UNIVERSIDADE
BEIRA INTERIOR

Orbital transfer optimization for LEO satellites

João Miguel Caetano dos Santos

Dissertação para obtenção do Grau de Mestre em
Engenharia Aeronáutica
(Mestrado Integrado)

Orientador: Prof. Doutor Kouamana Bousson

Outubro de 2023.

Declaração de Integridade

Eu, João Santos, que abaixo assino, estudante com número de inscrição 39544 do Mestrado Integrado em Engenharia Aeronáutica da Faculdade de Engenharia, declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o **Código de Integridade da Universidade da Beira Interior**.

Mais concretamente, afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, e que em particular atendi à exigida referencia de frases, extratos, imagens, e outras formas de trabalho intelectual, e assim assumo na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 06/10/2023

Assinado por: **João Miguel Caetano dos Santos**
Num. de Identificação: 14739001
Data: 2023.10.06 11:56:51+01'00'

Acknowledgements

Firstly I would like to thank my orientator in the thesis, Professor Kouamana Bousson for all the help and direction given during the thesis. I would also like to thank all my friends and family members for all the support given during the whole course, especially during the thesis elaboration.

I would also like to thank LAETA-UBI/AeroG for the scholarship “Verão com ciência 2022” which helped me to elaborate this thesis.

Resumo estendido

As transferências orbitais são importantes porque nos permitem realizar a transferência entre duas órbitas a diferentes altitudes, inclinações ou excentricidades atravez de uma orbita eliptica intermédia, entre os seus usos está a re-orbitação de satélites ou até mesmo a interceptação de mísseis, nesta dissertação vamos estudar diferentes métodos de realizar estas transferências Orbitais.
Podemos dividir a transferência de órbita em duas categorias principais, os métodos clássicos e os métodos não clássicos. O método clássico também conhecido como transferência de órbita de Hohmann foi descrito em 1925 por Walter Hohmann, o problema deste método é que obriga a orbita de tranferência a começar no apogeu ou perigeu da orbital inicial e a acabar no perigeu ou apogeu da órbita final respectivamente, devido a este problema o método acaba por consumir mais energia e tempo do que é realmente necessário. Para contornar estes problemas, usamos a transferência orbital não-Hohmann que nos permite executar a mesma começando em qualquer ponto da órbita inicial e terminando em qualquer ponto da órbita final, minimizando uma função de custo. Para a resolução do problema da transferência orbital não-Hohmann iremos utilizar o método directo que consiste em dividir a trajectória em vários segmentos evitando que seja necessário derivar a função de custo permitindo que o problema seja resolvido com o método de programação não-linear (NLP).
O objetivo principal desta tese é encontrar o método para cada tipo de transferência orbital que minimiza a energia global usada sem ter limitações quanto ao seu uso como os métodos clássicos tais como a transferência de Hohmann têm. Na dissertação foram estudados três casos, a órbita de transferência coplanar de Hohmann, a órbita de transferência coplanar não-Hohmann e a órbita de transferência não-coplanar não-Hohmann. Para modelar essas transferências orbitais foi utilizado o GMAT (General Mission Analysis Tool) que é uma ferramenta da NASA e que permite ao utilizador criar uma transferência orbital entre duas órbitas escolhidas pelo usuário com a ajuda das ferramentas que disponibiliza para o processamento dos dados tais como a ferramenta obter (achieve), propagar (propagate) e manobra (maneuver) podendo retirar do programa dados importantes como o tempo necessário para realizar a transferência orbital, a energia necessária a cada impulso, as coordenadas e velocidades do satélite na órbita em cada momento utilizando a ferramenta reportar (report). Nos casos não-Hohmann as variáveis de estado usadas no caso coplanar foram o raio, a anomalia verdadeira, a velocidade radial e a velocidade transversal e as variáveis de controlo foram a aceleração e a direção de aceleração de controlo. As variáveis de estado usadas no caso não coplanar foram as coordenadas esféricas da posição e os componentes esféricos da velocidade, as variáveis de controlo foram a aceleração e os ângulos de direção da tração.
Os resultados obtidos foram satisfatórios apesar de em alguns gráficos os resultados apresentarem alguns valores fora do esperado no início e no fim da transferência orbital isso deve-se ao baixo número de iterações e não porque o método e modelo utilizado não

foram os indicados. Os resultados gerais obtidos mostraram uma órbita de transferência consistente com a órbita de transferência obtida no GMAT. Com base nos resultados podemos concluir que os métodos das equações de Clohessy–Wiltshire para a transferência orbital de Hohmann e método da utilização das equações de movimento de satélites para as transferências orbitais não-Hohmann são indicados para resolver este tipo de problemas de transferência de órbita.

Palavras-chave

Transferência orbital não coplanar, transferência de orbita terrestre baixa, transferência não-Hohmann LEO, transferência orbital ótima.

Abstract

Orbital transfers are important because they allow us to perform a transfer between two orbits at different altitudes, inclinations, or eccentricities using an intermediary elliptical orbit, orbit transfers can be used to re-orbit satellites or even intercept missiles, in this dissertation we studied different methods of performing these Orbital transfers.
We can divide the orbit transfer into two main categories, the classic methods and the non-classic methods. The classic method also known as the Hohmann orbit transfer was described in 1925 by Walter Hohmann, the problem with this method is that it has the restriction of the orbit transfer having to start in the apogee or the perigee of the initial orbit and ending at the perigee or the apogee of the final orbit respectively. In order to bypass these problems, we use the non-Hohmann orbit transfer which allows us to execute the orbit transfer starting at any point of the initial orbit and finishing at any point of the final orbit, minimizing the cost function. The main objective of the thesis is to find the method for each type of orbit transfer minimizes the global energy used without having the limitations of the normal methods such as the Hohmann transfer. In the dissertation we will study three cases, the coplanar Hohmann transfer orbit, the coplanar non-Hohmann transfer orbit, and the non-coplanar non-Hohmann transfer orbit. In order to model these orbit transfers we used GMAT (General Mission Analysis Tool) which is a tool of NASA that allows the user to create an orbit transfer between two orbits chosen by the user. In the non-Hohmann cases the state variables used in the coplanar cases were the radius, the true anomaly, the radial velocity and the transverse velocity, the control variables were the acceleration and the control acceleration direction. For the non-coplanar case the state variables were the spherical coordinates of position and the spherical components of velocity, the control variables were the acceleration and the thrust direction angles. The results obtained were satisfactory despite in some graphs the results having some bumps at the beginning and end of the transfer orbit that was because of the low number of iterations.

Keywords

Non-coplanar orbital transfer, LEO transfer, non-Hohmann LEO transfer, optimal orbit transfer

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

Nomenclature

h	Angular momentum	$km^2 s^{-1}$
v,V	Velocity	km s^{-1}
r	radius	km
τ	Transfer time	s
Ri	Radii	-
μ	Gravitational parameter	$398600 km^3 s^{-2}$
Θ	Plane change	rad
R	Radius	km
γ	Flight path angle	rad
F	Force	N
ρ	Atmospheric density	kg m^{-3}
S	Projected area	km^2
C	Coefficient	-
α	Angle of attack	rad
σ	Bank angle	rad
ω	Angular velocity	rad s^{-1}
x,y,z	Position coordinates	km
ΔV_P	Velocity impulse	km s^{-1}
ΔV_A	Velocity impulse in A	km s^{-1}
ΔV_B	Velocity impulse in B	km s^{-1}
ΔV_C	Velocity impulse in C	km s^{-1}
ΔV_D	Velocity impulse in D	km s^{-1}
e(t)	Position error	km
H	Altitude	km
A	$\frac{S\rho_S}{2m}$	
K	Induced drag factor	$km^2 s^{-1}$
J	Performance index	-
$\mathcal{H}$	Hamiltonian	?
θ	Azimuth angle	rad
ϕ	Elevation angle	rad
$\alpha_r, \alpha_{\phi\theta}$	thrust direction angles	rad

NLP	Non-linear programming	-
LEO	Low earth orbit	-
r_i	initial radius	km
r_f	final radius	km
v_r	radial Velocity	km s^{-1}
v_θ	azimuth Velocity	km s^{-1}
v_ϕ	inclination Velocity	km s^{-1}

Dimensionless Numbers

a_n	Normalised radius	R_n/R_a
K_I	Integral gain	
K_P	Proportional gain	
K_D	derivative gain	
β	Inverse atmospheric slace height	
K	Induced drag factor	
δ	$\frac{\rho}{\rho_s}$	
b	$\frac{R_a}{H_a}$	

Superscripts and subscripts

i	Initial
f	Final/Exit
$\cdot$	First derivative
$\cdot\cdot$	Second derivative
e	Entry
s	Surface area
D	Drag
D_0	Parasitic drag
L	Lift
LR	maximum lift-to-drag ratio
n	Normalised
h	Normalised altitude
γ	Flight path angle

Chapter 1

Introduction

The Low-earth orbital transfers are needed in many situations such as the re-orbit of satellites and the interceptions of intercontinental missiles. One of the most used methods is the classic one also called the Hohmann transfer which was invented in 1925, by Walter Hohmann and the most fuel-optimal orbital transfer between two circular orbits, however, this dissertation will pore over itself in the modeling of non-conventional orbital transfers, coplanar, and non-coplanar.

1.1 Hohmann Orbit Transfer

1.1.1 Coplanar orbit transfer

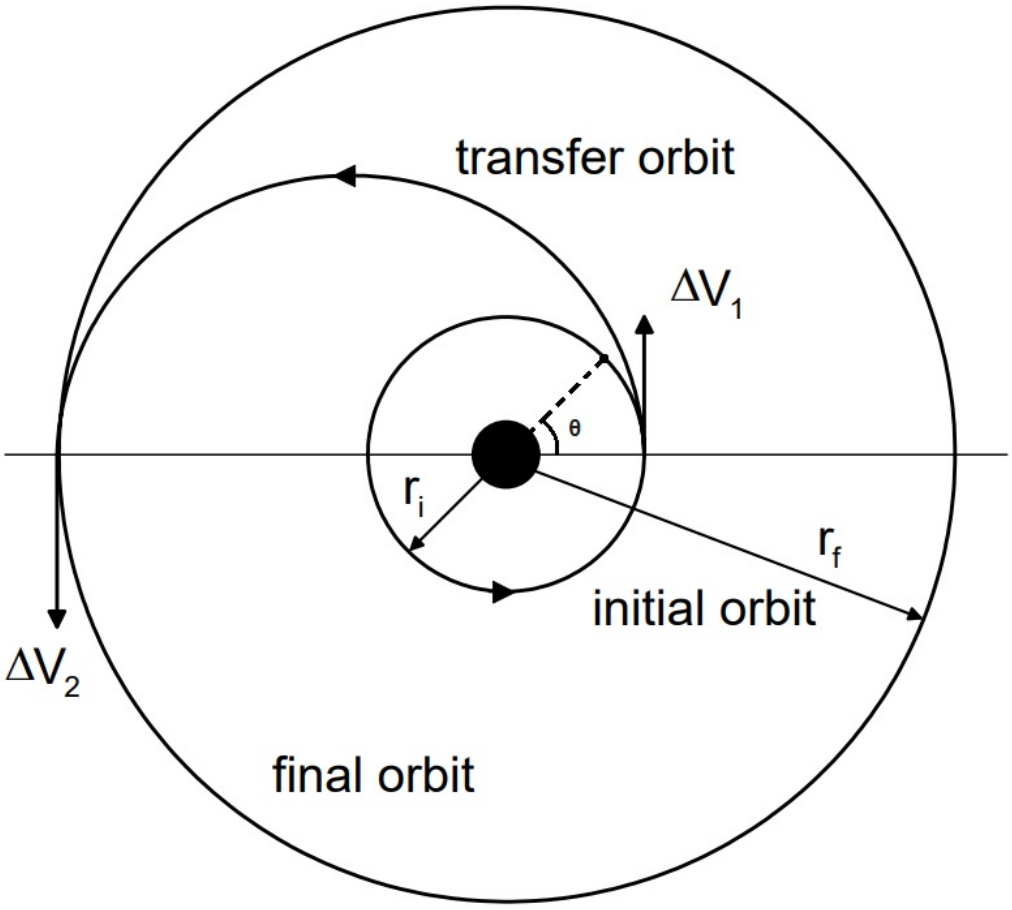


Figure 1.1: Coplanar Hohmann transfer [3]

The coplanar orbit transfer principle between two circular coplanar orbits was created in 1925 by the German engineer Walter Hohmann. This transfer that can be seen in fig.1.1 consists of a velocity impulse in the original circular orbit in the direction of the velocity vector into an elliptical orbit with a perigee equal to the altitude of the 1st circular orbit and an apogee equal to the altitude of the 2nd circular orbit, after this impulse a second impulse happens 180º from the 1st impulse into a circular orbit. Both the two main orbits (the circular ones) and the transfer orbit are in the same plane. These impulses change the velocity instantaneously therefore they represent an idealistic orbit transfer that doesn't happen in reality.

The two impulses can be obtained using the following equations:

1. For the angular moment of each orbit we have:

$$h_1 = \sqrt{2\mu} * \sqrt{\frac{r_i * r_i}{r_i + r_i}} \quad (1.1)$$

$$h_2 = \sqrt{2\mu} * \sqrt{\frac{r_f * r_f}{r_f + r_f}} \quad (1.2)$$

$$h_3 = \sqrt{2\mu} * \sqrt{\frac{r_i * r_f}{r_i + r_f}} \quad (1.3)$$

   with h_1 being the angular moment of the initial orbit, h_2 the angular moment of the final orbit, and h_3 the angular moment of the transfer orbit, μ being the gravitational parameter, r_i being the radius of the initial orbit and r_f the radius of the final orbit.

2. we can define the radius of the elliptical transfer orbit in function of the azimuth angle, θ as:

$$r_\theta = \frac{h_3^2}{\mu} \frac{1}{1 + e_3 cos(\theta)} \quad (1.4)$$

   With the eccentricity of the transfer orbit, e_3 being:

$$e_3 = \frac{r_f - r_i}{r_f + r_i} \quad (1.5)$$

3. Next we will calculate the velocities in A due to the initial orbit and transfer orbit and the velocities in B due to the final orbit and transfer orbit:

$$v_A)_1 = \frac{h_1}{r_i} \qquad v_A)_3 = \frac{h_3}{r_i} \quad (1.6)$$

$$v_B)_2 = \frac{h_2}{r_f} \qquad v_B)_3 = \frac{h_3}{r_f} \quad (1.7)$$

4. And finally for the velocities impulse we have:

$$\Delta v_A = |v_A)_3 - v_A)_1| \quad (1.8)$$

$$\Delta v_B = |v_B)_3 - v_B)_2| \quad (1.9)$$

$$\Delta v = \Delta v_A + \Delta v_B \quad (1.10)$$

5. the Transfer time,τ from the initial orbit to the final orbit is:

$$\tau = \pi \sqrt{\frac{\left(\frac{(r_i + r_f)}{2}\right)^3}{\mu}} \quad (1.11)$$

1.1.2 Non-coplanar orbit transfer

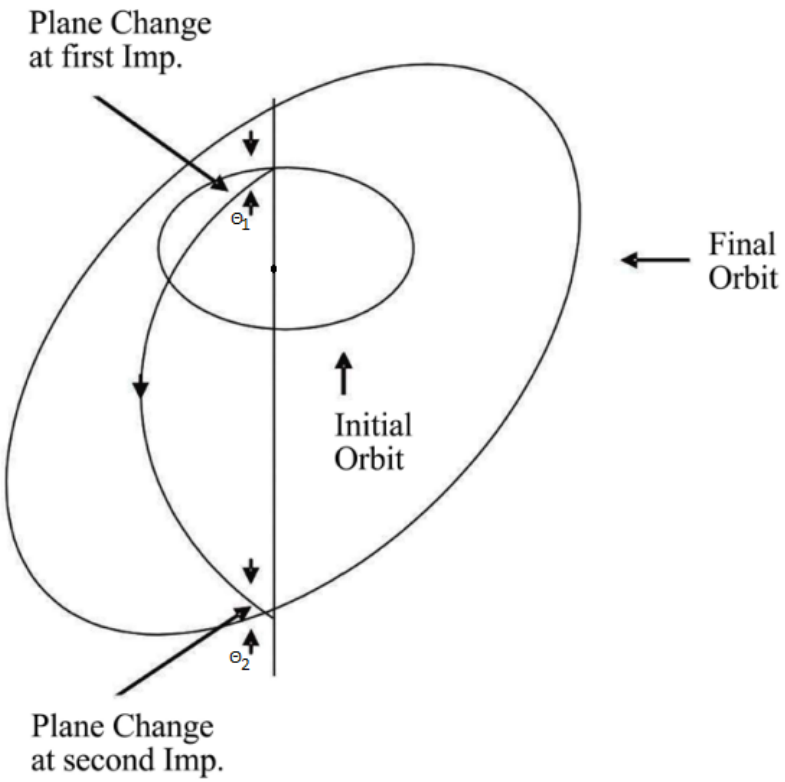


Figure 1.2: Non-coplanar orbit transfer, from Alqarni [1]

The orbit transfer between two circular non-coplanar orbits consists of an orbit transfer between circular orbits in different orbital inclinations. The first impulse will transfer the original circular orbit into an elliptical orbit(with an altitude at 0º equal to the initial orbit and an altitude at 180º equal to the final orbit) with a change in the plane of Θ_1 when compared to the original circular orbit. The second impulse will happen 180º after the first one and will transfer the elliptical orbit into a final circular orbital with a change in the plane of Θ_2 when compared to the elliptical orbit. In order to execute this orbit transfer with the minimum energy consumption, Θ_1 must be so that the partial derivative with respect to Θ_1 of the total impulse equals to zero.

The two impulses can be obtained using the following equations:

1. We start with defining the 3 normalized radii, Ri_1 being the radii of initial orbit, Ri_2 being the radii of the transfer orbit and Ri_3 being the radii of the final orbit, r_i being the radius of the initial orbit and r_f the radius of the final orbit:

$$Ri_1 = \sqrt{\frac{2 * r_f}{r_i + r_f}} \quad (1.12)$$

$$Ri_2 = \sqrt{\frac{r_i}{r_f}} \quad (1.13)$$

$$Ri_3 = \sqrt{\frac{2 * r_i}{r_i + r_f}} \quad (1.14)$$

2. Then for the velocities impulse and noting that Θ_1 is the first plane change and Θ_2 is the second plane change, we have:

$$\Delta v_1 = \sqrt{\frac{\mu}{r_i}} * \sqrt{1 + Ri_1^2 - 2Ri_1 cos\Theta_1} \quad (1.15)$$

$$\Delta v_2 = \sqrt{\frac{\mu}{r_f}} * \sqrt{Ri_2^2 + Ri_2^2 Ri_3^2 - 2Ri_2^2 Ri_3 cos\Theta_2} \quad (1.16)$$

3. Total velocity impulse and total plane change:

   Noting that the 1^{st} impulse is related to the 1^{st} plane change Θ_1 and the 2^{nd} impulse is related to the 2^{nd} plane change Θ_2, next, we can write the total impulse and the total plane change as follow:

$$\Delta v = \Delta v_1 + \Delta v_2 \quad (1.17)$$

$$\Theta_t = \Theta_1 + \Theta_2 \quad (1.18)$$

4. Then we can finally use the following equation of the partial derivative of the total velocity impulse with respect to the first plane change in order to get the plane change Θ_1 and Θ_2 that minimize the energy necessary:

$$\frac{\partial \Delta V}{\partial \Theta_1} = \frac{Ri_1 cos\Theta_1}{\sqrt{1 + Ri_1^2 - 2Ri_1 cos\Theta_1}} + \frac{Ri_2^2 Ri_3 (sin\Theta_t cos\Theta_1 - cos\Theta_t sin\Theta_1)}{\sqrt{Ri_2^2 + Ri_2^2 Ri_3^2 - 2Ri_2^2 Ri_3 cos(\Theta_t - \Theta_1)}} \quad (1.19)$$

To obtain θ_1 we have to find the value of θ_1 that equals this derivative to zero, now with θ_1 known we can use previous equation to get θ_2 and then use previous equations (1.15) and (1.16) to get the 1^{st} and 2^{nd} impulse.

