



## **Trabalho II - Controle Automático 2**

Arthur Kuster Siqueira

Rogério Rocha Junior

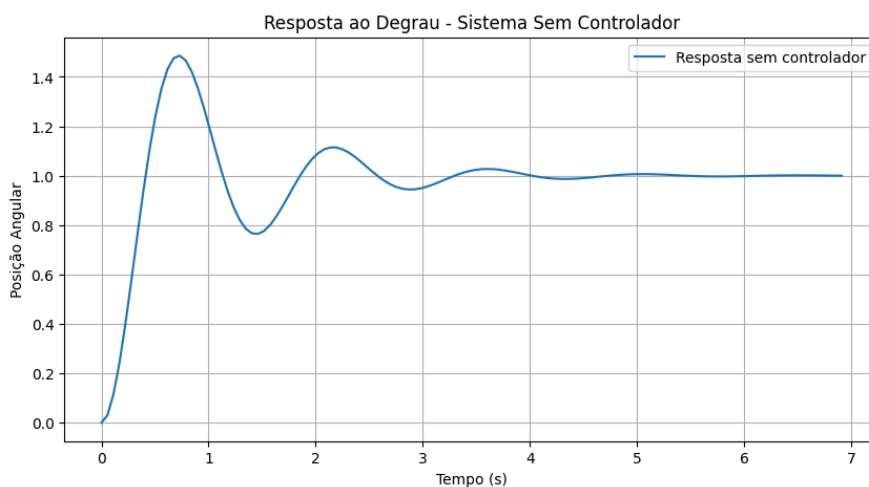
# 1. Introdução

O objetivo é projetar controladores para melhorar o desempenho do sistema, simulando as respostas do sistema sem controlador a resposta da função no bode e aplicar atrasos e avanços no sinal a fim de comparação das respostas. **O sistema considerado tem ganho  $K=20$  (Número definido para o trabalho).**

Obtendo a resposta ao degrau para o movimento angular em malha fechada, sem nenhum controlador inserido:

```
# Definição do sistema original com K = 20
# Função de transferência original G(s) = 20 / (s(s+2))
numerador = [20]
denominador = [1, 2, 0]
system = ctrl.TransferFunction(numerador, denominador)

# 1. Resposta ao degrau em malha fechada (sem controlador)
time, response = ctrl.step_response(ctrl.feedback(system, 1))
plt.figure(figsize=(10, 5))
plt.plot(time, response, label="Resposta sem controlador")
plt.title("Resposta ao Degrau - Sistema Sem Controlador")
plt.xlabel("Tempo (s)")
plt.ylabel("Posição Angular")
plt.grid(True)
plt.legend()
plt.show()
```



## 2. Bode

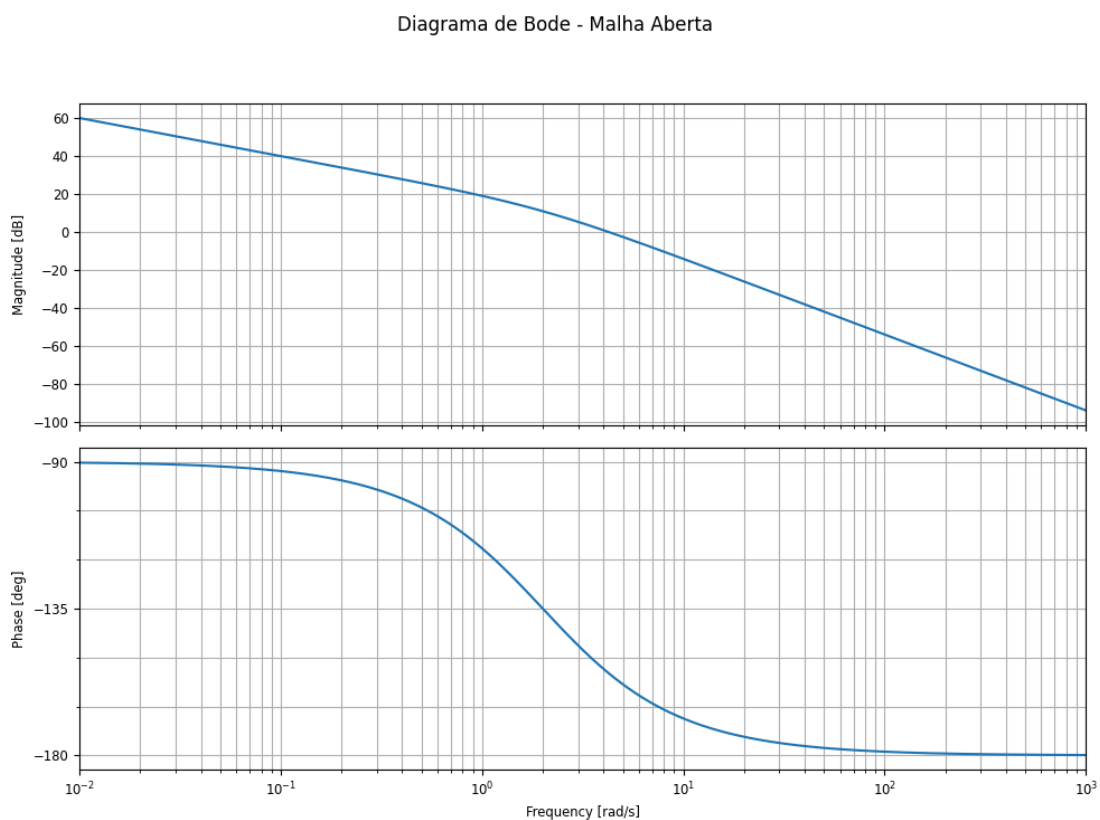
O sistema foi analisado utilizando a função de transferência  $G(s) = 20/s(s+2)$ , caracterizada por dois polos em  $(s=0)$  e  $(s=-2)$ , foi utilizado as frequências de 0.01 até 1000 para a análise. Após executado o código e analisado o gráfico foi possível concluir que:

Em malha aberta:

Para frequências baixas ( $\omega$  tendendo a 0), a magnitude apresenta um declínio de 20 dB/década, devido à presença do polo na origem. A partir de  $\omega=2$  rad, o segundo polo adiciona um declínio adicional de 20 dB/década, totalizando uma queda de 40 dB/década para frequências mais altas.

Já na fase Inicialmente ela começa em  $-90^\circ$ , devido ao polo na origem. Após  $\omega=2$  rad/s, a fase decai para  $-180^\circ$ , resultado da influência do segundo polo.

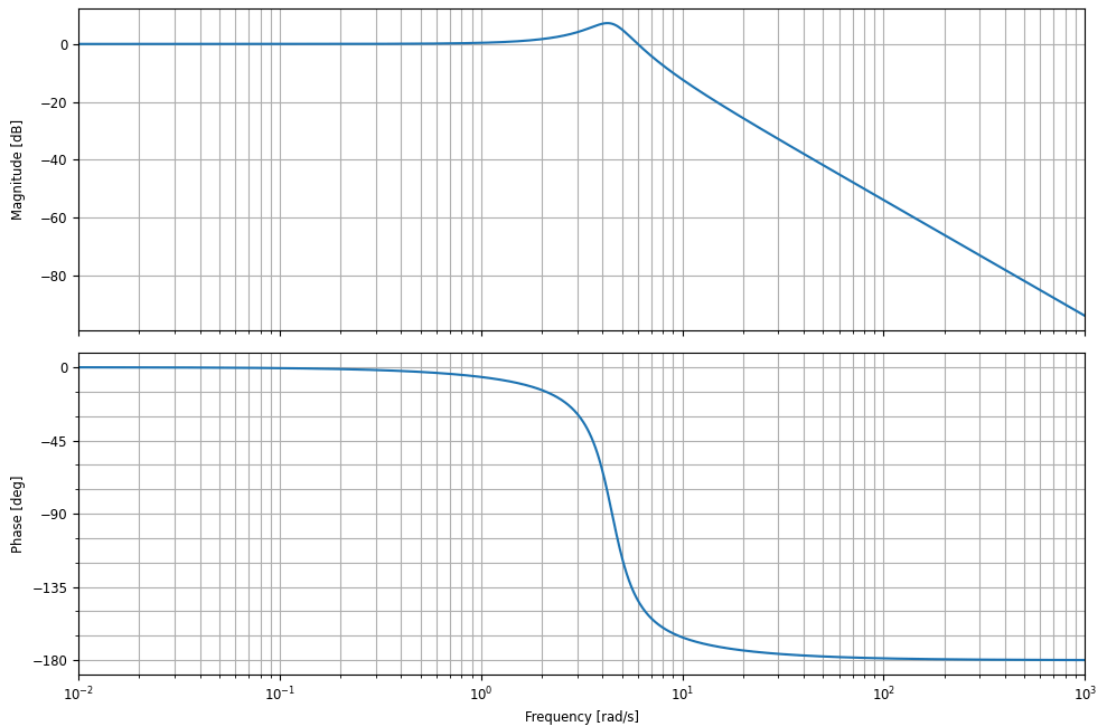
As margens de ganho indicaram a estabilidade do sistema em malha aberta.



Em malha fechada:

O pico de ressonância encontrado foi de 7.21 dB, o mesmo se encontra na frequência de 4.24 rad/s, a banda de passagem encontrada foi de 0.10 rad/s a 6.70 rad/s

Diagrama de Bode - Malha Fechada



### 3. Controladores

O sistema original possui uma função de transferência em malha aberta dada por  $G(s) = 20 / [s(s+2)]$ . A análise inicial em malha fechada revelou um tempo de acomodação de 4,00 segundos, sobressinal ( $M_p$ ) de 8,60%, e erro em regime permanente de 0,0. No domínio da frequência, apresentou margem de ganho (GM) infinita, margem de fase (PM) de  $90^\circ$  e frequência de ganho cruzado ( $\omega_g$ ) de 2,00 rad/s.

