

Projeto de Identificação de Sistemas e Controle PID

Guia para Estudantes de C213

Disciplina: Sistemas de Controle Automático

24 de setembro de 2025

Sumário

1	Controle de Processos Industriais	3
2	Modelagem Matemática de Processos	3
3	Resposta Típica de Sistemas de Primeira Ordem	4
4	Resposta Típica de Sistemas de Segunda Ordem	4
5	Malhas de Controle	4
5.1	Malha Aberta	5
5.2	Malha Fechada	5
6	Atraso de Transporte	5
6.1	Aproximação de Padé	5
7	Ensaio da Curva de Reação e Métodos de Identificação de Processos	5
7.1	Identificação de Sistemas	6
7.1.1	Método de Identificação de Smith	6
7.1.2	Método de Identificação de Sundaresan	6
7.2	Métricas de Avaliação de Identificação	6
7.3	Método de Ziegler-Nichols Malha Aberta - Curva de Reação	6
7.4	Método do Modelo Interno - IMC	7
7.5	Método CHR	7
7.6	Método Cohen e Coon	7
7.7	Método ITAE	7
8	Sistemas de Supervisão e Interface Homem-Máquina	8
9	Contexto e Objetivos	10
9.1	Objetivos do Projeto	10

10 Especificações do Projeto	10
10.1 Arquitetura da Aplicação	10
10.2 Funcionalidades Requeridas	11
10.2.1 Aba de Identificação	11
10.2.2 Aba de Controle PID	11
10.3 Requisitos Técnicos	12
11 Critérios de Avaliação	12
11.1 Funcionalidade (40%)	12
11.2 Interface do Usuário (25%)	13
11.3 Código e Documentação (20%)	13
11.4 Análise e Resultados (15%)	13
12 Entregáveis	13
12.1 Código Fonte	13
12.2 Documentação	13
12.3 Apresentação	14
12.4 GitHub	14
13 Recursos Adicionais	14
13.1 Links Úteis	14

1 Controle de Processos Industriais

O controle de processos industriais é uma área fundamental da engenharia que busca otimizar o desempenho de sistemas dinâmicos por meio do monitoramento e ajuste de variáveis críticas. Em contextos como a agricultura de precisão, o controle de variáveis ambientais, como temperatura e umidade em estufas inteligentes, desempenha um papel crucial na maximização da eficiência operacional, redução de custos e garantia da qualidade do produto. Por exemplo, em estufas, a regulação precisa da temperatura pode influenciar diretamente o crescimento de culturas, enquanto a umidade afeta a saúde das plantas e a prevenção de doenças.

O controlador PID (Proporcional–Integral–Derivativo) é uma das estratégias mais utilizadas devido à sua simplicidade, robustez e capacidade de lidar com sistemas dinâmicos, especialmente aqueles modelados como de primeira ordem com atraso de transporte (FOPDT). Este controlador ajusta a variável manipulada com base em três componentes: o erro proporcional, que corrige a diferença instantânea entre o setpoint e a saída; o erro integral, que elimina o erro em regime permanente; e o erro derivativo, que antecipa variações rápidas no sistema. A eficácia do PID depende de uma modelagem precisa do processo e da sintonização adequada de seus parâmetros (K_p , T_i , T_d).

Além disso, a automação de processos industriais permite a integração com sistemas de supervisão e interfaces homem-máquina (IHM), que facilitam a interação do operador com o sistema, fornecendo dados em tempo real e permitindo ajustes dinâmicos. Este projeto visa aplicar esses conceitos em um sistema térmico de estufa, combinando teoria de controle clássico com ferramentas computacionais para análise e visualização.

