

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

Jhonat Heberson Avelino de Souza ¹

Prof. Dr. Carlos Eduardo Trabuco Dórea ²

Prof. Dr. José Mário Araújo ³

¹jhonatheberson@gmail.com

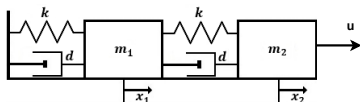
²cetdorea@dca.ufrn.br

³prof.jomario@gmail.com

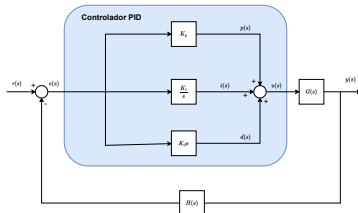
Sumário

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

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

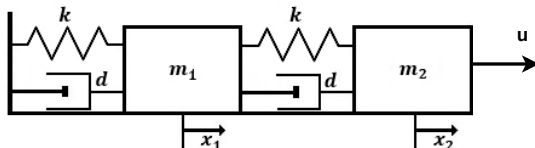
- Problema com realimentação de estado (DANTAS, 2019)
 $u(t) = f^T \dot{x}(t) + g^T x(t)$
- Alocação parcial de polos (DANTAS, 2019)
- Monovariável (DANTAS, 2019)
- Sistemas estáveis em malha aberta (DANTAS, 2019)
- Problema com realimentação derivativa (NUNES et al., 2021)
 $u(t) = f^T \ddot{x}(t) +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

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.

Figura 2: Sistema massa e mola



Fonte: (DANTAS, 2019)

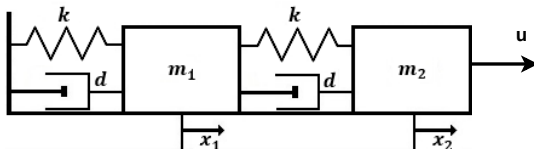
Equação do sistema

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

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

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

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:

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

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

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 $\hat{\mathbf{H}}(s) = (\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K} + q(s)\mathbf{b}\mathbf{I})^{-1}$ é definida como matriz de receptância de malha fechada e $\mathbf{H}(s) = (\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K})^{-1}$ 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:

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

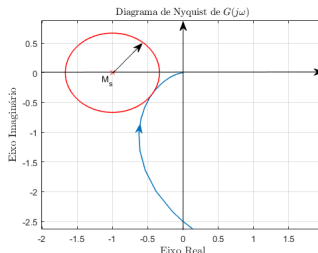
Resposta em frequência

- 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.

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

Definição do Problema - Estabilidade

Figura 4: Diagrama de Nyquist Exemplo de Circunferência. M_s



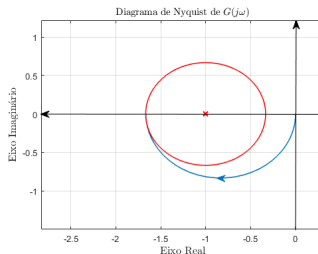
Fonte: (DANTAS, 2019)

Restrições de estabilidade

- Podemos afirmar pela teórica do critério de estabilidade de Nyquist que circunferência M_s representa quanto robusto é o sistema
- M_s é 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

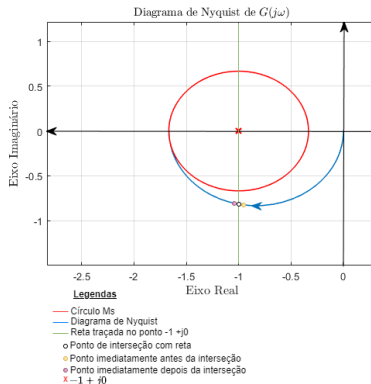
Definição do Problema - Instabilidade

Neste trabalho também aplicaremos os métodos testado para sistema de controle estáveis para sistemas instáveis, porém isso traz algumas complicações no desenvolvimento do algoritmo, para que atenda as funções de restrições.

