

# CINEMÁTICA E DINÂMICA DAS MÁQUINAS

Kleber Rodrigo da Silva Junior

22 de setembro de 2022

## 1 Definição dos Links

Para o trabalho foi utilizado a posse de 4 valores de diferentes para os links, tendo  $L1 = 300$ ,  $L2 = 120$ ,  $L3 = 180$  e  $L4 = 240$ . Porém essa escolha não pode ser qualquer valor, pois existe uma relação de existência do mecanismo dado pelo conjunto de Equação (1).

$$\begin{aligned} L2 + L3 + L4 &\geq L1 \\ L2 + L3 - L4 &\leq L1 \\ L2 + L1 + L4 &\geq L3 \\ L2 + L1 - L4 &\leq L3 \end{aligned} \tag{1}$$

Substituindo os valores escolhidos para os links na Equação (1), confirmamos assim a existência do mecanismo.

$$\begin{aligned} 120 + 180 + 240 &\geq 300 \rightarrow 540 \geq 300 \\ 120 + 180 - 240 &\leq 300 \rightarrow 60 \leq 300 \\ 120 + 300 + 240 &\geq 180 \rightarrow 660 \geq 180 \\ 120 + 300 - 240 &\leq 180 \rightarrow 180 \leq 180 \end{aligned}$$

## 2 Construção do Mecanismo

Para tratar o mecanismo, foi montado um esquema no software CAD/CAE FUSION360, apresentado na Figura 1, na forma aberta e cruzada. O modelo efetuou o giro completo, provando assim a eficiência das equações de existência.

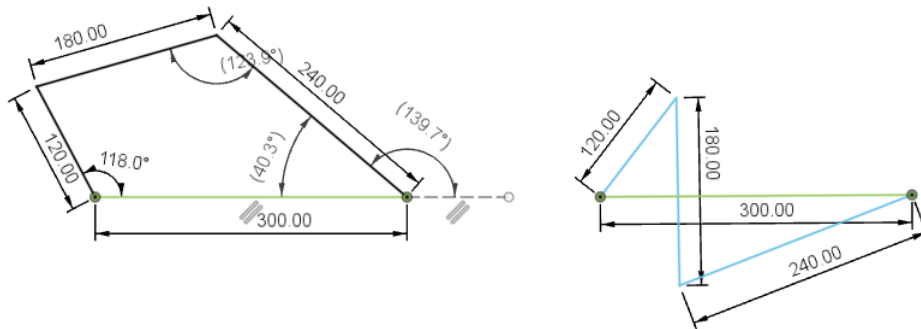


Figura 1: Mecanismo de 4 barras

Como pode-se observar na Figura 1, o modelo se trata de um mecanismo balancinho-manivela, melhor descrito pelo vídeo anexado junto ao documento.

### 3 Plotagem do Gráfico

Para a plotagem do gráfico utiliza-se a ferramenta de programação python, utilizando as equações que definem os ângulos a partir de  $\theta_2$ , dado pelo seguinte código:

```
1 #Codigo para solucao do problema do mecanismo de 4 barras
2 #Autor: Kleber Junior
3 import numpy as np
4 import math as m
5 import matplotlib.pyplot as plt
6 #Entrada:
7 L1 = 300
8 L2 = 120
9 L3 = 180
10 L4 = 240
11 theta2 = 1
12 eixo_z=[]
13 eixo_gamma=[]
14 eixo_alpha=[]
15 eixo_beta=[]
16 eixo_theta4=[]
17 eixo_theta2=[]
18
19 while (theta2<720):
20     z = m.sqrt((L1**2) + (L2**2) - (2*L1*L2*m.cos(theta2*m.pi/180.0)))
21     gamma = (180*m.acos(((z**2) - (L3**2) - (L4**2))/(-2*L3*L4))/m.pi)
22     alpha = (180*m.acos(((z**2) + (L4**2) - (L3**2))/(2*z*L4))/m.pi)
23     beta = (180*m.acos(((z**2) + (L1**2) - (L2**2))/(2*z*L1))/m.pi)
24     theta4 = (180-(alpha+beta))
25     eixo_z.append(z)
26     eixo_gamma.append(gamma)
27     eixo_alpha.append(alpha)
28     eixo_beta.append(beta)
29     eixo_theta4.append(theta4)
30     eixo_theta2.append(theta2)
31     theta2 = theta2 + 0.1
32
33 plt.figure(1)
34 plt.plot(eixo_theta2,eixo_z , 'b', label='z')
35 plt.plot(eixo_theta2,eixo_gamma , 'r', label='gamma')
36 plt.plot(eixo_theta2,eixo_alpha , 'y', label='alpha')
37 plt.plot(eixo_theta2,eixo_theta4 , 'g', label='theta4')
38
39 plt.ylabel('$\theta$', fontsize=12)
40 plt.xlabel('$\theta_2$', fontsize=12)
41 plt.grid(axis='both')
42 plt.legend(fontsize=10)
43 plt.title('Ângulos em função de  $\theta_2$ ')
44 plt.show()
```

Listing 1: Código python para simular os ângulos

No final da execução do código, ele plota um gráfico com a relação dos outros ângulos em função de  $\theta_2$  dado pela Figura 2, ambos os eixos estão em graus.

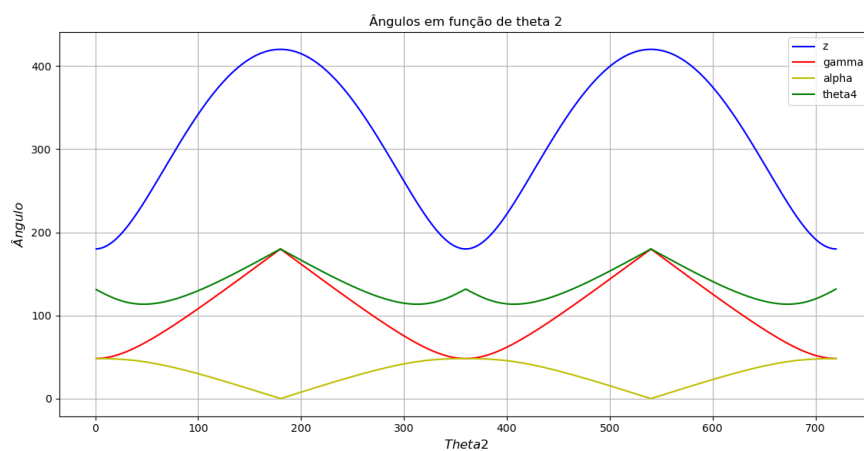


Figura 2: Gráfico

## Referências

1. Norton, R. (2011). *Cinemática e dinâmica dos mecanismos*. AMGH, 1 edition.