2 Modelagem Matemática de Processos

A modelagem matemática é o processo de representar o comportamento dinâmico de um sistema por meio de equações que descrevem a relação entre entradas e saídas. Em sistemas de controle, a função de transferência $G(s)$ no domínio de Laplace é amplamente utilizada para modelar processos lineares e invariantes no tempo. Para sistemas térmicos, como o controle de temperatura em estufas, o modelo FOPDT é frequentemente adotado devido à sua capacidade de capturar as dinâmicas de ganho, resposta temporal e atraso de transporte:

$$G(s) = \frac{k}{\tau s + 1} \cdot e^{-\theta s}$$

onde:

- k : Ganho estático, que representa a relação entre a variação da saída (Δy) e a variação da entrada (Δu), calculado como $k = \frac{\Delta y}{\Delta u}$.
- τ : Constante de tempo, que indica o tempo necessário para a saída atingir 63,2% do valor final após o atraso.
- θ : Atraso de transporte, que representa o intervalo de tempo antes que o sistema comece a responder à entrada.

A modelagem precisa é essencial para simulações, análises de estabilidade e projeto de controladores. Ferramentas como MATLAB (`tf`) ou Python (`control.tf`) permitem construir e simular essas funções de transferência, facilitando a análise do comportamento do sistema.

3 Resposta Típica de Sistemas de Primeira Ordem

Sistemas de primeira ordem, como circuitos RC, sistemas térmicos ou hidráulicos, são caracterizados por uma resposta suave e sem oscilações a entradas do tipo degrau. A função de transferência canônica é:

$$G(s) = \frac{k}{\tau s + 1}$$

A resposta ao degrau exibe uma ascensão exponencial, atingindo 63,2% do valor final em $t = \tau$, 86,5% em $t = 2\tau$, e aproximadamente 98% em $t = 4\tau$, segundo o critério de 2% para regime permanente. As métricas principais incluem:

- **Tempo de Subida (t_r):** Intervalo para a saída variar de 10% a 90% do valor final.
- **Tempo de Acomodação (t_s):** Tempo para a saída estabilizar dentro de $\pm 2\%$ do valor final.
- **Valor Final (VF):** Calculado pelo Teorema do Valor Final:

$$VF = \lim_{s \rightarrow 0} s \cdot SP(s) \cdot G(s)$$

4 Resposta Típica de Sistemas de Segunda Ordem

Sistemas de segunda ordem, como circuitos RLC ou sistemas com controladores PID, apresentam respostas mais complexas, que podem incluir oscilações. A função de transferência canônica é:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

onde:

- ω_n : Frequência natural, que determina a velocidade da resposta.
- ξ : Fator de amortecimento, que controla a presença e amplitude das oscilações.

A resposta ao degrau depende de ξ :

- $\xi < 1$: Subamortecido, com oscilações que diminuem com o tempo.
- $\xi = 1$: Criticamente amortecido, com resposta mais rápida sem oscilações.
- $\xi > 1$: Superamortecido, com resposta lenta e sem oscilações.

O *overshoot* (M_p) é a porcentagem pela qual a saída ultrapassa o valor final, sendo mais pronunciado em sistemas subamortecidos. O tempo de pico e a frequência das oscilações também são influenciados por ω_n e ξ .

5 Malhas de Controle

Os sistemas de controle operam em configurações de malha que definem como os sinais são processados para alcançar o desempenho desejado.

5.1 Malha Aberta

Em sistemas de malha aberta, a saída ($PV(s)$) não é realimentada para ajustar a entrada ($SP(s)$). A função de transferência é simplesmente $G(s) = \frac{PV(s)}{SP(s)}$. Essa configuração é simples e adequada para sistemas com pouca variabilidade, mas não compensa perturbações externas ou mudanças nas condições do processo.

5.2 Malha Fechada

Em malha fechada, a saída é comparada com o setpoint para gerar um sinal de erro ($E(s) = SP(s) - PV(s)$), que é usado pelo controlador para ajustar a entrada. A função de transferência do sistema é:

$$G(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)}$$

onde $G_c(s)$ é a função de transferência do controlador (e.g., PID) e $G_p(s)$ é a da planta. A realimentação melhora a robustez, reduz o erro em regime permanente e aumenta a estabilidade, mas pode introduzir complexidades como oscilações se mal projetada.

