

Controle PID de Vibrações em Sistemas de Segunda Ordem com Atraso Usando Receptância com Estabilidade Robusta e Otimização de Desempenho

Autor: Jhonat Heberon Avelino de Souza ¹

Orientador: Prof. Dr. Carlos Eduardo Trabuco Dórea ²

Coorientador: Prof. Dr. José Mário Araújo ³

¹jhonatheberon@gmail.com

²cetdorea@dca.ufrn.br

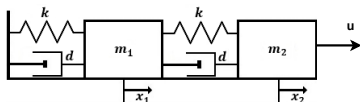
³prof.jomario@gmail.com

Programa de Pós-graduação em Engenharia Mecatrônica – UFRN

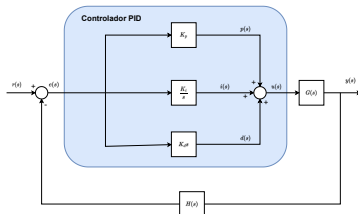
Sumário

- 1 Introdução
- 2 Fundamentação Teórica
- 3 Definição do Problema
- 4 Metodologia
- 5 Experimentos
- 6 Resultados
- 7 Conclusões

Figura 1: Sistema de Segunda ordem, e Controlador PID



(a) Sistema massa e mola



(b) Controlador PID

Fonte: o autor (2023)

Introdução - Trabalhos anteriores do grupo

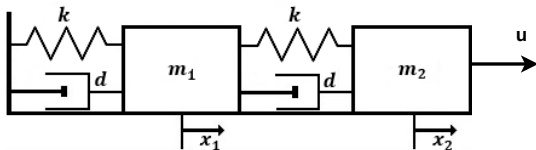
- Problema com realimentação de estado (DANTAS, 2019)
 $u(t) = f^T \dot{x}(t) + g^T x(t)$
- Alocação parcial de polos (DANTAS, 2019)
- Monovariável (DANTAS, 2019)
- Sistemas estáveis em malha aberta (DANTAS, 2019)
- Problema com realimentação derivativa (NUNES et al, 2021)
 $u(t) = f^T \ddot{x}(t) + g^T \dot{x}(t)$

Contribuímos com utilização do controlador Proporcional e Integrativo Derivativo (PID) com realimentação de saída para rastreamento e considerar sistemas estáveis e **instáveis** em malha aberta, otimizando índice *IAE* e robustez de forma **concorrente**.

Introdução - Objetivos

- Estudo de sistemas de segunda ordem com atraso.
- Propor uma solução para um problema de controle para um sistema de segunda ordem com atraso
- Controlador Proporcional Integrativo e Derivativo (PID).
- Definir o problema de controle nos termos da resposta em frequência usando Receptância
- Critério de estabilidade de *Nyquist* para definir a função as regras de otimização.
- Otimização do controlador com base no índice de *IAE*, concorrente ao critério de robustez.
- Aprimorar algoritmo heurístico de otimização (GA) para encontrar os ganhos do controlador PID que atenda os critérios estabelecidos.
- Publicação do artigo no SBAI

Figura 2: Sistema massa e mola



Fonte: DANTAS (2019)

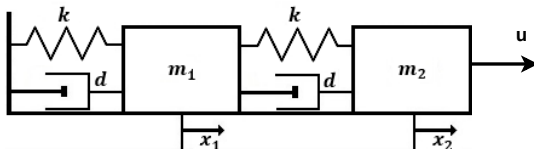
Equação do sistema

$$m_1 \ddot{x}_1(t) + d(2\dot{x}_1(t) - \dot{x}_2(t)) + k(2x_1(t) - x_2(t)) = 0 \quad (1)$$

$$m_2 \ddot{x}_2(t) + d(\dot{x}_2(t) - \dot{x}_1(t)) + k(x_2(t) - x_1(t)) = u \quad (2)$$

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} 2d & -d \\ -d & d \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} 2k & -k \\ -k & k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (3)$$

Figura 3: Sistema massa e mola



Fonte: DANTAS (2019)

