



UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
TRABALHO DE CONCLUSÃO EM ENGENHARIA DE  
CONTROLE E AUTOMAÇÃO



# **Estudo e refinamento do algoritmo dos mínimos quadrados recursivo aplicado em controle adaptativo**

*Autor: Pablo Knecht*

*Orientador: Profa. Dra. Lucíola Campestrini*

Porto Alegre, 2 de janeiro de 2018



# Sumário

<b>Agradecimentos</b>	<b>iii</b>
<b>Resumo</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Lista de Figuras</b>	<b>ix</b>
<b>Lista de Tabelas</b>	<b>xi</b>
<b>Lista de Abreviaturas e Siglas</b>	<b>xii</b>
<b>Lista de Símbolos</b>	<b>xiv</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Revisão Bibliográfica</b>	<b>3</b>
<b>3 Metodologia</b>	<b>5</b>
3.1 Controlador auto-ajustável . . . . .	5
3.1.1 Preditor de Smith e controlador a modelo interno . . . . .	5
3.1.2 Algoritmo dos Mínimos Quadrados Recursivo . . . . .	7
3.1.3 Exemplo de estimação de parâmetros de um sistema de primeira ordem . . . . .	9
3.2 Problemas práticos e correções do Controlador Auto-Ajustável . . . . .	10
3.2.1 <i>Wind-up</i> do estimador e atualização condicional . . . . .	10
3.2.2 Robustez da estimativa . . . . .	11
3.2.3 Projeção de parâmetros . . . . .	12
3.2.4 Filtro passa-altas . . . . .	12
3.3 Simulação do Controlador Auto-ajustável . . . . .	13
3.4 Controle adaptativo baseado em modelo de referência (MRAC) . . . . .	14
3.4.1 Formulação do MRAC . . . . .	15
3.4.2 Projeto de um MRAC para um sistema de primeira ordem . . . . .	15
<b>4 Resultados</b>	<b>17</b>
4.1 Simulações do Controlador Auto-Ajustável . . . . .	17
4.1.1 Identificação de parâmetros e ajuste automático do controlador . . . . .	17
4.1.2 <i>Wind-up</i> do estimador e atualização condicional . . . . .	17
4.1.3 Estimação robusta . . . . .	19
4.1.4 Utilização de múltiplos estimadores . . . . .	21

---

4.2	Implementação do MQR para CLP . . . . .	23
4.3	Simulações do MRAC . . . . .	24
<b>5</b>	<b>Conclusão</b>	<b>29</b>
5.1	Trabalhos Futuros . . . . .	30
<b>6</b>	<b>Referências</b>	<b>31</b>
	<b>Apêndices</b>	<b>33</b>
A	Algoritmo dos Mínimos Quadrados Recursivos para CLP . . . . .	34
B	MRAC com Preditor de Smith . . . . .	38

## **Agradecimentos**

Um muito obrigado especial a meus amigos de faculdade tanto aqui quanto na França, pela motivação e por tornarem meu dia a dia mais agradável. Agradeço a meu tutor de estágio, Stefan Capitaneanu e minha tutora de TCC, Lucíola Campestrini pela oportunidade e pela confiança depositada em mim.



## Resumo

Dois métodos de controle adaptativo foram explorados com o objetivo de aplicá-los em processos: o controlador auto-ajustável e o controle adaptativo por modelo de referência. O controlador auto-ajustável utiliza o algoritmo dos mínimos quadrados recursivo para estimar os parâmetros da planta, um preditor de Smith para compensar o tempo morto, e o controlador a modelo interno que é ajustado em função da estimativa de parâmetros. Como o estimador de parâmetros apresenta alguns problemas quando confrontado com fatores presentes na prática como ruído, perturbações e valor constante adicionado à saída do processo, o algoritmo é modificado seguindo técnicas presentes na literatura e por fim testado para um sistema de primeira ordem. São mostrados principalmente os efeitos do *wind-up* do estimador e a influência de perturbações na estimativa, aspectos estes que foram contornados com as modificações. Se bem ajustado, o controlador auto-ajustável cumpre a tarefa de manter o desempenho de controle em caso de mudança de parâmetros do processo e as modificações feitas no estimador se mostram eficazes nas simulações realizadas e no teste em um processo de controle de temperatura. Já o controle adaptativo por modelo de referência, nas simulações realizadas em um sistema de primeira ordem, se mostra eficiente no quesito adaptação de parâmetros mas seu desempenho é prejudicado caso o processo possua tempo morto.





## **Abstract**

Two methods of adaptive control were explored with the objective of applying them to processes: the self-tuning controller and the model reference adaptive control. The self-tuning controller uses the recursive least squares algorithm to estimate the plant's parameters, a Smith predictor to compensate for dead time, and an internal model controller that is adjusted as a function of the parameter estimation. As the parameter estimator presents some problems when faced with practical factors such as noise, disturbances and constant value added to the output of the process, the algorithm is modified following techniques present in the literature and finally tested for a first order system. The effects of the wind-up of the estimator and the influence of perturbations in the estimation are shown, as these aspects have been corrected with the modifications. If properly adjusted, the self-adjusting controller fulfills the task of maintaining control performance in case of process parameter changes and the modifications made in the estimator prove to be effective in the simulations performed and in the test for a temperature control process. On the other hand, the model reference control, in the simulations performed in a first order system, is efficient in the parameter adaptation, but its performance is impaired if the process has dead time.



## Lista de Figuras

1	Diagrama de blocos simplificado do sistema de controle auto-ajustável proposto. . . . .	5
2	Diagrama de blocos de um sistema controlado por um preditor de Smith. . . . .	6
3	Diagrama de blocos de um sistema controlado por um Controlador a Modelo Interno. . . . .	6
4	Diagrama de blocos de um sistema controlado por um preditor de Smith a modelo interno. . . . .	6
5	Esquema simplificado do funcionamento do algoritmo de MQR. . . . .	8
6	Diagrama de simulação Matlab/Simulink. . . . .	13
7	Diagrama completo de um sistema de primeira ordem controlado por um MRAC. . . . .	16
8	Simulação mostrando a identificação dos parâmetros e a adaptação do controlador para um caso ideal. . . . .	18
9	Simulação mostrando a comparação entre o controlador adaptativo e o não adaptativo para um caso ideal. . . . .	19
10	Simulação mostrando o wind-up do estimador em um caso sem excitação permanente. . . . .	20
11	Simulação do sistema em um caso sem excitação permanente e atualização condicional. . . . .	21
12	Simulação do sistema mostrando o efeito da constante de robustez com perturbações. . . . .	22
13	Simulação do sistema mostrando o efeito da constante de robustez variação de parâmetros. . . . .	23
14	Simulação do sistema utilizando três estimadores diferentes. . . . .	24
15	Estimação do ganho estático $k$ para um regulador de temperatura. Imagem cedida por <i>Schneider Electric</i> . . . . .	25
16	Simulação do controlador MRAC com parâmetros $\gamma = 0,5$ . . . . .	26
17	Simulação do controlador MRAC com parâmetros $\gamma = 2$ . . . . .	26
18	Simulação de um controlador PI com variação de parâmetros do processo. . . . .	27
19	Simulação do controlador MRAC sem adaptação dos ganhos $k_r$ e $k_x$ . . . . .	28
20	Diagrama de blocos do sistema de controle composto pela fusão de um preditor de Smith com o MRAC. Este método de controle combina a adaptação paramétrica do MRAC, que garante a adaptação do controlador às mudanças do processo e ao mesmo tempo possui as características e compensação de atraso do preditor de Smith. Para isso, é adotado um modelo de referência e o atraso de transporte da planta deve ser conhecido para que possa ser compensado. . . . .	38



## **Lista de Tabelas**

1	Parâmetros de configuração dos estimadores utilizados . . . . .	23
---	---	----



---

## Lista de Abreviaturas e Siglas

CLP	Controlador Lógico Programável. ii, 2, 23, 29, 34
IMC	Controlador a Modelo Interno ( <i>Internal Model Controller</i> ). 2, 5
MQR	Mínimos Quadrados Recursivos. 2, 3, 5, 7, 10, 11, 13, 14, 17, 18, 23, 29, 30
MRAC	Controle Adaptativo Baseado em Modelo de Referência. i, ii, ix, 2–4, 14–16, 24–29, 38
PID	Proporcional Integral Derivativo. 1, 3, 5
ST	<i>Structured Text</i> . 24





## Lista de Símbolos

$C_s(s)$	Compensador primário do Preditor de Smith. 5
$C_{IMC}(s)$	Compensador principal de um controlador a modelo interno. 6
$F(s)$	Função racional escolhida como a dinâmica em malha fechada de um sistema controlado por um controlador a modelo interno. 6, 14
$G(s)$	Função de transferência do sistema a ser controlado. 5, 7
$G_p(s)$	Modelo de primeira ordem com atraso utilizado em simulação. 13
$M$	Número de parâmetros do sistema do filtro adaptativo no algoritmo de estimativa. 7
$N$	Memória efetiva do AMQR. 9
$T_e$	Período de amostragem utilizado em simulação. 14
$\alpha(n)$	Variável auxiliar na demonstração do método de atualização condicional. 11
$\sigma$	Vetor de constantes desconhecidas. 15
$\gamma_r$	Ganho de adaptação da referência do MRAC. 16, 24, 25, 27
$\gamma_x$	Ganho de adaptação do estado do MRAC. 16, 24, 25, 27
$\hat{k}_r$	Ganho adaptativo que multiplica a referência. 15, 16
$\hat{k}_x$	Ganho adaptativo que multiplica o estado do sistema. 15, 16
$\hat{\sigma}$	Valor de estimado de do vetor sigma. 15, 16
$\lambda$	Fator de esquecimento. 7, 9, 17, 18, 21
$\mathbf{R}^{-1}$	Matriz de covariância do AMQR. 10, 17
$\mathbf{R}(n)$	Inverso da matriz de covariância. 9
$\Phi(x)$	Vetor de funções conhecidas. 15
$\mathbf{b}(n)$	Parâmetros estimados do sistema. 7, 9, 12, 22
$\mathbf{f}(n)$	Entrada do estimador de parâmetros. 7, 9, 10
$\mathbf{k}(n)$	Vetor de ganho de Kalman. 8

$\mu$	Constante que regula o quanto grandes erros são amenizados no AMQR. 11, 19–21
$\rho$	Sensibilidade da atualização condicional. 11, 18, 21
$\tau$	Constante de tempo do modelo de primeira ordem utilizado em simulação. 13, 17
$\tau_d$	Constante de tempo desejada em malha fechada. 13
$\theta$	Atraso de transporte ou tempo morto de um sistema. 5, 13
$\tilde{G}(s)$	Função de transferência do modelo do sistema a ser controlado. 5, 7, 14
$\tilde{\theta}$	Atraso de transporte ou tempo morto do modelo de um sistema. 5
$\xi(n)$	Sinal de erro que ameniza valores elevados. 11
$a$	Uma das constantes que caracteriza o modelo em espaço de estados. 15
$a_1$	Parâmetro de um sistema de primeira ordem discreto. 9, 14
$a_2$	Parâmetro de um sistema de primeira ordem discreto. 9, 14
$a_f$	Coefficiente que determina a frequência de corte de um filtro passa-altas. 12, 21
$b$	Uma das constantes que caracteriza o modelo em espaço de estados. 15
$c_{offset}$	Constante de uma função de transferência não linear. 12
$d(n)$	Saída do sistema estimada. 7, 10
$e(n)$	Sinal de erro definido como a diferença entre a saída real da planta e a saída calculada pelo estimador. 8, 11, 21, 22
$f(x)$	Função desconhecida e não linear. 15, 24
$k$	Ganho estático do modelo de primeira ordem utilizado em simulação. 13, 17
$r(t)$	Sinal de referência do controlador. 5, 15, 17, 19
$u(n)$	Sinal de controle amostrado. 7, 10

- $u(t)$  Sinal de controle. 5, 15
- $x(t)$  Variável de estado do sistema. 15
- $y(n)$  Saída amostrada do sistema controlado. 7, 10
- $y(t)$  Saída do sistema controlado. 5, 20



# 1 Introdução

Os algoritmos de controle são hoje parte fundamental na indústria, em especial na regulação de processos como refinarias, siderúrgicas, indústrias cimenteiras e de celulose e papel. Eles são responsáveis por manter as variáveis de interesse dentro dos limites adequados de operação, garantindo assim a qualidade do produto final e a agilidade de produção.

Atualmente, a maior parte dos controladores presentes na indústria são os clássicos Proporcional Integral Derivativo (PID), que executam sua função satisfatoriamente para a maioria das aplicações. Porém, vários motivos fazem com que estes controladores operem fora de sintonia, gerando custos adicionais que poderiam ser evitados. Dentre estes motivos está a identificação inadequada do modelo, possivelmente causada pela utilização um conjunto de dados inadequados, que geram um modelo não condizente. Outro fator comum é a mudança do ponto de operação e influências externas, juntamente com o envelhecimento dos equipamentos, que podem implicar na mudança do comportamento dinâmico do sistema.

Para corrigir esse problema, é possível monitorar o desempenho das malhas de controle (Kempf, 2003) e fazer a manutenção dos controladores mal sintonizados. Porém, esse método é demorado e custoso, além de ser uma medida corretiva e não preventiva. Outra alternativa é implementar controladores adaptativos, cuja sintonia é ajustada continuamente às mudanças do processo.

A teoria de controle adaptativo é um conjunto de técnicas utilizadas para modificar o comportamento ou sintonia de um controlador, que se dá em resposta a mudanças ou incertezas na dinâmica do sistema a ser controlado e suas perturbações (Åström and Wittenmark, 1995). Um simples exemplo prático de aplicação da adaptação de controladores são plantas onde há um fluxo que circula por tubulações e tanques. Como o fluxo é geralmente relacionado com a taxa de produção, assim que esta mudar acontece também uma mudança na dinâmica (ponto de operação) do processo. Essa mudança faz com que uma sintonia que funcionava bem para determinada taxa de produção possivelmente não apresente bom resultado nesta nova condição, havendo necessidade de uma nova sintonia do controlador para manter o desempenho de controle dentro do desejado.

Geralmente estas técnicas envolvem a identificação de parâmetros, incertos ou variantes no tempo, de modelos dos processos. Isso permite o cálculo de uma nova sintonia, mais adequada e que atenda às especificações de controle para o processo. Porém, existem também técnicas que não realizam a identificação de um modelo da planta, mas conseguem adaptar diretamente os ganhos do controlador. Isso caracteriza as duas categorias principais: controle adaptativo direto e controle adaptativo indireto. Na primeira categoria, o controlador é modificado sem que antes haja uma identificação de parâmetros ou modelo do sistema. Já na segunda categoria, as modificações no controlador são traduzidas pela alteração de parâmetros do controlador empregado, calculados através dos parâmetros identificados do sistema.

Ambas as técnicas, de controle adaptativo direto e indireto, serão exploradas em simulação buscando compreender suas vantagens e desvantagens. Os controladores aqui testados são o

Controle Adaptativo Baseado em Modelo de Referência (MRAC) (Lavretsky and Wise, 2013) e o controlador auto-ajustável (Åström and Wittenmark, 1973). Deseja-se então concluir sobre em qual situação e tipo de sistema cada técnica é empregada com maior benefício. Além disso, a análise se estende para a correção dos problemas mais corriqueiramente encontrados na utilização de algoritmos de identificação de parâmetros como o algoritmo dos mínimos quadrados recursivos.

