UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

MULTI-OBJECTIVE OPTIMISATION OF IMPULSIVE ORBITAL TRAJECTORIES

TRABAJO FINAL DEL

Grado en Ingeniería Aeroespacial

REALIZADO POR

José Tatay Sangüesa

TUTORIZADO POR

Santiago Emmanuel Moll López
José Antonio Moraño Fernández
Erika Vega Fleitas

FECHA: **Valencia, julio, 2019**

Multiobjective Optimisation of Impulsive Orbital Trajectories

José Tatay Sangüesa

July 2019

Abstract

Orbit transfer amounts to around 70% of all the propellant consumption in a satellite's mission lifetime. With the quantity of launches that are taking place nowadays, and those expected for the future, optimisation in this aspect has become a priority.

This project develops a transfer orbit optimiser that can offer quick and accurate results to help in the choice of the mission transfer orbit. Only from the keplerian elements of the initial and final orbits, it performs a multiobjective optimisation, and returns the optimal region relating propellant and time of flight.

Once this relation is plotted, the optimal point that best suits our mission priorities is chosen. The optimiser will return all the necessary data to perform the transfer, together with a representative plot of the orbit itself.

In order to build the optimiser, the relative two-body model and impulsive manoeuvres are assumed, although up to four impulses may be performed. Moreover, Lambert's problem is used to model the transfer, so that the optimisation dimensions can be reduced to the maximum extent, thus reducing the computational cost.

Regarding the optimisation algorithm used, gradient-based (SQP) and global search methods (Genetic Algorithm) are compared using a simple Hohmann transfer problem. Finally, the genetic algorithm is chosen and a brief discussion on its most relevant characteristics takes place.

Keywords: orbit transfer, multiobjective optimisation, Δv, time of flight, impulsive manoeuvres, Lambert's problem, genetic algorithm, orbit plotting.

Resumen

Las transferencias orbitales suponen alrededor de un 70% de todo el combustible presente en la vida útil de un satélite. Con la cantidad de lanzamientos que están teniendo lugar recientemente, y los que se esperan en los próximos años, la optimización en este ámbito se ha convertido en una prioridad.

Este proyecto desarrolla un optimizador de transferencias orbitales que ofrece resultados rápidos y precisos para ayudar en la elección de la órbita de transferencia necesaria para llevar a cabo la misión. Únicamente con los elementos orbitales keplerianos de la órbita inicial y final, el programa realiza un optimización multi-objetivo y devuelve una representación de la región óptima, relacionando el combustible total y el tiempo de vuelo.

A través del gráfico de la región óptima, se escoge el punto óptimo que mejor se ajusta a las prioridades de nuestra misión. El programa devuelve entonces todos los datos necesarios para llevar a cabo la transferencia, incluyendo una representación gráfica de la órbita.

Para construir el optimizador, se asume la validez del modelo relativo de dos cuerpos y trayectorias impulsivas, aunque se podrían realizar hasta cuatro impulsos. Además, para la modelización de la transferencia se emplea el problema de Lambert, reduciendo así las dimensiones del problema de optimización y el tiempo de cálculo.

Por último, en, al algoritmo de optimización, se estudian métodos basados en gradientes (SQP) y métodos de búsqueda global (Algoritmo Genético), y se comparan empleando una transferencia de Hohmann. Finalmente, se escoge el algoritmo genético y se discute brevemente la elección de sus características más relevantes.

Palabras clave: transferencia orbital, optimización multi-objetivo, tiempo de vuelo, Δv, maniobras impulsivas, problema de Lambert, algoritmo genético, representación de órbitas.

Resum

Les transferències orbitals suposen al voltant d'un 70% de tot el combustible present en la vida útil d'un satèl·lit. Amb la quantitat de llançaments que estan tenint lloc recentment, i tots els que s'esperen en els pròxims anys, l'optimització en aquest àmbit s'ha convertit en una prioritat.

Aquest projecte desenvolupa un optimitzador de transferències orbitals que oferix resultats ràpids i precisos per a ajudar en l'elecció de l'òrbita de transferència necessària per dur a terme la missió. Únicament amb els elements orbitals keplerians de l'òrbita inicial i final, el programa realitza una optimització multi-objectiu i torna una representació de la regió òptima, relacionant el combustible total i el temps de vol.

A través del gràfic de la regió òptima, es tria el punt òptim que millor s'adapta a les prioritats de la nostra missió. El programa torna totes les dades necessàries per a dur a terme la transferència, incloent una representació gràfica de l'òrbita.

Per a construir l'optimitzador, s'assumix la validesa del model relatiu de dos cossos i trajectòries impulsives, tot i que es considera que fins a quatre impulsos podrien ser realitzats. A més a més, el problema de Lambert s'utlitza per model·lar la transferència, amb el que s'aconseguix reduir les dimension del problema d'optimització, y es reduïx el tems de càlcul.

Respecte del algoritme d'optimització, s'estudien mètodes basats en gradients (SQP) i mètodes de busca global (Algoritme Genètic), i es comparen emprant una transferència de Hohmann. Finalment, s'escollix l'algoritme genètic i es discutix breument l'elecció de les seues característiques més rellevants.

Paraules clau: transferència orbital, optimització multi-objectiu, temps de vol, Δv, maniobres impulsives, problema de Lambert, algoritme genètic, representació d'òrbites.

Acknowledgements

I would like to express sincere gratitude to my thesis supervisor Santiago Emmanuel Moll López for his guidance throughout the project.

I would also like to thank professors Kathleen C. Howell and William Crossley, from Purdue University, for teaching me the necessary concepts to perform this work in such a great manner. Also, I want to thank Universitat Politècnica de València for setting the foundations of my technical knowledge all this years.

Very special thanks go to Luisa Cuadrado, who woke up in me the love for mathematics and has always pushed me harder and made me strive.

I can never thank enough my family, especially my parents, whose endless love, support and sacrificies have enabled not only this thesis completion, but all of my academic formation.

Last, but certainly not least, I want to mention my friends: Marc, Irene, Dani, Luis, Andriy, María, Paula, Alex, Blanca and Juanse. Their support and company throughout my life have made me who am I today, and I will always be grateful.

Valencia, June 2019

José Tatay Sangüesa

Contents

List of Figures

List of Tables

List of Symbols

Δv	Velocity change, measure of the total propellant
γ	Flight path angle
$\mathbf{P}$	Orbit period
μ	Orbital constant $= G * (m_s + m_{Earth})$
Ω	Right Ascension of the Ascending Node (RAAN)
ω	Argument of periapsis
$\overline{r}$	Position vector
θ^*	True anomaly
ε	Orbit total energy
a	Semi-major axis
b_i	Number of bits
e	Eccentricity
F_D	Drag force
F_G	Gravitational force
F_T	Thrust force
G	Gravitational constant $= 6.67408 * 10^{-11} m^3 kg^{-1} s^{-1}$
h	Specific angular momentum
i	Inclination
m_{Earth}	Earth's mass
m_s	Spacecraft's mass
p	Semi-latus rectum
r	Distance
r_a	Radius of the apoapsis
r_p	Radius of the periapsis
v	Velocity
X_0	Referring to the initial orbit
X_f	Referring to the final orbit

x_i^L	Lower bound
x_i^U	Upper bound
X_{t_i}	Referring to transfer arc $_i$

List of Abbreviations

BSA Bit String Affinity.

ECI Earth Centred Inertial.

GA Genetic Algorithm.

GEO Geostationary Earth Orbit.

GUI Graphic User Interface.

LEO Low Earth Orbit.

MEO Medium Earth Orbit.

RAAN Right Ascension of the Ascending Node.

SQP Sequential Quadratic Programming.

TOF Time Of Flight.

1 Introduction

1.1 Motivation

Space exploration is becoming one of the fastest developing research areas in recent times. It seems inevitable for humans to end up establishing colonies in the Moon, Mars or other celestial bodies. However, the cost of sending a rocket with meaningful payload into interplanetary flight projects involves spending millions of euros, so the margin of error is certainly slim.

Nevertheless, the colonisation of other planets or moons is not the only possibility that space offers. In fact, the most common space missions are designed to improve the quality of life of the Earth population. Artificial satellites are launched every month with communications, remote sensing or navigation purposes, among others. The advances in reusable launch vehicles (Space X, Blue Origin) have made considerably cheap to send satellites into orbit. Plus, with the development of CubeSats, owning an orbiting satellite has become available to a greater public, including small companies, universities or even high schools.

Launch vehicles are responsible for lifting the satellites outside from the Earth's atmosphere and carrying them to orbit. However, they are able to take the spacecraft to the desired region in space with a restricted accuracy. Plus, it is common that in one vehicle several satellites are launched, each with different orbital requirements. Hence, after the separation of the spacecraft from the launch vehicle there are some operations that the satellite must be able to perform in order to accomplish its mission. These operations can be described as follows:

1. **Orbit transfer:** Modifying the spacecraft's initial orbit to achieve the desired orbit. Usually, to improve the accuracy reached by the launch vehicle or to rendezvous with another body.

2. **Orbit maintenance:** Compensate the orbit perturbations to keep the spacecraft in the desired mission orbit.

3. **Attitude control:** Orient the spacecraft so that it can correctly perform its mission. Actually, this operation is performed with additional actuators and does not usually require propellant.

4. **De-orbiting:** Orbital manoeuvres performed at the spacecraft's end of life to leave the orbit or destroy the satellite.

According to the literature [1], orbit transfer is the operation that consumes more propellant, reaching up to 70% of the total Δv needed to perform the mission, as seen in Table 1.

As it can be seen, the minimisation of orbit transfer propellant requirements would have very positive consequences in satellite deployment mission. The savings in propellant could mean either lower mass that needs to be transported into space and thus, lower cost; or it could be used for orbit maintenance purposes, increasing the satellite's lifetime. Nonetheless, it is important to consider

Table 1: Propellant Consumption % of a Geostationary Satellite

Operation	**Propellant Consumption (%)**
Orbit Transfer	70.0
Orbit Maintenance	29.6
De-orbiting	0.4

other variables besides the propellant used, for instance, the Time Of Flight (TOF) taken to complete the transfer.