Pelo critério de *Nyquist*, testamos as raízes do polinômio característico de malha fechada devem ser parte real negativa, mas quando temos um sistema instável significa que temos raízes no SPD, e para tornar esse sistema estável, precisamos forçar que o número de voltas no ponto $-1 + j0$ seja igual ao número de raízes do polinômio característico de malha aberta, assim tornando o sistema estável.

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; DOREA; ARAUJO, 2021):

$$\min_{k_p, k_i, k_d} = \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \quad (17)$$

$$\text{s.a } \operatorname{Re}\{L(j\omega)\} \geq -1 + M_s^{-1} \quad \forall \quad \omega / \operatorname{Im}\{L(j\omega)\} = 0 \quad (18)$$

$$\begin{aligned} L(j\omega) &= \left(k_p + \frac{k_i}{s} + k_d s\right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau} \\ N &= -P \end{aligned} \quad (19)$$

Definição do Problema - Funções de otimização

$$\min_{k_p, k_i, k_d} = \int_0^t |e(t)| dt \quad (20)$$

$$s.a \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \leq \epsilon \quad (21)$$

$$\begin{aligned} L(j\omega) &= \left(k_p + \frac{k_i}{s} + k_d s \right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau} \\ N &= -P \end{aligned} \quad (22)$$

Definição do Problema - Funções de otimização

$$\min_{k_p, k_i, k_d} (1 - \alpha) * \int_0^t |e(t)| dt + \alpha \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \quad \forall \quad \alpha \in \{0, 1\} \quad (23)$$

$$s.a \quad \left(\min_{\omega_i} |L(j\omega) + 1| - M_s^{-1} \right)^2 \leq \epsilon \quad (24)$$

$$\begin{aligned} L(j\omega) &= \left(k_p + \frac{k_i}{s} + k_d s \right) \mathbf{I} \mathbf{H}(j\omega) \mathbf{b} e^{-j\omega\tau} \\ N &= -P \end{aligned} \quad (25)$$

Figura 7: Média e desvio-padrão do fitness de 20 execuções do algoritmo GA comparado ao PSO, nas condições da ref [3]. Resultados do algoritmo deste trabalho em negrito

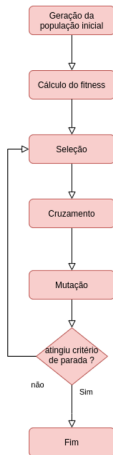
Função	Dimensão	PSO [3]	PSO	GA [3]	GA	GA [5]
Esfera	30	1.0454E+05 ±7.1998E+04	2.241E+03 ±7.030E+02	6.4415E+03 ±1.6876E+03	6.7564E+01 ±4.4760E+01	2.0532E+02 ± 4.6377E+01
Rosenbrock	2	7.0289E+08 ±4.8937E+08	7.0261E+08 ±3.666E+08	1.2493E+07 ±8.6725E+06	9.7689E+00 ±2.5590E+03	1.2280E+02 ±1.9833E+02
Rastrigin	30	5.4130E+02 ±1.5969E+01	1.7430E+02 ±2.506E+01	5.5900E+01 ±1.4294E+01	1.7650E+01 ±3.2255E+00	6.9160E+01 ± 9.5182E+00

Fonte: o autor (2023)

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 8: Fluxograma do algoritmo genético



Fonte: o autor (2023)

Figura 9: 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), M_s);
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), M_s);
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(1,:);
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 10: Pseudo código da função L