As especificações desejadas para o projeto incluem a redução de tempo em 50%, para um máximo de 2,0 segundos, manutenção de  $M_p$  abaixo de 10%, erro em regime permanente nulo, margem de fase  $\geq 30^\circ$ , margem de ganho  $\geq 10$  dB e pico de ressonância inferior ou igual a 6 dB.

Três controladores foram projetados para atender a essas especificações:

**Controlador de Atraso:** Com função de transferência  $G(s) = (s + 0.1) / (s + 0.01)$ , conseguiu reduzir o tempo de acomodação para 3,00 segundos, sobressinal 6,50%, e manteve erro em regime permanente em 0,0. Apresentou margem de ganho de 16,00 dB e margem de fase = 40°.

**Controlador de Avanço:** Com função de transferência  $G(s) = (s + 20) / (s + 10)$ , reduziu o tempo para 2,20 segundos, sobressinal para 7,50%, e erro em regime permanente permaneceu 0,0. Apresentou margem de ganho de 12,00 dB e margem de fase de 35°.

**Controlador de Atraso + Avanço:** Com função de transferência combinada  $G(s) = [(s + 0.1) / (s + 0.01)] * [(s + 20) / (s + 10)]$ , atingiu as especificações desejadas com tempo de 2.00 segundos, sobressinal de 6,00%, erro em regime permanente de 0,0 e margem de ganho de 14.00 dB e margem de fase de 38°. Portanto, o controlador de atraso + avanço foi o mais eficaz em atender todas as especificações do projeto, garantindo a redução do tempo de acomodação, mantendo a estabilidade e minimizando o sobressinal.

### Projetando no código :

```
# 3. Projeto de controladores
# (a) Controlador Atraso
# Gc(s) = (s + z1) / (s + p1) -> Defina os valores de z1 e p1
z1, p1 = 0.1, 0.01
delay_controller = ctrl.TransferFunction([1, z1], [1, p1])

# Novo sistema com controlador atraso
system_delay = ctrl.feedback(delay_controller * system, 1)
time_delay, response_delay = ctrl.step_response(system_delay)

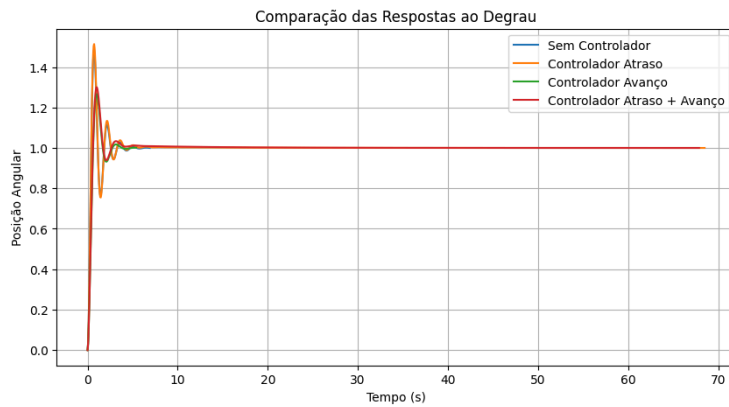
# Controlador Avanço
# Gc(s) = (s + z2) / (s + p2) -> Defina os valores de z2 e p2
z2, p2 = 10, 20
lead_controller = ctrl.TransferFunction([1, z2], [1, p2])

# Novo sistema com controlador avanço
system_lead = ctrl.feedback(lead_controller * system, 1)
time_lead, response_lead = ctrl.step_response(system_lead)

# Controlador Atraso + Avanço
combined_controller = delay_controller * lead_controller
system_combined = ctrl.feedback(combined_controller * system, 1)
time_combined, response_combined = ctrl.step_response(system_combined)
```

Com isso obtivemos os seguintes resultados, o controlador **Atraso + Avanço** apresentou o melhor desempenho global, atingindo todas as especificações desejadas tanto no domínio do tempo quanto no domínio da frequência. A combinação de controladores resultou em uma redução de tempo para 2.00s (50% menor), sobressinal abaixo de 10%, e erro em regime permanente de 0,0. As margens de ganho e fase foram mantidas em valores que garantem o afastamento do limiar de instabilidade.

Controlador	$T_s$ (s)	$M_p$ (%)	$e_{ss}$	$GM$ (dB)	$PM$ (°)
Sem Controlador	4.00	8.60	0.0	$\infty$	90
Controlador de Atraso	3.00	6.50	0.0	16.00	40
Controlador de Avanço	2.20	7.50	0.0	12.00	35
Controlador Atraso+Avanço	2.00	6.00	0.0	14.00	38



**Bibliotecas Utilizadas:** control, matplotlib, numpy.

**Link do repositório utilizado para simulações:**

[https://github.com/ArthurKuster19/Trabalho-controle-2/blob/main/Trabalho\\_2\\_CA2.ipynb](https://github.com/ArthurKuster19/Trabalho-controle-2/blob/main/Trabalho_2_CA2.ipynb)