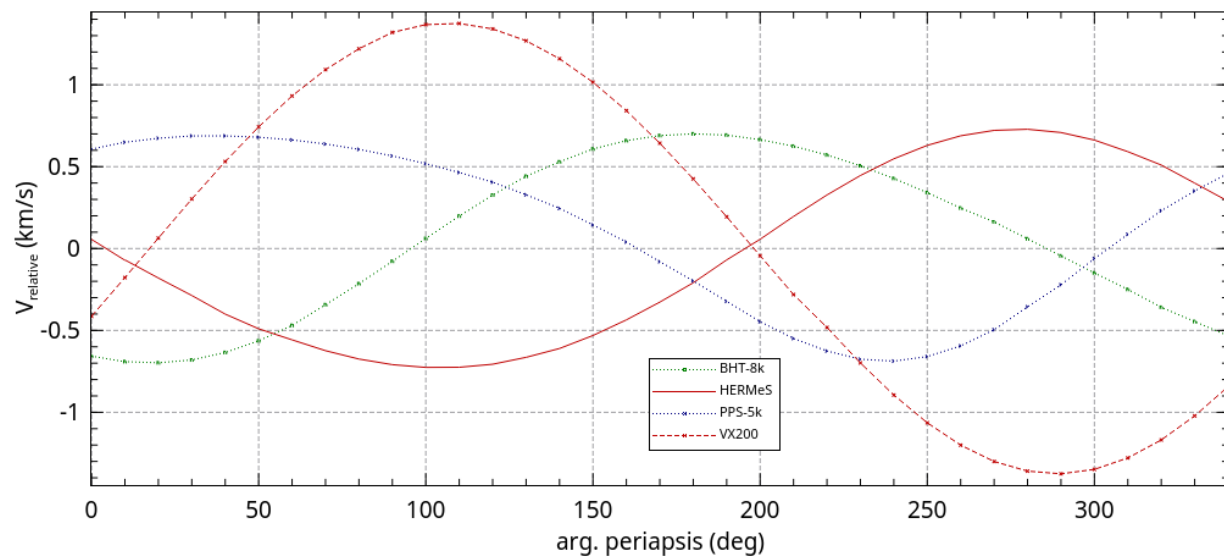Figure 5.5: Argument of perigee versus V_{relative} - Mission 3A

Figure 5.6: Argument of perigee versus V_{relative} - Mission 3B
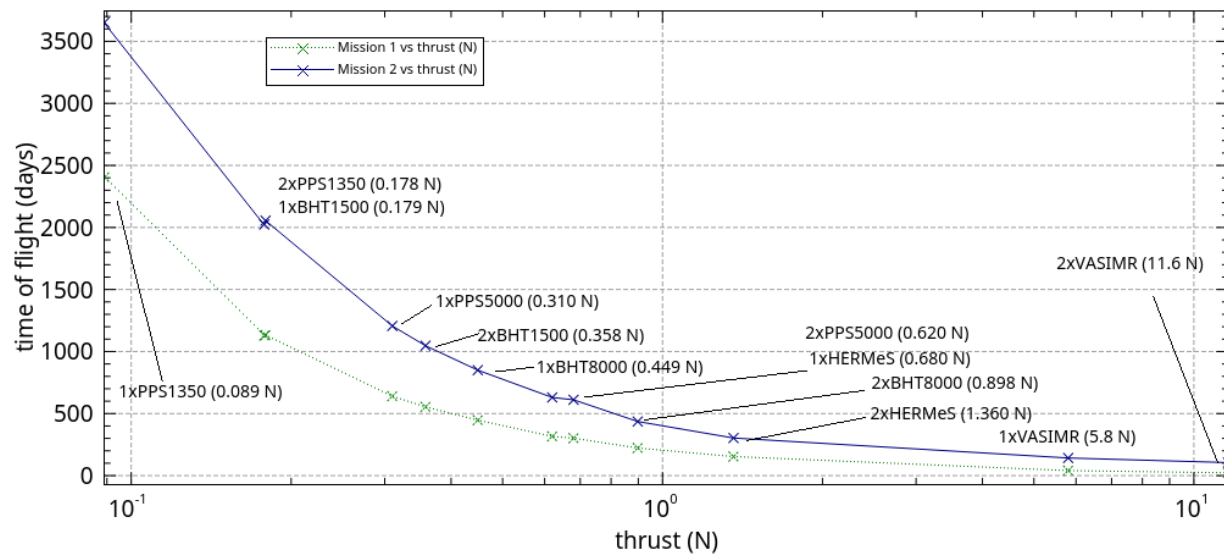
Figure 5.7: Time of flight from GTO to Earth SOI: Missions 1 and 2
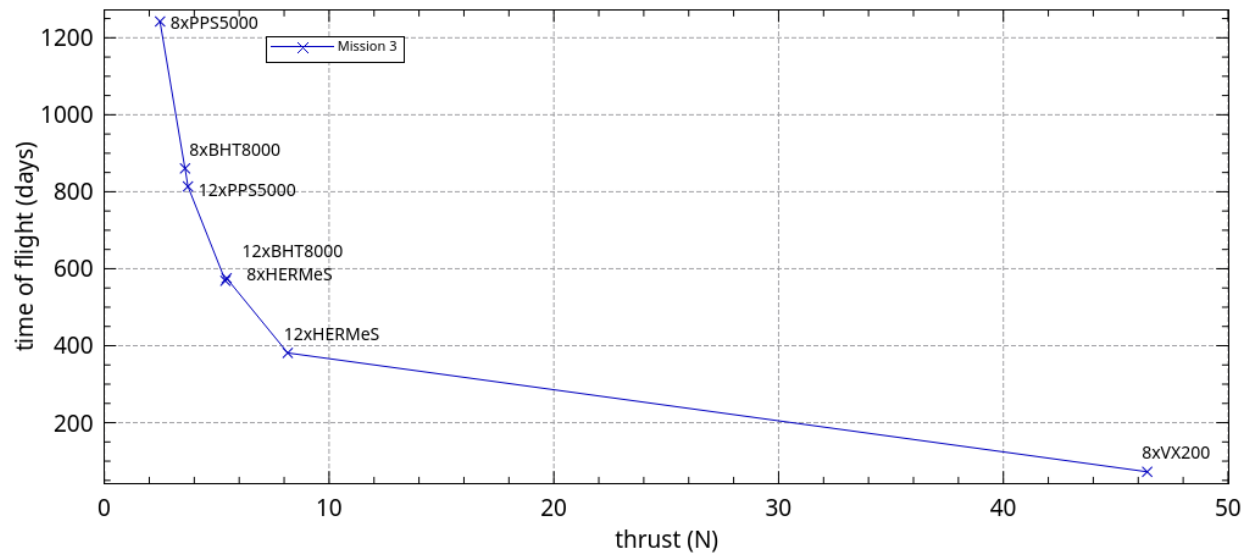
Figure 5.8: Time of flight from GTO to Earth SOI: Mission 3

Table 5.4: Thrust to initial wet mass ratio for each mission

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Thrust to initial wet mass ratio ($\mathrm{mN/kg}$)					
1A	1.160	0.036	0.0898	0.128	0.0178	0.062
1B	3.20	0.072	0.178	0.280	0.036	0.124
2A	0.890	0.027	0.069	0.100	0.140	0.047
2B	1.00	0.030	0.078	0.117	0.015	0.053
3A	0.610	N/A	0.047	0.071	N/A	0.033
3B	1.059	N/A	0.082	0.124	N/A	0.056

to run a plausible scenario, the injection orbit of NASA's Mars Reconnaissance Orbiter was used as the parking orbit. The orbit has an altitude of apoapsis of 44500 km, an altitude of periapsis of 426 km, both above Mars, and an inclination of 10 deg [34]. In addition, the right ascension of the ascending node was set to 0.0 deg and the true anomaly was set to 180.0 deg. As in the case of departing Earth, preliminary simulations confirmed that only the initial argument of periapsis of the orbit influenced the final relative velocity with respect to Mars at the boundary of the sphere of influence.

Figures 5.9 through 5.14 show the influence of the argument of periapsis of the initial orbit on the relative velocity with respect to Mars at the Mars sphere of influence boundary. In these simulations, I am attempting to leave the sphere of influence with the lowest possible excess velocity because the vehicle is Earth-bound. As of 2017, no spacecraft has returned from Mars, so there is no mission data to compare the excess velocity at Mars SOI. However, concerning the time of flight, an Hohmann transfer from a Mars circular orbit of 426 km altitude to the SOI boundary takes 27.4 days, and the same Hohmann transfer starting from an altitude of 44500 km would take 30.6 days.