Uma das características de processos que tem sido crítica em controle adaptativo é o atraso de transporte (ou tempo morto). De forma geral pode-se dizer que o conhecimento a priori (ou uma estimativa *online*) desse atraso é de suma importância para um bom desempenho de controladores adaptativos, em especial o auto-ajustável (De Keyser, 2016). O presente trabalho é focado em técnicas que possam ser aplicadas a sistemas com atraso de transporte, já que essa característica é vastamente encontrada na indústria de processos. Supondo que o atraso de transporte é conhecido e invariante no tempo para os casos estudados, foi implementado o método do controlador auto-ajustável Åström and Wittenmark (1995) que tem como controlador um Controlador a Modelo Interno (*Internal Model Controller*) (IMC) em conjunto com um Preditor de Smith (Smith, 1959) adaptado com novas estimativas do processo provenientes de um Algoritmo dos Mínimos Quadrados Recursivos (MQR).

Por ser o controlador auto-ajustável, o mais adaptado para processos com atraso segundo os testes aqui realizados, este controlador é programado em linguagem de CLPs para ser utilizado em processos de regulação de temperatura de uma sala e de secagem de leite para fabricação de soro (Capitaneanu and Knecht, 2017). Parte dos trabalhos apresentados neste documento foram realizados em um estágio em pesquisa na empresa *Schneider Electric*. O objetivo da pesquisa foi de encontrar um método de tornar um controlador a modelo interno (na forma de um Preditor de Smith), previamente desenvolvido para CLPs, capaz de adaptar seus parâmetros para compensar mudanças na dinâmica de processos.

A organização do trabalho se dá da seguinte forma:

- A Seção 3.1 apresenta a teoria da implementação do controlador auto-ajustável com um Preditor de Smith;
- Na Seção 3.2 são apresentadas técnicas para evitar problemas práticos de estimação de parâmetros;
- Uma breve explicação sobre a simulação do controlador auto-ajustável é dada em 3.3;
- Na Seção 3.4 é apresentada a teoria do controlador MRAC;
- A Seção 4.1 apresenta os resultados e análise das simulações com o controlador auto-ajustável;
- A apresentação do bloco do MQR é dada na Seção 4.2;
- A Seção 4.3 apresenta os resultados de simulação do MRAC;
- Na Seção 5 é dada a conclusão.

## 2 Revisão Bibliográfica

Os métodos de ajuste automático de controladores conhecidos como *Autotuners* foram o primeiro passo em direção à automatização da escolha de parâmetros. Um dos primeiros métodos utilizados foi o ajuste de controladores PID baseados no ponto crítico (Ziegler and Nichols, 1942). Esse é um método experimental que consiste em fazer o sistema oscilar utilizando um controlador proporcional unicamente, para a identificação do ganho crítico. Com a informação do período das oscilações e do ganho para o qual o sistema oscila sustentadamente, é possível calcular os parâmetros de um PID com base em uma tabela (Bazanella and Gomes da Silva Jr., 2006).

Porém, o interesse em controladores que podem se sintonizar automaticamente não permaneceu unicamente em algoritmos que dispensam o trabalho manual do projetista. Logo percebeu-se a necessidade de modificar o controlador enquanto o sistema controlado estivesse funcionando, no caso de mudança na dinâmica do processo.

Assim, no início da década de 50 o controle adaptativo foi um tema de grande interesse juntamente com o desenvolvimento de pilotos automáticos para aeronaves de alta performance. Isto se deve ao fato de que os controladores lineares de parâmetros constantes apresentavam bom desempenho em uma condição de operação, mas não durante qualquer regime de voo (Gregory, 1959). No entanto, nessa época o problema de controle adaptativo era muito complexo de lidar com as técnicas conhecidas, implicando que somente na década de 60 houve um avanço significativo nas ferramentas necessárias para lidar com os problemas da análise teórica do controle adaptativo. Nas décadas de 70 e 80 as primeiras provas teóricas de estabilidade surgiram, consolidando a base matemática de algumas técnicas. Mais tarde na década de 80, os primeiros controladores adaptativos apareceram comercialmente. Atualmente os controladores adaptativos estão presentes em bibliotecas de Controladores Lógico Programáveis como vários modelos da Schneider Electric ((Unity Pro, 2011)), *Invensys Ltd* entre outros.

O Controle Adaptativo Baseado em Modelo de Referência (MRAC) foi originalmente proposto por Whitaker et al. (1958). Mais tarde Parks (1966), Monopoli (1974) e Landau (1974) desenvolveram ainda mais o trabalho. Esse controlador consiste em especificar um modelo de referência, o qual representa o comportamento desejado em malha fechada para o sistema, e um algoritmo de adaptação que ajusta ganhos com o objetivo de minimizar a diferença entre a saída real e a de referência. Enquanto alguns autores trabalhavam em estender os conceitos do MRAC para sistemas multivariáveis (Barkana, 1983; Tao, 2014), outros simplificaram o controlador criando outro chamado *Simple Adaptive Control* (Barkana, 2013). Porém, esse método se apresentou muito sensível a dinâmicas não modeladas do sistema controlado, limitando sua aplicação.

O controlador auto-ajustável foi inicialmente proposto por Åström and Wittenmark (1973). Nesta publicação, os parâmetros do processo são estimados *online* pelo Algoritmo dos Mínimos Quadrados Recursivos (MQR) e usados para projetar um controlador de mínima variância. Quando os parâmetros convergem, o desempenho de controle desejado é alcançado. Além disso, foi mostrado que as especificações de controle são alcançadas mesmo quando a estimativa dos parâmetros

apresenta um viés em relação aos parâmetros corretos. Desde então, vários ajustes foram feitos corrigindo problemas frequentes de implementação (Åström, 2008; Åström, 1983). O método original e também modificações do controlador auto-ajustável encontraram aplicações em diferentes campos, que vão desde plantas químicas de grande escala (Fortescue et al., 1981) até sistemas de potência (Bazanella, 1993).

Já para lidar com o problema do atraso de transporte ou tempo morto dos processos, é possível adicionar o mesmo ao modelo de referência do MRAC, com alguns resultados na literatura (Allias et al., 2017), porém alguns autores experimentaram modificações desse método. Existe estudos que lidam com sistemas lineares e de tempo contínuo com atraso nos estados como em Mirkin and Gutman (2004). Há também tentativas de fundir um Preditor de Smith com o MRAC (Fan, 2010). Porém, essas técnicas ainda estão restritas à teoria por apresentarem muitas limitações. A primeira por ser aplicada a um tipo de sistema muito restrito e a segunda por não admitir erro na estimação do atraso de transporte. No caso do controlador auto-ajustável, que é composto de um estimador dos parâmetros do modelo da planta e um controlador qualquer, é possível utilizar técnicas já bem estabelecidas. Neste trabalho utiliza-se um Controlador a Modelo Interno (IMC) e um Preditor de Smith (Smith, 1959), que apesar de ter sido proposto há mais de 50 anos, ainda é um dos métodos mais utilizados para compensar o atraso do sistema controlado.



## 3 Metodologia

### 3.1 Controlador auto-ajustável

Deseja-se controlar um sistema cujos parâmetros são inicialmente conhecidos mas variam no tempo de forma incerta. Além disso, esse sistema pode apresentar atraso de transporte de valor conhecido e invariante no tempo. A estrutura do controlador será fixa, porém os valores de parâmetros serão atualizados conforme são observadas mudanças nos parâmetros do sistema. Dessa forma, o sistema de controle proposto é formado por um controlador IMC com preditor de Smith e o algoritmo dos mínimos quadrados recursivos (MQR) como estimador de parâmetros (Capitaneanu and Knecht, 2017). Uma ilustração é mostrada na Figura 1 onde  $r(t)$  é a referência,  $u(t)$  é o sinal de controle e  $y(t)$  é a saída do sistema.

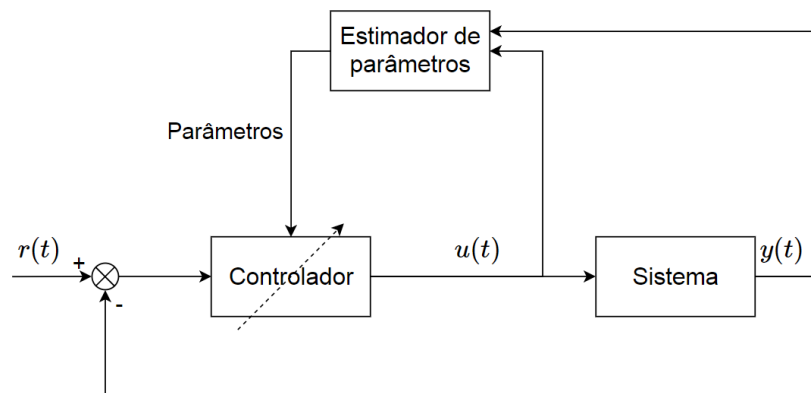


Figura 1: Diagrama de blocos simplificado do sistema de controle auto-ajustável proposto.

#### 3.1.1 Preditor de Smith e controlador a modelo interno

O preditor de Smith (Figura 2) é um controlador que possui um laço interno destinado a eliminar a saída atrasada e alimentar a saída predita para o compensador primário  $C_s(s)$ . No diagrama de blocos,  $G(s)$  denota uma função estável e racional que caracteriza a parte sem atraso da planta,  $\theta$  denota uma constante positiva que representa o atraso de transporte,  $\tilde{\theta}$  e  $\tilde{G}(s)$  são os modelos de  $\theta$  e  $G(s)$  respectivamente.

É comum encontrar na literatura o projeto de  $C_s(s)$  utilizando técnicas de projeto de PID (Yamanaka, 2003). Neste trabalho será usado um controlador a modelo interno (IMC) utilizando um preditor de Smith para sistemas com atraso (Yamanaka, 2003). Ou seja, rearranjando os blocos nos diagramas das Figuras 3 e 2, tem-se o diagrama da Figura 4 que mostra a equivalência desses dois controladores.

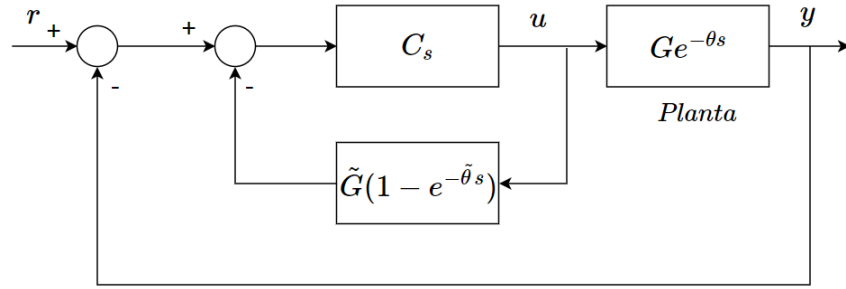


Figura 2: Diagrama de blocos de um sistema controlado por um preditor de Smith.

Para a sintonia do controlador, considera-se o diagrama da Figura 3, onde  $C_{IMC}(s)$  é um controlador com função de transferência mostrada na Equação (1), tendo que  $F(s)$  é uma função no formato da Equação (2), escolhida de forma a representar a dinâmica desejada em malha fechada.

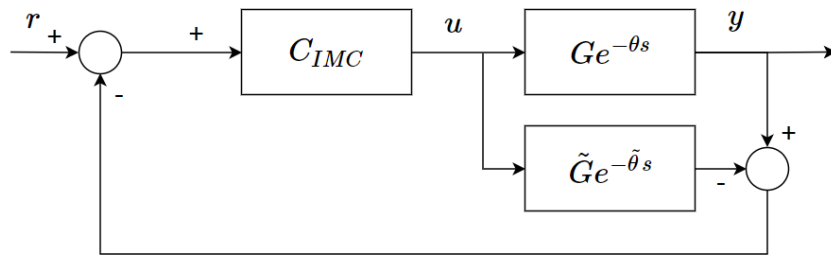


Figura 3: Diagrama de blocos de um sistema controlado por um Controlador a Modelo Interno.

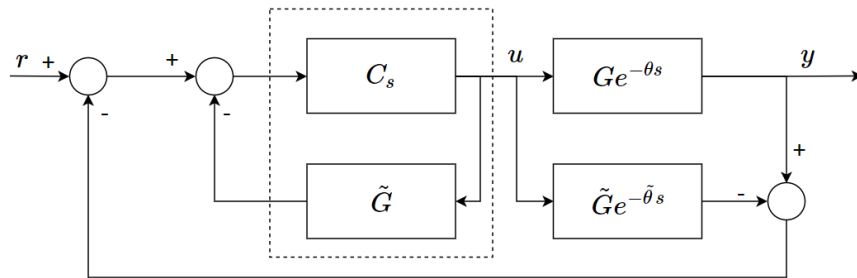


Figura 4: Diagrama de blocos de um sistema controlado por um preditor de Smith a modelo interno.

$$C_{IMC} = \tilde{G}^{-1}(s)F(s) \quad (1)$$

$$F(s) = \frac{1}{(\tau_d s + 1)^n} \quad (2)$$

Assim sendo, a resposta em malha fechada, no caso em que  $\tilde{G}(s) = G(s)$  e  $G^{-1}$  é de fase mínima, para o sistema completo é dada por:

$$Y(s) = F(s)e^{-\theta s}R(s) \quad (3)$$

Enfatiza-se aqui o motivo do emprego desse tipo de controlador, dado que é possível escolher arbitrariamente o comportamento esperado em malha fechada, desde que os limites físicos do sistema sejam respeitados. Dessa forma, qualquer desvio é causado pela mudança de parâmetros da planta, permitindo a avaliação do sistema de adaptação em funcionamento.

### 3.1.2 Algoritmo dos Mínimos Quadrados Recursivo

O Algoritmo de Mínimos Quadrados Recursivo (MQR) (Rowell, 2006) gera o mesmo resultado que o Método dos Mínimos Quadrados ordinário, que encontra a curva que melhor se ajusta aos dados, minimizando uma função custo representada pela soma dos quadrados das diferenças entre o valor estimado e os dados observados. Porém, no caso do MQR, cada novo ponto atualiza (recursivamente) os parâmetros de ajustes sem a necessidade de utilizar todos os pontos coletados.

O MQR pode ser entendido como um filtro discreto de resposta impulsiva finita de tamanho  $M$  com coeficientes  $\mathbf{b}_k$ ,  $k = 0, 1, 2, \dots, M - 1$ , que é capaz, a partir dos coeficientes calculados anteriormente  $\mathbf{b}(n - 1)$  e dos novos dados, calcular recursivamente os coeficientes  $\mathbf{b}(n)$  atuais que minimizam uma função de custo. Uma entrada  $\mathbf{f}(n)$ , que neste caso representa os dados de entrada ( $u(n)$ ) e de saída ( $y(n)$ ) amostrados do sistema, passa pelo filtro em questão que gera uma sequência de saída  $d(n)$ . Em cada etapa de amostragem, o algoritmo atualiza seus coeficientes usando um sinal de diferença  $e(n) = y(n) - d(n)$ . No caso deste estudo,  $y(n)$  é a saída amostrada do processo,  $\mathbf{b}(n)$  é a estimativa dos coeficientes da função de transferência do processo em tempo discreto e  $d(n)$  a saída estimada usando os parâmetros identificados. A Figura 5 explica de forma simplificada o funcionamento do algoritmo.