1.1.3 Bi-elliptical Hohmann Orbit Transfer

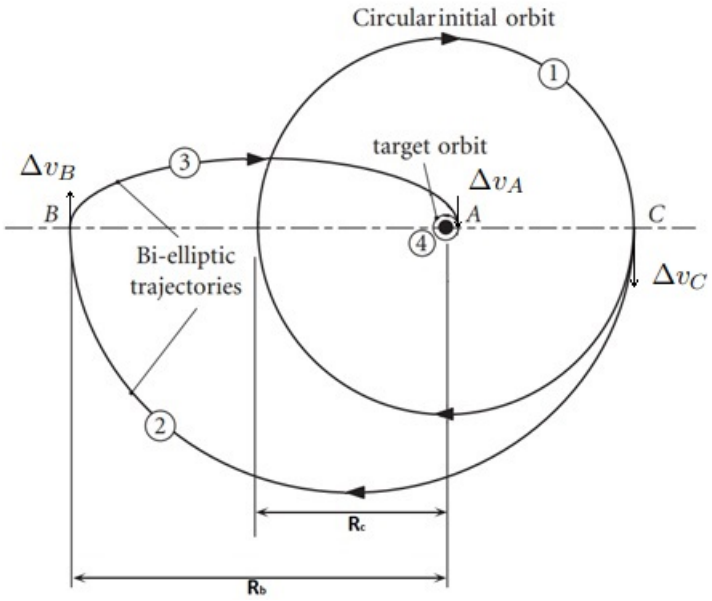


Figure 1.3: Bi-elliptical Hohmann Orbit Transfer, adapted from Howard Curtis [6]

A Bi-elliptical Hohmann transfer consists of three separate impulsive maneuvers. Starting with a circular orbit with a radius of R_c a 1^{st} impulse occurs in C resulting in an elliptical orbit With a perigee equal to the radius of the original circular orbit and an apogee of R_b, then 180º after the 1^{st} impulse a 2^{nd} impulse occurs resulting in a second elliptical orbit with an apogee of R_b this is equal to the apogee of the first elliptical orbit and a perigee of R_a. Then 180º after the 2^{nd} impulse a 3^{rd} and final impulse will transfer the 2^{nd} elliptical orbit into a circular orbit with a radius of R_a.

1. We start with defining the angular moment of each orbit with h_1 being the angular moment of the initial orbit, h_2 being the angular moment of the first elliptical orbit,

h_3 being the angular moment of the second elliptical orbit and h_4 being the angular moment of the target orbit, r_a, r_b and r_c being the radius as shown in the fig.1.3 :

$$h_1 = \sqrt{2\mu} * \sqrt{\frac{r_c}{2}} \quad \textbf{(1.20)}$$

$$h_2 = \sqrt{2\mu} * \sqrt{\frac{r_c * r_b}{r_c + r_b}} \quad \textbf{(1.21)}$$

$$h_3 = \sqrt{2\mu} * \sqrt{\frac{r_b * r_a}{r_b + r_a}} \quad \textbf{(1.22)}$$

$$h_4 = \sqrt{2\mu} * \sqrt{\frac{r_a}{2}} \quad \textbf{(1.23)}$$

2. we can define the radius of the elliptical transfer orbit in function of θ as:

$$r_\theta = \frac{h_2^2}{\mu} \frac{1}{1 + e_2 cos(\theta)} \ if \ 0 \leq \theta \leq 180 \quad \textbf{(1.24)}$$

$$r_\theta = \frac{h_3^2}{\mu} \frac{1}{1 + e_3 cos(\theta)} \ if \ \theta > 180 \quad \textbf{(1.25)}$$

With the eccentricity of the first and second elliptical orbit e_2 and 3_e being:

$$e_2 = \frac{r_b - r_c}{r_b + r_c} \quad \textbf{(1.26)}$$

$$e_3 = \frac{r_b - r_a}{r_b + r_a} \quad \textbf{(1.27)}$$

3. For the velocities in A due to the target orbit and second elliptical orbit, in B due to the first and second elliptical orbit, and in C due to the first elliptical orbit and the initial orbit we have that:

$$v_C)_1 = \frac{h_1}{r_c}; \qquad v_C)_2 = \frac{h_2}{r_c} \quad \textbf{(1.28)}$$

$$v_B)_2 = \frac{h_2}{r_b}; \qquad v_B)_3 = \frac{h_3}{r_b} \quad \textbf{(1.29)}$$

$$v_A)_3 = \frac{h_3}{r_a}; \qquad v_A)_4 = \frac{h_4}{r_a} \quad \textbf{(1.30)}$$

4. And finally using the module of subtraction of the velocities due to the different orbits in each point we obtain the following velocities impulse:

$$\Delta v_C = |v_C)_2 - v_C)_1| \quad (1.31)$$

$$\Delta v_B = |v_B)_3 - v_B)_2| \quad (1.32)$$

$$\Delta v_A = |v_A)_4 - v_A)_3| \quad (1.33)$$

With the total velocity impulses being the sum of all the velocity impulses calculated above.

$$\Delta v = \Delta v_A + \Delta v_B + \Delta v_C \quad (1.34)$$

which can be further simplified as follows :

$$\Delta v = \left[\sqrt{\frac{2(a_c + a_b)}{a_c a_b}} - \frac{1 + \sqrt{a_c}}{\sqrt{a_c}} - \sqrt{\frac{2}{a_b(1 + a_b)}}(1 - a_b)\right]\sqrt{\frac{\mu}{r_A}} \quad (1.35)$$

With the Dimensionless parameters a_c and a_b being:

$$a_c = \frac{r_C}{r_A}; \qquad a_b = \frac{r_B}{r_A} \quad (1.36)$$

5. the Transfer time, τ taken to execute the bi-elliptical transfer orbit is:

$$\tau = \pi\sqrt{\frac{\left(\frac{(r_c + r_b)}{2}\right)^3}{\mu}} + \pi\sqrt{\frac{\left(\frac{(r_b + r_a)}{2}\right)^3}{\mu}} \quad (1.37)$$

Now we can compare the Δv from the Coplanar Hohmann orbit transfer[1.1.1] with the Bi-elliptical Hohmann orbit transfer and see which a_c and a_b it is better to use one or another orbit transfer in order to reduce the fuel consumption, this comparison can be seen in the graphic 1.4.

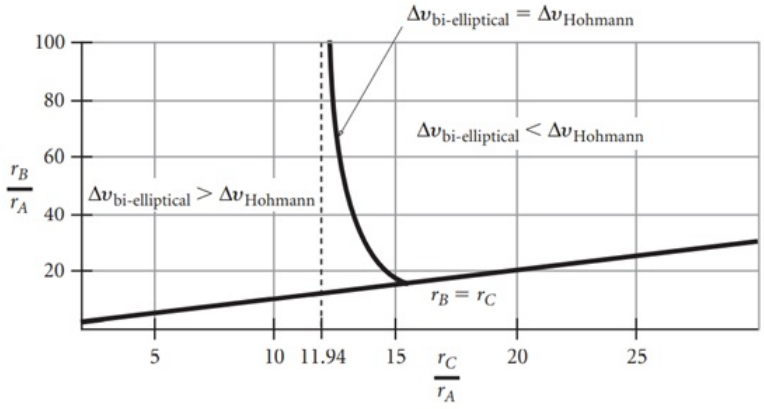


Figure 1.4: Bi-elliptical and Homann transfer fuel consumption comparison, from Howard Curtis [6]

1.2 Non-Hohmann transfer with a common apse line

A Non-Hohmann transfer consists of an orbital transfer between two coaxial orbits, this is that have the same apse line, in which the 1^{st} and 2^{nd} velocity impulses are not obligatory tangent to the initial and final orbit respectively. As shown in Fig1.5. the 1st impulse happens in A with an angle of θ_A from the apse line and the 2^{nd} impulse happens in B with an angle of θ_B from the apse line.

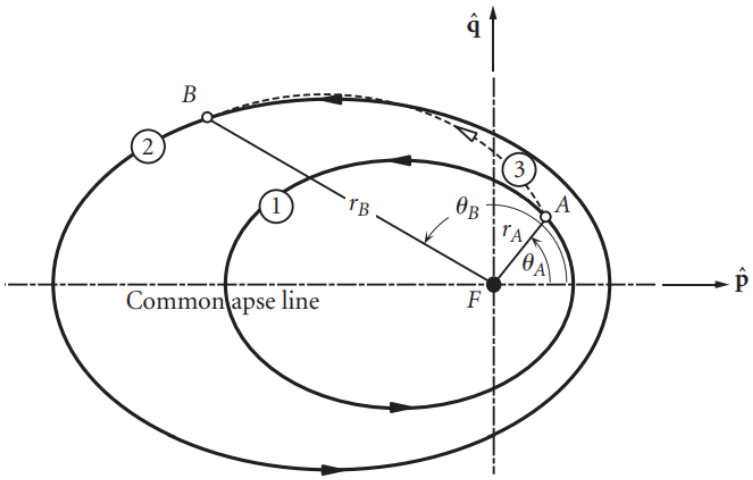


Figure 1.5: Non-Hohmann transfer between two coaxial elliptical orbits, from Howard Curtis [6]

1. From the Fig1.5 and knowing that h_3 is the angular moment of the transfer orbit, e_3 the eccentricity of the transfer orbit and θ_A and θ_B being the azimuth angle in the point A and B respectively we can write the radius r_A and r_B as follows:

$$r_A = \frac{h_3^2}{\mu} \frac{1}{1 + e_3 cos(\theta_A)} \quad (1.38)$$

$$r_B = \frac{h_3^2}{\mu}\frac{1}{1 + e_3 cos(\theta_B)} \quad (1.39)$$

2. Which we can then solve to obtain the eccentricity e_3 and the angular moment h_3 of the orbit transfer:

$$e_3 = \frac{r_B - r_A}{r_A cos(\theta_A) - r_B cos(\theta_B)} \quad (1.40)$$

$$h_3 = \sqrt{\mu r_A r_B}\sqrt{\frac{cos(\theta_A) - cos(\theta_B)}{r_A cos(\theta_A) - r_B cos(\theta_B)}} \quad (1.41)$$

3. With the eccentricity e_3 and angular moment h_3 of the transfer orbit we are able to write the radius of the transfer orbit in relation to the azimuth angle θ as follows:

$$r_\theta = \frac{h_3^2}{\mu}\frac{1}{1 + e_3 cos(\theta)} \quad (1.42)$$

4. Knowing the eccentricity of the initial orbit e_1, the eccentricity of the final orbit e_2, the the angular moment of the initial orbit h_1 and the angular moment of the final orbit h_2 we can calculate the total velocity impulse using the following equations:

$$\Delta v = \Delta v_A + \Delta v_B \quad (1.43)$$

With the velocity impulse in A, Δv_A and the velocity impulse in B, Δv_B being:

$$\begin{aligned} \Delta v_A &= \sqrt{v_{1_A}^2 + v_{3_A}^2 - 2 v_{1A} v_{3A} cos(\Delta\gamma_A)} \\ \Delta v_B &= \sqrt{v_{2_B}^2 + v_{3_B}^2 - 2 v_{2B} v_{3B} cos(\Delta\gamma_B)} \end{aligned} \quad (1.44)$$

Noting that $\Delta\gamma_A$ is the variation of the flight path angle in A and $\Delta\gamma_B$ is the variation of the flight path angle in B. With v_{1A} and v_{3A} being the velocity in A due to the initial and transfer orbit respectively and v_{2B} and v_{3B} being the velocities due to the final orbit and transfer orbit respectively. We can calculate v_{1A} using the following equation, we can also calculate the other 3 velocities using the same method:

$$v_{A_1} = \sqrt{v_{\perp_{A1}}^2 + v_{r_{A1}}^2} \quad (1.45)$$

$$v_{\perp_{A1}} = \frac{h_1}{r_A} \quad (1.46)$$

$$v_{r_{A1}} = \frac{\mu}{h_1} e_1 sin(\theta_A) \quad (1.47)$$

Noting that v_r is the radial velocity and $v_\perp$ being the tangential velocity.

And the variation of the flight path angle in A, $\Delta\gamma_A$ being obtained by the following equations and the variation of the flight path angle in B, $\Delta\gamma_B$ being obtained by using the same method:

$$\gamma_{A1} = tan^{-1}\left(\frac{v_{r_{A1}}}{v_{\perp_{A1}}}\right) \quad (1.48)$$

$$\gamma_{A3} = tan^{-1}\left(\frac{v_{r_{A3}}}{v_{\perp_{A3}}}\right) \quad (1.49)$$

$$\Delta\gamma_A = \gamma_{A3} - \gamma_{A1} \quad (1.50)$$

1.3 Aeroassisted maneuver

The first time that the use of aerodynamic forces was suggested to change the trajectory and velocity of a body was in 1961 by Howard London which later was named as Aeroassisted maneuvers in 1985 by Walberg. The use of atmospheric drag to reduce the apogee was first used by the spacecraft Hiten on the 19th of March 1991. The main advantage of using an aeroassisted maneuver between a GEO and LEO is that it economizes fuel(about 50% of fuel when compared to a Hohmann transfer) which can then increase the payload capacity of the spacecraft. The aeroassisted maneuver will consist of a multipass aerobraking strategy, this means that the transfer orbit will consist of various elliptical orbits.

1.3.1 Ideal aeroassisted maneuver

As shown in the fig.1.6 the vehicle leaves the high Earth orbit, a circular orbit with a radius of R_d, with a velocity impulse of ΔV_{di} and enters an elliptical orbit with a perigee of R_a and a flight path angle of 0^o. The grazing happens between E and F until enough velocity has been depleted by the drag, that when lift equals zero the vehicle ascends with a flight path angle final of zero. At C another velocity impulse ΔV_{ci} occurs resulting in a circular orbit LEO with a radius of R_c.

We can obtain the velocity impulses in DΔV_{di} and in C $\Delta V_d c_i$ using the following equations:

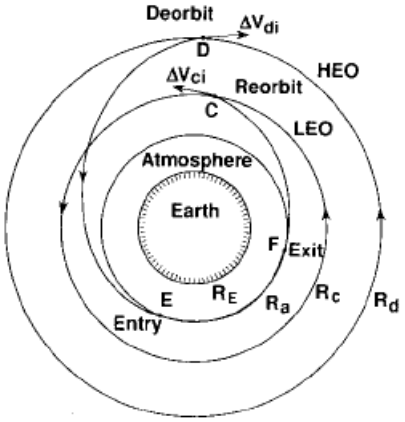


Figure 1.6: Ideal aeroassisted maneuver, from Naidu, Hibey and Charalambous [9]

$$\Delta V_{di} = \sqrt{\frac{\mu}{R_d}} - \sqrt{2\left(\frac{\mu}{R_a}\right) / \left(\left(\frac{R_d}{R_a}\right)\left(\frac{R_d}{R_a} + 1\right)\right)} \quad (1.51)$$

$$\Delta V_{ci} = \sqrt{\frac{\mu}{R_c}} - \sqrt{2\left(\frac{\mu}{R_a}\right) / \left(\left(\frac{R_c}{R_a}\right)\left(\frac{R_c}{R_a} + 1\right)\right)} \quad (1.52)$$

These velocity impulses can be normalized using the following dimensionless parameters:

$$a_d = \frac{R_d}{R_a}; \quad a_c = \frac{R_c}{R_a}; \quad \Delta v = \frac{\Delta V}{\sqrt{\frac{\mu}{R_a}}} \quad (1.53)$$

Resulting in the normalized velocity impulses ΔV_{ndi}, ΔV_{nci} and the total normalised velocity impulse ΔV_{ni} of:

$$\Delta V_{ndi} = \sqrt{\frac{1}{a_d}} - \sqrt{\frac{2}{a_d(a_d + 1)}} \quad (1.54)$$

$$\Delta V_{nci} = \sqrt{\frac{1}{a_c}} - \sqrt{\frac{2}{a_c(a_c + 1)}} \quad (1.55)$$

$$\Delta V_{ni} = \Delta V_{ndi} + \Delta V_{nci} \quad (1.56)$$

1.3.2 Realistic aeroassisted maneuver

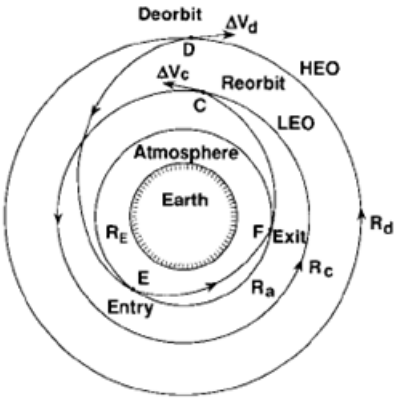


Figure 1.7: Realistic aeroassisted maneuver, from Naidu, Hibey and Charalambous [9]

The vehicle starts in a high Earth orbit (HEO), a circular orbit with a radius of R_d. A velocity impulse ΔV_d changes the orbit from HEO to an elliptical orbit(apogee of R_c), at E the spacecraft enters the atmosphere with a flight path angle of γ_e where it will lose velocity due to the drag and at F it leaves the atmosphere with a flight path angle of γ_f. When enough velocity has been depleted it ends with a circularization into LEO due to a velocity impulse of ΔV_c.

The velocity impulses ΔV_d, ΔV_d and the total velocity impulses can be obtained using the following equations:

$$\Delta V_d = \sqrt{\frac{\mu}{R_d}}\left[1 - \sqrt{2\left(\frac{1-R_d}{R_a}\right)\Big/\left(1-\left(\frac{R_d}{R_a}\right)^2/cos^2\gamma_e\right)}\right] \quad (1.57)$$

$$\Delta V_c = \sqrt{\frac{\mu}{R_c}}\left[1 - \sqrt{2\left(\frac{1-R_c}{R_a}\right)\Big/\left(1-\left(\frac{R_c}{R_a}\right)^2/cos^2\gamma_f\right)}\right] \quad (1.58)$$

These velocity impulses can be normalized using the dimensionless parameters in eq.1.53 after being normalized equal to:

$$\Delta V_{nd=}\sqrt{\frac{1}{a_d}} - \sqrt{\frac{2(1-a_d)}{a_d(1-a_d^2/cos^2\gamma_e)}} \quad (1.59)$$

$$\Delta V_{nc=}\sqrt{\frac{1}{a_c}} - \sqrt{\frac{2(1-a_c)}{a_c(1-a_c^2/cos^2\gamma_f)}} \quad (1.60)$$

$$\Delta V_n = \Delta V_{nd} + \Delta V_{nc} \quad (1.61)$$

1.3.2.1 Aeroassisted transfer controlling the radius

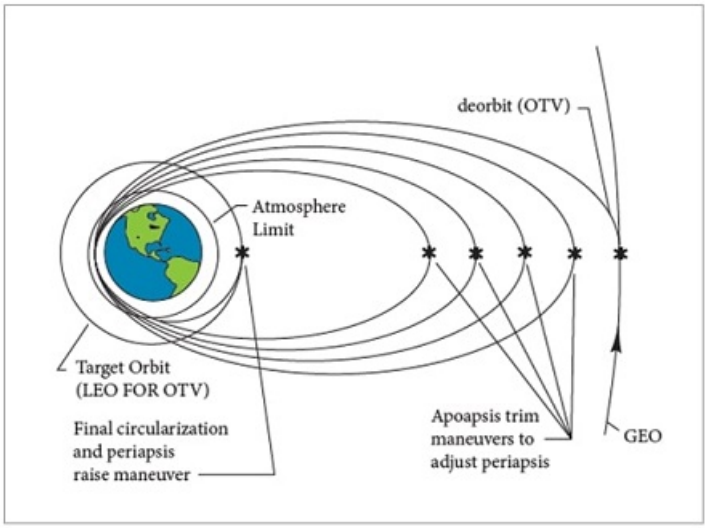


Figure 1.8: Aeroassisted maneuver between GEO and LEO, from Santos,Rocco and Carrara [12]

Starting in a GEO the spacecraft applies a velocity impulse resulting in an elliptical orbit with an apogee equal to the radius of the GEO and a perigee within the atmosphere limits, every next time the spacecraft passes in the atmosphere limits it loses velocity leading to a reduction of the apogee of the transfer orbit, this repeats until the spacecraft reaches the final apogee altitude when there is a second impulse applied to transfer the vehicle into the final orbit LEO. We will use an Aeroassisted Spacecraft maneuver Simulator based on the STRS (Spacecraft trajectory simulator), this simulator requires a reference trajectory and the trajectory that has been perturbated by external factors. We can see how much fuel is saved using the aerobraking maneuver technique using a PID (Proportional-Integral-Derivative). The main forces acting in the spacecraft will be the aerodynamic forces (drag and lift), thrusters force due to the velocity impulses, and gravitational forces.

