

**DAS** Departamento de Automação e Sistemas  
**CTC** Centro Tecnológico  
**UFSC** Universidade Federal de Santa Catarina

# Identificação online de parâmetros e estados de uma planta não linear

*Relatório submetido à Universidade Federal de Santa Catarina*

*como requisito para a aprovação da disciplina:*

*DAS 5511: Projeto de Fim de Curso*

*Bruno Martins Lima*

*Florianópolis, Fevereiro de 2016*



# Identificação online de parâmetros e estados de uma planta não linear

*Bruno Martins Lima*

Esta monografia foi julgada no contexto da disciplina  
**DAS 5511: Projeto de Fim de Curso**  
e aprovada na sua forma final pelo  
**Curso de Engenharia de Controle e Automação**

*Prof. Julio Elias Normey Rico*

---

**Banca Examinadora:**

Prof. Julio Elias Normey Rico  
**Orientador no Curso**

Prof. Hector Bessa Silveira  
Prof. Julio Elias Normey Rico  
**Responsáveis pela disciplina**

Dr. Agostinho Plucenio  
**Avaliador**

Gérson Jonatan Pinheiro dos Santos  
Júlio César Trevisan Dal Bem  
**Debatedores**

# Agradecimentos

Ao Daniel pela orientação, conselhos e ajuda por todo o período de faculdade.

À minha família por sempre ter me apoiado nos estudos, mesmo não entendendo do que se trata meu curso.

Às minhas flatmates Bárbara, Bruna e Carol por me mostrar uma visão de mundo diferente e pelos momentos inesquecíveis.

Aos meus amigos de turma pela companhia e apoio, mesmo nos momentos mais difíceis.

Ao Julio pela orientação, paciência e pela oportunidade de trabalhar com controle avançado.



# Resumo

O objetivo do trabalho é o estudo de algoritmos de identificação online de parâmetros e estados de plantas não lineares. Foi feito um estudo comparativo entre o Filtro de Kalman Extendido (FKE) e o Moving Horizon Estimation (MHE). A segunda técnica, em particular, apresenta vantagens no caso não linear pelo uso das restrições e do modelo não linear. Foi realizado, também, o estudo de técnicas de controle preditivo com ênfase no DMC. Para ilustrar os conceitos apresentados, foi realizado um estudo de caso com a planta Continuous Stirred Tank Reactor (CSTR) simulado em MATLAB.

**Palavras-chave:** Estimação de estados e parâmetros, filtro de Kalman, filtro de Kalman extendido, moving horizon estimation, controle preditivo, DMC recursivo.





# Abstract

The main objective of this work is the study of online state and parameter estimation algorithms of nonlinear systems. A comparative study between Extended Kalman Filter (EKF) and Moving Horizon Estimation (MHE) was made. The second technique, in particular, shows advantages in nonlinear cases because of the use of constraints and of the nonlinear model. It was also made the study of Model Predictive Control, DMC in particular. To illustrate the concepts presented, a case study was implemented in a plant called Continuous Stirred Tank React (CSTR) simulated in MATLAB.

**Keywords:** state and parameter estimation, Kalman filter, Extended Kalman filter, moving horizon estimation, model predictive control, recursive DMC.



# Lista de ilustrações

Figura 1 – Algoritmo MPC. . . . .	22
Figura 2 – Horizontes do MPC. . . . .	29
Figura 3 – Sistema unidimensional de um trem. . . . .	34
Figura 4 – Conhecimento inicial do sistema mostrando a distribuição da variável (posição). . . . .	34
Figura 5 – Conhecimento após uma amostra do sistema, utilizando somente o modelo. . . . .	34
Figura 6 – Aqui é mostrado as duas estimativas: uma do modelo e outra do medidor. . . . .	35
Figura 7 – Em verde, é mostrado a nova distribuição e a estimativa atualizada dos estados. . . . .	35
Figura 8 – Horizonte deslizante do MHE. . . . .	37
Figura 9 – Representação do CSTR. . . . .	43
Figura 10 – Estimação do parâmetro $U$ . Resultado de [1] no topo e implementação abaixo. . . . .	47
Figura 11 – Estimação do parâmetro $E_a$ . Resultado de [1] no topo e implementação abaixo. . . . .	48
Figura 12 – Estimação da concentração $C$ . Resultado de [1] no topo e implementação abaixo. . . . .	49
Figura 13 – Estimação da temperatura $T$ . Resultado de [1] no topo e implementação abaixo. . . . .	49
Figura 14 – Simulação do sistema em malha aberta. Temperatura em graus Celsius. . . . .	50
Figura 15 – Evolução dos estados do sistema em malha fechada. Temperatura em graus Celsius. . . . .	53
Figura 16 – Estimação dos parâmetros em malha fechada. . . . .	54
Figura 17 – Comparação entre MHE e o FKE na estimação da concentração ( $C$ ). . . . .	55
Figura 18 – Comparação entre MHE e o FKE na estimação da temperatura ( $T$ ). . . . .	55
Figura 19 – Comparação entre MHE e o FKE para na estimação do parâmetro $U$ . . . . .	56
Figura 20 – Comparação entre MHE e o FKE na estimação do parâmetro $E_a$ . . . . .	56



# Lista de tabelas

Tabela 1 – Comparação entre MHE e EKF. . . . .	40
Tabela 2 – Estados, entradas e parâmetros do CSTR. . . . .	45
Tabela 3 – Características dos estados e parâmetros desconhecidos do CSTR. . . . .	45
Tabela 4 – Parâmetros de implementação do MHE . . . . .	47
Tabela 5 – Sintonia do controlador. Horizontes em amostras e em segundos. . . . .	52



# Lista de abreviaturas e siglas

MPC	<i>Model Predictive Control</i>
NMPC	<i>Nonlinear Model Predictive Control</i>
MIMO	Processos com múltiplas entradas e saídas ( <i>Multi-Input-Multi-Outout</i> )
SISO	Processo Monovariável - <i>Single Input Single Output</i>
EKF	<i>Extended Kalman Filter</i>
FKE	<i>Filtro de Kalman Extendido</i>
MHE	<i>Moving Horizon Estimation</i>
DMC	<i>Dynamic Matrix Control</i>
QP	<i>Quadratic Programming</i>
SQP	<i>Sequential Quadratic Programming</i>
DAS	Departamento de Automação e Sistemas
MAC	<i>Model Algorithmic Control</i>
GPC	<i>Generalized Predictive Control</i>
EPSAC	<i>Extended Prediction Self-Adaptive Control</i>
UPC	<i>Unified Predictive Control</i>





# Sumário

1	INTRODUÇÃO . . . . .	17
1.1	NMPC e estimadores não lineares . . . . .	18
1.2	Metodologia e organização do trabalho . . . . .	18
2	FUNDAMENTOS DE MPC . . . . .	21
2.1	Controle Preditivo . . . . .	21
2.2	Modelo de predição . . . . .	22
2.2.1	Resposta impulsiva . . . . .	22
2.2.2	Resposta ao degrau . . . . .	22
2.2.3	Função de transferência . . . . .	23
2.2.4	Espaço de estados . . . . .	23
2.2.5	Modelos não lineares . . . . .	23
2.2.6	Modelo de predição do DMC . . . . .	24
2.2.6.1	DMC Recursivo . . . . .	25
2.2.7	Resposta livre não linear . . . . .	27
2.3	Função objetivo . . . . .	28
2.3.1	Restrições . . . . .	28
2.4	Sintonia . . . . .	29
2.5	Considerações finais . . . . .	30
3	ESTIMADORES EM CONTROLE PREDITIVO . . . . .	31
3.1	Estimadores . . . . .	31
3.1.1	Modelo não linear . . . . .	32
3.2	Filtro de Kalman . . . . .	32
3.2.1	Exemplo ilustrativo . . . . .	33
3.2.2	Filtro de Kalman Extendido . . . . .	35
3.3	Moving Horizon Estimation . . . . .	37
3.3.1	Formulação do MHE . . . . .	38
3.3.1.1	Arrival Cost . . . . .	38
3.3.2	Aspectos de implementação . . . . .	39
3.4	Conclusão . . . . .	40
4	ESTUDOS DE CASO: CSTR . . . . .	43
4.1	Descrição da planta . . . . .	43
4.1.1	Modelagem . . . . .	44
4.2	Implementação do MHE . . . . .	46

4.2.1	Resultados . . . . .	47
4.3	Implementação do controle . . . . .	50
4.3.1	Malha aberta . . . . .	50
4.3.2	Controlador e Sintonia . . . . .	51
4.3.2.1	Predição da resposta livre . . . . .	51
4.3.2.2	Cálculo da ação de controle . . . . .	51
4.3.2.3	Sintonia . . . . .	51
4.3.3	Resultados . . . . .	52
4.4	Implementação do FKE . . . . .	52
4.4.1	Resultados . . . . .	54
4.5	Conclusão . . . . .	54
5	CONCLUSÕES E PERSPECTIVAS . . . . .	59
5.1	Trabalhos futuros . . . . .	59
	REFERÊNCIAS . . . . .	61

# 1 Introdução

Os controladores preditivos, do inglês Model Predictive Control (MPC) surgiram no final da década de 70 e nos últimos anos tem se transformado em uma poderosa e prática técnica de controle de processos industriais. O MPC é uma das poucas técnicas avançadas de controle que tem alcançado impacto significativo em sistemas de controle industriais, principalmente na indústria petroquímica. A principal razão desse sucesso é, talvez, a capacidade do MPC em lidar com as seguintes situações:

- Possibilidade de aplicação em sistemas SISO (uma entrada para uma saída) e MIMO (múltiplas entradas para múltiplas saídas);
- A realimentação do sistema e o controle feedforward podem ser incluídos de forma direta na formulação do MPC;
- Restrições de entrada e saída podem ser incluídas na formulação da lei de controle através da otimização em linha;
- Compensação intrínseca de atrasos de transporte.

O MPC não é uma estratégia de controle específica, mas é o nome dado a um conjunto muito grande de métodos de controle que foram desenvolvidos considerando algumas idéias comuns baseadas no conceito de predição. Nos algoritmos MPC, com o controle e o erro atuais e um modelo calcula-se primeiro a predição da saída futura do processo; com estas informações um módulo de otimização calcula o sinal de controle a ser aplicado ao processo no próximo passo considerando a minimização de um determinado índice de desempenho e as restrições nas variáveis de processo e/ou de controle.

A maioria dos sistemas reais tem dinâmica não linear, porém, quando o processo opera numa pequena faixa, sua dinâmica pode ser aproximada satisfatoriamente por modelos lineares. Isto tem motivado o desenvolvimento de muitas técnicas de MPC que utilizam modelos lineares, como por exemplo, o DMC (Matriz Dinâmica de Controle), o MAC (Controle Algorítmico Baseado em Modelo) ou o GPC (Controle Preditivo Generalizado).

São muitas as aplicações de controle baseadas em softwares comerciais que implementam MPC linear nas refinarias de petróleo no Brasil e no mundo, como é o caso da Petrobras. As principais vantagens do MPC linear estão vinculadas as facilidades de obtenção de modelos lineares, se comparados aos não lineares, e as menores dificuldades para se resolver os problemas de otimização associados. Porém, quando os processos têm dinâmica muito não linear ou quando a faixa de operação é variável, então necessariamente

deverá ser tomado em conta o modelo não linear no projeto do controle, de forma que permita manter o desempenho desejado para o sistema em malha fechada.

## 1.1 NMPC e estimadores não lineares

Atualmente, em muitas aplicações de MPC linear na indústria do petróleo e gás, se encontram comportamentos aquém do esperado. Também, em muitas aplicações, como no controle dos processos das plataformas de petróleo, o controle avançado com base no MPC linear não tem sido usado por causa das dificuldades impostas pelas dinâmicas não lineares do processo. Resultados de pesquisa iniciais desenvolvidos no DAS apontam que o desempenho destes e outros processos poderia ser bastante melhorado utilizando-se técnicas de controle preditivo não linear.

Do ponto de vista conceitual o MPC não linear (NMPC) não apresenta dificuldades. Considerando a mesma função objetivo que no caso linear e um modelo não linear para o cálculo das predições, o algoritmo consiste novamente em encontrar o controle que minimize a função objetivo num horizonte deslizante. Do ponto de vista prático, a planta não linear apresenta como desafio principal a dificuldade para obter o sinal de controle pois, em geral, é necessário a solução de um problema otimização não linear.

Além das questões associadas aos problemas de otimização, outra parte importante no projeto de uma solução de controle é o estimador de estados e parâmetros. De forma geral, é impossível ou inviável ter acesso por meio de sensores a todos os estados do processo e muitas das vezes as medições apresentam ruídos significativos. Assim, estimadores de estados são necessários para poder ter as informações necessárias para o cálculo das predições usadas no algoritmo MPC. Além disso, várias plantas apresentam parâmetros que mudam com o tempo tornando o monitoramento deles importante para segurança ou performance do controlador. Para resolver esses problemas, é necessário um estimador.

No caso de plantas e controladores não-lineares, o projeto de um estimador de estados pode ser bastante complexo. Existem na literatura diversas abordagens para o projeto de estimadores, mas é importante estudar quais se adaptam melhor ao uso juntamente com técnicas de controle preditivo não linear. Assim, este Projeto de Fim de Curso propõe realizar o estudo e implementação de uma das técnicas chamadas Moving Horizon Estimation, que por suas características é a estratégia de estimação mais interessante para o uso em controladores MPC.

## 1.2 Metodologia e organização do trabalho

Como metodologia de trabalho se realizou num primeiro momento um estudo teórico de controle preditivo e estimadores. Em seguida se implementaram as estratégias

de controle e de estimação em ambiente MATLAB.

Como implementação prática, foi realizado um estudo de caso de um tanque reator de fluxo contínuo simulado em MATLAB. Neste exemplo, foi implementado e analisado um MHE para a estimação de dois estados e dois parâmetros. Em seguida, o mesmo estimador foi utilizado juntamente com um MPC simples para controlar a planta.

Numa segunda etapa, foi implementado o estimador baseado em filtro de Kalman para comparação com o MHE. Os resultados de simulação foram analisados para concluir sobre as propriedades do método desenvolvido.

O presente relatório está organizado da seguinte maneira. Os conceitos básicos de controle preditivo necessários para o entedimento do projeto encontram-se no capítulo 2. O capítulo 3 é dedicado ao estudo das técnicas de estimadores de estados e parâmetros com foco no Moving Horizon Estimation. O capítulo 4 é dedicado à implementação dos estimadores e aos resultados.



## 2 Fundamentos de MPC

### 2.1 Controle Preditivo

O controle preditivo baseado em modelo (MPC) é um conjunto de técnicas de controle avançado mais potente e provavelmente o que teve mais êxito na indústria. As principais causas deste sucesso são as aplicações em sistemas SISO e MIMO, possibilidade de incorporar restrições na lei de controle e o uso natural em sistemas com atraso.

O MPC teve início nas indústrias petroquímicas por volta dos anos 70, com MPC MIMO com restrições. A grande aceitação chamou atenção da academia e nos anos 80 surgiram os primeiros trabalhos formais sobre o assunto. Em meados dos anos 90 o MPC linear chegou ao auge e iniciou-se, então, o estudo teórico para os casos não lineares e híbridos. Entre as características comuns destes sistemas:

- Usam um modelo explícito do processo para prever a saída;
- Calculam as ações de controle para todo o horizonte a partir da minimização de uma função objetivo;
- O horizonte é deslizante, ou seja, é deslocado para frente a cada período de amostragem.

Na figura 1, pode ser visto um diagrama simples do MPC. Nela, é mostrado como o MPC utiliza explicitamente o modelo do processo para calcular uma previsão do comportamento futuro da planta. Em seguida, esse dado é comparado com a saída real da planta e utilizado no cálculo do controle. O controlador visa levar o processo para a referência respeitando as restrições do problema.

