

Relatório 4 - Manobras Orbitais (AGA0521)

Kethelin Parra Ramos - 9898349

I. INTRODUÇÃO

As transferências de Hohmann são manobras eficientes para mover, por exemplo, uma nave entre duas órbitas diferentes. A Fig.1 ilustra um exemplo da transferência de Hohmann simples.

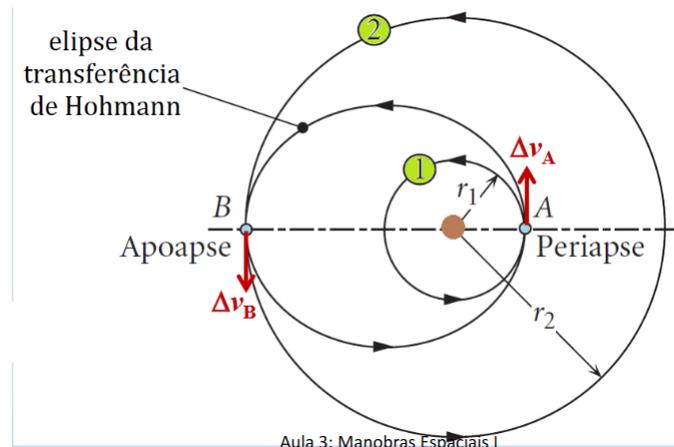


Figura 1: Exemplo de uma transferência de Hohmann simples. Fonte: slides de aula.

II. MÉTODOS

Para resolver os problemas propostos desta semana foi desenvolvido um único programa em Python. O código foi “quebrado” em algumas partes para que todos os passos das atividades fossem descritos de forma satisfatória.

Na atividade desta semana foi pedido a elaboração de um programa que calculasse os procedimentos da transferência de Hohmann simples, testando o exemplo de aula e um novo caso. Para isto, foi criada a função *HohmannSimples()* que seguia os passos mostrados em aula (código abaixo). Os dados utilizados estão listados na Tab.I.

```

1  #Bibliotecas
2  import numpy as np
3
4  """                                     Funções                                     """
5
6  #Função que recebe as informações das duas orbitas da transferência
7  def HohmannSimples(A_perigeu,B_apoapse,C_apogeu):
8      #Alguns dados
9      R_terra = 6378          #Raio da Terra (km)
10     M_terra = 5.9736E24     #Massa Terra (kg)
11     G = 6.67384e-11        #Cte gravitacional m³kg-1s-2
12     mu = G*M_terra*1E-9    #Parâmetro gravitacional km³s-2 G(m1+m2) ~ Gm1 para m1>>>m2
13
14     print("\nResultados intermediários:")
15     #Calculando parâmetros da orbita 1
16     rA = R_terra + A_perigeu # (km)
17     rC = R_terra + C_apogeu # (km)
18     e1 = (rC-rA)/(rC+rA)     #Excentricidade
19     h1 = np.sqrt(rA*mu*(1+e1*np.cos(0)))
20
21     #Calculando a velocidade no perigeu da orbita 1
22     if e1 == 0:
23         vA1 = np.sqrt(mu/rA)
24     else:
25         vA1 = h1/rA
26     print('rA = {:.f} km; rC = {:.f} km; e1 = {:.f}; h1 = {:.f} km2/s; vA1 = {:.f} km/s'
27           .format(rA,rC,e1,h1,vA1))
28     #Cálculo dos parâmetros orbitais da órbita 2 (elipse)
29     rB = R_terra + B_apoapse
30     e2 = (rB-rA)/(rB+rA)     #Excentricidade
31     h2 = np.sqrt(rA*mu*(1+e2*np.cos(0)))
32
33     #Calculando a velocidade no perigeu da orbita 2
34     vA2 = h2/rA
35     print('rB = {:.f} km; e2 = {:.f}; h2 = {:.f} km2/s; vA2 = {:.f} km/s'
36           .format(rB,e2,h2,vA2))
37     deltavA = vA2 - vA1 #A diferença de velocidade entre as órbitas 1 e 2 (no perigeu)

```

```

38
39     #Calculando a velocidade no ponto B na órbita 2
40     vB2 = h2/rB
41
42     #Como a órbita 3 é circular, a velocidade no ponto B
43     vB3 = np.sqrt(mu/rB)
44     print('vB2 = {:.f} km/s; vB3 = {:.f} km/s'.format(vB2,vB3))
45     deltavB = vB3-vB2 #A diferença de velocidade entre as órbitas 3 e 2 no ponto B
46
47     #Delta V total
48     deltaVtotal = np.fabs(deltavA)+np.fabs(deltavB)
49
50     return deltavA,deltavB,deltaVtotal

```

Tabela I: Dados dos pontos orbitais utilizados nas questões dessa semana.