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{b}u(t) \quad (4)$$

- Para um sistema com n graus de liberdade e m entradas, têm-se $\mathbf{M} \in \mathbb{R}^{n \times n}$, é uma matriz de massas, $\mathbf{C} \in \mathbb{R}^{n \times n}$ é uma matriz de amortecimento, $\mathbf{K} \in \mathbb{R}^{n \times n}$ é uma matriz de rigidez, $\mathbf{B} \in \mathbb{R}^{n \times m}$ é uma matriz de controle, $\mathbf{x} \in \mathbb{R}^n$ é o vetor de deslocamento e $\mathbf{u} \in \mathbb{R}^m$ é um vetor de entradas Single Input Single Output (SISO)

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{b}u(t) \quad (5)$$

Aplicando agora a transformada de Laplace a (5) obtém-se:

$$\mathbf{X}(s) = [\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K}]^{-1}\mathbf{b}\mathbf{U}(s) = \mathbf{H}(s)\mathbf{U}(s), \quad (6)$$

A saída do sistema definida como:

$$y(t) = \mathbf{l}\mathbf{x}(t) \quad (7)$$

em que $\mathbf{l} \in \mathbb{R}^{1 \times n}$ é uma matriz de composição de sensores.

Para uma dada referência $r(t)$, o erro de rastreamento é definido por:

$$e(t) = r(t) - y(t) \quad (8)$$

$$u(t) = k_p e(t - \tau) + k_i \int_0^t e(t - \tau) d\tau + k_d \frac{de(t - \tau)}{dt} \quad (9)$$

Fundamentação Teórica - Matriz de receptância

Aplicando a transformada de Laplace em (7), (8) e (9) obtém-se:

$$U(s) = -\left(k_p + \frac{k_i}{s} + k_d s\right)e^{-\tau s} \mathbf{I} \mathbf{X}(s) + V(s), \quad (10)$$

em que $q(s) = \left(k_p + \frac{k_i}{s} + k_d s\right)$ e $V(s) = q(s)R(s)$.

Da substituição de (10) em (5) resulta:

$$[\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K} + e^{-\tau s} q(s) \mathbf{b} \mathbf{l}] \mathbf{X}(s) = \mathbf{b} V(s). \quad (11)$$

Logo,

$$\mathbf{X}(s) = [\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K} + e^{-\tau s} q(s) \mathbf{b} \mathbf{l}]^{-1} \mathbf{b} V(s) = \hat{\mathbf{H}}(s) \mathbf{V}(s) \quad (12)$$

$$(\mathbf{A} + \mathbf{u}\mathbf{v}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}} \quad (13)$$

Aplicando a fórmula de *Sherman-Morrison* na inversa presente em (12), obtém-se a matriz de receptância de malha fechada do sistema, dada por:

$$\hat{\mathbf{H}}(s) = \mathbf{H}(s) - \frac{(k_p + \frac{k_i}{s} + k_d s)\mathbf{H}(s)\mathbf{b}\mathbf{I}\mathbf{H}(s)e^{-\tau s}}{1 + (k_p + \frac{k_i}{s} + k_d s)\mathbf{I}\mathbf{H}(s)\mathbf{b}e^{-\tau s}} \quad (14)$$

na qual $\hat{\mathbf{H}}(s) = (\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K} + q(s)\mathbf{b}\mathbf{I})^{-1}\mathbf{b}$ é definida como matriz de receptância de malha fechada e $\mathbf{H}(s) = (\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K})^{-1}\mathbf{b}$ como matriz de receptância de malha aberta, que, na prática, pode ser medida pela resposta em frequência $\mathbf{H}(j\omega)$.

