

Relatório 5 - Manobras Orbitais (AGA0521)

Kethelin Parra Ramos - 9898349

I. INTRODUÇÃO

As transferências de Hohmann são manobras eficientes que podem ser utilizadas em missões interplanetárias. Na Fig.2 temos a ilustração de um exemplo da transferência de Hohmann interplanetária.

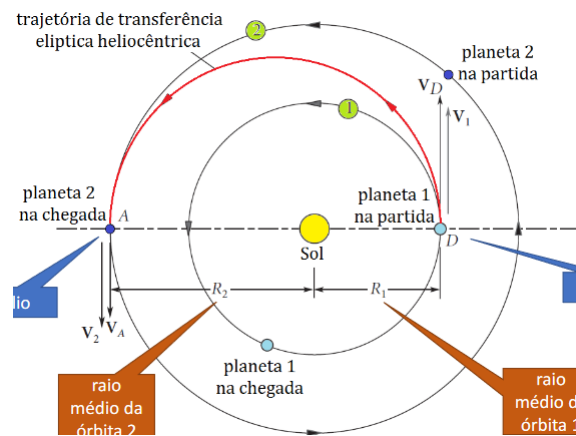


Figura 1: Exemplo de uma transferência de Hohmann interplanetária. Fonte: slides de aula.

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.

Foi elaborado um programa que calculasse os parâmetros necessários para uma transferência interplanetária, testando o exemplo Terra-Marte e o caso Terra-Júpiter. Para isto, foi criada a função *interplanetario()* que seguia os passos mostrados em aula (código abaixo). Todos os dados utilizados na atividade foram retirados do slide do enunciado.

```

1  #Bibliotecas
2  import numpy as np
3
4  """                                     Funções                                     """
5  #Função que calcula os parâmetros para uma transferência interplanetária
6  def interplanetario(mu_ini,r_ini,SMA,orb_ini,Isp,g0):
7      mu_sol = 1.327E11                    #Parâmetro gravitacional do Sol (km³/s²)
8      R_ini,R_fim = SMA[0],SMA[1]          #Semi eixos maiores dos planetas (inicial - final)
9
10     print("\nResultados intermediarios:")
11
12     #Calculo da velocidade de excesso hiperbólico
13     v_inf = np.sqrt(mu_sol/R_ini)*(np.sqrt(2*R_fim/(R_ini+R_fim))-1)
14     print('A velocidade de excesso hiperbólico = ',v_inf,' km/s')
15
16     #Calculo da velocidade da nave na órbita circular
17     v_c = np.sqrt(mu_ini/(r_ini+orb_ini))
18     print('A velocidade da nave na órbita circular = ',v_c,' km/s')
19
20     #Calculo do delta v necessário para definir a órbita hiperbólica de partida
21     deltav = v_c*(np.sqrt(2+(v_inf/v_c)**2)-1)
22
23     #Calculo do perigeu da hipérbole de partida
24     beta = np.arccos(1/(1+(v_inf/v_c)**2))
25     beta = np.degrees(beta)    #Deixando em graus
26
27     #Calculando a quantidade de propelente requerida
28     eq_foguete = 1 - np.exp(-deltav/(g0*Isp))
29
30     return deltav,beta,eq_foguete
31
32     #Calcula o período sinódico
33     def t_sinodico(t_ini,t_fim):
34         return (t_ini*t_fim)/np.abs(t_ini-t_fim)

```

III. RESULTADOS

```

1  # Atividade 5a
2
3  '''
4  a) Verifique se os resultados estão de acordo com o exemplo apresentado na Aula 5,
5  explicitando todos os resultados intermediários e o resultado final.
6  b) Calcule o período sinódico entre Marte e a Terra.
7
8  '''
9  #Dados (exercício a)
10 mu_Terra = 398600.  #Parâmetro gravitacional da Terra (km³/s²)
11 Rterra = 6378.      #Raio da Terra (km)
12 SMaterra = 149.6E6  #Semi-eixo maior da Terra (km)
13 SMamarte = 227.9E6  #Semi-eixo maior de Marte (km)
14 orb_ini = 300.      #Órbita inicial (km)
15 Isp = 300           #Impulso específico (s)
16 g0 = 9.81E-3        #Aceleração inicial (km/s²)
17
18 SMAs = [SMaterra, SMamarte]
19
20 resultados = interplanetario(mu_Terra,Rterra,SMAs,orb_ini,Isp,g0)
21
22 #Dados (exercício b)
23 T_terra = 365.26    #Período sideral da Terra (dias)
24 T_marte = 687.99    #Período sideral de Marte (dias)
25 t_syn = t_sinodico(T_terra,T_marte)
26
27 print("\nResultado final - Atividade 5a:")
28 print('a) dv = {:.f} km/s'.format(resultados[0]))
29 print('    beta = {:.f} graus'.format(resultados[1]))
30 print('    dM/m= {:.f}'.format(resultados[2]))
31 print('b) T_s = {:.f} dias'.format(t_syn))

```

```

Resultados intermediarios:
A velocidade de excesso hiperbólico = 2.9433246203696513 km/s
A velocidade da nave na órbita circular = 7.725835197559566 km/s

Resultado final - Atividade 5a:
a) dv = 3.589651 km/s
   beta = 29.160748 graus
   dM/m= 0.704689
b) T_s = 778.654688 dias

```

Figura 2: Saídas da atividade 5a.

```

1  # Atividade 5b
2
3  """
4  Aplique o programa da atividade anterior para:
5      a) uma missão semelhante, mas tendo Júpiter como destino.
6      b) Calcule o período sinódico entre Júpiter e a Terra.
7  """
8
9
10 #Dados (exercício a)
11
12 SMAjup = 778.6E6      #Semi-eixo maior de Marte (km)
13 orb_ini = 300.        #Órbita inicial (km)
14
15 SMAs = [SMAterra, SMAjup]
16
17 resultados = interplanetario(mu_Terra,Rterra,SMAs,orb_ini,Isp,g0)
18
19 #Dados (exercício b)
20 T_jupiter = 11.86*394 #Período sideral de Júpiter (anos ---> dias)
21 t_syn = t_sinodico(T_terra,T_jupiter)
22
23 print("\nResultado final - Atividade 5b:")
24 print('a) dv = {:.f} km/s'.format(resultados[0]))
25 print('   beta = {:.f} graus'.format(resultados[1]))
26 print('   dM/m= {:.f}'.format(resultados[2]))
27 print('b) T_s = {:.f} dias'.format(t_syn))

```

```
Resultados intermediarios:  
A velocidade de excesso hiperbólico = 8.793261390124545 km/s  
A velocidade da nave na órbita circular = 7.725835197559566 km/s  
  
Resultado final - Atividade 5b:  
a) dv = 6.299089 km/s  
   beta = 64.173276 graus  
   dM/m= 0.882388  
b) T_s = 396.232116 dias
```

Figura 3: Saídas da atividade 5b.

IV. CONCLUSÃO

Como o resultado da questão 5a coincidiu com o exemplo de aula, conclui que o programa está funcionando corretamente.