

# Modelagem e Resolução Numérica de Pêndulos Simples Amortecidos e Visualização Didática com Manim:

## Esboço

Hézio Delfino Lopes de Alcântara 231028343  
IF-UnB

(Data: 16 de dezembro de 2025)

### I. TESTANDO PARA DIFERENTES REGIMES DE AMORTECIMENTO

#### A. Regime Super-crítico

Para

$$\ddot{\theta} + 2\gamma\dot{\theta} + \omega_0^2\theta = 0, \quad \gamma > \omega_0, \quad (1)$$

a solução geral é

$$\theta(t) = C_1 e^{-(\gamma+\alpha)t} + C_2 e^{-(\gamma-\alpha)t}, \quad \alpha = \sqrt{\gamma^2 - \omega_0^2}. \quad (2)$$

As condições iniciais são

$$\theta(0) = \theta_0, \quad \dot{\theta}(0) = 0. \quad (3)$$

Resolvendo, obtemos o sistema

$$\begin{cases} C_1 + C_2 = \theta_0, \\ -(\gamma + \alpha)C_1 - (\gamma - \alpha)C_2 = 0. \end{cases} \quad (4)$$

Daí segue que

$$C_1 = \frac{\gamma - \alpha}{2\alpha} \theta_0, \quad C_2 = \frac{\gamma + \alpha}{2\alpha} \theta_0. \quad (5)$$

No código, ficamos com

```
theta0 = 0.1
gamma = 2.0
alpha = np.sqrt(gamma**2 - omega0**2)
C1 = (gamma - alpha)/(2*alpha) * theta0
C2 = (gamma + alpha)/(2*alpha) * theta0
theta_super = (C1 * np.exp(-(gamma + alpha)*t) +
               C2 * np.exp(-(gamma - alpha)*t))
```

#### B. Regime Crítico

$$\ddot{\theta} + 2\gamma\dot{\theta} + \omega_0^2\theta = 0, \quad \gamma = \omega_0 \quad (6)$$

A solução geral é:

$$\theta(t) = (C_1 + C_2 t) e^{-\gamma t} \quad (7)$$

Aplicando as condições iniciais desejadas, elas implicam em:

$$\theta(0) = \theta_0, \quad \dot{\theta}(0) = 0. \quad (8)$$

e

$$C_1 = \theta_0, \quad C_2 = \gamma\theta_0 \quad (9)$$

No código fica:

```
gamma_crit = omega0
x_crit = A * np.exp(-gamma_crit*t) * (1 + gamma_crit*t)
```

#### C. Regime Subcrítico

Por fim, para pequenas amplitudes:

$$\ddot{\theta} + 2\gamma\dot{\theta} + \omega_0^2\theta = 0 \quad \text{com} \quad 0 < \gamma < \omega_0 \quad (10)$$

No subcrítico

$$\gamma^2 - \omega_0^2 < 0 \quad (11)$$

Defina

$$\omega_d = \sqrt{\omega_0^2 - \gamma^2} \quad (12)$$

E então

$$r_{\pm} = -\gamma \pm i\omega_d \quad (13)$$

Cuja solução real é:

$$\theta(t) = e^{-\gamma t} (C_1 \cos(\omega_d t) + C_2 \sin(\omega_d t)) \quad (14)$$

Que possui forma equivalente a

$$\theta(t) = A e^{-\gamma t} \cos(\omega_d t + \phi) \quad (15)$$

No código fica:

```
gamma_sub = 0.05
omega_d = np.sqrt(omega0**2 - gamma_sub**2)
x_sub = A * np.exp(-gamma_sub*t) * np.cos(omega_d * t)
```

### D. Simulação com dados sintéticos e visualização do Espaço de Fase no *Manim*

Com o gráfico em mãos, é possível também extrair e organizar os dados correspondentes a cada regime de amortecimento em arquivos **.npz** distintos. Esses arquivos serão utilizados posteriormente em um código independente, desenvolvido em Python com a biblioteca *open-source* **Manim**, com o objetivo de simular e visualizar o espaço de fase do sistema.