A equação característica de (14) é definida como:

$$1 + (k_p + \frac{k_i}{s} + k_d s)\mathbf{I}\mathbf{H}(s)\mathbf{b}e^{-\tau s} = 0 \quad (15)$$

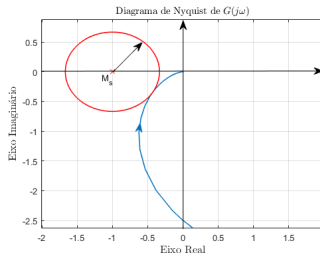
Resposta em frequência

- A matriz de receptância nos fornece a resposta em frequência $H(j\omega)$ do sistema
- O ganho de malha da equação característica em malha fechada é descrito na Equação 16.

$$L(s) = (k_p + \frac{k_i}{s} + k_d s) \mathbf{IH}(s) \mathbf{b} e^{-\tau s} \quad (16)$$

Definição do Problema - Estabilidade

Figura 4: Diagrama de Nyquist Exemplo de Circunferência. M_s



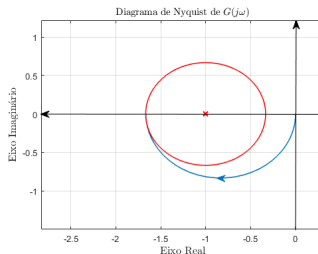
Fonte: DANTAS (2019)

Restrições de estabilidade

- Podemos afirmar pelo critério de estabilidade de Nyquist que a circunferência M_s representa quanto robusto é o sistema
- M_s é a menor distância entre o ponto $(-1, 0)$ e a curva de Nyquist

Definição do Problema - Instabilidade

Figura 5: Diagrama de Nyquist de um Sistema Genérico para Exemplo de um Caso de Instabilidade.



Fonte: DANTAS (2019)

Restrições de instabilidade

- Garantir que a curva de Nyquist não contenha a circunferência a englobando
- Enlaçar o ponto $(-1, 0)$, se evidencia um caso de instabilidade

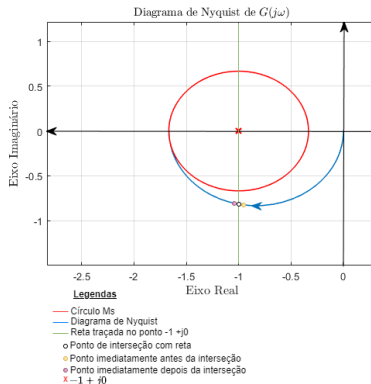
Definição do Problema - Instabilidade

$$N = -P \quad (17)$$

- Neste trabalho também aplicaremos os métodos testado para sistema de controle estáveis para sistemas instáveis
- Quando temos um sistema instável significa que temos raízes no SPD
- P o número de polos de malha aberta no semiplano direito
- N o número de voltas que o diagrama de *Nyquist* faz em torno do ponto crítico $-1 + j0$ no sentido horário.
- O diagrama de *Nyquist* deve dar tantas voltas no sentido anti-horário em torno de $-1 + j0$ quanto seja o número de polos de malha aberta no semiplano direito.

Definição do Problema - Instabilidade

Figura 6: Exemplificação da contagem de voltas que contem o ponto $-1 + j0$ e o sentido do diagrama de *Nyquist*



Fonte: o autor (2023)

Definição do Problema - Funções de otimização

Seguinte problema de otimização pode ser formulado semelhante ao proposto em (DANTAS, 2019):

$$\min_{k_p, k_i, k_d} = \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \quad (18)$$

$$s.a \text{ Re}\{L(j\omega)\} \geq -1 + M_s^{-1} \quad \forall \quad \omega / \text{Im}\{L(j\omega)\} = 0 \quad (19)$$

$$L(j\omega) = \left(k_p + \frac{k_i}{s} + k_d s\right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau}$$
$$N = -P \quad (20)$$

Definição do Problema - Funções de otimização

$$\min_{k_p, k_i, k_d} = \int_0^t |e(t)| dt \quad (21)$$

$$s.a \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \leq \epsilon \quad (22)$$

$$\begin{aligned} L(j\omega) &= \left(k_p + \frac{k_i}{s} + k_d s \right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau} \\ N &= -P \end{aligned} \quad (23)$$

Definição do Problema - Funções de otimização

$$\min_{k_p, k_i, k_d} (1 - \alpha) * \int_0^t |e(t)| dt + \alpha \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \quad \forall \quad \alpha \in \{0, 1\} \quad (24)$$

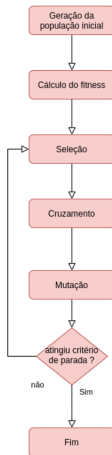
$$s.a \quad \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \leq \epsilon \quad (25)$$

$$L(j\omega) = \left(k_p + \frac{k_i}{s} + k_d s \right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau}$$
$$N = -P \quad (26)$$

