

Pedro Jacob Bosman Barros

Sintonia de controladores utilizando técnicas de otimização

Campina Grande, Paraíba

Agosto de 2022

Pedro Jacob Bosman Barros

Sintonia de controladores utilizando técnicas de otimização

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Campus Campina Grande, como parte dos requisitos necessários para a obtenção do grau de Bacharela em Ciências no Domínio da Engenharia Elétrica.

Universidade Federal de Campina Grande – UFCG
Centro de Engenharia Elétrica e Informática – CEEI
Departamento de Engenharia Elétrica – DEE

Orientador: Péricles Rezende Barros, Ph.D.

Campina Grande, Paraíba
Agosto de 2022

Agradecimentos

Os agradecimentos principais são direcionados aos meus pais, Péricles Barros e Heliante Bosman, bem como as minhas irmãs, Clarisse Pétua e Fabrícia Paola, por todo o incentivo e apoio, seja profissional ou emocional, que sempre me deram. Não existem palavras que descrevam minha gratidão à vocês.

Agradeço aos mestres desta jornada pela orientação deste e de outros trabalhos acadêmicos fundamentais na minha formação profissional, em especial: Ph.D. Péricles Barros pela orientação no desenvolvimento deste trabalho acadêmico; D.Sc. George Acioli, por ter aceitado compor a banca de avaliação deste trabalho.

Agradeço também aos professores e funcionários da Universidade Federal de Campina Grande pelos ensinamentos e experiências que vão além dos livros e por tornarem o ensino possível. Em especial, além dos já citados, os professores Alexandre Serres, Gutemberg Lira, Rafael Lima, Wamberto Queiroz, Gutemberg Júnior, Adolfo Herbster, Jaidilson Jó da Silva, Danilo Santos e aos funcionários do departamento, Adail Ferreira e Tchaikowsky Oliveira, que auxiliaram imensamente em todas as questões burocráticas da faculdade.

A minha companheira, Camyla Bandeira, por todo o carinho, apoio e por deixar todos os dias melhores ao longo desse ano. E aos meus grandes amigos, Layrton Reis, Hugo Addobbati, João Lula e Rodrigo Assis, sou imensamente grato por tê-los em minha vida e pela compreensão das ausências. Por fim, agradeço aos meus queridos amigos de curso e que levarei para a vida. Sem vocês este curso não teria graça.

Resumo

Este projeto consiste em encontrar um controlador ótimo para um sistema de controle com o processo de uma placa térmica (sistema de temperatura). Assim, para esse sistema, foi realizada a identificação, com ajuste para um modelo de 1º ordem com atraso, a sintonia do controlador PI, por meio das regras de sintonia SIMC-PID e a otimização dos parâmetros do controlador, por meio do método multicritérios \mathcal{H}^∞ . Nesse sentido, os cálculos realizados para a identificação, sintonia e otimização foram realizados por meio de códigos desenvolvidos no software MATLAB. Por fim, é apresentada uma avaliação sobre a sintonia dos parâmetros do controlador SIMC-PID e do controlador ótimo.

Palavras-chave: Multicritérios \mathcal{H}^∞ , Otimização, Controlador PID, Sintonia, MATLAB.

Abstract

This project consists on finding an optimal controller for a control system with a thermal plate process (temperature system). Thus, for this system, it was made the identification through the model-fitting to a 1st order plus time delay model, the tuning of a PI controller through the SIMC-PID tuning rules, and the optimization of the controller parameters through the multi criteria $\mathcal{H}\infty$. In this sense, the calculations performed for identification, tuning and optimization were made by routines developed in MATLAB software. Finally, an evaluation of the tuning parameters of the SIMC-PID controller and the optimal controller is presented.

Keywords: Multicriteria $\mathcal{H}\infty$, Optimization, PID Controller, Tuning, MATLAB.

Lista de ilustrações

Figura 1 – Diagrama de blocos - Malha de um típico sistema de controle	11
Figura 2 – Diagrama de blocos: Arquitetura do controlador PID Paralelo	17
Figura 3 – Malha fechada de um sistema SISO com planta $G(s)$ e controlador $K(s)$	19
Figura 4 – Interface do MATLAB	23
Figura 5 – Interface do <i>App Designer</i> - Desenvolvimento da Interface do Aplicativo	23
Figura 6 – Interface do <i>App Designer</i> - Desenvolvimento do Código do Aplicativo	24
Figura 7 – Placa Térmica	26
Figura 8 – Aplicativo para Obter Dados da Resposta ao Degrau da Placa Térmica	27
Figura 9 – Resposta ao Degrau - Aquecimento	28
Figura 10 – Resposta ao Degrau - Resfriamento	28
Figura 11 – Respostas ao Degrau do Modelo e dos Dados Experimentais - Aquecimento	29
Figura 12 – Respostas ao Degrau do Modelo e dos Dados Experimentais - Resfriamento	30
Figura 13 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID . . .	31
Figura 14 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID . . .	33
Figura 15 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID e Ótimo	34

Lista de abreviaturas e siglas

FOPTD	Modelo de 1º Ordem com Atraso
PID	Proporcional, Integral e Derivativo
PI	Proporcional e Integral
MOSFET	Transistores de Efeito de Campo de Óxido de Metal Semicondutor

Lista de símbolos

K	Ganho de Estado Estacionário do Processo
T	Constante de tempo do Processo
L	Tempo de Atraso do Processo
$y(t)$	Saída do Sistema
$u(t)$	Entrada do Sistema
$e(t)$	Sinal de Erro
h	Amplitude do Degrau
t	Tempo Contínuo
$\omega(t)$	Ruído Branco
$Z(N)$	Matriz de Dados
V_N	Função de Custo
k_p	Ganho Proporcional
k_i	Ganho Integral
k_d	Ganho Derivativo
$u_c(t)$	Controlador (Tempo)
$U_c(s)$	Controlador (Frequência)
τ_i	Tempo Integral
τ_c	Tempo Integral

Sumário

1	INTRODUÇÃO	10
1.1	Motivação	10
1.2	Objetivos	10
2	FUNDAMENTAÇÃO TEÓRICA	11
2.1	Sistemas de Controle	11
2.2	Identificação de Sistemas	12
2.2.1	Modelagem a Partir da Resposta ao Degrau - Método dos Mínimos Quadrados	13
2.3	Controlador PID	15
2.3.1	Controlador PID Paralelo	16
2.3.2	Conversão do Controlador PID Paralelo para a Forma de Constantes de Tempo	17
2.4	Sintonia de Controladores - SIMC-PID	18
2.5	Controladores PID Ótimos - Multicritérios \mathcal{H}_∞	19
2.5.1	Critério de Rejeição da Perturbação	20
2.5.2	Crítérios de Margem de Estabilidade	20
2.5.3	Otimização dos Parâmetros do Controlador - Multicritérios \mathcal{H}_∞	21
2.6	MATLAB	22
2.6.1	<i>App Designer</i>	22
3	DESENVOLVIMENTO	25
3.1	Placa Térmica - Sistema de Temperatura	25
3.2	Identificação do Sistema	26
3.3	Sintonia do controlador - SIMC-PID	30
3.4	Otimização com o método de multicritérios \mathcal{H}_∞	31
3.5	Avaliação dos Controladores - SIMC-PID e Ótimo	33
4	CONSIDERAÇÕES FINAIS	35
4.1	Trabalhos Futuros	35
	REFERÊNCIAS	36
	APÊNDICES	37
	APÊNDICE A – CÓDIGOS DO MATLAB UTILIZADOS	38
A.1	Parâmetros FOPTD - Identificação	38
A.2	Parâmetros k_p e τ_i - SIMC-PID	38

A.3	Otimização - Critérios \mathcal{H}_∞	39
------------	---	-----------

1 Introdução

Neste capítulo serão apresentados os objetivos do projeto proposto, bem como a motivação para o seu desenvolvimento.

1.1 Motivação

Um método (multicritérios $\mathcal{H}\infty$) para a avaliação geral e objetiva de controladores para todos os tipos de plantas é baseado em três critérios principais, todos relacionados a características vitais de desempenho e robustez. Ele pode ser usado para comparar controladores de diferentes estruturas, mas também como um método de projeto para encontrar a configuração ótima de parâmetros para um controlador de determinada estrutura (KRISTIANSSON; LENNARTSON, 2002). Assim, esse método é utilizado como objeto de estudo e proporciona avanços na área de controle, em que pode ser utilizado para a melhoria de sistemas de controle por meio de um controlador ótimo.

1.2 Objetivos

O Projeto de Conclusão de Curso proposto neste documento tem como objetivo geral encontrar um controlador ótimo para um sistema de controle com o processo de uma placa térmica (sistema de temperatura). Assim, para sua conclusão, são definidos os seguintes objetivos específicos:

- estudar e apresentar as ferramentas e os conceitos utilizados;
- identificar o modelo do sistema da placa térmica;
- realizar a sintonia do controlador por meio das regras de sintonia SIMC-PID;
- tentar encontrar os parâmetros do controlador ótimo por meio da solução de um problema de otimização por meio do método dos multicritérios $\mathcal{H}\infty$.

2 Fundamentação Teórica

2.1 Sistemas de Controle

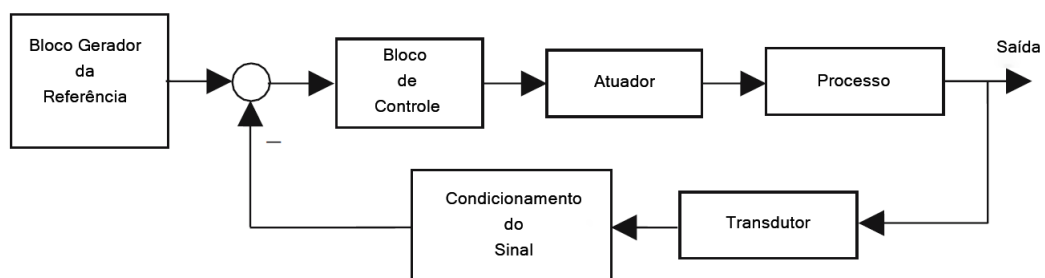
Nos processos, variáveis como a temperatura, a pressão, o nível, a densidade e o pH necessitam estar reguladas corretamente para que seja obtido, ao final do processo, o produto ou resultado desejado. Exemplos típicos de processos são as caldeiras, fornos, fornalhas e torres de destilação que estão frequentemente presentes na indústria. Desse modo, uma das maneiras de regular essas variáveis é por meio de um sistema de controle.