5.3.3 Via efficient correction of orbital elements

Escaping Earth and Mars using the Naasz control laws was attempted in a small number of simulations. In these, the targeted semi-major axis was the planet SOI, the eccentricity was kept to the original orbit, i.e. a GTO around Earth or the MRO injection orbit around Mars, and the

Figure 5.9: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission A1 Inbound

Figure 5.10: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission 1B Inbound

Figure 5.11: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission 2A Inbound
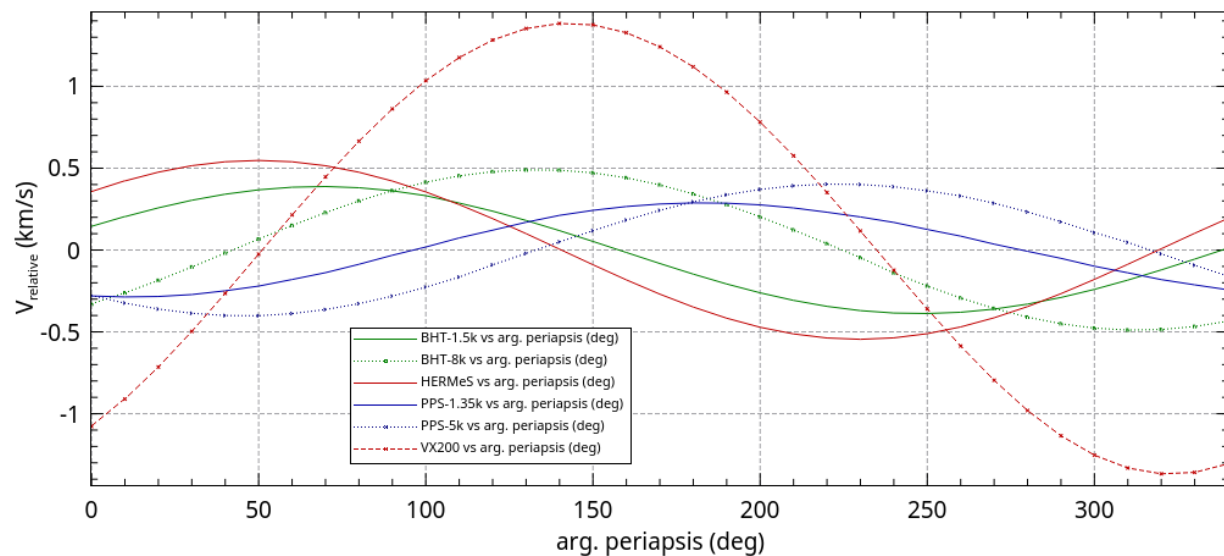