Busca dos ganhos do controlador

- Uso de meta-heurística para encontrar os ganhos
- Algoritmo Genético
- Resolve de forma genérica problemas de otimização
- Geralmente aplicadas a problemas para os quais não se conhece algoritmo eficiente

Figura 7: Fluxograma do algoritmo genético



Fonte: o autor (2023)

Figura 8: Pseudo código do algoritmo de busca

Algoritmo 1: Função de Busca

Entrada: $M, C, K, B, I, t, \omega, M_s$ e *simulink*

Saída: $gain = [K_p \ K_I \ K_d]$

```
1 populacao = random(n);
2 objetivo = "execute";
3 execucao = 0;
4 variabilidade = 1;
5 evolucao = 0;
6 alfa = random(0,1);
7 n = 100;
8 repita
9   para geracao=1 até geracao=n faça
10     avaliacao = []
11     se execucao > 0 então
12       populacao(2:n,:) = random(n-1);
13     fim
14     para i = 1 até populacao=n faça
15       L ← LFunction(M, C, K, B, I, t, ω, populacao(i, 1:3));
16       robustez ← RobustezFunction(L(s), Ms);
17       restrictions ← RestrictionsFunction(L(s), 0);
18       IAE ← IaeFunction(populacao, simulink);
19       ff ← FitnessFunction(IAE, robustez, alfa);
20       avaliacao(i,1:end) = [ff, restrictions];
21       i = i + 1;
22     fim
23     populacao = [populacao avaliacao];
24     populacao ← AssortmentFunction(populacao);
25     populacao ← CrossOverFunction(populacao(1:n,1:end-2));
26     L ← LFunction(M, C, K, B, I, t, ω, populacao(1, 1:3));
27     robustez ← RobustezFunction(L(s), Ms);
28     restrictions ← RestrictionsFunction(L(s), 0);
29     IAE ← IaeFunction(populacao, simulink);
30     ff ← FitnessFunction(IAE, robustez, alfa);
31     se (ff < 0.4 && restrictions < 0.9) || (execucao >
32       2 && Restrictions < 0.9) então
33       objetivo = "fim";
34       melhorIndividuo = populacao(i,:);
35       interromper;
36     fim
37     VariabilidadeFunction(variabilidade, evolucao, ff);
38     geracao = geracao + 1;
39   fim
40 execucao = execucao + 1;
41 até (objetivo ≠ "fim");
```

Fonte: o autor (2023)

Figura 9: Pseudo código da função L

Algoritmo 3: Função L

Entrada: M; C; K; B; I; τ ; ω e populacao

Saída: L

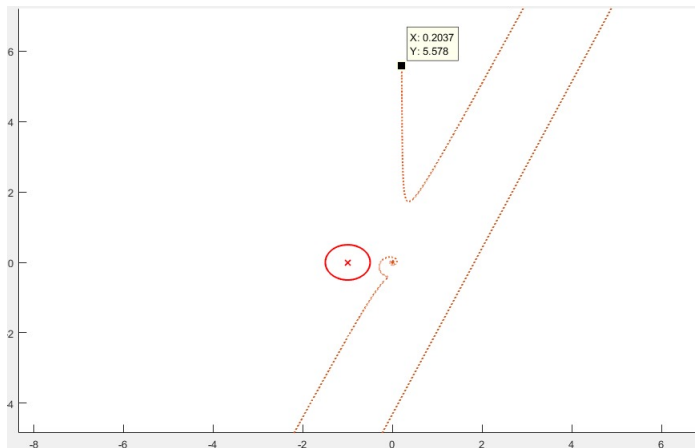
```
1 Kp = populacao(1);
2 Ki = populacao(2);
3 Kd = populacao(3);
4 para i = 1 até i = comprimento( $\omega$ ) faça
5    $\beta = j \times \omega(i)$ ;
6    $L(i) = (Kp + Ki/\beta + 0.0001) + \beta \times Kd) \times I \times (M \times \beta^2 + C \times \beta + K)^{-1} \times B \times e^{-\tau \times \beta}$ ;
7   i = i + 1;
8 fim
```

