



**SEM0530 Problemas de Engenharia Mecatrônica II**

Aula 4: Aproximação numérica de EDOs de 1<sup>a</sup>  
ordem

GABRIEL LUENEBERG - 14746439

PROFESSOR:

MARCELO A. TRINDADE

22/05/2024

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Objetivos</b>	<b>3</b>
<b>3</b>	<b>Resultados e Discussão</b>	<b>3</b>
3.1	Formulações . . . . .	3
3.2	Gráficos e resultados . . . . .	5
3.3	Código . . . . .	5
<b>4</b>	<b>Conclusão</b>	<b>7</b>

# 1 Introdução

Nem toda equação diferencial ordinária (EDO) pode ser resolvida analiticamente. Embora muitas EDOs tenham soluções exatas que podem ser expressas em termos de funções elementares, uma grande quantidade dessas equações não possui soluções que possam ser encontradas por métodos analíticos tradicionais. Isso se deve à complexidade das equações e às limitações dos métodos analíticos. A impossibilidade de resolver algumas EDOs analiticamente destaca a importância dos métodos numéricos e das ferramentas computacionais na análise e na solução de problemas de engenharia.

## 2 Objetivos

A prática tem como objetivo analisar e simular o movimento de um corpo seguindo uma trajetória específica em coordenadas polares, descrita por  $r(\theta) = 20(3 - 2\cos(\theta))$  cm, e uma velocidade prescrita  $v(t) = 0,05 * (100 + 39)(100 - t)$  mm/s. Inicialmente, calcula-se a velocidade angular  $\dot{\theta}$  necessária para manter essa velocidade, seguida pela determinação da evolução do deslocamento angular  $\theta(t)$  ao longo de três voltas completas, utilizando um algoritmo de integração numérica. Em paralelo, analisa-se a evolução do deslocamento radial  $r(t)$  ao longo do tempo, considerando a relação fornecida. Os resultados são apresentados em gráficos detalhados de  $\theta$  vs  $t$ ,  $r$  vs  $t$ ,  $\dot{\theta}$  vs  $t$ ,  $\dot{r}$  vs  $t$ ,  $r$  vs  $\theta$  e  $v$  vs  $t$ , permitindo uma visualização clara do comportamento do movimento. Por fim, calcula-se o tempo necessário para que o corpo complete três voltas completas.

## 3 Resultados e Discussão

### 3.1 Formulações

Em primeira instância, tem-se que o vetor velocidade de um corpo que executa uma trajetória descrita, como a do problema, em coordenadas polares, pode ser expresso por:

$$\vec{v}(t) = v_r(t)\vec{u}_r + v_\theta(t)\vec{u}_\theta \quad (1)$$

onde tem-se que:

$$v_r(t) = \frac{dr(t)}{dt} = \dot{r}(t) \quad \text{e} \quad v_\theta(t) = r(t)\frac{d\theta(t)}{dt} = r(t)\dot{\theta}(t) \quad (2)$$

De forma compacta:

$$\vec{v} = v_r\vec{u}_r + v_\theta\vec{u}_\theta, \quad \text{com} \quad v_r = \dot{r} \quad \text{e} \quad v_\theta = r\dot{\theta} \quad (3)$$

Para o problema em questão, derivou-se  $r(\theta)$  em função de  $\theta$  para se obter

$$\dot{r}(\theta) = 40 \sin \theta \frac{d\theta}{dt} \quad (4)$$

Assim,

$$v_r = 40 \sin \theta \frac{d\theta}{dt} \quad (5)$$

Sabe-se que o módulo de  $\vec{v}$  é dado por:

$$v(t) = |\vec{v}(t)| = \sqrt{v_r^2(t) + v_\theta^2(t)} \implies v^2(t) = v_r^2(t) + v_\theta^2(t) \quad (6)$$

Desse modo,

$$v^2(t) = \left( 40 \sin \theta \frac{d\theta}{dt} \right)^2 + \left[ r \frac{d\theta}{dt} \right]^2 \quad (7)$$

$$\implies \frac{d\theta}{dt} = \frac{v(t)}{\sqrt{1600\theta^2 + r(\theta)^2}} \quad (8)$$