Algoritmo 3: Função L

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

Saída: L

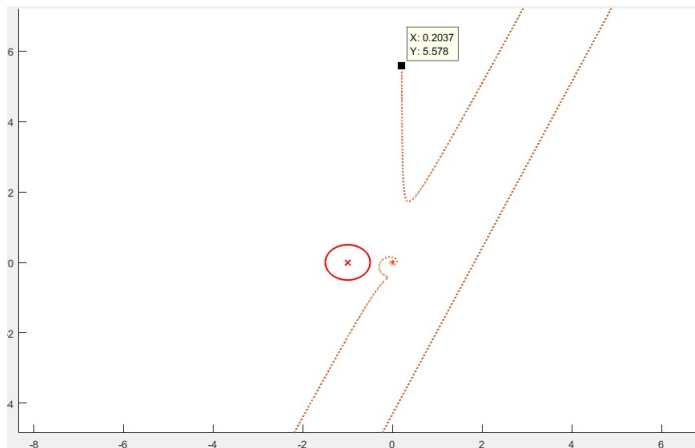
```
1 Kp = populacao(1);
2 Ki = populacao(2);
3 Kd = populacao(3);
4 para i = 1 até i = comprimento( $\omega$ ) faça
5    $\beta = j \times \omega(i)$ ;
6    $L(i) = (Kp + Ki/\beta + 0.0001) + \beta \times Kd) \times I \times (M \times \beta^2 + C \times \beta + K)^{-1} \times B \times e^{-\tau \times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

Fonte: o autor (2023)

Nyquist infinito

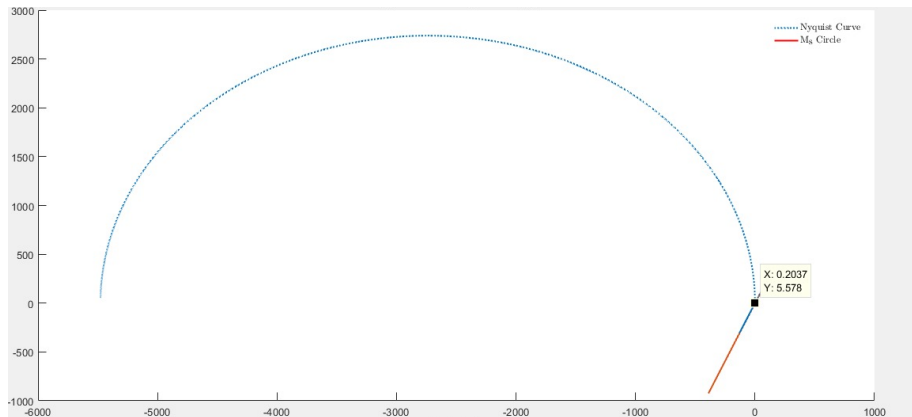
- Perturbação na ação integrativa
- Evita valores infinitos para cálculo do diagrama de *Nyquist*

Figura 11: Diagrama de *Nyquist* para $\omega \in [10^{-1}, 10^3]$



Fonte: o autor (2023)

Figura 12: Diagrama de *Nyquist* para $\omega \in [10^{-6}, 10^3]$



Fonte: o autor (2023)

Metodologia - Algoritmo verificando a variabilidade genética

Figura 13: Pseudo código da função de variabilidade

Algoritmo 8: Função de Variabilidade

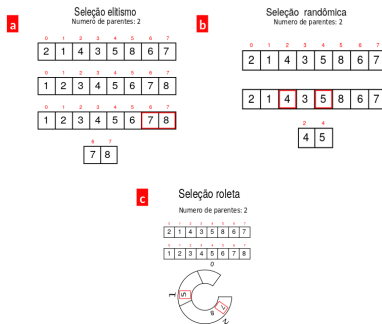
```
Entrada: variabilidade, evolucao, e ff
1 evolucao(variabilidade) = ff;
2 variabilidade = variabilidade + 1;
3 se tamanho(evolucao) > 10 então
4     variabilidade = 1;
5     evolucao = evolucao(2: end);
6     se var(evaluation) < 0.0000001 então
7         objetivo = "fim";
8         melhorIndividuo = 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 14: Ilustração dos métodos implementados de seleção. (a) Seleção elitismo, ordenando de forma crescente e escolhendo os indivíduos com maior fitness. (b) Seleção Randômica, que seleciona os pais aleatoriamente entre a população. (c) Seleção roleta, a qual ordena e rearranja para que simula uma roleta.