Fonte: o autor (2023)

Nyquist infinito

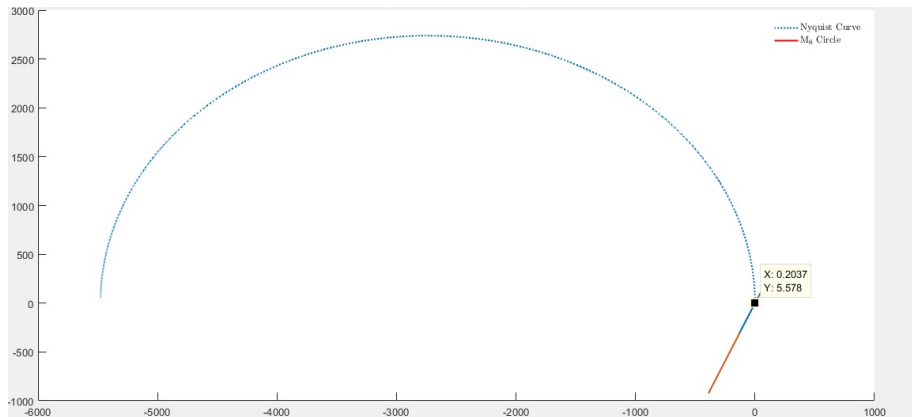
- Perturbação na ação integrativa
- Evita valores infinitos para cálculo do diagrama de Nyquist

Figura 10: Diagrama de *Nyquist* para $\omega \in [10^{-1}, 10^3]$



Fonte: o autor (2023)

Figura 11: Diagrama de *Nyquist* para $\omega \in [10^{-6}, 10^3]$



Fonte: o autor (2023)

Figura 12: Pseudo código da função de variabilidade

Algoritmo 8: Função de Variabilidade

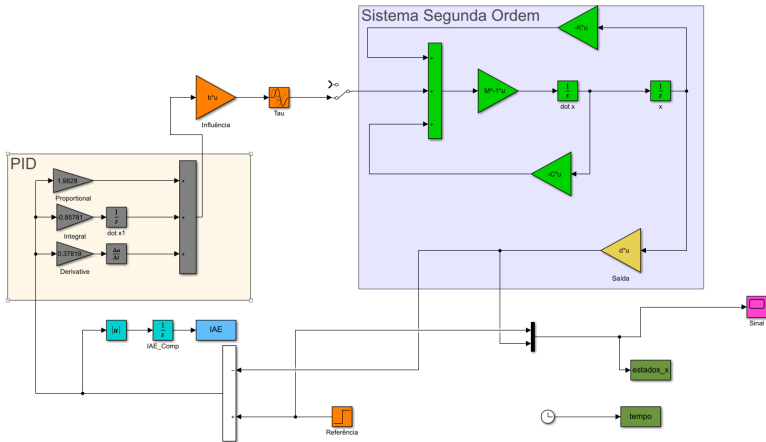
```
Entrada: variabilidade, evolucao, e ff
1 evolucao(variabilidade) = ff;
2 variabilidade = variabilidade + 1;
3 se tamanho(evolucao) > 10 então
4     variabilidade = 1;
5     evolucao = evolucao(2:end);
6     se var(evaluation) < 0.0000001 então
7         objetivo = "fim";
8         melhorIndividuo = populacao(1,:);
9         interromper;
10    fim
11 fim
```

Fonte: o autor (2023)

Verificação de variabilidade genética

- Análise de melhoria da população com base na interação do algoritmo

Figura 13: Simulação criada com MATLAB/Simulink®



Fonte: o autor (2023)

Experimentos - Exemplo 2

Considerando o método apresentado em RAM et al. (2011).

$$\mathbf{M} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & -0,9999 \\ -0,9999 & 1 \end{bmatrix} \mathbf{K} = \begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix}$$

