

RODA DE MAXWELL

Lucas Vargas, Renan Guedes e Bruno Rodrigues



Trabalho de Modelagem Matemática

IDENTIFICAÇÃO E ANÁLISE DO PROBLEMA

Consiste em um eixo preso a uma haste por meio de dois cabos, um em cada extremidade. No meio deste eixo, há uma roda que é atravessada pelo eixo, que permanece orientado na vertical.

**É uma ótima forma de demonstrar a conservação
de energia mecânica.**



DEMONSTRAÇÃO DO FUNCIONAMENTO

Ao girar o eixo, os fios se enrolam, elevando a roda até uma certa altura. Quando ela é solta, a roda começa a descer enquanto gira, transformando energia potencial em energia cinética de translação e rotação.

Quando os fios se desenrolam completamente, a rotação faz a roda subir novamente, enrolando os fios no sentido oposto. Esse movimento de subir e descer se repete várias vezes, com altura cada vez menor devido às perdas de energia.



IDENTIFICAÇÃO E ANÁLISE DO PROBLEMA

VARIÁVEIS FUNDAMENTAIS

CONSIDERADAS

- Massa do disco
- O raio do disco e do eixo
- O tamanho do cabo que conecta o disco com a haste (altura)
- Aceleração da gravidade
- Altura inicial

DESCONSIDERADAS

- Massa do eixo
- Energia do Calor gerado
- Energia do Som gerado
- Resistência do ar
- Atrito entre a linha e o eixo

ESCOLHA DE FERRAMENTA UTILIZADA NA MODELAGEM

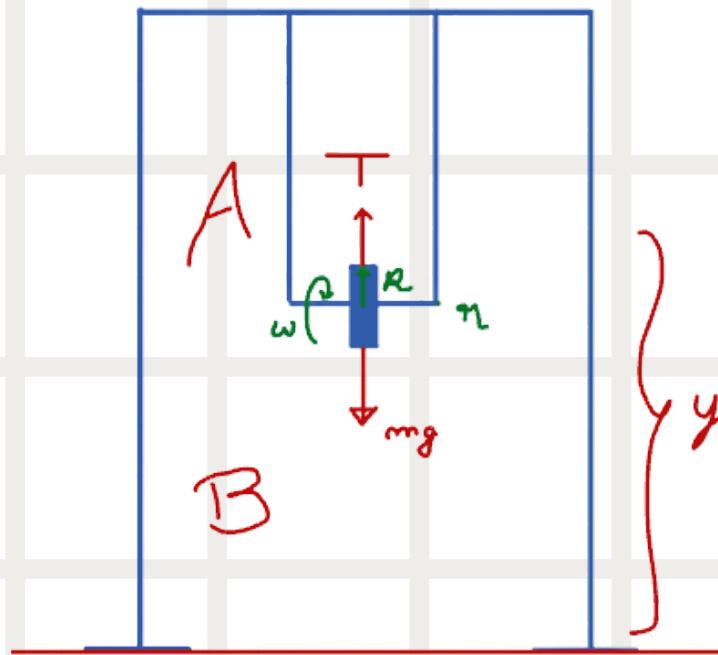
Mecânica Lagrangiana

Usamos o lagrangiano e as equações de Euler-Lagrange para encontrar as equações de movimento com restrições.

Conservação de energia:

A energia mecanica total (potencial + cinética) é conservada e a energia potencial gravitacional perdida é convertida em 2 formas de energia cinética: translacional e rotacional.

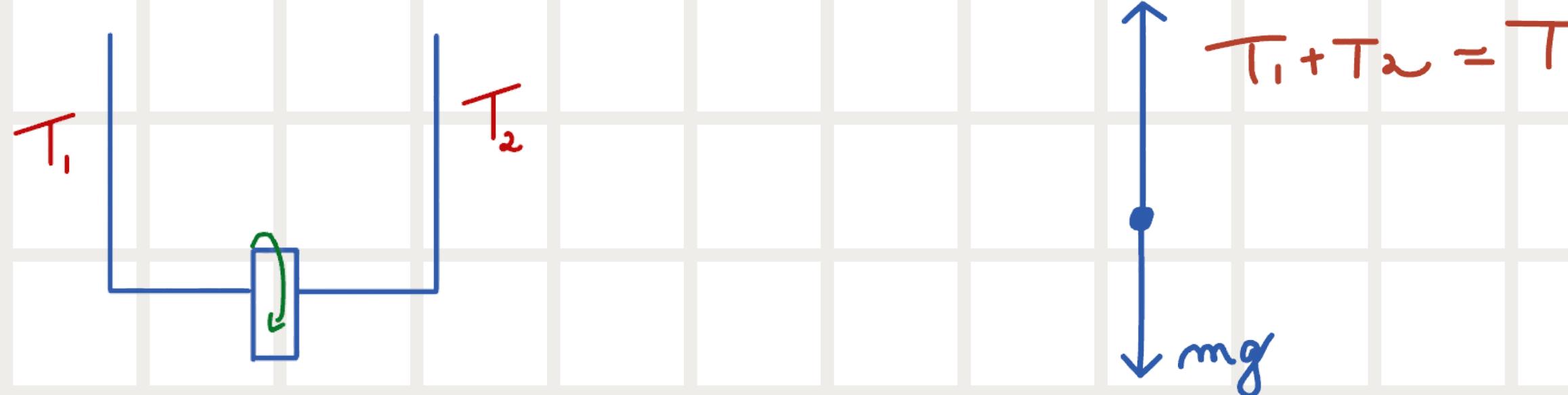
Modelando as equações e estruturas utilizadas no modelo



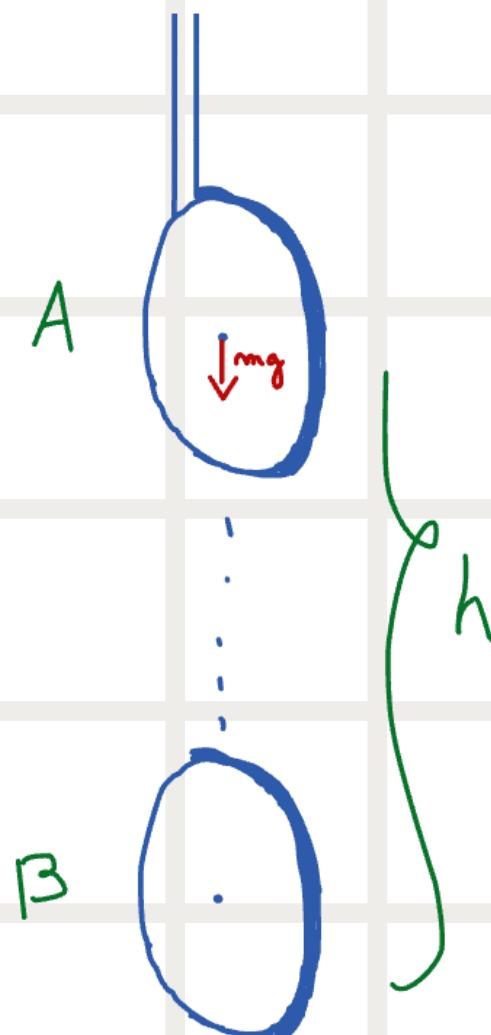
- T é a soma das tensões dos fios
- A e B são dois pontos, onde A que marca a altura inicial da roda e B o ponto mais baixo.
- ω é a velocidade angular

- y é a altura da base até o disco
- R é raio do disco, e r o raio do eixo
- m representa a massa do disco.

