

Arnaldo Pinheiro de Azevedo Júnior

Metodologia de Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo

São Luís - MA

2018

Arnaldo Pinheiro de Azevedo Júnior

Metodologia de Controle Preditivo Baseado em Modelo ***Fuzzy* Evolutivo**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão como requisito para obtenção do título de mestre em engenharia de eletricidade na área de concentração de Automação e Controle.

Universidade Federal do Maranhão – UFMA

Programa de Pós-Graduação em Engenharia de Eletricidade – PPGEE

Orientador: Ginalber Luiz de Oliveira Serra

São Luís - MA

2018

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

de Azevedo Júnior, Arnaldo Pinheiro.

Metodologia de Controle Preditivo Baseado em Modelo Fuzzy Evolutivo / Arnaldo Pinheiro de Azevedo Júnior. - 2018.

98 f.

Orientador(a): Ginalber Luiz de Oliveria Serra.

Dissertação (Mestrado) - Programa de Pós-graduação em Engenharia de Eletricidade/ccet, Universidade Federal do Maranhão, São Luis, 2018.

1. Controle preditivo. 2. Modelagem fuzzy evolutiva.
3. Pratical nonlinear model predictive control. I.
Serra, Ginalber Luiz de Oliveria. II. Título.

Metodologia de Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo

Arnaldo Pinheiro de Azevedo Júnior

Dissertação aprovada em 25 de setembro de 2018:

Orientador

Prof. Dr. Ginalber Luiz de Oliveira Serra,
IFMA

Examidador

Prof. Dr. Francisco das Chagas de Souza,
UFMA

Examidador

Prof. Dr. Gilmar Barreto,
UNICAMP

Examidador

Prof. Dr. Orlando Donato Rocha Filho,
IFMA

Agradecimentos

Agradeço a Jeová Deus por ter me concedido saúde e disposição para realizar e concluir este trabalho.

Agradeço aos meus pais, Arnaldo e Filomena, e ao meu irmão Lucas, que sempre deram o apoio espiritual, emocional e material.

Agradeço ao meu orientador, o professor Dr. Ginalber Luiz de Oliveira Serra, pela excelente orientação durante os anos de pesquisa.

Agradeço à Universidade Federal do Maranhão e à CAPES pelo apoio estrutural e financeiro para que essa pesquisa fosse desenvolvida.

Agradeço aos amigos que me apoiaram e a todos os professores que, direta ou indiretamente, fizeram parte da minha formação acadêmica.

“Para todas as coisas tenho força em virtude daquele que me confere poder.”
(Filipenses 4:13)

Resumo

Este trabalho tem como objetivo propor uma metodologia baseada na combinação do controle preditivo com a modelagem *fuzzy* evolutiva. O controle preditivo é uma técnica industrial avançada capaz de calcular o sinal de controle aplicado ao processo a partir de uma predição do seu comportamento futuro. A modelagem *fuzzy* evolutiva é uma técnica de identificação de modelos capaz de adquirir conhecimento do processo na forma de regras *fuzzy* **SE-ENTÃO**, além de evoluir sua estrutura e atualizar seus parâmetros. Esse trabalho propõe uma metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo capaz de controlar processos multivariáveis com dinâmica não linear. A técnica de controle preditivo utilizada foi o *Practical Nonlinear Model Predictive Control* que é capaz de calcular o sinal de controle a partir de uma aproximação do modelo de predição não linear do processo a ser controlado. O modelo de predição utilizado é obtido a partir de uma versão evolutiva da técnica de agrupamento *fuzzy* Gustafson-Kessel e um algoritmo recursivo de mínimos quadrados. O controlador proposto é capaz de melhorar o rastreamento de uma trajetória de referência por evoluir a estrutura do modelo de predição não linear a partir da extração de conhecimento dinâmico das entradas e saídas do processo. Para avaliar a metodologia proposta, a mesma foi aplicada ao controle de três processos *benchmarks* não lineares conhecidos da literatura.