Ao longo dos anos, diversos algoritmos de controle preditivo surgiram tanto na indústria como nos centros de pesquisa mas todos possuem três elementos básicos: modelo de previsão, função objetivo e obtenção da lei de controle. O *Dynamic Matrix Control* (DMC), que surgiu com o artigo [2], se destaca por utilizar somente a resposta ao degrau do sistema. Este tipo de modelo possui a vantagem de não necessitar de conceitos como função de transferência ou espaço de estados para ser entendido. Assim, sua aceitação é grande por pessoas sem conhecimento na área de controle, pois seu entendimento e sua obtenção é relativamente mais fácil.

Nos próximos itens serão explicados em detalhes os conceitos de MPC com foco em DMC e sistemas não-lineares. O uso de restrições e a extensão para o caso MIMO é explorada em [3], livro sobre controle preditivo. Em [4], são sugeridas modificações no

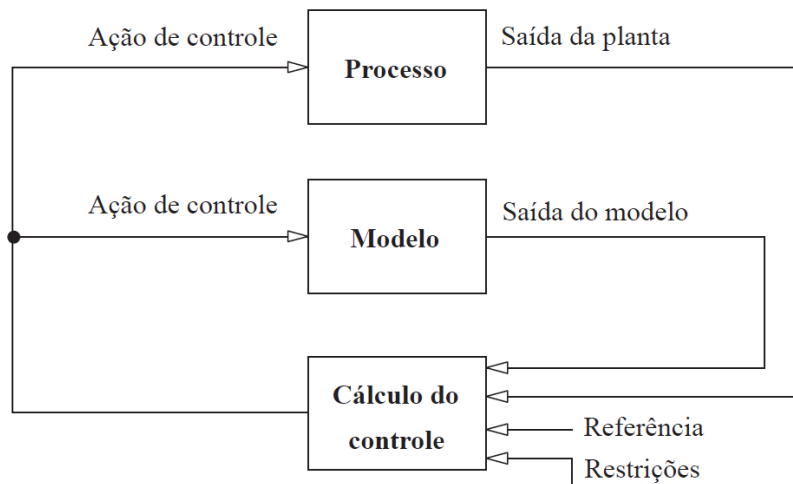


Figura 1 – Algoritmo MPC.

algoritmo DMC para melhorar certas propriedades como robustez e rejeição de perturbações através da inclusão de um filtro.

## 2.2 Modelo de predição

O modelo de predição é, provavelmente, o elemento mais importante dentro do controlador, dado que ele deve ser capaz de representar adequadamente a dinâmica do processo, deve permitir o cálculo das predições da saída do processo, ser intuitivo e, ao mesmo tempo, permitir uma análise teórica do sistema. Os modelos mais usados são:

### 2.2.1 Resposta impulsiva

É utilizado no MAC e em casos especiais no GPC. A relação entrada e saída é dada por:

$$y(t) = \sum_{i=1}^{\infty} h_i u(t-i) \quad (2.1)$$

onde  $h_i$  são as amostras da saída do processo correspondente à aplicação de um impulso na entrada do mesmo. Em geral, como esta sequência é infinita, a resposta é truncada para os primeiros  $N$  valores. Assim, o modelo só pode ser usado com plantas estáveis, onde  $h_i \rightarrow 0$  quando  $i \rightarrow \infty$ .

### 2.2.2 Resposta ao degrau

É usado pelo DMC e suas variantes. É similar ao anterior mas usa um degrau unitário como sinal de entrada. Por isso, este modelo tem as mesmas vantagens e inconvenientes



que o anterior. Para sistemas estáveis a resposta é dada por:

$$y(t) = y_o + \sum_{i=1}^{\infty} g_i \Delta u(t-i) \quad (2.2)$$

onde os  $g_i$  são as amostras da saída obtida ao aplicar o degrau e  $\Delta u(t) = u(t) - u(t-1)$ . A relação entre este modelo e o de resposta impulsiva é dada por:

$$h_i = g_i - g_{i-1} \quad g_i = \sum_{j=1}^i h_j \quad (2.3)$$

### 2.2.3 Função de transferência

Este modelo é usado no GPC, UPC, EPSAC, EHAC, entre outros. Utiliza o conceito de função de transferência como, por exemplo,  $G(z) = \frac{B(z)}{A(z)}$ . Esta representação tem como vantagens principais poder ser usada para plantas instáveis e precisar, em geral, de poucos parâmetros para descrever o comportamento do sistema. Já seu principal inconveniente é a necessidade de conhecer *a priori* a ordem dos polinômios  $A$  e  $B$  quando o modelo deve ser identificado a partir de dados experimentais.

### 2.2.4 Espaço de estados

Usa o conjunto de equações de estados e saídas no formato:

$$\begin{aligned} \mathbf{x}(t) &= A\mathbf{x}(t-1) + B\mathbf{u}(t-1) \\ \mathbf{y}(t) &= C\mathbf{x}(t) \end{aligned} \quad (2.4)$$

Sua principal vantagem é que pode ser usada diretamente para processos multivariáveis. Como inconvenientes, cabe mencionar que, em geral, os estados não tem significado físico e que é necessário o uso de observadores, aumentando assim a complexidade de cálculo do controle.

### 2.2.5 Modelos não lineares

Modelos não lineares genéricos também podem ser usados para descrever a dinâmica do processo quando os modelos lineares não geram bons resultados. Redes neurais ou lógica nebulosa (fuzzy logic) podem ser usadas em algumas aplicações como modelo de predição. Modelos fenomenológicos são também bastante usados nos casos que há conhecimento do processo. Em todos estes casos, a solução do problema apresenta alguns problemas adicionais ao do caso linear como:

- Obtenção da estrutura e os parâmetros do modelo do processo;

- O tempo de processamento necessário para o cálculo das predições que, para sistemas complexos, pode ser muito grande;
- O problema de controle se torna não linear e não-convexo, ou seja, não há garantias de ótimo global [3]<sup>1</sup>.

Uma das modificações do controle preditivo que procura contornar esses problemas é o cálculo da resposta livre a partir do modelo não linear da planta que será discutido na seção 2.2.7. Antes, será explicado o modelo de predição do DMC para expor os conceitos de MPC.

## 2.2.6 Modelo de predição do DMC

Nesta seção, será analisado em mais detalhes o o modelo de predição do DMC. A resposta ao degrau do sistema é expressa pela equação 2.5,

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t-i) + \hat{n}(t+k|t) \quad (2.5)$$

sendo  $\Delta u(t) = u(t) - u(t-1)$ ,  $g_i$  as amostras da saída no ensaio ao degrau e  $\hat{n}(t+k|t)$  as perturbações futuras. Assim, as predições futuras da saída dado as informações em  $t$  podem ser escritas da seguinte forma:

$$\hat{y}(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + \hat{n}(t+k|t) \quad (2.6)$$

Como a perturbação é considerada constante ao longo do horizonte a predição da mesma pode ser escrita por  $\hat{n}(t+k|t) = \hat{n}(t|t) = y_m(t) - \hat{y}(t|t)$ . Com isso pode-se rescrever:

$$\hat{y}(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + \hat{n}(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + y_m(t) - \sum_{i=1}^{\infty} g_i \Delta u(t-i)$$

$$\hat{y}(t+k|t) = \sum_{i=1}^k g_i \Delta u(t+k-i) + f(t+k) \quad (2.7)$$

onde  $f(t+k)$  é a resposta livre do sistema, dada pela equação 2.8:

$$f(t+k) = y_m(t) + \sum_{i=1}^{\infty} (g_{k+1} - g_i) \Delta u(t-1) \quad (2.8)$$

<sup>1</sup> O uso de modelos não lineares somente no cálculo da resposta livre não torna o a otimização não linear [3] contornando este problema.

Se o sistema é estável, os coeficientes  $g_i$  da resposta ao degrau tendem a um valor constante após  $N$  amostras, logo,  $g_{k+i} - g_i \approx 0$ , para  $i > N$ , com isso a resposta livre da equação 2.8 pode ser calculada conforme equação 2.9.

$$f(t+k) = y_m(t) + \sum_{i=1}^N (g_{k+1} - g_i) \Delta u(t-1) \quad (2.9)$$

Assim pode-se calcular as predições ao longo do horizonte de predição ( $k = 1, \dots, p$ ), considerando  $m$  ações de controle:

$$\begin{aligned} \hat{y}(t+1|t) &= g_1 \Delta u(t) + f(t+1) \\ \hat{y}(t+2|t) &= g_2 \Delta u(t) + g_1 \Delta u(t+1) + f(t+2) \\ &\vdots \\ \hat{y}(t+p|t) &= \sum_{i=p-m+1}^p g_i \Delta u(t+p-i) + f(t+p) \end{aligned}$$

As predições do sistema de forma matricial pode ser escrita conforme equação abaixo:

$$\hat{\mathbf{y}} = \mathbf{G}\mathbf{u} + \mathbf{f} \quad (2.10)$$

onde,  $\mathbf{G}$  é a matriz dinâmica do sistema,  $\mathbf{u}$  vetor das ações de controle futuras, e  $\mathbf{f}$  vetor da resposta livre do sistema:

$$\mathbf{G} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_m & g_{m-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_p & g_{p-1} & \cdots & g_{p-m+1} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+m) \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} f(t+1) \\ f(t+2) \\ \vdots \\ \vdots \\ f(t+p) \end{bmatrix} \quad (2.11)$$

Como se observa da equação 2.11, para o cálculo de  $\hat{\mathbf{y}}$  seria necessário armazenar os valores passados do controle. Uma forma de simplificar este cálculo consiste no uso do DMC recursivo apresentando abaixo.

### 2.2.6.1 DMC Recursivo

Em [4, 5], demonstrou-se a propriedade recursiva do DMC, que será apresentada a seguir. Esta implementação não altera o funcionamento do DMC mas ela é computacionalmente mais eficiente e mais simples de ser implementada.

As previsões de malha aberta em  $t + k$  dado as informações em  $t$  e  $t - 1$  são dadas a seguir:

$$\begin{aligned} y_o(t + k|t) &= \sum_{i=k+1}^{\infty} g_i \Delta u(t + k - i) \\ y_o(t + k|t - 1) &= \sum_{i=k+2}^{\infty} g_i \Delta u(t + k - i) \end{aligned}$$

A diferença entre a previsão de malha aberta em  $t + k$  dado  $t$ , e em  $t + k$  dado  $t - 1$ , é apenas a ação de controle nova  $\Delta u(t - 1)$  que não se conhecia em  $t - 1$ . Subtraindo as equações anteriores:

$$\begin{aligned} y_o(t + k|t) - y_o(t + k|t - 1) &= g_{k+1} \Delta u(t - 1) + \sum_{i=k+2}^{\infty} g_i \Delta u(t + k - i) \\ &\quad - \sum_{i=k+2}^{\infty} g_i \Delta u(t + k - i) \\ y_o(t + k|t) &= g_{k+1} \Delta u(t - 1) + y_o(t + k|t - 1) \end{aligned} \quad (2.12)$$

Assim, pelo resultado da equação 2.12, a previsão atual pode ser escrita pela previsão da amostra anterior mais a contribuição do incremento de controle.

A utilização do DMC recursivo é feita da seguinte forma: mantém-se em memória um vetor com  $M$  elementos<sup>2</sup>:  $\mathbf{Y}_{livre} = [y_o(t|t - 1), \dots, y_o(t + M - 1|t - 1)]^T$ . Seus elementos são as repostas livres dadas as ações de controle conhecidas até o instante  $t - 1$ . Ao iniciar o controlador DMC no instante  $t_0$ , dado que o sistema encontra-se em regime permanente, considera-se que as previsões de malha aberta sejam constantes e iguais ao valor de  $y(t_0)$ .

Ao iniciar a execução do algoritmo DMC no instante  $t$ , é preciso atualizar o vetor de resposta livre pois o incremento da ação de controle  $\Delta u(t - 1)$  agora é conhecido:

$$\mathbf{Y}_{livre} = \mathbf{Y}_{livre} + \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_M \end{bmatrix} \Delta u(t - 1) \quad (2.13)$$

Assim, obtém-se o vetor  $\mathbf{Y}_{livre}$  atualizado. Após obter o novo sinal de controle  $u(t)$ , é preciso “deslocar” os valores dentro do vetor  $\mathbf{Y}_{livre}$ , isto porque no instante posterior  $t + 1$ , serão precisos as repostas livres de  $t + 1$  até  $t + M$  baseadas nos dados até  $t$ . Então, o primeiro elemento é descartado pois é a previsão relativa ao instante  $t$  que já passou. O problema é que, ao se fazer este deslocamento, o último elemento, no caso a previsão em

<sup>2</sup> O valor de  $M$  deve ser significativamente maior que  $p$ .

$y(t + M|t)$ , é desconhecida. Porém, no caso estável,  $y_o(t + M|t) \cong y_o(t + M - 1|t)$ <sup>3</sup>, assim, o novo vetor  $\mathbf{Y}_{livre}$  será dado por:

$$\mathbf{Y}_{livre} = \begin{bmatrix} y_o(t + 1|t) \\ y_o(t + 2|t) \\ \vdots \\ y_o(t + M - 1|t) \\ y_o(t + M - 1|t) \end{bmatrix} \quad (2.14)$$

Assim, o cálculo da resposta livre considerada no DMC recursivo é dado da seguinte forma:

$$\mathbf{f}_r = \begin{bmatrix} y_o(t + 1|t) \\ \vdots \\ y_o(t + N|t) \end{bmatrix} + \mathbf{1}_N (y(t) - y_o(t|t)) \quad (2.15)$$

### 2.2.7 Resposta livre não linear

Para sistemas com não-linearidades importantes, o modelo linear descrito até aqui pode não ser o suficiente para a predição da saída. Assim, uma modificação no algoritmo se faz necessária. Uma das alternativas é dividir a predição do sistema em duas partes: a resposta livre ( $\mathbf{f}$ ) e a resposta forçada dada por  $\mathbf{Gu}$ .

A resposta livre é calculada utilizando um modelo não linear com as entradas do sistema constantes no horizonte. Já a forçada utiliza um modelo linear para encontrar a resposta devida aos incrementos de controle futuros. O resultado final é o mesmo:  $\hat{\mathbf{y}} = \mathbf{Gu} + \mathbf{f}$ .

A predição obtida desta forma é somente uma aproximação da verdadeira pois o princípio da superposição (que permite a divisão entre resposta livre e forçada) só é válido para sistemas lineares. Porém, essa técnica mostra melhores resultados que utilizar um modelo linearizado para ambas as respostas [3].

Esta técnica não modifica o cálculo do controle pela otimização de  $J$  pois a única diferença com o MPC linear é a resposta livre que utiliza o modelo não linear da planta. Isto ocorre pois a função objetivo continua linear nas variáveis de decisão ( $\Delta u$ ) o que leva a lei de controle simples mostrada na seção 2.3.

<sup>3</sup> Essa condição é mostrada em [4].

## 2.3 Função objetivo

Os vários algoritmos MPC propõem diferentes funções objetivo para obter a lei de controle, porém a grande maioria deles utiliza a mesma estrutura. A função objetivo geral para o caso SISO é apresentada na equação 2.16. Esta equação é facilmente estendida para o caso MIMO [3].

$$\mathbf{J} = \sum_{j=N_1}^{N_2} [\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2 \quad (2.16)$$

Onde,  $N_1, N_2$  são o mínimo e máximo horizonte de predição,  $N_u$  horizonte de controle,  $\lambda(j)$  são ponderações para ajuste do controle,  $\hat{y}(t+j|t)$  predições da saída,  $w(t+j)$  vetor de referências futuras e  $\Delta u(t+j-1)$  vetor de incrementos na ação de controle.  $\lambda(j)$  geralmente é constante para todo horizonte e, portanto, o índice  $j$  pode ser abandonado.