O MQR também possui um parâmetro chamado fator de esquecimento  $\lambda$  que impõe mais importância aos dados mais recentes no momento de estimar os parâmetros. Esse fator também pode ser modificado ao longo do tempo sem a necessidade de reiniciar a estimativa.

Um resumo do algoritmo dos Mínimos Quadrados Recursivos é mostrado a seguir, que de forma genérica, executa os cálculos que seguem a cada passo de amostragem. Um Exemplo de aplicação é mostrado na seção 3.1.3.

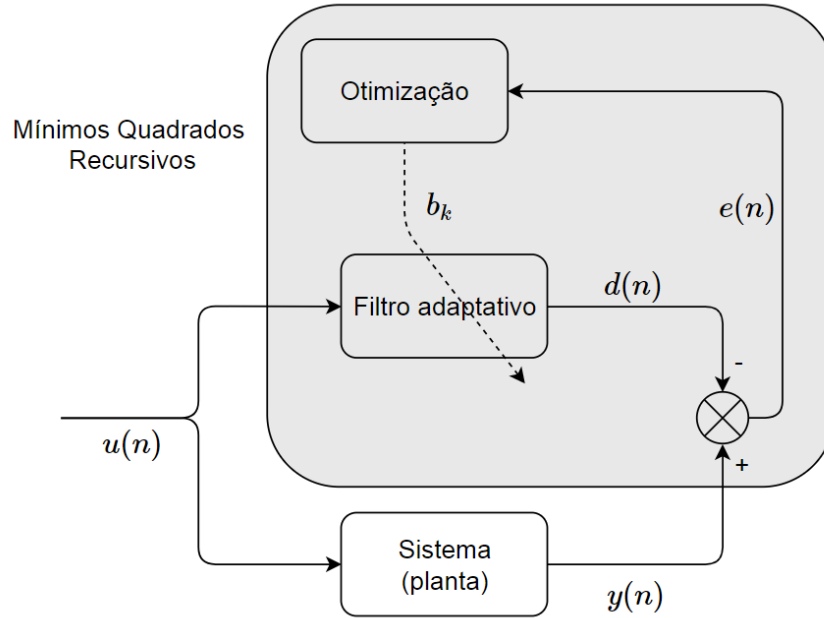


Figura 5: Esquema simplificado do funcionamento do algoritmo de MQR.

No início ou após a reinicialização: iniciar o algoritmo com as condições iniciais.

1. Atualizar o vetor  $\mathbf{f}(n)$ .

2. Atualizar a saída do filtro utilizando os coeficientes anteriores ( $\mathbf{b}(n - 1)$ ).

$$d(n) = \mathbf{f}^T(n)\mathbf{b}(n - 1) \quad (4)$$

3. Calcular o erro  $e(n)$ .

$$e(n) = y(n) - d(n) \quad (5)$$

4. Calcular o vetor ganho de Kalman  $\mathbf{k}(n)$

$$\mathbf{k}(n) = \frac{\mathbf{R}^{-1}(n - 1)\mathbf{f}(n)}{\lambda + \mathbf{f}^T(n)\mathbf{R}^{-1}(n - 1)\mathbf{f}(n)} \quad (6)$$

5. Atualizar  $\mathbf{R}^{-1}(n - 1)$  para a próxima iteração.

$$\mathbf{R}^{-1}(n) = \lambda^{-1}[\mathbf{R}^{-1}(n - 1) - \mathbf{k}(n)\mathbf{f}^T(n)\mathbf{R}^{-1}(n - 1)] \quad (7)$$

6. Atualizar os coeficientes do filtro para a próxima iteração.

$$\mathbf{b}(n) = \mathbf{b}(n - 1) + \mathbf{k}(n)e(n) \quad (8)$$

A matriz  $\mathbf{R}(n)$  representa a matriz de covariância da estimativa. Note que  $\mathbf{R}(n)$  não é utilizada no algoritmo, apenas a sua inversa, evitando a operação de inversão de matrizes utilizada no método dos mínimos quadrados ordinário.

Para a inicialização do algoritmo, se os valores dos coeficientes  $\mathbf{b}(n)$  são conhecidos a priori, eles podem ser utilizados para definir  $\mathbf{b}(0)$ . Se esse não for o caso, normalmente  $\mathbf{b}(0) = \mathbf{0}$ . De forma similar, o conhecimento do histórico de dados de entrada e saída do sistema podem ser utilizados para definir  $\mathbf{f}(n)$  para  $n \leq 0$ . Caso contrário,  $\mathbf{f}(n) = 0$  para  $n \leq 0$ .

Segundo a Equação (9), para iniciar a matriz  $\mathbf{R}^{-1}(0)$  é necessário o conhecimento prévio de  $\mathbf{f}(n) = 0$  para  $n \leq 0$ , e então calcular a inversa de  $\mathbf{R}(0)$ . Esse modo de inicialização, além de não ser prático, insere o risco de que  $\mathbf{R}^{-1}(0)$  seja singular.

$$\mathbf{R}(n) = \sum_{i=0}^n \lambda^{n-i} \mathbf{f}(i) \mathbf{f}^T(i) \quad (9)$$

Por isso, a matriz  $\mathbf{R}^{-1}(0)$  é iniciada como segue, sendo  $\delta$  um escalar positivo.

$$\mathbf{R}^{-1}(0) = \delta \mathbf{I} \quad (10)$$

Com a evolução de  $n$ , o efeito desse erro de inicialização é atenuado por conta do fator de esquecimento  $\lambda$ . Dessa forma, é necessário determinar experimentalmente um valor apropriado para  $\delta$ . No entanto, um valor adequado sugerido por Åström and Wittenmark (1995) é escolher  $\delta > 100\sigma_f^2$  onde  $\sigma_f^2$  é a variância do sinal de entrada  $\mathbf{f}(n)$  (Rowell, 2006).

O fator de esquecimento  $\lambda$  define a memória do sistema e afeta a convergência e a qualidade do seguimento dos parâmetros que variam no tempo. A memória efetiva  $N$  representa o número de iterações antes que o peso de contribuição de um ponto do histórico de dados baixe a  $e^{-1}$  de seu valor inicial que é igual a unidade. O valor de  $N$  é dado por (11).

$$N = \frac{\lambda}{1 - \lambda} \quad (11)$$

Geralmente,  $0,95 < \lambda < 0,995$  para  $19 < N < 199$  respectivamente.

### 3.1.3 Exemplo de estimação de parâmetros de um sistema de primeira ordem

É importante salientar que a estrutura e a ordem do filtro adaptativo do algoritmo dos MQR deve corresponder, ou ser o mais próximo possível, à estrutura e ordem do sistema real, já que as respostas desses são comparadas em cada iteração. Portanto, para estimar os parâmetros  $a_1$  e  $a_2$  de uma função de transferência de primeira ordem de tempo discreto (Equação (12)), é preciso primeiro determinar a equação de recorrência desse sistema.

$$G_d(z) = \frac{a_1}{z - a_2} \quad (12)$$

Utilizando a transformada  $z$  inversa, encontra-se a equação de recorrência (13), onde  $y(n)$  é a saída e  $u(n)$  é a entrada do sistema.

$$y(n) = u(n - 1)a_1 + y(n - 1)a_2 \quad (13)$$

Das equações anteriores conclui-se que os sinais  $u(n - 1)$  e  $y(n - 1)$  formam o vetor de entrada  $\mathbf{f}(n)$  do MQR,  $y(n)$  é o sinal de saída do processo,  $d(n)$  é a estimativa do sinal de saída do processo e os parâmetros  $\mathbf{b}(n)$  são as estimativas dos parâmetros  $a_1$  e  $a_2$ . O mesmo princípio é utilizado para estimar os parâmetros de um sistema de ordem superior. Em outras palavras, o vetor  $\mathbf{f}(n)$  é formado pelas variáveis conhecidas da equação de recorrência que representam o sistema a ser identificado.

## 3.2 Problemas práticos e correções do Controlador Auto-Ajustável

### 3.2.1 Wind-up do estimador e atualização condicional

Com o fator de esquecimento  $\lambda$ , se o sistema a ser estimado não é suficientemente excitado, observa-se um fenômeno chamado *wind-up* do estimador. De forma similar ao *wind-up* da ação integral de um controlador, o *wind-up* do estimador provoca o aumento dos valores da matriz de covariância  $\mathbf{R}^{-1}$ , o que é traduzido como uma incerteza elevada nos parâmetros estimados. Para exemplificar, é suposto que  $\mathbf{f} = 0$ , então as equações do MQR resultarão em:

$$\mathbf{b}(n + 1) = \mathbf{b}(n) \quad (14)$$

$$\mathbf{R}^{-1}(n + 1) = \frac{1}{\lambda} \mathbf{R}^{-1}(n) \quad (15)$$

Nesse caso, a estimativa dos parâmetros continua a mesma, mas os valores da matriz de covariância aumentam rapidamente a cada passo de amostragem, sendo que  $\lambda < 1$ . Portanto, o ganho do estimador  $\mathbf{R}^{-1}\mathbf{f}$  também aumenta. A implicação desse aumento é a variação brusca de parâmetros estimados assim que  $\mathbf{f}$  mude de valor. O mesmo fenômeno é observado se o vetor de regressão é diferente de zero, mas limitado a valores fixos.

Uma maneira de corrigir esse problema é parando a estimação assim que a excitação do sistema é insuficiente. As técnicas utilizadas para identificar a falta de excitação e parar o algoritmo de identificação são chamadas de algoritmos de atualização condicional. Geralmente são utilizados testes simples como uma zona morta em torno da referência da malha de controle, onde o sistema é considerado em regime permanente (não excitado) ou detecção de variações de magnitude de entrada e saída do sistema, que é utilizada para desativar a identificação quando necessário (Åström

and Wittenmark, 1995).

Uma das técnicas consiste em testar o valor de:

$$\mathbf{f}^T(n)\mathbf{R}^{-1}(n-1)\mathbf{f}(n) \quad (16)$$

Pois se o vetor de regressão  $\mathbf{f}(n)$  é constante, por consequência, após uma série de iterações:

$$\mathbf{f}_0^T \mathbf{R}^{-1} \mathbf{f}_0 = \alpha(n) \frac{\mathbf{f}_0^T \mathbf{R}^{-1} \mathbf{f}_0}{\lambda \alpha(n) + \mathbf{f}_0^T \mathbf{R}^{-1} \mathbf{f}_0} \quad (17)$$

Sendo que  $\alpha(n)$  é dado por:

$$\alpha(n) = \frac{1 - \lambda}{1 - \lambda^n} \quad (18)$$

Então, se  $n \rightarrow \infty$  então  $\alpha(n) \rightarrow 1 - \lambda$ . Essa técnica consiste em atualizar os parâmetros estimados somente se  $\mathbf{f}^T(n)\mathbf{R}^{-1}(n)\mathbf{f}(n) > \rho(1 - \lambda)$ . O valor de  $\rho$  é escolhido segundo o sistema e sua escolha é de suma importância para o bom funcionamento do estimador. A recomendação de Åström and Wittenmark (1995) é partir de  $\rho = 2$ , que representa um valor elevado e refiná-lo experimentalmente.

Outro modo de evitar o *wind-up* do estimador é mantendo a excitação do sistema. Para isso é possível enviar sinais de controle com ruído binário pseudo aleatório que excite o espectro de frequências de tal sistema (Bazanella, 1993). Também pode-se usar a diferença entre a saída do processo e sua referência para parar a estimação. Quando essa diferença for maior que determinado valor, o controlador irá atuar excitando o sistema. Dessa forma, nesse momento a estimativa pode ser ativada.

### 3.2.2 Robustez da estimativa

A performance do MQR é máxima quando o ruído presente no sistema e o sinal de erro são ruídos brancos. Como no algoritmo é utilizado o sinal de erro  $e(n)$  ao quadrado, os erros de grande magnitude em especial provocam uma variação ainda maior nos parâmetros estimados. Portanto, com o objetivo de amenizar o efeito de grandes erros, por vezes causados por perturbações, é possível substituir o sinal de erro  $e(n)$  em (8) por outra variável calculada em função de  $e(n)$ , mas que reduza os valores elevados de erro. Neste trabalho,  $e(n)$  é substituído por  $\xi(n)$ , definido em (19). O valor de  $\mu$  define o quanto o sinal de erro  $e(n)$  será distorcido e deve ser sintonizado experimentalmente. Essa modificação não causa nenhum erro entre os valores reais e estimados uma vez que estes tenham convergido.

$$\xi(n) = \frac{e(n)}{1 + \mu|e(n)|} \quad (19)$$

O efeito dessa modificação no algoritmo é a redução das consequências dos grandes erros, evitando que os parâmetros estimados variem de forma significativa por conta de perturbações.

### 3.2.3 *Projeção de parâmetros*

Mesmo utilizando as técnicas apresentadas anteriormente, uma perturbação prolongada pode fazer a estimação de parâmetros divergir dos valores considerados aceitáveis para determinado sistema. Portanto, conhecendo o processo a priori, é possível saber qual é a faixa de valores dentro da qual a estimativa pode estar e limitá-la a esses valores. Para assim fazer, basta testar se os valores estimados estão dentro da faixa aceitável e caso não estiverem, o valor de  $\mathbf{b}(n)$  é substituído por um valor admissível (dentro da faixa) após o último passo do algoritmo, antes de calcular o novo controlador (Åström and Wittenmark, 1995).

No entanto, caso algum valor estimado de  $\mathbf{b}(n)$  esteja ultrapassando os limites, ele ainda contribuirá para a estimativa. Isso faz com que os outros parâmetros estimados sofram influências daqueles que saem da faixa de valor aceitável e tem de ser modificados. Por isso, é importante escolher essa faixa de forma que os parâmetros possam variar livremente durante os períodos de funcionamento normal do sistema.

### 3.2.4 *Filtro passa-altas*

Certos processos são caracterizados por não terem uma resposta linear por, apresentarem um deslocamento em relação à origem. Em outras palavras, possuem um valor estático somado à saída do sistema, como na Equação (20).

$$Y(s) = \frac{k}{s\tau + 1}u(s) + c_{offset} \quad (20)$$

Nesse caso, se a constante  $c_{offset}$  não é conhecida, é necessário que o estimador continue capaz de identificar a função de transferência sem a constante. Para isso, é possível utilizar um filtro passa-altas que, após uma fase de estabilização, pode eliminar o sinal constante do sinal de entrada ou de saída do processo.

O filtro proposto é um filtro discreto, que tem a estrutura da Equação (21).

$$\frac{y_{out}}{y_{in}} = \frac{1 - z^{-1}}{1 - a_f z^{-1}} \quad (21)$$

O único parâmetro a escolher em um filtro desse tipo é o coeficiente  $a_f$ , que define a frequência de corte. Normalmente  $a_f$  é escolhido para que a frequência de passagem seja em torno de 10 vezes menor que a frequência do processo ou determinada experimentalmente.