Being the Drag F_D and Lift force F_L the following:

$$F_D = \frac{1}{2}\rho S V^2 C_D \quad (1.62)$$

$$F_L = \frac{1}{2}\rho S V^2 C_L \quad (1.63)$$

Noting that S is the reference area, V is the fluid relative velocity, ρ the air density, C_D the drag coefficient, and C_L the lift coefficient.

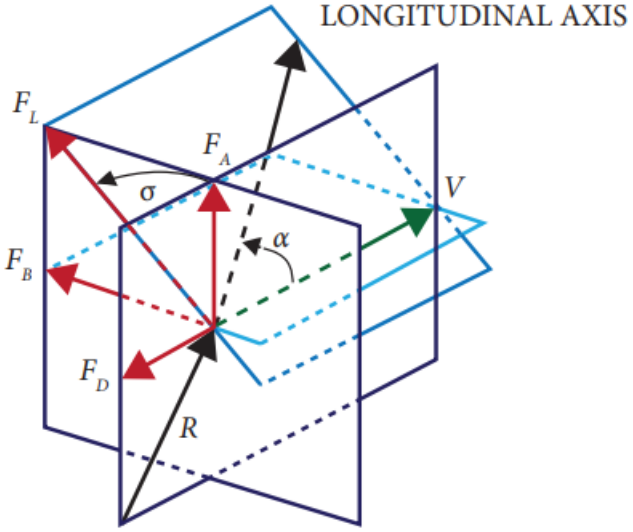


Figure 1.9: Attack angle α and bank angle σ representation, from Santos,Rocco and Carrara [12]

Knowing that the angle of attack[α] is measured between the longitudinal axis and $\hat{V}$ (velocity in relation to the atmosphere versor), and that the bank angle[σ] is measured between the lift plane and the plane formed by $\hat{V}$ and the position vector we can calculate the drag coefficient, the lift coefficient, the altitude lift C_A and the lateral lift C_B using the impact method:

$$C_D = 2(sin\alpha)^2 \quad (1.64)$$

$$C_A = C_L cos(\sigma) \quad (1.65)$$

$$C_L = 2sin(\alpha)cos(\alpha) \quad (1.66)$$

$$C_B = C_L sin(\sigma) \quad (1.67)$$

We can then decompose the lift force further into altitude lift[F_b] and lateral lift[F_a] forces

respectively:

$$F_a = \frac{1}{2}\rho S V^2 C_A \quad (1.68)$$

$$F_b = \frac{1}{2}\rho S V^2 C_B \quad (1.69)$$

Being **r** the radius versor and **V** the velocity versor we can then calculate **h** the angular momentum versor and **N** the altitude versor:

$$\mathbf{h} = \mathbf{r} \times \mathbf{V} \quad (1.70)$$

$$\mathbf{N} = \mathbf{V} \times \mathbf{h} \quad (1.71)$$

We then can obtain the Drag and Lift force:

$$F_D = -F_D \mathbf{V} \quad (1.72)$$

$$F_L = -F_A \mathbf{N} + F_B \mathbf{h} \quad (1.73)$$

The velocity in relation to the atmosphere versor can be calculated in the following way:

$$\mathbf{V} = \dot{\mathbf{r}} + \omega \times \mathbf{r} = \begin{bmatrix} \dot{x} + \omega y \\ \dot{y} - \omega x \\ \dot{z} \end{bmatrix} \quad (1.74)$$

Being the velocity vector in relation to the inertial system and ω the angular velocity vector of the Earth's rotation.

Then we can finally describe the spacecraft state using the coordinated X=[r V] with an inertial frame centered on Earth, and define the spacecraft's dynamic model with μ being the central body gravitational constant and ΔV_P the velocity impulse caused by propulsive thrusters when activated as:

$$\ddot{\mathbf{r}} = \frac{\mu}{r^3}\mathbf{r} - \frac{1}{2}C_D \rho S V \mathbf{V} + \Delta \mathbf{V_P} \quad (1.75)$$

To control the trajectory of the spacecraft we will use a PID controller c(t) that is aimed at correcting the deviation between the real trajectory and the reference trajectory of the spacecraft, with e(t) being the position error with K_P being the proportional gain, K_I the

integral gain, K_D the derivative gain and e(t) the position error, can be computed by:

$$c(t) = K_P e(t) + K_I \int e(t)dt + K_D \frac{de(t)}{dt} \quad (1.76)$$

Which can then be discretized with T being the sample period into :

$$u(k) = u(k-1) + \left(K_P + \frac{K_D}{T} + K_I T\right)e(k) + \left(- K_P - 2\frac{K_D}{T}\right)e(k-1) - \frac{K_D}{T}e(k-2) \quad (1.77)$$

1.3.2.2 Aeroassisted transfer controlling the radius, flight path angle, and velocity

In order to define the trajectory of the spacecraft we need the entry velocity v_{ne} and the exit velocity v_{nf}:

$$V_{ne} = \sqrt{\frac{2a_d(1-a_d)}{cos^2\gamma_e - a_d^2}}; \quad V_{nf} = \sqrt{\frac{2a_c(1-a_c)}{cos^2\gamma_f - a_c^2}} \quad (1.78)$$

Noting that a_d and a_c are the dimensionless parameters defined in eq.1.53, γ_e is the entry flight path angle and γ_f the exit flight path angle.
Some considerations are made such as during the aerobraking maneuver in the atmosphere the motion is planar, the control is obtained only by the lift, the Earth's rotation is neglected and drag is assumed parabolic.

knowing that:

$$A = \frac{S\rho_S}{2m}; \quad H = R - R_E; \quad \rho = \rho_s exp(-H\beta) \quad (1.79)$$

$$C_D = C_{D0} + KC_L^2 \quad (1.80)$$

With S being the aerodynamic reference area, ρ_S the fluid density, H the altitude, R the radius between the center of the spacecraft to the center of the earth, R_e being the radius of the earth, β being the inverse atmospheric scale height, C_D the drag coefficient, C_{D0} the zero-lift drag coefficient, K the induced drag factor and C_L the lift coefficient.
And knowing the dimensionless parameters:

$$\tau_n = \frac{t}{\sqrt{\frac{R_a^3}{\mu}}}; \quad V_n = \frac{V}{\sqrt{\frac{\mu}{R_a}}} \quad (1.81)$$

With τ_n being the normalized transfer time, and V_n the normalized velocity.

$$H_n = \frac{H}{H_a}; \quad b = \frac{R_a}{H_a}; \quad \delta = \frac{\rho}{\rho_s} = exp(-h\beta H_a) \quad (1.82)$$

With H_n being the normalized height, and δ being the normalized density.

$$\eta = \frac{C_L}{C_{LR}}; \quad C_{LR} = \sqrt{\frac{C_{D0}}{K}} \quad (1.83)$$

With η being the dimensionless parameters the relates the Lift coefficient C_L with the lift coefficient for maximum lift to drag ratio C_{LR}

$$A_1 = \frac{C_{D0} S \rho_S H_a}{2m}; \quad A_2 = \frac{C_{LR} S \rho_S H_a}{2m} \quad (1.84)$$

We can then obtain the equations of motion $\frac{dH}{dt}, \frac{dV}{dt}$ and $\frac{d\gamma}{dt}$ with γ being the flight path angle:

$$\frac{dH}{dt} = V sin\gamma \quad (1.85)$$

$$\frac{dV}{dt} = -AC_D V^2 exp(-H\beta) - \left(\frac{\mu}{R^2}\right) sin\gamma \quad (1.86)$$

$$\frac{d\gamma}{dt} = AC_L V exp(-H\beta) + \left[\frac{V}{R} - \frac{\mu}{R^2 V}\right] cos\gamma \quad (1.87)$$

That when normalised are:

$$\frac{dH_n}{dt} = bV_n sin\gamma \quad (1.88)$$

$$\frac{dV_n}{dt} = -A_1 b(1+\eta^2)\delta V_n^2 - \frac{V_n^2 sin\gamma_n}{(b-1+H_n)^2} \quad (1.89)$$

$$\frac{d\gamma}{dt} = A_2 b\eta\delta V_n + \frac{bV_n cos\gamma}{(b-1+H_n)} - \frac{b^2 cos\gamma}{(b-1+H_n)^2 V_n} \quad (1.90)$$

The performance index J will be defined as:

$$J = \Delta V_n = \Delta V_{nd} + \Delta V_{nc} \quad (1.91)$$

With ΔV_{nd} being the normalized velocity impulse at the deorbit from HEO and ΔV_{nc} being the normalized velocity impulse into the LEO obtained by combing the eq.1.78 and eq.1.59 and 1.60

$$\Delta V_{nd} = \sqrt{\frac{1}{a_d}} - \left(\frac{V_{ne}}{a_d}\right) cos(-\gamma_e) \quad (1.92)$$

$$\Delta V_{nc} = \sqrt{\frac{1}{a_c}} - \left(\frac{V_{nf}}{a_c}\right) cos(-\gamma_f) \quad (1.93)$$

We can conclude that in order to minimize the performance index J, the flight path angle at the entry of the atmosphere γ_e and the flight path angle at the exit of the atmosphere γ_f need to be zero then we can assume that γ(t)=0 during the grazing phase this is the altitude remains constant during this phase, therefore the optimal trajectory is equal to the ideal trajectory.

In order to minimize the fuel consumption in respect to the C_L we will use the Hamiltonian formula as follows:

$$\begin{aligned} \mathcal{H} \quad &= \lambda_h b V_n sin\gamma + \lambda_v - A_1 b(1+\eta^2)\delta V_n^2 - \tfrac{b^2 sin\gamma}{(b-1+H_n)^2} \\ &+ \lambda_\gamma A_2 b\eta\delta V_n + \tfrac{bV_n cos\gamma}{(b-1+H_n)} - \tfrac{b^2 cos\gamma}{(b-1+H_n)^2 V_n} \end{aligned} \quad (1.94)$$

With λ_h, λ_v, and λ_γ being the costates that are obtained from the following equations:

$$\frac{d\lambda_{Hn}}{d\tau} = -\frac{\delta\mathcal{H}}{\delta h} \quad (1.95)$$

$$\frac{d\lambda_{Vn}}{d\tau} = -\frac{\delta\mathcal{H}}{\delta v} \quad (1.96)$$

$$\frac{d\lambda_{\gamma}}{d\tau} = -\frac{\delta\mathcal{H}}{\delta \gamma} \quad (1.97)$$

Knowing that E_m is the maximum value of the lift-to-drag ratio and defining the optimal control as the following we can obtain η:

$$\frac{\delta\mathcal{H}}{\delta\eta} = 0 \quad (1.98)$$

$$\eta = E_m \frac{\gamma}{v\lambda_v} \qquad with \qquad E_m = (L/D)_{max} \quad (1.99)$$

We then define the boundary conditions of the non-linear TPBVP as:

$$h(\tau_n = 0) = 1; h(\tau_n = \tau_{nf}) = 1 \qquad (1.100)$$

$$(2 - v_e^2)a_d^2 - 2a_d + v_e^2 cos^2\gamma_e = 0; \quad (2 - v_f^2)a_c^2 - 2a_c + v_f^2 cos^2\gamma_f = 0 \quad (1.101)$$

We are then able to determine the optical control by calculating the non-linear TPBVP, obtained by the equations (1.95 to 1.97) and (1.88 to 1.90), which can be solved using a multiple shooting method as mentioned before.

1.4 Limitations

The Classic methods of orbital transfer such as the Hohmann transfer and bi-elliptical Hohmann transfer have the limitations of the orbit transfer being constrained to start in the periapsis or apoapsis of the initial orbit and ending at the periapsis or apoapsis of the final orbit. The problem with this limitation is that there are some specific cases in which it is required that the change of orbit happens in a different place from the periapsis or apoapsis, among those cases is for example ballistic missiles.
Another limitation is that in order to do the orbital transfer we have to do the calculations before in order to have enough propulsion all the time to keep the satellite in the transfer orbit.

1.5 Dissertation objectives

The objectives of this dissertation are to devise computational methods for the optimization of the non-conventional orbital transfers for LEO satelites, this is non-Hohmann orbit transfers by transforming the orbit transfer problem into an orbit optimization problem which allows us to start and end the orbit transfer at any point of the original and final orbit respectively. Another objective was, once the method was chosen which in this dissertation was a direct method that allows the transformation of the orbit transfer problem in a non-linear programming(NLP) method, to verify the methods of optimization of trajectories of orbital transfers computationally with simulations made in Matlab in real scenarios in this case in a low earth orbit.

1.6 Dissertation outline

In chapter 2 we will address the topic of coplanar Hohmann orbit transfer. The implementation of a computational design of this orbit will be done using the Clohessy–Wiltshire equations. This type of orbit is usually the most efficient type of orbit transfer to change the radius of the orbit however it faces some limitations such as the need to start and end at the periapsis and apoapsis. In Chapter 3 we will explain how to transform the optimization of trajectory problems into non-linear programming concepts. In chapter 4 we will address the topic of the non-Hohmann orbit transfer both coplanar and non-coplanar. The approach used will be a method of non-linear programming as explained in the previous chapter, unlike the type of orbit transfer used in Chapter 2 this one can start and end at any point of the initial and final orbit respectively.

Chapter 2

Coplanar Hohmann orbit transfer between circular orbits

2.1 Coplanar Hohmann orbit transfer design

To design this orbit transfer we will use the program GMAT also known as General Mission Analysis Tool is an open-source software developed by NASA and launched in 2007, it allows the user to design a space mission from low earth orbit to the lunar orbit or earth-mars transfer as well as orbit transfers between either coplanar or non-coplanar orbits, it also allows the user to design rendezvous maneuvers. In this paper in specific, this paper will be used to execute low earth orbit transfer, both coplanar and non-coplanar. In order to execute these maneuvers the software offers many tools such as the "spacecraft" in which we can choose many parameters such as the mass, the drag coefficient, and drag area, the tool "achieve" which allows us to determine the specifications of the orbit we want to achieve, the tool "propagate" that allows us to move the spacecraft within the same orbit, the tool "maneuver" that is used to decide when the spacecraft enters or leaves one orbit, and the "report" tool that allows the user to know the eccentricity, the radius, the true anomaly, the inclination, the Cartesian velocities among other parameters at a certain time interval during the space mission designed.

To design the transfer orbit in GMAT it is needed to first define the initial and final orbit. The initial and Final orbits were defined as shown in Table 2.1.

	Initial Orbit	Final Orbit
Radius[Km]	6500	8000
Eccentricity	0	0
Inclination[^{0}]	0	0

Table 2.1: Initial and Final Orbits

After defining the Initial and final orbits, we can then write the mission sequence as shown in Fig2.1

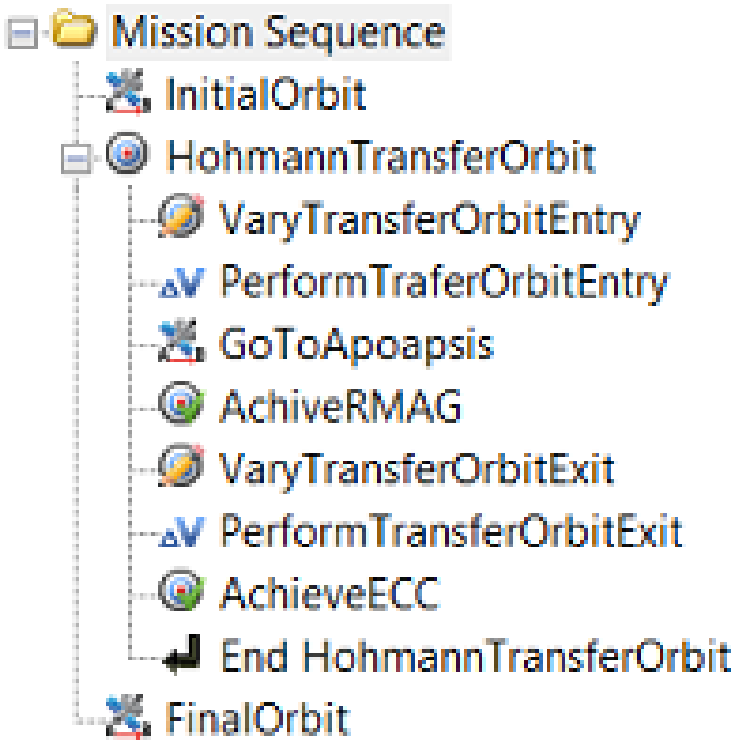


Figure 2.1: Mission Sequence

We start by adding a velocity impulse to transfer the spacecraft from the initial orbit to the transfer orbit which is then defined as having an Apoapsis of 8000km and by definition a periapsis of 6500km (equal to the radius of the initial orbit). After 180°, this is in the transfer orbit's apoapsis, we add another velocity impulse to transfer the spacecraft from the transfer orbit to the final orbit which is defined as having an eccentricity of 0(circular orbit). Obtaining the orbit transfer shown in fig.2.2

After defining the mission sequence, it is possible to obtain the magnitude of the velocity impulses that are needed to fulfill the mission requirement as shown in the fig2.3:
As shown in Fig.2.3 the velocity impulses are $\Delta v_1 = 0.39449 ms^{-1}$ and $\Delta v_2 = 0.37652 ms^{-1}$ and therefore a total velocity impulse of $\Delta v = 0.77091 ms^{-1}$.

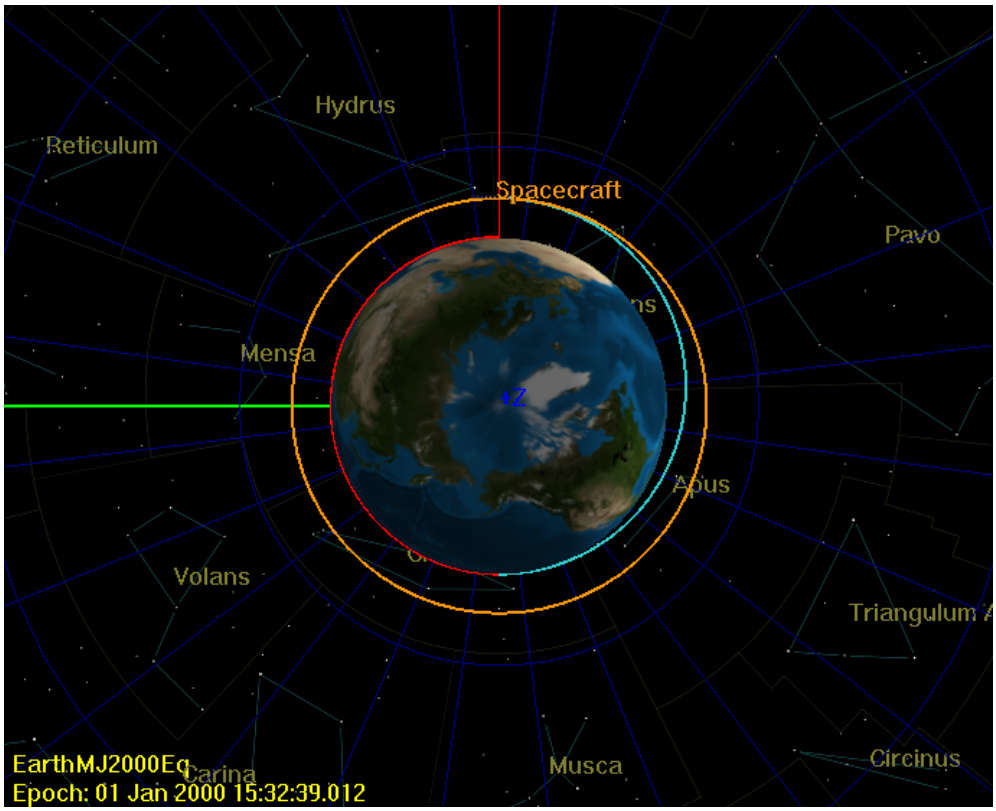


Figure 2.2: Coplanar orbit transfer in gmat

Solver Window - Target 'HohmannTransferOrbit' DC1 {SolveMode = Solve, ExitMode = Discard...