Questão	A (km)	B (km)	C (km)
4a	480	16000	800
4b	300	3000	300

III. RESULTADOS

Como solicitado no enunciado, os resultados dos cálculos realizados ao longo de todo o código estão listados na Tab.II. O código das duas atividades se encontra abaixo:

```

1     """                               Programa Principal                               """
2
3     #Atividade 4a - Exemplo da aula 3
4     A_perigeu = 480     #Ponto A (km)
5     B_apoapse = 16000  #Ponto B (km)
6     C_apogeu = 800    #Ponto C (km)
7
8     #Calculando as velocidades
9     item_a,item_b,dVtotal = HohmannSimples(A_perigeu, B_apoapse, C_apogeu)
10

```

```

11 print("\nResultado final Atividade 4a:")
12 print('a) \Deltav_A = {:.f} km/s'.format(item_a))
13 print('b) \Deltav_B = {:.f} km/s'.format(item_b))
14 print('\Deltav = {:.f} km/s'.format(dVtotal))
15
16 #Atividade 4b
17 A_perigeu = 300      #Ponto A (km)
18 B_apoapse = 3000    #Ponto B (km)
19 C_apogeu = 300      #Ponto C (km)
20
21 #Calculando as velocidades
22 dvA,dvB,dVtotal = HohmannSimples(A_perigeu, B_apoapse, C_apogeu)
23
24 #Calculando o tempo
25 R_terra = 6378      #Raio da Terra (km)
26 M_terra = 5.9736E24 #Massa Terra (kg)
27 G = 6.67384e-11*1E-9 #Cte gravitacional km³kg-1s-2
28 R_ell = (C_apogeu+B_apoapse)/2 #Media simples
29 periodo2 = np.sqrt((4*(np.pi**2)*(R_ell+R_terra)**3)/(G*M_terra))
30 tempo = periodo2/2 #Meia orbita
31 horas = tempo/3600 #Em horas
32
33 print("\nResultado final Atividade 4b:")
34 print('\Deltav_A = {:.f} km/s; \Deltav_B = {:.f} km/s'.format(dvA,dvB))
35 print('a) \Deltav = {:.f} km/s'.format(dVtotal))
36 print('b) tempo = {:.f} s = {:.f} hrs'.format(tempo,horas))

```

Tabela II: Resultados dos cálculos intermediários das atividades 4a e 4b.

Resultados Intermediários						
Atividade	r_A (km)	r_C (km)	e_1	h_1 (km ² /s)	$v_{A,1}$ (km/s)	r_B (km)
4a	6858	7178	0.022799	52881.011040	7.710850	22378
4b	6678	6678	0	51597.560840	7.726499	9378
Atividade	e_2	h_2 (km ² /s)	$v_{A,2}$ (km/s)	$v_{B,2}$ (km/s)	$v_{B,3}$ (km/s)	-
4a	0.530852	64695.095877	9.433522	2.891013	4.220805	
4b	0.168161	55767.426569	8.350917	5.946623	6.520049	

Como pedido no enunciado, foi calculado o tempo de transferência orbital apenas na atividade 4b. Sendo que este foi obtido a partir da terceira lei de Kepler aplicada na órbita 2 e como neste é realizado apenas uma meia volta, o resultado é dividido por 2:

$$T_2'^2 = \frac{4\pi^2}{GM_{terra}} (R_{terra} + R_e)^3 \therefore T_2 = \frac{T_2'}{2},$$

onde R_e é a média simples dos pontos B e C. Os resultados finais estão listados na Tab. III.

Tabela III: Resultados finais das duas atividades.

Resultados Finais				
Atividade	Δv_A (km/s)	Δv_B (km/s)	Δv_{total} (km/s)	Tempo (h)
4a	1.722672	1.329792	3.052464	-
4b	0.624418	0.573426	1.197845	0.994151

IV. CONCLUSÃO

Devido aos sucessivos erros de arredondamento causados pelas escolhas numéricas das constantes (p.ex., massa da Terra e a constante gravitacional), os resultados finais da reprodução do exemplo de aula (atividade 4a) divergiram em algumas casas decimais. Apesar disto, os resultados deram na mesma ordem de grandeza e, portanto, conclui que o programa está funcionando corretamente.