### 3.3 Simulação do Controlador Auto-ajustável

O sistema controlado  $G_p(s)$  adotado para as simulações, cuja função de transferência é a Equação (22), é um sistema de primeira ordem com atraso de transporte, onde  $k$  é o ganho estático,  $\tau$  é a constante de tempo e  $\theta$  é o atraso. É considerado que o atraso é conhecido e a ordem do sistema não pode variar no tempo. No entanto, o ganho estático e a constante de tempo do sistema são parâmetros variáveis, o que justifica o emprego de um controlador adaptativo. O objetivo será de fazer variar os parâmetros do sistema e testar as técnicas de controle auto-ajustável já apresentadas para manter as especificações de controle.

Um controlador IMC com o preditor de Smith foi escolhido pela sua simplicidade de implementação e capacidade de controlar sistemas com atraso. O Estimador de parâmetros MQR fará a identificação do modelo do processo em tempo real, cujos parâmetros serão empregados no controlador. A Figura 6 mostra o diagrama em simulação, onde o atraso de transporte introduzido entre o sinal de entrada do processo e o estimador - que é o mesmo do processo - permite a sincronização dos sinais de entrada e saída da planta, como se esta não tivesse atraso. Portanto, o MQR identifica somente a parte sem atraso do sistema controlado.

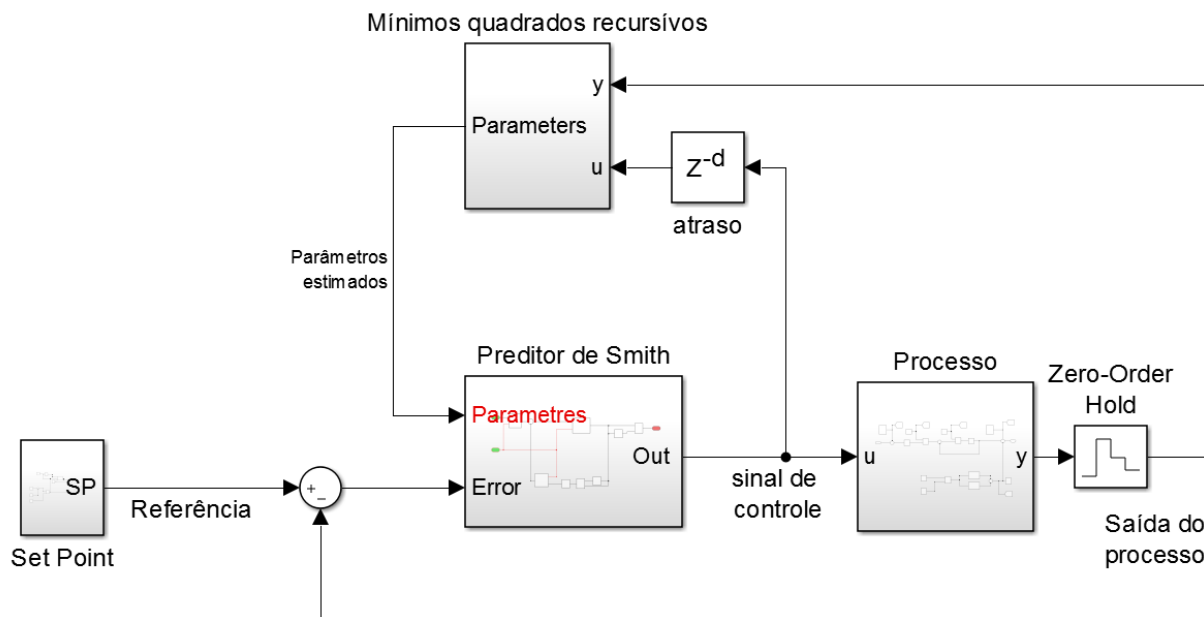


Figura 6: Diagrama de simulação Matlab/Simulink.

O processo inicial é definido tal que  $\tau = 1s$ ,  $k = 3$  e  $\theta = 0,8s$ . O controlador IMC é então projetado, com base no sistema inicial, para o qual a dinâmica escolhida é a dinâmica de um sistema de primeira ordem com constante de tempo de malha fechada desejada  $\tau_d$  que vale  $0,5s$ . Isso implica que o tempo de resposta em malha fechada é a metade do tempo de resposta em

malha aberta. Para o projeto do controlador, a função  $F(s)$  é dada pela Equação (23) e  $\tilde{G}(s)$  será identificada pelo estimador de parâmetros.

$$G_p(s) = \frac{k}{s\tau + 1} e^{-\theta s} \quad (22)$$

$$F(s) = \frac{1}{0,5s + 1} \quad (23)$$

No entanto, o MQR identifica os parâmetros do modelo discreto do sistema, mostrado na Equação (12), onde  $a_1$  e  $a_2$  são os parâmetros que serão estimados. Por isso, é feita a amostragem dos sinais de controle e de saída do processo com período de amostragem  $T_e = 0,1s$ . Por esse motivo, caso um IMC de tempo contínuo seja empregado, faz-se necessária a conversão do modelo discreto para contínuo. As conversões de parâmetros discretos em contínuos para o sistema da Equação (12) se adequar à Equação (22) são dados a seguir (Gohari, 2006):

$$\tau_{est} = -\frac{T_e}{\ln(a_2)} \quad (24)$$

$$k_{est} = \frac{a_1}{1 - a_2} \quad (25)$$

É importante ressaltar que, apesar de as simulações serem conduzidas utilizando um processo genuinamente de primeira ordem, isso não é necessário para obter-se uma boa regulação. Isso se deve ao fato de que muitos processos tem um comportamento que pode ser aproximado por uma função de transferência de primeira ordem, para os quais um controlador projetado seguindo essa hipótese alcança o desempenho esperado em malha fechada (Capitaneanu and Knecht, 2017).

### 3.4 Controle adaptativo baseado em modelo de referência (MRAC)

O MRAC é um método de controle adaptativo direto. Em outros termos, os parâmetros do controlador são estimados diretamente sem cálculos intermediários como a estimação dos parâmetros da planta. Nesse método, os parâmetros (ganhos) são atualizados enquanto o sistema está em funcionamento (*online*), utilizando uma lei de adaptação. A estrutura do MRAC lembra a de um regulador LQR (*Linear Quadratic Regulator*), por realimentação de estados, que emprega um dispositivo capaz de adaptar os ganhos de realimentação. Esta adaptação tem por objetivo minimizar a diferença entre a saída dos estados do modelo de referência desejado e a saída dos estados do sistema real (Lavretsky and Wise, 2013).

No presente estudo, somente o caso de sistemas com um estado foi abordado. Uma breve descrição teórica e os resultados de simulação serão apresentados.

### 3.4.1 Formulação do MRAC

Considerando um sistema cuja dinâmica é descrita pelo espaço de estados da Equação (26), onde  $x(t)$  é o espaço de estados do sistema,  $a$  e  $b$  são as constantes que caracterizam o sistema,  $u(t)$  é a entrada e  $f(x)$  é uma função não conhecida e não linear.

$$\dot{x}(t) = ax(t) + b(u(t) - f(x)) \quad (26)$$

$$f(x) = \sum_{i=1}^N \sigma_i \phi_i(x) = \sigma^T \Phi(x) \quad (27)$$

sendo que  $\sigma$  é um vetor de constantes desconhecidas  $\sigma = (\sigma_1 \dots \sigma_N)^T$ . Já  $\Phi(x)$  é um vetor de funções conhecidas  $\Phi(x) = (\phi_1(x) \dots \phi_N(x))^T$ .

É então necessário escolher um modelo de referência estável e de mesma ordem do processo controlado, mostrado na Equação (28).

$$\dot{x}_m(t) = a_m x_m(t) + b_m r(t) \quad (28)$$

O objetivo para o sinal de controle  $u(t)$  é tal que  $\lim_{t \rightarrow \infty} (x(t) - x_m(t)) = 0$ .

### 3.4.2 Projeto de um MRAC para um sistema de primeira ordem

No caso de um sistema de primeira ordem, a equação da variável de controle é a mostrada na Equação (29), onde  $\hat{k}_x$  é o ganho adaptativo que multiplica o estado  $x(t)$ ,  $\hat{k}_r$  é o ganho adaptativo que multiplica a referência  $r(t)$  e  $\hat{\sigma}$ .

$$u(t) = \hat{k}_x x(t) + \hat{k}_r r(t) + \hat{\sigma}^T \Phi(x) \quad (29)$$

Como os parâmetros a estimar *online* são  $\hat{k}_x$ ,  $\hat{k}_r$  e  $\hat{\sigma}$  e o objetivo é de seguir a resposta do modelo de referência da Equação (28), existem ganhos ideais tais que  $a + bk_x = a_m$  et  $bk_r = b_m$ . O erro de seguimento de referência é dado por  $e(t) = x(t) - x_m(t)$ . As leis de adaptação são dadas pelas equações 30, 31 e 32, onde  $sgn(*)$  é a função sinal. A demonstração teórica da elaboração desta técnica assim como a prova de estabilidade não será posta em exercício aqui, já que ela é facilmente encontrada na literatura (Lavretsky and Wise, 2013).

$$\dot{\hat{k}}_x = -\gamma_x x(t) e(t) sgn(b) \quad (30)$$

$$\dot{\hat{k}}_r = -\gamma_r r(t) e(t) sgn(b) \quad (31)$$

$$\dot{\hat{\sigma}} = -\Gamma_{\sigma} \Phi(\mathbf{x}) \text{esgn}(b) \quad (32)$$

O diagrama de blocos da figura 7 mostra a estrutura desse controlador, mostrando em detalhe a lei de adaptação *online* dos ganhos  $\hat{k}_x$  e  $\hat{k}_r$ , sem considerar a parcela de  $\hat{\sigma}$ . As constantes  $\gamma_r$  e  $\gamma_x$  são valores escolhidos empiricamente e representam os ganhos de adaptação do controlador. Ou seja, quanto maiores os ganhos de adaptação, mais bruscamente os parâmetros adaptados serão modificados.

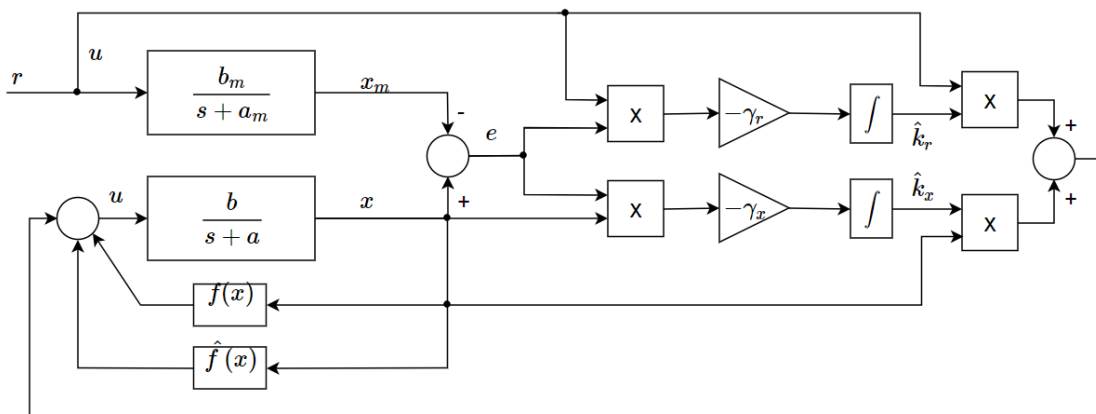


Figura 7: Diagrama completo de um sistema de primeira ordem controlado por um MRAC.

## 4 Resultados

### 4.1 Simulações do Controlador Auto-Ajustável

Os próximos tópicos serão destinados a apresentar os resultados de simulação do controlador Auto-Ajustável.

#### 4.1.1 Identificação de parâmetros e ajuste automático do controlador

O sistema descrito na seção 3.3 e ilustrado pela figura 6 é simulado no *Matlab/Simulink*. Inicialmente, nenhuma das alterações do algoritmo MQR apresentadas anteriormente são utilizadas para fim de demonstração das implicações de problemas práticos. Deste modo o único parâmetro a definir é  $\lambda = 0,99$ . Além disso, nenhum ruído ou perturbações são adicionados neste teste. A referência  $r(t)$  é uma onda quadrada de período de 20 segundos e amplitude de 100. Durante a execução, o parâmetro  $k$  é aumentado em 20% e o parâmetro  $\tau$  é reduzido em -25%. O resultado, mostrado na Figura 8, comprova que o estimador consegue realmente convergir para os parâmetros da planta neste caso ideal.

Já a Figura 9 mostra que o controlador IMC foi realmente adaptado conforme o processo mudava seus parâmetros. Para o primeiro degrau, o controlador sem adaptação apresentou melhor resultado pois o estimador ainda não teve tempo suficiente para encontrar os parâmetros. Porém, a partir do segundo degrau, o controlador adaptativo corrige o modelo interno do IMC e evita que o sistema possua o sobrepasso que é observado no controlador sem adaptação.

#### 4.1.2 Wind-up do estimador e atualização condicional

Como descrito na Seção 3.2.1, quando há falta de excitação do sistema estimado pelo MQR, os valores na matriz de covariância  $\mathbf{R}^{-1}$  aumentam. Esse aumento pode ser interpretado como uma incerteza dos valores de parâmetros estimados, já que pouco ou nenhuma informação nova têm chegado e os pontos mais antigos têm cada vez menos importância devido ao fator de esquecimento  $\lambda$ . Isso não acontece no teste da Figura 8 pois não há ruído e nem perturbações. Dessa forma, mesmo que a matriz de covariância cresça, o processo continua seguindo perfeitamente o modelo o qual o MQR identificou nos passos anteriores onde havia excitação suficiente.

A Figura 10 mostra o que acontece em uma situação mais próxima da realidade, onde ruído branco é inserido tanto no sinal de controle como na saída do processo. Nos momentos onde já não há excitação do processo, a matriz de covariância e a incerteza dos parâmetros cresce e os novos pontos recebidos pelo algoritmo têm grande peso na mudança dos parâmetros. Dessa forma, a única informação disponível no momento começa a influenciar fortemente a estimativa: o ruído. Nota-se então que, cada vez que o sistema entra em regime permanente, acontece uma grande variação dos parâmetros estimados que prejudicam a ação de controle. O instante de tempo e o tipo de variação

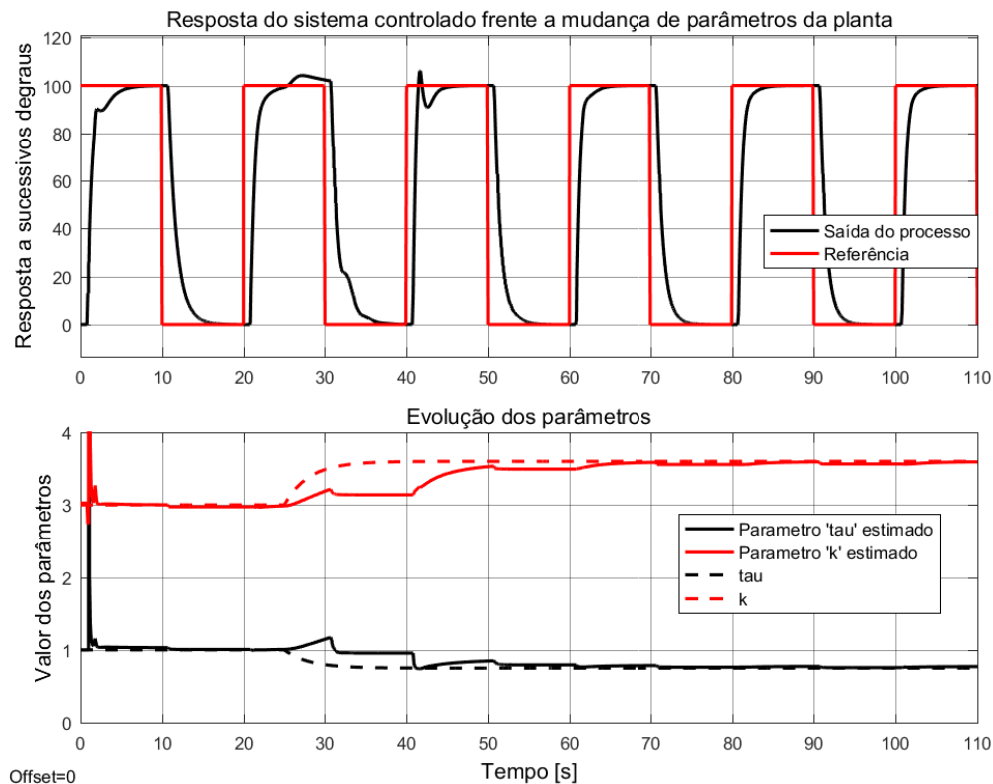


Figura 8: Simulação mostrando a identificação dos parâmetros e a adaptação do controlador para um caso ideal.

que os parâmetros sofrem são aleatórios já que estas são dependentes do ruído, que é aleatório. Para os testes nessa seção, o fator de esquecimento  $\lambda$  vale 0,95, pois assim os pontos mais antigos perdem importância na estimativa mais rapidamente.