Control Variable	Current Value	Last Value	Difference
TransferOrbitEntry.Element1	0.3944854230215605	0.3944854230215605	5.551115123125783e-17
TransferOrbitExit.Element1	0.3765227320558427	0.3765227320558427	0
Constraints	**Desired**	**Achieved**	**Difference**
(==) Spacecraft.Earth.RMAG	8000	7999.999999999985	-1.546140993013978e-11

CONVERGED

Figure 2.3: GMAT Solver

2.2 Computational Procedure

2.2.1 Interpolation

The interpolation of the data of the arc of transfer is needed because the data comes in the numerical format but in order to deal with it more efficiently in the computational context it is needed to represent that data with functions. For the interpolation of the data obtained by GMAT, we are going to use the cubic spline interpolation[8] which is a mathematical method used to obtain new points within the boundaries of known points. These new points are obtained from the interpolation functions that consist of multiple

cubic piecewise polynomials. Having $n+1$(x_1 to x_{n+1}) original points we will have n third degree polynomials f_1 to f_n.

The polynomials will be written as follows:

$$f(x) = \begin{cases} a_1x^3 + b_1x^2 + c_1x + d_1 & \text{if } x \in [x_1, x_2] \\ a_2x^3 + b_2x^2 + c_2x + d_2 & \text{if } x \in [x_2, x_3] \\ \dots \\ a_nx^3 + b_nx^2 + c_nx + d_n & \text{if } x \in [x_n, x_{n+1}] \end{cases} \tag{2.1}$$

To determine the coefficients of the polynomials we need to use several equations that are the following:

$$\begin{cases} f_1(x_1) = y_1 \\ f_1(x_2) = y_2 \\ f_2(x_2) = y_2 \\ \dots \\ f_n(x_n) = y_n \\ f_n(x_{n+1}) = y_{n+1} \end{cases} \tag{2.2}$$

Noting that the y_2 must be the same using the function f_1 or $f_2$for x_2 this is y_i must be the same using the polynomial f_{i-1} and f_i for x_i.

We also know that the first derivative and second derivative of 2 adjacent polynomials have the same value in the points they touch. For example, the first derivative of the polynomial 1 for x_2 has the same value as the first derivative of the polynomial 2 for x_2, and the same goes for the second derivative.

For the first derivative, we have:

$$\begin{cases} \frac{d}{dx} f_1(x) = \frac{d}{dx} f_2(x) \mid_{x=x_2} \\ \frac{d}{dx} f_2(x) = \frac{d}{dx} f_3(x) \mid_{x=x_3} \\ \dots \\ \frac{d}{dx} f_{n-1}(x) = \frac{d}{dx} f_n(x) \mid_{x=x_n} \end{cases} \tag{2.3}$$

For the second derivative, we have:

$$\begin{cases} \frac{d^2}{dx^2} f_1(x) = \frac{d^2}{dx^2} f_2(x) \mid_{x=x_2} \\ \frac{d^2}{dx^2} f_2(x) = \frac{d^2}{dx^2} f_3(x) \mid_{x=x_3} \\ \dots \\ \frac{d^2}{dx^2} f_{n-1}(x) = \frac{d^2}{dx^2} f_n(x) \mid_{x=x_n} \end{cases} \tag{2.4}$$

For the last coefficients of the two 1st and two last polynomials we will use the Not-a-Knot Spline condition in which we have that the third derivative in the same point has the same value:

$$\begin{cases} \frac{d^3}{dx^3} f_1(x) = \frac{d^3}{dx^3} f_2(x) \mid_{x=x_2} \\ \frac{d^3}{dx^3} f_{n-1}(x) = \frac{d^3}{dx^3} f_n(x) \mid_{x=x_n} \end{cases} \tag{2.5}$$

2.2.1.1 practical example

Considering a dataset of 10 points as shown in the table. the interpolation procedure will be as follows:

Time[s]	0	1	2	3	4	5
x[km]	0	10	18	24	28	30
y[km]	30	28	24	18	10	0

Table 2.2: Example of values for an interpolation

We start by writing the polynomials as such:

$$f_x(t) = \begin{cases} a_{x1}t^3 + b_{x1}t^2 + c_{x1}t + d_{x1} & \text{if } t \in [0,1] \\ a_{x2}t^3 + b_{x2}t^2 + c_{x2}t + d_{x2} & \text{if } t \in [1,2] \\ a_{x3}t^3 + b_{x3}t^2 + c_{x3}t + d_{x3} & \text{if } t \in [2,3] \\ a_{x4}t^3 + b_{x4}t^2 + c_{x4}t + d_{x4} & \text{if } t \in [3,4] \\ a_{x5}t^3 + b_{x5}t^2 + c_{x5}t + d_{x5} & \text{if } t \in [4,5] \end{cases} \tag{2.6}$$

$$f_y(t) = \begin{cases} a_{y1}t^3 + b_{y1}t^2 + c_{y1}t + d_{y1} & \text{if } t \in [0,1] \\ a_{y2}t^3 + b_{y2}t^2 + c_{y2}t + d_{y2} & \text{if } t \in [1,2] \\ a_{y3}t^3 + b_{y3}t^2 + c_{y3}t + d_{y3} & \text{if } t \in [2,3] \\ a_{y4}t^3 + b_{y4}t^2 + c_{y4}t + d_{y4} & \text{if } t \in [3,4] \\ a_{y5}t^3 + b_{y5}t^2 + c_{y5}t + d_{y5} & \text{if } t \in [4,5] \end{cases} \tag{2.7}$$

In order to obtain the coefficients the eq.2.2 to eq.2.5 need to be solved, from the eq.2.2 we have that:

$$\begin{cases} d_{x1} = 0 \\ a_{x1} + b_{x1} + c_{x1} + d_{x1} = 10 = a_{x2} + b_{x2} + c_{x2} + d_{x2} \\ 8a_{x2} + 4b_{x2} + 2c_{x2} + d_{x2} = 18 = 8a_{x3} + 4b_{x3} + 2c_{x3} + d_{x3} \\ 27a_{x3} + 9b_{x3} + 3c_{x3} + d_{x3} = 24 = 27a_{x4} + 9b_{x4} + 3c_{x4} + d_{x4} \\ 64a_{x4}t + 16b_{x4} + 4c_{x4} + d_{x4} = 28 = 64a_{x5} + 16b_{x5} + 4c_{x5} + d_{x5} \\ 125a_{x5}t + 25b_{x5} + 5c_{x5} + d_{x5} = 30 \end{cases} \tag{2.8}$$

$$\begin{cases} d_{y1} = 30 \\ a_{y1} + b_{y1} + c_{y1} + d_{y1} = 28 = a_{y2} + b_{y2} + c_{y2} + d_{y2} \\ 8a_{y2} + 4b_{y2} + 2c_{y2} + d_{y2} = 24 = 8a_{y3} + 4b_{y3} + 2c_{y3} + d_{y3} \\ 27a_{y3} + 9b_{y3} + 3c_{y3} + d_{y3} = 18 = 27a_{y4} + 9b_{y4} + 3c_{y4} + d_{y4} \\ 64a_{y4}t + 16b_{y4} + 4c_{y4} + d_{y4} = 10 = 64a_{y5} + 16b_{y5} + 4c_{y5} + d_{y5} \\ 125a_{y5}t + 25b_{y5} + 5c_{y5} + d_{y5} = 0 \end{cases} \tag{2.9}$$

Then from the eq.2.3 we have that:

$$\begin{cases} 3a_{x1} + 2b_{x1} + c_{x1} = 3a_{x2} + 2b_{x2} + c_{x2} \\ 12a_{x2} + 4b_{x2} + c_{x2} = 12a_{x3} + 4b_{x3} + c_{x3} \\ 27a_{x3} + 6b_{x3} + c_{x3} = 27a_{x4} + 6b_{x4} + c_{x4} \\ 48a_{x4} + 8b_{x4} + c_{x4} = 48a_{x5} + 8b_{x5} + c_{x5} \end{cases} \tag{2.10}$$

$$\begin{cases} 3a_{y1} + 2b_{y1} + c_{y1} = 3a_{y2} + 2b_{y2} + c_{y2} \\ 12a_{y2} + 4b_{y2} + c_{y2} = 12a_{y3} + 4b_{y3} + c_{y3} \\ 27a_{y3} + 6b_{y3} + c_{y3} = 27a_{y4} + 6b_{y4} + c_{y4} \\ 48a_{y4} + 8b_{y4} + c_{y4} = 48a_{y5} + 8b_{y5} + c_{y5} \end{cases} \tag{2.11}$$

Then from the eq.2.4 we have that:

$$\begin{cases} 6a_{x1} + 2b_{x1} = 6a_{x2} + 2b_{x2} \\ 12a_{x2} + 2b_{x2} = 12a_{x3} + 2b_{x3} \\ 18a_{x3} + 2b_{x3} = 18a_{x4} + 2b_{x4} \\ 24a_{x4} + 2b_{x4} = 24a_{x5} + 2b_{x5} \end{cases} \tag{2.12}$$

$$\begin{cases} 6a_{y1} + 2b_{y1} = 6a_{y2} + 2b_{y2} \\ 12a_{y2} + 2b_{y2} = 12a_{y3} + 2b_{y3} \\ 18a_{y3} + 2b_{y3} = 18a_{y4} + 2b_{y4} \\ 24a_{y4} + 2b_{y4} = 24a_{y5} + 2b_{y5} \end{cases} \tag{2.13}$$

Then from eq.2.5 we have that:

$$\begin{cases} a_{x1} = a_{x2} \\ a_{x4} = a_{x5} \end{cases} \tag{2.14}$$

$$\begin{cases} a_{y1} = a_{y2} \\ a_{y4} = a_{y5} \end{cases} \tag{2.15}$$

using all the equations described above we obtain the following coefficients:

$$
\begin{cases}
a_{x1} = 0.1716 \\
b_{x1} = -1.51482 \\
c_{x1} = 11.3432 \\
d_{x1} = 0 \\
a_{x2} = 0.1716 \\
b_{x2} = -1.51482 \\
c_{x2} = 11.3432 \\
d_{x2} = 0 \\
a_{x3} = -0.02899 \\
b_{x3} = -0.31123 \\
c_{x3} = -3.3829 \\
d_{x3} = 26.2426 \\
a_{x4} = 1.3928 \\
b_{x4} = -13.1072 \\
c_{x4} = 35.0051 \\
d_{x4} = 8.5569 \\
a_{x5} = 1.3928 \\
b_{x5} = -13.1072 \\
c_{x5} = 35.0051 \\
d_{x5} = 8.5569
\end{cases}
\tag{2.16}
$$

$$
\begin{cases}
a_{y1} = 0.1716 \\
b_{y1} = -1.51482 \\
c_{y1} = -0.6568 \\
d_{y1} = 30 \\
a_{y2} = 0.1716 \\
b_{y2} = -1.51482 \\
c_{y2} = -0.6568 \\
d_{y2} = 30 \\
a_{y3} = -0.02899 \\
b_{y3} = -0.3112 \\
c_{y3} = -15.3829 \\
d_{y3} = 56.2426 \\
a_{y4} = 1.3928 \\
b_{y4} = -13.1072 \\
c_{y4} = 23.0051 \\
d_{y4} = 33.5569 \\
a_{y5} = 1.3928 \\
b_{y5} = -13.1072 \\
c_{y5} = 23.0051 \\
d_{y5} = 38.5569
\end{cases}
\tag{2.17}
$$

With the coefficient values we can now write the polynomials as such:

$$
f_x(t) = \begin{cases}
0.1716t^3 - 1.51482t^2 + 11.3432t & \text{if } t \in [0,1] \\
0.1716t^3 - 1.51482t^2 + 11.3432t & \text{if } t \in [1,2] \\
-0.02899t^3 - 0.31123t^2 - 3.3829t + 26.2426 & \text{if } t \in [2,3] \\
1.3928t^3 - 13.1072t^2 + 35.0051t + 8.5569 & \text{if } t \in [3,4] \\
1.3928t^3 - 13.1072t^2 + 35.0051t + 8.5569 & \text{if } t \in [4,5]
\end{cases}
\tag{2.18}
$$

$$f_y(t) = \begin{cases} 0.1716t^3 - 1.51482t^2 - 0.6568t + 30 & \text{if } t \in [0,1] \\ 0.1716t^3 - 1.51482t^2 - 0.6568t + 30 & \text{if } t \in [1,2] \\ -0.02899t^3 - 0.3112t^2 - 15.3829t + 56.2426 & \text{if } t \in [2,3] \\ 1.3928t^3 - 13.1072t^2 + 23.0051t + 33.5569 & \text{if } t \in [3,4] \\ 1.3928t^3 - 13.1072t^2 + 23.0051t + 38.5569 & \text{if } t \in [4,5] \end{cases} \quad (2.19)$$

2.2.2 Relative dynamic equations

The model we are going to use for the control of the transfer orbit trajectory is going to be the Clohessy–Whiltshire equations that are deducted as follows:

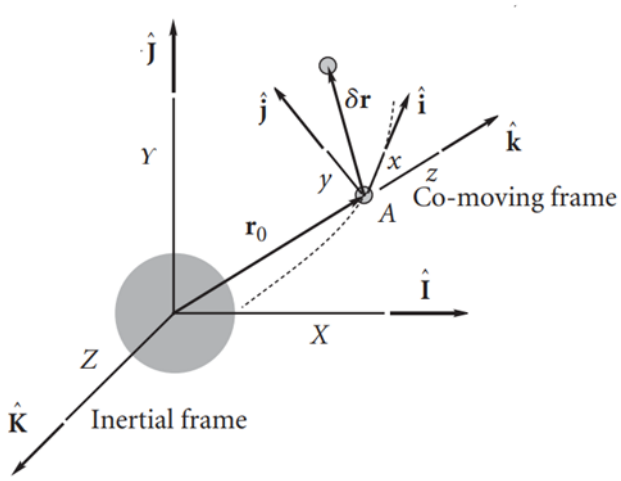


Figure 2.4: Clohessy–Wiltshire frame

As can see in Fig.2.4, the radius versor $\mathbf{r}_0$ with a magnitude of r lies along the axis x so we have that:

$$\hat{\mathbf{i}} = \frac{\mathbf{r}_0}{r_0} \quad (2.20)$$

We know that the relative acceleration versor $\ddot{\mathbf{r}}$ formula is:

$$\ddot{\mathbf{r}} = \ddot{\mathbf{r}_0} + \dot{\mathbf{w}} \times \delta\mathbf{r} + \mathbf{w} \times (\mathbf{w} \times \delta\mathbf{r}) + 2\mathbf{w} \times \delta\mathbf{v}_{rel} + \delta\mathbf{a}_{rel} \quad (2.21)$$

Noting that $\dot{\mathbf{w}}$ is ,$\mathbf{w}$ is the angular velocity versor , $\delta\mathbf{v}_{rel}$ is the relative velocity versor and $\delta\mathbf{a}_{rel}$ is the relative acceleration versor

And that the relative position, velocity, and acceleration versor formula are:

$$\delta\mathbf{r} = \delta x\hat{\mathbf{i}} + \delta y\hat{\mathbf{j}} + \delta z\hat{\mathbf{k}} \tag{2.22a}$$

$$\delta\mathbf{v}_{rel} = \delta\dot{x}\hat{\mathbf{i}} + \delta\dot{y}\hat{\mathbf{j}} + \delta\dot{z}\hat{\mathbf{k}} \tag{2.22b}$$

$$\delta\mathbf{a}_{rel} = \delta\ddot{x}\hat{\mathbf{i}} + \delta\ddot{y}\hat{\mathbf{j}} + \delta\ddot{z}\hat{\mathbf{k}} \tag{2.22c}$$

Knowing that $\dot{\mathbf{w}} = 0$ and $\mathbf{r} = \mathbf{r}_0 + \delta\mathbf{r}$ and with the equation (2.21) we obtain that:

$$\delta\ddot{\mathbf{r}} = \mathbf{w} \times (\mathbf{w} \times \delta\mathbf{r}) + 2\mathbf{w} \times \delta\mathbf{v}_{rel} + \delta\mathbf{a}_{rel} = \mathbf{w}(\mathbf{w} \cdot \delta\mathbf{r}) - w^2\delta\mathbf{r} + 2\mathbf{w} \times \delta\mathbf{v}_{rel} + \delta\mathbf{a}_{rel} \tag{2.23}$$

We can also write the angular velocity versor as:

$$\mathbf{w} = w\hat{j} \tag{2.24}$$

Obtaining that:

$$\mathbf{w} \cdot \delta\mathbf{r} = w\hat{\mathbf{j}} \cdot (\delta x\hat{\mathbf{i}} + \delta y\hat{\mathbf{j}} + \delta z\hat{\mathbf{k}}) = w\delta y \tag{2.25}$$

and

$$\mathbf{w} \times \delta\mathbf{v}_{rel} = w\hat{\mathbf{j}} \times (\delta\dot{x}\hat{\mathbf{i}} + \delta\dot{y}\hat{\mathbf{j}} + \delta\dot{z}\hat{\mathbf{k}}) = w\delta\dot{z}\hat{\mathbf{i}} - w\delta\dot{x}\hat{\mathbf{k}} \tag{2.26}$$

Finally we can combine the equations (2.24) (2.25) (2.26) and (2.22) into the equation (2.23) obtaining:

$$\delta\ddot{\mathbf{r}} = (-w^2\delta x + 2w\delta\dot{z} + \delta\ddot{x})\hat{\mathbf{i}} + (\delta\ddot{y})\hat{\mathbf{j}} + (-w^2\delta z - 2w\delta\dot{x} + \delta\ddot{z})\hat{\mathbf{k}} \tag{2.27}$$

Combining the equations (2.20) and (2.22) we know that:

$$\mathbf{r}_0 \cdot \delta\mathbf{r} = (r_0\hat{\mathbf{k}}) \cdot (\delta x\hat{\mathbf{i}} + \delta y\hat{\mathbf{j}} + \delta z\hat{\mathbf{k}}) = r_0\delta z \tag{2.28}$$

Knowing the equation of motion $\delta\ddot{\mathbf{r}}$:

$$\delta\ddot{\mathbf{r}} = -\frac{\mu}{r_0^3}\left[\delta\mathbf{r} - \frac{3}{r_0^2}(\mathbf{r}_0 \cdot \delta\mathbf{r})\mathbf{r}_0\right] \tag{2.29}$$

and noting that:

$$\frac{\mu}{r_0^3} = w^2 \tag{2.30}$$

With this information we can use the equations (2.22),(2.30),(2.28) into the equation of

motion (2.29) and obtain that:

$$\delta\ddot{\mathbf{r}} = (-w^2\delta x)\hat{\mathbf{i}} + (-w^2\delta y)\hat{\mathbf{j}} + (2w^2\delta z)\hat{\mathbf{k}} \quad (2.31)$$

And finally combining the equation (2.27) with (2.31) we obtain the following equations of Clohessy–Wiltshire that define the orbital relative motion:

$$\delta\ddot{x} + 2w\delta\dot{z} = 0 \quad (2.32a)$$
$$\delta\ddot{y} + w^2\delta y = 0 \quad (2.32b)$$
$$\delta\ddot{z} - 2w\delta\dot{x} - 3w^2\delta z = 0 \quad (2.32c)$$

And therefore having the model that we are going to use written as:

$$\ddot{x} + 2w\dot{z} = a_x \quad (2.33a)$$
$$\ddot{y} + w^2 y = a_y \quad (2.33b)$$
$$\ddot{z} - 2w\dot{x} - 3w^2 z = a_z \quad (2.33c)$$

In the case of the coplanar orbit transfer in which both the initial orbit the transfer orbit and the final orbit are in the same plane(z=0), We only have equations (2.33a) and (2.33b) for x and y written in the following way:

$$\ddot{x} = a_x \quad (2.34a)$$
$$\ddot{y} + w^2 y = a_y \quad (2.34b)$$

Which we can then write as:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{V}_x \\ \dot{V}_y \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & w^2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ V_x \\ V_y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \end{bmatrix} \quad (2.35)$$

being:

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2.36)$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & w^2 & 0 & 0 \end{bmatrix} \quad (2.37)$$