No caso que estamos tratando, a força atuando sobre o corpo não é tão simples de calcular como seria se fosse uma queda livre, e isso é possível ver pelo esquema ao lado:



Por isso, vamos começar analisando um caso mais simples, onde o comportamento seria de queda livre, um iôô



Levando em consideração a conservação de energia e que no ponto A apenas temos energia potencial gravitacional agindo no corpo, e que no ponto B, temos apenas energia cinética. A energia cinética é a soma da energia translacional e rotacional, assim podemos igualar ficando da seguinte maneira:

$$T = E_{\text{Trans}} + E_{\text{ROT}}$$

Agora substituindo e fazendo os cálculos, temos:

$$mgh = \frac{mv^2}{2} + \frac{I\omega^2}{2}$$

$$mgh = \frac{mv^2}{2} + \frac{\frac{mR^2}{2} \frac{v^2}{R}}{2}$$

$$gh = \frac{3v^2}{4}$$

$$I_z = \frac{1}{2}mr^2$$

$$v = rw$$

Lembrando dos nossos aprendizados de cinemática, temos o seguinte:

$$v^2 = v_0^2 + 2ah$$

No caso que estamos considerando, $v_0 = 0$, logo:

$$v^2 = 2ah$$

$$3v^2 = 4gh \rightsquigarrow 6ah = 4gh$$

$$a = \frac{4g}{6}$$

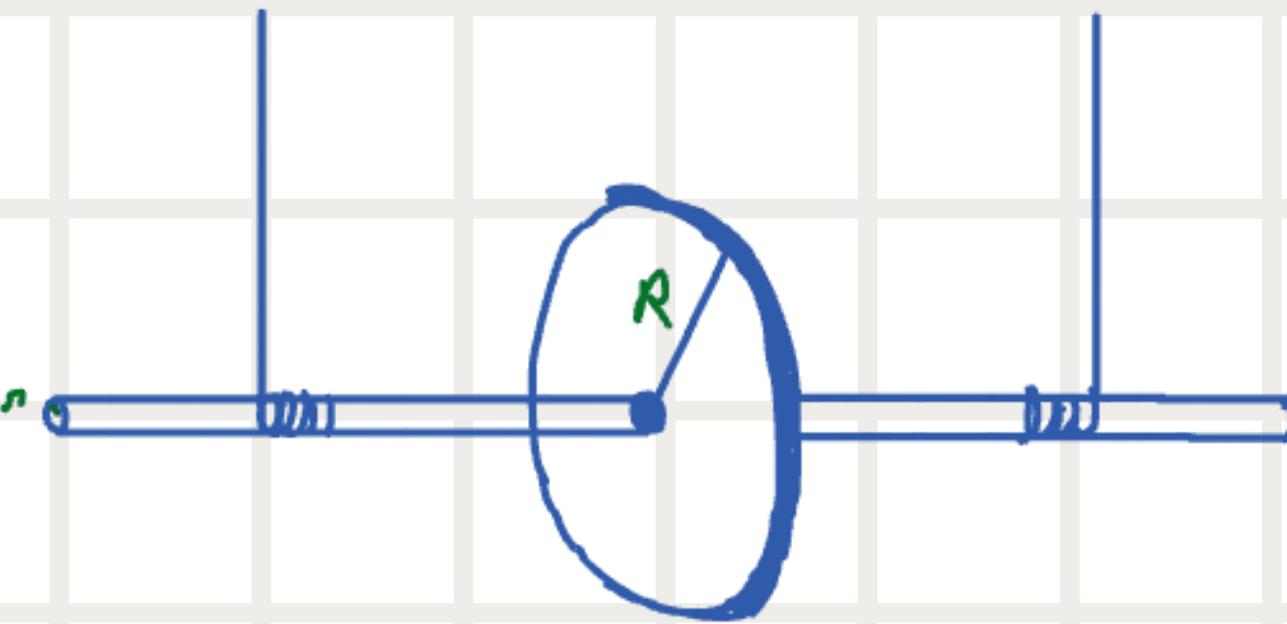
$$a = \frac{2g}{3}$$

Assim, descobrimos que a aceleração é inferior à aceleração da gravidade.

Voltando para o caso da roda de Maxwell

Aqui é importante ressaltar que estamos considerando no caso onde a roda é muito mais pesada que o eixo que a atravessa. Essas conclusões seriam um pouco diferentes se o eixo tivesse um peso parecido ou maior que o da roda.

$$mgh = \frac{mv^2}{2} + \frac{\frac{mR^2}{2} \left(\frac{v}{r}\right)^2}{2}$$

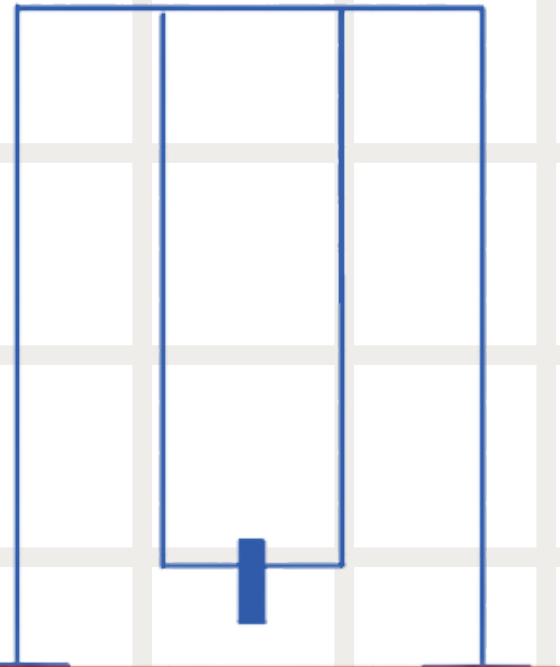


Perceba que estamos considerando neste caso $w = v/r$, sendo r o raio do eixo. Agora, resolvendo essa equação analogamente ao caso anterior, teremos:

$$a = \frac{g}{\left(1 + \frac{R^2}{2\pi^2}\right)}$$

Agora, em efeito de demonstração, vamos calcular a aceleração utilizando a Mecânica Lagrangiana:

Sebemos que no momento mais baixo a altura é 0, com isso temos:



$$E_k = \frac{m\dot{\psi}^2}{2} + \frac{I\omega^2}{2}$$

$$E_k = \frac{m\dot{\psi}^2}{2} + \frac{I\dot{\psi}^2}{2R^2}$$

Desenvolvendo E_k : $E_k = \frac{\dot{\psi}^2}{2} \left(m + \frac{I}{R^2} \right) \Leftrightarrow E_k = \frac{\dot{\psi}^2}{2} \left(m + \frac{I}{2r^2} \cdot \frac{I}{2} \cdot m R^2 \right) \Leftrightarrow E_k = m \cdot \frac{\dot{\psi}^2}{2} \left(1 + \frac{R^2}{2r^2} \right)$

E no momento mais alto, a altura = h e $E_k = 0$. Com isso, sabemos que somente temos a ação da energia potencial, que é mgh . Utilizando Lagrange vamos ter :

$$\mathcal{L} = T - V$$

$$L = \frac{m\dot{\psi}^2}{2} \left(1 + \frac{R^2}{2r^2} \right) - mgh$$

$$L = m \left(\frac{\dot{y}^2}{2} + \frac{\dot{\psi}^2 R^2}{4r^2} - gy \right)$$

A equação de movimento é dada pela equação de Euler-Lagrange :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{y}} \right) - \frac{\partial L}{\partial y} = 0$$

Calculando cada derivada de Euler-Lagrange:

$$\frac{\frac{\partial L}{\partial \dot{y}}}{\cdot} = m\ddot{y} + \frac{\dot{y}mR^2}{2\pi^2} \rightarrow \frac{d}{dt} \left(m\ddot{y} + \frac{\dot{y}mR^2}{2\pi^2} \right) = m\ddot{y} + \frac{\ddot{y}mR^2}{2\pi^2} \text{ e } \frac{\partial L}{\partial y} = -mg$$

Substituindo os valores obtidos na equacao de Euler-Lagrange teremos :

$$m\ddot{y} + \frac{\ddot{y}mR^2}{2\pi^2} + mg = 0 \Leftrightarrow \ddot{y} = \frac{-g}{\left(1 + \frac{R^2}{2\pi^2} \right)}$$

Com isso podemos perceber que a aceleração quando o disco sobe é igual com sentido oposto à encontrada anteriormente.
Isso confirma nossas expectativas e corrobora com os resultados que chegamos ao igualar as energias,

Utilizando a função que encontramos anteriormente da aceleração, nos calculamos o valor dela, que é -0.0349 m/s^2 . (Negativa pois está para baixo, assim como a velocidade está negativa antes do "tranco").