Figure 5.12: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission 2B Inbound
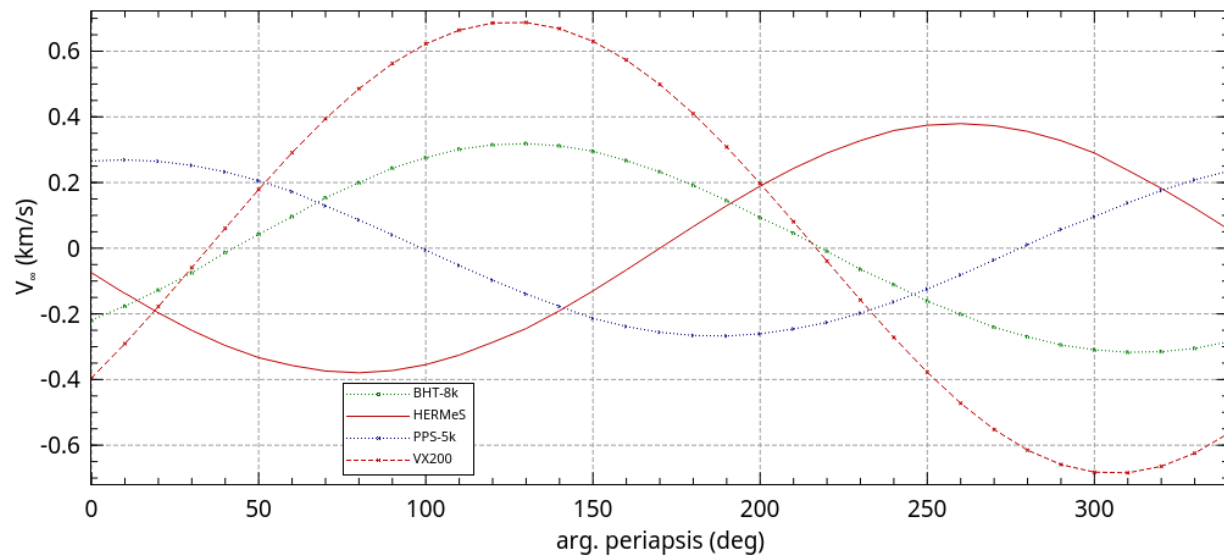
Figure 5.13: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission 3A Outbound
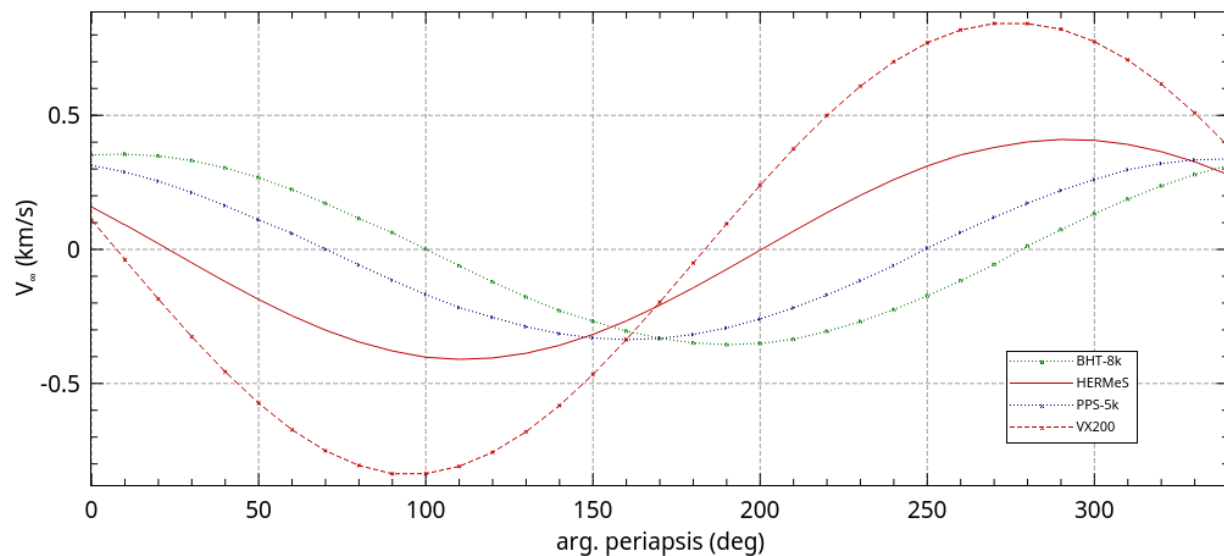
Figure 5.14: Argument of periapsis versus $V_{\text{Mars SOI}}$ - Mission 3B Inbound
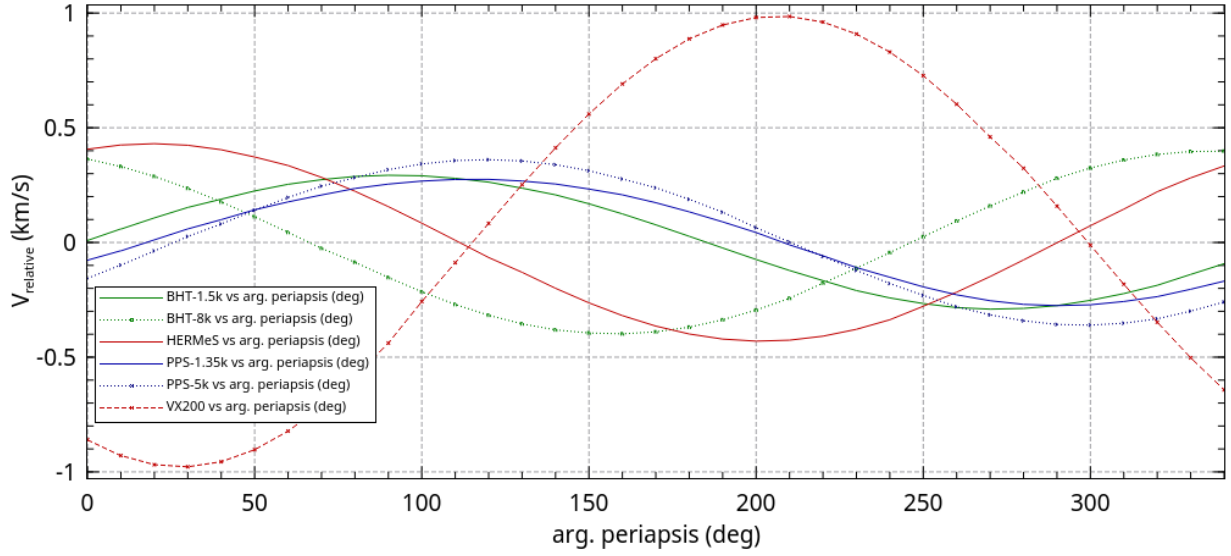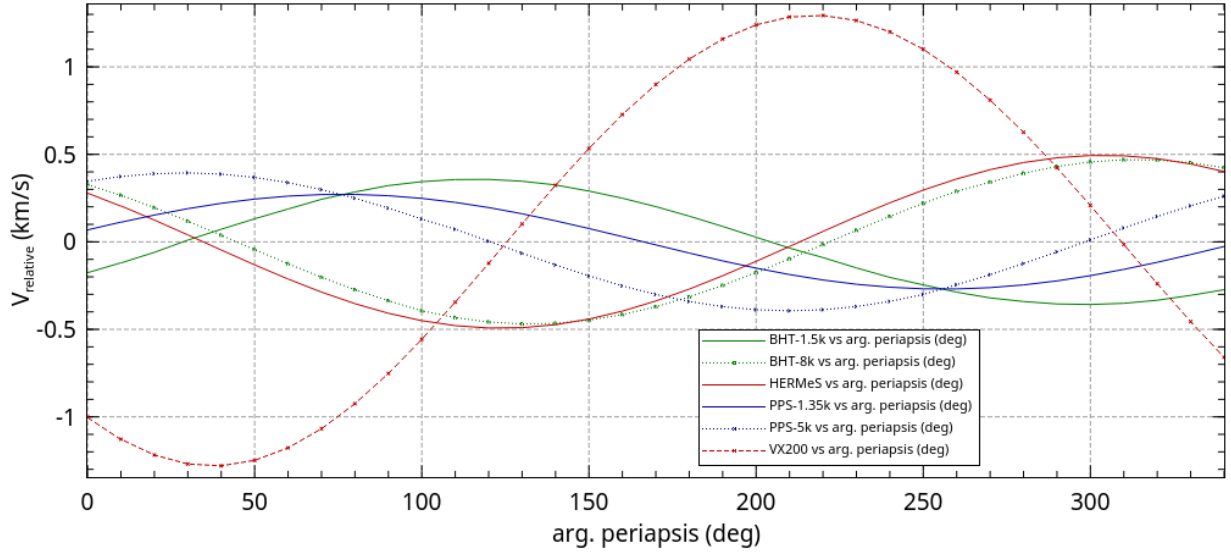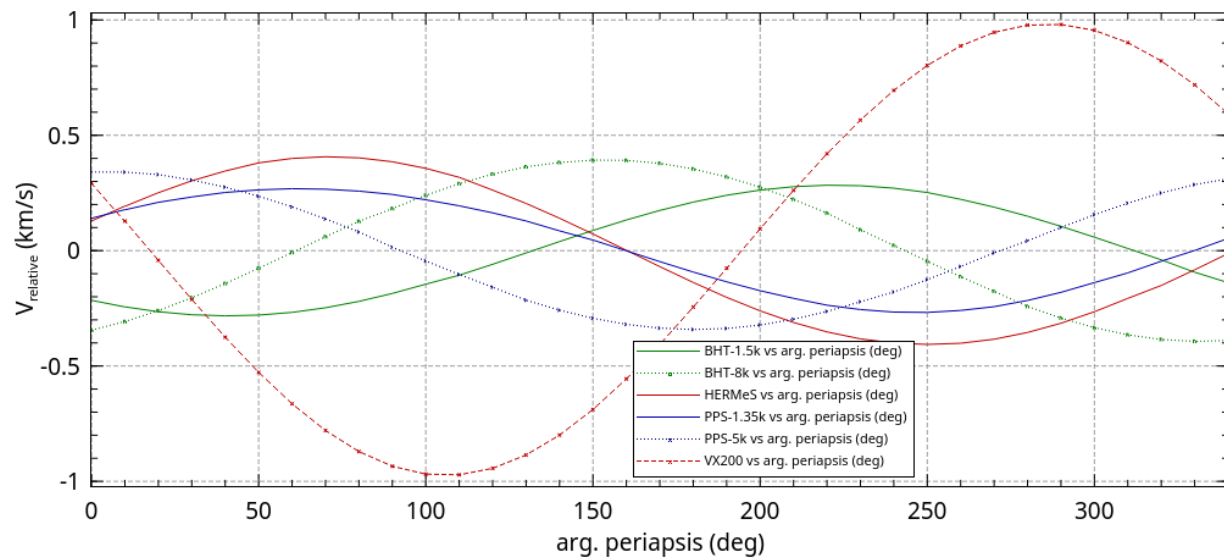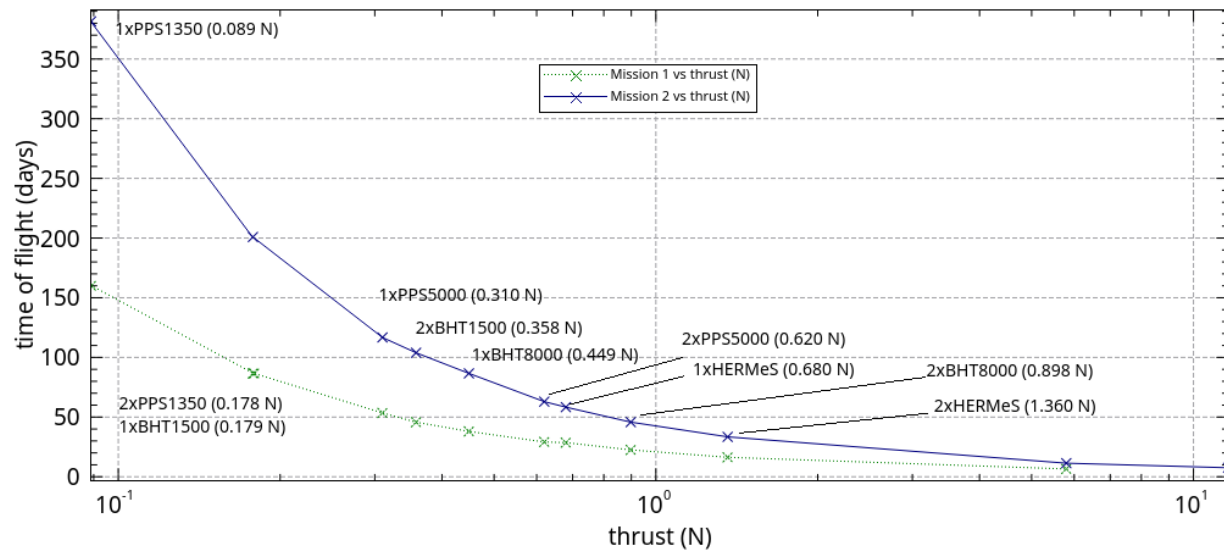
Figure 5.15: Time of flight from MRO injection orbit to Mars SOI: Missions 1 and 2
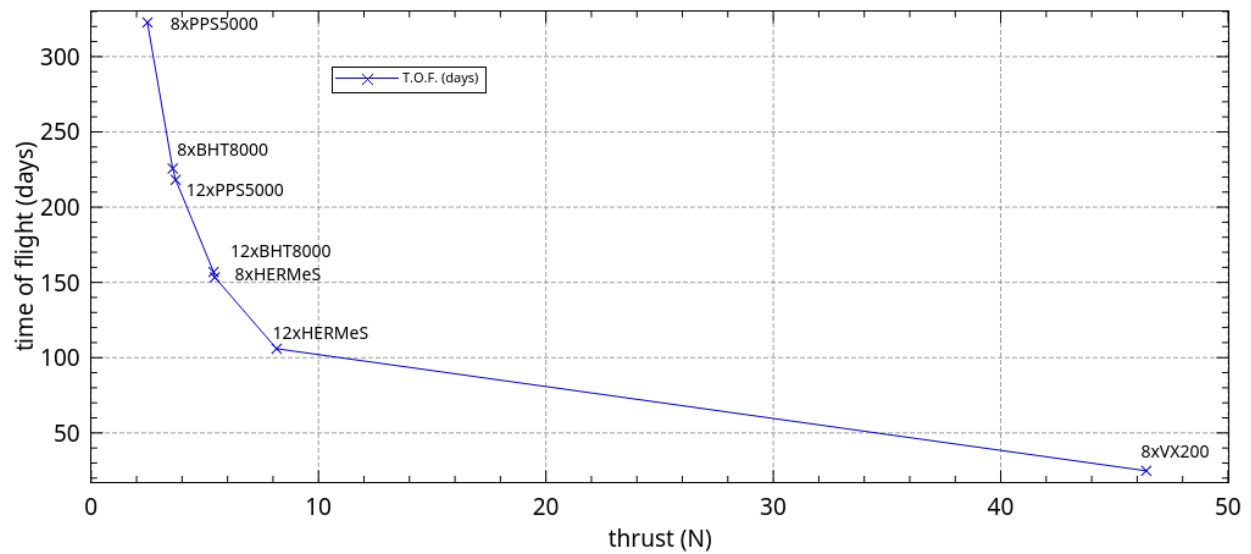
Figure 5.16: Time of flight from MRO injection orbit to Mars SOI: Mission 3