From the above, we can see that Matrix A depends on the angular velocity W, and as we know the angular velocity in an elliptical orbit(transfer orbit) changes with the change in the radius so we will study 2 cases. Having the Matrix, A and B for the LQR we can see that:
-In the first case we will use the average angular velocity w value calculated with the average between the maximum and minimum value of the angular velocity during the orbital transfer, and therefore having only one matrix A.
-In the second case we will have five matrices A each one with a different angular velocity being the angular velocities used W_{min}, W_{med1}, W_{med}, $W_{med2}$and W_{max} equally spaced as shown below:

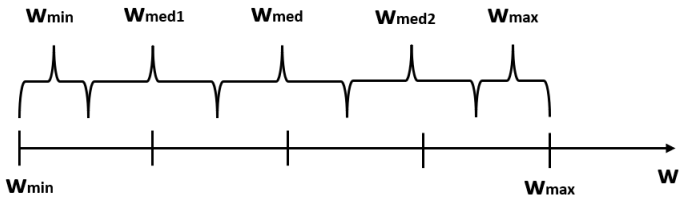


Figure 2.5: The angular velocities used in CaseII

Written mathematically we have that:

$$w = W_{max} \text{ ,if } \sqrt{\frac{\mu}{Raio^3}} >= \frac{(W_{med2} + W_{max})}{2} \quad (2.38)$$

$$w = W_{med2} \text{ ,if } \frac{W_{med2} + W_{max}}{2} >= \sqrt{\frac{\mu}{Raio^3}} > \frac{W_{med2} + W_{med}}{2} \quad (2.39)$$

$$w = W_{med} \text{ ,if } \frac{W_{med2} + W_{med}}{2} >= \sqrt{\frac{\mu}{Raio^3}} > \frac{W_{med1} + W_{med}}{2} \quad (2.40)$$

$$w = W_{med1} \text{ ,if } \frac{W_{med1} + W_{med}}{2} >= \sqrt{\frac{\mu}{Raio^3}} > \frac{W_{med1} + W_{min}}{2} \quad (2.41)$$

$$w = W_{min} \text{ ,if } \frac{W_{med1} + W_{med}}{2} >= \sqrt{\frac{\mu}{Raio^3}} > W_{min} \quad (2.42)$$

With $W_{min}, W_{med}, W_{med1}, W_{med2}$ and W_{max} being:

$$W_{min} = \sqrt{\frac{\mu}{R_{max}^3}} \quad (2.43a)$$

$$W_{max} = \sqrt{\frac{\mu}{R_{min}^3}} \quad (2.43b)$$

$$W_{med} = \frac{W_{min} + W_{max}}{2} \quad (2.43c)$$

$$W_{med1} = \frac{W_{min} + W_{med}}{2} \quad (2.43d)$$

$$W_{med2} = \frac{W_{max} + W_{med}}{2} \quad (2.43e)$$

In this model, the x, y, and z (in this specific problem only x, and y) are the distance between the theoretical orbit and the real orbit, in order to make this distance the minimum possible during the transfer we use the model described in 2.35 as a system controlled to regulate the dynamic of these same variables (x and y)

After having the Matrices A and B we will need to define the control matrices Q and R, with the matrix Q defining the weights of the states and the matrix R defining weights on the control input in the cost function, both these matrices are symmetric and positive-defined. The matrix Q and R are obtained using the following equations:

$$Q = \mu_Q * Id(4) = 25 * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 25 & 0 & 0 & 0 \\ 0 & 25 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 25 \end{bmatrix} \quad (2.44)$$

$$R = \mu_R * Id(2) = 0,90 * \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0,9 & 0 \\ 0 & 0,9 \end{bmatrix} \quad (2.45)$$

Having the Matrices A, B, Q, and R we can use the LQR (linear quadratic Regulator) to obtain the Matrix P that satisfies the equation (2.46).

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (2.46)$$

With the matrix P, we can obtain the matrix K as follows:

$$K = R^{-1}B^T P \quad (2.47)$$

We can write the continuous time state model (2.35) as follows:

$$\dot{x} = Ax + Bu_c \quad (2.48)$$

The control vector used for this state model will be as follows:

$$u_c = -Kx \quad (2.49)$$

From (2.48) and (2.49) we can write the discrete model as:

$$\begin{cases} x_{k+1} = A_d x_k + B_d u_{c,k} \\ u_{c,k} = -Kx_k \end{cases} \quad (2.50)$$

For that we need to discretize the matrices A and B as follows:

$$\begin{cases} A_d = e^{Ah} \\ B_d = (Ih + \frac{(Ah^2)}{2!} + ... + \frac{A^{N-1}h^N}{N!})B \end{cases} \quad (2.51)$$

With h being the discretization step and $||\frac{(Ah)^N}{N!}||_F < \varepsilon, \varepsilon$ being the precision
As we can see to start the iteration process we need to have x_0 first for which we used the following values:

$$x_0 = \begin{bmatrix} -10 \\ 10 \\ 0 \\ 0 \end{bmatrix} \quad (2.52)$$

2.2.3 Results

2.2.3.1 Case I

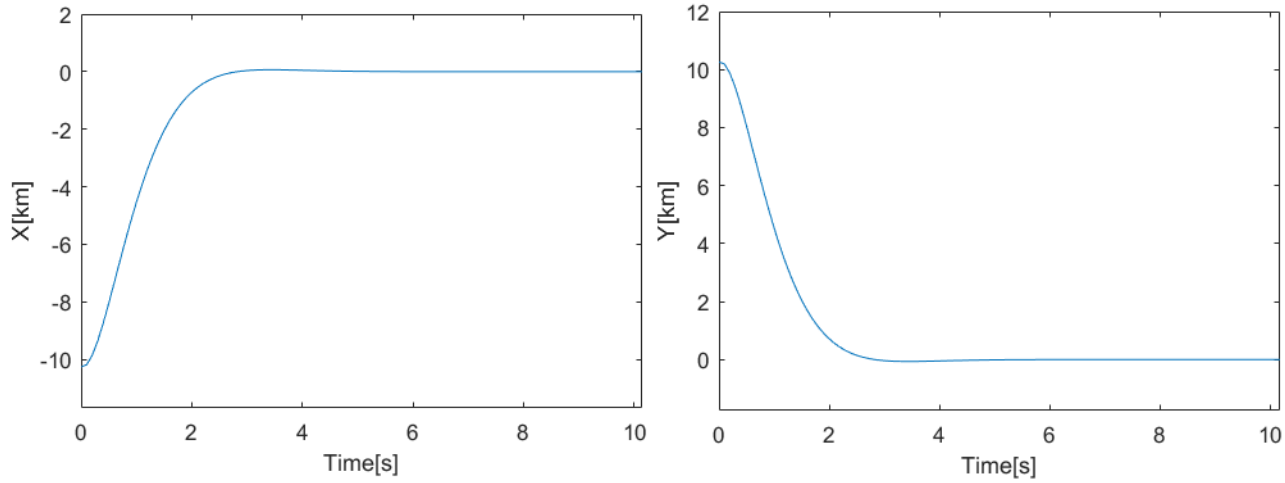


Figure 2.6: x and y vs time(Hohmann transfer Case I)

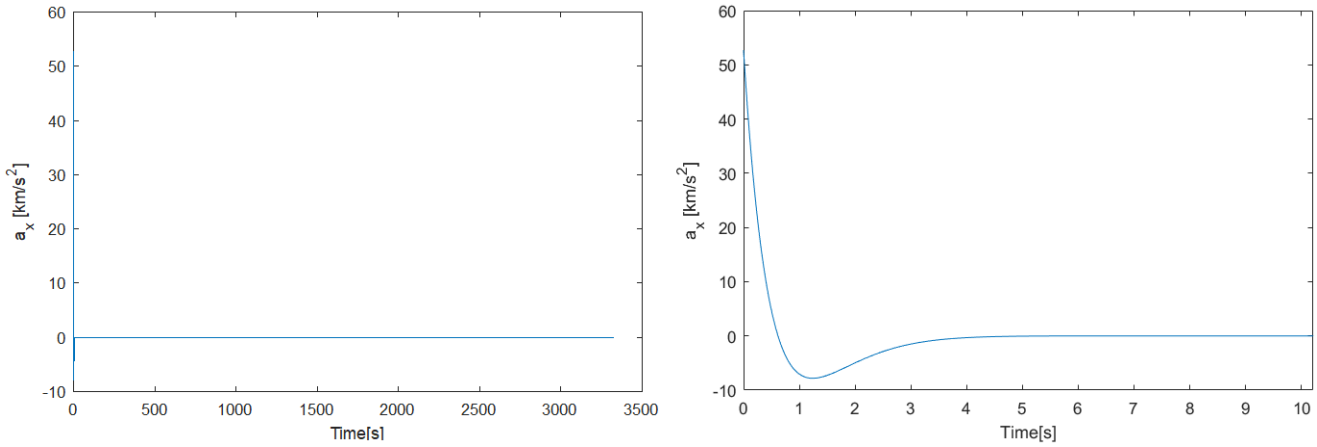


Figure 2.7: a_x vs time(Hohmann transfer Case I)

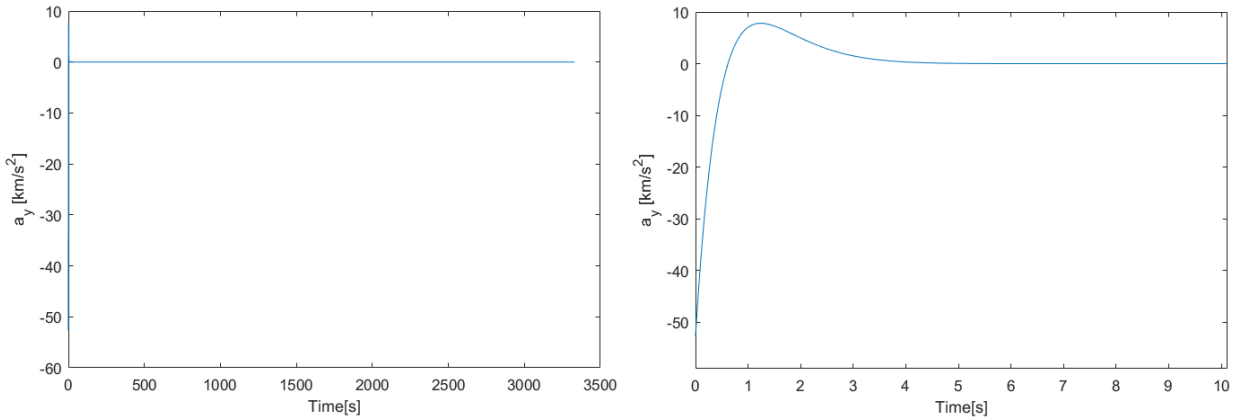


Figure 2.8: a_y vs time(Hohmann transfer Case I)

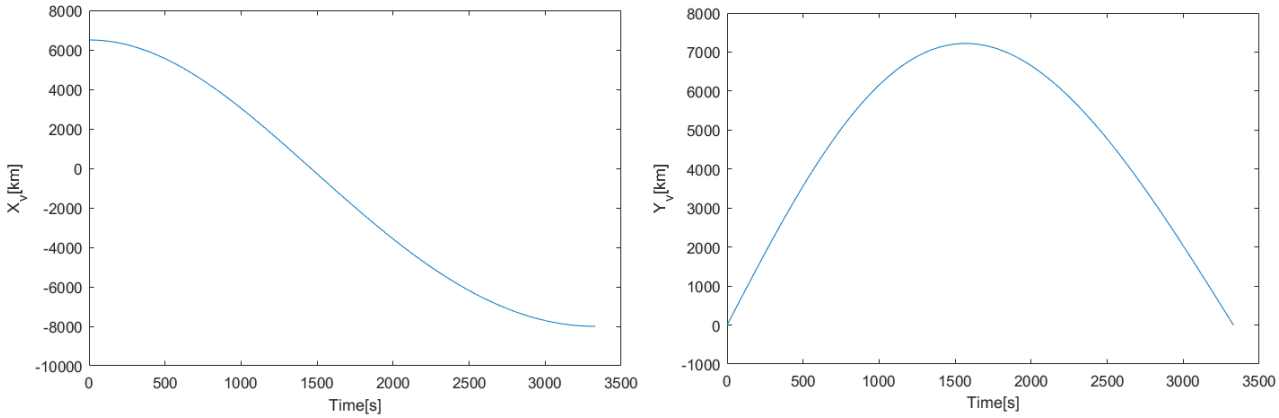


Figure 2.9: X_v vs time and Y_v vs time(Hohmann transfer Case I)

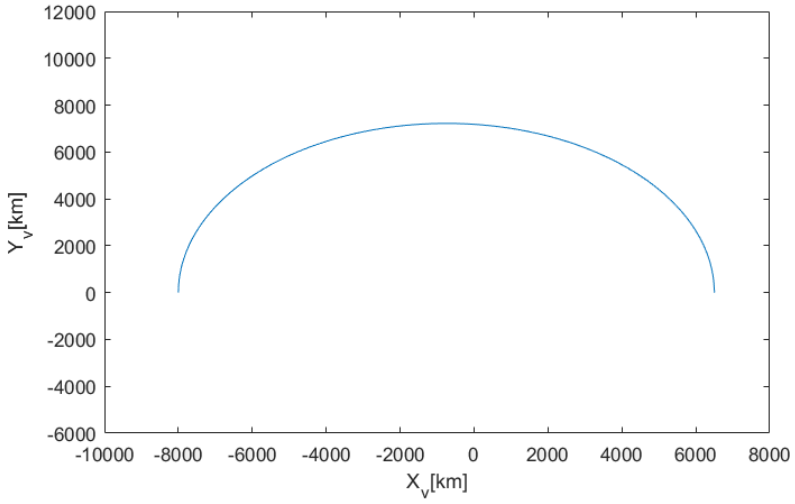


Figure 2.10: Y_v vs X_v(Hohmann transfer Case I)

2.2.3.2 Case II

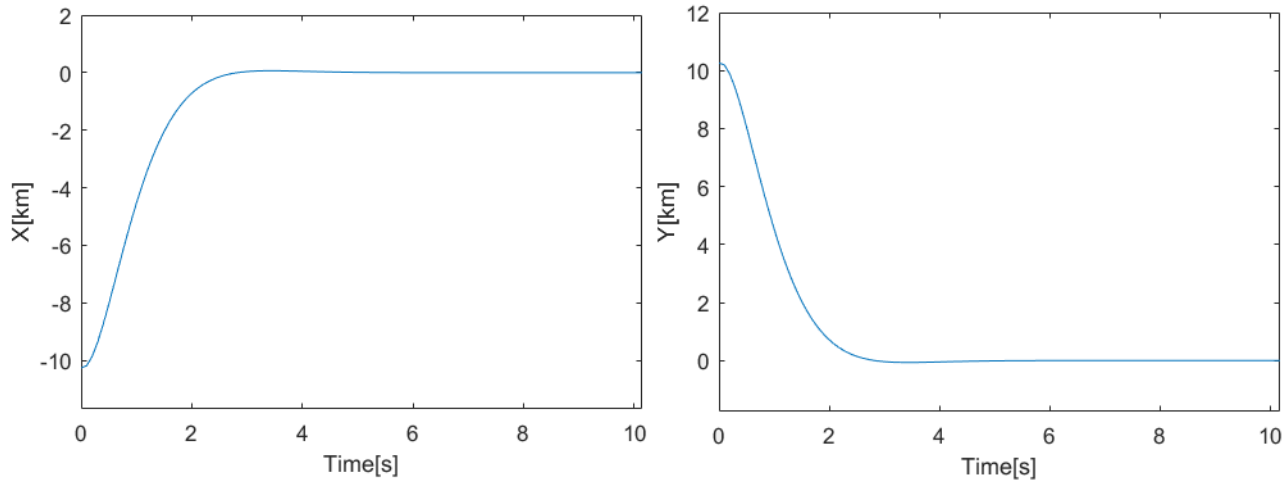


Figure 2.11: x and y vs time(Hohmann transfer Case II)

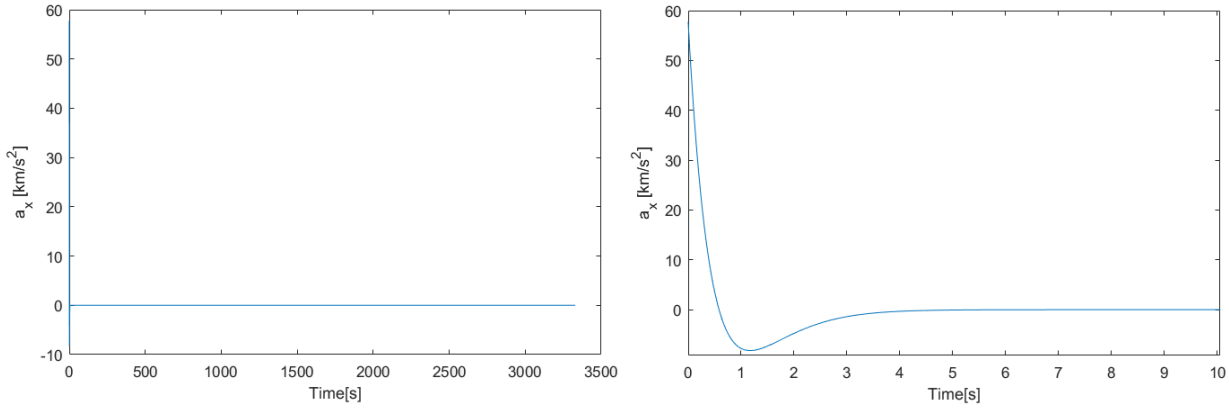


Figure 2.12: a_x vs time(Hohmann transfer Case II)

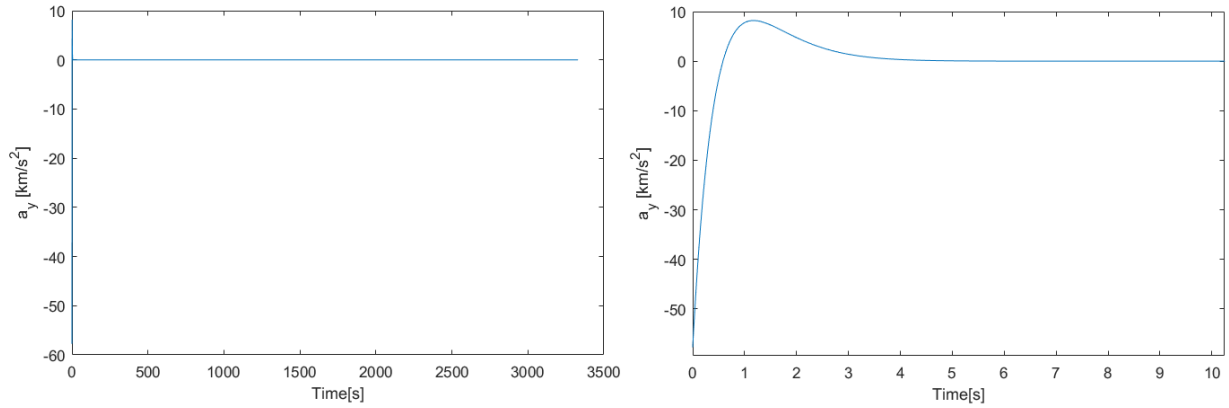


Figure 2.13: a_y vs time(Hohmann transfer Case II)

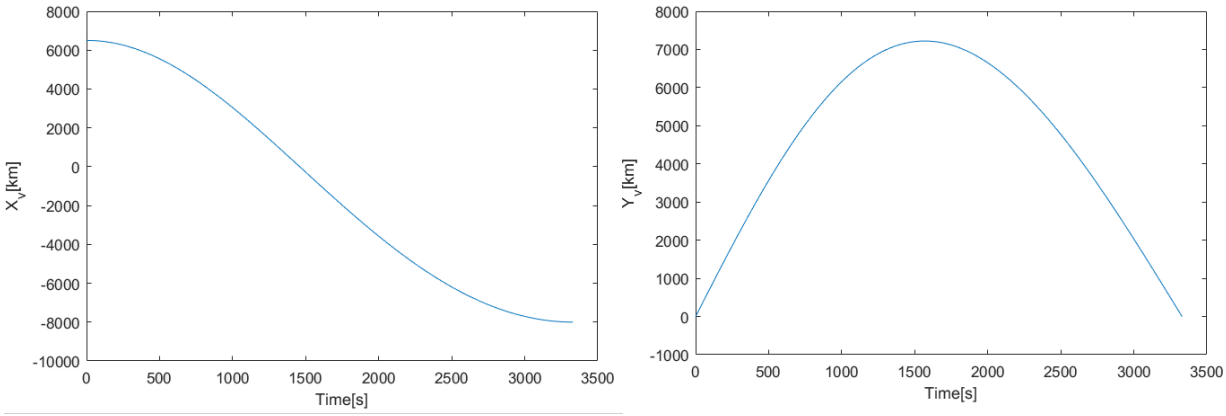


Figure 2.14: X_v vs time and Y_v vs time(Hohmann transfer Case II)