Podemos chegar na função da posição ao integrar em relação ao tempo duas vezes. Integrando a primeira vez, encontramos:

$$\dot{y}(\tau) = -\left(\frac{g}{1 + \frac{R^2}{2\pi^2}}\right)\tau + C_1$$

Como a velocidade inicial $V(0) = 0$, a constante de integração $C_1 = 0$.

E agora, integrando mais uma vez, temos:

$$y(\tau) = H_0 - \frac{\tau^2}{2} \left(\frac{g}{1 + \frac{R^2}{2\pi^2}}\right)$$

Como a posição inicial, ou seja, $h(0)$, foi definida como a variável H_0 , temos que essa é a constante de integração “ C_2 ”, que vale 55 cm no nosso experimento.

Podemos observar que essa equação encontrada se trata justamente da fórmula:

$$\dot{y}(\tau) = -\left(\frac{g}{1 + \frac{R^2}{2n^2}}\right)\tau + C_2 \rightarrow v(t) = v_0 + a \cdot t$$

Onde $v_0 = 0$, pois partimos do repouso

Observamos também isso na segunda equação integrada, que se trata da seguinte equação:

$$y(\tau) = H_0 - \frac{\tau^2}{2} \left(\frac{g}{1 + \frac{R^2}{2n^2}} \right) \rightarrow S = S_0 + v_0 t + \frac{at^2}{2}$$

S_0 sendo a variável H_0 , que é a altura inicial valendo 55 cm no nosso experimento, e V_0 , já dito antes, vale zero pois parte do repouso.

Escolha do Modelo

- Ao analisar qual solução utilizar, entre a solução analítica ou um método numérico, decidimos primeiramente usar a solução analítica pois a aceleração é constante, com isso, poderíamos integrar para obter as equações exatas da velocidade e posição por tempo, porém, sabemos as fórmulas exatas para a velocidade e para a posição na cinemática, então às utilizamos:

$$V = V_0 + at$$

$$S = S_0 + V_0 t + \frac{at^2}{2}$$

- Mas, por fim de comparações, decidimos escolher algum método numérico para realizar uma comparação dos resultados encontrados das duas maneiras diferentes, em velocidade e em posição.

Escolha do Método Numérico

- Escolhemos utilizar o método de Euler-Cromer.
- Escolhemos esse método pois ele considera a velocidade como variável ao longo do tempo, diferente do método de Euler por exemplo, que considera a velocidade como constante, o que não condiz com o nosso experimento.
- Além disso, usar um método muito completo como o de Runge-Kutta de quarta ordem seria desnecessário para o problema modelado aqui, já que, por não ter perda de energia na nossa modelagem, a aceleração será constante, já que não depende da altura e nem da velocidade, como podemos ver pela equação encontrada.

$$v_{n+1} = v_n + a_n \cdot \Delta t$$

$$h_{n+1} = h_n + v_{n+1} \cdot \Delta t$$

Modelando as equações e estruturas utilizadas no modelo

```
import sympy as sp
import numpy as np
import matplotlib.pyplot as plt

# Definição de variáveis

g_val = 9.81 # Aceleração da gravidade em metros por segundo ao quadrado
R_val = 0.07 # Raio do disco em metros
r_val = 0.003 # Raio da haste em metros
m_val = 0.45 # Massa em KG do disco
H0_val = 0.55 # Altura inicial em que será solto a roda em metros
dt = 0.001 # Variação de tempo em segundos, decidimos arbitrariamente como parâmetro de integração

# Momento de inércia para o eixo Z de um cilindro

# I = (m * R**2) / 2

# Definição das energias do sistema para Lagrange

# Energia cinética

# T = ((m * v**2) / 2) * (1 + R**2/2*r**2)

# Energia potencial gravitacional

# U = m * g * h

# Lagrangiano: L = T - U
```

Dados extraídos do artigo

<https://pt.scribd.com/document/530316484/P1461-Conservation-of-Energy-Maxwell-s-Wheel-1>

Esse código se refere ao cálculo de maneira analítica.

Todo o código que demonstrado está disponível no link do Google Colab no slide Fontes.

Nesta parte do código, definimos o Lagrangiano com base nas energias cinética e potencial do sistema que mostramos anteriormente,

Além disso, através da biblioteca sympy, chegamos na fórmula da aceleração utilizando as derivadas do lagrangiano

```
# Lagrangiano  
  
L = (((m * h_dot**2) / 2) * (1 + R**2/(2*r**2))) - (m * g * h)  
  
# Cálculo das derivadas para a equação de Lagrange  
  
DL_dh = sp.diff(L, h)  
DL_dh_dot = sp.diff(L, h_dot)  
d_dt_DL_dh_dot = sp.diff(DL_dh_dot, t)  
  
# Montamos a equação de Lagrange, EOM (Equation Of Movement)  
  
EOM = d_dt_DL_dh_dot - DL_dh  
  
h_ddot = sp.diff(h, t, 2) # Variável que desejamos isolar para integrarmos  
  
solucao_aceleracao = sp.solve(EOM, h_ddot) # isola a variável da aceleração  
  
# Imprimindo o resultado para checar  
  
print("Equação da aceleração é:")  
sp.pprint(solucao_aceleracao[0])  
  
Equação da aceleração é:  
2  
-2·g·r  
_____  
2 2  
R + 2·r
```

```

# Obter o valor numérico da aceleração
# Pega a fórmula simbólica da aceleração e substitui os símbolos que usamos nas variáveis
# g, R, r pelos seus valores numéricos definidos no início
a_eff_val = solucao_aceleracao[0].subs([
    (g, g_val),
    (R, R_val),
    (r, r_val)
])

# Converte o resultado do sympy para um número float
a_eff_val = float(a_eff_val)

print(f"Aceleração constante (Analítica): {a_eff_val:.4f} m/s^2")

# Usamos a equação da posição já conhecida na cinemática para descobrir o tempo total de queda
#  $h(t) = H_0 + 0.5*a*t^2 \Rightarrow 0 = H_0 + 0.5*a*t_{final}^2$ 
# t_final =  $\sqrt{-2 * H_0 / a}$ 
# Considerando que t_final é o tempo que a roda levou para chegar na altura zero
t_final = np.sqrt(-2 * H0_val / a_eff_val)

# Cria o array de tempo, de 0 até t_final, em passos 'dt'
t_array = np.arange(0, t_final + dt, dt)

print(f"Tempo de queda até h = 0 fica: t = {t_final:.2f} s")

# Aplicar as equações (as integrais exatas)

# Primeira integral (Equação da Velocidade:  $v(t) = a*t$ )
# (v_array_analitico irá guardar v(t) para cada ponto em t_array)
v_array_analitico = a_eff_val * t_array

# Segunda integral (Equação da Posição:  $h(t) = H_0 + 0.5*a*t^2$ )
# (h_array_analitico irá guardar h(t) para cada ponto em t_array)
h_array_analitico = H0_val + 0.5 * a_eff_val * t_array**2

# Correção: O último passo pode ficar < 0 por causa dos intervalos 'dt'
# Garantimos que a altura pare em 0.
h_array_analitico[h_array_analitico < 0] = 0