Os principais componentes de um típico sistema de controle podem ser agrupados por: controlador, atuador, processo e a comunicação. Caso o sistema seja realimentado, outro componente básico é a medição (JOHNSON; MORADI, 2005). Um típico sistema de controle realimentado com os seus respectivos componentes é apresentado na figura 1.

Nesse sentido, a ideia central de um sistema de controle realimentado é que a saída de um sistema pode ser medida e retransmitida a um controlador usado para fazer o controle. A partir disso, os principais requisitos do sistema são o de fazer com que a saída do sistema rastreie o sinal de comando da entrada, o de manter o sistema estável e o de rejeitar entradas de distúrbios (FRANKLIN; POWELL; EMAMI-NAEINI, 2013).

Assim, deve-se atentar para a importância do controlador nesse sistema, tendo em vista que é a unidade responsável por garantir a estabilidade e requisitos de desempenho em sistemas e processos que estão sujeitos a perturbações externas e incertezas de modelagem.

Figura 1 – Diagrama de blocos - Malha de um típico sistema de controle



Fonte: Adaptado de (JOHNSON; MORADI, 2005)

Uma descrição detalhada sobre cada um dos componentes de um sistema de controle é desenvolvida a seguir (JOHNSON; MORADI, 2005):

- Processo: O sistema para o qual variáveis físicas específicas devem ser controladas ou reguladas. ex: evaporadores, torres de destilação, estufa de secagem, caldeira de

calefação;

- Atuador: unidade de processo que fornece material ou potência de entrada para o processo. Pode ser considerado que age por meio de amplificação;
- Medição: Para que seja possível o controle, é importante a medição do processo, que tipicamente incorpora um transdutor e componentes de processamento do sinal. O transdutor compreende um sensor que detecta uma propriedade física específica. É possível que a medição seja ruidosa. O ruído será tratado pelo componente condicionante do sinal;
- Controle: unidade designada para gerar um sistema estável em malha fechada e alcançar uma dinâmica pré-especificada e requerimentos de performance do processo. A entrada da unidade de controle é normalmente um sinal de erro baseado na diferença entre um sinal de referência desejado e a saída medida do processo;
- Comunicação: as unidades e componentes na malha de controle são todas conectadas. Em malhas locais pequenas, o sistema de controle é diretamente conectado por fios e cabos, mas em processos distribuídos espacialmente, com salas de controle operacional distantes, componentes de comunicação (redes, transmissores, receptores) serão provavelmente necessários.

2.2 Identificação de Sistemas

A identificação de sistemas é um dos métodos utilizados para a modelagem matemática de um processo. Nesse sentido, ela é utilizada, em alguns casos, quando os parâmetros do modelo não são conhecidos e precisam ser estimados. Desse modo, a identificação é caracterizada por uma abordagem experimental em que os parâmetros das equações que definem o sistema são obtidos a partir dos seus dados de entrada e saída.

Assim, um procedimento de identificação de sistemas pode ser desenvolvido por meio das etapas: aquisição de dados de entrada e saída do sistema, seleção da estrutura do modelo e o cálculo dos parâmetros do modelo de acordo com métodos matemáticos.

O modelo que será utilizado para representação do sistema em estudo é o modelo de 1º Ordem com Atraso (FOPTD). Ele é utilizado na caracterização de processos que possuem um tempo de propagação entre o sinal de entrada e o sinal de saída. O modelo é descrito pela função de transferência dada por:

$$G(s) = \frac{K}{Ts + 1} \cdot e^{-Ls}, \quad (2.1)$$

sendo K o ganho de estado estacionário do processo, T a constante de tempo do processo e L o tempo de atraso do processo.

Portanto, o objetivo da identificação é determinar os parâmetros presentes na matriz θ , apresentada na equação 2.2, composta pelos parâmetros da função de transferência da equação 2.1.

$$\theta = \begin{bmatrix} K & KL & T \end{bmatrix} \quad (2.2)$$

2.2.1 Modelagem a Partir da Resposta ao Degrau - Método dos Mínimos Quadrados

A abordagem experimental utilizada para a identificação do sistema do projeto é feita por meio da resposta ao degrau do processo. Nesse sentido, deve-se supor inicialmente que o processo esteja em estado inicial nulo, e um degrau de amplitude h é aplicado no instante de tempo $t = 0$ à sua variável manipulada. Assim, tanto a entrada quanto a saída do sistema devem ser armazenadas durante o experimento de resposta ao degrau, desde a aplicação do degrau até o momento que a saída alcança o regime estacionário.

A partir do modelo FOPTD da equação 2.1, a saída $y(t)$ para uma entrada em degrau de amplitude h é, no domínio do tempo:

$$y(t) = h \cdot K \left(1 - e^{-\frac{t-L}{T}}\right) + \omega(t) \quad , \quad t \geq L, \quad (2.3)$$

em que $\omega(t)$ é o ruído branco presente na medição de $y(t)$. Essa equação pode ser escrita como:

$$e^{-\frac{t-L}{T}} = 1 - \frac{y(t)}{hK} + \frac{\omega(t)}{hK}, \quad t \geq L. \quad (2.4)$$

Integrando $y(t)$ de $t = 0$ até $t = \tau$ com $\tau \geq L$ ($y(t) = 0$ para $t < L$), e resolvendo:

$$\int_0^\tau y(t)dt = hK \left(t + Te^{-\frac{t-L}{T}}\right)\Big|_{t=L}^{t=\tau} + \int_0^\tau \omega(t)dt. \quad (2.5)$$

Substituindo 2.4 na equação anterior, e sabendo o fato de que $y(L) = 0$, é obtida a seguinte equação:

$$\int_0^\tau y(t)dt = hK \left[\tau - L - T\frac{y(\tau)}{hK}\right] + [T\omega(t)]\Big|_{t=L}^{t=\tau} + \int_0^\tau \omega(t)dt. \quad (2.6)$$

Assim, a partir dessa equação, definindo:

$$A(\tau) = \int_0^\tau y(t)dt, \quad (2.7)$$

e

$$\delta(\tau) = [T\omega(t)]\Big|_{t=L}^{t=\tau} + \int_0^\tau \omega(t)dt, \quad (2.8)$$

a equação 2.5 pode ser reescrita como:

$$\begin{aligned} A(\tau) &= hK \left[\tau - L - T \frac{y(\tau)}{hK} \right] + \delta(\tau) \\ &= \begin{bmatrix} h\tau & -h & -y(\tau) \end{bmatrix} \begin{bmatrix} K \\ KL \\ T \end{bmatrix} + \delta(\tau), \end{aligned} \quad (2.9)$$

e, portanto,

$$A(\tau) = \phi(\tau)^T \theta + \delta(\tau) \quad (2.10)$$

em que

$$\phi(\tau) = \begin{bmatrix} h\tau & -h & -y(\tau) \end{bmatrix}^T, \quad (2.11)$$

e

$$\theta = \begin{bmatrix} K & KL & T \end{bmatrix} \quad (2.12)$$

com o termo $\delta(\tau)$ sendo o ruído.

Assim, a partir do desenvolvimento exposto, foi obtida uma função de custo utilizando o método dos mínimos quadrados para que seja realizada a estimação dos parâmetros do vetor θ de modo que o erro de predição seja o menor possível. Nesse sentido, com a medição dos valores de entrada $u(t)$ e de saída $y(t)$, durante N períodos de amostragem, a matriz de dados $Z(N)$ é apresentada:

$$Z(N) = \begin{bmatrix} y(1) & y(2) & \cdots & y(N-1) & y(N) \\ u(1) & u(2) & \cdots & u(N-1) & u(N) \end{bmatrix}^T \quad (2.13)$$

A função de custo, por meio do método dos mínimos quadrados, é:

$$V_N(\theta, Z(N)) = \frac{1}{N} \sum_{\tau=1}^N [A(\tau) - \phi(\tau)^T \theta]^2, \quad (2.14)$$

de modo que a estimativa dos parâmetros mínimos quadrados $\hat{\theta}^{LS}(N)$ é obtida minimizando a função de custo 2.14:

$$\hat{\theta}^{LS}(N) = \arg \min_{\theta} V_N(\theta, Z(N)). \quad (2.15)$$

Tendo em vista que V_N é quadrática com relação à variável θ , é possível encontrar o valor mínimo da função 2.14 por meio da sua derivada, igualando o resultado a 0. Nesse sentido,

$$\frac{d}{d\theta} V_N(\theta, Z(N)) = \frac{2}{N} \sum_{\tau=1}^N \phi(\tau) (A(\tau) - \phi^T(\tau) \theta) = 0. \quad (2.16)$$

Desenvolvendo, o seguinte resultado é encontrado:

$$\sum_{\tau=1}^N \phi(\tau)A(\tau) = \sum_{\tau=1}^N \phi(\tau)\phi^T(\tau)\theta, \quad (2.17)$$

em que, pode-se definir:

$$R_{n \times n} = \sum_{\tau=1}^N \phi(\tau)\phi^T(\tau), \quad (2.18)$$

e,

$$f_{n \times 1} = \sum_{\tau=1}^N \phi(\tau)A(\tau). \quad (2.19)$$

Nesse sentido, separando θ na equação 2.17, para cada instante de amostragem N , assumindo que a inversa de 2.18 existe, uma estimativa para os parâmetros em θ pode ser calculada pela multiplicação da inversa da matriz $R_{n \times n}$ e de $f_{n \times 1}$:

$$\hat{\theta}(N) = \hat{\theta}^{LS}(N) = R^{-1}f \quad (2.20)$$

Assim, realizando a aquisição dos dados experimentais da resposta ao degrau, por meio da estimativa dos parâmetros em θ , os parâmetros do modelo aproximado de 1º ordem com atraso do processo podem ser encontrados. Nesse sentido, esse modelo aproximado do processo possibilita a realização do projeto do controlador e do sistema de controle.