- $\mathbf{B}' = \mathbf{I}' = [0 \ 1]$
- $\tau = 5s$
- $M_s = 1,6667$

Resultados da Otimização - Exemplo 2

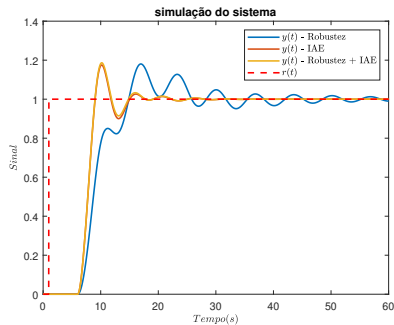
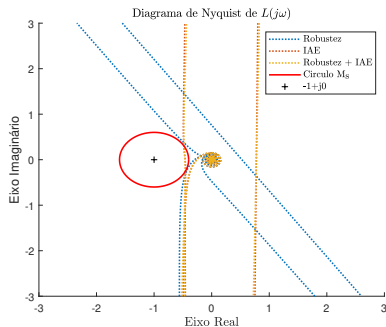
No caso do exemplo 2, por se tratar de um sistema mais complexo e, teoricamente, com maior dificuldade na sintonia dos ganhos do controlador, esses ganhos estão apresentados na Tabela 1,

Tabela 1: Ganhos ótimos e índice de desempenho para o Exemplo 2

Sintonia	k_p	k_i	k_d	IAE
Robustez + IAE	0,5145	0,2578	0,4004	7,3408
Robustez	0,1773	0,2229	0,4432	10,0418

Resultados da Otimização - Exemplo 2

Figura 14: Resultado exemplo 2



(a) Comparação do diagrama de *Nyquist* do Exemplo 2 (b) Comparação da simulação do Sistema do Exemplo 2

Fonte: o autor (2023)

Experimentos - Exemplo 3

Neste caso específico, foi feita uma pequena modificação do exemplo 2, com o objetivo de introduzir uma perturbação no modelo.

$$\mathbf{M} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & -0.9 \\ -1 & 1 \end{bmatrix}, \mathbf{K} = \begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix}$$

- $\mathbf{B}' = \mathbf{I}' = [0 \ 1]$
- $\tau = 0,5$

Resultados da Otimização - Exemplo 3

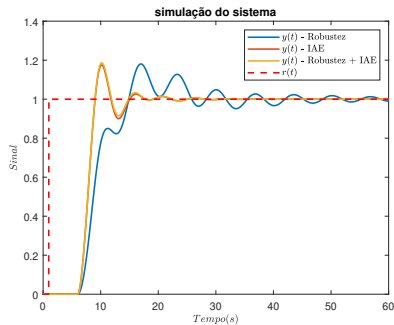
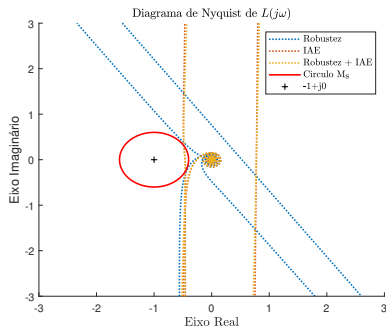
No exemplo 3, os ganhos e índices são evidenciados na Tabela 2.

Tabela 2: Ganhos ótimos e índice de desempenho para o Exemplo 3

Sintonia	k_p	k_i	k_d	IAE
Robustez + IAE	0,4373	0,7548	1,8163	2,4390
Robustez	0,2399	1,0990	1,6015	6,4692

Resultados da Otimização - Exemplo 3

Figura 15: Resultado exemplo 3



(a) Comparação do diagrama de *Nyquist* do Exemplo 3 (b) Comparação da simulação do Sistema do Exemplo 3

Fonte: o autor (2023)

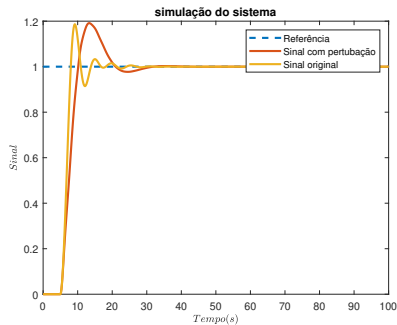
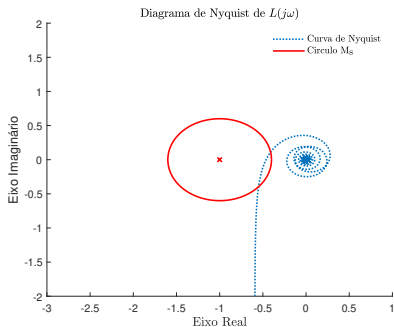
Resultados da perturbação no sistema - Exemplo 3

Simularemos uma perturbação do modelo, com objetivo de verificar quão os sistemas são robustos para o método proposto neste trabalho.