Para corrigir o problema, foi implementada a condição de atualização das estimativas descrita na Seção 3.2.1. Com esta modificação, o estimador para de fazer novos cálculos no momento em que os dados recebidos já não sejam oriundos de um sistema propriamente excitado. Essa condição introduz a escolha de um novo parâmetro do MQR que é  $\rho$ . Ele deve ser ajustado experimentalmente, por tentativa e erro, de forma que a estimativa seja ativada somente em períodos onde o processo é excitado. Nesse caso,  $\rho = 1,6$ .

A Figura 11 mostra o resultado da simulação nas mesmas circunstâncias porém utilizando a atualização condicional. A cada vez que o processo se aproxima do regime permanente, o MQR se desliga, evitando o wind-up.

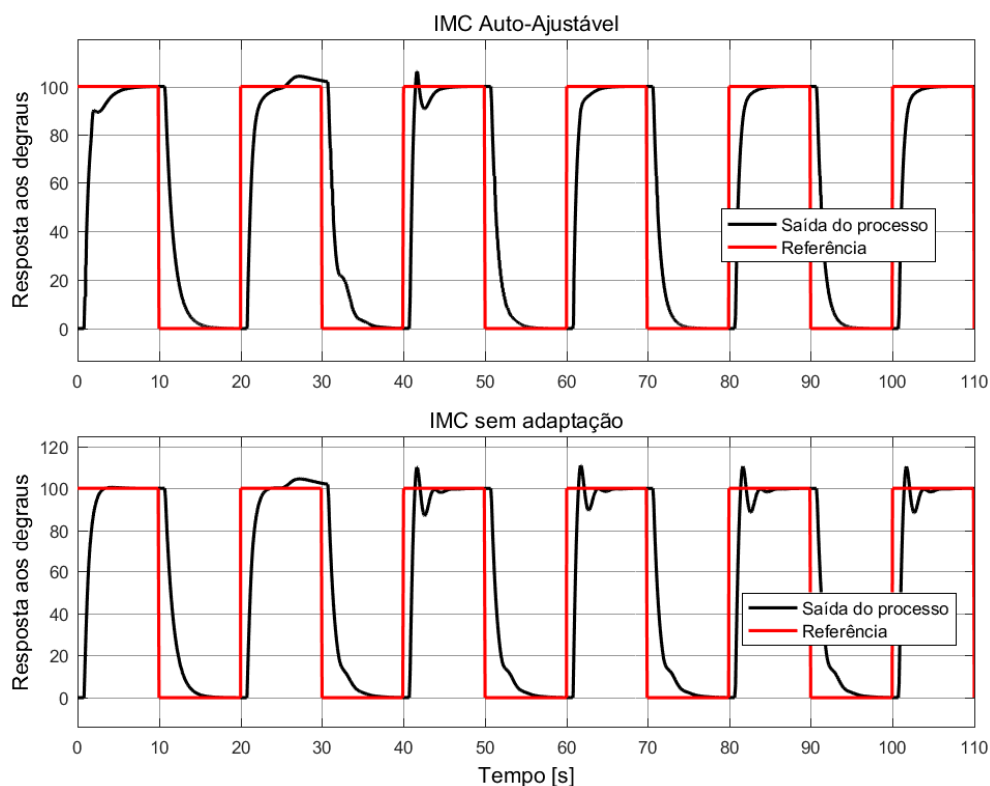


Figura 9: Simulação mostrando a comparação entre o controlador adaptativo e o não adaptativo para um caso ideal.

### 4.1.3 Estimação robusta

Nesta seção, serão mantidas as mesmas condições da seção anterior, porém são introduzidas perturbações na saída do processo. Estas perturbações agirão em períodos onde a referência  $r(t)$  permanece constante e o estimador desligado devido à ação da atualização condicional. O objetivo de tal teste é avaliar as vantagens e desvantagens da constante de robustez (Estimação robusta da seção 3.2.2) apresentada anteriormente, cujo valor adotado para as simulações é  $\mu = 0,08$ .

A Figura 12 mostra a resposta no tempo do sistema, perturbado em  $t = 45$  e  $t = 65$ s, e duas curvas mostrando a estimativa dos parâmetros utilizando a constante de robustez e como seria caso não a utilizasse. Nota-se que os parâmetros estimados afastam-se menos dos valores corretos quando a técnica de estimação robusta é empregada. Após as duas perturbações, houve uma diferença de 34,6% no tempo de estabilização do sistema sem constante de robustez em relação ao mesmo degrau para o sistema com constante de robustez, descontando o atraso de transporte.

No entanto, a constante de robustez influencia na velocidade de seguimento de mudanças reais dos parâmetros do processo. Como todo e qualquer erro entre a saída real e a saída estimada é reduzido por essa constante, é natural que a identificação de mudanças no processo seja mais

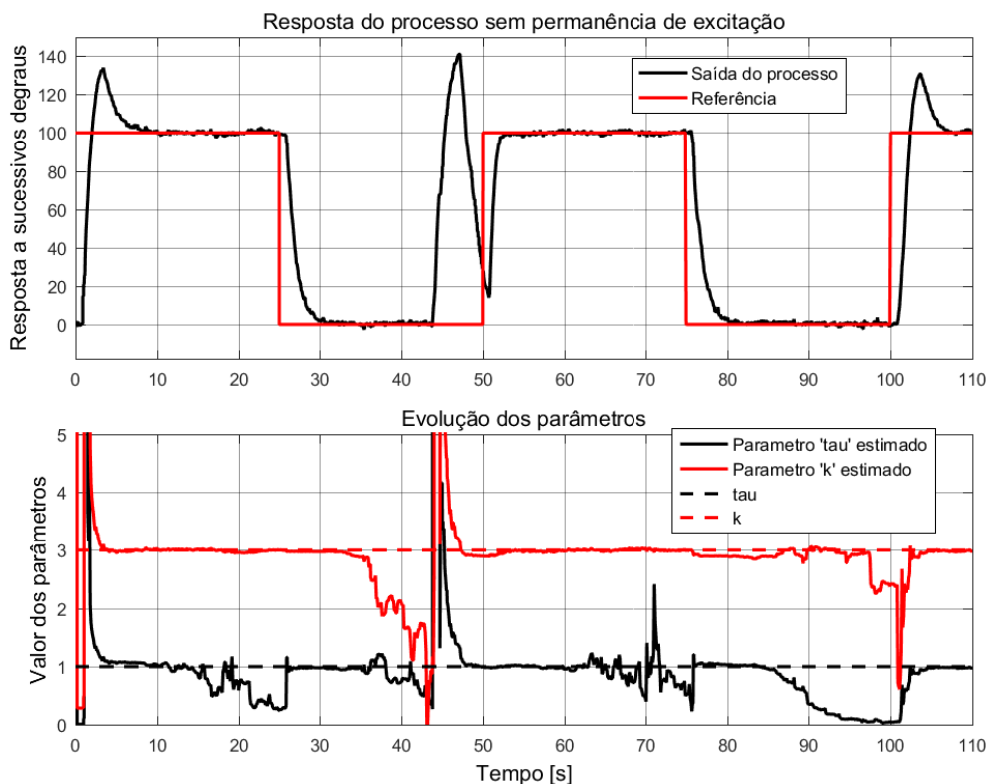


Figura 10: Simulação mostrando o wind-up do estimador em um caso sem excitação permanente.

lenta. A Figura 13 mostra esse efeito: apesar dos parâmetros estimados convergirem para os valores corretos em ambos os casos, o estimador sem a constante de robustez o faz mais rapidamente. A influência de  $\mu$  no controlador é observada principalmente no degrau de  $t = 40s$ , onde o valor máximo alcançado por  $y(t)$  acontece no mesmo instante para ambos os sistemas. Porém, o sistema com constante de robustez ultrapassa a referência em 6,4% enquanto sistema sem constante de robustez ultrapassa em 5,2%.

Como esperado, o emprego dessa modificação no algoritmo de estimação prejudica o seguimento de mudança de parâmetros da planta. Porém, esse efeito não é tão pronunciado quanto a influência causada pelas perturbações quando  $\mu = 0$ , justificando o emprego de tal técnica.

A sintonia de  $\mu$  é condicionada ao conhecimento do tipo de variações paramétricas sofridas pelo processo e do tipo de perturbações presentes no mesmo. Caso não haja grandes perturbações, é aconselhado reduzir o valor de  $\mu$ , pois caso contrário ela poderá atrapalhar no seguimento de mudança de parâmetros, principalmente se estes ocorrerem de forma repentina. No entanto, se as mudanças tem uma dinâmica muito lenta, pode-se escolher um valor maior para  $\mu$ , já que os parâmetros convergirão para o valor correto ao longo do tempo.



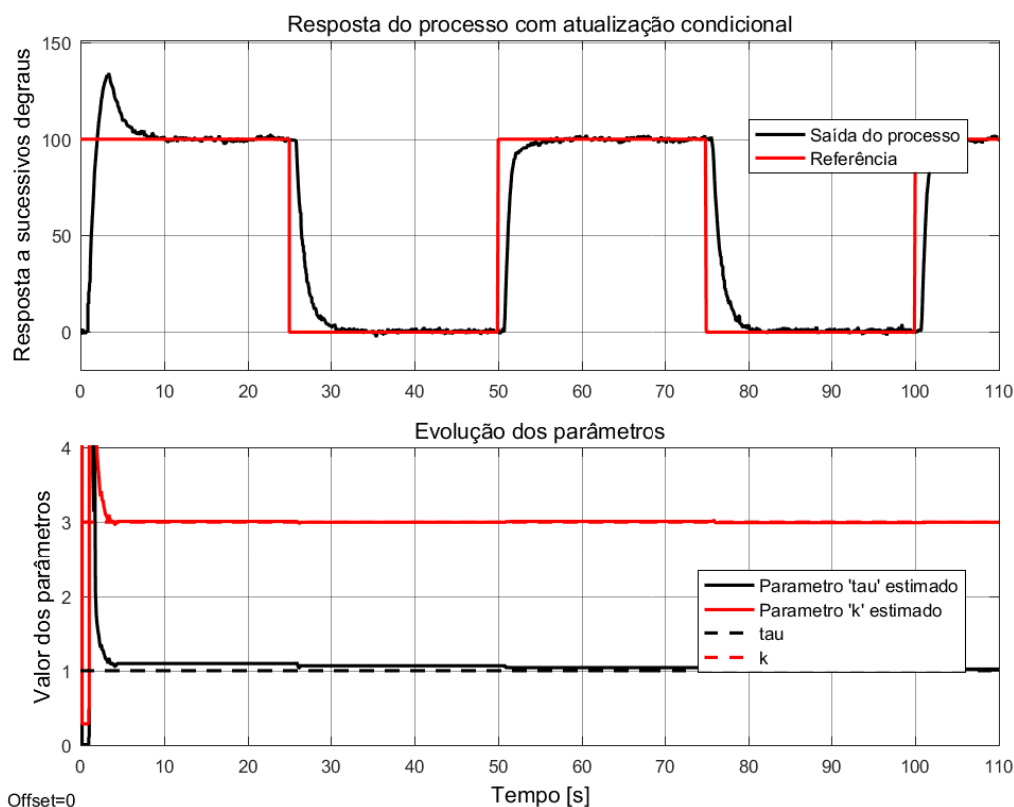


Figura 11: Simulação do sistema em um caso sem excitação permanente e atualização condicional.

#### 4.1.4 Utilização de múltiplos estimadores

O estimador de parâmetros do controlador auto-ajustável possui vários parâmetros a serem escolhidos na hora da implementação:  $\lambda$ ,  $\rho$ ,  $\mu$  e por vezes  $a_f$  e a ordem do modelo estimado. Nem sempre essas escolhas são evidentes, já que alguns desses parâmetros devem ser escolhidos experimentalmente. Por isso, é possível utilizar múltiplos estimadores em paralelo que são configurados de maneiras diferentes de forma a cobrir alguns dos cenários possíveis para o processo em questão.

Dessa forma, a cada passo de amostragem, deve-se escolher qual conjunto de parâmetros será empregado no controlador, ou seja, qual estimador possui o resultado mais próximo do real. Isso é feito comparando o sinal de erro  $e(n)$ , dado pela Equação (5): o estimador que apresentar o menor erro entre seu modelo interno e a saída real é provavelmente aquele que possui os parâmetros mais adequados.