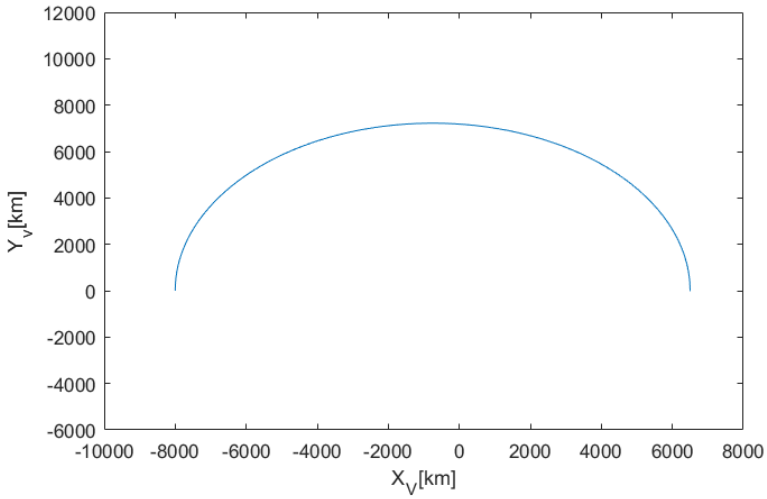


Figure 2.15: Y_v vs X_v(Hohmann transfer Case II)

2.2.4 Discussion of Results

In this subchapter, we will discuss the results obtained from using w different approaches to the same problem of Coplanar Hohmann transfer.
In the first case that we used only the average w(angular velocity), we can see in the figure that compares the accelerations a_x and a_y with the time we can see that the acceleration tended to zero after a few seconds, that was to be expected because the acceleration used in this approach is the relative acceleration(real acceleration minus expected acceleration). From the figures of the distance(x and y) from the theoretic position and real position, we can see that the distance tends to zero in just a few seconds which was to be expected

with the model that was used. From the figures of the position(X_v and Y_v) vs time, we can see that the position of the spacecraft changes smoothly with no abrupt change in the position as time passes which indicates that the model used is working as was expected. In the final figure, we can see the trajectory obtained from the model used which is equal to the one we had from the GMAT program showing that the results obtained are satisfactory.
In the second case in which we used five different values for the angular velocity(w) according to the radius at that point of the trajectory. In the first and second figures of the results about the relative acceleration vs time, we can see that after 5s the relative acceleration stabilizes in zero which was expected because we are talking about the difference between the real acceleration and the predicted acceleration which should equal 0 when the model used is indicated to the problem. From the figures of the distance(X and y) from the theoretic position and real position, we can see that the distance tends to zero in just a few seconds which was to be expected with the model that was used. In the third figure about the position vs time, we can see that there is no sudden spike in the position as time passes and in the last figure we can view the trajectory obtained from the model used in Matlab and conclude that it is equal to the one created in GMAT, therefore, the model used to modulate the orbit transfer trajectory is satisfactory. When comparing the two cases we can see that the results obtained do not vary despite the different approaches in the angular velocity used, that must be the case because the difference of the angular velocities used in case II are so identical that it doesn't affect the overall result.

Chapter 3

Transformation of an optimization of trajectory problem into a non-Linear programming problem

In this chapter we will explain the method used to transform the optimization of trajectory problem into a non-linear programming method, showcasing the formulation of the non-linear programming as well as the transcription method used for it.

3.1 Formulation into a Non-linear programming problem

In order to solve an optimization of trajectory problem we have to transform it into a non-linear programming problem[10],[7],[5],[4]. We can formulate the optimization of the trajectory problem as a problem with N phases, with each phase having its own time interval and the consequent phase time starting at the end time of the previous phase. Within each phase, the problem is defined as a system of dynamic variables. With N_y state variables y and N_u control variables u.
The dynamic of this system is defined by a set of first-order differential equations also known as state equations. With p being a vector of parameters independent of time:

$$\dot{y} = \mathbf{f}[\, y(t), u(t), \mathbf{p}, t\,] \quad (3.1)$$

With the y being the state vector. The dynamic variables are constrained at t_0 as follow:

$$\psi_0 = \psi[\, \mathbf{y}(t_0), \mathbf{u}(t_0), t_0\,] \quad (3.2)$$

And constrained at t_f by the terminal conditions defined by:

$$\psi_f = \psi[\, \mathbf{y}(t_f), \mathbf{u}(t_f), t_f\,] \quad (3.3)$$

In addition, the solution must satisfy the algebraic path constraints:

$$\mathbf{g} = \psi[\, \mathbf{y}(t), \mathbf{u}(t), t\,] \quad (3.4)$$

With **g** being an N_g dimension vector.

The dynamic variables must respect the boundary conditions:

$$\psi_{0l} \leq \psi[\, \mathbf{y}(t_0), \mathbf{u}(t_0), \mathbf{p}, t_0 \,] \leq \psi_{0u} \quad (3.5)$$

$$\psi_{fl} \leq \psi[\, \mathbf{y}(t_f), \mathbf{u}(t_f), \mathbf{p}, t_f \,] \leq \psi_{fu} \quad (3.6)$$

$$g_l \leq \mathbf{g}[\, \mathbf{y}(t), \mathbf{u}(t), \mathbf{p}, t \,] \leq g_u \quad (3.7)$$

As well as in the state and control variables:

$$y_l \leq y_t \leq y_u \quad (3.8)$$

$$u_l \leq u_t \leq u_u \quad (3.9)$$

The performance functions to be minimized have the following form:

$$J(u) = \int_{t_0}^{t_f} \mathbf{q}[\, \mathbf{y}(t), \mathbf{u}(t), \mathbf{p}(t), t \,]\, dt \quad (3.10)$$

The functions above are named as the vector of continuous functions.

The performance index takes the following form as the result of the numerical expression of the integral in eq.3.10

$$J = \phi[y(t_0), t_0, y(t_1), p_1, t_1, ..., y(t_{N-1}), t_{N-1}, p_{N-1}, y(t_N), p_N, t_N \,] \quad (3.11)$$

We can write the set of dynamic variables as such:

$$\mathbf{z} = \begin{bmatrix} \mathbf{y}(t) \\ \mathbf{u}(t) \end{bmatrix} \quad (3.12)$$

Then we will use a transcription method that are used for solving two-point boundary value problems, there are two types of transcription methods, indirect transcription method and direct transcription methods. The main difference between the direct and indirect collocation methods is that the direct method can be applied without explicitly deriving the necessary conditions(adjoint, transversality, maximum principle) while the indirect method requires this derivation to be made to encounter the solution.

3.2 Indirect Transcription

We will start by defining the indirect transcription method which was the one used in this thesis. We start by computing the unknown initial values $v0 = v(t0)$, with v_j for $j = 0, ..., (N-1)$ being the initial value for the dynamic variables at the beginning of each segment, such that the theoretic and real value of the final value of the dynamic variables is the same

$$0 = \phi[\nu(t_f), t_f] \quad (3.13)$$

For some value of $t_f > t_0$

$$\dot{\nu} = f[\nu(t), t] \quad (3.14)$$

Then we use the multiple shooting method that allows us to solve ODE's with an initial guess. We start by defining the NLP variables as such:

$$x = \nu_0, \nu_1, ..., \nu_{N-1} \quad (3.15)$$

Imposing that the segments join at the boundaries.

$$c(x) = \begin{bmatrix} \nu_1 - \bar{\nu}_0 \\ \nu_2 - \bar{\nu}_1 \\ . \\ . \\ . \\ \phi[\nu_N, t_f] \end{bmatrix} \quad (3.16)$$

After the multiple shooting method, we start by imposing the following constraints, with h being the time length of each segment

$$0 = \nu_{j+1} - \bar{\nu}_j = \nu_{j+1} - (\nu_j + h_j f_j) \quad (3.17)$$

With J being $j = 0, ..., N-1$
Then we use the hermite-simpson method for the defect constraint in the collocation method:

$$0 = \nu_{j+1} - \nu_j - (h_{j+1}/6)[f_{j+1} + 4\bar{f}_{j+1} + f_j] = \zeta_j \quad (3.18)$$

also written as:

$$\bar{\nu}_{j+1} = \frac{1}{2}[\nu_j + _{j\ +1}] +)h_{j+1}/8)[f_j - f_{j+1}] \quad (3.19)$$

3.3 Direct Transcription

Another type of transcription method is the direct transcription method which is currently used because it does not require the derivation of the necessary conditions (adjoint, transversality, maximum principle, etc). Another reason for the use of this method is that it does not require a prior specification of the sequence of segments of the trajectory with path inequalities. Both of these reason make this method more suitable for complicated applications.

In the next chapters about the non-Hohmann transfer orbit the non-linear programming methods used to optimize this type of orbit in order to minimize the global energy used with the objective function J will be based on the concepts analyzed in this chapter.

Chapter 4

Non-Hohmann transfer orbit methods

Until now we only talked about the classic method which is the Hohmann transfer in which the vehicle begins in the apoapsis or periapsis of the original orbit and ends in the apoapsis or periapsis of the destiny orbit. With the non-classic method that will be deepened in this chapter, we can perform the orbit transfer starting at any point of the initial orbit and ending at any point of the target orbit, which allows us to save time on the transfer orbit since we don't have to wait to reach the periapsis/apoapsis of the initial orbit and we don't have to make a half revolution (180º) allowing us to choose the target point on the destiny orbit. We will use a direct method approach to this problem in which the trajectory is divided into various segments avoiding the need to derive the optimality variables and allowing the problem to be solved with a nonlinear programming method(NLP). In sections 4.1 and 4.2 we will tackle the non-Hohmann method for a 2D orbit transfer which is a coplanar transfer. In sections 4.3 and 4.4 we will tackle the 3D orbit transfer version, which is the non-coplanar orbit transfer.
To solve this type of Non-Hohmann transfer orbit problem we begin by describing the boundary conditions as so:

$$\begin{cases} t(t_0) = t_0 \\ r(t_0) = r_0; \\ \theta(t_0) = \theta_0; \\ \phi(t_0) = \phi_0 \; v_r(t_0) = v_{r_0;} \\ v_\theta(t_0) = v_{\theta_0;} \\ v_\phi(t_0) = v_{\phi_0;} \\ t(t_f) = t_f \\ r(t_f) = r_f; \\ \theta(t_f) = \theta_{f;} \\ \phi(t_0) = \phi_f \\ v_r(t_f) = v_{r_f;} \\ v_\theta(t_f) = v_{\theta_f;} \\ v_\phi(t_f) = v_{\phi_f;} \end{cases} \tag{4.1}$$

In the equation 4.1 we can see the boundary conditions used in the case of the non-coplanar non-Hohmann transfer orbit, if the transfer orbit happens to be coplanar we will not have the phi and v_ϕ variables.

We will also use the following motion equations as constraint equations:

For the coplanar case:

$$\dot{r} = v_r \tag{4.2a}$$
$$\dot{\theta} = \frac{v_\theta}{r} \tag{4.2b}$$
$$\dot{u} = asin(\beta) + \frac{v_\theta^2}{r} - \frac{\mu}{r^2} \tag{4.2c}$$
$$\dot{v} = acos(\beta) - \frac{v_\theta v_r}{r} \tag{4.2d}$$

For the non-coplanar case:

$$\dot{r} = v_r \tag{4.3a}$$
$$\dot{\theta} = \frac{v_\theta}{rcos(\phi)} \tag{4.3b}$$
$$\dot{\phi} = \frac{v_\phi}{r} \tag{4.3c}$$
$$\dot{v}_r = \frac{v_\theta^2}{r} + \frac{v_\phi^2}{r} - \frac{\mu}{r^2} + acos(\alpha_r) \tag{4.3d}$$
$$\dot{v}_\theta = -\frac{v_r v_\theta}{r} + \frac{v_\theta v_\phi tan(\phi)}{r} + asin(\alpha_r)sin(\alpha_{\phi\theta}) \tag{4.3e}$$
$$\dot{v}_\phi = -\frac{v_r v_\phi}{r} - \frac{v_\theta^2 tan(\phi)}{r} + asin(\alpha_r)cos(\alpha_{\phi\theta}) \tag{4.3f}$$

The objective function utilized will be the one that minimizes the energy used:

$$J = \int_0^{tf} \frac{1}{2} u^2(t) dt \tag{4.4}$$

In order to obtain the results we start firstly by making a guess for the first values of the variables strings. This guess is done by using a linear regression from the variable values at t_0 to the variable values at t_f. This linear regression is done by using the following equations:

$$X(k) = \frac{X_f - X_0}{N} * k + X_0, K \in [1, N] \tag{4.5}$$

With N being the number of nodes used and X being the variable.
We can then use the fmincon tool that allows us to minimize an equation based on some restrictions that we choose. In this case, the equation to be minimized is the total energy used eq.4.4, The restriction equations will be the boundary conditions eq.4.1 because at the beginning of the transfer orbit and at the end of the transfer orbit the values of the variables need to be equal to the ones obtain from GMAT. The other restriction will be

the equation of motions eq.4.2a of eq.4.3a conforming the orbit transfer is coplanar or non-coplanar, these restrictions are to restrict the values of the variables between the initial point and final point of the transfer orbit, because the transfer orbit has to abide by the equations of motion. After this, we will be able to obtain the results as shown in the following subchapters.

4.1 Non-Hohmann coplanar orbit transfer design

4.1.1 Orbit dynamics Equations

We started by choosing the initial and final orbit in GMAT and then choosing the departure point and arrival point. In figure 4.1 we can see the design of the orbits including the transfer orbit and in the table4.1 we can see further information about these orbits.

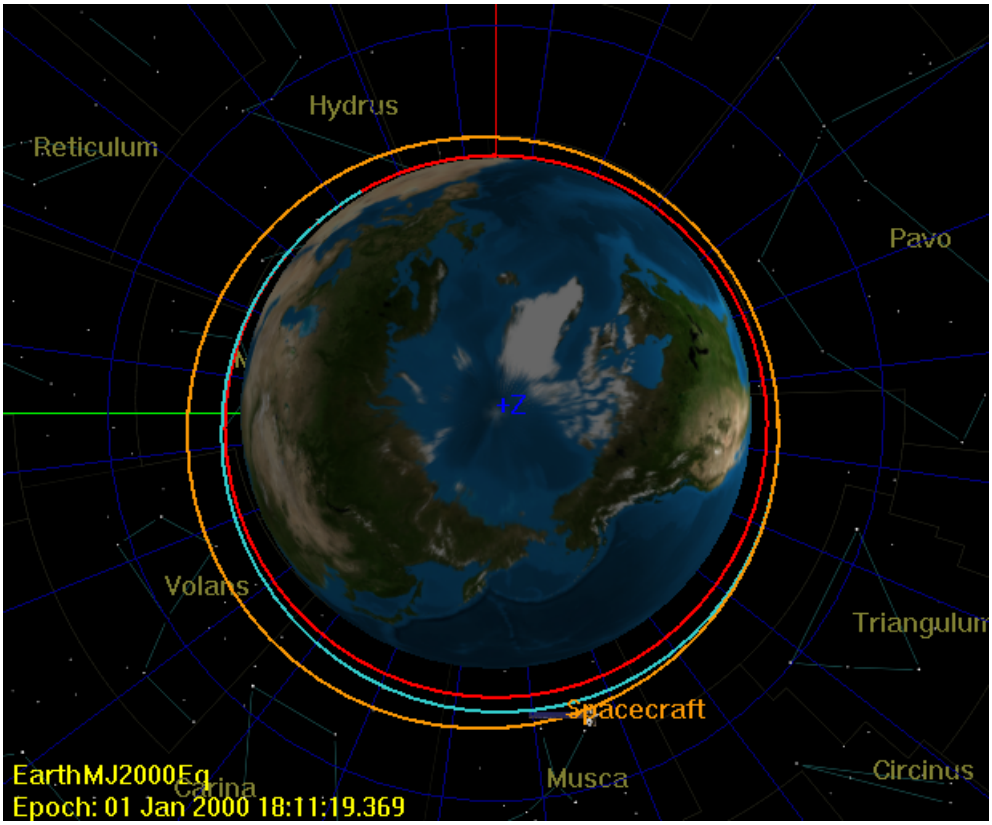
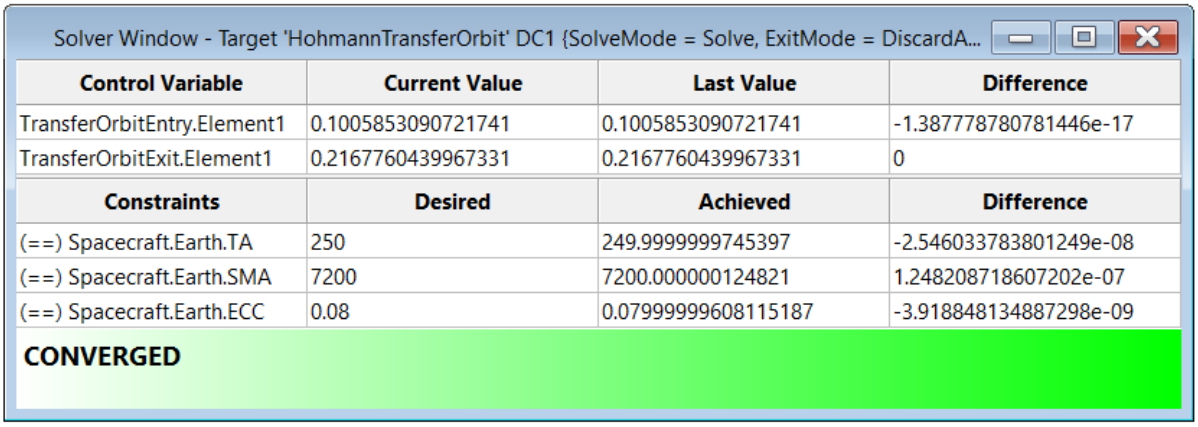


Figure 4.1: Non-Hohmann coplanar transfer in GMAT

	Initial orbit	Final orbit
a[km]	7000	7200
Eccentricity	0.05	0.08
Inclination	0	0

Table 4.1: Orbits information.

The Transfer orbit begins in the 1st orbit at a radius of 6694km and a true anomaly of the 1st orbit off 30°, and ends at the second orbit at a radius of 7344,5km and a true anomaly of the second orbit of 250°



Control Variable	Current Value	Last Value	Difference
TransferOrbitEntry.Element1	0.1005853090721741	0.1005853090721741	-1.387778780781446e-17
TransferOrbitExit.Element1	0.2167760439967331	0.2167760439967331	0
Constraints	**Desired**	**Achieved**	**Difference**
(==) Spacecraft.Earth.TA	250	249.9999999745397	-2.546033783801249e-08
(==) Spacecraft.Earth.SMA	7200	7200.000000124821	1.248208718607202e-07
(==) Spacecraft.Earth.ECC	0.08	0.07999999608115187	-3.918848134887298e-09

CONVERGED

Figure 4.2: GMAT Solver

As shown in the fig.4.2 first velocity impulse being $\Delta v_1 = 0.10059 ms^{-1}$ and the second velocity impulse being $\Delta v_2 = 0.21678 ms^{-1}$ and therefore a total velocity impulse of $\Delta v = 0.31737 ms^{-1}$.