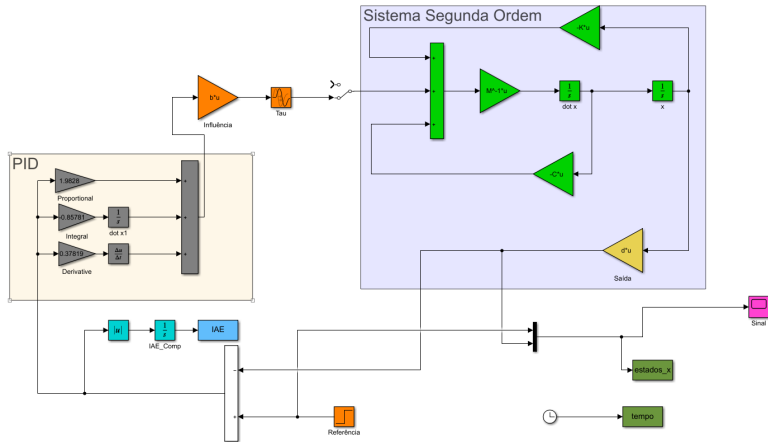
Fonte: o autor (2023)

Figura 15: Algoritmos de cruzamento (crossover). (a) Cruzamento uniforme, em que o novo cromossomo (abaixo) é formado selecionando genes aleatórios de cada um dos pais. (b) Cruzamento de dois pontos, em que o novo cromossomo é formado pelas extremidades de um dos pais e a parte central do outro. Os pontos de corte (tracejados) são sorteados aleatoriamente para cada indivíduo. (c) Cruzamento de um ponto, em que o novo indivíduo é gerado com o início do cromossomo de um dos pais e o final do outro. O ponto de corte (linha tracejada) é decidido aleatoriamente para cada indivíduo.



Fonte: o autor (2023)

Figura 16: Simulação criada com MATLAB/Simulink®



Fonte: o autor (2023)

Experimentos - Exemplo 1

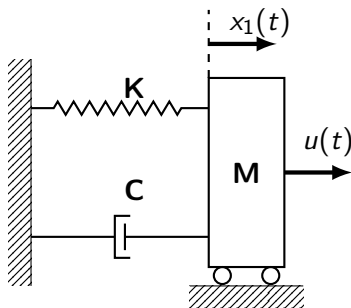
Na Figura 17, é apresentado um exemplo clássico de aplicação do sistema massa-mola-amortecedor de um grau de liberdade. O problema de controle por realimentação de estados é definido para $\mathbf{M} = 1$, $\mathbf{C} = 0,01$, $\mathbf{K} = 5$, $\mathbf{b} = 1$ e $\mathbf{l} = 1$ com atraso igual a $\tau = 0,1s$ como no Exemplo 2 de ram2009state.

O objetivo é encontrar valores de \mathbf{K}_p , \mathbf{K}_i e \mathbf{K}_d , de forma que o diagrama de *Nyquist* de $L(j\omega)$ esteja completamente fora da área delimitada pela circunferência de raio M_s^{-1} , a qual é um parâmetro de projeto definido. A definição de M_s estabelece margens de ganho e de fase para o sistema, e valores comuns de M_s estão no intervalo de $1,22 \leq M_s \leq 1,667$ (SKOGESTAD; POSTLETHWAITE, 2007).

Experimentos - Exemplo 1

O método proposto neste trabalho foca nos parâmetros de projeto baseados no diagrama de Nyquist do ganho de malha do sistema $L(j\omega)$, considerando os conceitos de estabilidade segundo os critérios de Nyquist (DANTAS, 2019). O objetivo da solução é obter um sistema relativamente robusto, tendo como meta $M_s = 1,66$.