Classical orbit transfers are only able to provide optimal results at very specific conditions, whereas in the vast majority of the cases there exists no classical optimal orbit transfer strategy. Although at some cases superposition of classical orbit transfers could be used, this does not guarantee the optimal result. Hence, the aim of this project is to develop and implement a program able to provide optimal solutions, in terms of propellant and time of flight, for any orbit transfer in the near-Earth region.

1.2 Scope and Project Outline

This project describes the implementation and functioning of an orbit transfer optimiser. The program provides an accurate, reliable and intuitive way to obtain the optimal Δv vs. time of flight distribution plot so that the user can decide the compromise that is willing to accept for the mission. It has been prepared to consider up to four-impulse transfers, as it is believed that this is the maximum number of impulses needed for a 3D optimal transfer [2, 3, 4]. Moreover, the algorithm is able to handle constraints and it will ensure that no orbit will place the spacecraft closer to 200km from the Earth's surface, to avoid collision danger. Finally, the calculations have been made under the restrictive assumptions of: an isolated two-body system with perfect centrobaric bodies, the Newtonian gravity model, and impulsive manoeuvres. All of this simplifications and some other will be commented in further sections.

Regarding the project outline, the following sections can be encountered:

- Theoretical background of the orbital motion model and the optimisation algorithms used, including multiobjective optimisation.
- Explanation of the methodology used and the problem setting.
- Comparison of different optimisation algorithms by applying them to problems with known solutions.
- Program presentation with a real transfer example. Results discussion.
- Budgeting on the project total cost.
- Project conclusions including comparison with other orbit optimisers and future work discussion.

2 Theoretical Model

2.1 Orbital Mechanics

Satellites and spacecrafts move in orbits through space. In this section, some basic concepts on orbital motion will be introduced, including the solution for the restricted two-body problem. We will start from the general orbital motion formula development and particularise it to our problem.

Once the general motion of a spacecraft is known, the parameters that describe the orbit's shape and size will be discussed, as they will used to represent the spacecraft's trajectory in space, together with its current position and velocity.

2.1.1 Orbital Motion

In order to develop the orbital motion formulas, first we need to define the coordinate system. The Earth Centred Inertial (ECI) reference frame will be employed. This system uses the Earth's centre as the origin of coordinates, and the Equator to determine the fundamental plane. Moreover, the principal direction is chosen to coincide with the vernal equinox, which is obtained by drawing a line between the Earth and the Sun in the first day of spring [5]. Figure 1 shows the reference system used, which is also known as geocentric-equatorial coordinate frame.

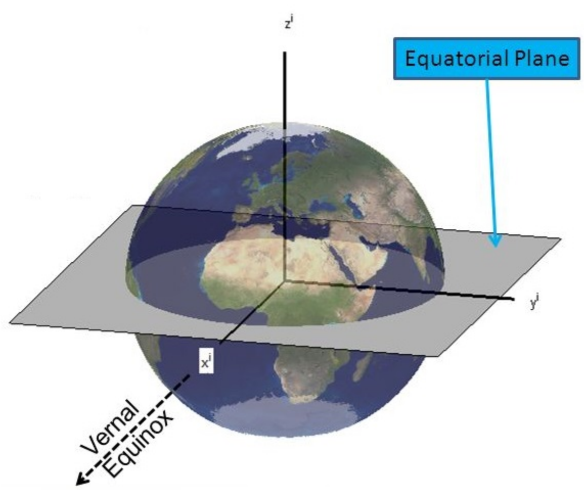

Figure 1: ECI Reference Frame [6]

Furthermore, the simplifying assumptions need to be stated. It is important to note that these assumptions allow us to solve the problem in a simpler manner, but also restrict the reliability of our solutions to the extent that they can be applied. The main assumptions are enumerated below.

1. The ECI reference frame is sufficiently inertial. Thus, Newton's laws apply.
2. The spacecraft's mass is negligible compared to the Earth's, $m_s \ll m_{Earth}$.
3. Earth is perfectly spherical and with uniform density, so we can consider it a centrobaric body, i.e., a point mass. Hence, the Newtonian gravity model applies, which is represented by Equation 1.

$$\overline{F_G} = -G\frac{m_1 m_2}{r^3}\overline{r} \tag{1}$$

4. Manoeuvres will be considered impulsive, hence no thrust force will be applied.
5. The spacecraft is located high enough above the Earth's atmosphere that the drag force can be neglected, $F_D \approx 0$.
6. Other forces such as those due to solar radiation or electromagnetic fields are negligible. Therefore, the only acting force is gravity F_G.
7. The spacecraft's mass m_s is constant, so $\Delta m_s = 0$.

Applying all of these assumptions to Newton's Second Law we obtain the restricted n-body equation of motion, represented in Equation 2. As it can be seen, this is a differential equation, which needs to be solved to obtain the spacecraft's trajectory.

$$m_i\ddot{\overline{r_i}} = -G\sum_{\substack{j=1 \\ j\neq i}}^{n} \frac{m_i m_j}{r_{ij}^3}\overline{r_{ij}} \tag{2}$$

Analysing this equation, it can be seen that we require a total of $6n$ integrals of the motion to completely solve the n-body problem. This results from the number of equations present: n bodies $\times$ 3 dimensions $\times$ 2 nd order eq. $= 6n$. If it is further assumed that the spacecraft is moving very close to Earth, the gravitational influences of other bodies may be neglected, leaving the number of necessary constants to 12. Unfortunately, since Euler's time (1707-1783) only ten integrals of the motion are known, which come from the conservation of linear momentum (6), angular momentum (3), and total energy (1) [7]. This means that the restricted two-body problem is not solvable in this form, as two additional constants would be needed. In order to solve this problem we need to rewrite Equation 2 to express the relative motion of the two bodies involved, represented by vector $\overline{r}$ and its derivatives. The new Equation 3 is now solvable, although its solution is not trivial.

$$\ddot{\overline{r}} + \mu\frac{\overline{r}}{r^3} = 0 \tag{3}$$

where $\mu = G(m_1 + m_2)$.

The solution of Equation 3 requires the Kepler laws together with some calculus and geometry notions and can be found in several references [8, 9, 10, 11]. Its solution can be found in Equation 4 and expresses a relation for the magnitude of the position vector.

$$r = \frac{k_1}{1 + k_2 \cos \theta^*} \tag{4}$$

where θ^* is the polar angle from the orbits periapsis to the spacecraft's location. k_1 and k_2 are constants that depend on the orbit's shape and size, which are characterised by the orbital elements. Actually, $k_1 = p$ and $k_2 = e$.

Equation 4 is known as the conic equation because its solutions represent one of the conic sections (depending on the constants). The conic sections are the circle, the ellipse, the parabola and the hyperbola and arise from the different planes that can intersect a cone, as seen in Figure 2.

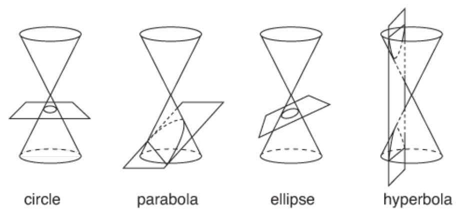

Figure 2: Conic Sections [12]

2.1.2 Orbital Elements

There exist different sets of elements that can be used to describe a spacecraft's motion in space. For instance, a spacecraft's orbit and position could be completely defined with ECI position and velocity vector components (r_x, r_y, r_z, v_x, v_y, v_z) or the perifocal frame (orbit frame) components ($r_p, r_q, r_w, v_p, v_q, v_w$). Although these systems are simple to work with calculations (during orbit transfers and manoeuvres), as well as for representation purposes, they do not provide much physical sense regarding the orbit characteristics.

However, there exists another set of orbital elements, called Keplerian, classical or conventional elements that can be used to describe a spacecraft's orbit in space, and the location of the spacecraft inside this orbit. Five of the six orbital elements are constant and describe the orbit shape and size, whereas the sixth parameter establishes the position of the spacecraft. The Keplerian element set that will be used in this project is $\{a, e, i, \Omega, \omega, \theta^*\}$. A short description of each of the elements can be found in Table 2. Figure 3 shows the orbital elements inside an orbit.

There are other important concepts that need to be discussed regarding the orbital elements. Firstly, the line of nodes is the intersection of the orbit plane with the fundamental plane. In addition, there are other parameters that can be obtained from combinations of the previous parameters and facilitate the calculations or have physical meaning. They are collected in Table 3.

Table 2: Keplerian Orbital Elements

Keplerian Elements	**Symbol**	**Definition**
Semi-major axis	a	Describes the orbit's size as it represents half the length of the conic.
Eccentricity	e	Describes the shape of the orbit, and the kind of conic that is being represented.
Inclination	i	Represents the angle between the equatorial and the orbital planes.
RAAN	Ω	Represents the angle from the vernal equinox (ECI X-axis) to the ascending node.
Argument of Perigee	ω	Represents the angle from the ascending node to the orbit's closest point to Earth (perigee), always measured in the direction of spacecraft's motion.
True Anomaly	θ^*	Angle that indicates the position of the spacecraft in the orbit measured from the perigee and in the direction of motion.