4.2 Computational procedure

We start by defining the Equations of motion that can be shown in eq.4.6, the state variables were the radius r, θ the true anomaly, u the radial velocity and v the transverse velocity and the control variables were a being the acceleration and β being the control acceleration direction. In fig.4.3 we can see the representation of the polar coordinates orbit transfer.

$$\dot{r} = v_r \tag{4.6a}$$

$$\dot{\theta} = \frac{v_\theta}{r} \tag{4.6b}$$

$$\dot{u} = asin(\beta) + \frac{v_\theta^2}{r} - \frac{\mu}{r^2} \tag{4.6c}$$

$$\dot{v} = acos(\beta) - \frac{v_\theta v_r}{r} \tag{4.6d}$$

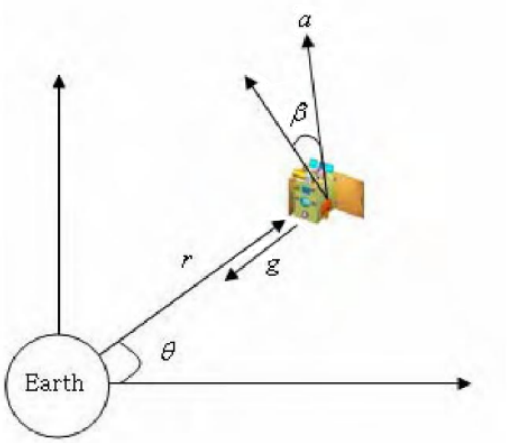


Figure 4.3: 2D polar coordinates orbit transfer
[2]

We used the Matlab tool "fmincon" which allows us to find the minimum of a function(the objective/cost function) with nonlinear constraint functions and an initial guess. The objective function chosen is shown in eq.4.8 that was chosen to minimize the overall energy used in the transfer, the restrictions used were the equations of motion eq.4.6 and the initial and final coordinates shown in eq.4.7, for the first estimate we used a linear progression between the initial and final values for the polar coordinates and the velocities, for the control variables a and β we used the value 0.05.

$$\begin{cases} t_0 = 0s \\ r(t_0) = 6834.343977km; \\ v_r(t_0) = 0.394580374km/s; \\ v_\theta(t_0) = 8.590995868km/s; \\ \theta(t_0) = 0.215551283; \\ t_f = 2084.841046s \\ r(t_f) = 10000km; \\ v_r(t_f) = 1.588877143km/s; \\ v_\theta(t_f) = 5.871450397km/s; \\ \theta(t_f) = 2.094395103; \end{cases} \tag{4.7}$$

The objective/cost function utilised was:

$$J = \int_0^{tf} \frac{1}{2} u^2(t) dt \tag{4.8}$$

4.2.1 Results

The results obtained for the state coordinates were:

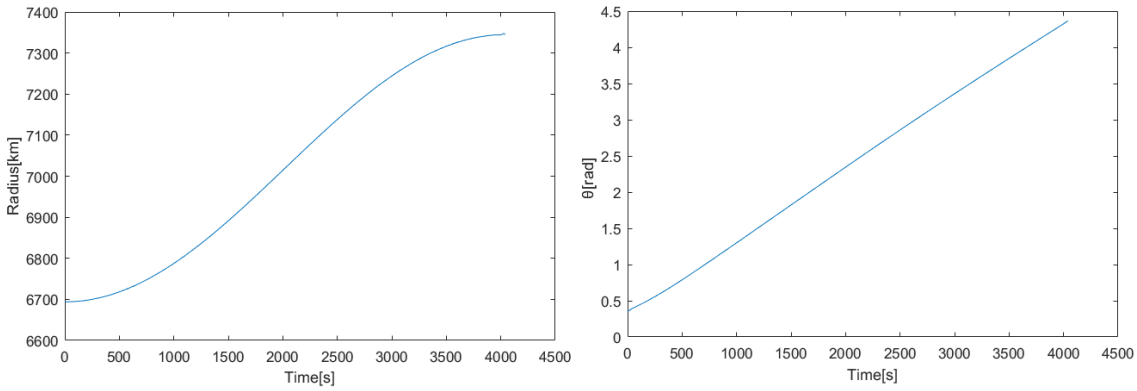


Figure 4.4: Radius and θ vs time(Non-Hohmann transfer)

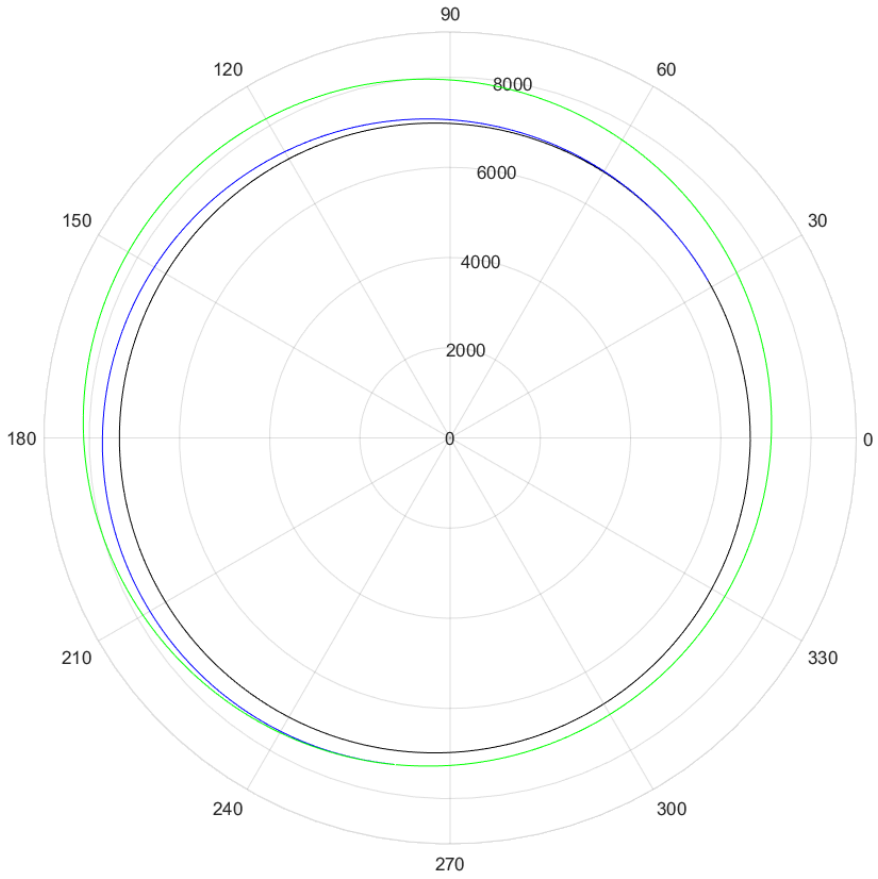


Figure 4.5: Radius vs θ(Non-Hohmann transfer)

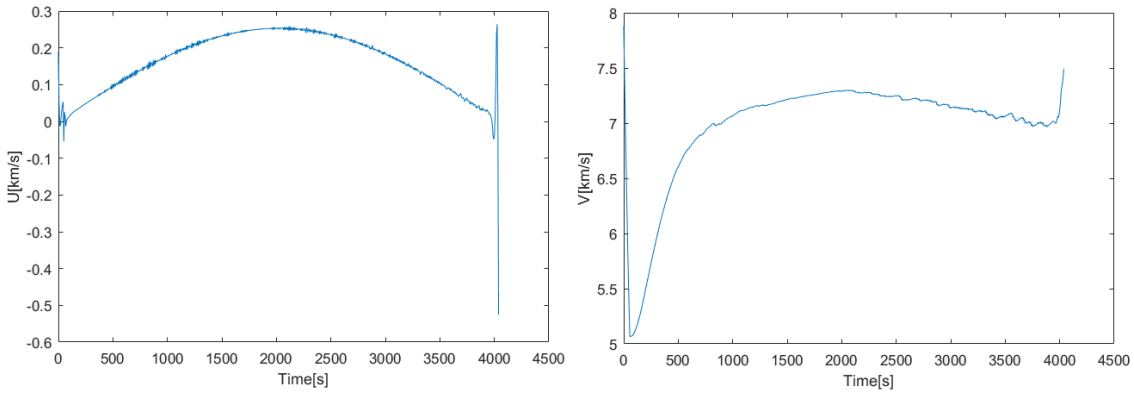


Figure 4.6: U(v_r)$and V(v_\theta) vs time(Non - Hohmann transfer)$

And for the control variables the results obtained were:

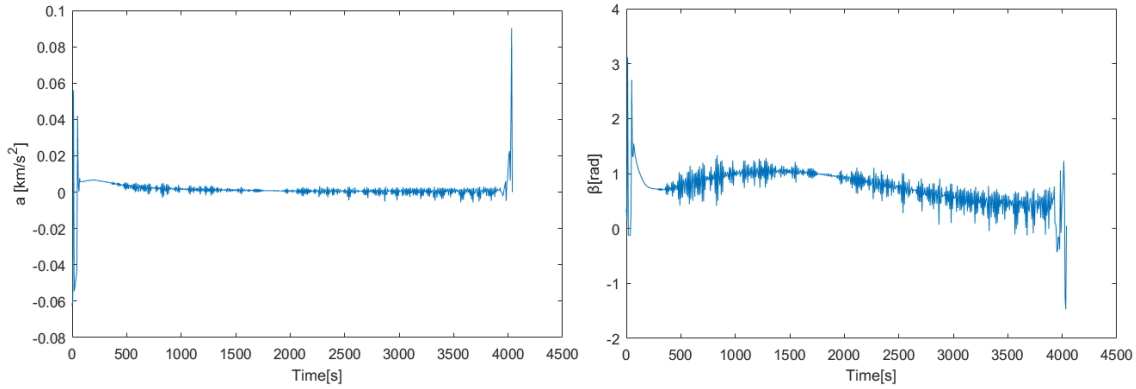


Figure 4.7: α and β vs time(Non-Hohmann transfer)

4.2.2 Discussion of Results

In this subchapter, we will discuss the results of the non-Hohmann coplanar orbit transfer that were obtained using a direct method which allowed us to solve the problem with an NLP (non-linear programming method) in Matlab.
As we can see from the table4.2 the number of iterations needed to obtain the local minimum that satisfies the constraints was 278 and the value of the objective function was 0.12734. From the first figure we can see that the radius stays basically the same at the beginning of the transfer and at the end, this is to be expected because the orbit transfer trajectory must have the beginning section and the ending section adjacent to the origin and target orbit respectively, in the radius graphic we can see that it has an almost linear relation with the time which is to expect because the orbit transfer is almost circular as we can see in the next figure. In the second figure, we can see that the orbit transfer trajectory obtained from the model used in Matlab is exactly the same one that was expected based on the GMAT orbit design. In the third figure about the polar velocities vs time we can see that the velocity spikes abruptly at the beginning and end of the orbit transfer while changing smoothly during the rest of the orbit transfer, this is to be expected because at the beginning and end of the orbit transfer there is the need to transition from the initial orbit and to the final orbit. In the final figure about the control variables, acceleration, and the control acceleration direction the results are pretty similar to the ones obtained in the velocity graphics. We have a big spike in both of the control variables at the beginning and end of the transfer orbit because of the transition from the initial orbit to the transfer orbit and from the transfer orbit to the final orbit, while in the meantime the oscillation in the control variables in much smaller, especially in the acceleration.

Constraint tolerance	$1e^{-5}$
Optimality tolerance	$1e^{-3}$
Number of Iterations	278
Objective function value	0.12734

Table 4.2: Matlab Fmincon Output

4.3 Non-coplanar and non-Hohmann orbit transfer between elliptical orbits

We started by choosing the initial and final orbit in GMAT and then choosing the departure point and arrival point. In Fig.4.8 we can see the design of the orbits including the transfer orbit and in the table 4.3 we can see further information about these orbits.

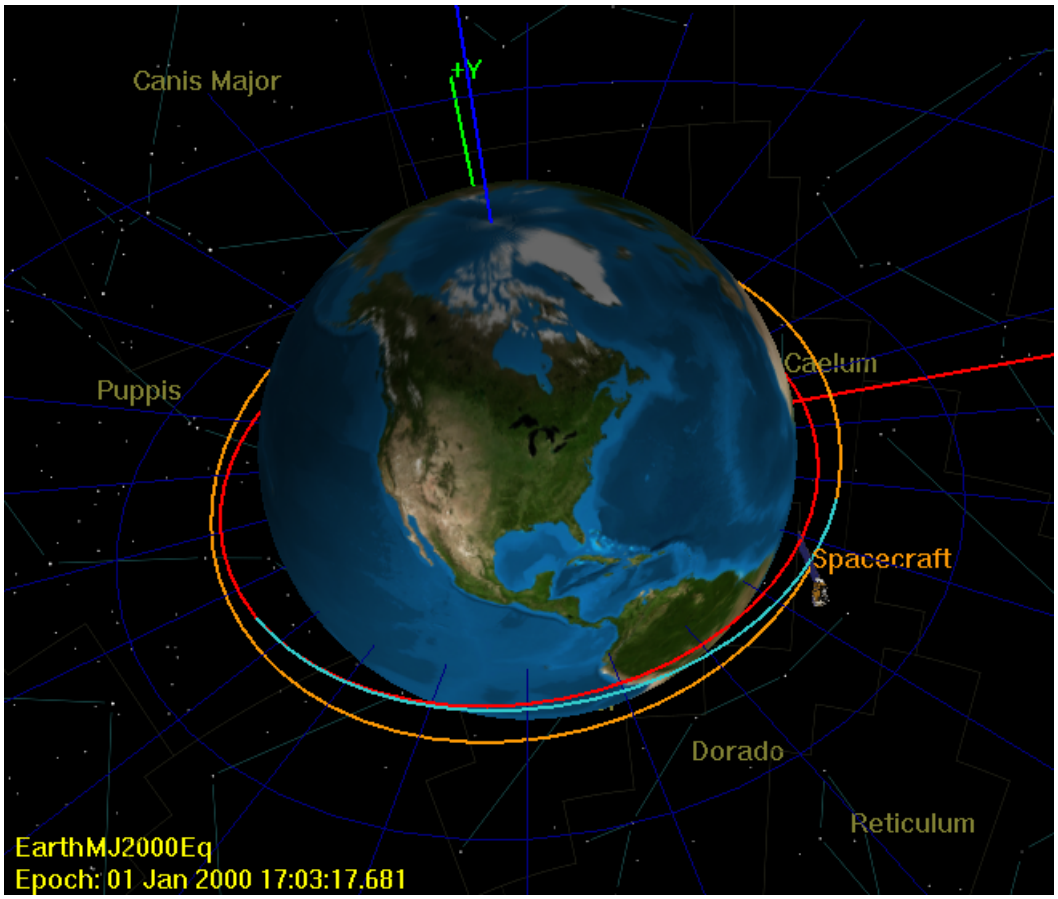
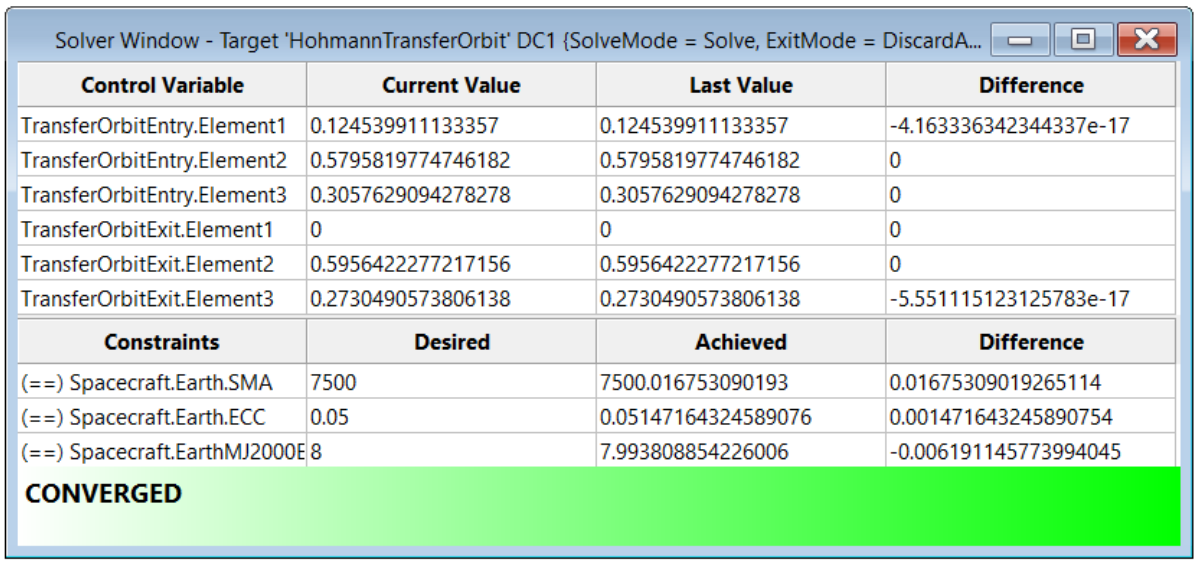


Figure 4.8: Non-Hohmann non-coplanar transfer in GMAT

	Initial orbit	Final orbit
a[km]	7200	7500
Eccentricity	0.05	0.05
Inclination[°]	0	8

Table 4.3: Orbits information.

The Transfer orbit begins in the first orbit at a radius of 7490,77km and a true anomaly of the first orbit off 210° and ends at the second orbit at a radius of 7430,71km and a true anomaly of the second orbit of 15°



Control Variable	Current Value	Last Value	Difference
TransferOrbitEntry.Element1	0.124539911133357	0.124539911133357	-4.163336342344337e-17
TransferOrbitEntry.Element2	0.5795819774746182	0.5795819774746182	0
TransferOrbitEntry.Element3	0.3057629094278278	0.3057629094278278	0
TransferOrbitExit.Element1	0	0	0
TransferOrbitExit.Element2	0.5956422277217156	0.5956422277217156	0
TransferOrbitExit.Element3	0.2730490573806138	0.2730490573806138	-5.551115123125783e-17

Constraints	Desired	Achieved	Difference
(==) Spacecraft.Earth.SMA	7500	7500.016753090193	0.01675309019265114
(==) Spacecraft.Earth.ECC	0.05	0.05147164324589076	0.001471643245890754
(==) Spacecraft.EarthMJ2000E	8	7.993808854226006	-0.006191145773994045

CONVERGED

Figure 4.9: GMAT Solver

As shown in the fig.4.9 first velocity impulse being $\Delta v_1 = 0.66702 ms^{-1}$ and the second velocity impulse being $\Delta v_2 = 0.655244 ms^{-1}$ and therefore a total velocity impulse of $\Delta v = 1.32226 ms^{-1}$.

4.4 Implementation of the non-Hohmann non-coplanar orbit transfer in Matlab

We start by defining the Equations of motion that can be shown in eq.4.9. The state variables are the r, θ, and ϕ that are the spherical coordinates, and the v_r, v_θ and v_ϕ that are the spherical components of the velocity. The control variables are the acceleration a and the thrust direction angles α_r and $\alpha_{\phi\theta}$.In Fig.4.10 we can see the representation of the spherical coordinates orbit transfer.