Procura-se, então, calcular os incrementos de controles futuros ( $\Delta u(t+j-1)$ ) que minimizam a função  $\mathbf{J}$ , utilizando o modelo do sistema dado por  $\hat{\mathbf{y}} = \mathbf{G}\mathbf{u} + \mathbf{f}$ . Para o caso sem restrições, o problema pode ser escrito na forma de programação quadrática (QP) e há uma solução analítica para a equação dada pela equação 2.17. A demonstração deste resultado pode ser encontrado em [4].

$$\mathbf{u} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{w} - \mathbf{f}_r) = \mathbf{K}(\mathbf{w} - \mathbf{f}_r) \quad (2.17)$$

Como o ganho  $\mathbf{K}$  só depende de  $\mathbf{G}$  e  $\lambda$ , ele é constante e pode ser calculado offline, i.e fora do laço de controle. Como o algoritmo é de horizonte deslizante, unicamente o primeiro sinal de controle é aplicado (primeiro elemento do vetor  $\mathbf{u}$ ), assim a lei de controle final é:

$$\Delta u(t) = \mathbf{K}_1(\mathbf{w} - \mathbf{f}_r) \quad (2.18)$$

onde  $\mathbf{K}_1$  é a primeira linha da matriz  $\mathbf{K}$ .

### 2.3.1 Restrições

As restrições mais comuns são as do tipo *bound* no controle, incremento de controle e saída como mostra o conjunto de equações 2.19. Restrições na entrada representam, geralmente, os limites físicos dos atuadores como valor máximo e mínimo de abertura de válvulas ou a máxima velocidade de variação de servomotores. Já as restrições nas saídas podem estar presentes devido a questões econômicas ou de segurança. Por estes motivos,

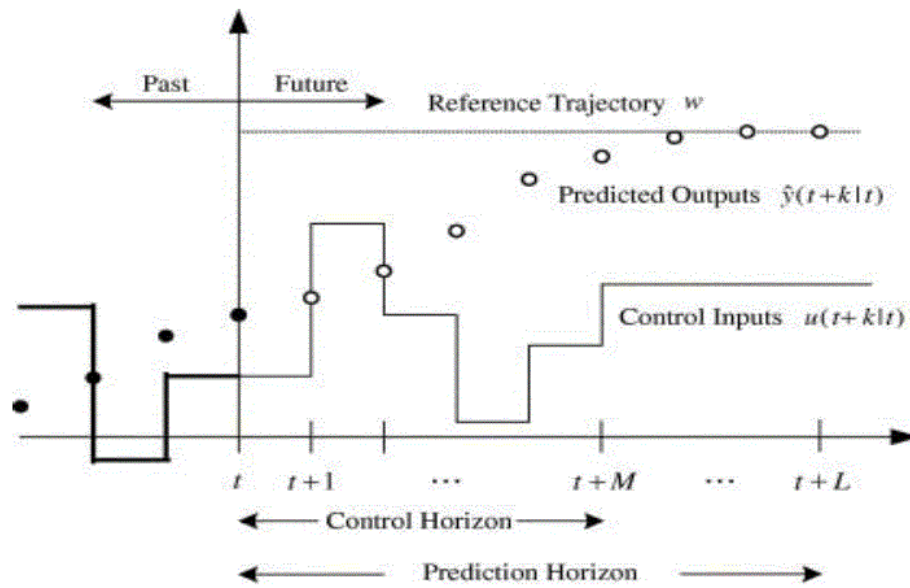


Figura 2 – Horizontes do MPC.

a inclusão das restrições na função objetivo que se deseja minimizar é importante.

$$\begin{aligned}
 u_{min} &\leq u(t) \leq u_{max} \\
 \Delta u_{min} &\leq \Delta u \leq \Delta u_{max} \\
 y_{min} &\leq y(t) \leq y_{max}
 \end{aligned}
 \tag{2.19}$$

Nestes casos, a solução do mínimo de  $J$  não pode ser obtida analiticamente e requer uma carga de cálculo bem maior que no caso sem restrições. Apesar desta dificuldade, hoje existem ferramentas computacionais que resolvem os problemas de otimização numérica de forma rápida e eficiente como o Sequential Quadratic Programming (SQP) ou o ponto interior. Assim, a capacidade dos algoritmos MPC de levar as restrições em consideração é o principal motivo do seu sucesso nas aplicações industriais.

## 2.4 Sintonia

Há dois aspectos que podem ser modificados no MPC SISO<sup>4</sup> para sintonizar o controlador [4]: a ponderação do controle e o tamanho dos horizontes. O aumento da ponderação  $\lambda$  é utilizado para penalizar os incrementos de controle tornando a atuação menos agressiva.

Além da ponderação, há dois horizontes que devem ser ajustados: horizonte de previsão e de controle. O horizonte de controle  $N_u$  indica em quais instantes futuros a entrada do sistema pode ser mudada. Um horizonte pequeno reduz os graus de liberdade

<sup>4</sup> Para o controlador MIMO, deve ser considerado, também, ponderações nos erros das diferentes saídas.

do controlador resultando em uma atuação menos agressiva. O horizonte de controle é sempre menor que o horizonte de predição, como pode ser visto na figura 2.

$N_1$  e  $N_2$  definem o início e o fim do horizonte de predição, respectivamente. O início é especificado, geralmente, como o atraso da variável mais um, ou seja,  $N_1 = d + 1$ . Já o fim do horizonte pode ser ajustado para dar mais ou menos importância para o regime permanente.

## 2.5 Considerações finais

Neste capítulo foram apresentados os conceitos fundamentais de controle preditivo com foco nas ferramentas que foram utilizadas no estudo de caso do capítulo 4. Nesta implementação, será utilizada o cálculo da resposta livre pelo modelo não linear e a atualização da mesma de forma recursiva.

No próximo capítulo será discutido em mais detalhes dois estimadores com foco em sistemas não lineares: o filtro de Kalman e o estimador de horizonte deslizante (MHE).



## 3 Estimadores em controle preditivo

Neste capítulo, será apresentado uma revisão e comparação de duas técnicas de estimação visando o uso em sistema de controle preditivo. Como as plantas geralmente tem características não lineares, multivariáveis e com restrições, esses fatores foram levados em conta na avaliação dos algoritmos.

Uma das principais ferramentas utilizadas como observador de estados é o filtro de Kalman e suas variações como o Unscented Kalman Filter [6] e o Extended Kalman Filter [1]. Outra técnica bastante estudada na literatura para a estimação de estados, principalmente para sistemas não lineares, é a denominada Moving Horizon Estimation (estimador de horizonte deslizante) [7]. Nestes estudos, são mostradas as dificuldades na aplicação do filtro de Kalman e suas extensões em sistemas não lineares e as vantagens de usar o MHE nestes casos.

### 3.1 Estimadores

A determinação de variáveis de um sistema, tanto estados quanto parâmetros, é uma parte essencial em muitas estratégias de controle já que esse tipo de informação é utilizada (de forma implícita ou explícita) na representação interna da planta usada para a sintonia e/ou cálculo do controle. Porém, na prática, é comum não se ter acesso diretamente a todas as variáveis necessárias, dado que as mesmas podem ser imensuráveis ou o custo de instalação de um sensor ser muito alto para a aplicação particular.

Em outros casos, mesmo tendo sensores instalados em todos os estados do sistema que interessam para o controlador, o ruído presente nos sinais oriundos dos medidores podem tornar as medidas não apropriadas para seu uso direto nos algoritmos de controle. Assim, estimadores são necessários para a implementação eficiente da lei de controle. O problema geral de estimação se resume em [1]:

Dado as informações passadas de um sistema multivariável não linear (saídas, entradas e estimações passadas), estimar os estados e parâmetros atuais.

Para resolver este problema de estimação, o primeiro passo é definir o modelo que vai ser utilizado para representar a planta e isto será feito na seção seguinte.

### 3.1.1 Modelo não linear

Neste trabalho, assume-se que o sistema não linear pode ser representado pelo seguinte modelo discreto:

$$\begin{aligned}\mathbf{x}_{k+1} &= f(\mathbf{x}_k, \mathbf{p}_k, \mathbf{u}_k) + \boldsymbol{\omega}_k^{(1)} \\ \mathbf{y}_k &= g(\mathbf{x}_k, \mathbf{p}_k) + \mathbf{v}_k\end{aligned}\quad (3.1)$$

onde  $\mathbf{x}$  é o vetor de estados,  $\mathbf{u}$  é o vetor de entradas conhecido,  $\mathbf{p}$  o vetor de parâmetros desconhecidos,  $\boldsymbol{\omega}^{(1)}$  um vetor de erro nos estados,  $\mathbf{y}$  o vetor de saídas e  $\mathbf{v}$  um vetor de ruído na saída. De forma geral,  $\boldsymbol{\omega}^{(1)}$  e  $\mathbf{v}$  pode ser considerados como ruídos gaussianos de média zero e covariância nula.

Nesta formulação, os parâmetros do sistema são adicionados como estados usando o modelo discreto  $\mathbf{p}_{k+1} = \mathbf{p}_k + \boldsymbol{\omega}_k^{(2)}$ <sup>1</sup>, sendo  $\boldsymbol{\omega}_k^{(2)}$  uma sequência aleatória de valores com média nula. Assim, escrevendo as equações de estado no formato padrão, o sistema aumentado resulta em<sup>2</sup>:

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{p}_{k+1} \end{bmatrix} = \begin{bmatrix} f(\mathbf{x}_k, \mathbf{p}_k, \mathbf{u}_k) \\ \mathbf{p}_k \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}_k^{(1)} \\ \boldsymbol{\omega}_k^{(2)} \end{bmatrix}\quad (3.2)$$

## 3.2 Filtro de Kalman

O filtro de Kalman (FK) é um estimador de estados ótimo que minimiza a variância do erro de predição a um passo de um sistema linear representado pelo modelo:

$$\begin{aligned}\mathbf{x}_{k+1} &= A\mathbf{x}_k + B\mathbf{u}_k + \boldsymbol{\omega}_k \\ \mathbf{y}_k &= C\mathbf{x}_k + \mathbf{v}_k\end{aligned}\quad (3.3)$$

onde  $\mathbf{x}$  é o vetor de estados,  $\mathbf{u}$  é o vetor de entrada conhecido,  $\boldsymbol{\omega}$  um vetor de erros nas equações de estado,  $\mathbf{y}$  o vetor de medidas e  $\mathbf{v}$  um vetor de ruído na saída.  $\boldsymbol{\omega}$  e  $\mathbf{v}$  são ruídos de média zero e de variância definida por matrizes quadradas diagonais  $Q$  e  $R$ , respectivamente. As dimensões de  $Q$  e  $R$  dependem de  $\boldsymbol{\omega}$  e  $\mathbf{v}$ .

O FK é um algoritmo que estima os estados  $\mathbf{x}$  do modelo a partir das entradas  $\mathbf{u}$  e saídas  $\mathbf{y}$  do processo de forma recursiva que compreende dois passos: predição e atualização. No primeiro, o filtro estima o estado atual ( $\hat{\mathbf{x}}_{k|k-1}$ ) e sua incerteza ( $P_{k|k-1}$ ) utilizando o modelo da planta e a entrada passada. No próximo passo (atualização), a estimativa é comparada com a saída real ( $\mathbf{y}_k$ ) e os estados e sua incerteza são atualizados utilizando um ganho  $K_k$ <sup>3</sup>. Essa matriz de ganho age como uma média ponderada da saída dando

<sup>1</sup> Esse modelo é denominado *random walk process* [1].

<sup>2</sup> A formulação do parâmetro como estado do sistema foi necessária para comparação do MHE com o EKF. Ela não é obrigatória no MHE.

<sup>3</sup> O "steady state Kalman filter" pode reduzir o custo computacional encontrando um  $K$  e  $P$  constantes [8].

mais importância para as medições que possuem menos incerteza. De forma resumida, o algoritmo funciona da seguinte forma [8]:

1. Em  $k = 0$ , inicializa o estimador com  $\hat{\mathbf{x}}_{1|0}$  e  $P_{1|0}$ ;
2. Predição do estado atual utilizando  $\hat{\mathbf{x}}_{k|k-1} = A\hat{\mathbf{x}}_{k-1} + B\mathbf{u}_{k-1}$ ;
3. Estimativa de  $P$  pela equação  $P_k = AP_{k-1}A^T + Q$ ;
4. Cálculo do ganho através de  $K_k = P_kC^T(CP_kC^T + R)^{-1}$ ;
5. Cálculo do erro de predição utilizando a equação de saída do sistema:  $\mathbf{e}_k = \mathbf{y} - C\hat{\mathbf{x}}_{k|k-1}$ ;
6. Correção da predição por  $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k\mathbf{e}_k$ ;
7. Atualização da covariância do erro por  $P_k = P_k - K_kCP_k$ ;
8. Volta ao passo 2.

A atualização da matriz  $P_k$  depende de  $P_{k-1}$ , das equações do modelo e da matriz de covariância do ruído do estado  $Q$  e da saída  $R$ . Por causa disso, o filtro é recursivo e não precisa guardar todo o histórico de medição e atuação. A estimação é realizada utilizando somente as entradas e saídas atuais e a matriz  $P_k$  que traz a informação da incerteza da estimação, ou seja, as saídas passadas não são usadas explicitamente.

Para sistemas lineares, o filtro de Kalman é a solução mais utilizada na estimação de estados em presença de ruído. Porém, o desempenho do filtro depende de um conhecimento prévio dos processos estocásticos gaussianos que descrevem o comportamento dos estados, das perturbações e do ruído na saída. Como normalmente esse tipo de informação não está disponível ou ela varia com o tempo, a eficiência do observador pode ser prejudicada [1].

### 3.2.1 Exemplo ilustrativo

O filtro de Kalman é difícil de entender diretamente das equações mostradas mas há vários exemplos que ajudam a entender o problema. Aqui será utilizado caso de [9] utilizando o modelo da equação 3.3.

Na Figura 3, temos um sistema unidimensional simples que procura medir a posição de um carro com uma antena. Os estados do sistema são a posição e a velocidade do carro sendo que a posição é medida pela antena. Como entradas, temos o torque aplicado ao carro e ele é conhecido. A relação entre entradas e estados é conhecida, ou seja, as matrizes da equação 3.3 são dadas. O estado real do sistema não é conhecido e o filtro de Kalman realiza uma estimativa deles combinando o modelo do sistema com a medida ruidosa.

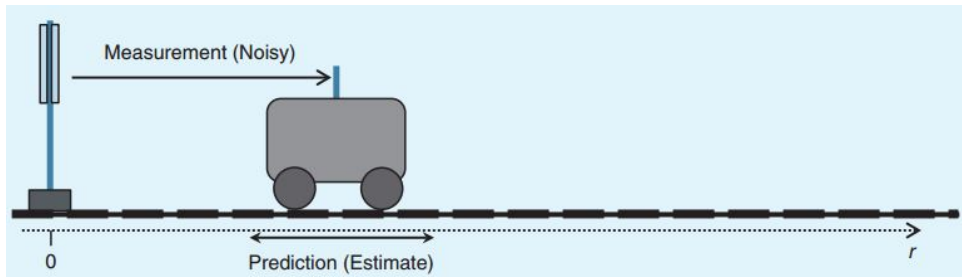


Figura 3 – Sistema unidimensional de um trem.

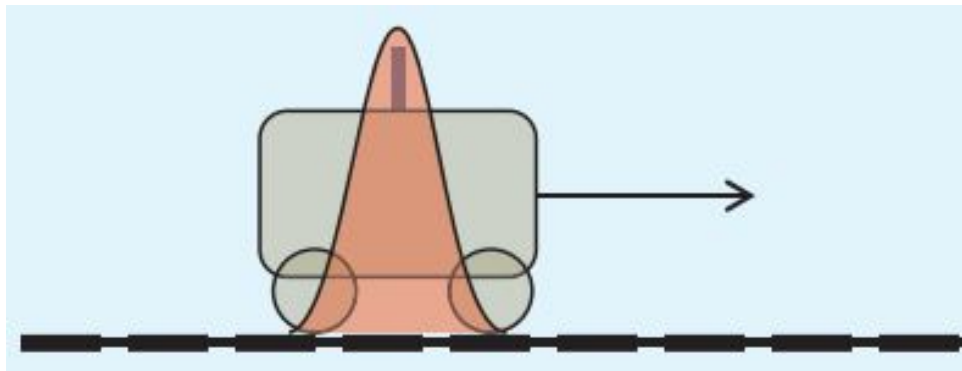


Figura 4 – Conhecimento inicial do sistema mostrando a distribuição da variável (posição).