Para realizar a simulação da perturbação, analisaremos o sistema do exemplo 2, com ganhos obtidos da simulação para este exemplo, executaremos a simulação do exemplo 3, o qual representa uma perturbação no modelo, devido à variação da Matriz **C**, de um exemplo para outro.

Resultados da perturbação no sistema - Exemplo 3

Figura 16: Resultado otimizando concorrência entre IAE e robustez



(a) Diagrama de *Nyquist* com perturbação no sistema (b) Simulação do Sistema com perturbação no sistema

Fonte: o autor (2023)

Experimentos - Exemplo 5

Este exemplo, adaptado de SHAPIRO (2005), aborda um caso instável em malha aberta, cujas matrizes são dadas por:

$$\mathbf{M} = \begin{bmatrix} 10 & 0 \\ 0 & 11 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 4 & 1 \\ 1 & 5 \end{bmatrix}, \mathbf{K} = \begin{bmatrix} 8 & 4 \\ -4 & 9 \end{bmatrix}$$

- $\mathbf{b}^T = [-1 \ 1]$ e $\mathbf{l} = [1 \ 0]$
- $\tau = 0.5\text{s}$
- $M_s = 1,6667$
- $s_{1,2} = 0,0039 \pm j0,9$

Resultados da Otimização - Exemplo 5

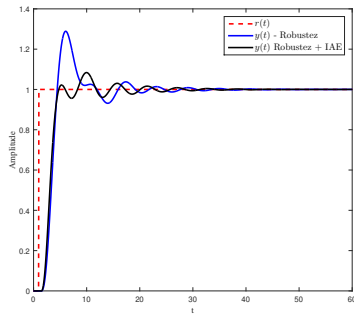
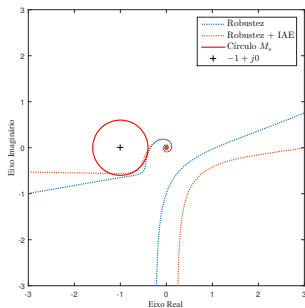
- A Tabela 3 permite a comparação dos controladores projetados, e é possível notar um desempenho melhorado em relação ao *IAE* com a aplicação do método proposto.
- A estabilidade em malha fechada pode ser verificada no diagrama de *Nyquist* completo ($\omega \in \mathbb{R}$) com visão expandida, onde são constatados os dois envoltimentos requeridos do ponto crítico $-1 + j0$.

Tabela 3: Ganhos ótimos e índice de desempenho para o Exemplo 5

Sintonia	k_p	k_i	k_d	<i>IAE</i>
Robustez + <i>IAE</i>	-4,4787	-3,3738	-11,1194	2,6374
Robustez	-2,1975	-4,3664	-10,8658	3,5080

Resultados da Otimização - Exemplo 5

Figura 17: Resultado exemplo 5



(a) Comparação do diagrama de *Nyquist* do Exemplo 5 (b) Comparação da simulação do Sistema do Exemplo 5

Fonte: o autor (2023)

- PID com realimentação de saída para rastreamento.
- Estudo da eficiência para essa técnica utilizando controlador PID.
- Avaliação do critério de estabilidade de *Nyquist*.
- Avaliação do robustez do sistema como parametro de busca.
- Avaliação do índice *IAE* como parametro de busca.
- Estudo da concorrência entre a robustez do sistema e índice *IAE*.
- Avaliação da função *Fitness* desenvolvida que considera concorrência entre os parâmetros de busca.
- Trabalhos futuros: Verificar eficiência de outro algoritmo heurístico (PSO).

Obrigado!