```

```

# Agora plotaremos os gráficos que queremos mostrar

# Vamos plotar a magnitude da velocidade, já que por estar contrária ao
# Apenas para deixar o gráfico mais intuitivo
v_mag_analitico = np.abs(v_array_analitico)

# Arrays de descida já existem:
# t_array (0 -> t_final)
# h_array_analitico (H0 -> 0)
# v_array_analitico (0 -> -v_max)

# Criar os dados da subida
# Usamos [1:] para não duplicar o ponto do meio (em t_final, h=0)
t_ascent = t_array[1:] + t_final
h_ascent = np.flip(h_array_analitico)[1:] # 'flip' inverte o array

# A velocidade de subida (h_dot) é positiva.
# v_array_analitico era negativo. Flip inverte a ordem.
# *-1 inverte o sinal (de negativo para positivo).
v_ascent = np.flip(v_array_analitico)[1:] * -1

# Concatenar para criar os arrays do ciclo completo
t_oscilacao = np.concatenate((t_array, t_ascent))
h_oscilacao = np.concatenate((h_array_analitico, h_ascent))
v_oscilacao = np.concatenate((v_array_analitico, v_ascent))

# Printa o tempo total da oscilação até chegar de volta no H0, onde v =
print(f"Tempo da oscilação até subir para a altura inicial novamente: {{"

```

Definição e aplicação das fórmulas mencionadas sobre velocidade e posição (primeira e segunda integrais da aceleração).

Com os cálculos feitos, geramos gráficos para visualizarmos melhor os resultados da nossa modelagem, e também compararmos com o esperado.

Gráfico da Posição vs Tempo

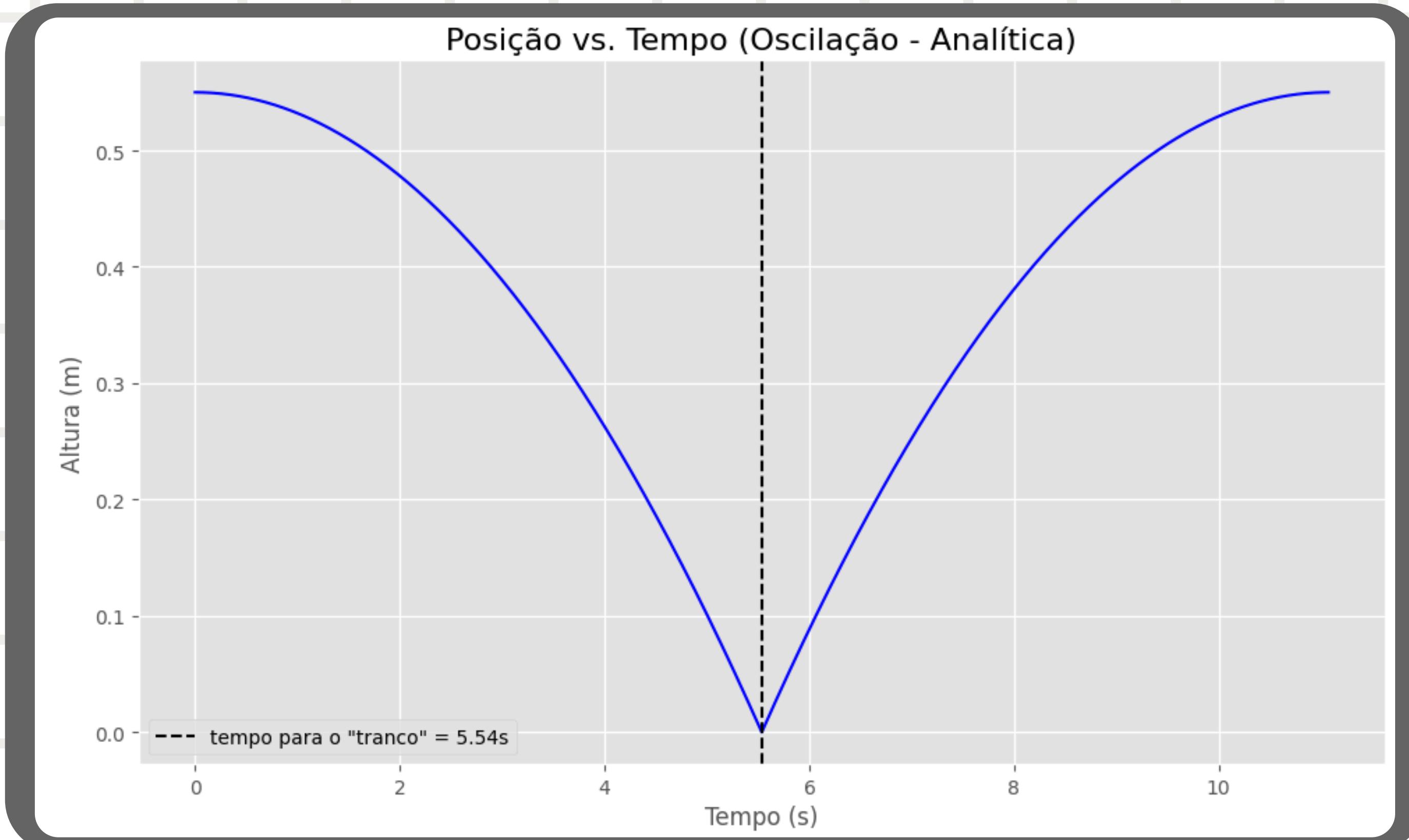
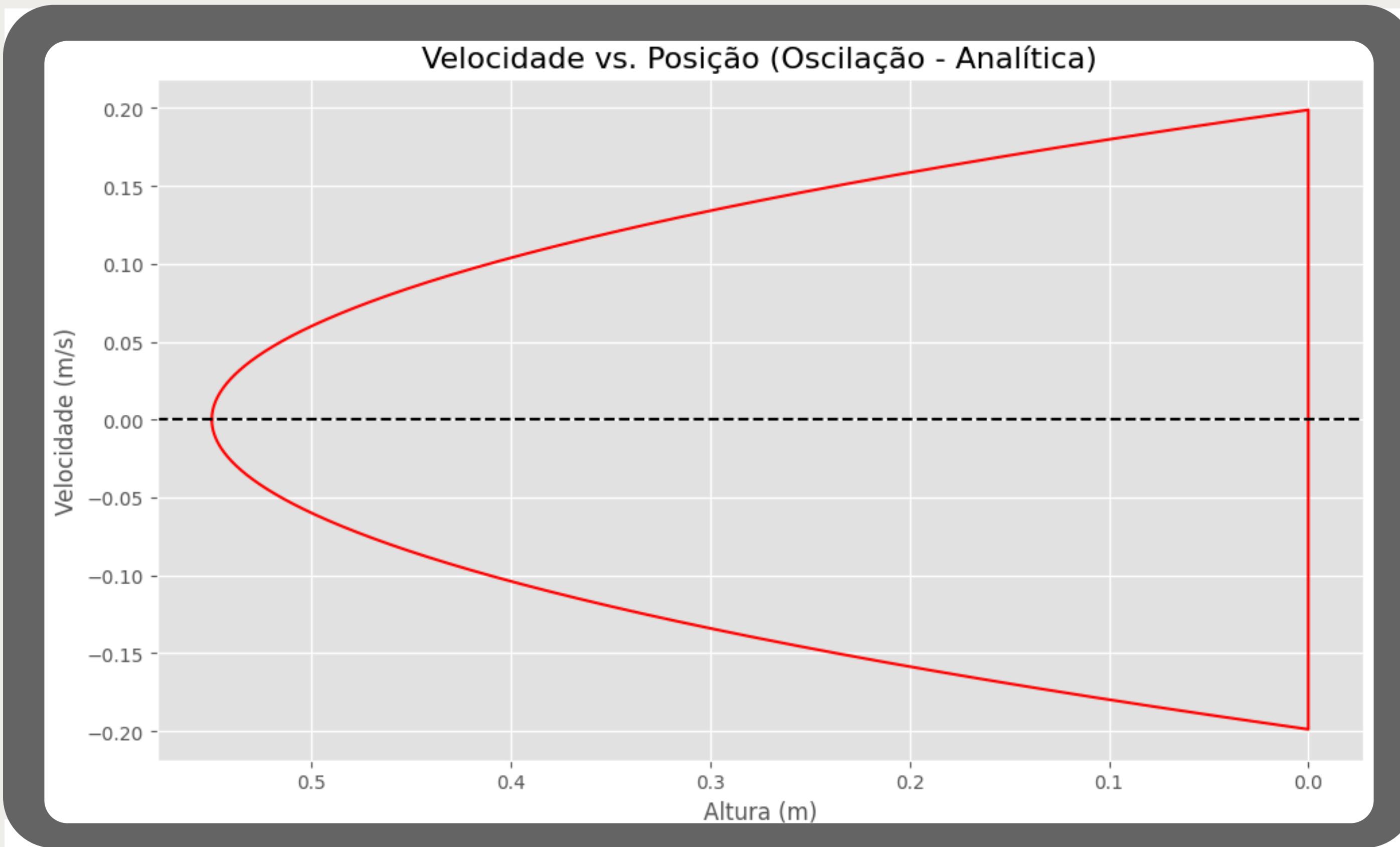