6 Atraso de Transporte

O atraso de transporte (θ) é o tempo necessário para que uma mudança na entrada se reflita na saída, comum em sistemas térmicos, hidráulicos ou de redes devido a inércias físicas ou delays de comunicação. No modelo FOPDT, o termo $e^{-\theta s}$ representa esse atraso, impactando a estabilidade e a sintonização do controlador, especialmente em malha fechada. Por exemplo, em uma estufa, o atraso pode surgir devido à distância entre o aquecedor e o sensor de temperatura.

6.1 Aproximação de Padé

A aproximação de Padé converte o termo irracional $e^{-\theta s}$ em uma fração racional para facilitar simulações em ferramentas como o Python:

$$R_N(s) = \frac{Q_N(-\theta s)}{Q_N(\theta s)}, \quad Q_N(s) = \sum_{j=0}^N \frac{(N+j)!}{j!(N-j)!} s^{N-j}$$

A ordem N determina a precisão da aproximação, com ordens mais altas oferecendo maior fidelidade, mas aumentando a complexidade computacional.

7 Ensaio da Curva de Reação e Métodos de Identificação de Processos

O ensaio da curva de reação é um experimento em malha aberta que aplica uma entrada degrau ao sistema e registra a resposta da saída ao longo do tempo. Esse ensaio é usado para identificar os parâmetros do modelo FOPDT (k , τ , θ) a partir de dados experimentais, permitindo a construção de uma função de transferência precisa.

7.1 Identificação de Sistemas

A identificação de sistemas envolve ajustar um modelo matemático aos dados experimentais para capturar o comportamento dinâmico do processo. Para o modelo FOPDT, os parâmetros são estimados com base na resposta ao degrau:

$$k = \frac{\Delta y}{\Delta u}, \quad \Delta u = V_1 - V_0$$

Os métodos de Smith e Sundaresan são amplamente utilizados para estimar τ e θ .

7.1.1 Método de Identificação de Smith

Proposto por Cecil L. Smith em 1972 [1], o método de Smith traça uma linha tangente na inflexão da curva de resposta, usando os tempos correspondentes a 28,3% (t_1) e 63,2% (t_2) do valor final:

$$\tau = 1.5 \cdot (t_2 - t_1), \quad \theta = t_2 - \tau$$

Este método é eficaz para sistemas com atrasos moderados, mas pode ser sensível a ruídos na inflexão.

7.1.2 Método de Identificação de Sundaresan

Desenvolvido por Sundaresan e Krishnaswamy em 1978 [2], este método usa os tempos correspondentes a 35,3% (t_1) e 85,3% (t_2) do valor final, sendo mais robusto a ruídos devido à não dependência de tangentes:

$$\tau = \frac{2}{3} \cdot (t_2 - t_1), \quad \theta = 1.3t_1 - 0.29t_2$$

7.2 Métricas de Avaliação de Identificação

O erro quadrático médio (EQM) é usado para avaliar a precisão do modelo identificado em relação aos dados experimentais:

$$EQM = \sqrt{\frac{\sum_{i=1}^n [\hat{y}_i(t) - y_i(t)]^2}{n}}$$

onde $\hat{y}_i(t)$ é a saída do modelo identificado e $y_i(t)$ é a saída experimental. O método com menor EQM é considerado o mais adequado.

7.3 Método de Ziegler-Nichols Malha Aberta - Curva de Reação

O método de Ziegler-Nichols em malha aberta utiliza os parâmetros do ensaio da curva de reação para sintonizar o controlador PID:

$$K_p = \frac{1.2\tau}{k\theta}, \quad T_i = 2\theta, \quad T_d = \frac{\theta}{2}$$

Este método é projetado para garantir uma taxa de decaimento de aproximadamente $\frac{1}{4}$, mas pode resultar em respostas com *overshoot* significativas.