Figura 17: Sistema massa-mola-amortecedor de um grau de liberdade



Resultados da Otimização - Exemplo 1

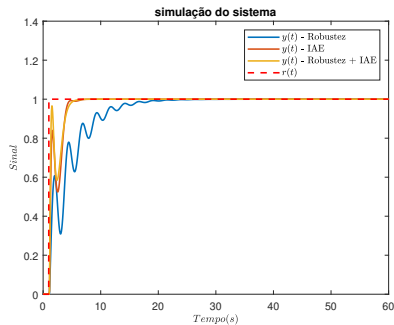
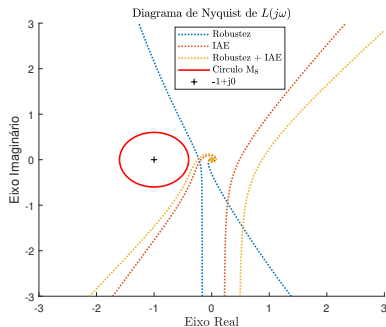
A análise comparativa dos parâmetros otimizados diretamente e observaremos os benefícios de realizar a busca pelo critério de robustez ou pelo índice de *IAE*. A ideia inicial é comparar os gráficos de *Nyquist* e as simulações do sistema, sobrepondo os resultados o quais são os dados são sumarizados na Tabela 1.

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

Sintonia	k_p	k_i	k_d	<i>IAE</i>
Robustez + IAE	2.9222	5,4647	3,6449	0,9149
Robustez	1,4160	1,5225	0,8718	3,2841

Resultados da Otimização - Exemplo 1

Figura 18: Resultado exemplo 1



(a) Comparação do diagrama de Nyquist do Exemplo 1 (b) Comparação da simulação do Sistema do Exemplo 1

Fonte: o autor (2023)

Experimentos - Exemplo 2

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

Considerando o método apresentado em ram2011partial, vamos utilizar como exemplo prático uma matriz de entradas $\mathbf{B}' = \mathbf{I}' = [0 \ 1]$ e uma constante de atraso $\tau = 5$.

Novamente a busca pela solução do problema descrito pela equação 23 é realizada para o círculo $M_s = 1,6667$.

Resultados da Otimização - Exemplo 2

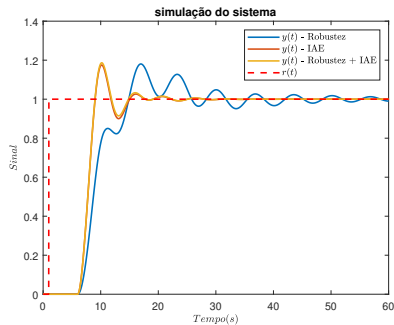
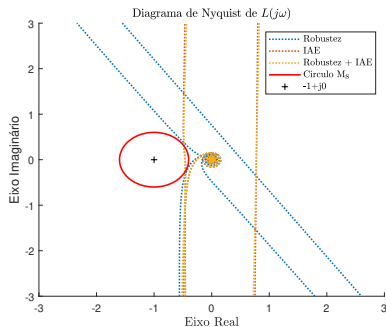
No caso do exemplo 2, por se tratar de um sistema mais complexo e, teoricamente, com maior dificuldade na sintonia dos ganhos do controlador, esses ganhos estão apresentados na Tabela 2,

Tabela 2: 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 19: 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 sistema.

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

Para uma matriz de entradas $\mathbf{B}' = \mathbf{I}' = [0 \ 1]$, e uma constante de atraso $\tau = 0,5$.

Resultados da Otimização - Exemplo 3

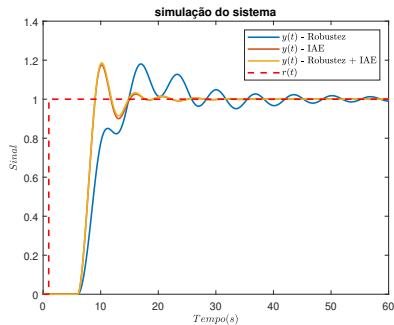
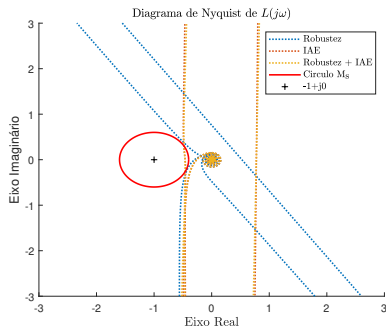
No exemplo 3, os ganhos e índices são evidenciados na Tabela 3, o qual reflete o que podemos observar na Figura 21a que há uma maior distância do diagrama em relação a ambas as otimizações.