A seguir, apresenta-se o código responsável pelo carregamento de um *dataset* e pelo salvamento separado dos dados referentes aos regimes subcrítico (ou subamortecido), crítico e supercrítico.

```

# ===== SALVANDO DATASETS =====
# Subcrítico
np.savez(
    "pendulo_subcritico.npz",
    t=t,
    x_clean=x_sub,
    x_noisy=x_noise0,
    A=A,
    omega0=omega0,
    gamma=gamma_sub
)

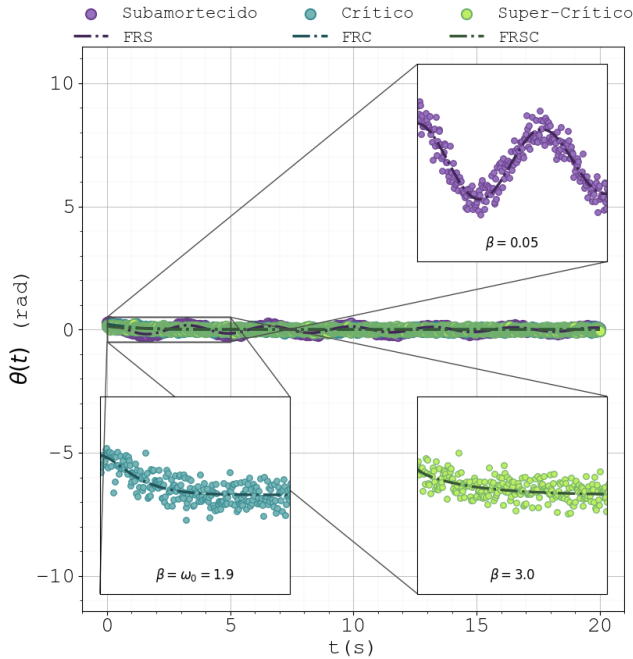
# Crítico
np.savez(
    "pendulo_critico.npz",
    t=t,
    x_clean=x_crit,
    x_noisy=x_noise2,
    A=A,
    omega0=omega0,
    gamma=gamma_crit
)

# Supercrítico
np.savez(
    "pendulo_supercritico.npz",
    t=t,
    x_clean=x_super,
    x_noisy=x_noise1,
    theta0=theta0,
    omega0=omega0,
    gamma=gamma_super
)

# ===== CARREGANDO DATASETS =====
data = np.load("pendulo_subcritico.npz")

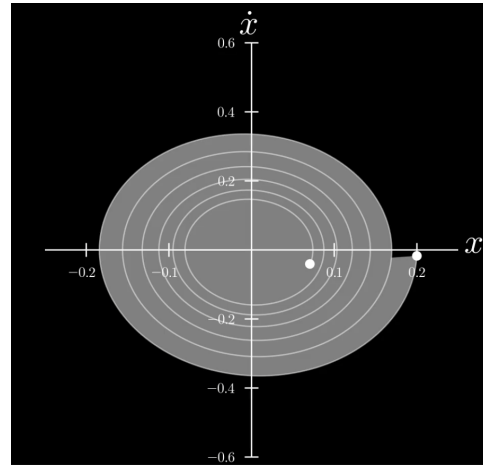
t = data["t"]
x = data["x_noisy"]
gamma = data["gamma"]

```

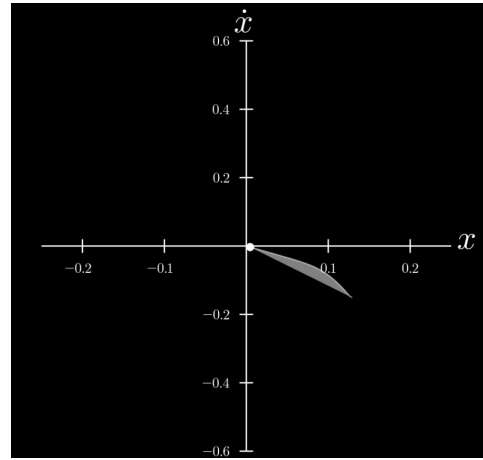


**Figura 1:** Gráfico mostrando os diferentes regimes com dados sintéticos gerados com ruído pseudoaleatório de  $seed = 0$  começando com amplitude inicial  $A_0 = 0.2\text{rad}$ .

**Nota:** Tanto no regime de amortecimento crítico quanto subcrítico, a ausência de oscilações faz com que a trajetória no espaço de fase não apresente uma estrutura geométrica bem definida. Além disso, a reconstrução da velocidade a partir da derivada discreta de  $x(t)$  no regime crítico torna-se numericamente instável para tempos longos, o que dificulta a obtenção de uma trajetória contínua e confiável no espaço de fase.



(a) Espaço de fase para regime de amortecimento Subcrítico ( $\beta = 0.05$ )



(b) Espaço de fase para regime de amortecimento Supercrítico ( $\beta = 3.0$ )