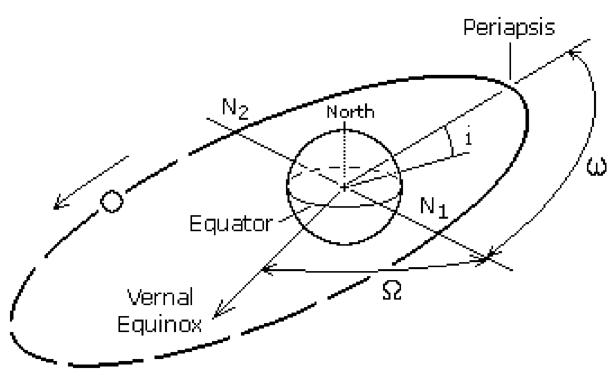

Figure 3: Orbital Elements [13]

Table 3: Other Orbital Elements

Orbital Elements	**Symbol**	**Description**	**Formula**
Semi-latus Rectum	p	Distance from the Earth to the spacecraft when $\theta^* = 90^{\circ}$.	$p = a(1-e^2)$
Energy	ε	Total energy of the orbit, calculated as kinetic plus potential. As we saw, it is conserved so it is constant at every point.	$\varepsilon = \frac{-\mu}{2a}$
Specific Angular Momentum	h	Modulus of the cross product between the position and velocity vectors. It is constant through the orbit.	$h = \sqrt{\mu p}$
Period	$\mathbb{P}$	Time that a spacecraft takes to travel a full orbit.	$\mathbb{P} = 2\pi\sqrt{\frac{a^3}{\mu}}$
Radius of the Perigee	r_p	Distance to the orbit's closest point to Earth.	$r_p = a(1-e)$
Radius of the Apogee	r_a	Distance to the orbit's furthest point to Earth.	$r_a = a(1+e)$
Velocity	v	Spacecraft velocity modulus.	$v = \sqrt{\mu(\frac{2}{r} - \frac{1}{a})}$
Flight Path Angle	γ	Angle between the velocity vector and the tangent line to the orbit in the spacecraft's current position.	$\gamma = \arccos\frac{h}{rv}$

It is simple to express the position and velocity vectors in the perifocal frame by using the relations shown in Equation 5. Thus, it is important to know how to change from perifocal (pqw) to ECI (xyz) reference system and viceversa. The most common way is by using the following 3-1-3 rotation matrix represented in Equation 6.

$$\overline{r}_{pqw} = \begin{bmatrix} r\cos\theta^* \\ r\sin\theta^* \\ 0 \end{bmatrix} \quad \overline{v}_{pqw} = \begin{bmatrix} -\sqrt{\frac{\mu}{p}}\sin\theta^* \\ \sqrt{\frac{\mu}{p}}(e+\cos\theta^*) \\ 0 \end{bmatrix} \tag{5}$$

$$\overline{r}_{xyz} = R(\Omega, i, \omega) * \overline{r}_{pqw} \rightarrow R = \begin{bmatrix} c_\Omega c_\omega - s_\Omega c_i s_\omega & -c_\Omega s_\omega - s_\Omega c_i c_\omega & s_\Omega s_i \\ s_\Omega c_\omega + c_\Omega c_i s_\omega & -s_\Omega s_\omega + c_\Omega c_i c_\omega & -c_\Omega s_i \\ s_i s_\omega & s_i c_\omega & c_i \end{bmatrix} \tag{6}$$

2.1.3 Orbital Manoeuvres

In this section, the manoeuvring problem will be addressed. In general, there are two principal orbital manoeuvres models according to the thrusting type: continuous and impulsive thrusting. Continuous thrusting is a more accurate model, but it also adds complexity to the calculations as the variation of the spacecraft's position during the transfer is also considered. On the other hand, impulsive transfers assume that the manoeuvre duration is zero. Hence, the problem becomes much more simple, the position of the spacecraft is kept constant and its velocity varies, changing the orbit characteristics, as represented by Equation 7.

$$\overline{r}(t_0^+) = \overline{r}(t_0^-), \quad \overline{v}(t_0^+) = \overline{v}(t_0^-) + \overline{\Delta v} \tag{7}$$

It is interesting to notice that in order to perform an orbit transfer between two orbits that do not intersect, a minimum of two impulses will be required, one to manoeuvre from the initial to the transfer orbit and the second one, to adapt to the final orbit once the desired position is reached.

Moreover, the initial and final orbits are expressed in Keplerian orbital elements but the manoeuvre is normally expressed in the ECI reference frame. Hence it is important to know how to convert from ECI to Keplerian using the formulas described in the previous section. The exact procedure can be found in the literature [14].

2.1.4 Orbit Transfer Optimisation

The orbit transfer optimisation problem can be stated as the determination of the trajectory of a spacecraft that satisfies an initial and final conditions while minimising some quantities [15]. The most relevant quantities when analysing transfer orbits are the required propellant (represented by the total Δv in impulsive manoeuvres) and the TOF.

The orbit transfer optimisation problem does not have general analytic solutions. In fact, there only exist optimal known solutions for very specific cases, whereas for the rest, optimisation algorithms need to be employed. One of the most common cases with known optimal solutions is the case with two co-planar circular orbits. The Hohmann transfer is the most efficient solution (in terms of Δv) in case the radius ratio between both orbits (r_2/r_1) is less than 11.94 [8]. It consists of an elliptical orbit tangent to both circular orbits at its apses (perigee and apogee), as illustrated by Figure 4. The total impulse velocity requirement can be obtained by using Equation 8, while the total time of flight calculation is shown in Equation 9 [11].

$$\Delta v_T = \sqrt{\frac{\mu}{r_1}}\left(\sqrt{\frac{1}{r_2/r_1}} - \sqrt{\frac{2}{(r_2/r_1)[1+(r_2/r_1)]}} + \sqrt{\frac{2(r_2/r_1)}{1+(r_2/r_1)}} - 1\right) \tag{8}$$

$$TOF = \frac{\mathbb{P}_t}{2} = \pi\sqrt{\frac{a_t^3}{\mu}} = \pi\sqrt{\frac{(r_1+r_2)^3}{8\mu}} \tag{9}$$

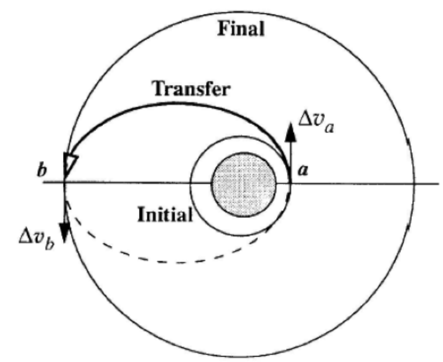

Figure 4: Hohmann Transfer Orbit [16]

However, if the ratio between the orbits radius is greater than 11.94, the bi-elliptical transfer becomes more propellant-efficient than the Hohmann transfer [17], especially when plane changes are required. This transfer orbit requires three tangential impulses. The first burn boosts the spacecraft into an elliptical orbit at a distance r away from the body. Generally, the further the spacecraft travels, the lower the total propellant. The second impulse, performed at apogee, sends the spacecraft into a second ellipse, with perigee at a distance r_2. The final impulse injects the spacecraft into the desired orbit. This is shown in Figure 5.

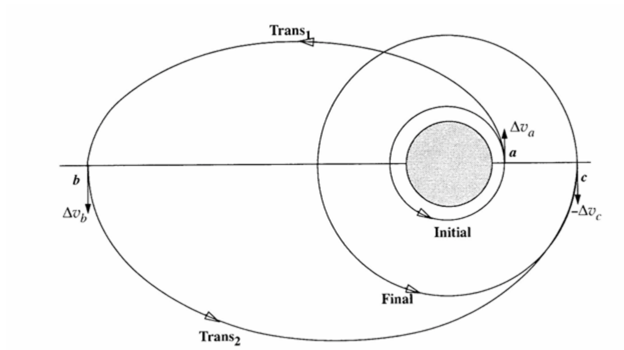

Figure 5: Bi-elliptic Transfer Orbit [18]

However, this decrease in total Δv entails an increase in the total TOF too, so a compromise must been achieved and decide if it is wise to trade the savings obtained for the time lost. In the limit, when $r \to \infty$, the transfer becomes bi-parabolic and the propellant waste is minimum, although time would be infinity. In the planar case, bi-parabolic transfer can offer improvements of around 10% so they are not often used, although they are interesting in plane changes [19].

2.2 Optimisation Algorithms

In general, non-linear optimisation algorithms can be divided into two groups: Calculus-based and Global/Non-smooth methods. The first group is formed by fast and accurate algorithms and are suitable when the objective function is continuous, not several local minimums exist and the function gradient is either analytically given or numerically computed (smooth function). These algorithms can perform successfully even if a poor initial guess is provided. On the other hand, global search optimisation algorithms do not need any additional information about the objective function, hence, they are called zero-order methods. However, they are usually slower and less reliable than gradient-based methods and are only recommended if the objective function is strictly non-smooth. Another disadvantage is that the problem constraints need to be incorporated through pseudo-objective functions with penalties.

For this project, both gradient and non-gradient based algorithms have been selected. A Sequential Quadratic Programming method will represent the former group whereas a Genetic Algorithm will represent the latter.

2.2.1 Sequential Quadratic Programming

The Sequential Quadratic Programming (SQP) method was chosen because it is "arguably, one of the best algorithms for constrained, non-linear optimisation" [20]. This is because, contrary to other direct methods, it solves for the search direction from a sub-problem with quadratic objective and linear constraints. In general, it makes the problem well posed and easy to solve, if the function is sufficiently smooth.

In addition, the SQP algorithm is available in MATLAB's optimisation toolbox inside the function *fmincon*.

2.2.2 Genetic Algorithm

The Genetic Algorithm (GA) is a global search method based on the biological evolutionary laws. Technically it is not a calculus-based "optimiser" but rather a probabilistic-based "searcher" that looks for the best variable combination. It is a computational model of evolution that mimics natural selection and reproduction, forcing the "fittest" to survive and reproduce, generating better individuals each generation.

The globally optimal solutions are always searched within a predefined search space. This space is obtained by discretising the continuous variables into a population of points according to a number of bits. Therefore it will depend on the variable limits but also on the resolution that we want our solution to have.

After the solution space is defined, variables are coded using binary strings. The string of n bits representing a variable is called a *gene*, the concatenation of all the variable genes forms a *chromosome*, and the addition of all the genes corresponds to an *individual*. Genetic algorithms use a *population* of individuals that changes for each generation, evolving to become fitter and more optimal.

The initial population is usually randomly generated and can consist of as many individuals as desired. Increasing the number of individuals will help improve the method results, but will also increase the computational cost. As a compromise, four times the total number of bits is recommended by some authors [21]. Regarding the stopping criteria, several approaches exist. Although a maximum number of generations is always enforced as a safety measure, the user may choose to stop the process when the best solution has not changed for several iterations, or when the chromosome are nearly homogeneous, meaning that the population is already concentrated in a small portion of the solution space. In this case, the last approach has been chosen, as recommended in the literature [22], which will be represented by means of the Bit String Affinity (BSA) value.

After the main concepts of the method and the stopping criteria have been defined, we need to comment on how each generation evolves from the previous one. In order to do so, three operations must take place: selection, crossover and mutation.

Selection: This operator basically represents Darwin's law of "survival of the fittest". The individuals of the population are paired between them and a tournament selection is performed [23]. Better (lower objective function result) individuals survive to be parents for next generation. On the other hand, the worse individuals are discarded and lost. Some other possible selection criteria include ranking or roulette wheel.