A Figura 14 mostra a simulação de um sistema com o mesmo controlador IMC, mas que utiliza 3 estimadores configurados como mostrado na Tabela 1. Nota-se pelo terceiro gráfico da figura que o estimador mais adequado a esse sistema é o número 1. Porém, há momentos em que os outros estimadores são utilizados, justificando o uso destes. Como mencionado anteriormente, podem ser

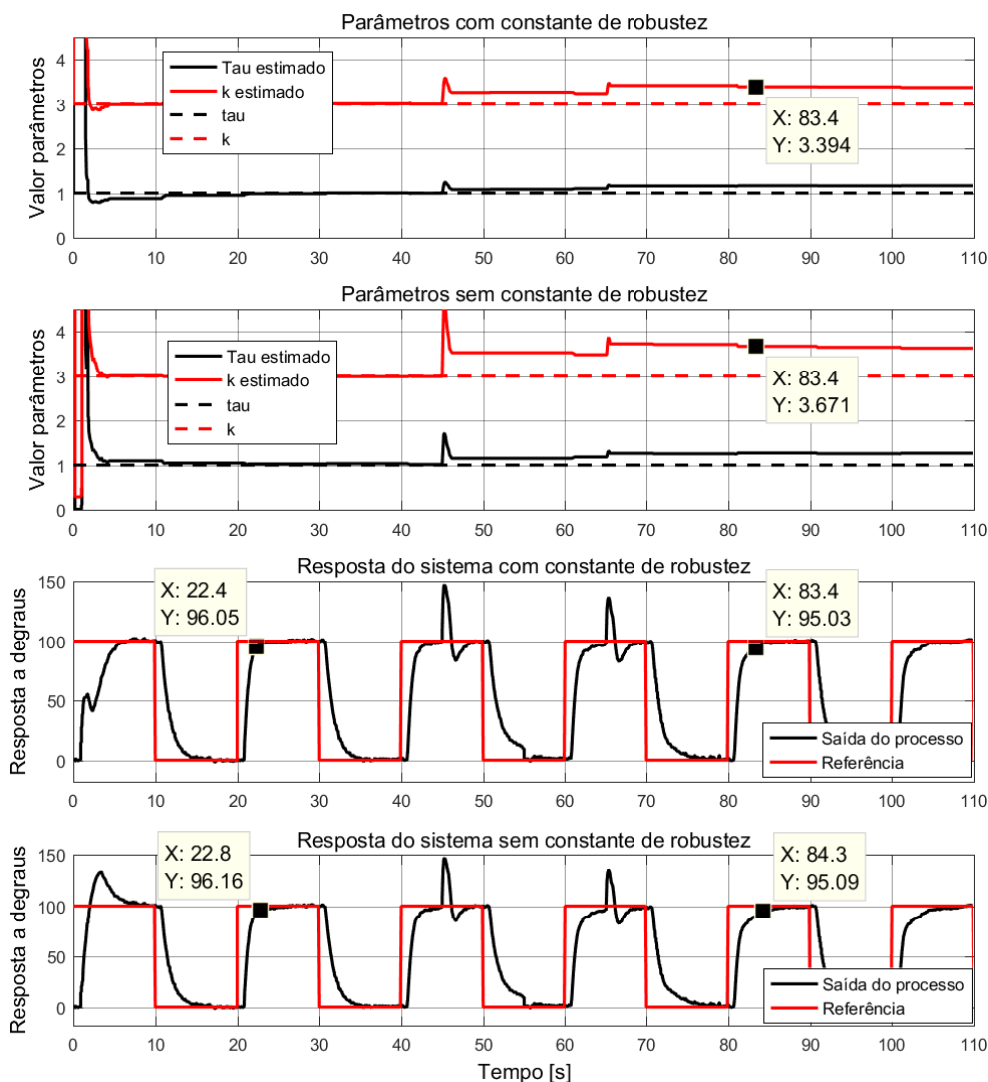


Figura 12: Simulação do sistema mostrando o efeito da constante de robustez com perturbações.

utilizados ainda mais estimadores com configurações e ordem variadas, tendo como limite somente a capacidade computacional do dispositivo onde esse algoritmo está sendo executado.

Dentre outras estratégias que podem ser adotadas, quando utilizando múltiplos estimadores, estão o reset periódico e o armazenamento dos dados de saída do estimador. Fazendo um reset periódico e não sincronizado de cada estimador permite que estes identifiquem mais rapidamente os parâmetros atuais do sistema, sem o peso das medidas passadas. Já o registro dos sinais de saída do estimadores, sinal de erro  $e(n)$  e saída  $\mathbf{b}(n)$ , são úteis para análise posterior, sendo possível tirar informações sobre os parâmetros do sistema: como eles variam e quando a estimativa é perturbada.

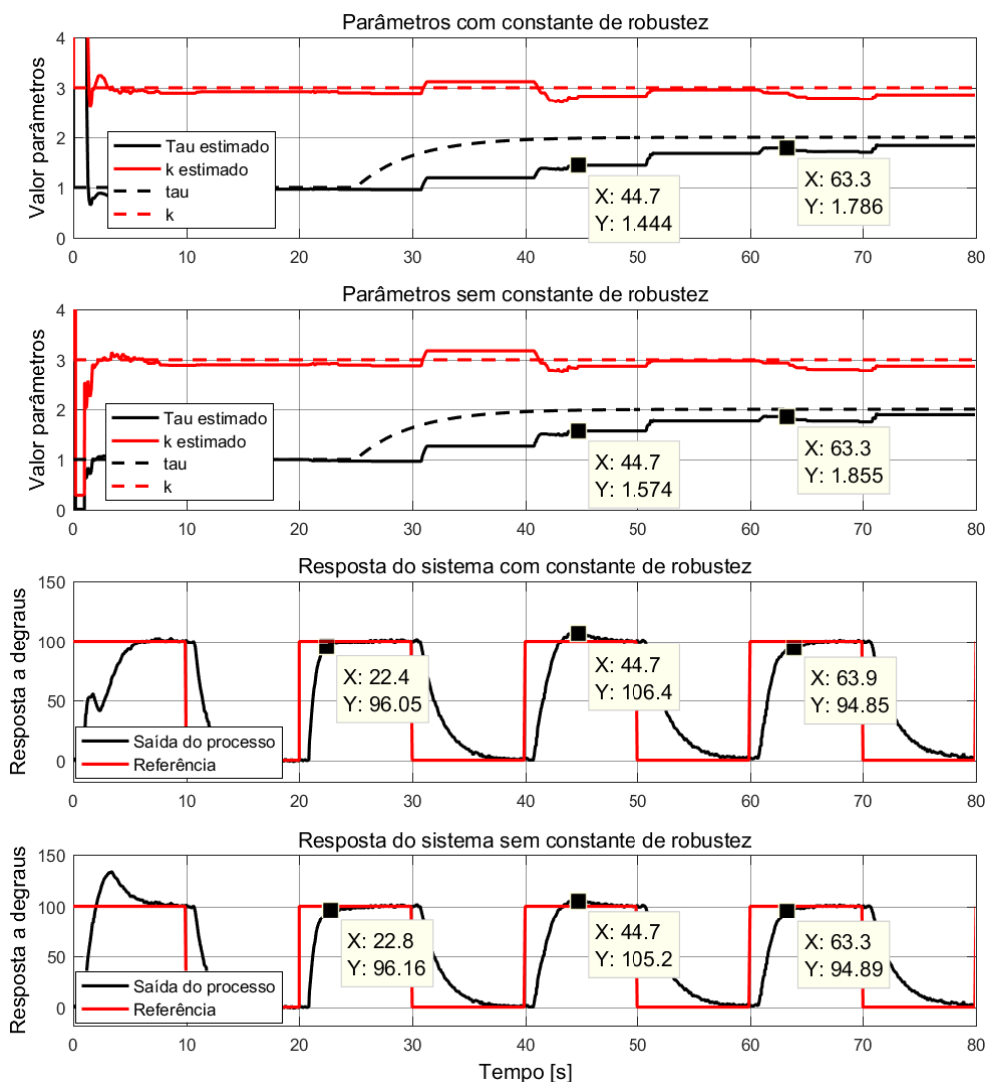


Figura 13: Simulação do sistema mostrando o efeito da constante de robustez variação de parâmetros.

Estimador	$\lambda$	$\rho$	$\mu$
1	0,99	1,5	0
2	0,98	1,5	0
3	0,95	1,5	0,08

Tabela 1: Parâmetros de configuração dos estimadores utilizados

## 4.2 Implementação do MQR para CLP

O MQR apresentado no presente trabalho, juntamente com todas as modificações para evitar os problemas práticos, foi programado em CLP. Foi criado um bloco no software *Unity Pro*, na

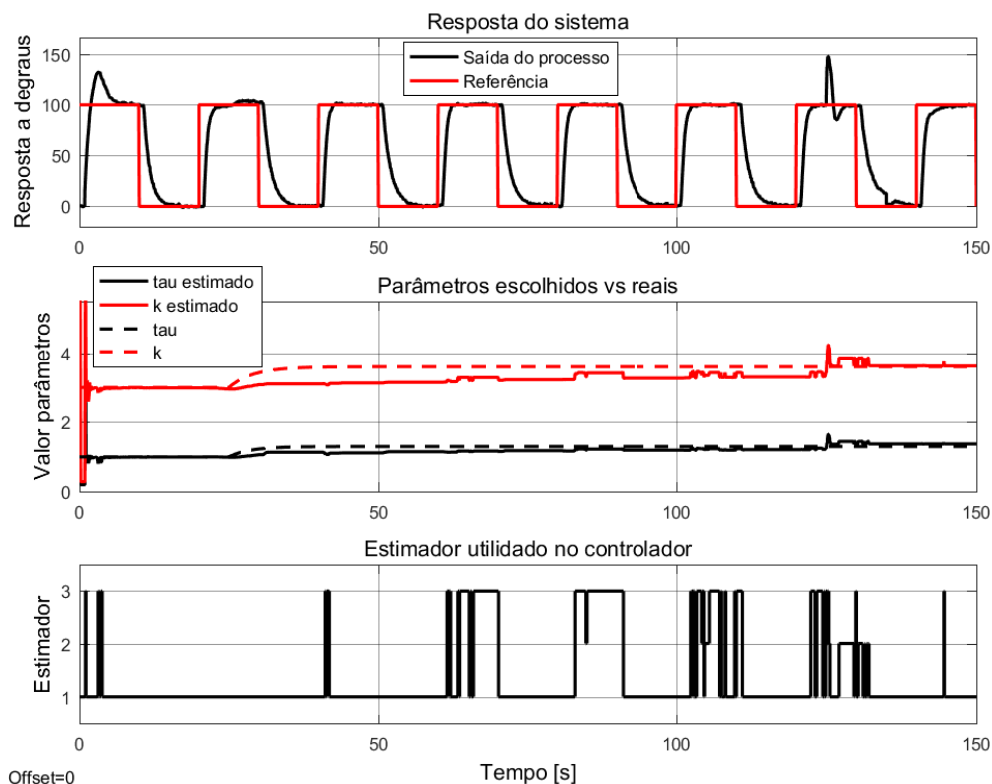


Figura 14: Simulação do sistema utilizando três estimadores diferentes.

linguagem padronizada *Structured Text* (ST), que permite sua utilização em um grande número de CLPs da *Schneider Electric*. O código em ST é mostrado no anexo A.

Esse bloco foi testado, juntamente com um controlador a modelo interno, para controlar a temperatura de uma sala (Capitaneanu and Knecht, 2017). A figura 15 mostra a identificação do ganho estático  $k$  desse sistema, que estabiliza em um valor após alguns passos de amostragem. Conclui-se que os valores estimados são coerentes com os encontrados a partir de outras formas de modelagem. Devido a um contrato de confidencialidade, alguns dados são omitidos.

### 4.3 Simulações do MRAC

O controlador MRAC foi testado para um sistema simples de primeira ordem, sem o termo  $f(x)$ . O processo escolhido tem como modelo a Equação (26), para a qual inicialmente  $b = 3$  e  $a = -1$ . A partir de  $t = 10s$ , o parâmetro  $a = -3$ , e a partir de  $t = 20s$ ,  $b = 2$ . Os ganhos de adaptação  $\gamma_r$  e  $\gamma_x$  são escolhidos arbitrariamente para testar sua influência. É também de maneira experimental que esses valores são sintonizados. O modelo de referência escolhido é dado pela Equação (28), sendo que nestes testes  $a_m = -4$  e  $b_m = -4$ .

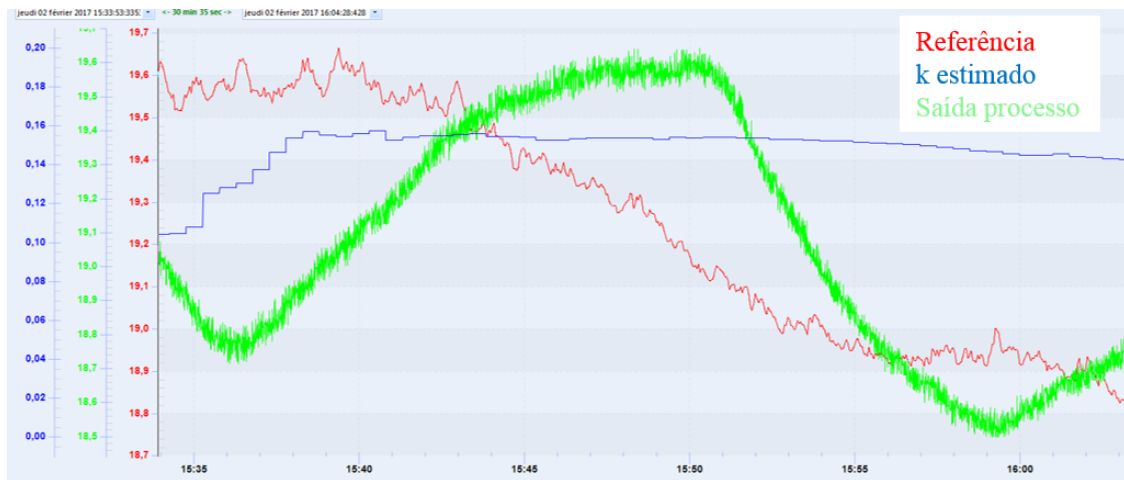


Figura 15: Estimação do ganho estático  $k$  para um regulador de temperatura. Imagem cedida por *Schneider Electric*.