2.3 Controlador PID

O controlador proporcional, integral e derivativo (PID) é de longe a forma mais dominante de realimentação em uso atualmente (ÅSTRÖM; HÄGGLUND, 2001). Na maioria das aplicações de sistemas de controle, 90-95% das malhas de controle são da forma do PID (BORASE et al., 2021).

A realimentação proporcional, integral e derivativa é baseada no passado (I), presente (P) e futuro (D) do erro de controle $e(t)$ (ÅSTRÖM; HÄGGLUND, 2001), em que esse erro é o resultado da subtração do valor do sinal de comando da entrada, ou seja, o valor da referência $r(t)$ pelo valor da saída medida $y_m(t)$. Portanto, $e(t) = r(t) - y_m(t)$.

Para o controlador PID, o termo proporcional (P), a partir do erro, incorpora mudanças proporcionais apropriadas para a saída de controle. É utilizado quando a ação do controlador têm que ser proporcional a magnitude do erro do sinal do processo.

Outro termo é o integral (I), que examina a variável de processo ao longo do tempo e corrige a saída de forma a reduzir o deslocamento da variável de processo em relação a referência desejada. É utilizado quando a ação do controlador têm que corrigir qualquer

desvio permanente de um sinal de referência de valor constante. O controle integral supera a deficiência do controle proporcional, eliminando o desvio sem utilizar um ganho de controlador excessivamente grande.

Por fim, o termo derivativo (D) monitora a taxa de mudança da variável do processo e, portanto, altera a saída quando há variações incomuns. Se o controlador pode utilizar a taxa de variação do sinal de erro como entrada, então isso pode ser considerado como um elemento de predição para a ação de controle. Portanto, o controle derivativo utiliza a taxa de variação de um sinal de erro.

Desse modo, a família de controladores PID é construída a partir de várias combinações dos termos proporcional, integral e derivativo, utilizadas conforme necessário para atender a requisitos de desempenho específicos (JOHNSON; MORADI, 2005). Nesse sentido, dois elementos dessa família são o controlador PID paralelo e o controlador PID na forma de constantes de tempo.

2.3.1 Controlador PID Paralelo

A arquitetura do controlador PID paralelo é apresentada na figura 2 e possui como representações matemáticas no domínio do tempo e da frequência:

- Domínio do Tempo:

$$u_c(t) = k_P e(t) + k_I \int^t e(\tau) d\tau + k_D \frac{de}{dt}; \quad (2.21)$$

- Domínio da Frequência:

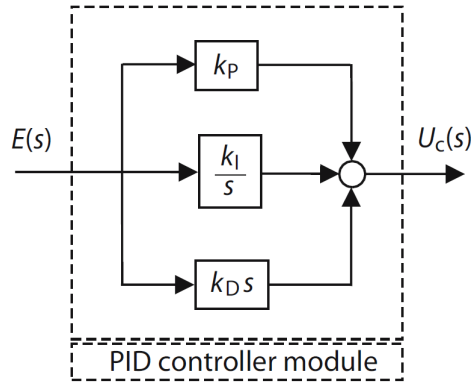
$$U_c(s) = \left[k_P + \frac{k_I}{s} + k_D s \right] E(s). \quad (2.22)$$

Nessas fórmulas, k_P é o ganho proporcional, k_I o ganho integral e k_D é o ganho derivativo, e o controlador opera de acordo com o sinal de erro de referência medido: $e(t) = r(t) - y_m(t)$ (Tempo) e $E(s) = R(s) - Y_m(s)$ (Laplace).

A forma paralela é conhecida por ser da forma PID desacoplada, haja vista que qualquer mudança numérica em qualquer coeficiente, de forma individual, modifica a contribuição somente no caminho do seu termo.

Apesar de que o desacoplamento permite uma sintonia direta dos parâmetros do controlador k_P , k_I , k_D , as representações industriais do controlador PID geralmente utilizam a forma de constantes de tempo para os parâmetros do PID (JOHNSON; MORADI, 2005). Tendo em vista essa importância na indústria, é apresentada a seguir a arquitetura do controlador PID na forma de constantes de tempo.

Figura 2 – Diagrama de blocos: Arquitetura do controlador PID Paralelo



Fonte: (JOHNSON; MORADI, 2005)

2.3.2 Conversão do Controlador PID Paralelo para a Forma de Constantes de Tempo

A partir do domínio do tempo, o controlador PID na forma de constantes de tempo pode ser facilmente derivado do controlador PID paralelo. Nesse sentido, a partir da equação 2.21, o ganho k_P é fatorado:

$$u_c(t) = k_P \left(e(t) + \frac{k_I}{k_P} \int^t e(\tau) d\tau + \frac{k_D}{k_P} \frac{de}{dt} \right). \quad (2.23)$$

Em sequência, são definidas duas constantes de tempo: $\tau_i = \frac{k_P}{k_I}$ e $\tau_d = \frac{k_D}{k_P}$. A partir disso, a arquitetura do controlador PID na forma de constantes de tempo possui como representações matemáticas no domínio do tempo e da frequência:

- Domínio do Tempo:

$$u_c(t) = k_P \left(e(t) + \frac{1}{\tau_i} \int^t e(\tau) d\tau + \tau_d \frac{de}{dt} \right); \quad (2.24)$$

- Domínio da Frequência:

$$U_c(s) = k_P \left[1 + \frac{1}{\tau_i s} + \tau_d s \right] E(s). \quad (2.25)$$

Nessa forma, o controlador deixa de possuir a propriedade desacoplada presente na forma do controlador PID paralelo, haja vista que o parâmetro k_P interage com os termos integral e derivativo. Além disso, é importante observar que para anular o termo integral da forma com constantes de tempo deve-se ter τ_i com grande magnitude, e a operação $\tau_i \rightarrow \infty$ deve ser considerada (JOHNSON; MORADI, 2005).

2.4 Sintonia de Controladores - SIMC-PID

Tendo em vista o controlador PID (seção 2.3), cada um de seus termos (PID) possui parâmetros que devem ser calculados e definidos de forma adequada para o funcionamento do processo, ou seja, deve-se realizar a sintonia do controlador. O projetista do sistema de controle ajusta cada parâmetro das três funções de controle para obter o desempenho desejado do processo (BORASE et al., 2021).

Uma das maneiras de tentar calcular os parâmetros do controlador, de forma a garantir a estabilidade e requisitos de desempenho dos processos, é por meio de regras de sintonia. Desse modo, as regras de sintonia SIMC-PID foram utilizadas no projeto proposto.

Assim, as regras de sintonia SIMC-PID obtêm os parâmetros do controlador PI a partir de modelos de 1º ordem com atraso e os parâmetros do controlador PID a partir de modelos de 2º ordem com atraso. Além disso, as regras de sintonia SIMC-PID satisfazem os seguintes objetivos (SKOGESTAD, 2003):

1. as regras de sintonia devem ser bem motivadas e, de preferência, baseadas em modelo e derivadas analiticamente;
2. deve ser simples e fácil de memorizar;
3. deve funcionar bem em uma ampla gama de processos.

Devido ao projeto proposto ser baseado no modelo de 1º ordem com atraso, e a fim de calcular os parâmetros do controlador PI, foram utilizadas as regras de sintonia SIMC-PID. Portanto, de (SKOGESTAD, 2003), a partir do modelo de 1º ordem com atraso da equação 2.26,

$$G(s) = \frac{K}{T_s + 1} \cdot e^{-Ls}, \quad (2.26)$$

e do controlador PI na forma de constantes de tempo da equação 2.27,

$$U_{PI}(s) = k_P \cdot \left[1 + \frac{1}{\tau_i s} \right] \cdot E(s), \quad (2.27)$$

os parâmetros do controlador k_p e τ_i são calculados pelas equações:

$$k_p = \frac{1}{K} \cdot \frac{T}{\tau_c + L}; \quad \tau_i = \min(T, 4 \cdot (\tau_c + L)); \quad (2.28)$$

nelas, τ_c é a constante de tempo da malha fechada e é o único parâmetro de ajuste. Nesse sentido, para obter um ganho de controlador positivo e diferente de zero, τ_c pode ser

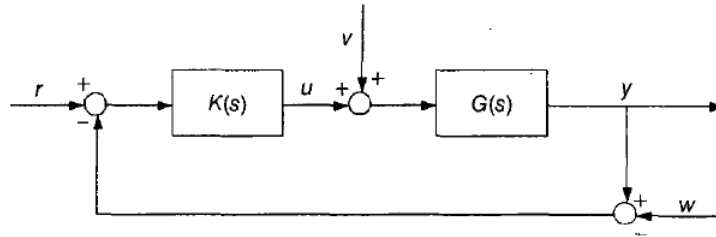
escolhido livremente entre $-L < \tau_c < \infty$. Para uma rápida velocidade de resposta e boa rejeição de perturbação, é desejável um τ_c de menor magnitude. Para estabilidade, robustez e pequena variação de entrada, é desejável um τ_c de maior magnitude.

2.5 Controladores PID Ótimos - Multicritérios \mathcal{H}_∞

Um método baseado em 3 critérios relacionados às características de performance e robustez foi utilizado para calcular os parâmetros ótimos de um controlador PID.

Em primeira instância, para realizar a análise dos critérios utilizados na formulação de um problema de otimização, deve-se considerar o sistema de controle de uma entrada e uma saída (SISO) da figura 3. Neste sistema, $G(s)$ é a planta do processo, $K(s)$ é o controlador, e o sinal de referência $r(t)$, as perturbações do processo $v(t)$ e o ruído de medição $w(t)$ são as entradas. Além disso, a saída do sistema $y(t)$, o sinal de controle $u(t)$ e o sinal do erro $e(t) = r(t) - y(t)$ são as principais saídas do sistema (KRISTIANSSON; LENNARTSON, 2002).