Gráfico da Velocidade vs Posição



Método de Euler-Cromer

```
# Pega o número de pontos do t_array (definido na célula analítica)
n_pontos = len(t_array)

# Cria arrays "vazios" para guardar os resultados da descida numérica
h_descida_numerico = np.zeros(n_pontos)
v_descida_numerico = np.zeros(n_pontos)

# Define as condições iniciais (t = 0)
h_descida_numerico[0] = H0_val
v_descida_numerico[0] = 0.0

# Loop de Integração (Descida)
# Este loop simula a física passo a passo
for i in range(n_pontos - 1):
    # Passo 1: v_novo = v_antigo + a*dt
    v_descida_numerico[i+1] = v_descida_numerico[i] + (a_eff_val * dt)

    # Passo 2: h_novo = h_antigo + v_novo*dt
    h_descida_numerico[i+1] = h_descida_numerico[i] + (v_descida_numerico[i+1] * dt)

# Correção: Garante que a altura pare em 0
h_descida_numerico[h_descida_numerico < 0] = 0

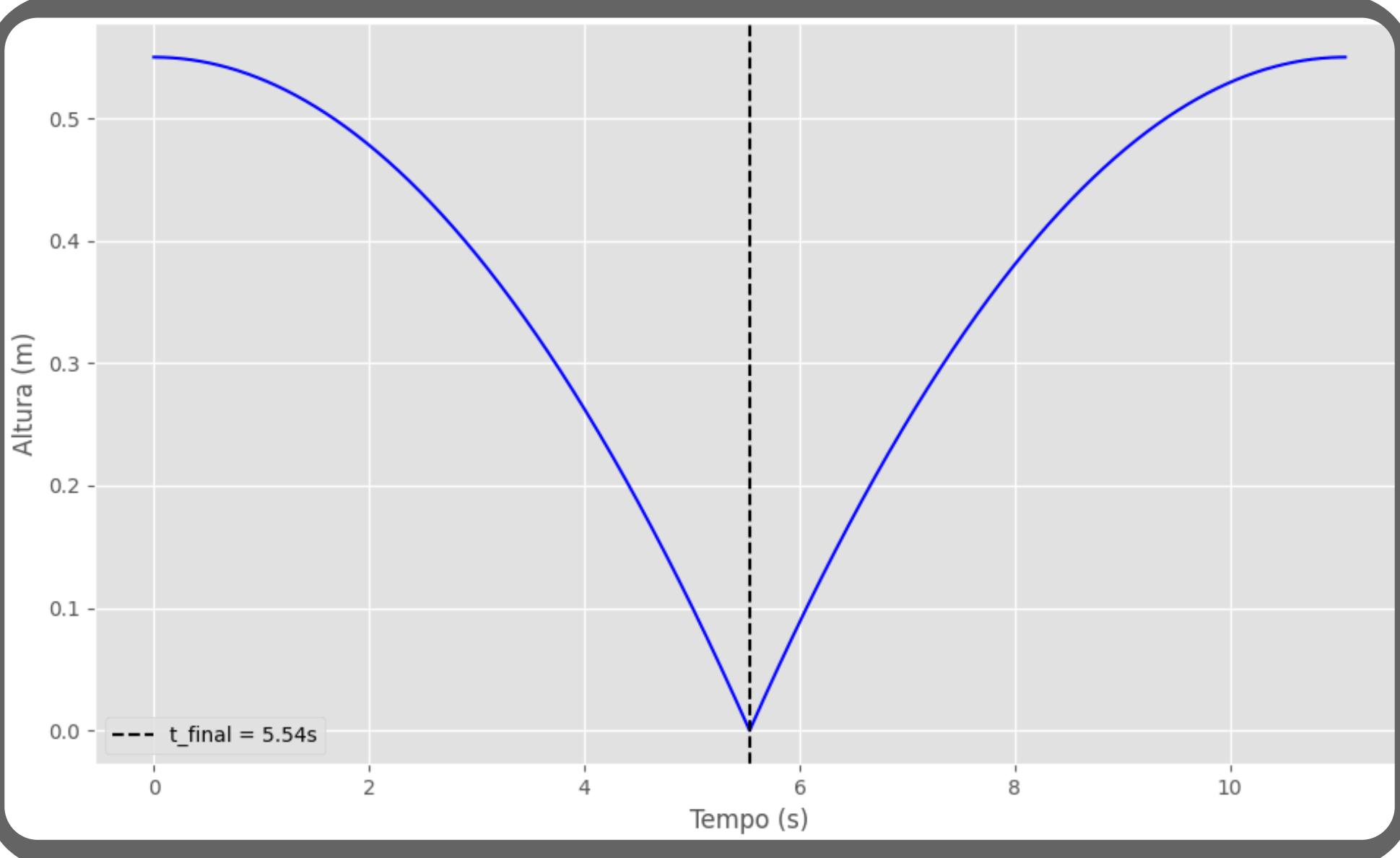
# Criar os dados da subida numérica
# Usamos [1:] para não duplicar o ponto do meio
h_subida_numerico = np.flip(h_descida_numerico)[1:]
v_subida_numerico = np.flip(v_descida_numerico)[1:] * -1

# Concatenar para criar os arrays do ciclo completo
# (0 t_oscilacao é o mesmo da célula analítica, podemos reusá-lo)
h_oscilacao_numerico = np.concatenate((h_descida_numerico, h_subida_numerico))
v_oscilacao_numerico = np.concatenate((v_descida_numerico, v_subida_numerico))
```