Palavras-chave: controle preditivo. modelagem *fuzzy* evolutiva. practical nonlinear model predictive control.

Abstract

The objective of this work is to propose a methodology based on the combination of predictive control and evolving fuzzy modeling. Predictive control is an advanced industrial technique, capable of calculating the control signal applied to the process from a prediction of its future behavior. Evolving fuzzy modeling is a model identification technique, capable of acquisition of Knowledge of the process in the form of **IF-THEN** fuzzy rules, as well as evolving its structure and updating its parameters. This work proposes a predictive control methodology based on an evolving fuzzy model capable of controlling multivariable processes with nonlinear dynamics. The predictive control technique used is the Practical Nonlinear Model Predictive Control, which calculates the control signal from an approximation of the non-linear prediction model of the process to be controlled. The prediction model used is obtained from an evolving version of the Gustafson-Kessel fuzzy clustering technique and the least squares recursive algorithm. The proposed controller is able to improve its tracking capabilities of a reference trajectory, because, it evolves the structure of the non-linear prediction model from the extraction of dynamic knowledge of the inputs and outputs of the process to be controlled. In order to evaluate the proposed methodology, it was applied to the control of three non-linear benchmarking processes known in the literature.

Keywords: predictive control. evolving fuzzy modeling. practical nonlinear model predictive control.

Lista de ilustrações

Figura 1 – Estrutura geral do MPC: a região delimitada em azul representa a estrutura do controlador preditivo baseado em modelo.	23
Figura 2 – Princípio do horizonte deslizante.	30
Figura 3 – Estrutura geral da Modelagem <i>Fuzzy</i> Evolutiva.	41
Figura 4 – Formato dos grupos criados a partir de um conjunto de dados usando-se diferentes métricas de distância: grupo demarcado pela linha sólida resultante de um agrupamento baseado na norma Euclidiana; grupo demarcado pela linha tracejada resultante de um agrupamento baseado na norma Mahalonobis.	43
Figura 5 – Estrutura geral da Metodologia de Controle Preditivo Baseado em Modelo <i>Fuzzy</i> Evolutivo proposta: a região delimitada em azul representa a estrutura do controlador proposto.	49
Figura 6 – Diagrama esquemático de um processo SISO não linear CSTR.	63
Figura 7 – Experimento 1: dados de entrada q_c [l/min] e saída C_a [mol/l] do processo utilizados para identificação do modelo <i>fuzzy</i> TS inicial.	64
Figura 8 – Experimento 1: validação do modelo <i>fuzzy</i> TS inicial, onde a linha contínua em azul indica a saída do modelo $\tilde{y}$ [mol/l], enquanto que a linha tracejada vermelha representa a saída real C_a [mol/l].	64
Figura 9 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).	65
Figura 10 – Experimento 1: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 1).	66
Figura 11 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$. (cenário 1).	66
Figura 12 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 1).	67
Figura 13 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).	67
Figura 14 – Experimento 1: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 2).	68
Figura 15 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).	68
Figura 16 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 2).	69
Figura 17 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 3).	69
Figura 18 – Experimento 1: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 3).	70
Figura 19 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 3).	70

Figura 20 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 3).	71
Figura 21 – Diagrama esquemático de um processo MIMO não linear com quatro tanques acoplados. A altura dos níveis de água nos tanques 1 e 2 são controlados por duas bombas.	72
Figura 22 – Experimento 2: entradas u_1 [V] e u_2 [V] aplicadas ao processo para identificação do modelo <i>fuzzy</i> TS inicial.	73
Figura 23 – Experimento 2: saídas y_1 [cm] e y_2 [cm] medidas do processo para identificação do modelo <i>fuzzy</i> TS inicial.	73
Figura 24 – Experimento 2: validação do modelo <i>fuzzy</i> TS inicial, onde as linhas contínuas são as saídas de validação y_1 [cm] e y_2 [cm], enquanto que as linhas tracejadas são as saídas do modelo $\tilde{y}_1$ [cm] e $\tilde{y}_2$ [cm].	74
Figura 25 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).	75
Figura 26 – Experimento 2: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 1).	75
Figura 27 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 1).	76
Figura 28 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 1).	76
Figura 29 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 1).	77
Figura 30 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).	77
Figura 31 – Experimento 2: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 2).	78
Figura 32 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).	78
Figura 33 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 2).	79
Figura 34 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 2).	79
Figura 35 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 3).	80
Figura 36 – Experimento 2: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 3).	80
Figura 37 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 3).	81
Figura 38 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 3).	81
Figura 39 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 3).	82
Figura 40 – Experimento 3: Processo SISO Não Linear Térmico.	83