Figura 3 – Malha fechada de um sistema SISO com planta $G(s)$ e controlador $K(s)$



Fonte: (KRISTIANSSON; LENNARTSON, 2002)

A função de transferência da malha é:

$$L(s) = G(s)K(s) \quad (2.29)$$

E as principais funções sensibilidade são:

- Função Sensibilidade:

$$S(s) = \frac{1}{1 + L(s)} \quad (2.30)$$

- Função Sensibilidade Complementar:

$$T(s) = \frac{L(s)}{1 + L(s)} = 1 - S(s) \quad (2.31)$$

- Função Sensibilidade da Perturbação:

$$G_{vy}(s) = \frac{G(s)}{1 + L(s)} = G(s)S(s) \quad (2.32)$$

Assim, um conjunto de critérios \mathcal{H}^∞ apresentados em (LENNARTSON; KRISTIANSOON, 1997) foram utilizados para o projeto e avaliação de controladores PID.

2.5.1 Critério de Rejeição da Perturbação

O primeiro critério está relacionado com a região de baixas frequências. Nesse sentido, um dos objetivos de um sistema de controle é o de compensar as entradas de distúrbios, dentre elas, as perturbações de carga $v(t)$. Portanto, a partir da equação 2.32, ou seja, da função de transferência das perturbações de carga $v(t)$ para a saída do sistema $y(t)$:

$$G_{vy}(s) = \frac{G(j\omega)}{1 + G(j\omega)K(j\omega)}, \quad (2.33)$$

quando o controlador $K(s)$ possui ação integral, $(1 + G(j\omega)K(j\omega)) \approx (G(j\omega)K(j\omega))$ para as baixas frequências. Assim, por meio dessa aproximação, desenvolvendo a equação 2.33, percebe-se que:

$$|G_{vy}(j\omega)| \approx \frac{\omega}{k_i}. \quad (2.34)$$

Pode-se concluir que, para as baixas frequências, a compensação dos distúrbios é melhorada quando o ganho integral k_i aumenta (KRISTIANSOON; LENNARTSON, 2002). Nesse sentido, um critério \mathcal{H}^∞ relacionado com a performance do sistema é apresentado:

$$\begin{aligned} J_v &= \max_{\omega} \frac{1}{\omega} |G_{vy}(j\omega)| \\ &= \left\| \frac{1}{s} G_{yv} \right\|_{\infty}. \end{aligned} \quad (2.35)$$

No critério, o peso de frequência $1/\omega$ está incluído para enfatizar ainda mais os distúrbios de carga para a região de baixa frequência.

2.5.2 Critérios de Margem de Estabilidade

Em sistemas de controle, nas frequências médias, a robustez é alcançada por meio da garantia de boas margens de estabilidade. Assim, a função de transferência $G(j\omega)K(j\omega)$ deve ser mantida à uma distância aceitável do ponto crítico $(-1, 0)$ no gráfico de Nyquist.

Para garantir isso, a menor distância para o ponto crítico $(-1, 0)$, $\min_{\omega} |1 + G(j\omega)K(j\omega)|$, é introduzida como uma medida de estabilidade (LENNARTSON, 2012). Portanto, considerando a função de sensibilidade já apresentada:

$$S(j\omega) = \frac{1}{1 + G(j\omega)K(j\omega)}, \quad (2.36)$$

a magnitude do seu denominador equivale, na frequência ω , à distância do ponto crítico $(-1, 0)$ para $G(j\omega)K(j\omega)$. Portanto, o máximo ganho de $S(j\omega)$ é um critério \mathcal{H}^∞ :

$$\begin{aligned} M_S &= \max_{\omega} |S(j\omega)| \\ &= \|S(s)\|_{\infty}. \end{aligned} \quad (2.37)$$

Assim, a magnitude de M_S é o inverso da menor distância ao ponto crítico $(-1, 0)$ do gráfico de Nyquist. Nesse sentido, um menor valor de M_S significa uma maior margem de estabilidade. Em muitas situações, M_S é uma medida suficiente de robustez de frequências médias (KRISTIANSSON; LENNARTSON, 2002).

Por fim, quando é necessário um amortecimento da resposta ao degrau ou um aumento da margem de fase, sem deixar muito mais lenta a resposta do sistema, a função sensibilidade complementar é utilizada para formular um critério \mathcal{H}^∞ . Assim, por meio dessa função:

$$T(s) = \frac{G(j\omega)K(j\omega)}{1 + G(j\omega)K(j\omega)}, \quad (2.38)$$

o seu máximo ganho é um critério \mathcal{H}^∞ :

$$\begin{aligned} M_T &= \max_{\omega} |T(j\omega)| \\ &= \|T(s)\|_{\infty}. \end{aligned} \quad (2.39)$$

Uma restrição à M_T também controla o amortecimento do sistema sem reduzir muito a magnitude de M_S . Além disso, é importante considerar esse critério para processos instáveis, incluindo os que possuem ação integral (KRISTIANSSON; LENNARTSON, 2002).

2.5.3 Otimização dos Parâmetros do Controlador - Multicritérios \mathcal{H}^∞

A partir dos critérios apresentados, é considerado que o controlador possui um ou mais parâmetros livres para sintonia. Assim, um controlador ótimo é um controlador otimizado que inclui todos os parâmetros disponíveis do controlador no vetor de sintonia ρ . Como exemplo, para o controlador PI, ele possui dois parâmetros livres, $\rho = [k_p \ k_i]$. Quando ambos são utilizados no procedimento de otimização, o controlador ótimo é alcançado (KRISTIANSSON; LENNARTSON, 2002).

Assim, o projetista deve sintonizar os parâmetros do controlador em ρ de modo a obter o comportamento desejado de malha fechada. Nesse sentido, para o método dos multicritérios \mathcal{H}^∞ , o comportamento desejado de malha fechada é definido e medido por meio dos critérios \mathcal{H}^∞ apresentados.

Um método objetivo para avaliar diferentes métodos de projeto é minimizar um critério em relação aos parâmetros de sintonia ρ , enquanto restrições são introduzidas aos

outros critérios (LENNARTSON, 2012). Nesse sentido, um critério a ser minimizado é o J_v , a fim de obter uma melhor performance de rejeição de perturbação, e os critérios sujeitos à restrições são o M_S e o M_T , de modo a garantir a robustez e as margens de estabilidade do sistema.

Assim, os parâmetros ótimos dos controladores podem ser encontrados resolvendo o seguinte problema de otimização com restrições:

$$\begin{aligned} \min_{\rho} J_v(\rho); \\ M_S(\rho) \leq 1, 7; \quad M_T(\rho) \leq 1, 3; \end{aligned} \quad (2.40)$$

em que os valores 1, 7 e 1, 3, definidos para M_S e M_T , são comuns e razoáveis para garantir a estabilidade em diversos sistemas de controle.

Por fim, por meio desses procedimentos de otimização multicritérios com restrições, e dos próprios critérios \mathcal{H}_∞ , diferentes controladores podem ser projetados e avaliados em condições iguais. Ademais, deve-se observar que os algoritmos de otimização demandam um bom e confiável modelo para o processo ser controlado, além de possivelmente demandarem valores iniciais específicos para convergirem de forma a encontrar uma solução.

2.6 MATLAB

O MATLAB é uma plataforma de programação e de computação numérica voltada para a análise de dados, desenvolvimento de algoritmos e a criação de modelos, utilizada por milhões de engenheiros e cientistas (MATHWORKS, 1994-2021). A interface do MATLAB é apresentada na figura 4.

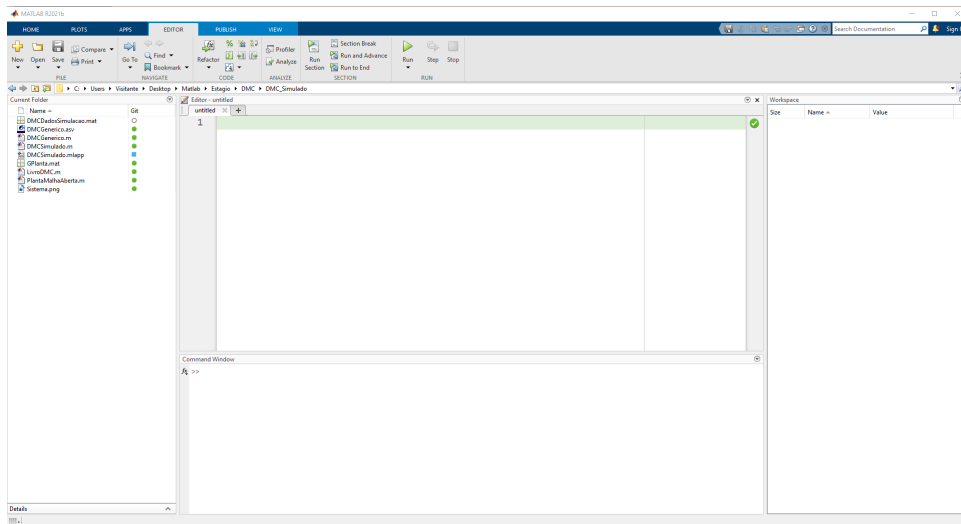
Algumas das áreas em que a plataforma pode ser utilizada são robótica, sistemas de controle, aprendizado de máquina, processamento de sinais e de imagens, testes e medições. Essa vasta quantidade de aplicações é justificada pelas capacidades e ferramentas disponíveis.

Os algoritmos são desenvolvidos no software por meio da linguagem MATLAB. Ao executar um código, o MATLAB possui um intérprete que "lê" o código linha por linha. Dentre as suas capacidades, o desenvolvimento de aplicativos para web e para a área de trabalho se dá por meio da ferramenta *App Designer*.