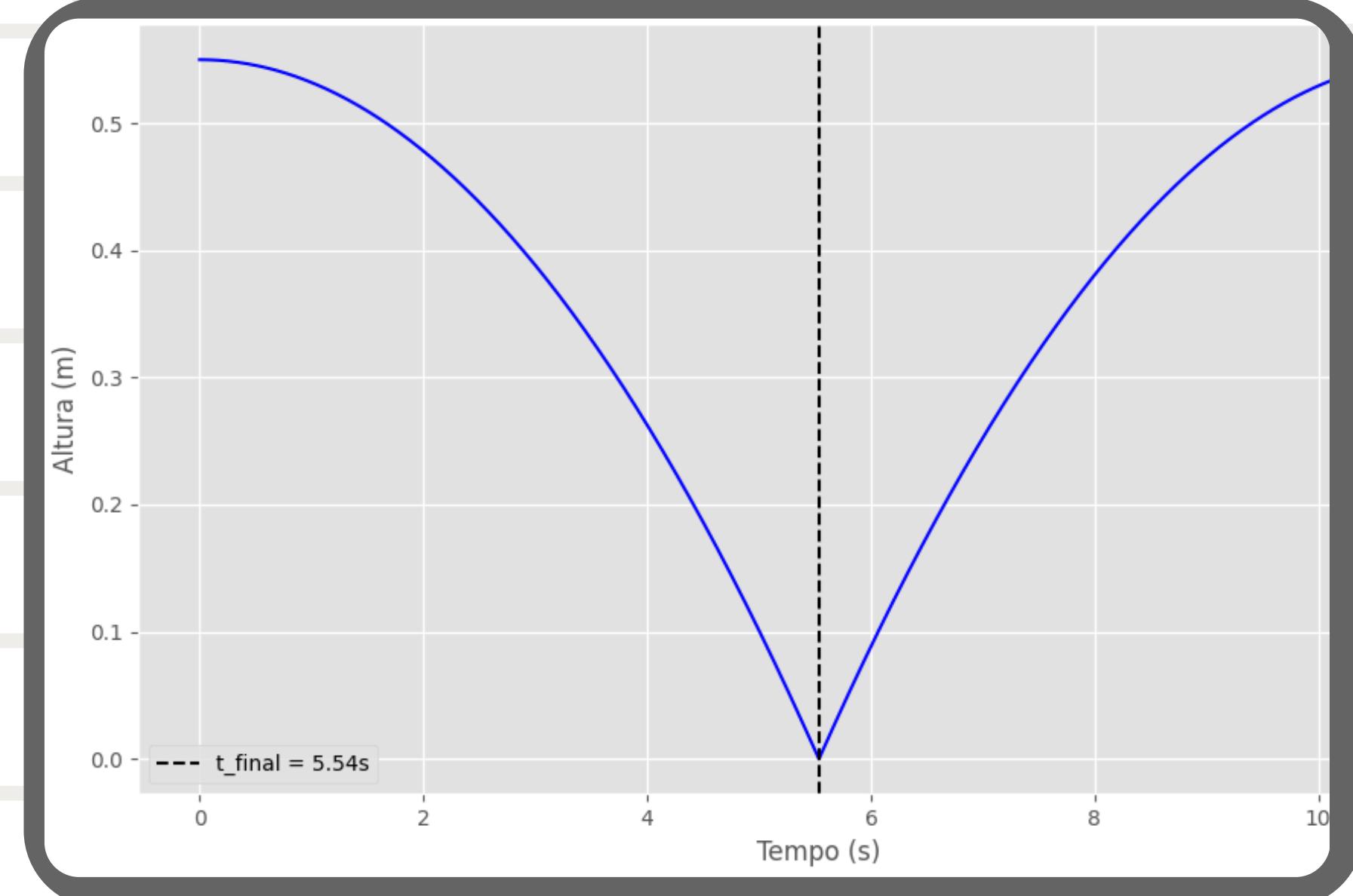
Utilizamos este método para justamente realizar as comparações com o nosso analítico e provar que resulta no mesmo.

Comparando os dois métodos temos:

Solução Analítica

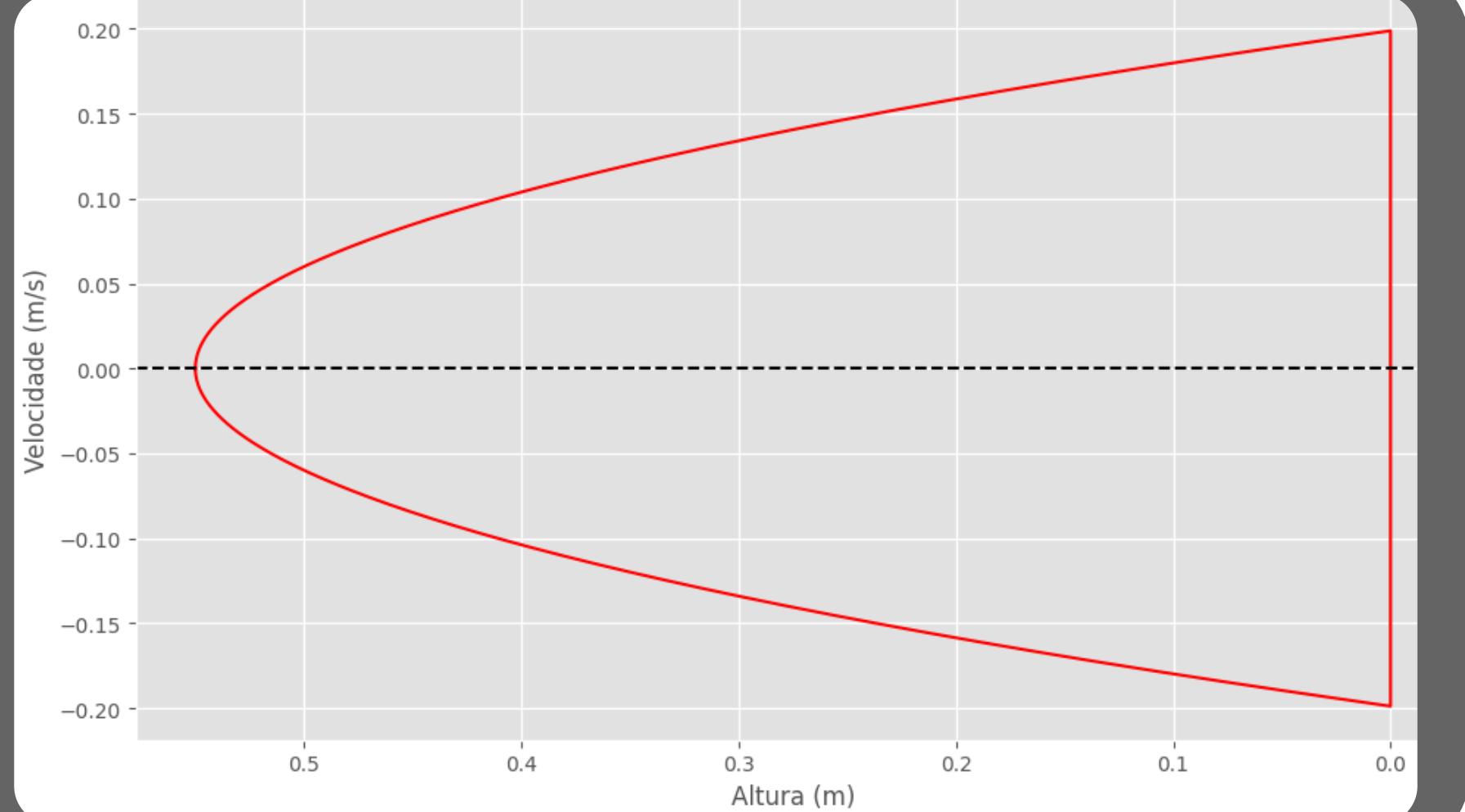


Método de Euler-Cromer

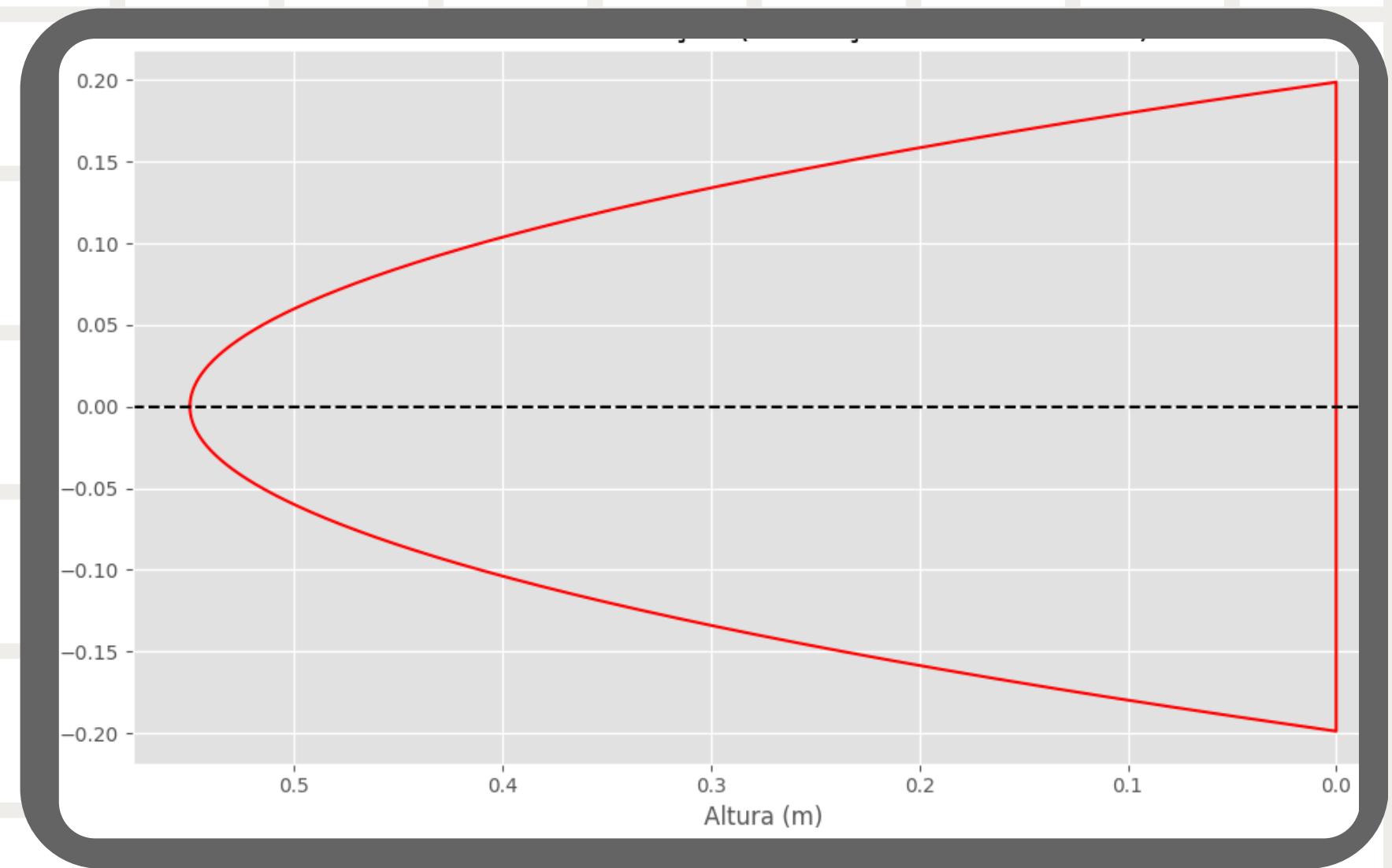


Comparando os dois métodos temos:

Solução Analítica



Método de Euler-Cromer



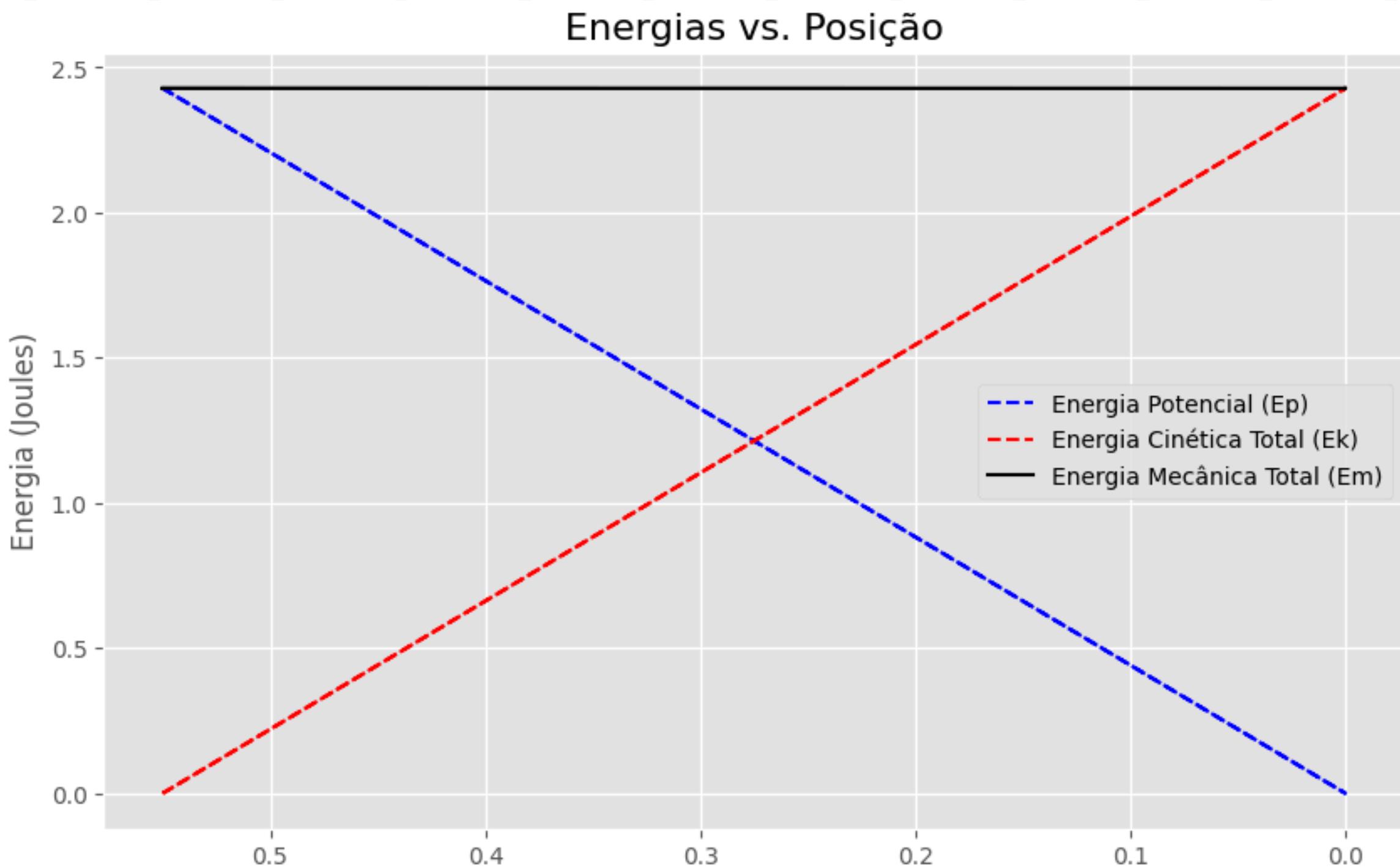
Conclusão comparando os métodos

Observamos que ambos os métodos dão os mesmos resultados de posição e velocidade, o que corrobora com a conclusão inicial da escolha do uso da solução analítica para o problema, que apesar de mais simples, pois apenas pegamos as fórmulas conhecidas que descrevem a velocidade e a posição na cinemática, é exatamente o suficiente para uma resposta satisfatória.