targeted argument of periapsis was that which maximized the excess velocity when leaving Earth and minimized it when leaving Mars. Neither of these scenarios led to useful results as the orbit targeter would not consistently read the sphere of influence of the planet. Figure 5.17 compares the use of the Naasz control laws BHT-8000 Earth departure spirals for missions 2A and 2B, and Figure 5.18 compares the same engine in configurations 3A and 3B departing Mars.

5.4   Interplanetary segment

As per the patched conics approximation, the following scenarios were started at an Earth position and stopped at a radius corresponding to that of Mars around the Sun in the case of the outbound missions, and vice versa for the inbound missions. As a comparison for time of flight, an Hohmann transfer using that criteria would take 256 days. Some scenarios ran out of fuel prior to reaching the target, in which case the value is marked as N/F, for "No fuel." A hard limit on the maximum flight time was set to 3652 days (about ten years): any mission stopped after reaching that propagation time is marked as N/C for "No convergence."

5.4.1   Earth to Mars leg

Constant thrust in the velocity direction

Tables 5.5, 5.7 and 5.9 respectively present the time of flight, the relative velocity of the vehicle at Mars and the fuel usage for each scenario.

The Mars Reconnaissance Orbiter was launched immediately on an interplanetary trajectory and took 175 days to start its approach for orbital injection. As an impulse interplanetary mission design, MRO was launched at the most opportune date with respects to Earth and Mars' synodic period. Similarly, the Mars DRM should allow for a 174 days interplanetary journey for the crew and 202 days for the cargo which precedes the crew arrival by 765 days. [20, Section 4.1]

Concerning the time of flight from Earth to Mars, only the VASIMR takes a shorter or similar amount of time to reach the Mars distance compared to the ideal case of MRO, and so for all hypothetical missions. The synodic period of Earth and Mars is 782 days. Impulse interplanetary
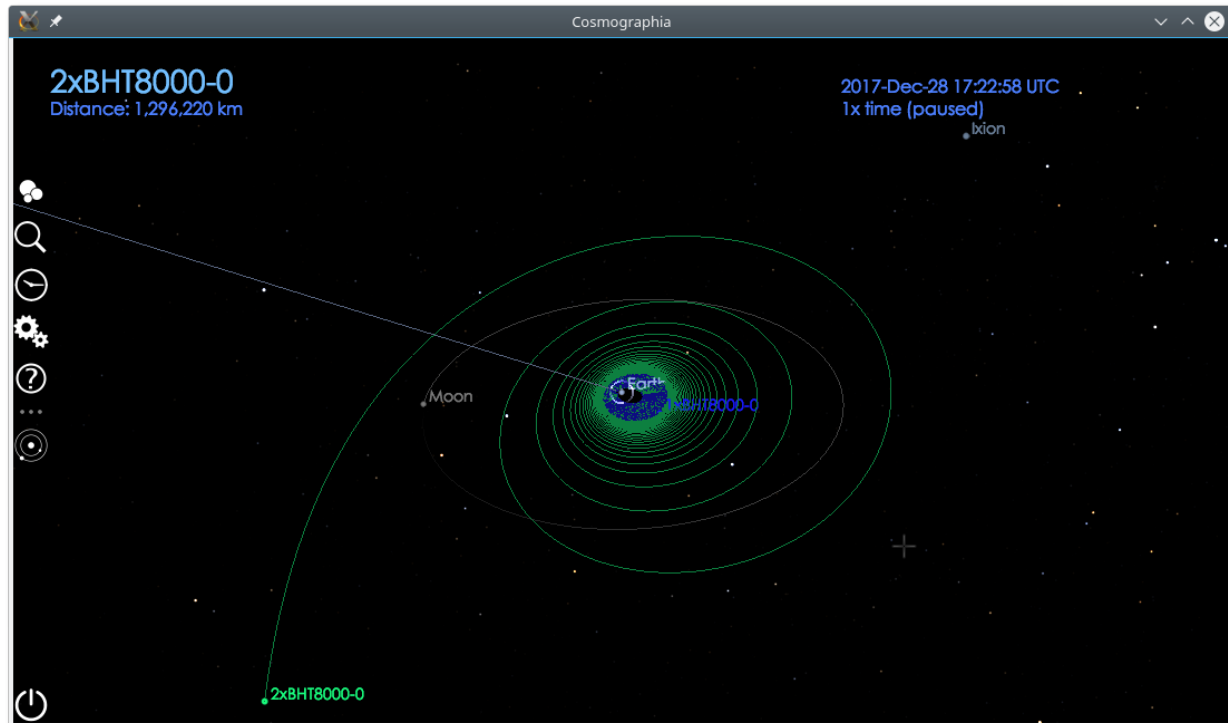
Figure 5.17: Naasz laws for Earth departure using BHT-8000 for missions 2A (blue) and 2B (green)
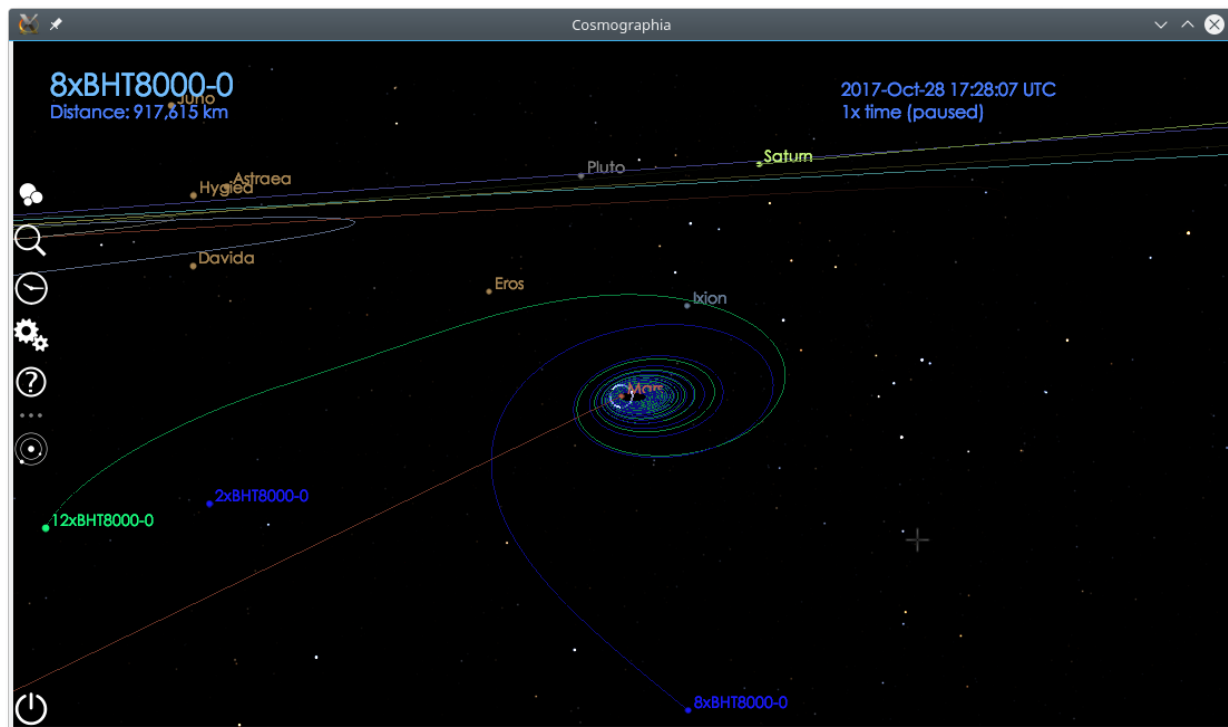
Figure 5.18: Naasz laws for Earth departure using BHT-8000 for missions 3A (blue) and 3B (green)

missions require an approximate 175 day transfer to reach Mars when launched around the most opportune time within a synodic period. Hence, from a competition perspective, it is sensible to assume that any mission duration longer than 903 days would not be interesting for potential customers in terms of time of flight as waiting for the next launch window would allow for the payload to arrive at a similar time. Clusters of HERMeS, and VASMIR allow all their missions to fulfill this time of flight constraint. The BHT-8000 fulfills this constraint on all missions except 3A, in a 8-thruster configuration. The PPS-5000 fulfills this constraint only in the dual-thruster configuration, i.e. missions 1B and 2B. The BHT-1500 requires 810 days to reach Mars in mission 1B, which allows meets the criteria. None of the PPS-1350 configurations meet the 903 day time of flight criteria.

In the case of continuous thrusting, a lower relative velocity at Mars arrival should allow the spacecraft to perform an orbital injection. In impulse interplanetary mission design, the vehicle targets a B-Plane and performs a chemical burn. In the case of continuous thrusting, an equivalent burn for an orbital injection would require a much longer time. Therefore, it is preferable to arrive at destination with a low relative velocity. The VX-200 arrives at Mars with a very high relative velocity in all scenarios. All other engine combinations arrive with less than $1.0\,\mathrm{km/s}$, with the exception of mission 1B equipped with a HERMeS cluster.

Finally, from a fuel usage perspective, missions 1A and 1B provide good insight into the maximum payload mass which can hauled to Mars. In fact, these missions were designed to have almost only fuel mass. We can therefore see that in the worst case, i.e. mission 1B with two VX-200, just over 3000 kg of payload can be transported. In the best case, i.e. mission 1A with a single HERMeS, just over 4000 kg of payload can be hauled to Mars. However, this is a patched conic design where the fuel needed for the orbit raising is not considered. Indirect optimization of continuous thrust trajectories allows to optimize on fuel or time of flight, or a function of both. [12]

Naasz control law

Table 5.5: Interplanetary time of flight of each Mars-bound mission thrusting in the velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Time of flight (days)					
1A	127	1600	682	459	3190	924
1B	95	810	337	255	1576	481
2A	142	2055	850	611	N/C	1206
2B	105	1047	436	303	2024	630
3A	169	N/A	1254	849	N/A	1764
3B	140	N/A	833	598	N/A	1180

In each of the following runs, the Naasz method were used. Only the semi-major axis and the inclination were non-free parameters, and set to those of Mars. The simulations were ran with two waypoints: the first performed the mentioned orbit targeting and the second allowed the spacecraft to cruise, i.e. continue on its orbit without any thrusting, until its radius from the Sun was that of the average distance of Mars from the Sun.