2.6.1 *App Designer*

Dentre as capacidades do MATLAB, o desenvolvimento de aplicativos para web e para a área de trabalho se dá por meio do *App Designer*. Esse ambiente integrado permite com que o usuário construa simultaneamente a Interface Gráfica do Usuário (*Graphical User Interface* - GUI) e o seu código por trás.

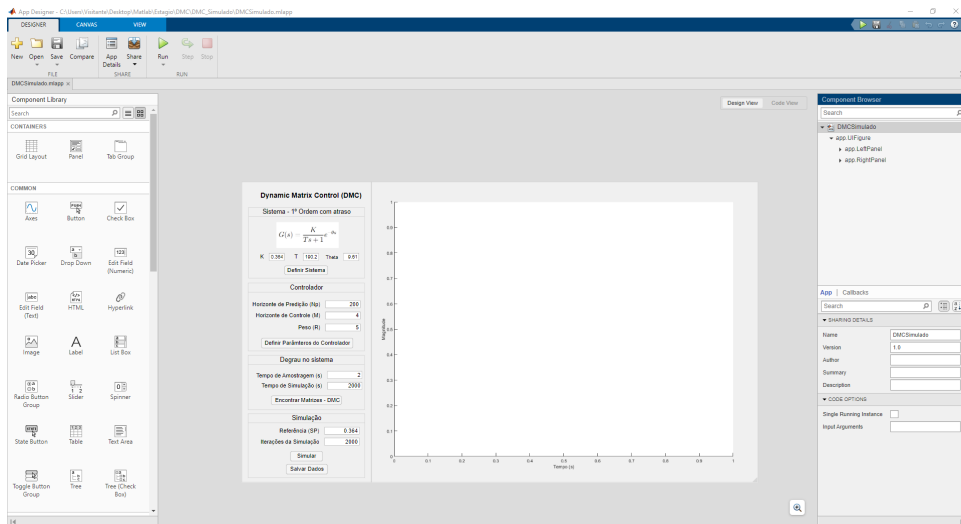
Figura 4 – Interface do MATLAB



Fonte: Elaboração própria

A figura 5 apresenta a interface do *App designer* que possibilita a construção da GUI. Nesse sentido, ao arrastar e soltar componentes visuais para definir a interface gráfica, o *App Designer* gera automaticamente o código orientado a objetos que especifica o esquema (*layout*) e a GUI.

Figura 5 – Interface do *App Designer* - Desenvolvimento da Interface do Aplicativo



Fonte: Elaboração própria

Além da GUI, é possível programar o comportamento do aplicativo. A figura 6 apresenta a interface do *App Designer* que possibilita a construção do código em MATLAB associado à interface gráfica.

Assim, o editor separa o código nas seguintes partes:

- códigos predefinidos do sistema, com configurações padrões dos aplicativos, e que

3 Desenvolvimento

O estudo e a compreensão dos princípios apresentados no Capítulo 2 formam a base para o desdobramento do projeto proposto. Assim, foi realizada a identificação, a sintonia do controlador PID e a otimização dos parâmetros do controlador para um sistema de temperatura (Placa Térmica).

Em primeira instância, foram obtidos os dados experimentais da resposta ao degrau do sistema. A partir disso, foi realizado o ajuste dos dados da resposta ao degrau com a resposta de um modelo de 1º ordem com atraso (equação 3.1). Em sequência, foi realizada a sintonia dos parâmetros do controlador PI para o sistema por meio do método SIMC-PID, e, partindo dessa sintonia do controlador, foi realizada a otimização dos seus parâmetros.

$$G(s) = \frac{K}{Ts + 1} \cdot e^{-Ls}, \quad (3.1)$$

sendo K o ganho de estado estacionário do processo, T a constante de tempo do processo e L o tempo de atraso do processo.

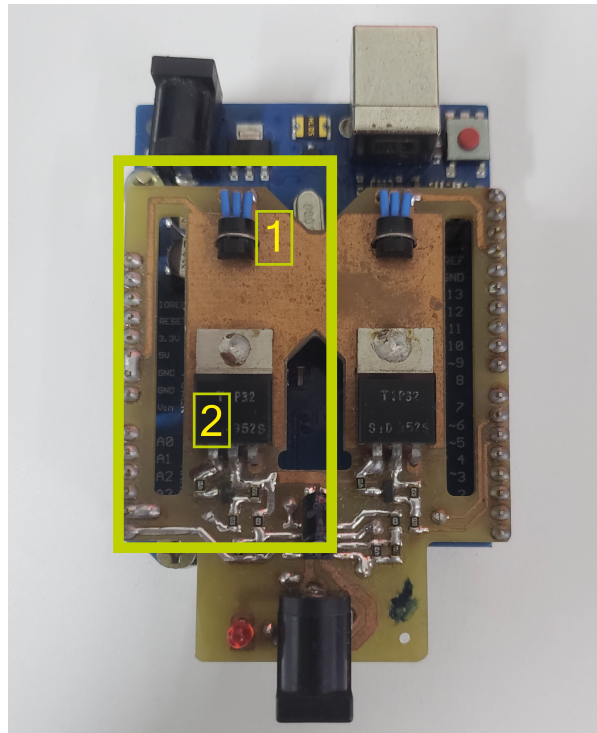
3.1 Placa Térmica - Sistema de Temperatura

A placa térmica é um sistema de temperatura e está apresentada na figura 7. Nesse sentido, o sistema é composto por 2 sensores de temperatura e 2 transistores de efeito de campo de óxido de metal semicondutor (MOSFET). A disposição desses componentes foi feita de modo a permitir a propagação de calor por meio de uma fina lâmina de cobre que se deposita sobre um material de fibra de vidro, ambos materiais condutores de calor. Assim, o processo térmico ocorre ao aplicar um nível de tensão aos terminais dos MOSFETs, de modo a dissiparem calor, que irá se propagar e atingir os sensores de temperatura.

Desse modo, o processo da placa térmica é caracterizado como um sistema de duas entradas e duas saídas (TITO), cujas entradas, as variáveis manipuladas (MV), são os níveis de tensão aplicados aos MOSFETs, e as saídas, as variáveis controladas ou de processo (PV), são as temperaturas medidas utilizando os sensores.

Para esse projeto, somente um único MOSFET e único sensor de temperatura são utilizados, sendo tratado como um processo é monovariável. Na figura 7, o sensor de temperatura está representado pelo índice 1 e o MOSFET pelo índice 2.

Figura 7 – Placa Térmica



Fonte: Autoria própria, 2022

3.2 Identificação do Sistema

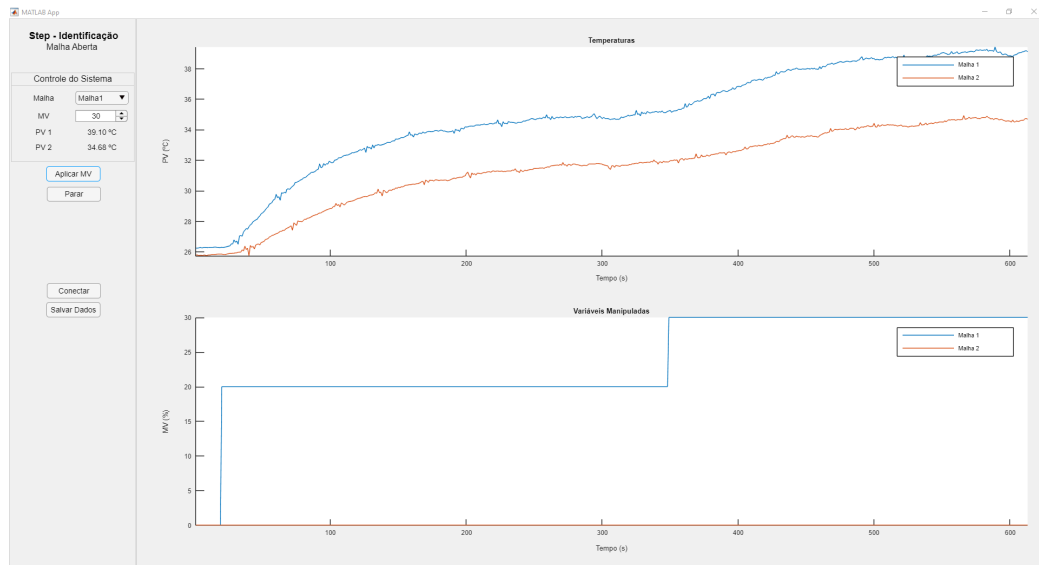
Para realização dos ensaios de identificação, foi desenvolvido um aplicativo por meio da ferramenta *App Designer* do software MATLAB, cuja interface é apresentada na figura 8.

Assim, os principais objetivos do aplicativo eram o de permitir a modificação das variáveis manipuladas, a fim de aplicar um degrau ao sistema, e o de realizar a leitura das variáveis de processo da placa térmica. A partir disso, o aplicativo dispõe uma visualização gráfica dessas variáveis e uma opção para salvar os dados experimentais.

O ensaio para obtenção dos dados experimentais consistiu, inicialmente, em aplicar um degrau positivo para levar o sistema à um ponto de operação desejado. Após isso, a fim de obter os dados da resposta ao degrau do sistema, foi aplicado outro degrau positivo ao sistema. Nesse sentido, a partir da constatação visual (gráficos do aplicativo) de que o sistema alcançou o regime estacionário, foi aplicado um degrau negativo de mesma magnitude do anterior, de modo a obter os dados da resposta ao degrau negativo. Ao passo em que o sistema alcançou o regime estacionário, retornando ao ponto de operação, os dados do experimento foram salvos.

Para a identificação do sistema por meio do método dos mínimos quadrados, é

Figura 8 – Aplicativo para Obter Dados da Resposta ao Degrau da Placa Térmica



Fonte: Autoria própria, 2022

necessário que os dados experimentais da resposta ao degrau estejam formatados de maneira que a resposta do processo esteja com estado inicial nulo e que o degrau seja aplicado no instante de tempo $t = 0$. Desse modo, os dados obtidos no ensaio foram normalizados para cumprir esses requisitos.