Tem-se, dessa maneira, que  $\frac{d\theta}{dt}$  se apresenta como uma função de  $t$  e  $\theta$ , ou seja,

$$\frac{d\theta}{dt} = \psi(\theta, t) \quad (9)$$

Por meio da expressão (8), pode-se utilizar um algoritmo de integração numérica por meio da linguagem *Python* para obter a evolução do deslocamento angular  $\theta(t)$  da partícula, e consequentemente determinar as outras evoluções desejadas.

$$t_{6\pi} = 1,8s \quad (10)$$

## 3.2 Gráficos e resultados

Para determinar a evolução do deslocamento angular, utilizou-se o método *solve\_ivp* da biblioteca *scipy.integrate* para obter o vetor de valores de  $\theta(t)$ , considerando o vetor de tempo passado como argumento. Vale ressaltar que, a princípio, não era conhecido o tempo decorrido após três voltas completas. Sendo assim, estipulou-se um intervalo arbitrário de teste em que  $t$  variou de 0 a 100.

Em seguida, encontrou-se a primeira posição no vetor  $\theta(t)$  a partir da qual já se havia percorrido três voltas, ou seja,  $6\pi$  rad. O elemento de posição correspondente no vetor tempo é aproximadamente o tempo de três voltas. Assim, foi possível determinar também as evoluções de  $r(t)$ ,  $\dot{r}(t)$  e  $\dot{\theta}(t)$  em função do tempo. Os gráficos puderam também ser construídos. Com a discretização utilizada, chegou-se à seguinte aproximação para o tempo  $t_{6\pi}$  decorrido após 3 voltas:

$$t_{6\pi} = 1,8s \quad (11)$$

A seguir são mostrados os gráficos correspondentes a algumas evoluções relacionadas ao movimento descrito pela partícula.