Tables 5.6, 5.8 and 5.10 respectively present the time of flight, the relative velocity of the vehicle at Mars and the fuel usage for each scenario. The time of flight for the Mars-bound missions lead to the exact same results as with a constant thrust in the velocity direction with a 903 day criteria, i.e. acceptable time of flight results are found using all clusters but the PPS-1350, and depending on the mission.

The results from the Naasz efficient correction laws should not directly be compared to those of a continuous thrust in the velocity direction: only the Naasz method corrects for the inclination. Hence, it was expected that the time of flight be slightly longer, and fuel consumption slightly higher since this control law corrects the orbital elements without taking fuel consumption into account. As expected, the relative velocity is less than $1 \, \mathrm{km/s}$ in all cases, apart from mission 1B with the VX-200.

Table 5.6: Interplanetary time of flight of each Mars-bound mission using efficient correction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Time of flight (days)					
1A	160	N/C	701	461	N/C	947
1B	148	828	335	254	N/C	486
2A	168	N/C	871	630	N/C	1273
2B	N/F	1099	436	301	N/C	647
3A	184	N/A	1330	870	N/A	1889
3B	167	N/A	853	616	N/A	1242

5.4.2   Mars to Earth return journey

The initial conditions of the return trajectory are that of the Mars position and velocity at the start of the simulation. As the vehicle is Earth-bound, it must reduce its radius and velocity in order to match that of Earth. As in section 5.4.1, two batches of simulations were ran. The first performs a constant thrust in the anti-velocity direction throughout the simulation until the vehicle reaches the radius of Earth's orbit. The second batch of runs uses the Naasz laws on the semi-major axis and inclination, all other orbital parameters are free. Similarly to the previous section, the Earth position and velocity at arrival is not taken into consideration.

Constant thrust in the anti-velocity direction

In Table 5.11 the simulation results state that doubling the number of thrusters for mission 1 and 2 approximately divides the time of flight by a factor of 2 for all lower thrust engines, i.e. the PPS engines and the BHT-1500. In the case of the other engines, there is no such factor of proportionality. These seemingly surprising results are due to the spacecraft intercepting the Earth's orbit much sooner than in the other case, cf. Figure 5.19.

The synodic period of the Earth and Mars is approximately 781 days. Looking a the time of flight for a round trip, i.e. from Earth to Mars and back, while not counting the orbital injection and orbit raising at both planets, the following missions and engines allow for a round trip to occur in less than one synodic period: all VX-200 for all missions, and BHT-8000 and HERMeS for 1A, 1B, 2B. All other missions require longer than a synodic period to perform a round trip.

Table 5.7: Relative velocity with respect to Mars for each Mars-bound mission thrusting in the velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Relative velocity at Mars (km/s)					
1A	11.031	0.117	0.462	0.371	0.183	0.174
1B	21.063	0.235	0.512	1.464	0.108	0.510
2A	8.431	0.056	0.179	0.545	N/C	0.221
2B	16.681	0.197	0.390	0.684	-0.126	0.045
3A	5.365	N/A	0.200	0.159	N/A	0.189
3B	8.638	N/A	0.193	0.550	N/A	0.222

Via efficient correction of orbital elements

When applying a semi-major axis and inclination efficient correction for the return journey, and keeping all other orbital elements free, only eleven of the 36 simulations reach the the desired orbital radius with any fuel remaining. A frequent issue is for the orbit targeting waypoint to continuously try to correct the orbital elements but the correction leads to an error slightly greater than the tolerance. Figures 5.20 and 5.21 show the example of mission 2B where the spacecraft crosses the radius of the Earth orbit eight months after departure from Mars, but its orbit inclination is not yet correct, and converged to an orbit which crosses that the Earth's. Moreover, performing an efficient correction of the eccentricity as well as the semi major axis and inclination leads to a very slow convergence (nine years in the case of 2B), cf Figure 5.22. Finally, all but mission 3A using a HERMeS cluster results in the Naasz control laws requiring much more time than a constant thrust in the anti-velocity direction.
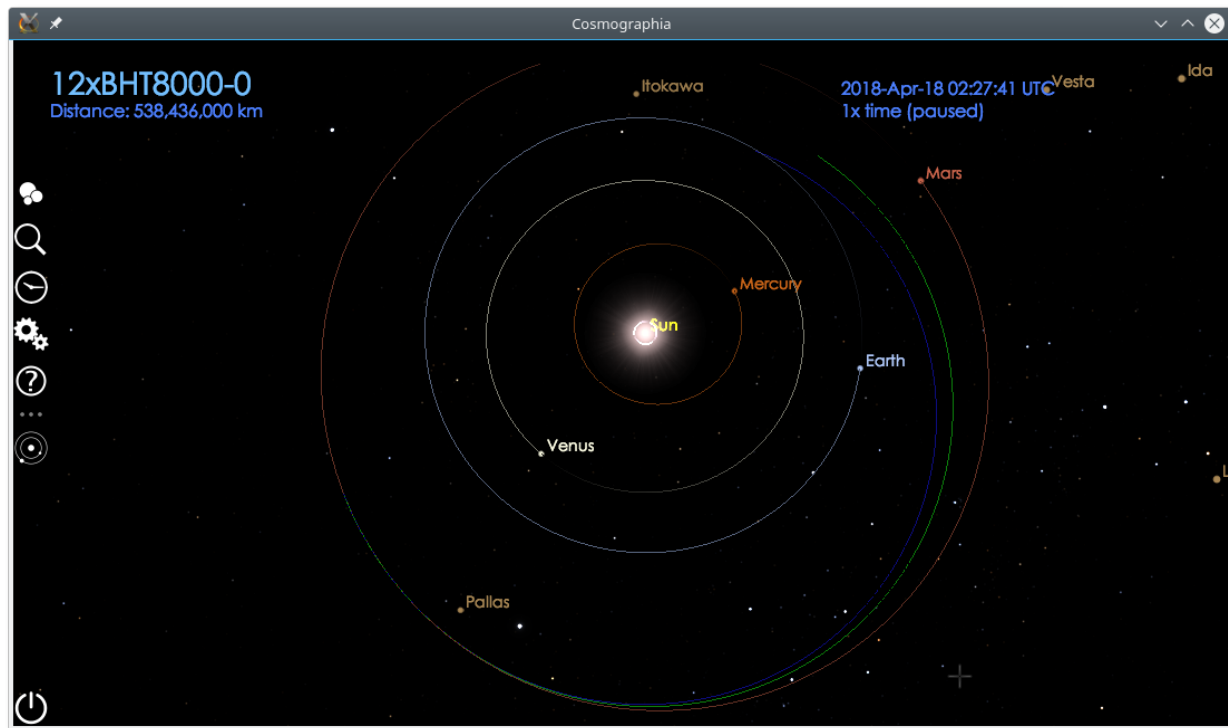
Figure 5.19: Efficiently corrected trajectory of 8xBHT-8000 (green) vs 12xBHT-8000 (blue) for Mission 3
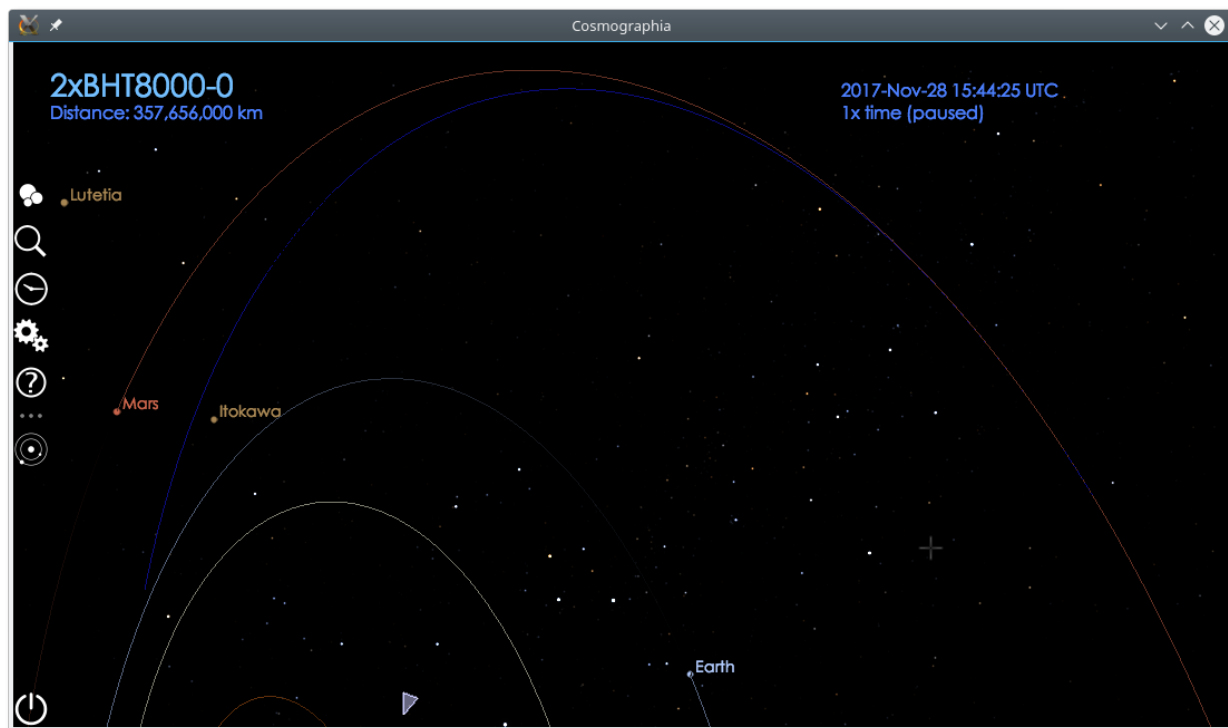