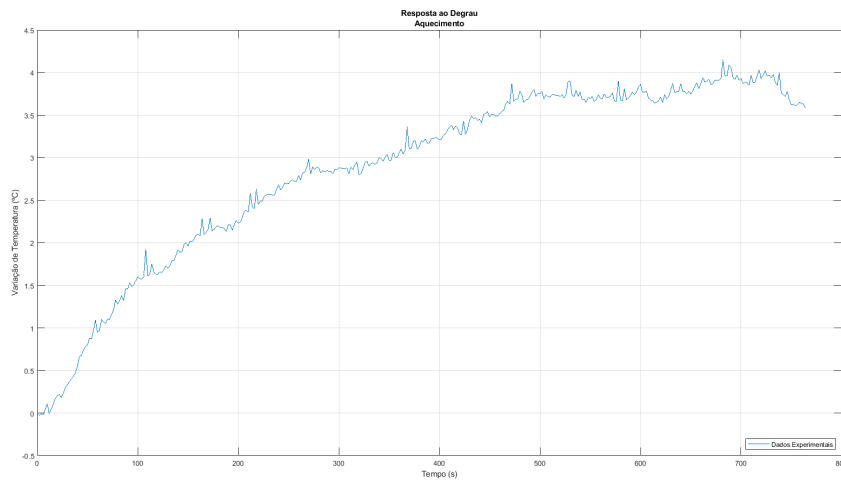
A normalização dos dados experimentais da resposta ao degrau foi efetuada por meio de 2 operações:

- seleção dos valores em um intervalo que começa a partir do instante da aplicação do degrau até o estado estacionário, como forma de simular a aplicação do degrau no instante de tempo $t = 0$.
- subtração de todos os valores pela amplitude inicial, como forma de simular o estado inicial nulo.

Assim, pelo aspecto físico do sistema, e caso o ambiente do processo esteja com uma temperatura menor que a temperatura do ponto de operação inicial, um degrau positivo aplicado ao sistema significa um aumento de temperatura, ou seja, um aquecimento. De modo semelhante, um degrau negativo se caracteriza pela diminuição da temperatura, ou seja, um resfriamento. Nesse sentido, os dados experimentais de aquecimento e resfriamento, normalizados, estão apresentados nas figuras 9 e 10, respectivamente.

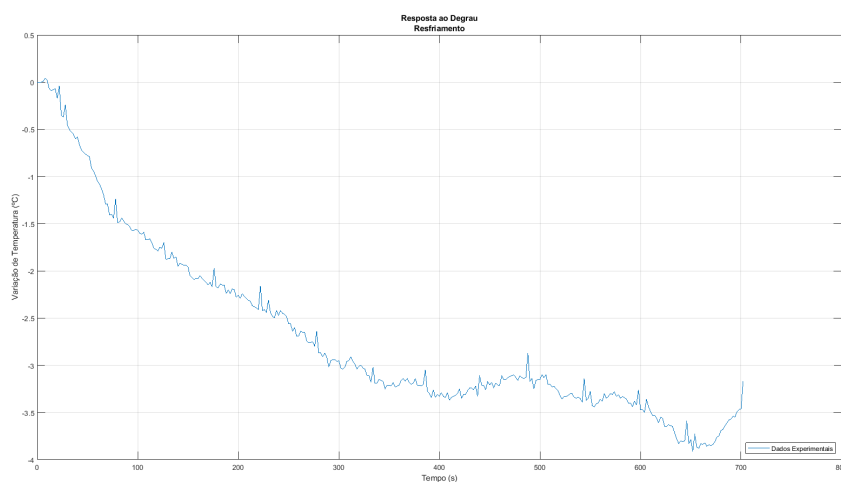
A partir dos dados experimentais e das figuras, devido ao modo com que ocorrem as trocas de calor no processo, é possível observar que a dinâmica da resposta ao degrau para o aquecimento é, de forma visual, diferente da dinâmica de resfriamento. Nesse sentido, para a dinâmica do aquecimento, as trocas de calor ocorrem de forma "ativa", por meio

Figura 9 – Resposta ao Degrau - Aquecimento



Fonte: Autoria própria, 2022

Figura 10 – Resposta ao Degrau - Resfriamento



Fonte: Autoria própria, 2022

da dissipação de calor dos MOSFETs, e de forma "passiva", por meio das trocas de calor com o ambiente. Entretanto, para a dinâmica do resfriamento, as trocas ocorrem somente de forma "passiva", por meio das trocas de calor com o ambiente. Desse modo, foram inicialmente obtidos 2 modelos da resposta ao degrau na identificação, 1 para a dinâmica de aquecimento e outro para a de resfriamento.

É comum nas indústrias de processos que uma simples modelagem baseada em dados experimentais da resposta ao degrau se tornem um exercício de ajuste de modelo com as respostas de sistemas de primeira ordem (JOHNSON; MORADI, 2005). Dessa maneira, um modelo de sistema utilizado na aproximação da resposta ao degrau de diversos processos reais é o modelo de 1º ordem com atraso (FOPTD), apresentado na equação 3.1.

De forma equivalente aos processos industriais, as respostas ao degrau da placa térmica se assemelham às respostas ao degrau do modelo FOPTD.

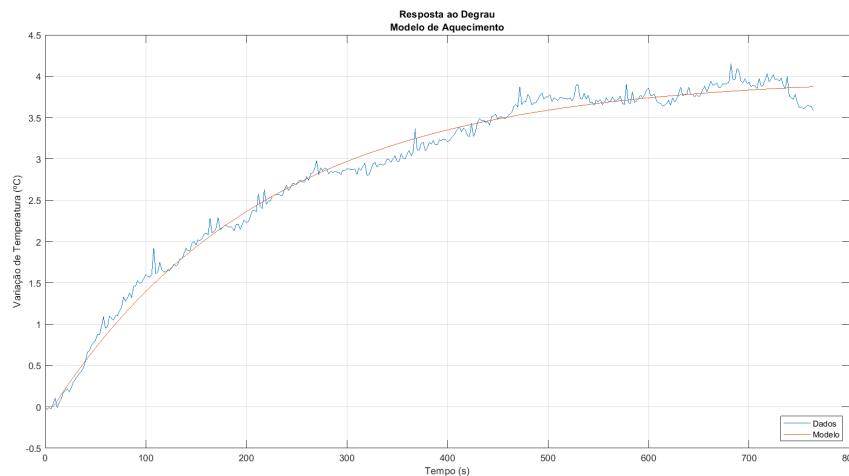
Assim, a identificação dos modelos de aquecimento e resfriamento do processo consistiu em calcular os parâmetros da equação FOPTD cuja resposta ao degrau melhor se ajusta às respostas ao degrau dos dados experimentais. Assim, esse exercício de ajuste de modelo foi realizado utilizando um simples código do MATLAB, apresentado no apêndice A.1, que calcula os parâmetros K , T e L da equação 3.1.

Para os dados experimentais da resposta ao degrau de aquecimento, a identificação resultou no seguinte modelo:

$$G_{\text{Aquecimento}}(s) = \frac{0,3991}{214,4s + 1} \cdot e^{-7,87s}, \quad (3.2)$$

sendo $K = 0,3991$, $T = 214,4$ e $L = 7,87$. Além disso, o gráfico que possui a resposta ao degrau desse modelo e dos dados experimentais da resposta ao degrau de aquecimento é apresentado na figura 3.2.

Figura 11 – Respostas ao Degrau do Modelo e dos Dados Experimentais - Aquecimento



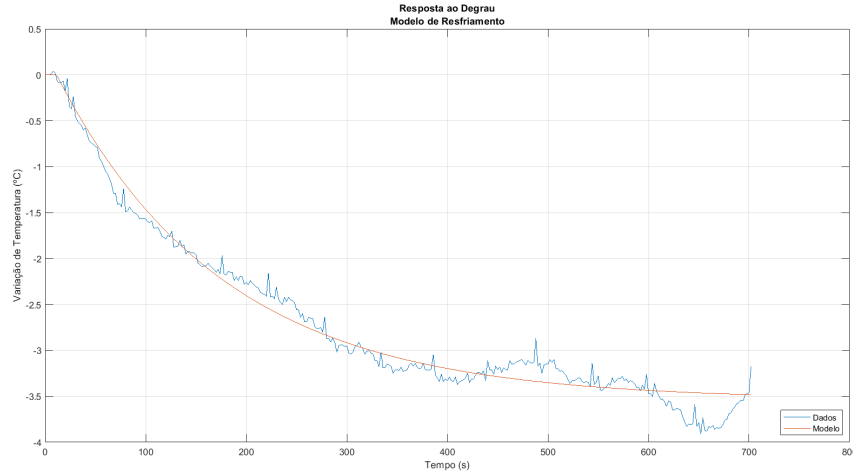
Fonte: Autoria própria, 2022

Para os dados experimentais da resposta ao degrau de resfriamento, a identificação resultou no seguinte modelo:

$$G_{\text{Resfriamento}}(s) = \frac{0,3538}{165,9s + 1} \cdot e^{-11,4s}, \quad (3.3)$$

sendo $K = 0,3538$, $T = 165,9$ e $L = 11,4$. Além disso, o gráfico que possui a resposta ao degrau desse modelo e dos dados experimentais da resposta ao degrau de resfriamento é apresentado na figura 3.3.

Figura 12 – Respostas ao Degrau do Modelo e dos Dados Experimentais - Resfriamento



Fonte: Autoria própria, 2022

Portanto, com forma de obter um único modelo do sistema a partir dos modelos de aquecimento e resfriamento, foi realizada a média aritmética de cada um dos parâmetros das duas equações. Desse modo, foi realizada a média aritmética entre os parâmetros $K_{Aquecimento}$ e $K_{Resfriamento}$, entre $T_{Aquecimento}$ e $T_{Resfriamento}$ e entre $L_{Aquecimento}$ e $L_{Resfriamento}$.