Gráfico das energias do sistema de acordo com a variação da altura.

- Podemos observar que a energia cinética inicia como zero Joules e vai aumentando.
- Já a energia potencial gravitacional começa em seu valor máximo, aproximadamente 2.4 Joules, e diminui até chegar em zero na altura zero.

Observamos também que, na altura máxima (repouso), a energia mecânica é igual a energia potencial gravitacional, e na altura zero (imediatamente antes do tranco), a energia mecânica é igual a energia cinética.



Comparação com os dados do artigo

O título do artigo que pegamos os dados é [Conservation of Energy-Maxwell's Wheel 1](#). Abaixo faremos a comparação entre os cálculos encontrados na nossa modelagem, chamaremos essa coluna de resultado, enquanto a outra será o esperado pelos dados do artigo, e por fim, a última será o erro, que é a diferença entre os dois

D (cm)	V resultado	V esperado	Erro aprox.
15	0,10	0,11	0,01
20	0,12	0,13	0,01
25	0,13	0,14	0,01
30	0,15	0,15	0,00
35	0,16	0,16	0,00
40	0,17	0,18	0,01
45	0,18	0,19	0,01
50	0,19	0,20	0,01
55	0,20	0,21	0,01

D (cm)	t resultado	t esperado	Erro aprox.
15	2,89	2,63	0,26
20	3,34	3,06	0,28
25	3,73	3,47	0,26
30	4,09	3,82	0,27
35	4,42	4,05	0,37
40	4,72	4,45	0,27
45	5,01	4,71	0,30
50	5,28	5,06	0,22
55	5,54	5,18	0,36

Comparação com os dados do artigo

D (cm)	Epot resultado	Epot esperada	Erro aprox. Epot	Ekin resultado	Ekin esperada	Erro aprox. Ekin
0	2,43	2,43	0,00	0,00	0,00	0,00
15	1,77	2,21	-0,44	0,66	0,67	-0,01
20	1,55	1,99	-0,45	0,88	0,93	-0,05
25	1,32	1,77	-0,45	1,10	1,1	0,00
30	1,10	1,55	-0,45	1,32	1,36	-0,04
35	0,88	1,32	-0,44	1,54	1,56	-0,02
40	0,66	1,1	-0,44	1,76	1,85	-0,09
45	0,44	0,88	-0,44	1,98	2,04	-0,06
50	0,22	0,66	-0,44	2,20	2,23	-0,03
55	0,00	0	0,00	2,42	2,45	-0,03

Fontes:

Artigos:

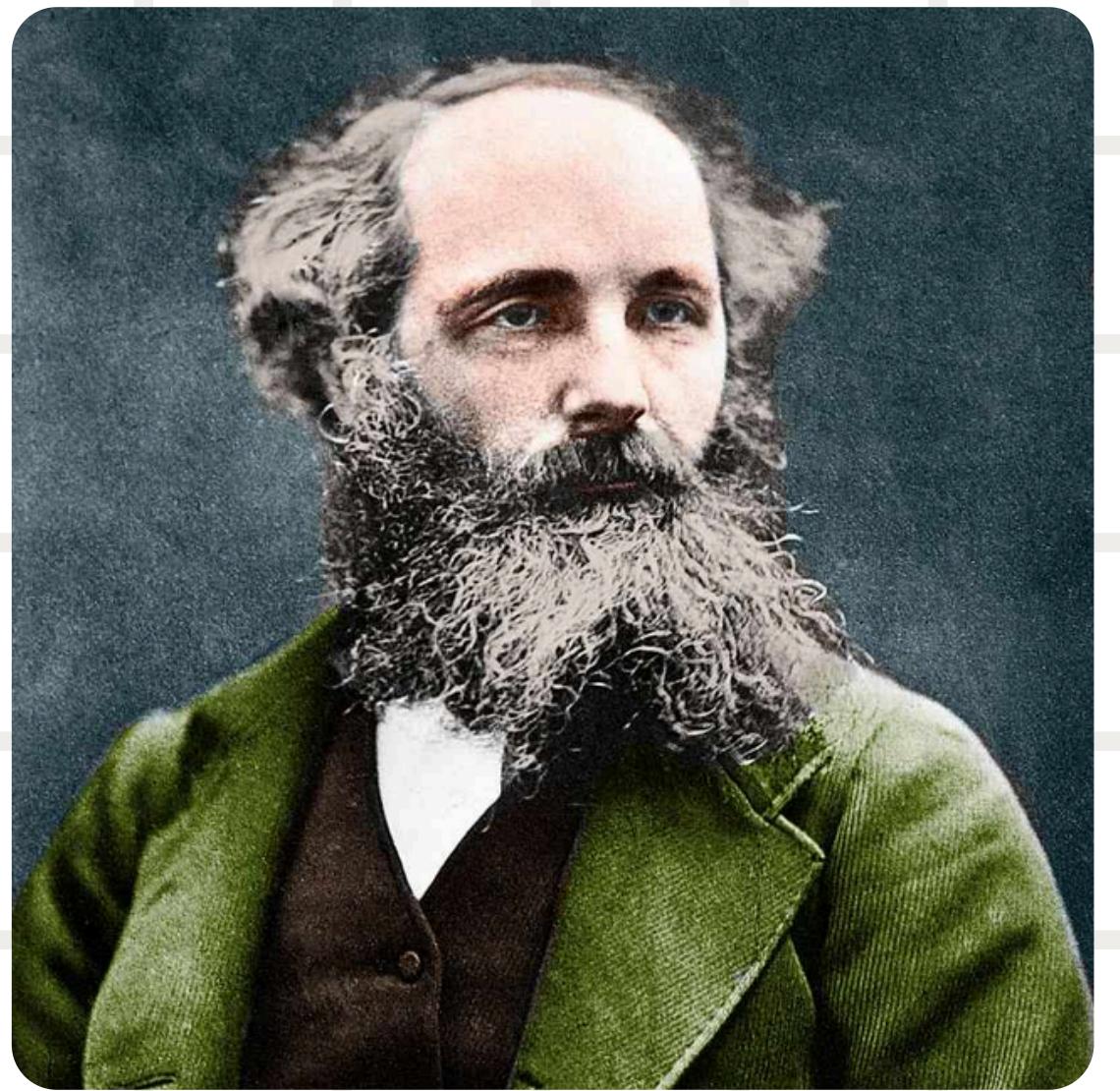
- Conservation of Energy-Maxwell's Wheel 1
- Mechanical energy conservation/Maxwell's wheel with measure Dynamics
- Maxwell's Wheel
- Analysis of the Motion of Maxwell's wheel Using Tracker Video Analysis and Measurement of the Moment of Inertia

Vídeos bases

- How does the Anti-Gravity Wheel work? Maxwell's Wheel
- Maxwell's Wheel
- Maxwell's Wheel science desk toy from United Scientific

Link para o Google Colab

- Trabalho ModMat



Fim