7.4 Método do Modelo Interno - IMC

O método IMC [4] utiliza a função de transferência do processo para sintonizar o controlador PID, com um parâmetro ajustável λ que controla a velocidade da resposta:

Controlador	K_p	T_i	T_d	Critério Desempenho
PID	$\frac{2\tau + \theta}{k \cdot (2\lambda + \theta)}$	$\tau + \frac{\theta}{2}$	$\frac{\tau \cdot \theta}{2\tau + \theta}$	$\frac{\lambda}{\theta} > 0.8$

Tabela 1: Parâmetros do controlador PID e critério de desempenho.

Valores menores de λ produzem respostas mais rápidas, mas menos robustas, enquanto valores maiores aumentam a estabilidade à custa de maior lentidão.

7.5 Método CHR

O método CHR [5] oferece duas configurações para o problema servo:

- **Sem overshoot:**

$$K_p = \frac{0.6\tau}{k\theta}, \quad T_i = \tau, \quad T_d = \frac{\theta}{2}$$

- **20% de overshoot:**

$$K_p = \frac{0.95\tau}{k\theta}, \quad T_i = 1.357\tau, \quad T_d = 0.473\theta$$

Este método é útil para balancear velocidade e estabilidade, dependendo dos requisitos do sistema.

7.6 Método Cohen e Coon

O Método de Cohen e Coon foi proposto para Sintonia de Sistemas com Atraso de Transporte mais elevados, quando o processo possui Fator de Incontrolabilidade maior que 0.3. Assim como o Método de Ziegler-Nichols, o modelo é resultado do Ensaio da Curva de Reação. As regras de sintonia são como na Tabela 2.

Tabela 2: Parâmetros do PID para a Sintonia pelo Método de Cohen e Coon

Controlador	K_p	T_i	T_d
PID	$\frac{\tau}{k \cdot \theta} \cdot \frac{16\tau + 3\theta}{12\tau}$	$\theta \cdot \frac{32 + 6\theta/\tau}{13 + 8\theta/\tau}$	$\frac{4\theta}{11 + 2\theta/\tau}$

7.7 Método ITAE

O ITAE é um Índice de Desempenho baseado na minimização do erro, resultado da Integral da Multiplicação do Erro Absoluto pelo Tempo (*ITAE – Integral of Time multiplied by Absolute Error*). Um Sistema com Controle ITAE reduz grandes erros iniciais, assim como erros posteriores ocorridos na resposta transitória, tornando as oscilações bem amortecidas e reduzindo consideravelmente o *overshoot*.

As regras para Sintonia ITAE apresentadas na Tabela 3 são regidas por seis constantes, com $A = 0.965$, $B = -0.85$, $C = 0.796$, $D = -0.147$, $E = 0.308$ e $F = 0.929$.

Tabela 3: Parâmetros do PID para a Sintonia ITAE

Controlador	K_p	T_i	T_d
PID	$\frac{A}{k} \cdot \left(\frac{\theta}{\tau}\right)^B$	$\frac{\tau}{C + D \cdot (\theta/\tau)}$	$\tau \cdot E \cdot \left(\frac{\theta}{\tau}\right)^F$