Por fim, resultante da média aritmética, o modelo FOPTD para o sistema de temperatura monovariável da placa térmica está apresentado na equação 3.4:

$$G_{PlacaTérmica}(s) = \frac{0,3764}{190,2s + 1} \cdot e^{-9,61s}, \quad (3.4)$$

sendo $K = 0,3764$, $T = 190,2$ e $L = 9,61$.

3.3 Sintonia do controlador - SIMC-PID

A partir do modelo de 1º ordem com atraso obtido na seção 3.2, referente à placa térmica:

$$G_{PlacaTérmica}(s) = \frac{0,3764}{190,2s + 1} \cdot e^{-9,61s}, \quad (3.5)$$

sendo $K = 0,3764$, $T = 190,2$ e $L = 9,61$, os parâmetros do controlador PI foram ajustados de acordo com as regras de sintonia SIMC-PID.

Assim, por meio das equações:

$$k_p = \frac{1}{K} \cdot \frac{T}{\tau_c + L}; \quad \tau_i = \min(T, 4 \cdot (\tau_c + L)); \quad (3.6)$$

foi desenvolvido um simples código no software MATLAB que calcula esses parâmetros do controlador PI, apresentado no apêndice A.2. Desse modo, para as equações em 3.6, um bom valor para τ_c é obtido igualando essa constante de tempo ao tempo de atraso do processo, $\tau_c = L$, uma vez que isso resulta em uma resposta razoavelmente rápida, com uso moderado de entrada e boas margens de robustez (SKOGESTAD, 2003).

Portanto, utilizando o código MATLAB, os valores dos parâmetros do controlador PI calculados foram:

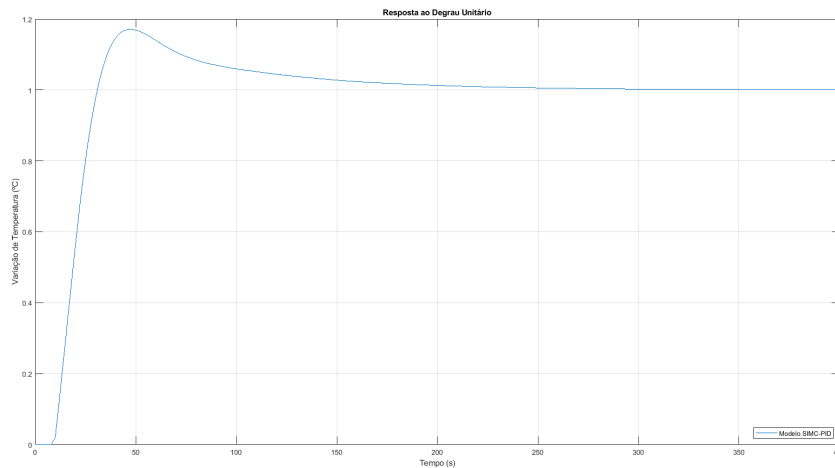
$$\begin{aligned} k_p &= \frac{1}{0,3764} \cdot \frac{190,2}{9,61 + 9,61} \\ &= 26,2770; \end{aligned} \quad (3.7)$$

e

$$\begin{aligned} \tau_i &= \min(190,2, 4 \cdot (9,61 + 9,61)) \\ &= 76,8985; \end{aligned} \quad (3.8)$$

Para esses parâmetros, a simulação da resposta ao degrau unitário do sistema de controle é apresentada na figura 13.

Figura 13 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID



Fonte: Autoria própria, 2022

3.4 Otimização com o método de multicritérios \mathcal{H}_∞

A partir dos critérios da seção 2.5, um método objetivo para avaliar diferentes métodos de projeto foi apresentado. O método consiste em minimizar um critério em relação aos parâmetros de sintonia $\rho = [k_p \quad k_i]$ do controlador PI, enquanto restrições são introduzidas aos outros critérios (LENNARTSON, 2012).

Para esse projeto, um critério a ser minimizado foi o J_v , a fim de obter uma melhor performance de rejeição de perturbação, e os critérios sujeitos à restrições são o M_S e o M_T , de modo a garantir a robustez e as margens de estabilidade do sistema. Portanto, os parâmetros ótimos do controlador PI podem ser encontrados resolvendo o seguinte problema de otimização com restrições:

$$\begin{aligned} \min_{\rho} J_v(\rho); \\ M_S(\rho) \leq 1,7; \quad M_T(\rho) \leq 1,3; \end{aligned} \quad (3.9)$$

em que os valores 1,7 e 1,3, definidos para M_S e M_T , são comuns e razoáveis para garantir a estabilidade em diversos sistemas de controle.

Assim, para encontrar uma solução para o problema de otimização, foi desenvolvido um código no software MATLAB, apresentado no apêndice A.3. No código, o MATLAB calcula a solução do problema em 3.9 por meio da rotina *fmincon*.

Devido ao algoritmo exigir uma condição inicial conveniente para que a solução convirja durante a tentativa de minimizar o valor do critério J_v , e como forma de comparar com o controlador calculado em SIMC-PID, as condições iniciais dos parâmetros em ρ foram escolhidas iguais aos parâmetros do controlador SIMC-PID calculados em 3.3.

A partir dessa condição, utilizando o código desenvolvido no MATLAB, foi encontrada uma solução para esse problema de otimização dos parâmetros do controlador PI para o sistema de controle com o processo da placa térmica. Desse modo, os parâmetros ótimos calculados do controlador PI são:

$$k_p = 25,7850 \quad (3.10)$$

e

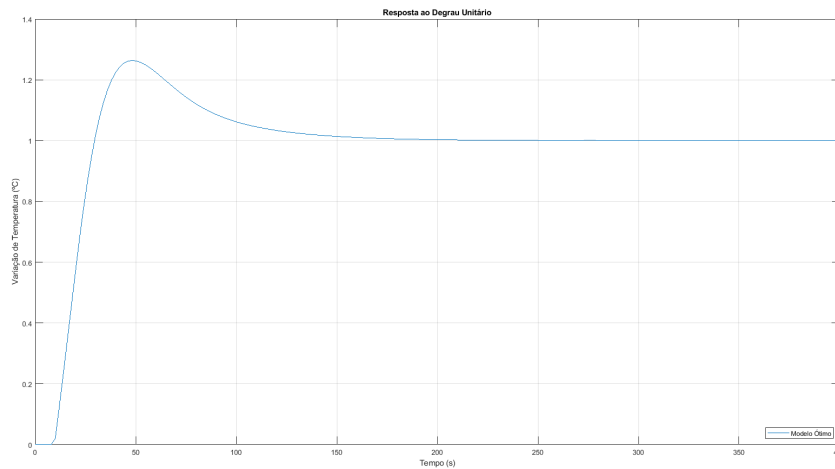
$$k_i = 0,4913. \quad (3.11)$$

Tendo em vista que $\tau_i = \frac{k_p}{k_i}$, o valor de τ_i para o controlador ótimo é:

$$\tau_i = 52,4830. \quad (3.12)$$

Para esses parâmetros, a simulação da resposta ao degrau unitário do sistema de controle é apresentada na figura 14.

Figura 14 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID



Fonte: Autoria própria, 2022

3.5 Avaliação dos Controladores - SIMC-PID e Ótimo

Os critérios apresentados em 2.5 estão relacionados às características de performance e robustez. Nesse sentido, a partir deles, diferentes controladores podem ser avaliados em condições iguais (KRISTIANSSON; LENNARTSON, 2002). Desse modo, a comparação entre os critérios \mathcal{H}_∞ dos controladores SIMC-PID e ótimo, para o sistema de controle com o processo da placa térmica, é apresentada na tabela 1:

Tabela 1 – Comparação dos critérios \mathcal{H}_∞ - SIMC-PID e Ótimo

Sintonia	J_v	M_s	M_t
SIMC-PID	2.9242	1.6507	1,1565
Ótimo	2.0354	1.7000	1.3000

Fonte: Autoria própria, 2022

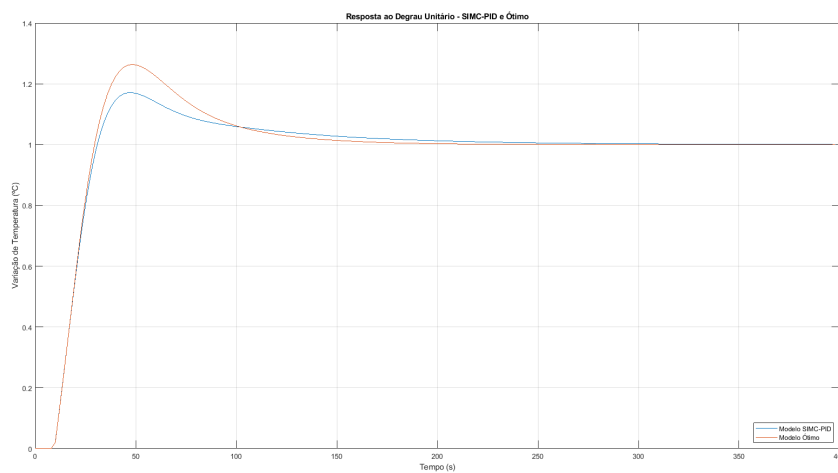
Desse modo, em comparação ao calculado com o método SIMC-PID, por meio da solução do problema de otimização, foi possível encontrar parâmetros do controlador que melhoram a performance para a compensação da perturbação de carga.

Além disso, por meio de M_s e M_t , é possível observar que a solução do problema de otimização encontra valores que garantem as margens de estabilidade definidas para esses critérios, restringido-os durante o cálculo de minimização de J_v . Nesse sentido, melhorias de desempenho podem ser alcançadas sem perda de robustez, sendo possível, às vezes, serem assumidos valores não tradicionais para os parâmetros do controlador.