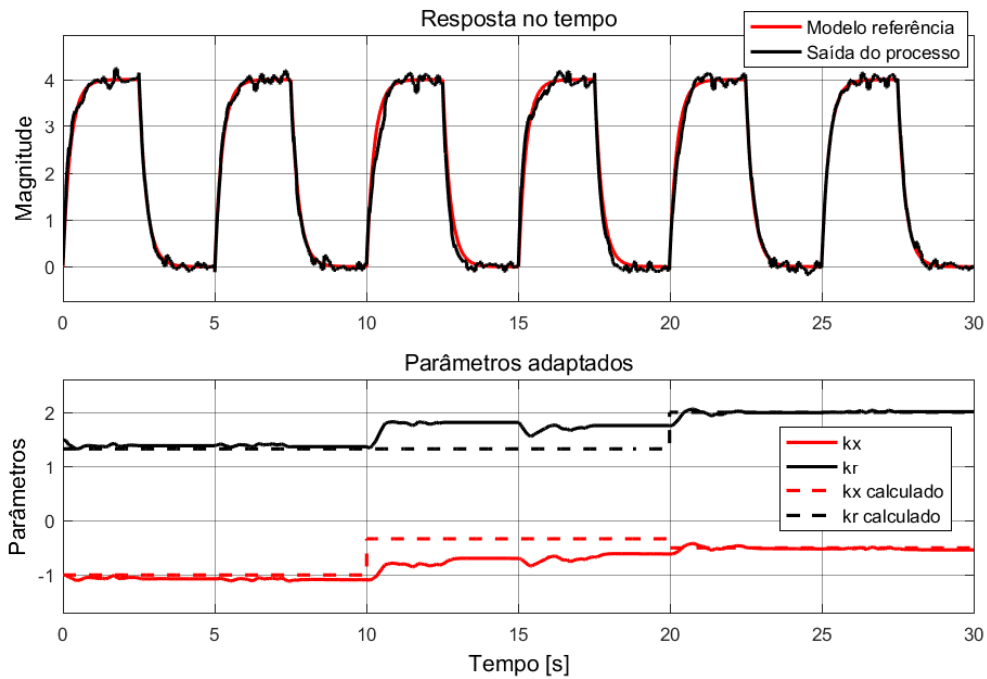
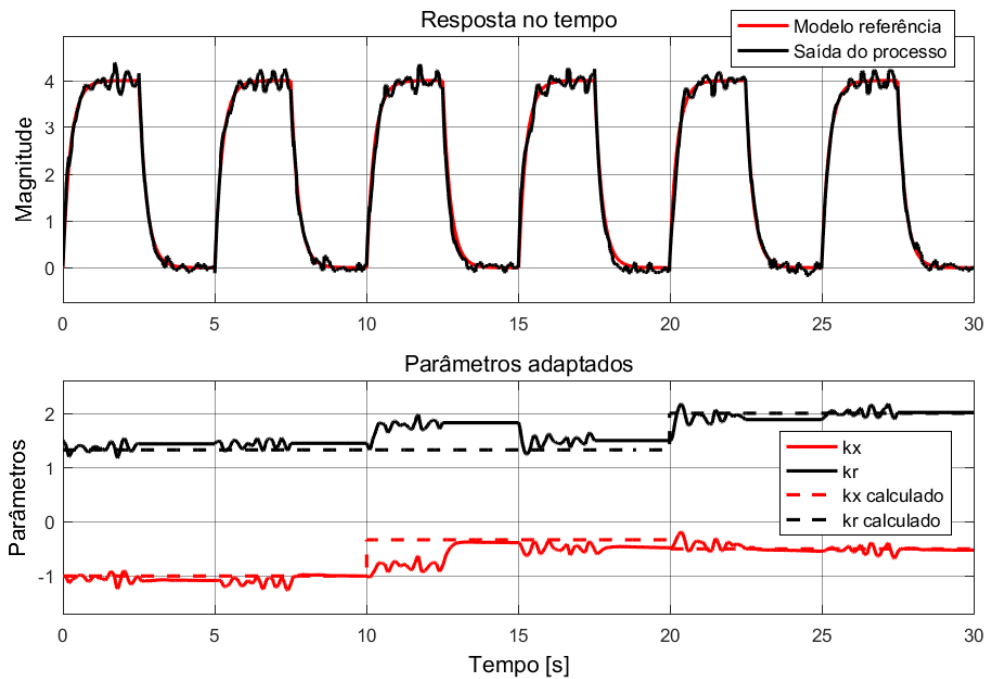
As Figuras 16 e 17 mostram, em um caso ideal com ruído branco nos sinais de controle e de saída, a resposta no tempo e a adaptação de parâmetros em comparação com os ganhos ideais calculados. Utiliza-se  $\gamma_r=0,5$  e  $\gamma_x=0,5$  para a primeira e  $\gamma_r=2$  e  $\gamma_x=2$  para a segunda figura. Notavelmente, há uma diferença no tempo necessário para adaptar os ganhos para cada um dos casos. Quanto maior o ganho  $\gamma$ , mais rápida é a adaptação e também mais oscilatória.

Para reforçar as comparações entre os sistemas, um controlador PI (Proporcional Integral) foi projetado para que o sistema em malha fechada se comportasse como o modelo de referência, para  $b = 3$  e  $a = -1$ . A Equação (33) mostra tal controlador. Em seguida, utiliza-se as mesmas condições das simulações anteriores para avaliar o desempenho do PI frente ao desempenho do MRAC para as mesmas variações de parâmetros.

O resultado é mostrado na Figura 18, nota-se que o PI desempenha seu papel corretamente até acontecerem variação de parâmetros do sistema. A partir desse momentos, o PI já não consegue manter a resposta dinâmica desejada para o sistema em malha fechada. Essa figura mostra que o mecanismo de adaptação do MRAC é o que realmente mantém o desempenho do sistema em malha fechada, sendo que um controlador padrão como o PI não é adequado para um sistema com tal variação de parâmetros.

$$C(s) = \frac{4s + 1}{3s} \quad (33)$$

Para testar o desempenho do MRAC para sistemas com tempo morto, são feitas duas simulações com dois atrasos de transporte diferentes, que são acrescentados ao processo e modelo de referência das simulações anteriores (Allias et al., 2017). Com a adição do atraso, os ganhos  $\gamma$  foram sintonizados novamente por tentativa e erro para cada caso. Para o processo com atraso de 0,1 segundos, os ganhos são  $\gamma_r=0,1$  e  $\gamma_x=0,1$ . Já para o processo com atraso de 0,25 segundos, os

Figura 16: Simulação do controlador MRAC com parâmetros  $\gamma = 0,5$ .Figura 17: Simulação do controlador MRAC com parâmetros  $\gamma = 2$ .

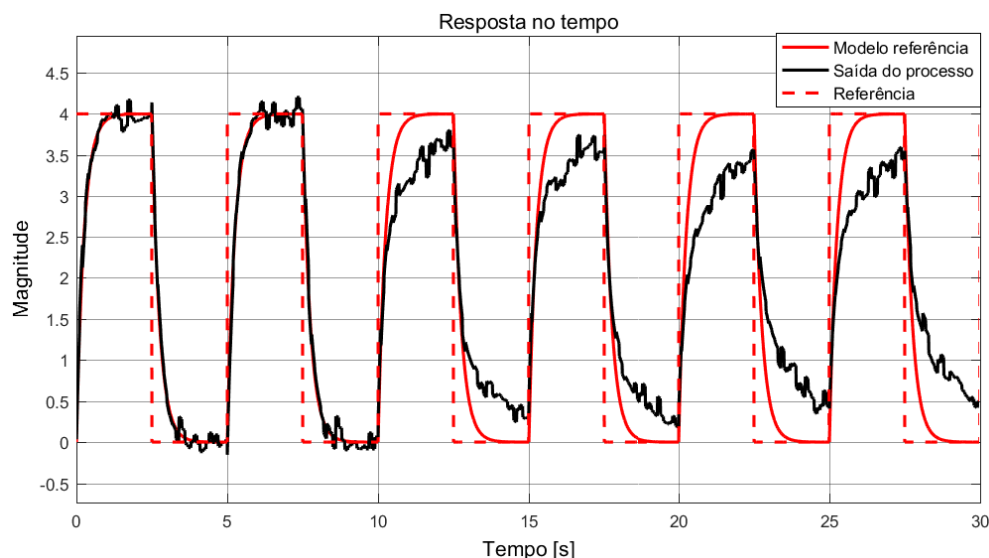


Figura 18: Simulação de um controlador PI com variação de parâmetros do processo.

ganhos são  $\gamma_r=0,05$  e  $\gamma_x=0,05$ .

A Figura 19 mostra a resposta no tempo e a adaptação paramétrica dos dois sistemas mencionados. Observou-se que quanto maior for o atraso de transporte em relação à constante de tempo do processo, menor devem ser os ganhos de adaptação  $\gamma$  com o intuito de evitar oscilações na resposta. Porém, com ganhos pequenos, a adaptação se torna lenta, prejudicando a capacidade de compensar a mudança do processo. Além disso, se o processo possuir um tempo morto muito grande, mesmo que sejam encontrados ganhos para os quais o sistema em malha fechada é estável, o desempenho do MRAC será prejudicado.

Apesar dos resultados, essa configuração de fácil implementação do MRAC requer o conhecimento da ordem do sistema para funcionar corretamente. Essa restrição normalmente é um impedimento na utilização desse tipo de controlador quando se trata de processos, já que estes costumam ter ordem muito grande ou desconhecida e comportamento não linear. Além disso, esta estrutura não é compatível com sistemas com atraso de transporte. Estudos sobre a fusão entre o Preditor de Smith e o MRAC (Fan, 2010) foram desenvolvidos, gerando sistemas de controle complexos como o mostrado na Figura 20 na seção B em anexo. Porém os testes realizados no âmbito desse trabalho mostraram que esse controlador é muito sensível ao valor do atraso de transporte, que deve ser conhecido. Uma leve diferença entre o atraso modelado e o real leva o sistema à instabilização.

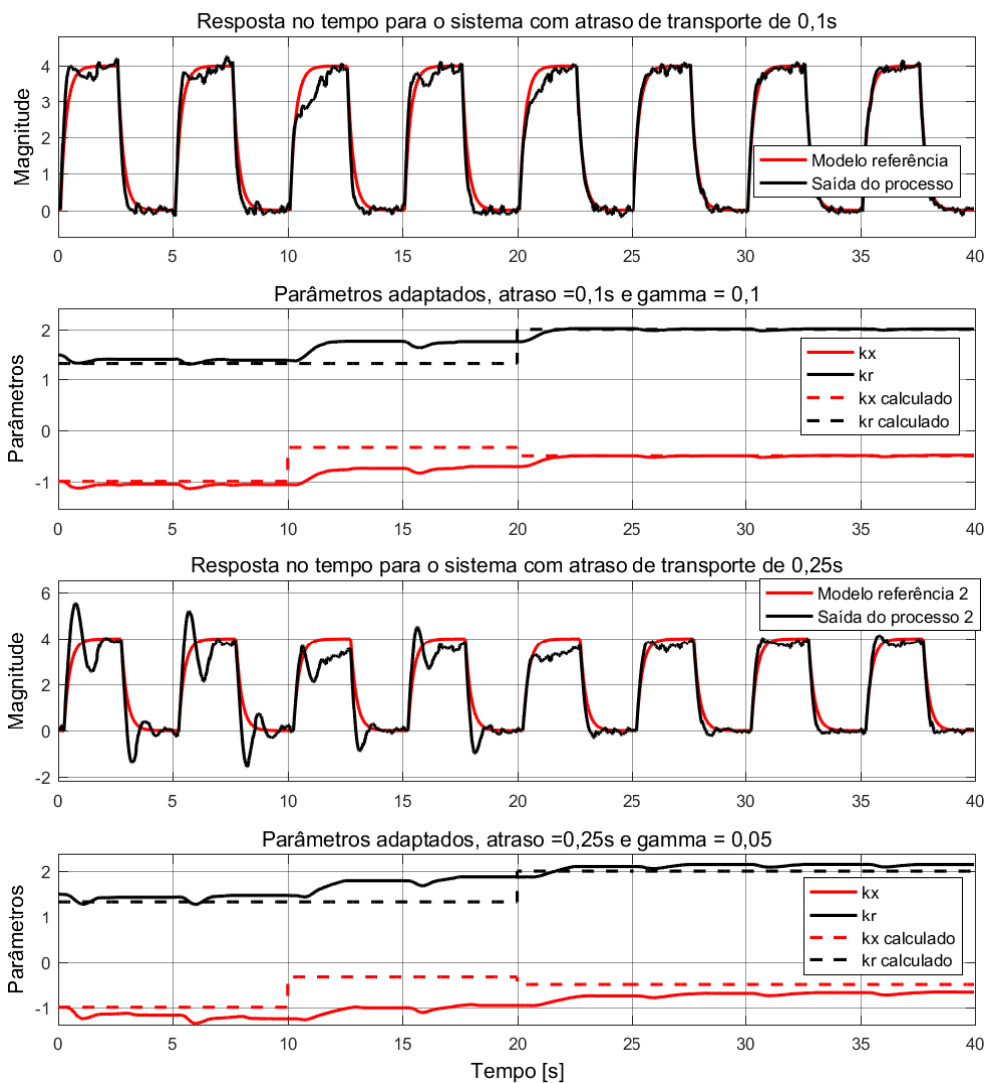


Figura 19: Simulação do controlador MRAC sem adaptação dos ganhos  $k_r$  e  $k_x$ .



## 5 Conclusão

As simulações do controlador auto-ajustável realizadas neste trabalho mostram que mesmo para controlar um sistema simples, é crucial conhecer vários aspectos do processo. Dentre as informações importantes estão a ordem, a velocidade de variação de parâmetros do processo e quais parâmetros variam efetivamente. Além disso, o nível de ruído, frequência e intensidade das perturbações e a faixa de valores válidos para os parâmetros estimados influenciam na sintonia do estimador. Dessa forma, um controlador adaptativo com estimação de parâmetros possui seus próprios parâmetros a serem sintonizados, sendo parte deles escolhidos experimentalmente, através de simulações ou de forma empírica.

O método auto-ajustável tem a vantagem de ser facilmente modificável para exigências de um processo em específico, como alterações no algoritmo de identificação e escolha do tipo do controlador empregado. O desacoplamento entre estimação e controle também pode ser explorado. Como demonstrado nesse trabalho, é possível utilizar vários estimadores em paralelo e escolher dentre eles os melhores parâmetros. Além disso, é possível alterar outros parâmetros manualmente no controlador caso isso seja de interesse.

Sem que haja uma forma de identificar a falta de excitação do processo e parar o estimador, os parâmetros estimados divergem, prejudicando fortemente a ação do controlador como mostrado na Figura 10. Já se o processo é afetado por perturbações de amplitude ou duração significativas, estas podem prejudicar a estimativa já que os sinais recebidos pelo estimador não representam a dinâmica do processo. Dessa forma, sem uma constante de robustez bem sintonizada, mostrada em simulação na Figura 12, o controlador receberá parâmetros não condizentes que prejudicam seu desempenho. Ainda, caso por algum motivo os parâmetros estimados assumam valores fora da faixa aceitável, estes serão transferidos para o controlador que poderá gerar um sinal de controle que tira o processo da sua região de funcionamento normal, podendo causar danos materiais e humanos. Para evitar tal problema, o bloco para CLP mostrado na Seção 4.2 conta com a limitação dos parâmetros apresentada na Seção 3.2.3.

Durante este trabalho, foi mostrado que o método de controle auto-ajustável, em especial a parte do estimador MQR, apresenta vários problemas quando confrontado com a realidade dos processos industriais. Porém, foi mostrado que as modificações do algoritmo dos mínimos quadrados recursivo propostas na literatura o tornaram adequado para utilização na indústria, como exemplificado pelo funcionamento adequado do bloco do algoritmo MQR para CLPs, mostrado na Seção 4.2.

Já o MRAC apresenta bons resultados tanto no quesito adaptação a mudanças do processo quanto na manutenção das especificações de controle. Porém, tem o desempenho fortemente dependente do conhecimento da ordem do sistema controlado, além de não ter apresentado bons resultados para sistemas com qualquer atraso de transporte. Os pontos fortes desse método são a simplicidade de implementação e a facilidade de sintonia. Como foi mostrado, para o sistema de primeira ordem utilizado nas simulações, esse controlador requer a sintonia de apenas dois ganhos e os problemas encontrados no controlador auto-ajustável como o *wind-up* do estimador não se

aplicam.

Assim, por serem capazes de controlar sistemas com parâmetros incertos ou inicialmente desconhecidos, os controladores adaptativos podem ter seu nicho mal interpretado. Os resultados desse trabalho indicam que um controlador capaz de se adaptar aos parâmetros do sistema controlado, apesar de não precisar do conhecimento exato dos parâmetros da planta, não pode ser visto como a solução para toda e qualquer falta de informação sobre o processo. Pelo contrário, para que a adaptação às mudanças do sistema seja alcançada e as especificações de controle mantidas, pode ser necessário mais conhecimento de algumas características processo do que para controlar sistemas de parâmetros constantes. Com essas informações, conclui-se que os cuidados necessários na implementação de tais algoritmos são justificados quando existe a necessidade e a possibilidade de implementação de um controlador adaptativo.