Figure 5.20: Non convergence of the efficiently corrected orbit for return journey due to inclination
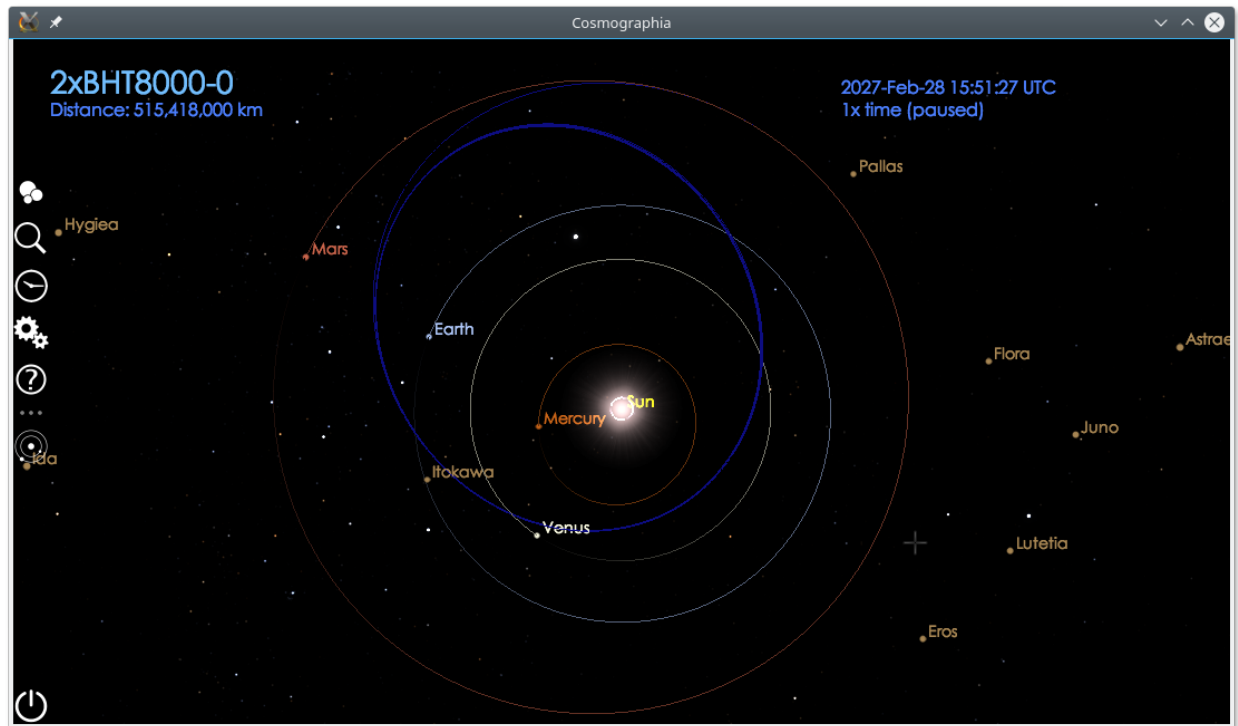
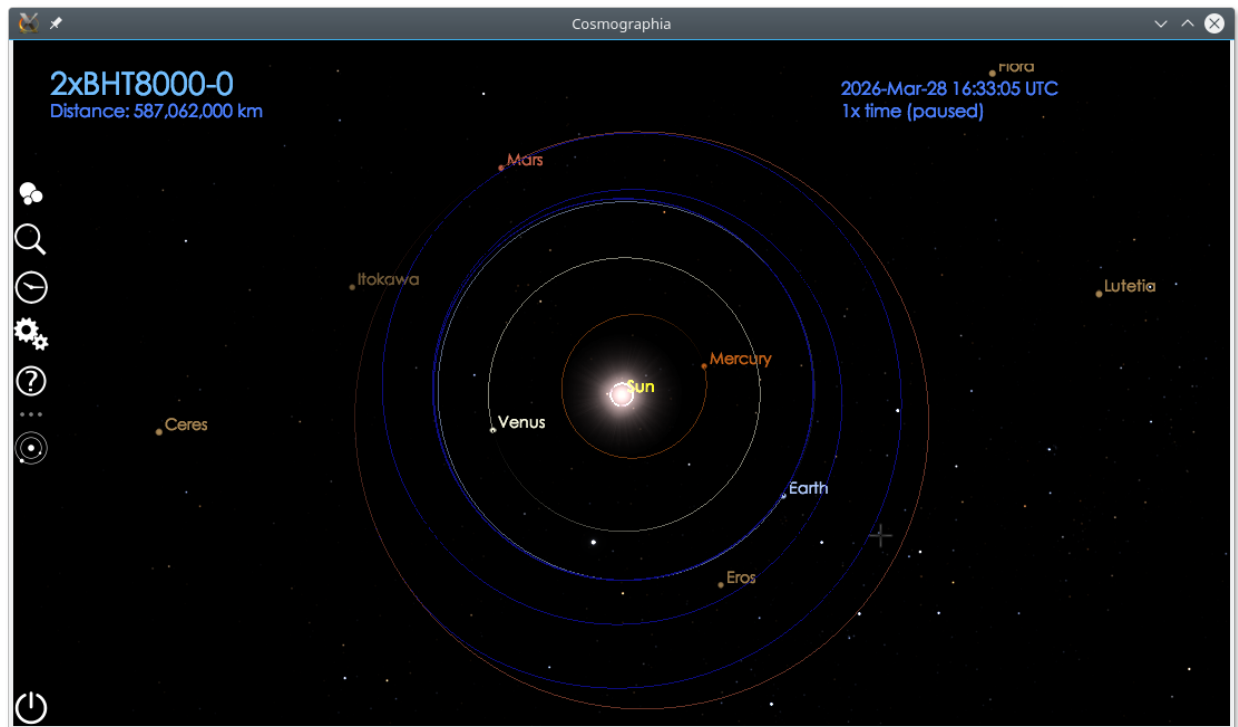Figure 5.21: Non convergence of the Naasz control laws for return journey

Figure 5.22: Slow convergence of the Naasz control laws method with fixed a, i, e for return journey

Table 5.8: Relative velocity with respect to Mars for each outbound mission using Naasz control

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Relative velocity at Mars ($\mathrm{km/s}$)					
1A	1×10^{-5}	N/C	3×10^{-5}	3×10^{-5}	N/C	-3×10^{-5}
1B	-4×10^{-5}	-3×10^{-5}	-3×10^{-5}	-5×10^{-5}	N/C	-3×10^{-5}
2A	-5×10^{-5}	N/C	-3×10^{-5}	-3×10^{-5}	N/C	-3×10^{-5}
2B	N/F	-3×10^{-5}	-3×10^{-5}	-5×10^{-5}	N/C	-3×10^{-5}
3A	-6×10^{-5}	N/A	-3×10^{-5}	-3×10^{-5}	N/A	-3×10^{-5}
3B	-2×10^{-5}	N/A	-3×10^{-5}	-3×10^{-5}	N/A	-3×10^{-5}

Table 5.9: Remaining fuel at Mars for each Mars-bound mission thrusting in the velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Remaining fuel at destination (kg)					
1A	3674.0	3646.6	3779.0	4071.2	3484.1	3597.6
1B	3021.1	3629.7	3793.1	3972.5	3501.7	3538.9
2A	1517.8	1261.9	1477.3	1763.2	N/C	1168.7
2B	793.0	1229.5	1441.0	1775.0	1076.2	1088.9
3A	9925.5	N/A	6029.6	10257.8	N/A	2582.4
3B	6386.7	N/A	6102.2	9476.1	N/A	2506.9

Table 5.10: Remaining fuel at Mars for each Mars-bound mission using Naasz control laws

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Remaining fuel at destination (kg)					
1A	4510.2	N/C	3823.3	4104.5	N/C	3579.5
1B	4513.1	3630.4	3867.1	4161.0	N/C	3613.5
2A	2362.8	N/C	1462.4	1827.2	N/C	1114.6
2B	N/F	1200.5	1501.3	1884.0	N/C	1188.1
3A	16436.9	N/A	5431.0	10060.9	N/A	1373.7
3B	16526.4	N/A	5975.3	10247.8	N/A	1882.3