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

Sintonia	k_p	k_i	k_d	IAE
Robustez + IAE	0,7548	1,8163	0,4856	2,4390
Robustez	-0,2399	1,0990	1,6015	6,4692

Resultados da Otimização - Exemplo 3

Figura 20: 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)

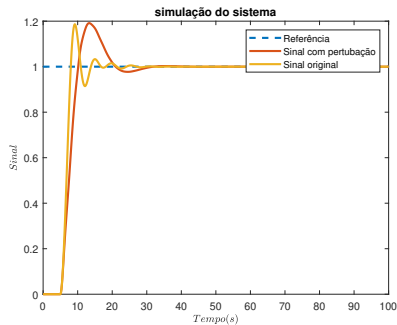
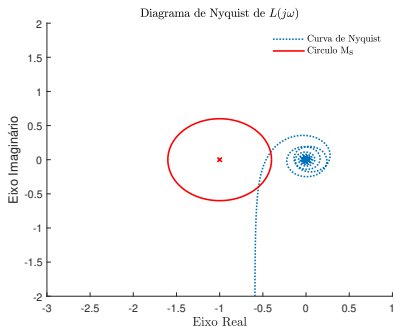
Resultados da perturbação no sistema - Exemplo 3

Simularemos uma perturbação do sistema, 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 sistema, devido à variação da Matriz \mathbf{C} , de um exemplo para outro.

Resultados da perturbação no sistema - Exemplo 3

Figura 21: 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 4

Este exemplo explora a aplicação do método de sintonia em um sistema com atraso longo e restrição de robustez menos rígida. As matrizes do sistema são:

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

Dois casos são explorados nesse exemplo: o caso co-localizado - sensor e atuador juntos no mesmo grau de liberdade, ou seja, $\mathbf{l} = \mathbf{b}^T$ com os parâmetros, $\tau = 5\text{s}$, $\mathbf{l} = \mathbf{b}^T = \begin{bmatrix} 1 & 0 \end{bmatrix}$ e $M_s = 2$. e o caso não co-localizado, na qual a distribuição do sensor e do atuador é distinta com

$$\mathbf{b}^T = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \mathbf{l} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

Resultados da Otimização - Exemplo 4

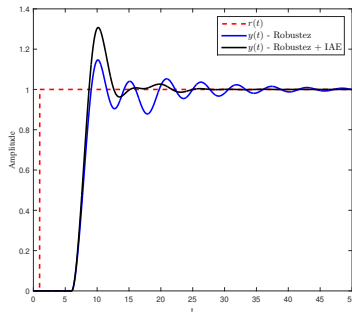
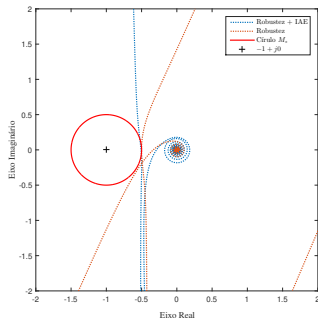
Os resultados para o caso co-localizado, são sumarizados na Tabela 4, onde estão registrados os ganhos dos controladores sintonizados e os valores do índice *IAE*

Tabela 4: Ganhos ótimos e índice de desempenho para o caso co-localizado - Exemplo 4

Sintonia	k_p	k_i	k_d	<i>IAE</i>
Robustez + IAE	0,5710	0,2829	0,4713	7,4872
Robustez	0,5193	0,2424	0,1804	7,9387

Resultados da Otimização - Exemplo 4

Figura 22: Resultado exemplo 4, caso 1



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

Fonte: o autor (2023)

Resultados da Otimização - Exemplo 4

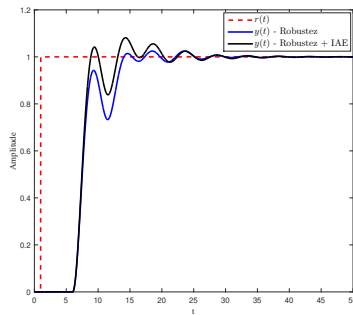
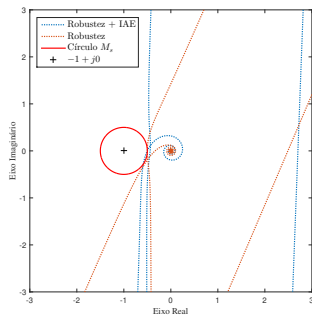
Os ganhos dos controladores sintonizados e os valores do índice IAE podem ser vistos na Tabela 5.