8 Sistemas de Supervisão e Interface Homem-Máquina

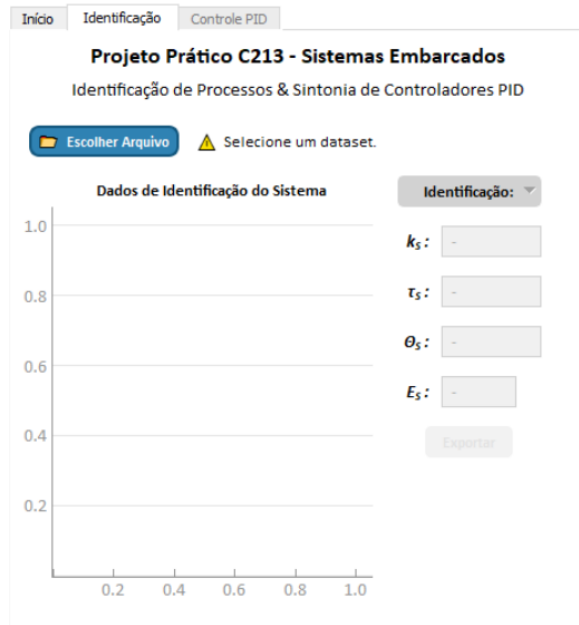
Sistemas de supervisão e interfaces homem-máquina (IHM) são ferramentas essenciais para o monitoramento e controle em tempo real de processos industriais. A IHM permite aos operadores visualizar dados, configurar setpoints e analisar métricas de desempenho, como tempo de subida, *overshoot* e erro em regime permanente. No contexto deste projeto, a IHM deve ser desenvolvida em Python (usando PyQt5 e Qt Designer) ou MATLAB e incluir:

A IHM multi-abas das Figuras 1a e 1b apresenta uma interface de exemplo para sistemas com Controle PID, permitindo diversas interações do usuário. Na segunda tela, é possível selecionar um conjunto de dados base para o processo aplicado, realizar a identificação pelos Métodos de Smith e Sundaresan e verificar os resultados. Na terceira tela, há duas seleções para definir a forma de Sintonia do Controlador PID.

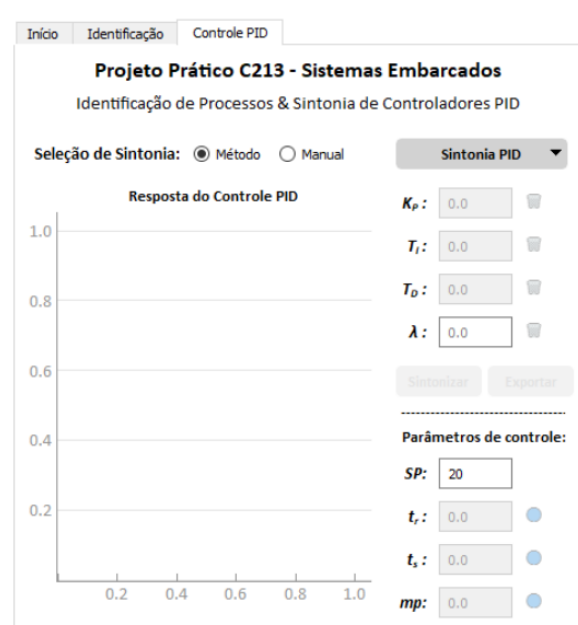
- O conjunto de dados selecionado em 1a é verificado como válido e apresenta o resultado para o Método de Identificação com menor EQM. É possível selecionar os dados de qualquer método em uma lista suspensa. A seleção de dados libera o acesso à aba 11b;
- Em 1b, a opção *Método* permite que o usuário selecione um entre os Métodos Clássicos de Sintonia. Nessa situação, os parâmetros K_p , T_i e T_d são calculados e não podem ser editados. Apenas é permitida entrada para o parâmetro λ caso seja selecionado o Método IMC;
- Em 1b, se selecionada a opção *Manual*, os valores dos parâmetros K_p , T_i e T_d podem ser editados, enquanto a seleção de Métodos Clássicos e edição para λ são bloqueadas. Há ícones que permitem a limpeza dos valores digitados neste modo.

Há dois botões de ação no sistema. Se habilitada a opção *Manual*, o botão *Sintonizar* realiza o Controle PID com parâmetros manuais, previamente verificando a estabilidade do processo para o conjunto de valores. Após a sintonia, por qualquer forma válida, o botão *Exportar* salva o gráfico como uma imagem, permitindo acesso futuro aos dados.