## **5.1 Trabalhos Futuros**

Como o algoritmo de identificação é parte essencial do controlador auto-ajustável e somente o MQR foi testado, este trabalho seria complementado com testes de outros estimadores de parâmetros. Dentre as possibilidades estão os algoritmos: Mínimos Quadrados Estendidos, *Recursive Maximum-Likelihood*, Erro de Saída com Modelo de Predição Estendido (Landau, 2005) ou ainda um MQR com fator de esquecimento variável (Fortescue et al., 1981).. Cada um deles possui particularidades: vantagens e desvantagens que podem ser exploradas dependendo do tipo do processo controlado. Além disso, pode-se desenvolver estimadores para processos de ordem mais elevadas. Quanto à escolha do controlador, pode-se testar o controlador discreto de mínima variância inicialmente proposto por Åström and Wittenmark (1973).

É possível também estimar juntamente com os parâmetros do processo, parâmetros de eventuais perturbações caso estas sejam recorrentes. Para isso, basta modificar a forma do modelo do estimador para integrar o sistema da perturbação. Dessa forma, pode-se evitar que o processo e a estimação sejam perturbados, já que pode-se utilizar a estimativa para compensar seu efeito.

Ainda como trabalhos futuros sugere-se o teste do controlador auto-ajustável em uma planta de nível presente no Laboratório de Sistemas de Controle, Automação e Robótica da Universidade Federal do Rio Grande do Sul, que conta com dispositivos tipicamente industriais, onde vários trabalhos foram desenvolvidos (Campestrini et al., 2016), (Boeira, 2016). Como se trata de um sistema de múltiplas entradas e saídas, tanto o estimador quanto o controlador precisam ser adaptados para este caso.

## 6 Referências

- Allias, M. H., Muhammad, Z., Yusoff, Z. M., Rahiman, M. H. F., 2017. Implementation of first order model reference adaptive control (mrac) on regulating temperature of essential oil extraction process. International Conference on System Engineering and Technology.
- Åström, K. J., 1983. Theory and applications of adaptive control—a survey. *Automatica* 19 (5), 471–486.
- Åström, K. J., Wittenmark, B., 1973. On self-tuning regulators. *Automatica*. 9, 185.
- Barkana, I., 1983. Direct multivariable model reference adaptive control with applications to large structural systems. Ph.D. thesis, Rensselaer Polytechnic Institute.
- Barkana, I., 2013. Simple adaptive control - a stable direct model reference adaptive control methodology - brief survey. *Int. J. Adapt Control Signal Process.* 28(7), 567–603 (2014). doi:10.1002/acs.2411.
- Bazanella, A., Gomes da Silva Jr., J., 2006. *Sistemas de controle: Princípios e métodos de projeto*. UFRGS Editora.
- Bazanella, A. S., 1993. *Controle Auto-Ajustável de Sistemas de Potência*. Master's thesis, UFSC.
- Boeira, E. C., 2016. Projeto de controladores descentralizados em um processo de nível multivariável através do método do ponto crítico. Trabalho de Conclusão de Curso UFRGS.
- Campestrini, L., Eckhard, D., Bazanella, A. S., 2016. Data-driven model reference control design by prediction error identification. *Journal of the Franklin Institute*.
- Capitaneanu, S., Knecht, P., 2017. Online adaptive controller for continuous and batch processes. ISA Food and Pharmaceutical Industries Division Symposium.
- De Keyser, R., 2016. Adaptive dead-time estimation. In: *Proceedings of the IFAC Adaptive Systems in Control and Signal Processing Conference*. pp. 385–389.
- Fan, E. Z. W., 2010. Simulation research of paper-making process using model reference adaptive smith predictor. 3rd International Conference on Advanced Computer Theory and Engineering(ICACTION).
- Fortescue, T. R., Kershenbaum, L. S., Ydstie, B. E., 1981. Implementation of self-tuning regulators with variable forgetting factors. *Automatica*. V7, N6.
- Gohari, P., 2006. Discrete equivalents. <http://web.cecs.pdx.edu/~tymerski/ece452/6.pdf>, acessado: 24/05/2016.
- Gregory, P., 1959. Self adaptive flight control. Flight Control Symposium.

- Kempf, A., 2003. Avaliação de Desempenho de Malhas de Controle. Master's thesis, UFRGS.
- Landau, I., 1974. A survey of model reference adaptive techniques—theory and applications. *Automatica* 10 (4), 353–379.
- Landau, I. D., 2005. Commande des systèmes, conception, identification e mise en oeuvre. [http://www.gipsa-lab.grenoble-inp.fr/~ioandore.landau/identificationandcontrol/french/index\\_FR\\_TableDeMatières\\_Chapitre6.htm](http://www.gipsa-lab.grenoble-inp.fr/~ioandore.landau/identificationandcontrol/french/index_FR_TableDeMatières_Chapitre6.htm), acessado: 01/12/2017.
- Lavretsky, E., Wise, K. A., 2013. Robust and Adaptive Control With Aerospace Applications, first edition Edition. Springer London Heidelberg New York Dordrecht.
- Mirkin, B. M., Gutman, P.-O., 2004. Adaptive control of linear time delay systems. In: American Control Conference, 2004. Proceedings of the 2004. Vol. 2. IEEE, pp. 1241–1246.
- Monopoli, R. V., 1974. Model reference adaptive control with an augmented error signal. *IEEE Transactions on Automatic Control* 19 (5), 474–484.
- Parks, P., 1966. Liapunov redesign of model reference adaptive control systems. *IEEE Transactions on Automatic Control* 11 (3), 362–367.
- Rowell, D., 2006. Introduction to recursive-least-squares (rls) adaptive filters. <http://ocw.mit.edu/courses/mechanical-engineering/2-161-signal-processing-continuous-and-discrete-fall-2008/study-materials/rls.pdf>, acessado: 24/05/2016.
- Smith, O. J. M., 1959. A controller to overcome dead time. *ISA J.*, 6, 28/33.
- Tao, G., 2014. Multivariable adaptive control: A survey. *Automatica*. 50 (11): 2737–2764.
- Unity Pro, U. M., 2011. Unity pro user manual, schneider electric.
- Whitaker, H. P., Yamron, J., Kezer, A., 1958. Design of model-reference adaptive control systems for aircraft. Massachusetts Institute of Technology, Instrumentation Laboratory.
- Yamanaka, N. A. . K., 2003. Smith predictor control and internal model control - a tutorial. SICE Annual Conference in Fukui.
- Ziegler, J. G., Nichols, N. B., 1942. Optimum settings for automatic controllers. *Transactions of the ASME*.
- Åström, K., 2008. Adaptive Control. Dover.
- Åström, K. J., Wittenmark, B., 1995. Adaptive Control, second edition Edition. Addison-Wesley.

## **Apêndices**

## A Algoritmo dos Mínimos Quadrados Recursivos para CLP

```
(*
This Algorithm implements the classic Recursive Least Squares of
a
first order system, with some modifications regarding the
covariance
windup (conditional update) and robust estimation reducing the
effect
of large errors.
```

Continuous system:  $H(s)=kp/(tau*s+1)$

Discrete system:  $y(n)=u(n-1)*b2 + y(n-1)*b1$

Variables:

f: regressors

```
f= |y(n-1)|
    |u(n-1)|
```

b: identified parameters

```
b= |b1|
    |b2|
```

R\_1: Covariance matrix

```
R_1= |r0  r1|
      |r2  r3|
```

k: Kalman filter gain

```
k= |k0|
    |k1|
```

\*)

```
SAMPLETM (INTERVAL := Te, Q => tick);
```

```
if tick = TRUE then
```

```
    (* Initialisation *)
```

```
    If (initialise=TRUE) then
```

```
        f[0]:=0.0;
```

```
        f[1]:=0.0;
```

```

    realTe:=time_to_real(Te)/1000.0;
    b[0]:=EXP_REAL(-realTe/(tau_init));
    b[1]:=kp_init*(1.0-EXP_REAL(-realTe/tau_init));

    kp:=kp_init;
    tau:=tau_init;
    kp_est:=kp;
    tau_est:=real_to_time(tau*1000.0);

    R_1[0]:=10000.0;
    R_1[1]:=0.0;
    R_1[2]:=0.0;
    R_1[3]:=10000.0;

    initialise:=FALSE;
end_if;

(* Estimation Error *)
estimationError:=y-(f[0]*b[0]+f[1]*b[1]);

(* Estimation Reset *)
if (reset = TRUE) then
    R_1[0]:=10000.0;
    R_1[1]:=0.0;
    R_1[2]:=0.0;
    R_1[3]:=10000.0;

    b[0]:=EXP_REAL(-realTe/(tau_init*1000.0));
    b[1]:=kp_init*(1.0-EXP_REAL(-realTe/tau_init));

    kp:=kp_init;
    tau:=tau_init;
    kp_est:=kp;
    tau_est:=real_to_time(tau*1000.0);
elseif (enable = TRUE) then (* Recursive algorithm *)
    aux := f[0]*(R_1[0]*f[0] + f[1]*R_1[2]) + f[1]*(f
        [0]*R_1[1] + f[1]*R_1[3]);

    if (aux > (condUpCnt*(1.0-forgettingFactor)))
        then (* Conditional Update test *)

```

```

(* Kalman Filter Gain *)
k[0]:=(R_1[0]*f[0]+R_1[1]*f[1])/
  forgettingFactor+aux);
k[1]:=(R_1[2]*f[0]+R_1[3]*f[1])/
  forgettingFactor+aux);

(* Covariance matix update *)
R_1_old[0]:=R_1[0];
R_1_old[1]:=R_1[1];
R_1_old[2]:=R_1[2];
R_1_old[3]:=R_1[3];

R_1[0]:=(R_1_old[0]-(k[0]*(f[0]*R_1_old
  [0]+f[1]*R_1_old[2])))/
  forgettingFactor;
R_1[1]:=(R_1_old[1]-(k[0]*(f[0]*R_1_old
  [1]+f[1]*R_1_old[3])))/
  forgettingFactor;
R_1[2]:=(R_1_old[2]-(k[1]*(f[0]*R_1_old
  [0]+f[1]*R_1_old[2])))/
  forgettingFactor;
R_1[3]:=(R_1_old[3]-(k[1]*(f[0]*R_1_old
  [1]+f[1]*R_1_old[3])))/
  forgettingFactor;

(* Robust estimation: reduces the effect
  of large errors *)
eps:=estimationError/(1.0+robustCnt*abs(
  estimationError));

(* Estimation update *)
b[0]:=b[0]+k[0]*eps;
b[1]:=b[1]+k[1]*eps;

if b[0]<=0.0 then
  b[0]:=EXP_REAL(-realTe/(tau_init)
  );
  test4:=true;
end_if;
if b[0]=1.0 then
  b[0]:=EXP_REAL(-realTe/(tau_init)

```



```

    );
end_if;

realTe:=time_to_real(Te)/1000.0;
tau := -realTe/LN_REAL(b[0]);
kp := b[1]/(1.0 - b[0]);

(* Testing limits for bounded estimation
*)
if tau < (1.0 - paraLimit[0]) * tau_init then
    tau := (1.0 - paraLimit[0]) * tau_init;
    b[0] := EXP_REAL(-realTe / tau);
    test1 := true;
elseif tau > (1.0 + paraLimit[0]) * tau_init
then
    tau := (1.0 + paraLimit[0]) * tau_init;
    b[0] := EXP_REAL(-realTe / tau);
    test2 := true;
end_if;

if kp < (1.0 - paraLimit[1]) * kp_init then
    kp := (1.0 - paraLimit[1]) * kp_init;
    b[1] := kp * (1.0 - b[0]);
    test3 := true;
elseif kp > (1.0 + paraLimit[1]) * kp_init then
    kp := (1.0 + paraLimit[1]) * kp_init;
    b[1] := kp * (1.0 - b[0]);

end_if;

kp_est := kp;
tau_est := real_to_time(tau * 1000.0);

end_if;
end_if;

(* Regressors update *)
f[0] := y;
f[1] := u;
end_if;
tick := FALSE;

```

## B MRAC com Preditor de Smith

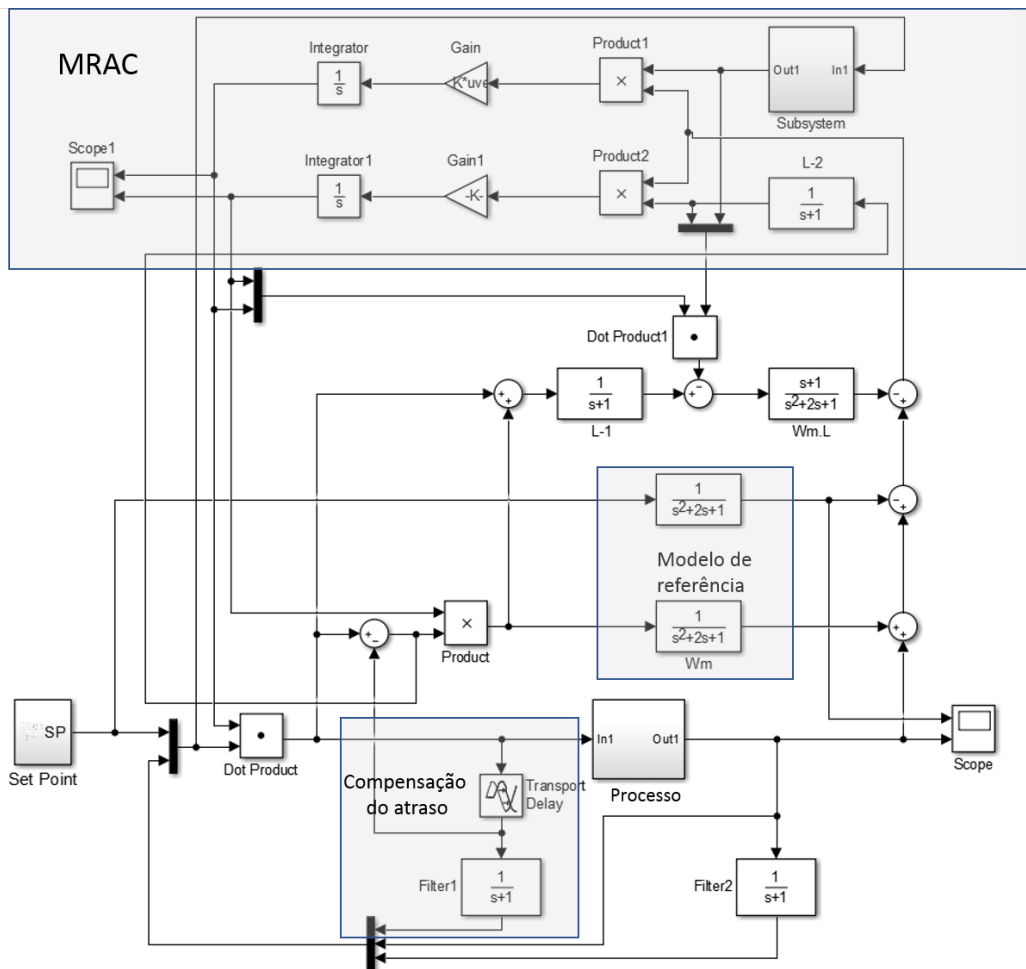


Figura 20: Diagrama de blocos do sistema de controle composto pela fusão de um preditor de Smith com o MRAC. Este método de controle combina a adaptação paramétrica do MRAC, que garante a adaptação do controlador às mudanças do processo e ao mesmo tempo possui as características e compensação de atraso do preditor de Smith. Para isso, é adotado um modelo de referência e o atraso de transporte da planta deve ser conhecido para que possa ser compensado.