Crossover: Also called "breeding" or "mating" operator, it represents the reproduction function, where two parents will produce two children, so that the total population number remains constant. In this project the uniform crossover will be employed, although other versions exist such as the single-point crossover. According to this form of mating, each bit is chosen from either one of the parents with equal probability. In addition, the second children will receive the bit from the parent that was not chosen at first. Table 4 exemplifies the previous idea.

Table 4: Uniform Crossover Example

Parents	**Children**
10101010	10101110
00101111	00101011

Mutation: Finally, this operator corresponds to the mutating possibilities of the individuals during the crossover. It operates at the bit level, with a very low probability (usually between 0.002 and 0.1) [24] a bit that should be a 0 will become a 1, and viceversa. Mutation arises from the need to search in the neighbourhood of the current point while maintaining diversity in the population.

After these three operators have been applied to the population, a new generation of individuals is created and the process is repeated until any of the stopping criteria is met.

2.3 Multiobjective Optimisation

Multiobjective optimisation is a branch inside the optimisation discipline that addresses problems which require simultaneous optimisation of multiple objectives [25]. These objectives can be coupled or competing. The solutions can either be dominated, if a design solution is optimal for every objective function, or, usually, the solution is not a single variable combination but a set of designs. In this case a Pareto-optimal solution [26] would be obtained in which no improvement in one objective can be made without degrading at least another objective. The portion of space represented by all non-dominated design points in the solution is called Pareto frontier, and it illustrates the available trade-offs between objectives. Figure 6 shows the Pareto frontier (blue) in a $f_1 - f_2$ plot.

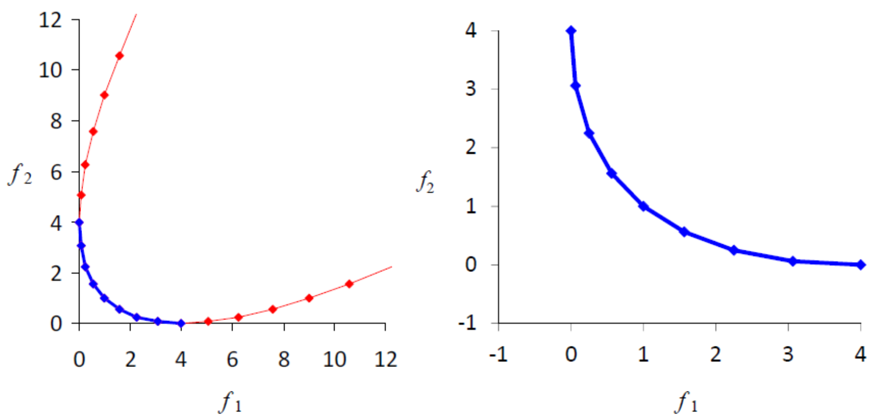

Figure 6: Pareto Frontier Example and Optimal Region Detail

Several approaches exist to address multiobjective problems. From simpler methods that convert them into single-objective problems, such as the Weighted Sum Approach, to more complex ideas as the ε-constraint, the min-max, or the goal attainment approaches.

In this project, the ε-constraint approach will be used, which comes Game Theory and is also called "gaming" approach. It basically consists of treating multiple objectives through inequality constraints. In this sense, we need to identify a primary objective and place limits (ε_i) on the remaining objectives, ensuring that the constraints are satisfied. By changing the objective limits, the Pareto frontier can be built.

3 Methodology

3.1 Optimal Orbit Transfer Problem

The optimal orbit transfer problem can be stated as follows. Given an initial orbit described by the Keplerian elements $a_0, e_0, i_0, \Omega_0, \omega_0$ and a final orbit described by $a_f, e_f, i_f, \Omega_f, \omega_f$, find the region of the optimal solutions that minimise total Δv and time of flight. Moreover, the minimum distance to the Earth's surface at any time must be greater than 200km to ensure the mission safety. Moreover, as we are using the ε-constraint approach, Δv will be treated as the main objective, whereas time of flight will be the constrained one.

However, there exist multiple ways to define the variables that our objective function will modify to find the optimal transfer. This project tries to reduce the dimension of the optimisation problem by reducing the number of variables needed, which would improve considerably the algorithm convergence time, reducing the computational cost. In order to do so, the transfer problem will be solved using Lambert arcs [8, 27]. This approach allows to find a transfer arc with the initial and final coordinate points and the time of flight along the transfer arc by solving an iterative method. Although a iterative method that solved the Lambert problem had been programmed, a faster, more reliable one was found in the literature [28] so it was decided to use the more advance version. One of the best advantages of using the Lambert arcs approach, is that it is guaranteed that the spacecraft will leave from and arrive to the desired orbits, thus eliminating the error possibility.

In addition, the problem needs to be accommodated to solve the transfer applying two, three or four impulses. No more impulses are considered as it is thought that four is the maximum number of impulses needed for optimal trajectories [2, 3, 4]. In the two-impulse case, we do not need any more variables, as there will only be one transfer arc between the two orbits. Nevertheless, we can reduce the number of variables as the initial and final coordinates are restricted: they must belong to the initial and final orbits, respectively. Hence, the problem was set in terms of the true anomalies (θ_0^* and θ_f^*) reducing the problem in 4 dimensions.

Regarding the multi-impulse transfer problem, some additional variables are needed to determine where the new impulses will be performed. As we want to continue with the Lambert problem approach to avoid the error constraints, and in this case, the intermediate impulse locations are not restricted, we need four more variables for each new impulse: three that define the impulse location and one for the transfer time to the next impulse point. Hence, for the multi-impulse problem the impulse locations and the transfer time between them are defined as variables and the function computes and builds Lambert arcs to connect them. A more detailed diagram can be seen in Figure 7.

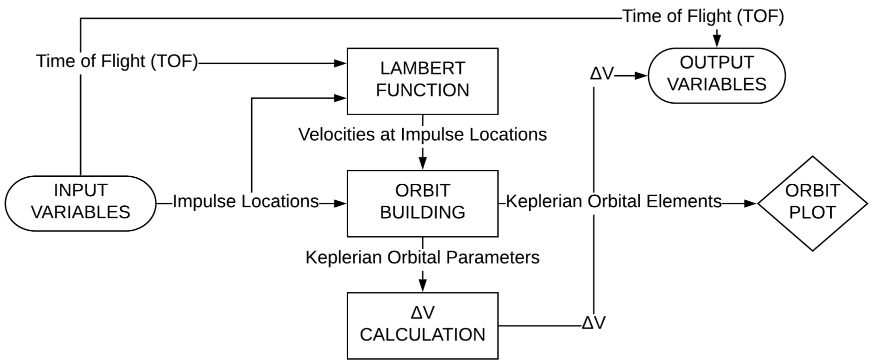

Figure 7: Objective Function Diagram

3.2 Optimisation Algorithms Settings

In this section, we will comment the particularities of each of the optimisation methods. Having two very different kinds of algorithms (a calculus-based and a global search type) means that each method requires the information to be treated in some specific way. Some of the aspects that will be discussed are the choice of initial solutions, the variable limits, how the algorithms handle the constraints, or the solution accuracy and resolution.

3.2.1 SQP Method

As it has been commented in previous sections, the SQP algorithm is a very fast gradient-based method. However, given the nature of the objective function, with nested orbital functions, it is impossible to obtain an analytic expression for the gradient. Hence, the algorithm will use numerical gradients. Moreover, some tolerances and stop-criteria have been specified, concretely a step tolerance of 10^{-2} and a maximum number of function evaluations of 300, as a safety measure.

With respect to the input variables, although the SQP algorithm allows the variables to remain unbounded (by setting the lower and upper limits to $-\infty$ and ∞ respectively) we opted for limiting the values as much as possible to facilitate convergence. Therefore, the true anomaly values are bounded between 0^o and 360^o, the time of flight variables must be positive and are limited by the ε-constraint approach, so the limit changes in successive evaluations. Lastly, the impulse location coordinates remain unbounded. All of these bounds are also useful to determine the initial solution. In case the optimal transfer is known (Hohmann case) a close initial solution can be easily provided. Nevertheless, this is hardly the case, so generally several random initial solutions within the variable bounds are provided. Several initial solutions are given to ensure that the algorithm is not stuck on a local minimum. Another possibility is the combination of both optimisation algorithms: using genetic algorithm to obtain a good initial guess for the SQP method. This concept will be further exploited in following sections.

On the other hand, output variables (Δv and TOF) have an accuracy equal to the inputted tolerance (10^{-2}). Although this tolerance might appear to be low for an optimisation algorithm, there is no practical meaning in increasing it, as the simplifications made surely would affect the solution accuracy to a greater extent. Plus, an error of 10^{-2} km or seconds in orbital mechanics is acceptable.

Finally, MATLAB's *fmincon* function is able to work with no-linear equality and inequality constraints but needs to read them from a separate function to the objective function. With this additional function, the program ensures that the mission never gets closer than 200km to Earth's surface (inequality) and checks that the initial and final points correspond to the desired orbits (equality).

3.2.2 Genetic Algorithm

The genetic algorithm is not a calculus-based method but a searcher: it tries different variable combinations until the stopping criterion decides that the optimum has been found. This stopping criteria, as commented before, is the BSA which compares the individual's chromosomes and stops the search when they are considerably similar (over 90%).

There are two main disadvantages to this method. First the solution space must be bounded and its accuracy comes determined by the number of bits allocated to each variable. In this sense, increasing the solution space or the number of bits will increase the algorithm accuracy but will also increase the computational cost. As the variables are coded in binary, the resolution of variable x_i can be obtained using the formula expressed in Equation 10.

$$R_i = \frac{x_i^U - x_i^L}{2^{b_i} - 1} \tag{10}$$

where x_i^U = Upper bound
x_i^L = Lower bound
b_i = Number of bits

Table 5 shows the limits, number of bits, and resolution of each of the variable types: true anomalies, time of flight, and impulse coordinates.

Table 5: Genetic Algorithm Variable Accuracy

Variable	**Resolution**	**Bounds**	**Bits**[1]
True Anomaly	0.7045°	$0^o \leq \theta^* \leq 360^o$	9
Time of Flight	1min	0h $\leq TOF \leq \varepsilon$-constraint h	10
Impulse Location	100km	$-r \leq x, y, z \leq r$	14

[1]For a time limit of 15h. It will depend on the ε-constraint value.