Na seção *Controle*, um campo de entrada permite ao usuário definir o valor para o SetPoint considerando uma entrada do tipo degrau. O valor inicial para o SetPoint é o mesmo do conjunto de dados do processo analisado, selecionado em 11a. A seguir, demais campos parametrizam os pontos de interesse da curva de resposta: tempos de subida e acomodação e índice de overshoot. As interações do sistema permitem marcar pontos com descrição de dados no gráfico.



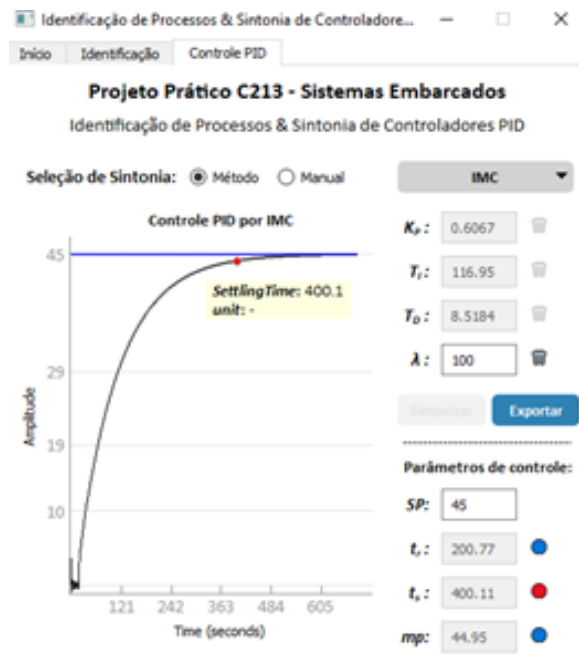
(a) Aba 'Identificação' para verificação de conjuntos de dados.



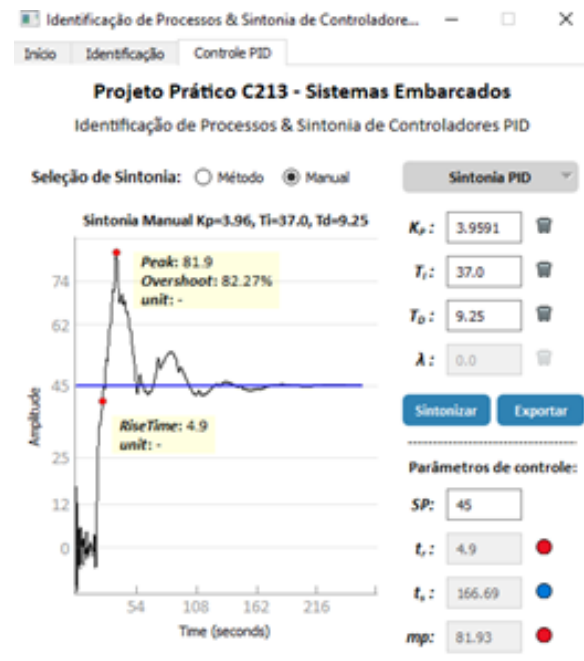
(b) Aba 'Controle PID' para parametrização do Controlador e métricas de resposta.

Figura 1: Descrição da aplicação da IHM para sistemas com Controle PID.

As Figuras 2a e 2b apresentam o funcionamento da interface para as duas opções de seleção da forma de Sintonia, por Métodos ou Manual, respectivamente, além da alteração do SetPoint e indicação dos pontos e valores de referência. Se o conjunto de dados possuir metadados de unidades de medida ou grandezas, as métricas são incluídas na figura e labels.



(a) Funcionamento da Interface para seleção de Métodos Clássicos de Sintonia.



(b) Funcionamento da Interface para seleção de parâmetros manuais.

Figura 2: Detalhamento das situações de aplicação da interface.