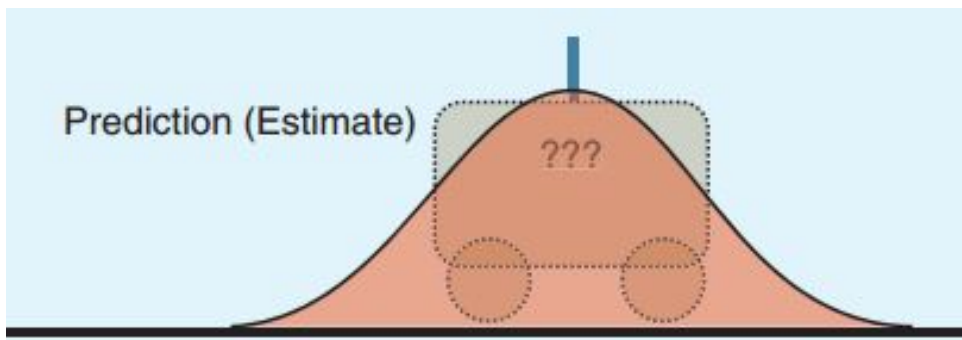


Figura 5 – Conhecimento após uma amostra do sistema, utilizando somente o modelo.

A primeira etapa, como citada anteriormente, é a predição e ela é realizada da seguinte forma. No instante inicial, temos conhecimento da posição e velocidade e estas variáveis possuem uma distribuição Gaussiana<sup>4</sup> em torno do valor nominal. Essa ideia pode ser visualizada na Figura 4. No próximo instante de amostragem, utilizando o modelo, temos uma nova estimativa da posição e velocidade, como pode ser mostrada na Figura 5. Note que agora a distribuição da variável, ou seja, a incerteza aumentou. Isso no terceiro passo do algoritmo da seção passada com a equação  $P_k = AP_{k-1}A^T + Q$ .

O próximo passo é a atualização que utiliza a medição neste período de amostragem. Esta medição que utiliza a antena também possui uma distribuição Gaussiana em torno do seu valor nominal, como pode ser visto na Figura 6. A melhor estimativa do estado real

<sup>4</sup> Aqui assume-se que o ruído é Gaussiano mas há formulações para outras distribuições.

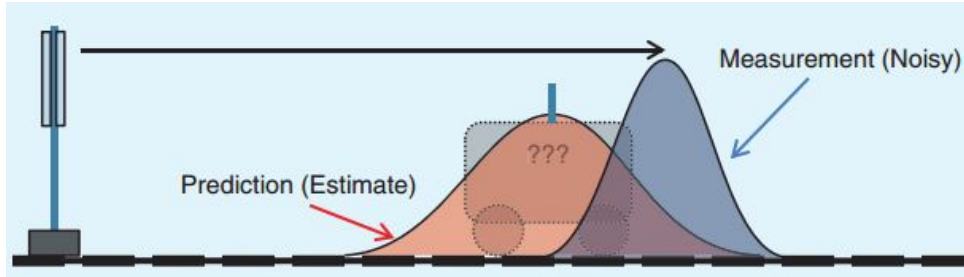


Figura 6 – Aqui é mostrado as duas estimativas: uma do modelo e outra do medidor.

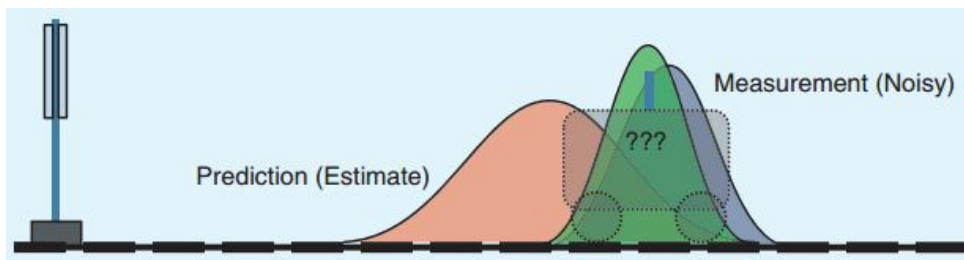


Figura 7 – Em verde, é mostrado a nova distribuição e a estimativa atualizada dos estados.

do sistema é a fusão das duas informações disponíveis, a do modelo e a do sensor. Assim, o filtro de Kalman busca combinar as duas distribuições gerando um novo valor médio e uma nova distribuição (Figura 7). O valor atualizado é dado pela equação  $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k \mathbf{e}_k$  e a nova incerteza calculada por  $P_k = P_k - K_k C P_k$ .

Para encontrar o ganho  $K_k$  e a matriz  $P_k$ , é utilizada uma interessante propriedade da função Gaussiana: o produto de duas funções Gaussianas é outra função do mesmo tipo. Com essa propriedade, chega-se no ganho  $K_k = P_k C^T (C P_k C^T + R)^{-1}$  e nas equações de atualização  $P_k = P_k - K_k C P_k$  e  $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k \mathbf{e}_k$ . Mais detalhes sobre isso pode ser encontrado em [9] ou em outras referências.

### 3.2.2 Filtro de Kalman Extendido

Como analisado no item anterior, o FK é definido para estimação em sistemas lineares. Para casos não lineares é possível realizar uma modificação do algoritmo, denominado Filtro de Kalman Extendido (FKE). O FKE se baseia numa aproximação linear de primeira ordem do sistema a cada período de amostragem e, então, o filtro de Kalman linear é aplicado. Observando as equações 3.1.1 e 3.2, é feita uma aproximação de  $A$  e  $C$  da seguinte forma:

$$\begin{aligned} A_k &= \left. \frac{\delta f}{\delta \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_{k-1}} \\ C_k &= \left. \frac{\delta g}{\delta \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k|k-1}} \end{aligned} \quad (3.4)$$

Assim, o passo-a-passo do algoritmo é dado por:

1. Em  $k = 0$ , inicializa o estimador com  $\hat{\mathbf{x}}_{1|0}$  e  $P_{1|0}$ ;
2. Predição do estado atual utilizando  $\hat{\mathbf{x}}_{k|k-1} = f(\mathbf{x}_{k-1}, \mathbf{p}_k, \mathbf{u}_{k-1})$ ;
3. Cálculo de  $A_k$  e  $C_k$  pela equação 3.4;
4. Estimativa de  $P$  pela equação  $P_k = A_k P_{k-1} A_k^T + Q$ ;
5. Cálculo do ganho através de  $K_k = P_k C_k^T (C_k P_k C_k^T + R)^{-1}$ ;
6. Cálculo do erro de predição utilizando a equação de saída do sistema:  $\mathbf{e}_k = \mathbf{y} - C_k \hat{\mathbf{x}}_{k|k-1}$ ;
7. Correção da predição por  $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k \mathbf{e}_k$ ;
8. Atualização da covariância do erro por  $P_k = P_k - K_k C_k P_k$ ;
9. Volta ao passo 2.

Essa extensão do FK apresenta algumas desvantagens. Como o filtro é recursivo, um pequeno erro de linearização pode se propagar e gerar uma convergência lenta e estimações com erro em regime permanente [1]. Isto pode levar performance sub-ótima e até a divergência da estimacão [6].

Além disso, o filtro de Kalman não permite incorporar conhecimento de regiões factíveis das variáveis desconhecidas, como é o caso de sistemas físicos reais, onde as variáveis controladas não podem assumir valores fora de uma faixa (limites inferior e superior predeterminados). Essas restrições poderia ser utilizada para evitar que o observador convirja para resultados incompatíveis com a realidade [10].

Em [10], se analisa o FKE mostrando algumas falhas do algoritmo para o caso específico de plantas químicas. É mostrado matematicamente que um sistema apresenta múltiplos estados que satisfazem as medidas de regime permanente, o FKE pode convergir para uma solução incorreta e até mesmo impossível de acordo com a planta. Isso pode ocorrer mesmo sem a presença de ruído e sem erros de modelagem da planta.

Em um exemplo de um reator químico, em [10] é mostrado que o FKE converge para concentrações negativas que são soluções das equações de regime permanente do sistema, mesmo não sendo fisicamente factíveis. O uso de restrições na formulação do estimador resolveria este problema, mas isto não pode ser incluído de forma simples no filtro de Kalman.

Visando contornar esses problemas do caso não linear e do uso das restrições, a técnica chamada Moving Horizon Estimation foi proposta [1, 10], descrita na próxima seção.

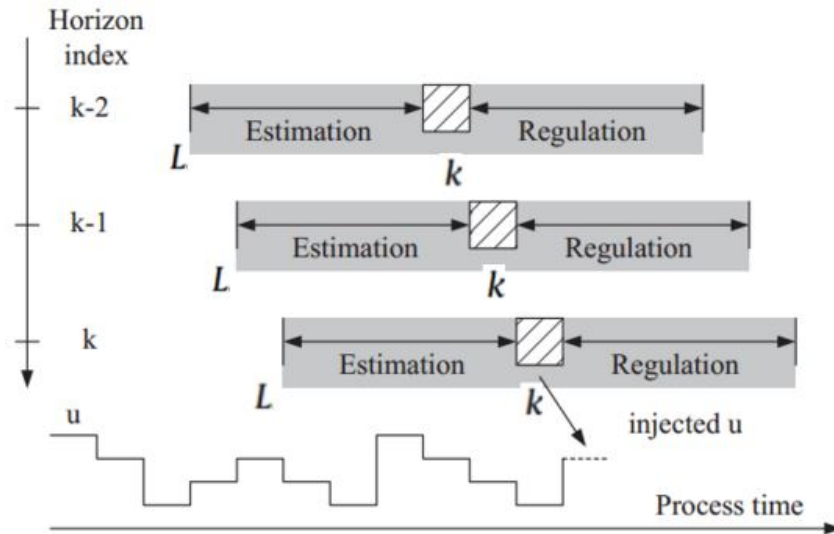


Figura 8 – Horizonte deslizante do MHE.

### 3.3 Moving Horizon Estimation

O MHE é um algoritmo que usa princípios de identificação caixa-cinza e controle preditivo para realizar a estimação de estados e parâmetros do sistema. Nesta técnica, utiliza-se uma otimização baseada em mínimos quadrados para ajustar as condições iniciais e os parâmetros do sistema de forma a alinhá-los com os valores medidos dentro de um certo horizonte.

A formulação do MHE é semelhante a do MPC: utiliza uma função custo a ser minimizada, um modelo interno dinâmico do sistema e restrições nas variáveis e parâmetros. Neste caso, porém, o horizonte é do tempo atual para atrás, as entradas e saídas passadas são conhecidas e os estados e parâmetros que são as incógnitas.

Outra semelhança é que, da mesma forma que no MPC, o horizonte é deslizante, como pode ser visto na figura 8. A cada iteração, o instante inicial  $L$  da janela de estimação é deslocado de uma amostra, acompanhando a evolução do tempo do processo. Somente as entradas e saídas do horizonte são utilizadas no algoritmo.

O MHE resolve alguns problemas do FK citados na seção 3.2, como a possibilidade de uso de restrições nas variáveis e o uso explícito do modelo não linear (sem necessidade de uma linearização a cada passo). Porém, o algoritmo possui uma maior complexidade computacional já que a cada período de amostragem é realizada uma otimização não linear como, por exemplo, o SQP. Isso pode levar a um maior tempo de processamento, principalmente com o uso de restrições [10] e, assim, necessitar de um período de amostragem maior.

### 3.3.1 Formulação do MHE

A formulação do Moving Horizon Estimation é baseada na minimização de uma função custo sujeito a restrições conforme mostrada abaixo:

$$\begin{aligned}
 \min_{\mathbf{x}_j, \mathbf{p}, \boldsymbol{\omega}_j} & \left( \sum_{j=L}^k \|\mathbf{y}_j - g(\mathbf{x}_j, \mathbf{p})\|_R^2 + \sum_{j=L}^{k-1} \|\boldsymbol{\omega}_j\|_Q^2 \right), \\
 \text{s.a.} & \quad \mathbf{x}_{j+1} = f(\mathbf{x}_j, \mathbf{p}, \mathbf{u}_j) + \boldsymbol{\omega}_j, \quad j = L, \dots, k-1 \\
 & \quad \left. \begin{aligned} & \mathbf{x}_{j,\min} \leq \mathbf{x}_j \leq \mathbf{x}_{j,\max}, \\ & \mathbf{w}_{j,\min} \leq \mathbf{w}_j \leq \mathbf{w}_{j,\max}, \\ & \mathbf{p}_{\min} \leq \mathbf{p} \leq \mathbf{p}_{\max} \end{aligned} \right\} \quad j = L, \dots, k
 \end{aligned} \tag{3.5}$$

onde  $\|\boldsymbol{\alpha}\|_A^2 = \boldsymbol{\alpha}^T A^T A \boldsymbol{\alpha}$ ,  $L$  é o índice do início do horizonte de estimação ( $L = k - M$ , sendo  $M$  o tamanho do horizonte) e o sistema dinâmico é o mesmo descrito na seção 3.1.1.

As matrizes positivas definidas  $R$  e  $Q$  são medidas quantitativas da confiança do modelo de saída e a dinâmica do sistema, respectivamente. Geralmente, elas são matrizes diagonais em que cada elemento é proporcional à raiz quadrada do módulo do valor esperado da variável correspondente. Ou seja, quanto menores os valores dos elementos da matriz  $R$  ou  $Q$ , maior a certeza na variável correspondente e menor a sua variação [1].

A função custo no formato de mínimos quadrados procura minimizar, respectivamente, dois aspectos do sistema: a diferença entre a saída real e estimada; e o erro  $\boldsymbol{\omega}$  na dinâmica dos estados. Pode-se entender o resultado dessa otimização como a sequência de estados (estimados) que melhor se adequa às observações do sistema.

Resumindo o algoritmo, a estimação é efetuada como se segue:

1. Deslizamento do horizonte (figura 8 como referência);
2. Adição das saídas  $\mathbf{y}_k$  e entradas  $\mathbf{u}_k$  atuais no horizonte de estimação;
3. Resolução do problema de otimização dado por 3.5 por algum método numérico;
4. Uso do resultado da estimação ( $\mathbf{x}_k$  e  $\mathbf{p}$ );
5. Volta ao passo 1.

#### 3.3.1.1 Arrival Cost

Outro aspecto que pode ser inserido na função custo do MHE é o *arrival cost* (custo de chegada). Semelhante ao custo terminal no controle preditivo, este termo agrega, através de  $\bar{\mathbf{x}}_L$  (estimação dos estados no início do horizonte),  $\bar{\mathbf{p}}_L$  (estimação dos parâmetros em  $L$ ) e  $P_L$  (matriz de covariância), informações fora do horizonte de estimação que começa em  $L$ . Da mesma forma que no controle, ele é essencial para garantir a estabilidade do estimador e melhorar sua performance.



Uma das formas de definir este custo de chegada é mediante um termo quadrático ponderado como se mostra na equação 3.6.

$$\left\| \begin{array}{c} \mathbf{x}_L - \bar{\mathbf{x}}_L \\ \mathbf{p} - \bar{\mathbf{p}}_L \end{array} \right\|_{P_L}^2 \quad (3.6)$$

Adicionando esse termo, o problema de otimização é dado por:

$$\begin{aligned} \min_{\mathbf{x}_j, \mathbf{p}, \boldsymbol{\omega}_j} & \left( \left\| \begin{array}{c} \mathbf{x}_L - \bar{\mathbf{x}}_L \\ \mathbf{p} - \bar{\mathbf{p}}_L \end{array} \right\| + \sum_{j=L}^k \|\mathbf{y}_j - g(\mathbf{x}_j, \mathbf{p})\|_{R_j}^2 + \sum_{j=L}^{k-1} \|\boldsymbol{\omega}_j\|_Q^2 \right), \\ \text{s.a.} & \left. \begin{array}{l} \mathbf{x}_{j+1} = f(\mathbf{x}_j, \mathbf{p}, \mathbf{u}_j) + \boldsymbol{\omega}_j, \quad j = L, \dots, k-1 \\ \mathbf{x}_{j,\min} \leq \mathbf{x}_j \leq \mathbf{x}_{j,\max}, \\ \mathbf{w}_{j,\min} \leq \mathbf{w}_j \leq \mathbf{w}_{j,\max}, \\ \mathbf{p}_{\min} \leq \mathbf{p} \leq \mathbf{p}_{\max} \end{array} \right\} \quad j = L, \dots, k \end{aligned} \quad (3.7)$$

Assim como no MPC, este termo pode ser descartado ao utilizar um horizonte de estimação suficientemente grande pois na equação 3.7 o arrival cost perde importância na função objetivo. Intuitivamente, pode-se fazer uma analogia com métodos de identificação em que pequenos horizontes carregam pouca informação do sistema o que leva à uma estimação ruim. Aumentando o horizonte causa uma diminuição desse problema.