The r bound distance for the impulse location variables depends on the TOF ε-constraint limit. In order to minimise the solution space limitation, an upper-bound distance is estimated. To do so, we considered the furthest the spacecraft can go to ensure it travels back the same distance in the limited time of flight would be if it travelled an orbit whose period was the time of flight. Hence the distance away from the Earth would be equal to twice the semi-major axis of such orbit (considering it to be very eccentric). A 10% extra was added to avoid being too restrictive. Equation 11 shows the actual expression derivation.

$$TOF_{max} = \varepsilon = \mathbb{P} = 2\pi\sqrt{\frac{a^3}{\mu}} \rightarrow r = 2(1+10\%)a = 2.2(4\pi^2\varepsilon^2\mu)^{1/3} \tag{11}$$

Following the literature recommendations [21], the population size is four times the total number of bits, whereas the mutation probability is obtained from Equation 12. The initial population is obtained randomly.

$$P_m = \frac{N_{bits}+1}{2 * Pop_{size} * N_{bits}} = \frac{N_{bits}+1}{8N_{bits}^2} \tag{12}$$

Another great disadvantage of the genetic algorithm and, in general, all the global search methods, is that they cannot handle constraints. Therefore, a penalty needs to be added to the objective function. If any of the constraints is broken, a quantity proportional to the constraint violation is added to the Δv result, artificially worsening the result and forcing the algorithm to search for other solutions. However, we must ensure that the penalty is of the same order as the objective function result, in order to guarantee a meaningful contribution that will make the algorithm change the search direction. Hence, there are some steps that were followed to achieve it. First, as shown in Equation 13, all constraints are normalised to the unit. To obtain the same order as the objective function, we use the $\lceil \log_{10} \rceil$ properties, where $\lceil . \rceil$ represent the *ceil* operator, that give the number of non-decimal figures of a given number. Equation 14 shows how to compute the factor that needs to be multiplied to the normalised constraint to achieve our purpose. Plus, the objective function has an absolute value operator and the constraint will only activate the penalty function when violated (positive). Therefore, the log domain will not present any trouble.

$$g_i(x) \leq c_i \rightarrow G_i = \frac{g_i(x)}{c_i} - 1 \leq 0 \tag{13}$$

$$Penalty_i = 10^{\lceil \log_{10}|f| \rceil + 1} * G_i \tag{14}$$

Finally, due to the random nature of this method, it is good practice to run the algorithm multiple times to increase the possibilities that the optimal point has been found. In this project, 25 consecutive runs of the genetic algorithm will be performed. Although this will increase the computation time, it is necessary to properly discuss the validity of the results obtained.

3.3 Optimisation Algorithms Comparison

In order to compare the behaviour of both optimisation algorithms, they will be presented the same problem, one whose solution can be analytically obtained. This way it will be possible to compare the accuracy of the algorithms' results, as well as the computational cost of each of them.

It was decided to start with a simple Hohmann transfer between two concentric co-planar circular orbits. This was done because the Hohmann solution is one of the most characteristic results in orbital transfers and it is easy to obtain the analytic solution. Table 6 shows the initial and final orbit parameters.

Table 6: Hohmann Transfer Problem Data

Initial Orbit		**Final Orbit**	
Parameter	**Value**	**Parameter**	**Value**
a_0	15000 km	a_f	35000 km
e_0	0	e_f	0
i_0	0^o	i_f	0^o
Ω_0	0^o	Ω_f	0^o
ω_0	0^o	ω_f	0^o

The transfer orbit parameters can be easily obtained and are shown in Table 7 [8, 9]. Figure 8 shows the actual transfer. The total Δv and time of flight can be obtained from Equations 8 and 9 respectively.

Table 7: Hohmann Orbit Transfer Solution

Parameter	**Value**
a_t	25000 km
e_t	0.4
i_t	0^o
Ω_t	0^o
ω_t	0^o
$\theta^*_{tf} - \theta^*_{t0}$	180^o
Δv	1.7051 km/s
TOF	5.4637 h

As this problem is symmetric, there is a degree of freedom regarding the initial and final true anomalies. Hohmann transfer only requires that the difference between both angles equals 180^o, but does not determine the initial and final values precisely. Hence, to reduce the problem's dimension, the initial true anomaly θ^*_0 has been fixed to 0^o.

The transfer orbital parameters obtained with each algorithm will be compared to the solution quantities above in order to assess the quality of the results. Moreover, other aspects will be studied, such as the computational time.

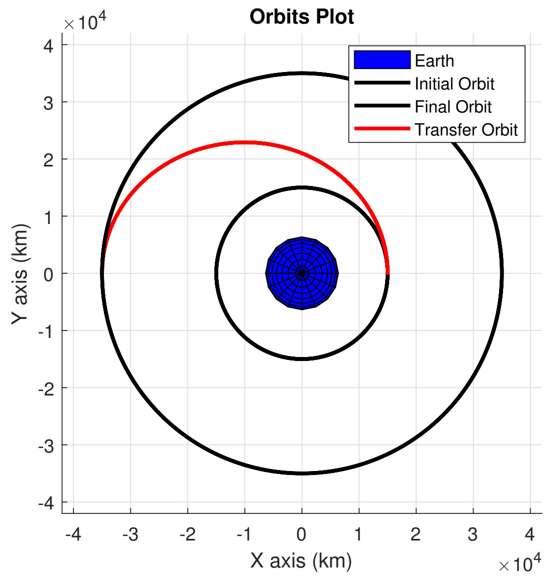

Figure 8: Hohmann Transfer Plot

3.3.1 Results

The results obtained by both algorithms can be seen on Table 8. It can be seen that the genetic algorithm offers very precise results whereas the SQP method presents some errors in the orbital parameters that lead to larger, unacceptable errors in the objective values results. In fact, the results obtained by the SQP method are even worse than expected, as not only does it worsen the GA solution, but also is completely unable to find a solution close to the optimal point. Given a random initial input, errors of the order of 463% and -75% were obtained. On the contrary, the GA precision could be further improved by increasing the number of bits that define each variable.

Table 8: Hohmann Orbit Transfer Results

	SQP Method[2]		**Genetic Algorithm**	
	Value	**Error(%)**	**Value**	**Error(%)**
a_t	25007 km	0.028%	25000.083 km	0.0003%
e_t	0.4011	0.275%	0.4000024	0.0006%
i_t, Ω_t, ω_t	0º	0%	0º	0%
$\theta^*_{tf} - \theta^*_{t0}$	170.692º	5.17%	179.6477º	0.196%
Δv	1.8223 km/s	6.871%	1.7052 km/s	0.006%
TOF	5.1834 h	5.13%	5.4477 h	0.293%

Several possibilities were checked to improve the SQP algorithm behaviour, including step and optimality tolerances, unconstrained optimisation, maximum function evaluations, initial guess, and even changing the gradient-based algorithm used. However the results were always very poor, confirming that the objective function is not suitable for a gradient-based method, as several authors had already pointed out [15, 29, 30]. Figure 9 shows a visual plot of the objective function solution distribution. As it can be seen, the local minimums and discontinuities are common making it impossible for gradient-based methods to reach a solution.

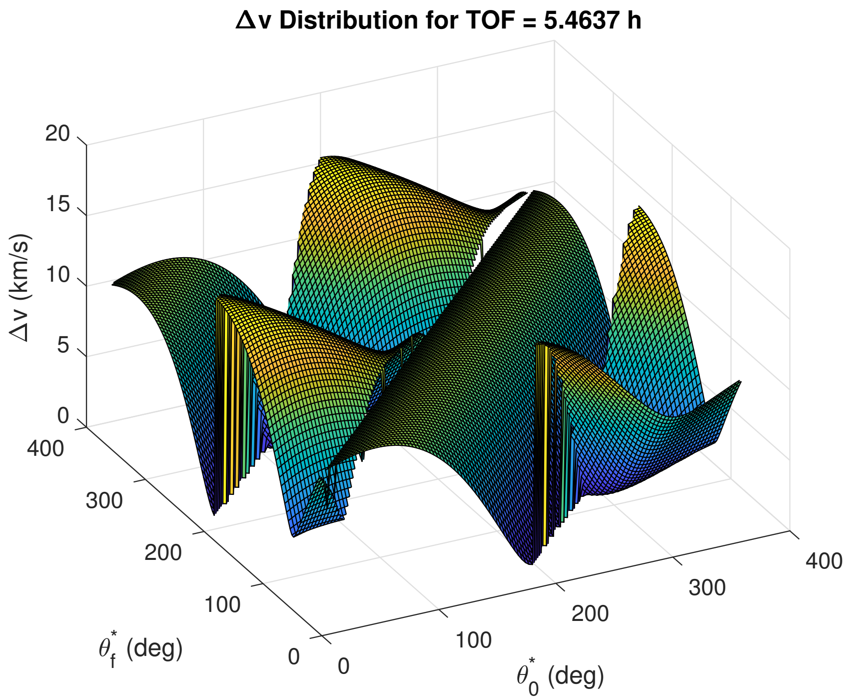

Figure 9: Objective Function Plot

After the results obtained in this section, it was decided that the only method that would be used to develop our orbital optimiser would be the genetic algorithm. Besides having no other alternative, as gradient-based methods have been proven useless against this kind of functions, GA has shown very good results, with errors below 1% in every case, providing feasible solutions, and being able to perform the calculations in less than one minute (average computing time 54 seconds).

[2] Case where initial solution is given from Genetic Algorithm output.

Given the need to compare transfer orbits with different number of impulses, three optimising functions were developed: two-impulse, three-impulse, and four-impulse optimiser. Table 9 shows the characteristics of each function.

Table 9: Optimising Function Characteristics

Optimiser	**Dimension**	**Bits***	**Iterations**
Two-Impulse	3	28	25
Three-Impulse	7	80	15
Four-Impulse	11	132	5

4 Results

Although we have seen that some orbit transfers have analytically optimal solutions, the majority of space missions require specific orbits to develop its purposes. The optimal transfers required to reach the mission orbit cannot generally be analytically obtained. This project, given the assumptions in which it is based, will focus on Earth related missions, whose aim is generally Earth observation, meteorological, scientific or for navigation and telecommunication purposes. The most common Earth related orbits in which satellites operate are Low Earth Orbit (LEO), Medium Earth Orbit (MEO), Geostationary Earth Orbit (GEO), and the Molniya orbit. These four orbits are represented in Figure 10.