9 Contexto e Objetivos

O presente projeto tem como objetivo aplicar tais fundamentos matemáticos e metodológicos no desenvolvimento de uma aplicação computacional que permita aos estudantes: carregar um dataset experimental, identificar o modelo FOPDT da planta térmica, implementar diferentes técnicas de sintonia PID e visualizar o desempenho do sistema em malha aberta e malha fechada. Assim, busca-se integrar a teoria de controle clássico com uma aplicação prática e didática que facilite a compreensão dos efeitos de cada estratégia de sintonia em um processo real.

9.1 Objetivos do Projeto

Este projeto tem como finalidade que os estudantes desenvolvam uma aplicação completa de identificação de sistemas e projeto de controladores PID, integrando conhecimentos teóricos e habilidades práticas de programação. Ao completar este projeto, os estudantes serão capazes de:

- Carregamento de datasets experimentais (`.mat`).
- Identificação por Smith da curva em malha aberta e exibição dos parâmetros identificados (k , τ , θ).
- Projetar controladores PID usando métodos clássicos de sintonia
- Seleção de métodos de sintonização (Ziegler-Nichols, IMC, CHR, CC e ITAE) de acordo com excel dos grupos.
- Visualização de gráficos em tempo real, comparando respostas em malha aberta e fechada controlada.
- Painel de métricas: tempo de subida (t_r), tempo de acomodação (t_s), *overshoot* (M_p) e erro em regime permanente (malha aberta).
- Ajuste fino dos parâmetros PID
- Criação de uma interface gráfica
- Analisar respostas de sistemas de controle e suas características
- Visualizar dados e resultados de forma efetiva

10 Especificações do Projeto

10.1 Arquitetura da Aplicação

A aplicação deve seguir o padrão Modelo-Vista-Controlador (MVC) com os seguintes componentes:

1. **Interface do Usuário:** Três abas principais

- Aba de Identificação de Sistemas

- Aba de Controle PID
- Aba de Gráficos
- Aba de Login/Autenticação (Opcional)

2. Módulos de Processamento:

- Carregamento e processamento de dados
- Algoritmos de identificação
- Métodos de sintonia PID
- Simulação de respostas

3. Visualização:

- Gráficos interativos
- Legendas automáticas
- Marcadores de pontos importantes
- Exportação de figuras

10.2 Funcionalidades Requeridas

10.2.1 Aba de Identificação

- Carregamento de arquivos .mat com dados experimentais
- Visualização de dados de entrada e saída
- Implementação dos métodos Smith
- Cálculo automático do EQM (Erro Quadrático Médio)
- Exportação de gráficos

10.2.2 Aba de Controle PID

- Seleção entre sintonia automática e manual
- Implementação de múltiplos métodos de sintonia:
 - Ziegler-Nichols MA
 - IMC (com parâmetro λ)
 - CHR sem sobressinal
 - CHR com sobressinal
 - Cohen e Coon
 - ITAE
- Simulação de resposta do sistema controlado
- Cálculo automático de características de resposta

- Marcadores visuais em pontos importantes
- Legendas identificativas nos gráficos

Sintonize um controlador PID de acordo com os métodos especificados para seu grupo na Tabela 4. Em seguida, verifique o comportamento do sistema controlado por meio da resposta ao degrau unitário e analise o desempenho em termos de: Para a sintonia baseada no modelo de *Internal Model Control* (IMC), a escolha do parâmetro λ é livre e deve ser justificada de acordo com o desempenho desejado. Em geral:

- Valores pequenos de λ resultam em uma resposta mais rápida, mas com maior sensibilidade a ruídos e incertezas do modelo.
- Valores maiores de λ tornam a resposta mais lenta, mas garantem maior robustez e menor sobreleitura devido ao atraso.

Assim, a seleção de λ deve equilibrar a rapidez da resposta e a robustez do sistema controlado, devendo ser explicitamente justificada no relatório de cada grupo.