O custo de chegada no MHE foi pouco explorado neste trabalho por sua complexidade. Porém, a partir dos artigos encontrados, percebeu-se sua importância na implementação de um algoritmo robusto e eficiente [10–12]. Existem diversas formas de calcular a ponderação do *arrival cost* sendo uma delas, inclusive, o filtro de Kalman estendido [1].

### 3.3.2 Aspectos de implementação

Como a formulação do MHE é semelhante ao do MPC, de forma geral, os mesmos métodos de otimização utilizados para o cálculo do controlador podem ser usados no estimador. Pode-se usar qualquer método de otimização não linear com restrições como, por exemplo, o Sequential Quadratic Programming (SQP) ou método do ponto interior. Uma boa revisão dos métodos de forma simples e didática pode ser encontrada em [13].

A implementação do MHE em tempo real tem sido estudada por vários autores [13], que propõem modificações no algoritmo para que ele possa ser utilizado em plantas rápidas. Por exemplo, em [11] o método denominado *direct multiple shooting* é usado para resolver as equações diferenciais de forma eficiente. Além disso, o máximo possível de informação é precalculada como, por exemplo, a Hessiana e o gradiente da função objetivo. No trabalho de Diehl e co-autores [12], é proposta uma atualização numérica eficiente do *arrival cost* baseada em uma fatorização QR.

Tabela 1 – Comparação entre MHE e EKF.

Característica	MHE	EKF
Restrições	Permite	Não permite
Convergência	Converge mais rápido	Infactível às vezes
Custo computacional	Bem mais alto	Baixo
Complexidade	Mais alto	Baixo

Estas modificações levam a um algoritmo mais complexo e a um maior tempo de desenvolvimento, porém, permite o uso do MHE com tempos de amostragem menores. Apesar de não ter sido explorado neste trabalho, percebeu-se a importância dessas modificações caso o MHE seja utilizado como estimador numa planta real.

### 3.4 Conclusão

O filtro de Kalman é um excelente método para estimação de estados, muito bem consolidado e estudado na academia. Porém, possui várias desvantagens para ser usado em sistemas não lineares, sendo as principais:

- Não permite o uso de restrições;
- Possível convergência para estados infactíveis;
- Sensibilidade a erros de modelagem;
- Resposta transiente pior que o MHE [1].

O Moving Horizon Estimation, apesar de não ser muito conhecido, é uma extensão direta da ideia do controle preditivo: utilizar uma janela deslizante e otimização para resolver um problema. Por causa disso, pode-se utilizar as mesmas ferramentas para implementar o algoritmo e ainda utilizar ideias de controle para analisar e sintonizar o estimador.

Uma das grandes vantagens do MHE é a possibilidade de usar qualquer tipo de restrições, o que permite evitar alguns problemas dos sistemas não lineares [10]. Claro que o uso destas implica em um aumento no considerável custo computacional tornando difícil a aplicação em plantas rápidas ( $T_s$  na ordem de segundos). Apesar disso, já foram citados alguns artigos que introduzem técnicas de tempo real e isto pode ser um campo de estudo importante caso o MHE seja usado numa solução de engenharia.

A tabela 1 mostra de forma resumida os principais aspectos a considerar na comparação entre o MHE e o FKE. Assim, neste projeto, com vistas ao uso dos estimadores em aplicações de controle preditivo não linear com restrições, optou-se pelo estudo específico da técnica Moving Horizon Estimation (MHE). No próximo capítulo, são mostrados os

resultados da implementação do MHE em uma planta química e, em seguida, a comparação com o FKE.



## 4 Estudos de caso: CSTR

Para estudar a implementação do MHE e analisar a sua performance, foi proposto, inicialmente, um estudo de caso bem conhecido da literatura. Primeiramente foi implementado o estimador na planta CSTR para comparação direta com o estimador de [1]. Em seguida, foi implementado um controle preditivo baseado no DMC que utiliza a estimação do MHE para o cálculo da atuação. Na última parte, uma comparação entre o MHE e o FKE foi efetuada.

### 4.1 Descrição da planta

A planta escolhida foi o *Continuous Stirred Tank Reactor* (CSTR), sistema de dois estados e com não-linearidades importantes, descrita em [14], utilizando os mesmos parâmetros do modelo definidos em [1]. O esquema do reator se mostra na figura 9.

Assume-se que o líquido é perfeitamente misturado e que nele ocorre uma reação exotérmica irreversível ( $A \rightarrow B$ ), ou seja, a medida que a reação acontece, a temperatura da solução aumenta. O fluxo que entra é o mesmo que sai do tanque de forma que o mesmo

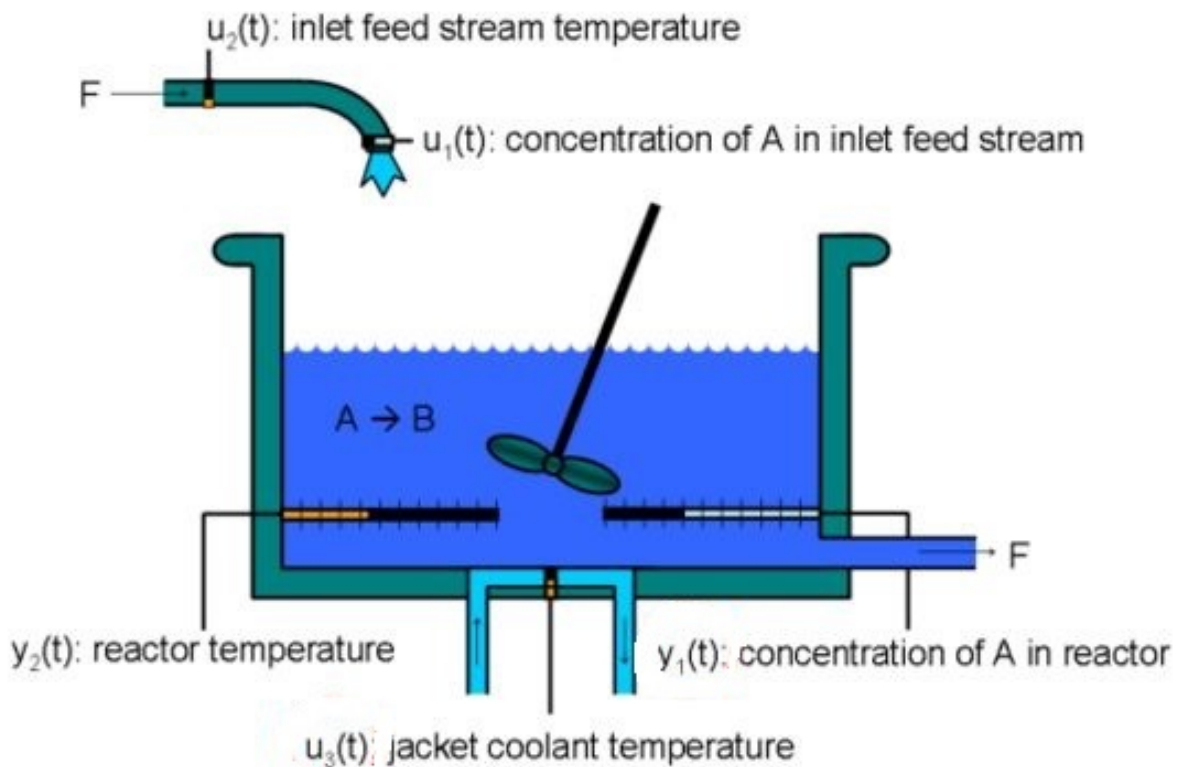


Figura 9 – Representação do CSTR.

possui volume constante. O sistema possui, também, uma camisa de refrigeração em volta do tanque para controlar a temperatura.

O reator possui três variáveis de entrada: concentração  $A$  de entrada ( $u_1$ ), temperatura de entrada ( $u_2$ ) e temperatura de resfriamento ( $u_3$ ). Assume-se que não se tem acesso às duas primeiras entradas e elas serão consideradas perturbações. Assim, a temperatura de resfriamento será utilizada para controlar a concentração de  $A$ .

O objetivo do processo é manipular a temperatura de resfriamento ( $u_3$ ) para controlar a concentração do reagente  $A$  que sai ( $y_1$ ) rejeitando as variações da concentração e temperatura que entram ( $u_1$  e  $u_2$ ). Ao mesmo tempo, é necessário manter a temperatura do reator ( $y_2$ ) em limites razoáveis (entre 150 e 200 graus Celsius).

### 4.1.1 Modelagem

As equações do sistema assim como os valores numéricos foram retirados de [1]. Os detalhes da modelagem podem ser encontrados no artigo de Jang [14]. Nesta etapa inicial, o objetivo é estimar alguns parâmetros desconhecidos do modelo e também realizar uma estimação dos estados. Neste último caso, a estimação dos estados se faz necessária porque se considera que as medidas dos mesmos (a saída) são muito afetadas pelo ruído.

Utilizando as equações físicas do reator e tendo como referência o modelo dinâmico explicado em 3.1.1, temos as seguintes equações de estado do sistema:

$$\begin{bmatrix} \dot{C} \\ \dot{T} \end{bmatrix} = \begin{bmatrix} \frac{q}{V}(C_0 - C) - kCe^{-\frac{E_a}{T}} \\ \frac{q}{V}(T_0 - T) - \frac{\Delta H}{\rho C_p} kCe^{-\frac{E_a}{T}} - \frac{UA}{\rho C_p V}(T - T_c) \end{bmatrix} \quad (4.1)$$

Todos os parâmetros terão seus valores conhecidos, com exceção dos dois últimos: Coeficiente de transferência térmica  $U$  e energia de ativação  $E_a$ . Para estes dois, será utilizado o MHE para estimação. Um resumo de todos os parâmetros pode ser encontrado na tabela 2.

Para a discretização do sistema utilizado tanto no controle quanto na simulação, será realizada uma aproximação *forward* da derivada dada por:

$$\dot{x} = f(x) \implies \dot{x}_{k-1} = f(x_{k-1}) \implies \frac{x_k - x_{k-1}}{T_s} = f(x_{k-1}) \implies x_k = x_{k-1} + T_s f(x_{k-1})$$

sendo  $T_s$  o tempo de amostragem em segundos. Assim, as equações de estado do sistema

Tabela 2 – Estados, entradas e parâmetros do CSTR.

Tipo	Abreviação	Descrição	Unidade	Valor inicial
Estado/Saída	$C$	Concentração de A no reator	$\frac{gmol}{cm^3}$	$3.531 \times 10^{-7}$
Estado/Saída	$T$	Temperatura do reator	$K$	440.9
Perturbação	$C_0$	Concentração de A na entrada	$\frac{gmol}{cm^3}$	$6.5 \times 10^{-6}$
Perturbação	$T_0$	Temperatura do fluido de entrada	$K$	350
Entrada	$T_c$	Temperatura de resfriamento	$K$	340
Parâmetro	$q$	Vazão volumétrica	$\frac{cm^3}{s}$	10
Parâmetro	$V$	Volume do reator	$cm^3$	1000
Parâmetro	$k$	Fator pré-exponencial não-térmico	$\frac{1}{s}$	$7.86 \times 10^{12}$
Parâmetro	$\Delta H$	Calor da reação	$\frac{cal}{gmol}$	-27000
Parâmetro	$A$	Área do tanque	$cm^2$	10
Parâmetro	$\rho$	Densidade	$\frac{g}{cm^3}$	0.001
Parâmetro	$C_p$	Capacidade térmica	$\frac{cal}{gK}$	1
Parâmetro	$U$	Coefficiente de transferência térmica	$\frac{cal}{cm^2 s K}$	$5 \times 10^{-4}$
Parâmetro	$E_a$	Energia de ativação	$K$	14090

Tabela 3 – Características dos estados e parâmetros desconhecidos do CSTR.

Variável	Estimação	Valor real	Unidade	Covariância do ruído
$C$	$3.753e - 7$	$3.531e - 7$	$\frac{gmol}{cm^3}$	$10^{-7}$
$T$	446.5	440.9	$K$	$10^{-7}$
$U$	$4.76e - 4$	$5e - 4$	$\frac{cal}{cm^2 s K}$	0
$E_a$	13377	14090	$K$	0

discretizado são dadas por<sup>1</sup>:

$$\begin{bmatrix} C_k \\ T_k \\ U_k \\ (E_a)_k \end{bmatrix} = \begin{bmatrix} C_{k-1} + T_s \left( \frac{q}{V} (C_0 - C_{k-1}) - k C_{k-1} e^{\frac{-E_a}{T_{k-1}}} \right) \\ T_{k-1} + T_s \left( \frac{q}{V} (T_0 - T_{k-1}) - \frac{\Delta H}{\rho C_p} k C_{k-1} e^{\frac{-E_a}{T_{k-1}}} - \frac{U A}{\rho C_p V} (T_{k-1} - T_c) \right) \\ U_{k-1} \\ (E_a)_{k-1} \end{bmatrix} \quad (4.2)$$

A tabela 3 mostra a comparação entre os estados iniciais reais e estimados. Para simulação, considera-se também um ruído aditivo nos estados de média nula com a covariância mostrada na tabela.

As saídas são os dois primeiros estados do sistema, ou seja,  $C$  e  $T$ . Além do ruído nos estados, há também o ruído aditivo (conforme o modelo da equação 3.2) diretamente

<sup>1</sup> A notação  $k - 1$  foi omitida em alguns termos do lado direito da equação para facilitar a leitura.

nas saídas com a covariância dada pela matriz  $R$ . Vale ressaltar que o ruído na saída é maior que o dos estados.

$$R = \begin{bmatrix} 0.005 & 0 \\ 0 & 5 \times 10^{-5} \end{bmatrix} \quad (4.3)$$

## 4.2 Implementação do MHE

O algoritmo MHE foi implementado com a toolbox de identificação caixa-cinza do MATLAB utilizando o SQP com a aproximação<sup>2</sup> de Gauss-Newton para o cálculo da Hessiana. Assim, a cada período de amostragem, a toolbox resolve o problema de otimização especificado na seção 3.3.1 utilizando o modelo não linear do sistema e o horizonte de estimação composto pelas  $M$  saídas e entradas passadas. O resultado disso é uma estimação do estado atual do CSTR e dos parâmetros  $U$  e  $E_a$ .

Neste ensaio, todos os valores reais dos parâmetros e entradas do sistema são constantes conforme a tabela 2. Ou seja, o sistema se comporta livremente a partir dos estados iniciais especificados na tabela 3. Procura-se observar, assim, a evolução da estimação dos estados e parâmetros da planta no tempo. Além disso, nenhum tipo de restrição foi considerada no algoritmo de estimação.

Abaixo, apresenta-se passo-a-passo o pseudocódigo utilizado para testar o estimação por MHE.

1. Simula a planta por um período de amostragem utilizando a aproximação forward da derivada gerando  $\mathbf{x}$ ;
2. Adiciona o ruído nos estados  $\mathbf{x}$  de acordo com a tabela 3;
3. Adiciona o ruído de  $R$  em  $\mathbf{x}$  gerando a saída  $\mathbf{y}$ ;
4. As últimas  $M$  amostras das saídas e entradas são coletadas<sup>3</sup>;
5. O problema de otimização do MHE (equação 3.5) é resolvido pelo toolbox do MATLAB utilizando as amostras do horizonte. É gerado, assim, a estimação dos estados e parâmetros;
6. Espera o período de amostragem e volta para o passo 1.