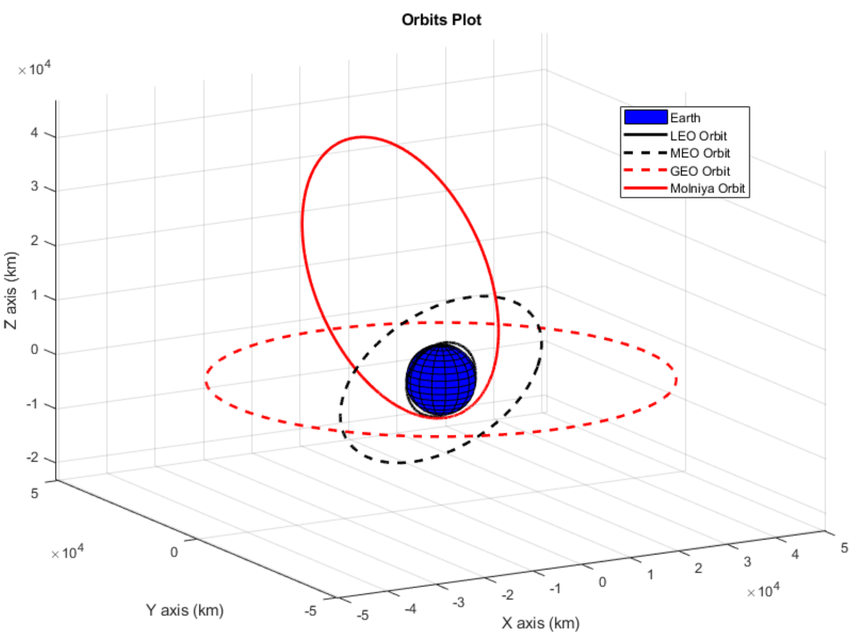

Figure 10: Earth Related Orbits Plot

In this project, a transfer from a Molniya to a GEO orbit will be optimised to show the capabilities of the optimiser program. These orbits have been chosen for several reasons: first, they are in different planes which will show the 3D capabilities of the algorithm, which is able to solve considerably more difficult transfers than a co-planar Hohmann; plus, being three-dimensional, allows for a more visual representation of the results. Finally, it is an orbit transfer with possible applications in satellite missions, as the Molniya and GEO orbits are generally used for communication purposes. The orbital parameters that define each of the orbits are shown in Table 10.

Table 10: Initial and Final Orbit Parameters

Initial Orbit (Molniya)		**Final Orbit (GEO)**	
Parameter	**Value**	**Parameter**	**Value**
a_0	26600 km	a_f	42164 km
e_0	0.74	e_f	0
i_0	63.4^o	i_f	0^o
Ω_0	0^o	Ω_f	0^o
ω_0	280^o	ω_f	0^o

One of the most important aspects of space missions is reliability, which can be defined as the quality of performing well. A mission will be more reliable the less complex it results, hence it can be more interesting to choose a two-impulse transfer solution with a slightly higher Δv consumption, than a four-impulse one. Therefore, our solution will visually differentiate between the number of impulses required.

Having said this, the main contribution of this optimiser is its multiobjective optimisation capabilities. This construction allows the space mission designer to have a quick, visual representation of the optimal region. Thus, the compromise between TOF and Δv can be easily chosen. This is why the main result from this optimiser can be considered to be the Pareto frontier showing the optimal region.

Once the Pareto frontier is shown, a point in the optimal region should be chosen, which is done by indicating the time of flight for the mission and the number of impulses desired. Then, the program will provide all the necessary data for the transfer completion. This information includes Δv vector components, impulse location coordinates and/or θ^*, and the transfer orbit simulation plot, together with its keplerian orbital elements.

In order to find the Pareto frontier plot, the TOF ε-constraint limits need to be discussed. To obtain a reference value from where to choose the limits, a TOF-unconstrained optimisation run was performed. This was achieved by establishing a very large TOF-limit so the overall minimum Δv corresponding time of flight for the two-impulse case was computed.

The unconstrained optimisation results gave 10.7h as the reference time of flight. Hence, it was decided to analyse the following time of flight values to obtain a representation of the Pareto frontier: 500h, 50h, 25h, 15h, 12h, 10h, 8h, 5h, 3h, 1h.

4.1 Pareto Frontier

The Pareto frontier obtained for the orbit transfer from a Molniya to a GEO orbit for different times of flight can be found in Figure 11. As it can be seen, the different zones of the plot correspond to different number of impulses.

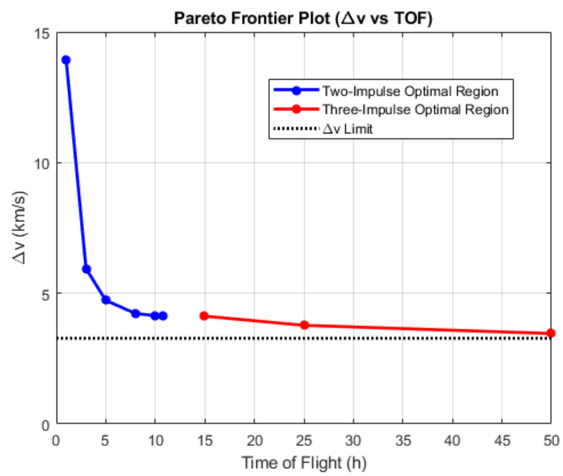

Figure 11: Pareto Frontier Results

It is interesting to see that there is a blank area between the two and three-impulse optimal regions. This is because around this area a non-optimal region appears: the two-impulse region would start increasing from the TOF = 10.7h point and the three-impulse does not improve that Δv value until close to the TOF = 15h point. The four-impulse case did not produce any of the optimal points. The numerical results obtained can be found on Table 11. It can be seen, for the TOF = 12h value, that the optimiser has chosen approximately the same solution than for the TOF = 10.7h case.

Table 11: Pareto Frontier Results

TOF$_{limit}$ (h)	**Δv (km/s)**	**TOF (h)**	**Impulses**
500	3.2769	496.1	3
50	3.4601	50.0	3
25	3.7707	25.0	3
15	4.1121	15.0	3
12	4.1375	10.7	2
10	4.1420	10.0	2
8	4.2255	8.0	2
5	4.7249	5.0	2
3	5.9451	3.0	2
1	13.9200	1.0	2

4.2 Transfer Orbits

After analysing the Pareto frontier plot, we need to do the mission design engineer job and choose the transfer orbit that best adapts to our conditions. It could be seen on Figure 11 that the TOF = 500h, represented by the dotted line, did not produce a meaningful improvement in the propellant consumption. Hence, we have chosen three different solutions to contemplate different cases regarding the mission priorities.

4.2.1 Time Of Flight = 10h

According to our engineering judgment, this option represents the best compromise between propellant and TOF. It requires a total Δv of 4.14 km/s, which is a high value due to the plane change requirements. It is also performed using only two impulses, which increases the mission reliability. The transfer data needed is shown in Tables 12 and 13. A plot of the transfer orbit can be seen on Figure 12.

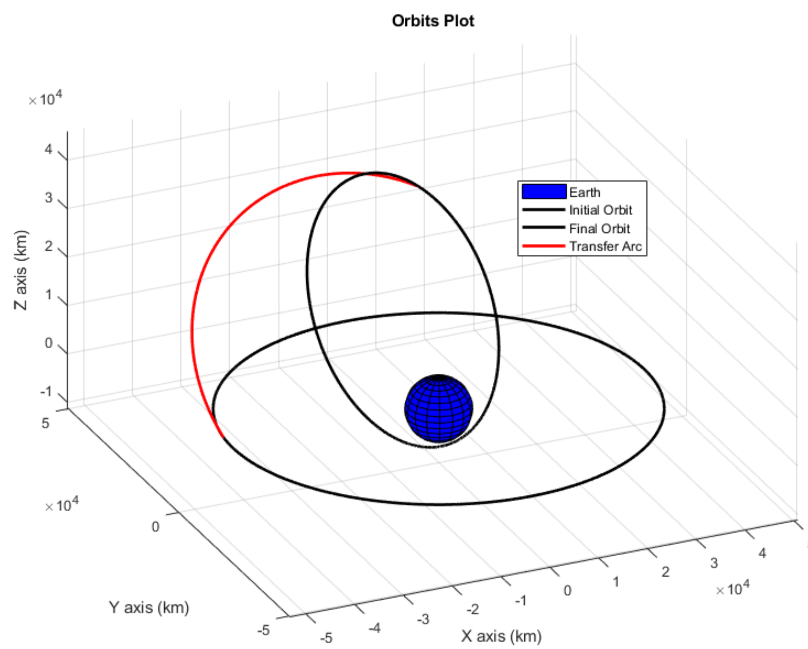

Figure 12: TOF = 10h Orbit Plot

Table 12: TOF = 10h Transfer Orbit Data

Impulse Location		**Impulse Coordinates**		**TOF**
1 (°)	**2 (°)**	**1 (km/s)**	**2 (km/s)**	**1 (h)**
164.1	173.3	-0.8857	-1.4199	10.0
		0.3094	-1.7637	
		0.0191	2.2664	

Table 13: TOF = 10h Transfer Orbit Parameters

Parameter	**Value**
a_t	36304.0 km
e_t	0.4520
i_t	62.95°
Ω_t	-6.693°
ω_t	314.16°
θ^*_d	133.0°
θ^*_a	225.8°

4.2.2 Time Of Flight = 50h

This case represents a situation in which time of flight is not relevant for the mission, but as much propellant as possible needs to be saved. It is also used to represent how a three-impulse orbit would look like if that output was chosen. It is interesting to see in Figure 13 how the transfer arc separates from Earth and performs the manoeuvre near the periapsis, where the velocities are smaller. Therefore, the Δv values are also reduced, as seen in Table 15. The transfer arc keplerian elements can be seen in Table 14.