Tabela 5: Ganhos ótimos e índice de desempenho para o caso não co-localizado - Exemplo 4

Sintonia	k_p	k_i	k_d	IAE
Robustez + IAE	0,5878	0,3866	0,5382	7,2212
Robustez	0,5460	0,3437	0,5229	7,5067

Resultados da Otimização - Exemplo 4

Figura 23: Resultado exemplo 4, caso 2



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

Fonte: o autor (2023)

Experimentos - Exemplo 5

Este exemplo, adaptado de shapiro2005stability, aborda um caso instável em malha aberta, cujas matrizes são dadas por:

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

com $\mathbf{b}^T = [-1 \ 1]$ e $\mathbf{l} = [1 \ 0]$ além de um atraso $\tau = 0.5\text{s}$. Para garantia de margens, é dado $M_s = 1,6667$. Neste sistema dois polos de malha aberta localizam-se no semiplano direito, $s_{1,2} = 0,0039 \pm j0,9$. Da mesma forma que nos exemplos anteriores, projetam-se controladores PID conforme o método proposto, e considerando-se apenas o critério de robustez para fins de comparação.

Resultados da Otimização - Exemplo 5

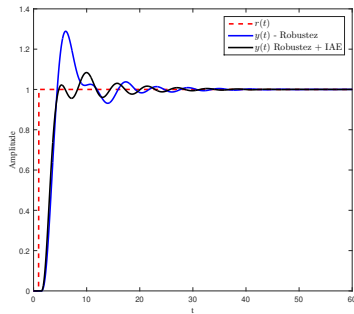
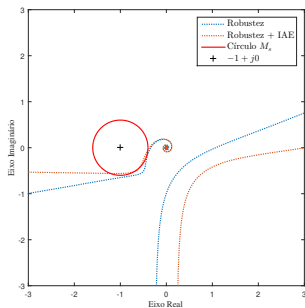
A Tabela 6 permite a comparação dos controladores projetados, e é possível notar um desempenho melhorado em relação ao IAE com a aplicação do método proposto. A estabilidade em malha fechada pode ser verificada no diagrama de *Nyquist* completo ($\omega \in \mathbb{R}$) com visão expandida, onde são constatados os dois envoltimentos requeridos do ponto crítico $-1 + j0$.

Tabela 6: Ganhos ótimos e índice de desempenho para o Exemplo 5

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

Resultados da Otimização - Exemplo 5


Figura 24: 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).

 DANTAS, N. J. B. *Projeto de Controladores para Sistemas de Segunda Ordem com Atraso via Resposta em Frequência*. Dissertação (Dissertação (mestrado) — Mestrado em Engenharia Mecatrônica) — Universidade Federal do Rio Grande do Norte, Natal, 2019.

 DANTAS, N. J. B.; DOREA, C. E. T.; ARAUJO, J. M. Partial pole assignment using rank-one control and receptance in second-order systems with time delay. *Meccanica*, Springer Science and Business Media LLC, v. 56, n. 2, p. 287–302, jan. 2021. Disponível em: <https://doi.org/10.1007/s11012-020-01289-w>.

 NUNES, L. A. et al. Controle por realimentação derivativa de estados de sistemas dinâmicos de segunda ordem com atraso. In: *Anais do Simpósio Brasileiro de Automação Inteligente-SBAI*. [S.l.: s.n.], 2021.

 SKOGESTAD, S.; POSTLETHWAITE, I. *Multivariable feedback control: analysis and design*. [S.l.]: Wiley New York, 2007. v. 2.