Em termos de implementação, o MHE proposto tem algumas diferenças com o estudado em [1]. A tabela 4 mostra estas diferenças de implementação.

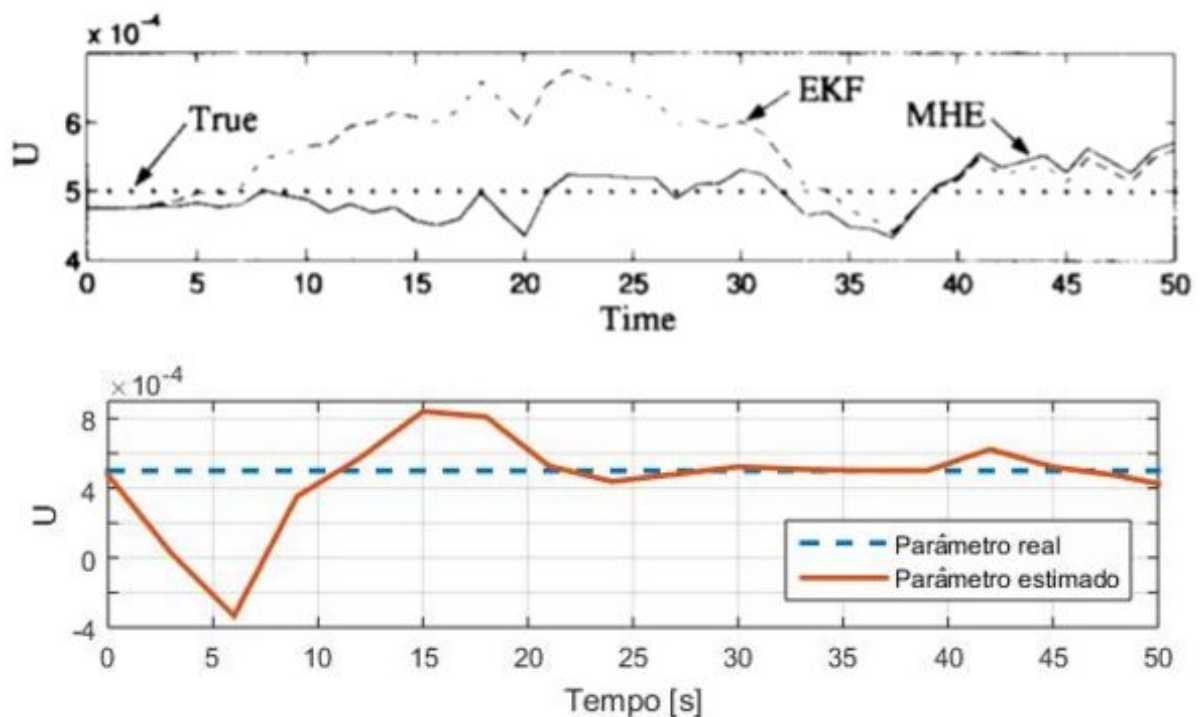
<sup>2</sup>  $H \approx J^T J$ . O uso dessa aproximação no MHE pode ser encontrada em [12].

<sup>3</sup> Se  $t < M$ , como acontece no início da simulação, são coletadas as informações que estiverem disponíveis.



Tabela 4 – Parâmetros de implementação do MHE

Característica	Implementado	Artigo
Tempo de amostragem	3s	1s
Horizonte de estimação	15 amostras	2 amostras
Cálculo do Arrival cost	Não	Sim, usando EKF
Ruído nos parâmetros	Não	Sim
Ruído nos estados	Sim	Sim
Ruído na saída	Sim	Sim

Figura 10 – Estimação do parâmetro  $U$ . Resultado de [1] no topo e implementação abaixo.

Os resultados comparativos entre o MHE do artigo e o implementado neste trabalho são mostrados a seguir.

#### 4.2.1 Resultados

O artigo [1] tem como objetivo mostrar que o MHE com um horizonte de duas amostras já traz uma melhora na resposta transiente do observador de estados, comparado com o filtro de Kalman estendido puro, como pode ser visto na Figura 10. Porém, ele utiliza um filtro de Kalman para estimar o *arrival cost* do algoritmo o que não é implementado neste estudo.

A estimação de  $U$  (Figura 10) teve uma resposta inicial muito ruim apresentando, inclusive, valores negativos em duas amostras. Isto ocorre pois nesta implementação,

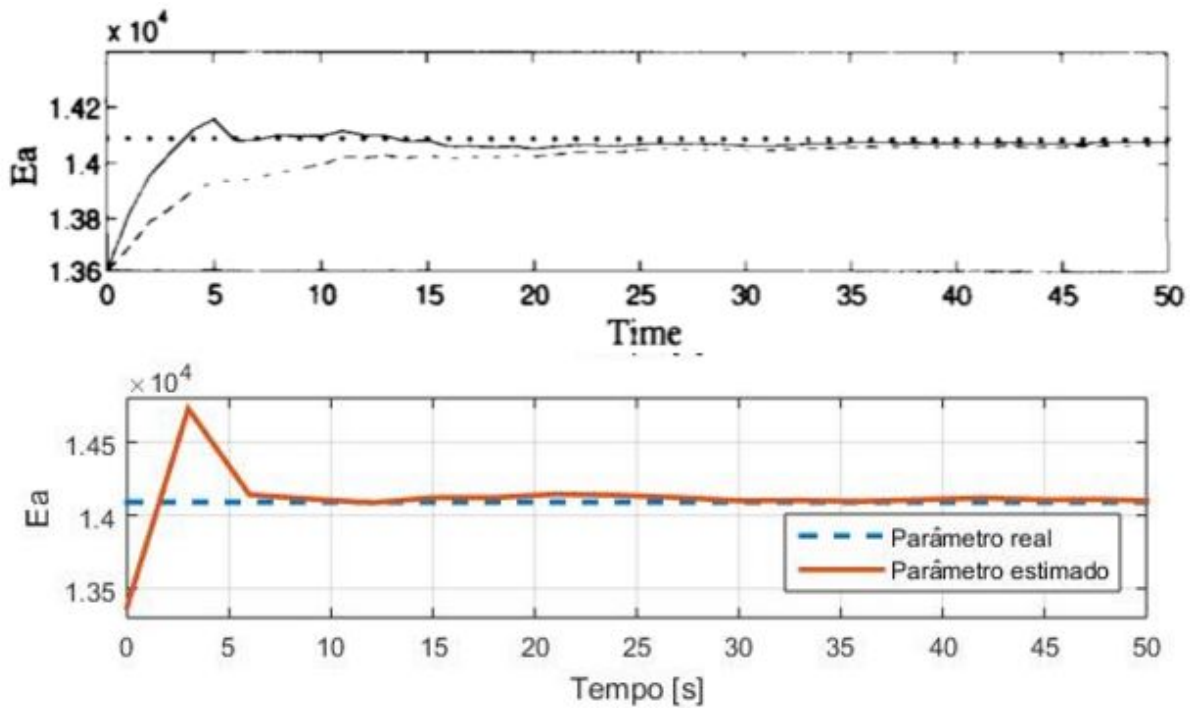


Figura 11 – Estimação do parâmetro  $E_a$ . Resultado de [1] no topo e implementação abaixo.

tomou-se o cuidado de aumentar o horizonte de predição de 1 até 15 nos momentos iniciais o que causou uma estimação ruim neste período. O sistema implementado em [1] não teve esse problema pois o horizonte é significativamente menor: duas amostras. Porém, após 20 segundos a estimação já converge para o valor real. Não houve problemas significativos na estimação de  $E_a$  (Figura 11) .

A estimação dos dois estados apresentaram bons resultados, como pode ser observado nas figuras 12 e 13. Percebe-se que mesmo com ruído na saída significativo (em azul), o algoritmo converge bem para o valor real após três amostras (9 segundos).

O tempo de processamento médio entre o período de amostragem foi de 0.4 segundos no caso sem restrições e de 2 a 4 segundos com restrições para o caso com horizonte de 15 amostras

Testes foram realizados variando o tamanho do horizonte de estimação. Para o caso sem restrições, um aumento no horizonte leva a melhores estimações dos estados e parâmetros sem uma diferença computacional significativa. Isso muda bastante, porém, quando se considera restrições nas variáveis. Nesta situação, o aumento do horizonte leva a uma maior complexidade no problema de otimização causando um tempo de processamento maior que o tempo de amostragem. Seria necessário, então, utilizar algoritmos de otimização mais eficientes para resolver o problema do MHE.

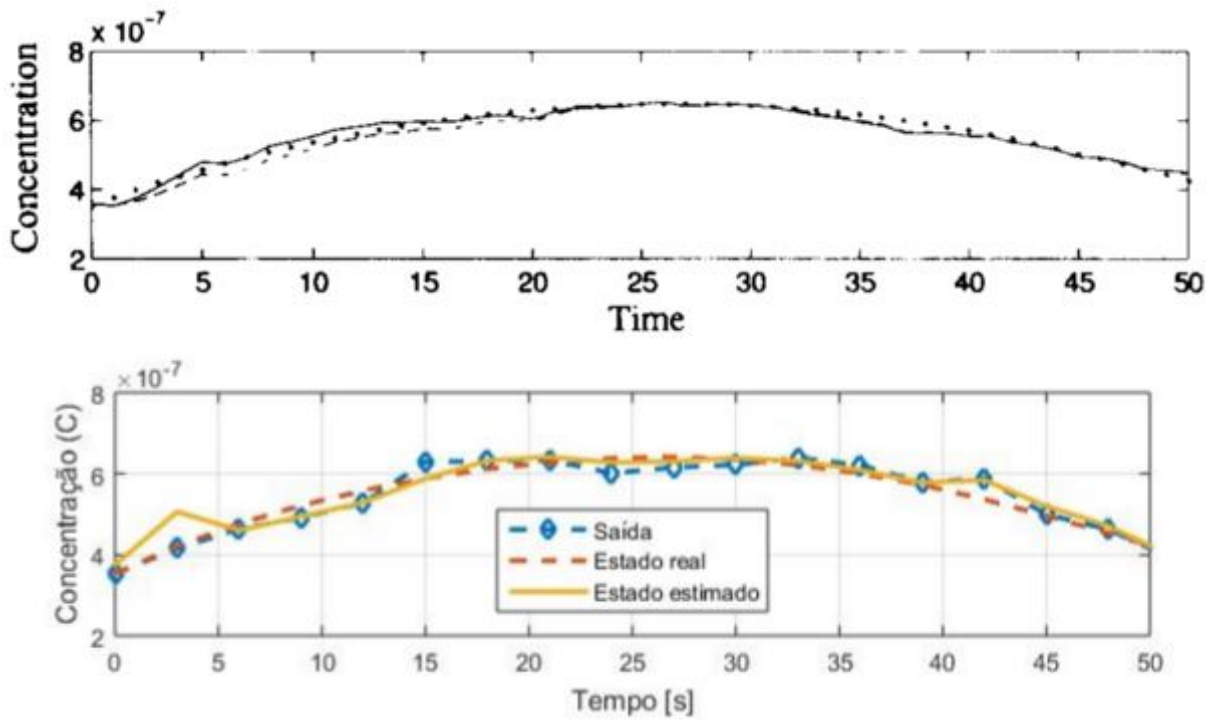


Figura 12 – Estimação da concentração  $C$ . Resultado de [1] no topo e implementação abaixo.

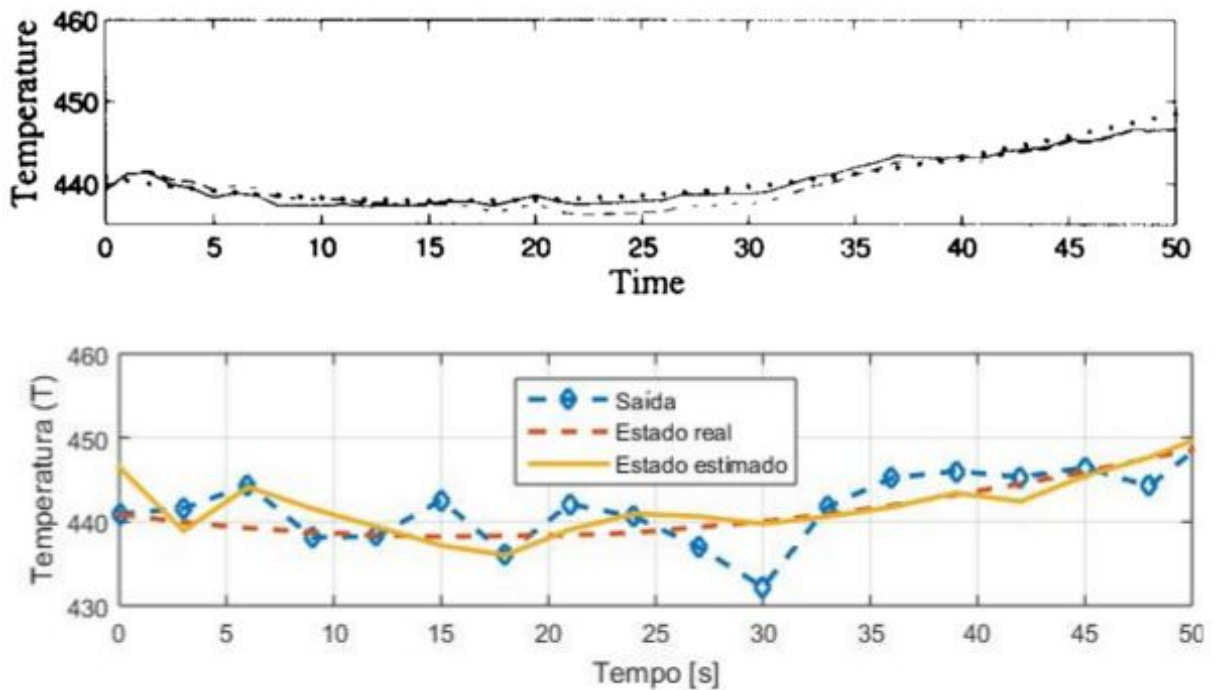


Figura 13 – Estimação da temperatura  $T$ . Resultado de [1] no topo e implementação abaixo.

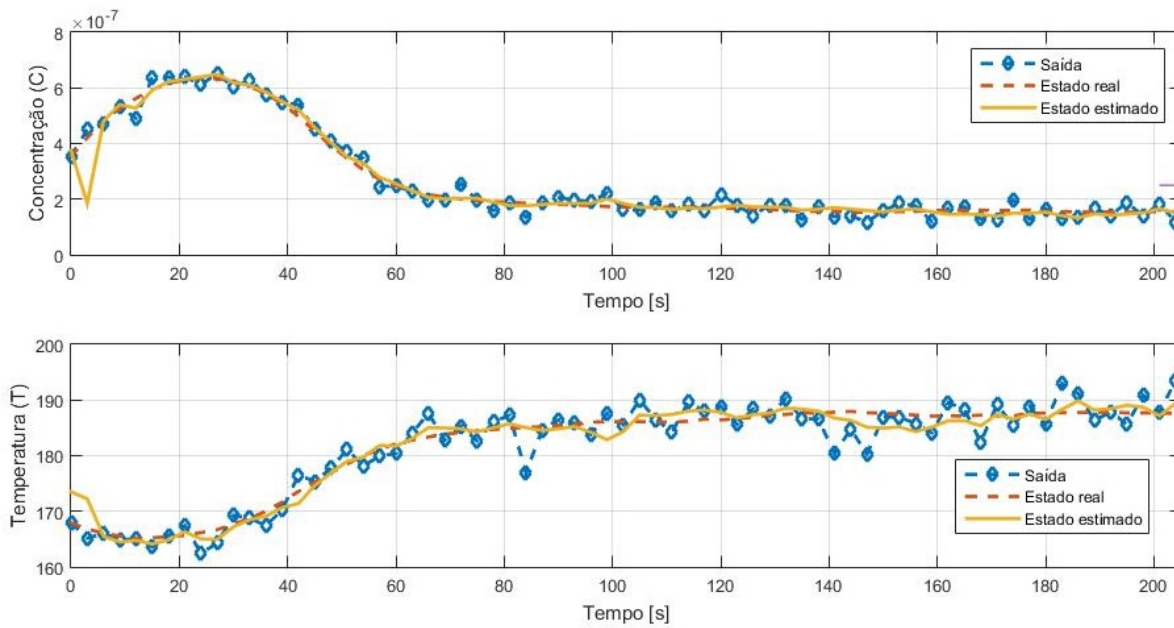


Figura 14 – Simulação do sistema em malha aberta. Temperatura em graus Celsius.