Table 5.11: Interplanetary time of flight of each Earth-bound mission thrusting in the anti-velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Time of flight (days)					
1A	162	1357	391	336	2476	818
1B	133	441	303	263	993	344
2A	157	947	374	323	2019	789
2B	130	412	292	254	945	331
3A	201	N/A	898	791	N/A	1393
3B	175	N/A	787	366	N/A	883

Table 5.12: Interplanetary time of flight of each Earth-bound mission using Naasz control laws

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Time of flight (days)					
1A	N/F	1701	N/F	471	3212	949
1B	N/F	829	353	N/F	1698	478
2A	N/F	1467	N/F	N/F	2876	848
2B	N/F	880	N/F	N/F	1444	N/F
3A	N/F	N/A	1301	864	N/A	1795
3B	N/F	N/A	849	697	N/A	N/F

Table 5.13: Relative velocity with respect to Earth for each Earth-bound mission thrusting in the anti-velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Relative velocity at Earth ($\mathrm{km/s}$)					
1A	-5.019	0.984	2.433	1.810	1.326	0.823
1B	-4.701	2.601	1.132	0.113	2.180	1.873
2A	-5.231	2.034	2.286	1.602	1.739	0.447
2B	-4.666	2.549	0.863	-0.219	2.017	1.661
3A	-2.541	N/A	1.686	0.544	N/A	1.270
3B	-4.220	N/A	0.455	2.216	N/A	1.536

Table 5.14: Relative velocity with respect to Earth for each Earth-bound mission using Naasz control laws

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Relative velocity at Earth ($\mathrm{km/s}$)					
1A	N/F	9×10^{-5}	N/F	1.078	8×10^{-5}	1.978
1B	N/F	5.311	9.426	N/F	7×10^{-5}	1.021
2A	N/F	0.174	N/F	N/F	3.791	5.805
2B	N/F	3.400	N/F	N/F	1.837	N/F
3A	N/F	N/A	4.270	6.175	N/A	3.502
3B	N/F	N/A	5.956	1×10^{-4}	N/A	N/F

Table 5.15: Remaining fuel at Earth for each Earth-bound mission thrusting in the anti-velocity direction

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Remaining fuel at destination (kg)					
1A	3306.2	3852.2	4299.6	4319.8	3823.0	3758.3
1B	2220.2	4253.9	3915.4	3932.8	4055.6	3955.9
2A	1362.0	2199.0	2329.0	2344.7	1801.5	2040.5
2B	287.6	2303.4	1955.0	1971.6	2101.0	1994.9
3A	7215.4	N/A	11142.0	11181.5	N/A	7079.5
3B	2045.243	N/A	7075.7	15103.8	N/A	7916.0

Table 5.16: Remaining fuel at Earth for each Earth-bound mission using Naasz control laws

Engine	VX-200	BHT-1500	BHT-8000	HERMeS	PPS-1350	PPS-5000
Mission	Remaining fuel at destination (kg)					
1A	N/F	3624.8	N/F	4045.4	3478.4	3559.0
1B	N/F	3596.7	3734.0	N/F	3476.7	3547.8
2A	N/F	1759.1	N/F	N/F	1633.2	1712.8
2B	N/F	1510.4	N/F	N/F	1627.4	N/F
3A	N/F	N/A	5373.1	10003.1	N/A	2211.8
3B	N/F	N/A	5766.7	10003.7	N/A	N/F

Chapter 6

Conclusion

The efficient orbital element correction methods for specific orbit targeting lead to acceptable results around both the Earth and Mars. However, they do not provide good candidates for departing the sphere of influence of Earth and are only moderate candidates for departing the gravity well of Mars. It was expected for continuous thrusting to require longer times of flight compared to a bang-bang control, but the results of realistic scenarios are prohibitively long. The use of specific spiral laws, such as the Pinkham or Lawden spiral, may lead to better results for the escape trajectory.

Moreover, neither the Naasz or Ruggiero efficient correction methods is applicable to interplanetary mission design. They fail to converge on a number of scenario, require more fuel than thrusting in the velocity direction, and often lead to higher relative velocities at arrival. Thrusting in the velocity or anti-velocity direction systematically converges, but does not correct the inclination of the orbit prior to arrival. Concerning the time of flight, only the VASMIR engine leads to similar or better flight times compared to previous Earth to Mars missions, but the relative velocity at arrival at Mars is too high to the spacecraft to perform an orbital injection, weakening the case of this engine. In all cases, the fuel requirements are very high in order for the vehicle to not run out of fuel prior to arriving at destination.

From a business perspective, reusable spacecraft for cargo transportation between different celestial bodies would greatly lower the price to celestial destinations through investment amortization. This research shows that several constraints for a viable business are not yet met with

current technologies. First, in order to be highly competitive with interplanetary rocket launches, the semi-cycler would ideally arrive at destination in a short time than an impulse interplanetary mission. Secondly, the time between Earth departures, i.e. the spacecraft departs Earth, arrives at Mars, returns to Earth and is prepared for next departure, should happen in less than one synodic period. This would allow for the semi-cycler operator to rival in timing with rocket launches. Thirdly, the spacecraft should be able to deliver to Mars at least as much mass as a rocket launch. In terms of time to arrival and time until next departure, all scenarios but the ones using a cluster of PPS-1350 fulfill the above constraints. In addition, several hypothetical engine configurations would likely allow for a round-trip from Earth to Mars without a refueling. Additional simulations should be ran where each leg of the journey does not reset the scenario in order to confirm the likelihood of these missions. Specifically of interest would be missions 1A and 1B as they seems to have the most fuel remaining at the destination of each trajectory leg simulated. However, the time of flight needed to depart Earth's gravity well is prohibitive in most cases, implying that most electric propulsion thrusting technology available today, even clustered, is not suitable for large payload transportation. A possible solution could involve using impulse thrusting for the Earth escape leg, which would save several hundred days of mission time. In terms of mission operations, one could envision refurbishing a chemical propulsion stage used only for to escape the gravity well, supposing enough fuel is kept to reach an Earth orbit after the boost.

Further work could perform global optimizations the trajectory design and specifically include B-plane targeting both at Earth and at Mars. The cost function could be defined either on the fuel mass, the time of flight, or as a function of both. Additionally, there may be other useful highly elliptical orbits around the Earth which are easily accessible from a common launch trajectory. Moreover, a more precise mission design could skip the patched conics approximation while also taking into account third body perturbations. Finally, a precise mass estimate of the tugging spacecraft should be considered, and the power requirements also be taken into consideration both for the mass estimates of the power subsystem and for availability of power at each time step of integration.

Bibliography

[1] AdAstra. Vasimr operating principles. `http://www.adastrarocket.com/HiResImagesForPublicRelease/VASIMR_operating_principles.jpg`. [Online; accessed 20-March-2017, Copyright Ad Astra Rocket Company ©all rights reserved].

[2] Mars Architecture Steering Group Addendum. Human Exploration of Mars - Design Reference Mission 5 Addendum. NASA Johnson Space Center, 2009.

[3] Buzz Aldrin. Cyclic trajectory concepts. In Interplanetary Rapid Transit Study, JPL, 1985.

[4] Tyler R Anderson. Hyperbolic rendezvous for proposed earth-mars cyclers. In AIAA/AAS Astrodynamics Specialist Conference, page 5267, 2016.

[5] SpaceX News article. Reusability: The key to making human life multi-planetary. `http://www.spacex.com/news/2013/03/31/reusability-key-making-human-life-multi-planetary`, 2015. [Online; accessed 20-March-2017].

[6] Rick Hudson Austin Clements. Proposal: Eliminate stw stack re-scanning. `https://github.com/golang/proposal/blob/master/design/17503-eliminate-rescan.md`, 2016. [Online; accessed 19-March-2017].

[7] The Golang authors. The go memory model. `https://golang.org/ref/mem`, May 2014. [Online; accessed 19-March-2017].

[8] The Kst authors. Kst - Visualize your data. `https://kst-plot.kde.org/`.

[9] Sam Boyer. The saga of go dependency management. `https://blog.gopheracademy.com/advent-2016/saga-go-dependency-management/`, December 2016. [Online; accessed 19-March-2017].

[10] Brendan Tracey, et al. Gonum github organization page. `https://github.com/gonum`. [Online; accessed 19-March-2017].

[11] Georges H. W. Bush. Remarks on the 20th anniversary of the apollo 11 moon landing. `https://bush41library.tamu.edu/archives/public-papers/712`, 1989. [Online; accessed 19-March-2017].

[12] MIRKO CATUCCI. Indirect optimization of time-fixed space transfers with variable specific impulse engine. 2014.

[13] Nicolas Arcis Claude Boniface. An overview of electric propulsion activities at cnes. In 34th International Electric Propulsion Conference, IEPC-2015-05, Kobe-Hyogo, Japan, 2015.

[14] Busek Corporation. Bht-1500 hall effect thruster datasheet. `http://www.busek.com/index_htm_files/70000702%20BHT-1500%20Data%20Sheet%20Rev.pdf`. [Online; accessed 26-March-2017].