Table 14: TOF = 50h Transfer Orbit Parameters

Transfer Arc 1		**Transfer Arc 2**	
Parameter	**Value**	**Parameter**	**Value**
a_{t1}	66075.9 km	a_{t2}	75555.9 km
e_{t1}	0.6900	e_{t2}	0.4420
i_{t1}	61.16°	i_{t2}	2.36°
Ω_{t1}	-8.975°	Ω_{t2}	-27.476°
ω_{t1}	-14.14°	ω_{t2}	-1.49°
$\theta^*_{a_{t1}}$	90.5°	$\theta^*_{a_{t2}}$	200.4°
$\theta^*_{d_{t1}}$	195.1°	$\theta^*_{d_{t2}}$	1.49°

Table 15: TOF = 50h Transfer Orbit Data

Impulse Location			Impulse Coordinates (km/s)			TOF (h)	
1 (°)	2 (km)	3 (°)	1	2	3	1	2
	-102618		-1.1764	-0.3633	-0.3095		
152.2	15433	332.5	0.7686	-0.9088	-0.5313	28.5	21.5
	-1389.7		0.4533	0.9299	-0.1523		

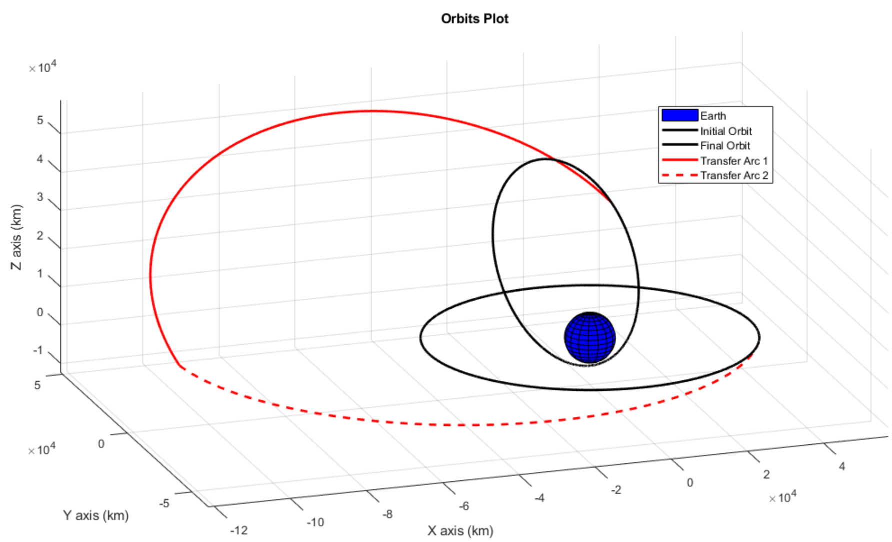

Figure 13: TOF = 50h Orbit Plot

4.2.3 Time Of Flight = 1h

Finally, this situation represents the complete opposite case, in which propellant is not an issue and we need to reach the desired orbit as fast as possible. The Δv values increase considerably as seen in Table 16 and it is interesting to see in Table 17 that the transfer orbit would be hyperbolic, although this is not easily seen in Figure 14.

Table 16: TOF = 1h Transfer Orbit Data

Impulse Location		Impulse Coordinates		TOF
1 (°)	2 (°)	1 (km/s)	2 (km/s)	1 (h)
		-6.7707	4.8682	
219.1	174.0	0.7052	-1.8840	1.0
		-1.0882	4.7032	

Table 17: TOF = 1h Transfer Orbit Parameters

Parameter	**Value**
a_t	-12647.3 km
e_t	3.135
i_t	70.03°
Ω_t	-5.99°
ω_t	121.73°
θ^*_d	19.75°
θ^*_a	58.27°

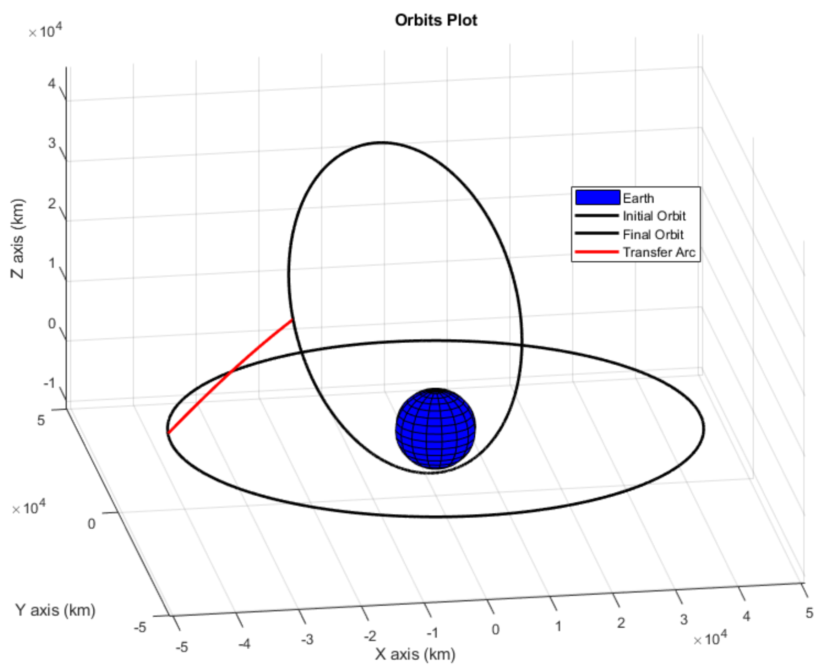

Figure 14: TOF = 1h Orbit Plot

5 Budget & Project Specification

5.1 Budget

This section comprises the estimated economic cost of the presented project. The budget includes optimiser programming, report writing and presentation building. Moreover, the computer equipment and program licenses cost have been added. Table 18 summarises the project overall cost.

Table 18: Project Budget

	Hours	**Hour cost (€/h)**	**Total Cost (€)**
First-Year Engineer	225	25	5625
Laptop Computer	225	2	450
Matlab License	150	0.5	75
Latex License	50	0.5	25
Microsoft Powerpoint	25	0.5	12.5
Project Printing			25
		TOTAL	**6187.5**
		TOTAL + TAX	**7486.88**

The project total cost, including a 21% tax, amounts to:

 # Seven thousand, four hundred and eighty-six euros with eighty-eight cents #

5.2 Project Specifications

In the aim of maximising worker productivity while minimising the risks for its health, the conditions in which the work is performed must be taken into account.

In this project, most of the work was performed using a computer. In order to respect the minimum safety and health provisions regarding work with this equipment the engineer must ensure:

- To maintain a proper body posture and work in a comfortable and ergonomic chair, to prevent from future muscular injuries.
- To work in a properly lit position reducing the damage and fatigue that the screen causes in the worker's vision.
- The connections to the electric grid must have the necessary security measures to avoid accidents that could damage computer equipment or cause injury to the worker.
- To control the workload, regulating in this way the hours dedicated and the amount of breaks in the workday.

Apart from these measures, the project engineer is able to work from any place desired, including, but not limited to, offices, libraries, or its own house. This place should allow the engineer to develop the highest amount of concentration and limit the distractions.

Regarding the technical aspects of the project, the hardware and software elements used for its development are presented.

With respect to the hardware, an ASUS X555L laptop computer was used with an Intel(R) Core^{TM} i7-5500U CPU (2.40GHz) processor, an 8GB RAM, a NVidia GeForce 920M graphic card, and a 64-bit Windows 10 operating system.

As far as the software is concerned, the following programs were used:

- **Matlab vR2018a.** For algorithm programming, computational tasks and orbits plotting.
- **Overleaf v2019.** For the project report writing and formatting.
- **Microsoft Powerpoint v2016.** For the project presentation building.

6 Conclusions & Future Work

6.1 Literature Comparison

We would like to compare the capabilities and the results of our orbit optimiser with other previous projects. Since the derivation of the Hohmann transfer in 1925 [31] and the prove of its optimality in 1963 [32] were developed, significant contributions have been made in this area. These include gradient derivation [33], analytical solutions [3], direct and indirect optimisation [34, 29, 30]. These were found to provide deficient solutions given the complexity and nonlinearities present in the optimal orbit transfer problem [15]. There has also been some work in evolutionary algorithms such as ours [35, 36, 37].

Regarding genetic algorithms, we have selected three relevant works that can be compared with our proposed algorithm. Cacciatore & Toglia, (2008) [23] studied minimum fuel impulsive orbit transfers with a constraint in time of flight. They solve Lambert's problem as well, so the problem setting is very similar. Their main contribution includes the effects of genetic algorithm parameters to the solution, such as the selection operator criterion. They show that the tournament criterion outperforms every other configuration (for instance the inverse roulette wheel) if we include the computational time in our criteria. This backs up our configuration. Unfortunately, they do not offer any error measure so that we can compare our results to theirs.

Zhang, et al (2015) [38], one of the most recent papers, developed a "Two-impulse transfer between co-planar elliptic orbits" genetic algorithm optimiser. Our algorithm however, despite being more general (three-dimensional and multi-impulse) is able to obtain considerably lower errors with respect to the optimal solution. While they reach the optimal point with a 1-1.5% relative error in Δv , our algorithm is able to do so with a 0.006% error in Δv which is far more desirable.

Moreover, Abdelkhalik (2005) [39] also employed a Lambert problem formulation for his genetic algorithm optimiser and was able to find the Hohmann transfer with a 0.045% Δv and 1.515% TOF relative error, whereas we obtain a 0.006% error for Δv and 0.293% error for TOF.

Finally, Yilmaz (2012) [40] programmed a multi-impulse genetic algorithm optimiser, although the optimisation problem was six-dimensional for the two-impulse case, whereas our setting allows it to be reduced three dimensions, improving computational cost and reducing complexity. The error obtained in this case for the Hohmann Δv total magnitude is around 0.1%.

Overall, it can be seen that the proposed algorithm has a better accuracy than the previous methods. There exists other applications of genetic algorithms in the optimal orbit transfer problem but they involve rendezvous, which are not covered in this project. On the other hand, this project introduces multi-objective optimisation techniques to take time of flight into account and build a Pareto frontier.

6.2 Conclusions

After comparing our optimiser capabilities and results with other projects we will analyse the conclusions obtained. Not only is this optimiser more versatile than most of the comparable ones, that either are very problem-specific, or are limited to the co-planar or the two-impulse case; but also this program is considerably more accurate. It has proven to be able to obtain the optimal result with a relative error at least one order of magnitude lower than its competitors.