### 4.3 Implementação do controle

O objetivo do observador de estados é gerar uma estimativa para ser usada no cálculo da ação de controle. Por isso, foi implementado um MPC SISO juntamente com o MHE para verificar se o sistema completo apresenta bons resultados.

Como citado anteriormente, somente a temperatura de resfriamento ( $u_3$ ) será utilizada para controlar a concentração pois assume-se que as outras entradas estão fora do alcance do controlador sendo consideradas como perturbações. A concentração ( $u_1$ ) e temperatura de entrada ( $u_2$ ) são mantidas constantes durante toda a simulação e a temperatura do reator ( $T$ ) é mantida livre (sem controle). O estado inicial do sistema é o mesmo mostrado na tabelas 2 e 3.

A simulação do sistema é dividida em duas partes. Na primeira, é feita um ensaio de malha aberta do sistema até chegar em regime permanente e, em seguida, é introduzido o MPC.

#### 4.3.1 Malha aberta

Nesta primeira parte, são utilizados todos os parâmetros especificados na seção anterior sendo uma direta extensão daquela simulação. O principal objetivo deste ensaio foi observar o tempo de acomodação do sistema para a sintonia e esperar o regime permanente para acionar o controlador.

O resultado da simulação pode ser visto na figura 14. A concentração ( $C$ ) apresentou um tempo de 5% de 70s e um tempo de 2% de 120s, aproximadamente. A temperatura

do reator ( $T$ ) apresentou resultados semelhantes. Aos 200 segundos, o controlador é configurado e acionado conforme os parâmetros abaixo abaixo.

### 4.3.2 Controlador e Sintonia

O controlador implementado foi o DMC recursivo com o cálculo da resposta livre pelo modelo não linear.

#### 4.3.2.1 Predição da resposta livre

Inicialmente, em  $t = 200$ , a resposta livre  $\mathbf{Y}_{livre}$  é encontrada realizando uma simulação do modelo não linear discreto da planta com as entradas constantes. Como estados iniciais da simulação, é utilizado a estimativa efetuada aos 200 segundos. Assim,  $\mathbf{Y}_{livre}$  é semelhante à continuação da simulação mostrada na figura 14.

Nas amostras seguintes, a resposta livre é atualizada recursivamente conforme a seção 2.2.6.1 utilizando a estimativa da concentração ( $\hat{x}_1$ ) para a correção da predição, conforme a equação abaixo:

$$\mathbf{f}_r = \begin{bmatrix} y_o(t+1|t) \\ \vdots \\ y_o(t+N|t) \end{bmatrix} + \mathbf{1}_N (\hat{x}_1(t) - y_o(t|t)) \quad (4.4)$$

#### 4.3.2.2 Cálculo da ação de controle

A partir do ponto de operação mostrado na figura 14, utilizou-se o modelo não linear discreto da planta (equação 4.2) para levantar a resposta ao degrau utilizado no DMC. Foi aplicado um degrau unitário na temperatura de resfriamento e observado a resposta da concentração. Com essa informação foi construída a matriz  $G$  como especificado na seção 2.2.6.

A partir da inicialização do controlador, o modelo interno não foi mais modificado, ou seja, a matriz  $G$  não é recalculada e assume-se que o sistema se comporta de forma linear. Com certeza essa não é uma aproximação razoável da planta e um algoritmo de controle não linear traria, a princípio, melhores resultados.

#### 4.3.2.3 Sintonia

Para a sintonia, optou-se por aumentar<sup>4</sup> a ponderação do controle ( $\lambda$ ) para ter uma ação mais conservadora de forma a reagir menos às estimativas ruidosas. O horizonte de predição e de controle foram escolhidos observando os resultados de malha aberta da seção anterior. A tabela 5 resume os valores escolhidos.

<sup>4</sup> Ajuste feito empiricamente.

Tabela 5 – Sintonia do controlador. Horizontes em amostras e em segundos.

Parâmetro	Valor
$\lambda$	5
$N_2$	20 amostras (60s)
$N_u$	10 amostras (30s)

### 4.3.3 Resultados

Nesta segunda parte da simulação, foi testado o conjunto de estimador mais controlador no seguimento de referência. Não houve mudança no valor real dos parâmetros nem nas duas perturbações ( $u_1$  e  $u_2$ ).

Aos 200 segundos, é configurado e acionado o controlador para o seguimento de uma referência de  $2.5 \times 10^{-7}$  na concentração ( $C$ ). Observando a figura 15, o sistema estabiliza com um erro de 5% depois de aproximadamente 100s, apesar de uma pequena oscilação em regime permanente. Essa oscilação ocorre principalmente devido ao erro de estimação causado pela saída ruidosa de  $C$ .

Aos 400 segundos, a referência é diminuída para  $1 \times 10^{-7}$ . O controlador reage imediatamente atuando na temperatura de resfriamento e a concentração desce para o valor desejado, sem oscilações perceptíveis dessa vez. A temperatura ao longo da simulação não é controlada mas ela não varia muito permanecendo na faixa de 185 graus.

A estimação dos parâmetros se mantém estável em torno do valor real nesta parte da simulação. Nota-se na figura 16 que mudanças na referência ou no controle não afetam a estimação já que as informações anteriores de saída e entrada são utilizadas no MHE.

## 4.4 Implementação do FKE

O filtro de Kalman estendido foi implementado conforme o algoritmo especificado na seção 3.2.2 para comparação com o MHE. Todas as características da planta foram mantidas e o modelo discreto com a aproximação forward foi utilizado para calcular a estimação "a priori"  $\hat{\mathbf{x}}_{k|k-1}$ . Como estimativa inicial para o filtro de Kalman, foram utilizados os valores da tabela 3. A estimação inicial de  $P$  é dada por:

$$P_{1|0} = \text{diag}[ 0.01 \quad 0.005 \quad 0.02 \quad 0.02 ]$$

Para facilitar a comparação entre os dois estimadores, foi calculado o erro normalizado em relação à variável real. Assim, o erro normalizado é calculado pela equação 4.5,

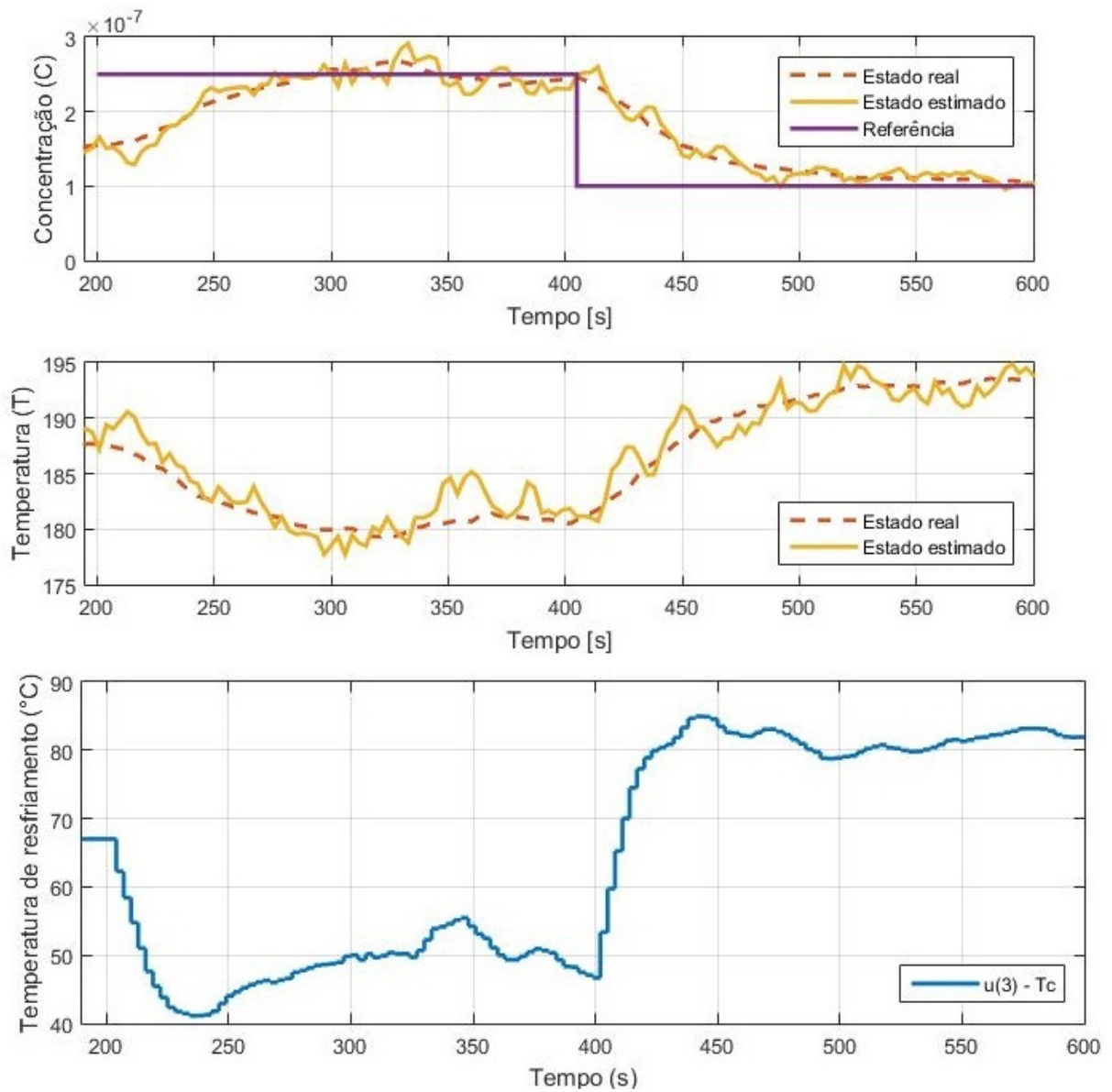


Figura 15 – Evolução dos estados do sistema em malha fechada. Temperatura em graus Celsius.

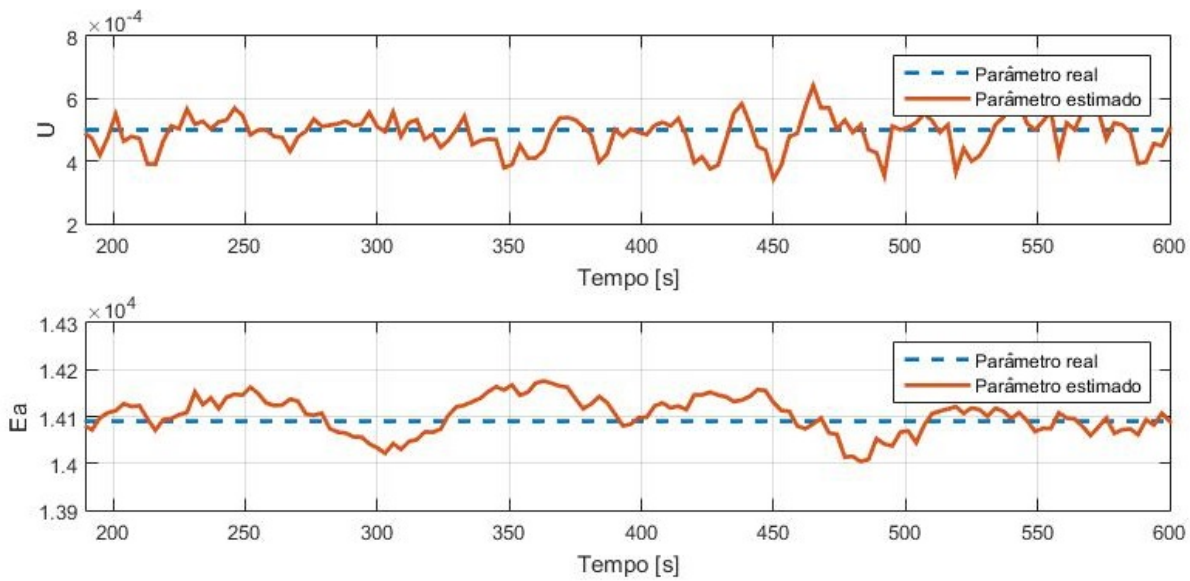


Figura 16 – Estimação dos parâmetros em malha fechada.

sendo  $x_k$  o valor real e  $\hat{x}_k$  a estimação.

$$\text{erro normalizado} = \left\| \frac{\text{valor real} - \text{valor estimado}}{\text{valor real}} \right\| = \left\| \frac{x_k - \hat{x}_k}{x_k} \right\| \quad (4.5)$$

A planta com o FKE como estimador foi simulado em MATLAB durante 200 segundos. Os resultados deste ensaio e a comparação com o MHE implementado são mostrados na próxima seção.

#### 4.4.1 Resultados

Na figura 17, podem ser vistas as estimações dos estados pelo FKE. Pelo gráfico superior, não há diferença perceptível estimadores. Avaliando o erro normalizado, porém, percebe-se que o MHE apresenta um desempenho ligeiramente melhor. Na estimação da temperatura, o FKE e o MHE possuem desempenho igualmente bons como pode ser visto na figura 18.

A figura 19 mostra a estimação de  $U$  neste ensaio. Para este parâmetro, o MHE produz estimações piores que o FKE nos instantes iniciais mas, após 25 segundos, os dois apresentam comportamentos semelhante. Não houve diferenças significativas entre os estimadores para o parâmetro  $E_a$  (figura 20).

### 4.5 Conclusão

Neste capítulo foi implementado em MATLAB um MHE básico para direta comparação com um artigo encontrado na literatura. A primeira parte do estudo de caso tinha



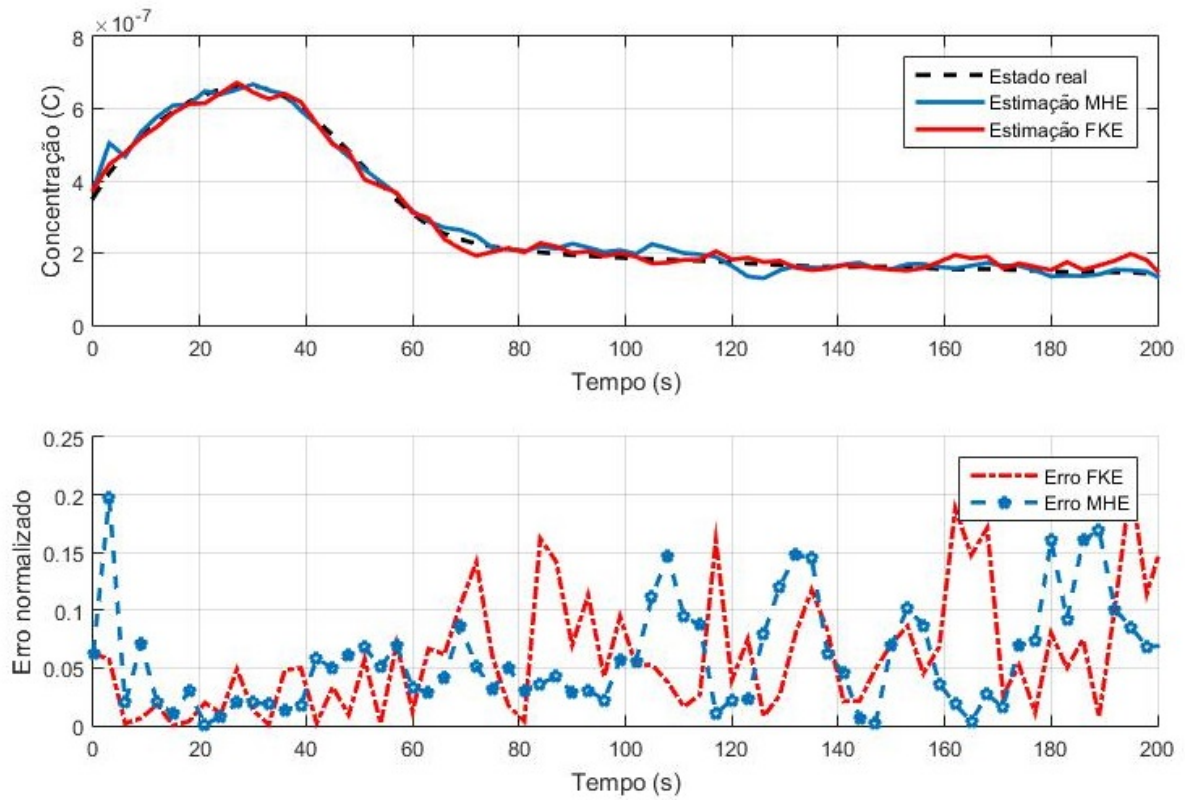


Figura 17 – Comparação entre MHE e o FKE na estimação da concentração (C).