GRUPO	MÉTODO
1	CHR + ITAE
2	Ziegler-Nichols + CC
3	CHR com sobresinal + ITAE
4	IMC + CC
5	Ziegler-Nichols + ITAE
6	CHR + CC
7	IMC + ITAE
8	CHR com sobresinal + CC
9	Ziegler-Nichols + ITAE
10	CHR + CC

Tabela 4: Distribuição dos grupos e métodos de sintonia PID.

10.3 Requisitos Técnicos

- **Linguagem:** Python 3.8 ou superior- MATLAB 2024Ra
- **Controle:** biblioteca python-control

11 Critérios de Avaliação

11.1 Funcionalidade (40%)

- Implementação correta dos algoritmos de identificação
- Funcionamento dos métodos de sintonia PID
- Simulação precisa das respostas do sistema
- Cálculo correto das métricas de desempenho

11.2 Interface do Usuário (25%)

- Design intuitivo e profissional
- Navegação fluida entre abas
- Gráficos claros e informativos
- Legendas e marcadores apropriados

11.3 Código e Documentação (20%)

- Código bem estruturado e comentado
- Uso apropriado de programação orientada a objetos
- Tratamento adequado de erros
- Documentação clara das funções

11.4 Análise e Resultados (15%)

- Interpretação correta dos resultados
- Comparação entre métodos de identificação
- Análise dos efeitos de diferentes sintonias
- Apresentação profissional dos resultados

12 Entregáveis

12.1 Código Fonte

- Código completo da aplicação
- Arquivos de configuração e dependências
- Datasets de teste
- README com instruções de instalação e uso

12.2 Documentação

- Documentação técnica do código
- Explicação dos algoritmos implementados
- Análise dos resultados obtidos

12.3 Apresentação

- Demonstração ao vivo da aplicação
- Explicação das decisões de design
- Análise comparativa dos métodos
- Discussão sobre limitações e melhorias futuras

12.4 GitHub

- O projeto pode ser realizado em **Python (preferencial)** ou em **MATLAB**.
- O repositório GitHub deve conter código-fonte, documentação (README.md), prints da IHM e resultados.
- Cada integrante deve realizar commits significativos.
- Será valorizada a clareza da explicação matemática, organização do código e qualidade da interface gráfica.

13 Recursos Adicionais

13.1 Links Úteis

- Documentação PyQt5: <https://doc.qt.io/qtforpython/>
- Documentação PyQtGraph: <https://pyqtgraph.readthedocs.io/>
- Biblioteca Python Control: <https://python-control.readthedocs.io/>
- Documentação NumPy/SciPy: <https://numpy.org/doc/>

Referências

- [1] Smith, C. L., *Digital Computer Process Control*, Intext Educational Publishers, 1972.
 - [2] Sundaresan, K. R., Krishnaswamy, P. R., *Estimation of time delay parameters in linear systems*, Industrial & Engineering Chemistry Process Design and Development, 1978.
 - [3] Ziegler, J. G., Nichols, N. B., *Optimum settings for automatic controllers*, Transactions of the ASME, 1942.
 - [4] Rivera, D. E., Morari, M., Skogestad, S., *Internal Model Control: PID Controller Design*, Industrial & Engineering Chemistry Process Design and Development, 1986.
 - [5] Chien, K. L., Hrones, J. A., Reswick, J. B., *On the automatic control of generalized passive systems*, Transactions of the ASME, 1952.
- Ogata, K. (2010). *Engenharia de Controle Moderno*. 5ª Edição.

- Åström, K. J., & Hägglund, T. (2006). *Controle PID Avançado*.
- Seborg, D. E., Edgar, T. F., & Mellichamp, D. A. (2016). *Dinâmica e Controle de Processos*. 4^a Edição.
- Dorf, R. C., & Bishop, R. H. (2016). *Sistemas de Controle Modernos*. 13^a Edição.