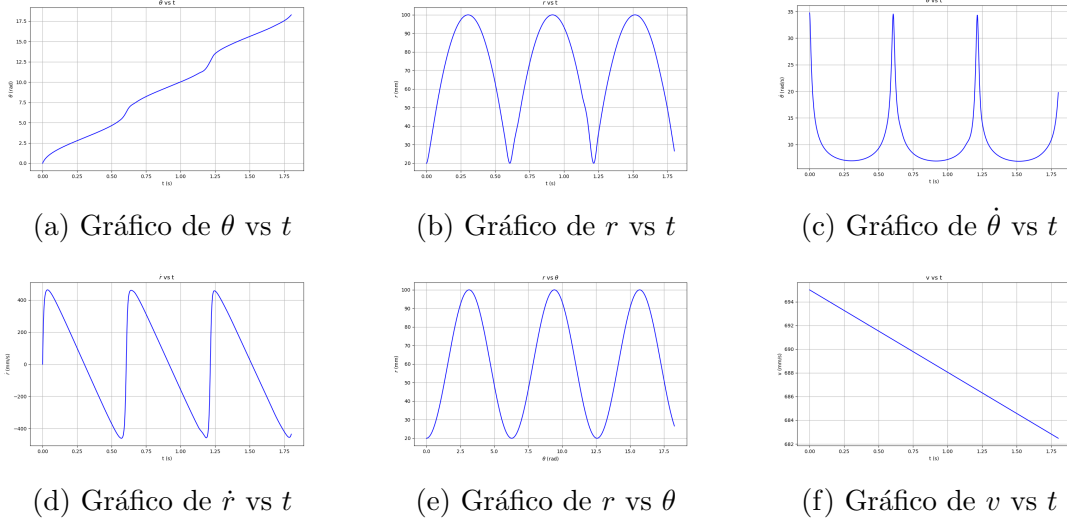


Figura 1: Gráficos gerados

## 3.3 Código

Os códigos utilizados para a obtenção das evoluções e seus gráficos encontra-se a seguir

```
1 import numpy as np
2 from scipy.integrate import solve_ivp
3 import matplotlib.pyplot as plt
```

```

4
5 N = 39
6
7 def r(theta):
8     return 20 * (3 - 2 * np.cos(theta))
9
10 def v(t):
11     return 0.05 * (100 + N) * (100 - t)
12
13 def theta_ponto(t, theta):
14     return v(t) / np.sqrt(1600 * np.sin(theta)**2 + r(theta)**2)
15
16 def r_ponto(t, theta):
17     return 40 * np.sin(theta) * theta_ponto(t, theta)
18
19 theta_initial = 0
20 t = (0, 100)
21
22 sol = solve_ivp(theta_ponto, t, [theta_initial], t_eval=np.linspace(t
    [0], t[1], 1000))
23
24 index = np.abs(sol.y[0] - 6 * np.pi).argmin()
25 tempo = sol.t[index]
26
27 t = (0, tempo)
28 print(tempo)
29 sol = solve_ivp(theta_ponto, t, [theta_initial], t_eval=np.linspace(t
    [0], t[1], 1000))
30
31 plt.figure(figsize=(10, 6))
32 plt.plot(sol.t, sol.y[0], 'b')
33 plt.xlabel('t (s)')
34 plt.ylabel(r'$\theta$ (rad)')
35 plt.title(r'$\theta$ vs t')
36 plt.grid(True)
37 plt.show()
38
39 val_r = r(sol.y[0])
40
41 plt.figure(figsize=(10, 6))
42 plt.plot(sol.t, val_r, 'b')
43 plt.xlabel('t (s)')
44 plt.ylabel(r'$r$ (mm)')
45 plt.title(r'$r$ vs t')
46 plt.grid(True)
47 plt.show()
48
49 val_theta_ponto = theta_ponto(sol.t, sol.y[0])
50
51 plt.figure(figsize=(10, 6))
52 plt.plot(sol.t, val_theta_ponto, 'b')
53 plt.xlabel('t (s)')
54 plt.ylabel(r'$\dot{\theta}$ (rad/s)')
55 plt.title(r'$\dot{\theta}$ vs t')
56 plt.grid(True)
57 plt.show()
58
59 val_r_ponto = r_ponto(sol.t, sol.y[0])

```

```

60
61 plt.figure(figsize=(10, 6))
62 plt.plot(sol.t, val_r_ponto, 'b')
63 plt.xlabel('t (s)')
64 plt.ylabel(r'$\dot{r}$ (mm/s)')
65 plt.title(r'$\dot{r}$ vs t')
66 plt.grid(True)
67 plt.show()
68
69 plt.figure(figsize=(10, 6))
70 plt.plot(sol.y[0], val_r, 'b')
71 plt.xlabel(r'$\theta$ (rad)')
72 plt.ylabel(r'$r$ (mm)')
73 plt.title(r'$r$ vs $\theta$')
74 plt.grid(True)
75 plt.show()
76
77 val_v = v(sol.t)
78
79 plt.figure(figsize=(10, 6))
80 plt.plot(sol.t, val_v, 'b')
81 plt.xlabel('t (s)')
82 plt.ylabel('v (mm/s)')
83 plt.title('v vs t')
84 plt.grid(True)
85 plt.show()

```

## 4 Conclusão

Os resultados obtidos por meio das formulações e cálculos apresentados nesta seção fornecem uma visão detalhada do movimento da partícula. Foi possível determinar a evolução do deslocamento angular  $\theta(t)$  da partícula, bem como as evoluções de  $r(t)$ ,  $\dot{r}(t)$  e  $\dot{\theta}(t)$  em função do tempo. Os gráficos gerados a partir desses cálculos ilustram claramente as evoluções relacionadas ao movimento da partícula. Em suma, os resultados apresentados oferecem uma melhor compreensão do problema em questão. Eles destacam a utilidade e a eficácia das técnicas de aproximação numérica de EDOs de 1ª ordem na resolução de problemas complexos de engenharia.