This precision is mainly due to a great resolution in the problem variables and to a great choice in the genetic algorithm characteristics that determine how the global search is performed. But also to the problem setting choice with respect to the orbital mechanics involved.

Moreover, the computational cost of the algorithm is completely acceptable for the purpose of this project: to obtain an optimal transfer that serves as a reference for developing more complex models. This cost is, at maximum 5 minutes per time of flight value, and it can be reduced to under a minute in the two-impulse case. This includes all the repetitive calculations needed to ensure the genetic algorithm convergence. This fast behaviour can be obtained, without a loss in the variables accuracies, thanks to a great understanding of the orbital mechanics foundations and the Lambert's problem, which allows reducing the problems dimension.

Additionally, this project has been conducted in a very complete way. The objectives were known from the beginning and we used our every bit of our knowledge to ensure that the best possible answer was provided. The comparison between various optimisation algorithms is an example. Although the SQP method was undeniably unfit for this application, it is a very powerful tool that should always be taken into consideration when solving an optimisation problem. Genetic algorithm, on the other hand, has proven to be a very reliable search method and an ideal one to use in this kind of problems.

Furthermore, this problem setting allow the program to always come up with a solution, even if not the optimal one, it will be a close one. In this sense, with sufficient variable resolution, we do not need to worry about our spacecraft not reaching the desired orbit, which increases the reliability of the results obtained.

In addition, the multiobjective optimisation aspect of the project presents an added advantage with respect to other projects as we believe that the Pareto frontier obtained can be very helpful in the orbit mission design, as it visually represents the compromises between the different options.

As far as the budget is concerned, we believe that we have been able to reduce cost to the maximum, demanding a reasonable amount for the work performed.

Overall, we consider this project to be completely successful in fulfilling every single objective that was considered, overcoming the expectations in several areas. However, there is always room for improvements, and in the following section, we will discuss some possible extensions that could be done to further develop this orbit optimiser.

6.3 Future Work

In order to continue improving this optimiser, there are some aspects that could be further analysed in future projects.

Firstly, to increase the functionality of the optimiser, the rendezvous problem should be studied and included inside the program possibilities. As it has been shown, the program does a great work finding the optimal transfer path, but if rendezvous is required, the synodic period between the orbits would certainly limit the time window to perform the manoeuvres, reducing the mission reliability. Rendezvous can be very easily handled with Lambert's problem, as it is easy to find the relation between the final position vector and the time of flight. This would relate two variables, reducing the optimisation problem dimension and therefore, lowering the computational cost.

Together with rendezvous, interplanetary flight would be a thrilling improvement to make. Besides changing the central body from Earth to the Sun (which would be trivial) the program could be adapted to take into account possible flybys where the spacecraft could use gravitational assists from other solar bodies. However, this could conflict with our two-body assumption so another model should be used, for instance, the patched conics approach.

Another interesting development of the project is to study the effects of the different genetic algorithm parameters (e.g. number of bits, BSA limit, maximum number of generations, population size, mutation probability, etc.) in the algorithm convergence time and accuracy. This work has been performed for the given problem but it would be interesting to study and document the effects more in-depth.

Finally, a Graphic User Interface (GUI) could be developed to ease the data input and the algorithm parameters change. It would also be more intuitive to see the results and select the desired orbits characteristics, hence obtaining all the necessary data impulse coordinates, intermediate TOFs, etc. to perform the mission.

References

[1] Delft University Aerospace Engineering. *Spacecraft engineering: Delta-v (velocity increment) budget.* URL: http://lr.tudelft.nl. (Retrieved: 05.09.2019).

[2] T. Edelbaum. "How many impulses?" In: *Astronautics and Aeronautics* 5 (1967), pp. 64–69.

[3] J. Doll F. Gobetz. "A survey of impulsive trajectories". In: *AIAA Journal* 7 (1969), pp. 801–834.

[4] J. Betts. "Survey of numerical methods for trajectory optimization". In: *Journal of Guidance and Control* 21.2 (1998), pp. 193–207.

[5] University of Colorado - Colorado Springs. *Orbit in Space Coordinate Frames and Time. MAE 5410-Astrodynamics Lecture-5.* URL: https://www.uccs.edu/. (Retrieved: 05.13.2019).

[6] Embry-Riddle Aeronautical University. *EE 495: Navigation Mathematics.* URL: https://erau.edu/. (Retrieved: 05.13.2019).

[7] Purdue University. *AAE 532: Orbit Mechanics. Integrals of the Motion.* URL: https://www.purdue.edu/. (Retrieved: 05.16.2019).

[8] Howard D. Curtis. *Orbital Mechanics for Engineering Students.* 3rd ed. Elsevier, 2014.

[9] Jerry E. White Roger R. Bate Donald D. Mueller. *Fundamentals of Astrodynamics.* 1st ed. Dover Publications, 1971.

[10] Jerry Jon Sellers. *Understanding Space.* 2nd ed. MCGraw Hill, 2004.

[11] Vladimir A. Chobotov. *Orbital Mechanics.* 3rd ed. AIAA, 2002.

[12] CliffsNotes. *The Four Conic Sections.* URL: https://www.cliffsnotes.com/study-guides/algebra/algebra-ii/conic-sections/the-four-conic-sections. (Retrieved: 06.06.2019).

[13] Rocket & Space Technology. *Orbital Mechanics, Orbital Elements.* URL: http://www.braeunig.us/space/orbmech.htm. (Retrieved: 06.08.2019).

[14] W. D. McClain D. A. Vallado. *Fundamentals of Astrodynamics and Applications.* 4th ed. Microcosm Press & Springer, 2007.

[15] B. Conway. *Spacecraft Trajectory Optimization.* 1st ed. Cambridge University Press, 2010.

[16] Github. *Why won't/didn't New Horizons stop at Pluto and instead had to fly by?* URL: http://rantonels.github.io/capq/q/OM2.html. (Retrieved: 06.11.2019).

[17] Robert Silber Rudolph F. Hoelker. "The Bi-Elliptical Transfer Between Co-Planar Circular Orbits". In: *Planetary and Space Science* 7 (1961), pp. 164–175.

[18] University of Colorado Boulder. *Basic Orbital Transfers.* URL: http://ffden-2.phys.uaf.edu/webproj/211_fall_2016/Jadyn_Cook/Jadyn_Cook/Page03.html. (Retrieved: 06.11.2019).

[19] H. L. Roth. "Bi-Elliptic Transfer with Plane Change". In: *Astrodynamics Department Guidance and Control Subdivision Electronics Division* (1965).

[20] Jon W. Tolle Paul T. Boggs. "Sequential Quadratic Programming". MA thesis. University of North Carolina, 1996.

[21] William Crossley. *Genetic Algorithm Introduction.* 2018. AAE 550: Purdue University.

[22] Daniel Raymer William Crossley Kamlesh Nankani. "Comparison of Bit-String Affinity and Consecutive Generation Stopping Criteria for Genetic Algorithms". In: *42nd AIAA Aerospace Sciences Meeting and Exhibit* (2004).

[23] C. Toglia F. Cacciatore. "Optimization Of Orbital Trajectories Using Genetic Algorithms". In: *Journal of Aerospace Engineering,Sciences and Applications* (2008), pp. 58–69.

[24] M. Finck R. N. Greenwell J. E. Angus. "Optimal Mutation Probability for Genetic Algorithms". In: *Mathematical. Computational Modelling* 21.8 (1995), pp. 1–11.

[25] André H. Deutz Michael T. M. Emmerich. "A tutorial on multiobjective optimization: fundamentals and evolutionary methods". In: *Natural Computing* (2018).

[26] Panos Pardalos Altannar Chinchuluun. "A survey of recent developments in multiobjective optimization". In: *Annals of Operations Research* 154.1 (2007), pp. 29–50.

[27] Dario Izzo. "Revisiting Lambert's problem". In: *Original Article* (2014).

[28] Rody Oldenhius. *MATLAB Robust solver for Lambert's orbital-boundary value problem.* URL: https://nl.mathworks.com/matlabcentral/fileexchange/26348. (Retrieved: 12.16.2018).

[29] B. Conway P. Enright. "Optimal Finite Thrust Spacecraft Trajectories Using Collocation And Nonlinear Programming". In: *Journal of Guidance, Control and Dynamics* 14.5 (1991), pp. 981–985.

[30] P. J. Enright. "Discrete Approximations To Optimal Trajectories Using Direct Transcription And Nonlinear Programming." In: *Journal of Guidance, Control and Dynamics* 15.4 (1992), pp. 994–1002.

[31] W. Hohmann. "Die Erreichbarkeit der Himmelskörper". In: *Original Article* (1925).

[32] R. Barrar. "An Analytical Proof that the Hohmann-Type Transfer is the True Minimum Two-Impulse Transfer". In: *Astronautical Acta* 9 (1963), pp. 1–11.

[33] M. Hadelsman P. Lion. “Primer Vector On Fixed-Time Impulsive Trajectories”. In: *AIAA* 6.1 (1968), pp. 127–132.

[34] S. W. Paris C. R. Hargraves. “Direct Trajectory Optimization Using Nonlinear Programming and Collocation.” In: *Journal of Guidance, Control, and Dynamics,* (1987), pp. 338–342.

[35] D. B. Spencer Y. H. Kim. “Optimal Spacecraft Rendezvous Using Genetic Algorithm.” In: *Journal of Spacecraft and Rockets* (2002), pp. 859–865.

[36] G. Colasurdo D. P. Santos A. F. Prado. “Four-Impulsive Rendezvous Maneuvers for Spacecrafts in Circular Orbits Using Genetic Algorithms”. In: *Mathematical Problems in Engineering* (2012).

[37] B. Conway M. Potani. “Particle swarm optimization applied to impulsive orbital transfers.” In: *Acta Astronautica* (2012), pp. 141–155.

[38] Dongbai Li Gang Zhang Guangfu Ma. “Two-impulse transfer between coplanar elliptic orbits using along-track thrust”. In: *Original Article* (2015).

[39] O. M. Abdelkhaik. “Orbit Design And Estimation For Surveillance Missions Using Genetic Algorithms.” PhD thesis. Texas A&M University, 2005.

[40] Ahmet Yilmaz. “Orbit Transfer Optimzation Using a Spacecraft with Impulsive Thrust Using Genetic Algorithm”. MA thesis. Middle East Technical University, 2012.