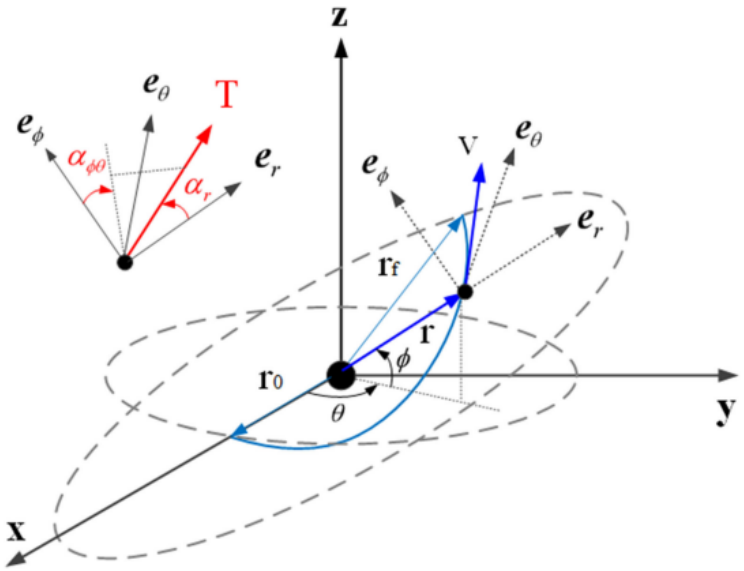


Figure 4.10: 3D spherical coordinates orbit transfer
[11]

$$\dot{r} = v_r \tag{4.9a}$$

$$\dot{\theta} = \frac{v_\theta}{rcos(\phi)} \tag{4.9b}$$

$$\dot{\phi} = \frac{v_\phi}{r} \tag{4.9c}$$

$$\dot{v}_r = \frac{v_\theta^2}{r} + \frac{v_\phi^2}{r} - \frac{\mu}{r^2} + acos(\alpha_r) \tag{4.9d}$$

$$\dot{v}_\theta = -\frac{v_r v_\theta}{r} + \frac{v_\theta v_\phi tan(\phi)}{r} + asin(\alpha_r)sin(\alpha_{\phi\theta}) \tag{4.9e}$$

$$\dot{v}_\phi = -\frac{v_r v_\phi}{r} - \frac{v_\theta^2 tan(\phi)}{r} + asin(\alpha_r)cos(\alpha_{\phi\theta}) \tag{4.9f}$$

Just as in the 4.2, we used the Matlab tool "fmincon" which allows us to find the minimum of a function(the objective/cost function) with nonlinear constraint functions and an initial guess. The objective function chosen is shown in eq.4.11 that was chosen to minimize the overall energy used in the transfer, the restrictions used were the equations of motion eq.**??** and the initial and final coordinates shown in eq.4.10, for the first estimate we used a linear progression between the initial and final values for the polar coordinates and the velocities, for the control variables a, α_r and $\alpha_{\phi\theta}$ we used the value 0.05.

$$\begin{cases} t_0 = 0s \\ r(t_0) = 7490.7766km; \\ \theta(t_0) = -2.61799; \\ \phi(t_0) = 0 \\ v_r(t_0) = -0.181050226km/s; \\ v_\theta(t_0) = 7.296693388km/s; \\ v_\phi(t_0) = 6.87622E-5km/s; \\ t_f = 2123.153752s \\ r(t_f) = 7393.87066km; \\ \theta(t_f) = 2,65103; \\ \phi(t_f) = 0.139626; \\ v_r(t_f) = 0.093341349km/s; \\ v_\theta(t_f) = 7.483739058km/s; \\ v_\phi(t_f) = 0.568300774km/s; \end{cases} \tag{4.10}$$

The objective/cost function utilised was:

$$J = \int_0^{tf} \frac{1}{2} u^2(t) dt \tag{4.11}$$

4.4.1 Results

For the state variables the results obtained were

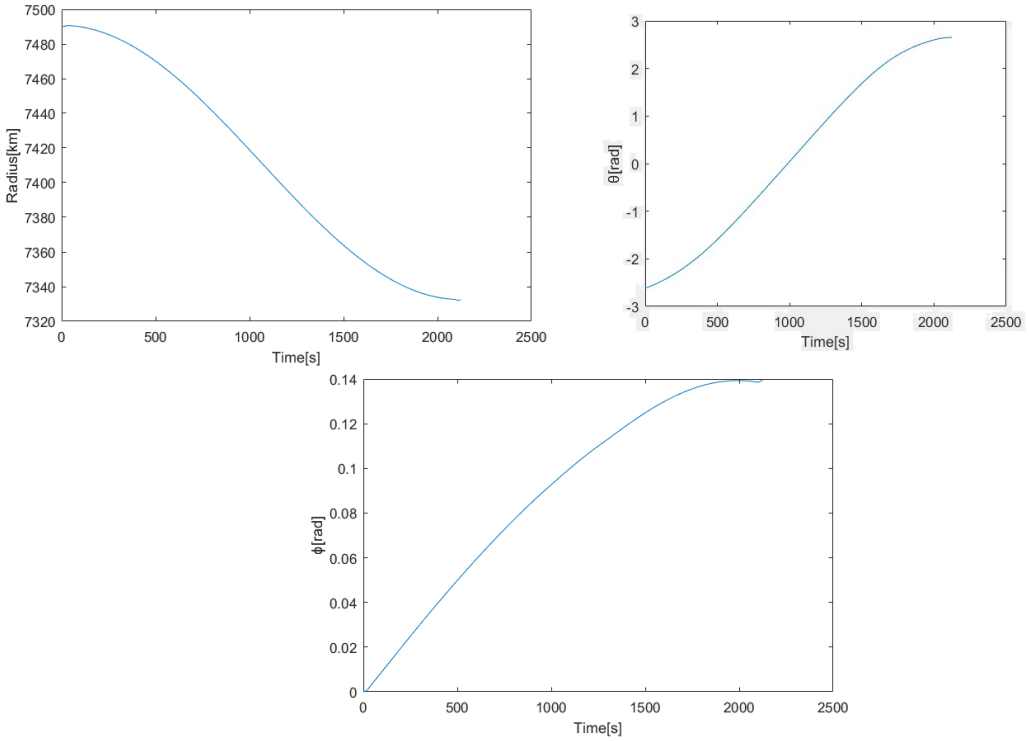


Figure 4.11: Radius, θ and ϕ vs time(Non-Hohmann non-coplanar orbit transfer)

For the control variables the results obtained were

4.4.2 Discussion of Results

In this subchapter, we will discuss the results of the non-Hohmann non-coplanar orbital transfer obtained by using a direct method which allowed us to solve the problem with an NLP (non-linear programming) method in Matlab.
As we can see from the table4.4 the number of iterations needed to obtain the local minimum that satisfies the constraints was 146 and the value of the objective function was 0.049323.
From the first figure, we can see that the radius as has happened in the co-planar non-Hohmann orbit transfer stays basically the same at the beginning and end of the orbit transfer in order for the orbit transfer to be adjacent to the initial orbit and final orbit at the respective points. We can also see that the theta varies linearly with time as expected, in the /phi graph we can see that it increased faster in the beginning and then started to slow down until reaching the final inclination of 0.14rad, we see that the final section of the transfer the inclination graph as a little bump this must be due to a low number of nodes used in the iteration. In the second figure, we can see the 3D Trajectory of the

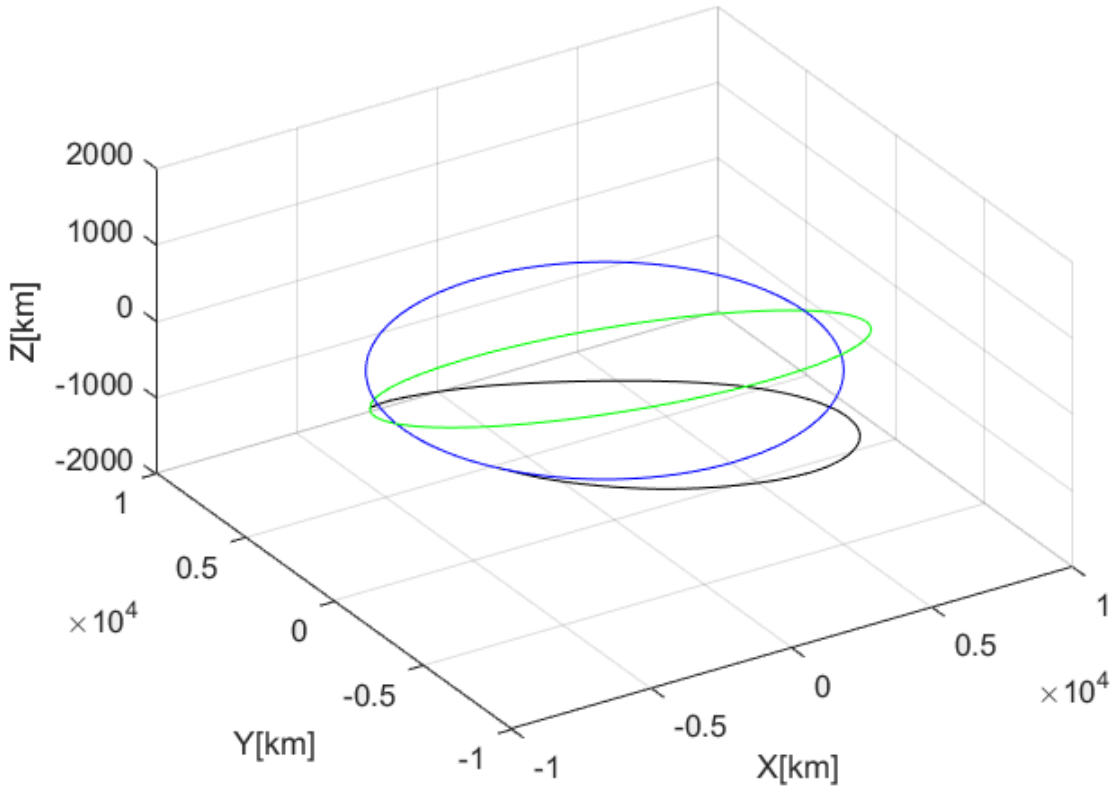


Figure 4.12: 3D representations of the Non-Hohmann non-coplanar orbit transfer

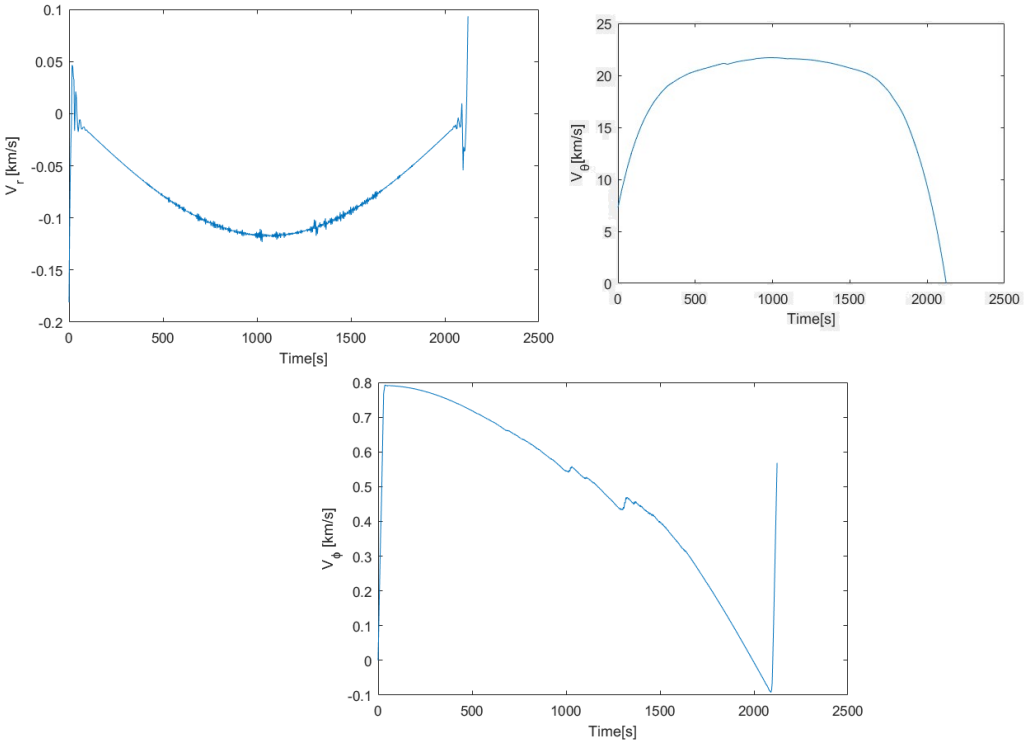


Figure 4.13: v_r,v_θ and v_ϕ vs time(Non-Hohmann non-coplanar orbit transfer)

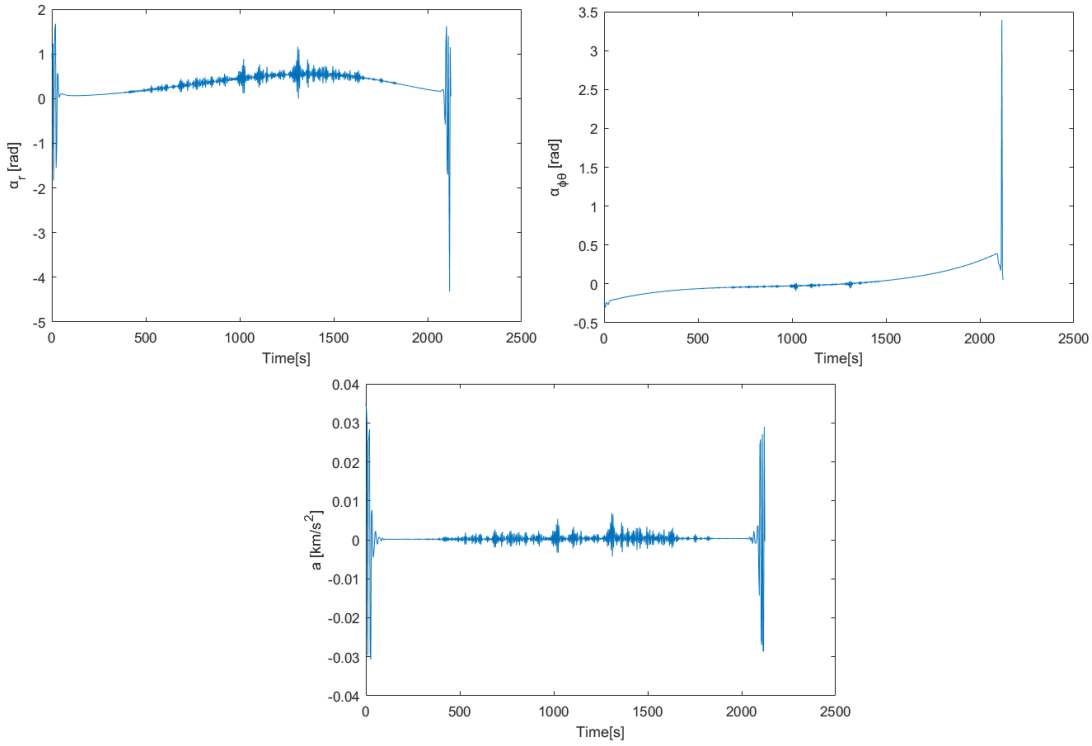


Figure 4.14: a, α_r and $\alpha_{\phi\theta}$ vs time(Non-Hohmann non-coplanar orbit transfer)

orbit transfer obtained from the Matlab implementation and verify it is similar to the one obtained from GMAT. In the third figure of velocities vs time, we can see that the velocity values vary very smoothly with time except at the beginning of the orbit transfer and at the end of the orbit transfer which is to be expected because of the change of orbits. In the final figure about the control variables, acceleration, and the thrust direction angles the results are pretty similar to the ones obtained in the velocity graphics. We have a big spike in both of the control variables at the beginning and end of the transfer orbit because of the transition from the initial orbit to the transfer orbit and from the transfer orbit to the final orbit, while in the meantime the oscillation in the values of the control variables is much smaller.

Constranit tolerance	$1e^{-5}$
Optimality tolerenace	$1e^{-3}$
Number of Iterations	146
Objective function value	0.049323

Table 4.4: Matlab Fmincon Output

Chapter 5

Conclusions and Future Work

The objectives of the dissertation are aimed at dealing with methods and models for optimal orbit transfer for LEO satelites. The main objective revolves around the non-Hohmann transfers because the Hohmann transfer has some limitations on its use such as its inability to start and end the transfer orbit at any point of the original and final orbit respectively since it has to start and end at the periapsis/apoapsis. The approach used to combat the limitations of the transfer method is the transformation of the transfer orbit into an optimization problem. In this paper, we studied 3 examples, the coplanar Hohmann orbit transfer, the coplanar non-Hohmann orbit transfer, and the non-coplanar non-Hohmann orbit transfer. In the coplanar Hohmann orbit transfer, we used an approach of relative dynamic equations based on the Clohessy–Wiltshire equations in which the state variables used are the relative position which is the position difference between the real trajectory and the projected trajectory, and the control variables are the acceleration Cartesian components. We can see from the results obtained that the projected trajectory and the real trajectory quickly converge showing that this approach to the coplanar Hohmann orbit transfer problem is well-adjusted and creates the results that would be expected in an orbit controller. To solve the non-Hohmann orbital transfer problem we will use the direct method which consists of dividing the trajectory into several segments avoiding the need to derive the optimization variables allowing the problem to be solved with the nonlinear programming method (NLP). In the coplanar and non-coplanar non-Hohmann orbit transfers the approach used was a direct method in which the trajectory is divided in various segments avoiding the need to derive the optimality variables and allowing the problem to be solved with a nonlinear programming method(NLP) for which we used fmincon in Matlab. For the coplanar case, the state variables were the polar coordinates of position and velocity and the control variables were the acceleration and the thrust angle. For the non-coplanar case, the state variables were the spherical coordinates of position and velocity, and the control variables were the acceleration and the thrust direction angles alpha r and $\alpha_{\phi\theta}$. We can see from the results that the trajectory obtained from the optimization is what we expected, however, it could have been more refined if more nodes were added to the trajectory in the beginning and in the final part of the trajectory where the acceleration graphics has its spikes, we can see that the overall energy used is low which shows that this is a good approach to the problem, we can also see that this approach allows us to find the minimum overall energy necessary in to execute the orbit transfer.

In the future, it would be interesting to increase the knowledge of other direct methods as well as the indirect methods which are based on Pontryagin's minimum principle which requires the derivation of the adjoint and transversality equations as well as the minimization of the Hamiltonian in respect to the control, and afterward compare both methods in the different case to see which one is more suitable and delivers the best results. It would also be interesting to study other types of orbits such as the Geostationary Earth Orbits.

Bibliography

[1] Alqarni A.A. Mathematical Procedures for the Non-Coplanar Tangential Transfers between Circular Orbits. *Pen Access Library Journal, 8: e7101*, 2021. doi: https://doi.org/10.4236/oalib.1107101. xiii, 3

[2] Lee D.h. Cho D.h. and Tahk M.j. Orbit Transfer Trajectory Optimization With Electric Engine. *2006 SICE-ICASE International Joint Conference*, pages 4093–4098, 2006. doi: https://doi.org/10.1109/SICE.2006.315122. 49

[3] Eagles D. The Hohmann Orbit Transfer. MATLAB Central File Exchange. Retrieved May 8, 2023, https://www.mathworks.com/matlabcentral/fileexchange/38942-the-hohmann-orbit-transfer. xiii, 1

[4] Kirk D.E. *Optimal Control Theory (An Introduction)*. Dover publications edition, 2004. 41

[5] Hull D.G. Conversion of Optimal Control Problems into Parameter Optimization Problems. *Journal of Guidance, Control and Dynamics*, Vol. 20, No. 1, 1997. doi: https://doi.org/10.2514/2.4033. 41

[6] Curtis H. *Orbital Mechanics for Engineering Students*. Elsevier,first edition, 2004. ISBN: 9780080470542. xiii, 5, 8

[7] Betts J.T. Survey of Numerical Methods for Trajectory Optimization. *Journal of Guidance, Control and Dynamics*, Vol. 21, No. 2, 1998. doi: https://doi.org/10.2514/2.4231. 41

[8] Bayern A.M. Kong Q., Siauw T. *Python Programming and Numerical Methods*. Academic press 1st edition, 2021. ISBN: 9780128195499. 23

[9] Hibey J.L. Naidu D.S. and Charalambous C.D. Fuel-optimal trajectories for aeroassisted coplanar orbital transfer problem. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 2, pages 374-381, 1990. doi: https://doi.org/10.1109/7.53464. xiii, 11, 12

[10] Lozano J.A. Shirazi A., Ceberio J. Spacecraft Trajectory Optimization: A review of Models, Objectives, Approaches and Solutions. *Progress in Aerospace Sciences*, Volume 102, 2018, Pages 76-98, ISSN 0376-0421. doi: https://doi.org/10.1016/j.paerosci.2018.07.007. 41

[11] Z. Wang and M. J. Grant. Minimum-Fuel Low-Thrust Transfers for Spacecraft: A Convex Approach. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54 no. 5, pages 2274-2290, Oct. 2018. doi:https://doi.org/10.1109/TAES.2018.2812558. 55

[12] Santos W.G., Rocco E.M., and Carrara V. Trajectory Control During an Aeroassisted Maneuver Between Coplanar Circular Orbits. *Journal of Aerospace Technology and Management 6*, pages 159–168, 2014. doi: https://doi.org/10.5018/jatm.v6i2.351. xiii, 13, 14