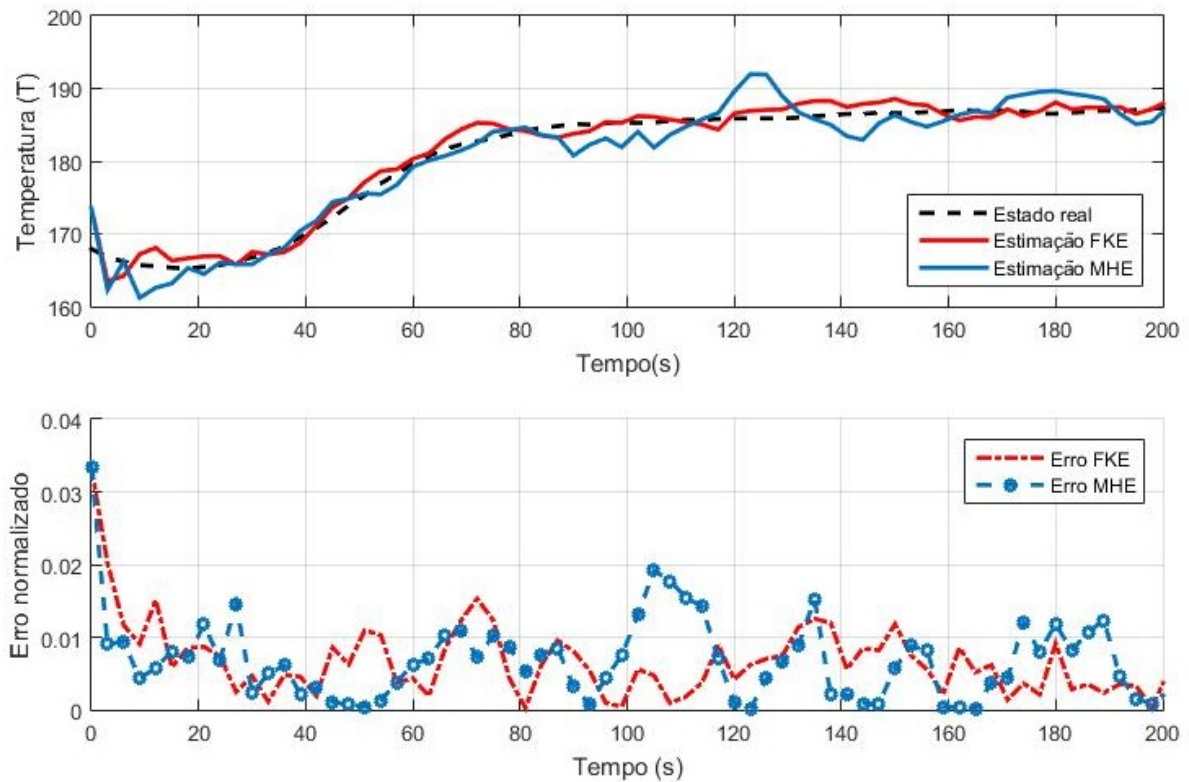


Figura 18 – Comparação entre MHE e o FKE na estimação da temperatura (T).

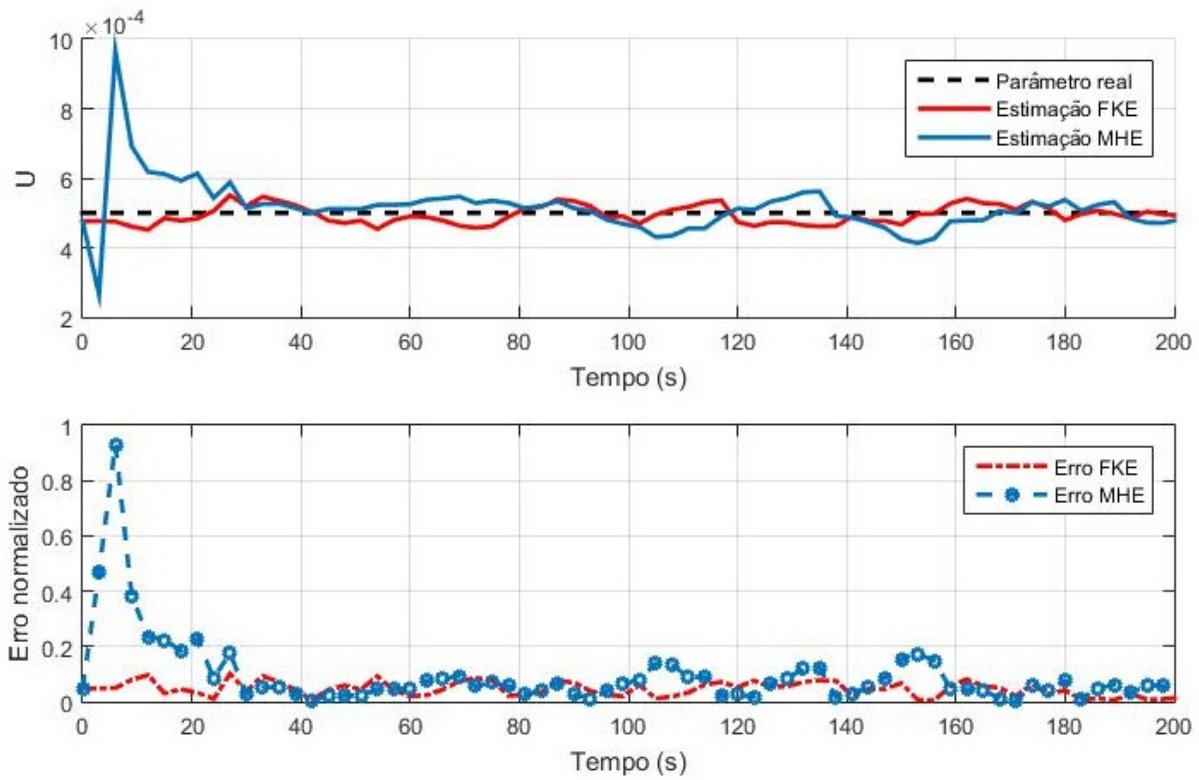


Figura 19 – Comparação entre MHE e o FKE para na estimação do parâmetro  $U$ .

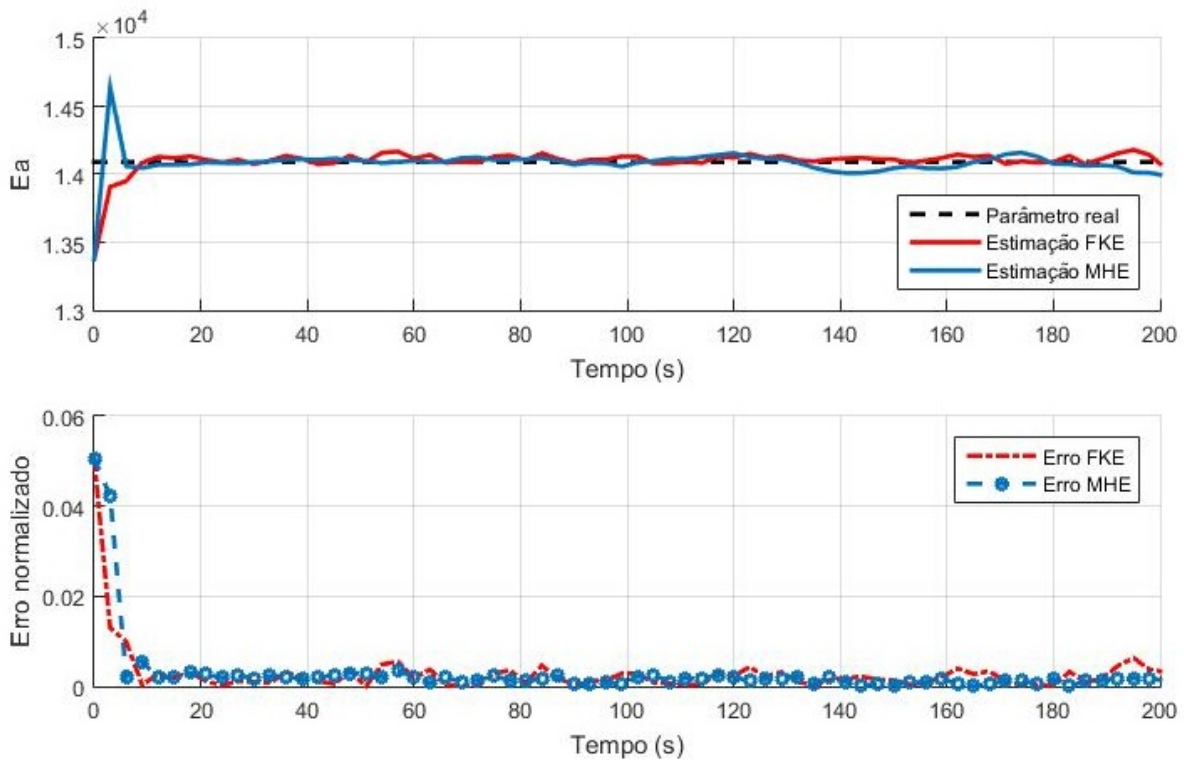


Figura 20 – Comparação entre MHE e o FKE na estimação do parâmetro  $E_a$ .

como objetivo garantir o bom funcionamento do estimador. Em seguida, um MPC foi projetado e utilizado junto com o MHE para aproximar mais de um caso real. O conjunto controlador mais observador produziu resultados satisfatórios mesmo na presença de forte ruído na saída.

Em seguida, foi implementado o filtro de Kalman estendido para comparação com o MHE implementado. Os dois estimadores apresentaram resultados semelhantes, com exceção dos instantes iniciais em que o horizonte de predição do MHE era demasiado pequeno. Uma forma de contornar este problema é acionar o MHE com o sistema em regime permanente. Assim, os estados em  $t < 0$  podem ser considerados iguais ao inicial ( $t = 1$ ).



## 5 Conclusões e Perspectivas

Durante a pesquisa bibliográfica, foram encontrados vários artigos comentando sobre desvantagens de usar o filtro de Kalman em plantas não lineares. Como esse tipo de modelo é importante na área de petróleo e gás e para controle preditivo, outro estimador se faz necessário. Assim, o MHE foi escolhido como objeto de estudo para a solução do problema de estimação.

O MHE é um algoritmo de fácil entendimento, principalmente para quem tem conhecimento em MPC, só que apresenta desafios na implementação. Assim como no MPC, suas principais vantagens aparecem quando a planta é não linear e com restrições e esta situação exige complexos pacotes de otimização o que aumenta significativamente o custo computacional.

Para testar o estimador escolhido, foi realizado um estudo de caso com uma planta química bem comum de ser encontrada na literatura: o CSTR. Em MATLAB, foi implementado o MHE para estimar dois estados e dois parâmetros e, em seguida, um MPC não linear recursivo foi projetado. Mesmo com ruídos significativos e erros de estimação dos estados, o conjunto MHE mais DMC produziu bons resultados.

### 5.1 Trabalhos futuros

Há vários pontos que faltam ser investigados neste assunto. Sobre o MHE, falta investigar, implementar e testar algoritmos mais "baixo-nível" para tentar melhorar a velocidade computacional. Para facilitar o desenvolvimento, foi utilizada a biblioteca de identificação do MATLAB mas este pacote não permite muitos ajustes na otimização. Além disso, pode-se procurar soluções em tempo real para o MHE como citado anteriormente.

Outro tópico importante é o *arrival cost*. Além dele ser importante para a estabilidade do algoritmo, ele promete convergências mais rápidas e horizontes menores [1]. O tamanho do horizonte é especialmente importante quando se considera restrições no sistema pois ele deixa o problema de otimização muito mais complexo. Portanto, a implementação do arrival cost levaria a estimações rápidas e uso de restrições sem um custo computacional elevado.



# Referências

- 1 ROBERTSON, D. G.; LEE, J. H. A least squares formulation for state estimation. *Journal of process control*, Elsevier, v. 5, n. 4, p. 291–299, 1995. Citado 15 vezes nas páginas 9, 31, 32, 33, 36, 38, 39, 40, 43, 44, 46, 47, 48, 49 e 59.
- 2 CUTLER, C. R.; RAMAKER, B. L. Dynamic matrix control - a computer control algorithm. n. 17, p. 72, 1980. Citado na página 21.
- 3 CAMACHO, E. F.; ALBA, C. B. *Model predictive control*. [S.l.]: Springer Science & Business Media, 1999. Citado 4 vezes nas páginas 21, 24, 27 e 28.
- 4 LIMA, D. M. *Predictor-based Robust Control of Dead-Time Processes*. Tese (Doutorado) — Universidade Federal de Santa Catarina, 2015. Citado 5 vezes nas páginas 21, 25, 27, 28 e 29.
- 5 MORARI, M.; LEE, J. H.; GARCÍA, C. E. *Model Predictive Control*. [S.l.]: Prentice Hall, 2002. Citado na página 25.
- 6 WAN, E.; MERWE, R. V. D. et al. The unscented kalman filter for nonlinear estimation. In: IEEE. *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. [S.l.], 2000. p. 153–158. Citado 2 vezes nas páginas 31 e 36.
- 7 JØRGENSEN, J. B. *Moving Horizon Estimation and Control*. Tese (Doutorado) — Technical University of Denmark, Denmark, 2004. Citado na página 31.
- 8 SIMON, D. Kalman filtering. *Embedded systems programming*, v. 14, n. 6, p. 72–79, 2001. Citado 2 vezes nas páginas 32 e 33.
- 9 FARAGHER, R. et al. Understanding the basis of the kalman filter via a simple and intuitive derivation. *IEEE Signal processing magazine*, v. 29, n. 5, p. 128–132, 2012. Citado 2 vezes nas páginas 33 e 35.
- 10 HASELTINE, E. L.; RAWLINGS, J. B. Critical evaluation of extended kalman filtering and moving-horizon estimation. *Industrial & engineering chemistry research*, ACS Publications, v. 44, n. 8, p. 2451–2460, 2005. Citado 4 vezes nas páginas 36, 37, 39 e 40.
- 11 DIEHL, M. et al. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, Elsevier, v. 12, n. 4, p. 577–585, 2002. Citado na página 39.
- 12 KÜHL, P. et al. A real-time algorithm for moving horizon state and parameter estimation. *Computers & chemical engineering*, Elsevier, v. 35, n. 1, p. 71–83, 2011. Citado 2 vezes nas páginas 39 e 46.
- 13 DIEHL, M.; FERREAU, H. J.; HAVERBEKE, N. Efficient numerical methods for nonlinear mpc and moving horizon estimation. Springer, p. 391–417, 2009. Citado na página 39.

14 JANG, S. S.; JOSEPH, B.; MUKAI, H. Comparison of two approaches to on-line parameter and state estimation of nonlinear systems. *Industrial & Engineering Chemistry Process Design and Development*, ACS Publications, v. 25, n. 3, p. 809–814, 1986. Citado 2 vezes nas páginas 43 e 44.