[15] Busek Corporation. Bht-8000 hall effect thruster datasheet. `http://www.busek.com/index_htm_files/70000703%20BHT-8000%20Data%20Sheet%20Rev-.pdf`. [Online; accessed 26-March-2017].

[16] D'Amario et al. Galileo trajectory design, 1992.

[17] Airbus Defense and Space. Solution to reuse space launchers. `https://airbusdefenceandspace.com/reuse-launchers/`. [Online; accessed 20-March-2017].

[18] ESA. Electric Spacecraft Propulsion [on SMART-1]. `http://sci.esa.int/smart-1/34201-electric-spacecraft-propulsion/?fbodylongid=1535`, 2012. [Online; accessed 20-March-2017].

[19] Github Inc. Github user statistics. `https://github.com/blog/2345-celebrating-nine-years-of-github-with-an-anniversary-sale`. [Online; accessed 17-April-2017].

[20] Mars Architecture Steering Group. Human Exploration of Mars - Design Reference Mission 5. NASA Johnson Space Center, 2009.

[21] Nobel Ariel Hatten. A critical evaluation of modern low-thrust feedback-driven spacecraft control laws. Master's thesis, The University of Texas at Austin, 2012.

[22] NASA History. Daniel saul goldin. `https://history.nasa.gov/dan_goldin.html`, 2005. [Online; accessed 20-March-2017].

[23] Rick Hudson. Go gc [garbage collection]: Latency problem solved. In GopherCon Denver, July 8, 2015, 2015. [Online; accessed 19-March-2017].

[24] James R. Wertz, et al. Space Mission Design Engineering: The New SMAD. Space Technology Library, Microcosm Press, 2011.

[25] NASA JPL. Technology: Ion propulsion. `http://dawn.jpl.nasa.gov/technology/ion_prop.asp`. [Online; accessed 19-March-2017].

[26] Damon F Landau and James M Longuski. Guidance strategy for hyperbolic rendezvous. Journal of guidance, control, and dynamics, 30(4):1209–1213, 2007.

[27] User 'Lasunncty'. Keplerian Orbital Elements. `https://commons.wikimedia.org/w/index.php?title=File:Orbit1.svg&oldid=230891706`, 2010. [Online; accessed 16-April-2017, licensed as Creative Commons Attribution-Share Alike 3.0 Unported].

[28] Airbus Safran Launchers. Chemical bi-propellant thrusters family. `http://www.space-propulsion.com/brochures/bipropellant-thrusters/bipropellant-thrusters.pdf`. [Online; accessed 25-March-2017].

[29] Doug Messier. Pulsed Plasma Thrusters. `http://www.parabolicarc.com/2015/03/31/ad-astra-rocket-company-wins-nasa-propulsion-contract/`, 2015. [Online; accessed 20-March-2017].

[30] Elon Musk. Making humans a multiplanetary species. In IAC 2016, Mexico, 2016.

[31] Bo J Naasz. Classical element feedback control for spacecraft orbital maneuvers. Master's thesis, Virginia Polytechnic Institute and State University, 2002.

[32] NAIF and Chris Laurel. SPICE-enhanced Cosmographia User's Guide. `https://cosmoguide.org/`.

[33] NASA. Pulsed Plasma Thrusters. `https://www.nasa.gov/centers/glenn/about/fs23grc.html`, 2004. [Online; accessed 20-March-2017].

[34] NASA. Mars reconnaissance orbiter launch (press kit). `https://www.nasa.gov/pdf/124378main_mro-launch-Aug051.pdf`, 2005. [Online; accessed 19-March-2017].

[35] Andres Dono Perez, Oriol Tintore Gazulla, George Lewis Teel, Nghia Mai, Joseph Lukas, Sumadra Haque, Eddie Uribe, Michael Keidar, and Elwood Agasid. Modular pulsed plasma electric propulsion system for cubesats. 2014.

[36] Peter Y Peterson, Hani Kamhawi, Wensheng Huang, George Williams, James H Gilland, John Yim, Richard R Hofer, and Daniel A Herman. Nasa's hermes hall thruster electrical configuration characterization. In 52nd AIAA/SAE/ASEE Joint Propulsion Conference, page 5027, 2016.

[37] Anastassios Petropoulos. Low-thrust orbit transfers using candidate lyapunov functions with a mechanism for coasting. In AIAA/AAS Astrodynamics Specialist Conference and Exhibit, page 5089, 2004.

[38] Anastassios E Petropoulos. Simple control laws for low-thrust orbit transfers. In AIAA/AAS Astrodynamics Specialist Conference. Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2003, 2003.

[39] Anastassios E Petropoulos. Refinements of the q-law for low-thrust orbit transfers. In 15th AAS/AIAA Space Flight Mechanics Conference, 2005.

[40] Anastassios E Petropoulos and Seungwon Lee. Optimisation of low-thrust orbit transfers using the q-law for the initial guess. In AAS/AIAA Astrodynamics Specialist Conference. Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2005., 2005.

[41] Rob Pike. Go at google: Language design in the service of software engineering. `https://talks.golang.org/2012/splash.article`, 2012. [Online; accessed 19-March-2017].

[42] David S. F. Portree. Humans to Mars: Fifty Years of Mission Planning, 1950-2000. NASA History Office, 2001.

[43] Christopher Rabotin. Space Mission Design - A continuous thrust interplanetary mission propagator and vizualizer. `https://github.com/ChristopherRabotin/smd`.

[44] Christopher Rabotin. gokalman (doi: 10.6084/m9.figshare.4765366.v1). `https://figshare.com/articles/gokalman/4765366`, March 2017.

[45] Christopher Rabotin. ode (10.6084/m9.figshare.4765021.v1). `https://figshare.com/articles/ode/4765021`, March 2017.

[46] Christopher Rabotin. smd (doi: 10.6084/m9.figshare.4765018.v1). `https://figshare.com/articles/smd/4765018`, March 2017.

[47] Oona Raisanen. Electrostatic ion thruster. `https://commons.wikimedia.org/w/index.php?title=File:Electrostatic_ion_thruster-en.svg&oldid=158672186`, 2012. [Online; accessed 20-March-2017, licensed as Creative Commons Attribution-Share Alike 3.0 Unported].

[48] Justin Ray. Ula unveils its future with the vulcan rocket family. `https://spaceflightnow.com/2015/04/13/ula-unveils-its-future-with-the-vulcan-rocket-family/`, 2015. [Online; accessed 20-March-2017].

[49] Robert G. Jahn, Edgar Y. Choueiri. Encyclopedia of Physical Science and Technology 3rd Ed., Vol. 5 - Electric Propulsion. Princeton University, 2003.

[50] Paola Rossetti, Massimo Saverdi, and Leonardo Biagioni. Electric-propulsion systems. PROGRESS IN ASTRONAUTICS AND AERONAUTICS, 223:223, 2008.

[51] A Ruggiero, P Pergola, S Marcuccio, and M Andrenucci. Low-thrust maneuvers for the efficient correction of orbital elements. In 32nd International Electric Propulsion Conference, pages 11–15, 2011.

[52] Sam Boyer, Andrew Gerrand. dep roadmap. `https://github.com/golang/dep/wiki/Roadmap`, March 2017. [Online; accessed 19-March-2017].

[53] Jared P Squire, Chris S Olsen, FR Chang Díaz, Leonard D Cassady, Benjamin W Longmier, Maxwell G Ballenger, Mark D Carter, Tim W Glover, Greg E McCaskill, and EA Bering III. Vasimr® vx-200 operation at 200 kw and plume measurements: future plans and an iss ep test platform. In 32nd International Electric Propulsion Conference, 2011.

[54] Blue Origin Press team. Our approach to technology. `https://www.blueorigin.com/technology`. [Online; accessed 20-March-2017].

[55] SpaceX Press team. Reusability. `http://www.spacex.com/reusability-key-making-human-life-multi-planetary`. [Online; accessed 20-March-2017].

[56] ULA. Transportation enabling a robust cislunar space economy. `http://www.ulalaunch.com/uploads/docs/Published_Papers/Commercial_Space/2016_Cislunar.pdf`, 2016. [Online; accessed 20-March-2017].

[57] David A. Vallado. Fundamentals of Astrodynamics and Applications, 4th edition. Space Technology Library, Microcosm Press, 2013.

[58] Various contributors. Computer language benchmarks game. `http://benchmarksgame.alioth.debian.org/u64q/which-programs-are-fastest.html`. [Online; accessed 27-March-2017].

[59] V Vial, N Cornu, E Coulaud, and D Arrat. Pps® ng: Hall effect thruster for next generation spacecraft. In 32th International Electric Propulsion Conference, IEPC-2011-120, Wiesbaden, Germany, 2011.

[60] Werner von Braun. The Mars Project (original title: Das Marsprojekt). University of Illinois Press, 1953.

[61] Wiley J. Larson, Linda K. Pranke. Human Spaceflight: Mission Analysis and Design. McGraw-Hill, 1999.

[62] B. Zubrin. Mars direct, humans to the red planet by 1999. In 41st Congress of the International Astronautical Federation, 1990.

[63] Zurbach, et al. A 20kw high power hall effect thruster for exploration. In IAC 2010, 2010.