Por fim, é importante observar que o controlador encontrado por meio das regras de sintonia SIMC-PID obteve parâmetros com valores próximos ao do controlador ótimo,

garantindo uma boa performance e robustez para esse sistema de controle. A comparação das respostas ao degrau unitário dos sistemas de controle para o controlador SIMC-PID e o controlador ótimo é apresentada na figura 15

Figura 15 – Resposta ao Degrau Unitário do Sistema de Controle - SIMC-PID e Ótimo



Fonte: Autoria própria, 2022

4 Considerações Finais

Neste trabalho de conclusão de curso obteve-se como resultado final a solução de um problema de otimização por meio do método dos multicritérios $\mathcal{H}\infty$. O intuito do problema de otimização era o de encontrar os parâmetros do controlador ótimo do sistema da placa térmica, de modo a melhorar a performance do sistema sem perder a robustez desejada.

O projeto pontua sobre os sistemas de controle, a identificação de sistemas, os controladores PID, a sintonia dos controladores por meio das regras de sintonia SIMC-PID e a otimização por meio do método dos multicritérios $\mathcal{H}\infty$, com esses conceitos aplicados ao processo da placa térmica.

Este trabalho de conclusão de curso demandou e aprofundou diversos conhecimentos adquiridos durante a graduação. Ademais, proporcionou o desenvolvimento das habilidades e conhecimentos acerca da área de controle. Assim, espera-se que o trabalho motive novos projetos de controladores ótimos.

4.1 Trabalhos Futuros

Como proposta de melhorias para pesquisas similares, tem-se:

- avaliar os controladores no sistema de controle com o processo físico da placa térmica;
- apresentar as propriedades por trás dos critérios $\mathcal{H}\infty$, dentre elas, a sua influência no desempenho e na robustez do sistema;
- analisar outras condições iniciais para o problema de otimização.

Referências

- ÅSTRÖM, K. J.; HÄGGLUND, T. The future of pid control. *Control engineering practice*, Elsevier, v. 9, n. 11, p. 1163–1175, 2001. Citado na página 15.
- BORASE, R. P. et al. A review of pid control, tuning methods and applications. *International Journal of Dynamics and Control*, Springer, v. 9, n. 2, p. 818–827, 2021. Citado 2 vezes nas páginas 15 e 18.
- FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. *Sistemas de controle para engenharia*. [S.l.]: Bookman Editora, 2013. Citado na página 11.
- JOHNSON, M. A.; MORADI, M. H. *PID control*. [S.l.]: Springer, 2005. Citado 4 vezes nas páginas 11, 16, 17 e 28.
- KRISTIANSSON, B.; LENNARTSON, B. Robust and optimal tuning of pi and pid controllers. *IEE Proceedings-Control Theory and Applications*, IET, v. 149, n. 1, p. 17–25, 2002. Citado 5 vezes nas páginas 10, 19, 20, 21 e 33.
- LENNARTSON, B. Multi criteria \mathcal{H}_∞ optimal pid controllers from an undergraduate perspective. *IFAC Proceedings Volumes*, Elsevier, v. 45, n. 3, p. 697–702, 2012. Citado 3 vezes nas páginas 20, 22 e 31.
- LENNARTSON, B.; KRISTIANSSON, B. Pass band and high frequency robustness for pid control. In: IEEE. *Proceedings of the 36th IEEE Conference on Decision and Control*. [S.l.], 1997. v. 3, p. 2666–2671. Citado na página 20.
- MATHWORKS. *Página principal do MATLAB no site MathWorks*. 1994–2021. Acessado em: 29 de setembro de 2021. Disponível em: <<https://www.mathworks.com/products/matlab.html>>. Citado na página 22.
- SKOGESTAD, S. Simple analytic rules for model reduction and pid controller tuning. *Journal of process control*, Elsevier, v. 13, n. 4, p. 291–309, 2003. Citado 2 vezes nas páginas 18 e 31.

Apêndices

APÊNDICE A – Códigos do Matlab Utilizados

A.1 Parâmetros FOPTD - Identificação

Código em MATLAB para encontrar a estimação do parâmetros de um sistema de 1º ordem com atraso, a partir de dados experimentais, por meio do método matemático dos mínimos quadrados.

```

1 function [K, T, L] = parametrosFOPTD(u,y,Deltat)
2
3 Theta = zeros(1,3);
4 R = zeros(size(Theta,2));
5 f = zeros(size(Theta,2),1);
6
7 h = max(u);
8 for k=1:length(y)
9     Phi = [h*k*Deltat -h -y(k)]';
10    R = R + Phi*Phi';
11    A = sum(y(1:k)) * Deltat;
12    f = f + Phi*A;
13 end
14 Theta = R\f;
15 K = Theta(1);
16 L = Theta(2)/K;
17 T = Theta(3);
18 end

```

A.2 Parâmetros k_p e τ_i - SIMC-PID

Código em MATLAB para encontrar os parâmetros de um controlador PI por meio das regras de sintonia SIMC-PID, a partir de um sistema de 1º ordem com atraso.

```

1 function [Kp,Ti] = Simc1Ordem(G,MultiplicadorTauC_L)
2
3 K = G.num{1}(2);
4 T = G.den{1}(1);
5 L = G.OutputDelay;

```

```

6 tauC = MultiplicadorTauC_L*L;
7
8 Kp = (1/K)*(T/(tauC+L));
9 Ti = min(T,4*(tauC+L));
10 end

```

A.3 Otimização - Critérios \mathcal{H}_∞

Código em MATLAB para encontrar uma solução para o problema de otimização que utiliza os critérios \mathcal{H}_∞ . O critério a ser minimizado foi o J_v , e os critérios sujeitos à restrições são o M_S e o M_T . No código, o MATLAB calcula a os parâmetros do controlador ótimo por meio da rotina *fmincon*.

```

1 clc, clear, close all;
2
3 %% Frequencias limites do grafico de Bode
4 % Bode Utilizado p/ encontrar a norma do inf
5 bodeFreqMin = 1e-10;
6 bodeFreqMax = 20;
7 bodeFreq = {bodeFreqMin,bodeFreqMax};
8
9 %% Definicao do sistema
10 K = 0.3764;
11 T = 190.2;
12 L = 9.61;
13 G = tf(K,[T 1]);
14 G.outputd = L;
15
16 %% Controlador inicial
17 [GParametros(1), GParametros(2)] = Simc1Ordem(G(1,1),1);
18 Kp = GParametros(1);
19 Ti = GParametros(2);
20 Ki = Kp/Ti;
21
22 kIniciais = [Kp; Ki];
23 K = pid(Kp, Ki);
24
25 %% Funcoes de transferencia - vy, wu, S e T do controlador Inicial
26 %Utilizado para calculo dos criterios de robustez da condicao inicial
27 s = tf('s');
28 Gvy = G / (1 + G*K);
29 Gwu = K / (1 + G*K);
30 S = 1 / (1+G*K);
31 T = 1 - S;

```



```
32
33 %% Calculo dos criterios de robustez para a condicao inicial
34 JvSIMC = max(bode(Gvy/s,bodeFreq));
35 MsSIMC = max(bode(S,bodeFreq));
36 MtSIMC = max(bode(T,bodeFreq));
37
38 %% Restricoes do problema de otimizacao - Jv otimizado
39 Ms = 1.7;      %MsSIMC;
40 Mt = 1.3;      %MtSIMC;
41
42 %% Busca do controlador otimo
43 options = optimoptions('fmincon', 'Display', 'iter');
44 A = [];      b = [];
45 Aeq = [];    beq = [];
46 lb = [0;0];  ub = [];
47
48 GanhosK = fmincon(@(GanhosK) objfunJv(GanhosK,G,bodeFreq),kIniciais, ...
49     A, b, Aeq, beq, lb, ub, @(GanhosK) ...
50     nonLinCon(GanhosK, G, Ms, Mt, bodeFreq), options);
51
52 KpOpt = GanhosK(1);
53 KiOpt = GanhosK(2);
54
55 %% Calculo dos criterios de robustez para o controlador final
56 JvOpt = objfunJv(GanhosK,G,bodeFreq);
57 C = nonLinCon(GanhosK, G, Ms, Mt,bodeFreq);
58 MsOpt = C(1)+Ms;
59 MtOpt = C(2)+Mt;
60
61 %%Apresentar Resultados
62 fprintf('\n\n\n');
63 fprintf('JvOpt = %.4f , JvInicial = %.4f\n',JvOpt,JvSIMC);
64 fprintf('MsOpt = %.4f , MsDefinido = %.4f , MsCInicial: ...
65     %.4f\n',MsOpt,Ms,MsSIMC);
66 fprintf('MtOpt = %.4f , MtDefinido = %.4f , MtCInicial: ...
67     %.4f\n\n',MtOpt,Mt,MtSIMC);
68
69 fprintf('Controlador Inicial: Kp = %.4f, Ki: ...
70     %.4f\n',kIniciais(1),kIniciais(2));
71 fprintf('Controlador Otimo: Kp = %.4f, Ki: ...
72     %.4f\n\n',GanhosK(1),GanhosK(2));
73
74 % Grafico: Resposta do sistema com controlador inicial
75 % Sistema Inicial
76 sysInicial = feedback(G*K,1);
77 subplot(2,1,1);
```

```
75 step(sysInicial);
76 % Grafico: Resposta do sistema com controlador otimo
77 KOpt = pid(GanhosK(1), GanhosK(2));
78 sysOpt = feedback(G*KOpt,1);
79 subplot(2,1,2);
80 step(sysOpt);
81
82 %% Funcoes Utilizadas
83
84 % Funcao objetivo do problema de otimizacao
85 function Jv = objfunJv(k, G,bodeFreq)
86     K = pid(k(1), k(2));
87     Gvy = G/(1+G*K);
88     s = tf('s');
89     Jv = max(bode(Gvy/s,bodeFreq));
90 end
91
92 % Restricoes nao lineares do problema de otimizacao
93 function [C, Ceq] = nonLinCon(k, G, Ms, Mt,bodeFreq)
94     K = pid(k(1), k(2));
95     S = 1/(1 + G*K);
96     T = G*K*S;
97     Snorm = max(bode(S,bodeFreq));
98     Tnorm = max(bode(T,bodeFreq));
99     C = [Snorm - Ms; Tnorm - Mt];
100     Ceq = [];
101 end
```