Figura 41 – Experimento 3: dados de entrada u [V] e saída y [°C] do processo utilizados para identificação do modelo <i>fuzzy</i> TS inicial.	83
Figura 42 – Experimento 3: validação do modelo <i>fuzzy</i> TS inicial, onde a linha contínua em azul indica a saída do modelo $\tilde{y}$ [°C], enquanto que a linha tracejada vermelha representa a saída real y [°C].	84
Figura 43 – Experimento 3: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).	84
Figura 44 – Experimento 3: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 1).	85
Figura 45 – Experimento 3: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 1).	85
Figura 46 – Experimento 3: resposta da saída y [°C] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado u [V] (cenário 1).	86
Figura 47 – Experimento 3: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).	86
Figura 48 – Experimento 3: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 2).	87
Figura 49 – Experimento 3: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).	87
Figura 50 – Experimento 3: resposta da saída y [°C] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado u [V] (cenário 2).	88

Lista de tabelas

Tabela 1 – Parâmetros do processo SISO não linear CSTR.	63
Tabela 2 – Parâmetros do processo MIMO não linear com quatro tanques acoplados.	71

Lista de abreviaturas e siglas

AHLTNM	<i>Adaptive Habitually Linear and Nonlinear Model.</i>
ARIMA	<i>Autoregressive Integrated Moving Average.</i>
ARX	<i>Auto Regressive with eXogenous input.</i>
MPC	Controle Preditivo Baseado em Modelo.
CSTR	<i>Continous-Stirred Tank Reactor.</i>
DENFIS	<i>Dynamic Evolving Neuro-Fuzzy Inference System.</i>
DMC	<i>Dynamic Matrix Control.</i>
eTS	<i>evolving Takagi-Sugeno.</i>
eTS+	<i>evolving Takagi-Sugeno +.</i>
exTS	<i>Extended Evolving Takagi-Sugeno.</i>
EPSAC	<i>Extended Prediction Self Adaptive Control.</i>
EHAC	<i>Extended Horizon Adaptive Control.</i>
FCM	<i>Fuzzy c-means.</i>
FLEXFIS	<i>Flexible Fuzzy Inference Systems.</i>
FMLE	<i>Fuzzy Maximum Likelihood Estimates Clustering.</i>
FWLS	<i>Fuzzily Weighted Least Squares.</i>
GK	<i>Gustafson-Kessel.</i>
GPC	<i>Generalized Predictive Control.</i>
MAC	<i>Model Algorithmic Control.</i>
MIMO	<i>Multiple-Input Multiple-Output.</i>
NLH-DMC	<i>Non Linear Hammerstein - Dynamic Matrix Control.</i>
PNMPC	<i>Pratical Nonlinear Model Predictive Control.</i>
PS	Preditor de Smith.

QP	<i>Quadratic Programming.</i>
RFWLS	<i>Recursive Fuzzily Weighted Least Squares.</i>
SimpleTS	<i>Simplified Method for Learning Evolving Takagi-Sugeno.</i>
SISO	<i>Single-Input Single-Output.</i>
SQP	<i>Sequential Quadratic Programming.</i>
TS	<i>Takagi-Sugeno.</i>
UPC	<i>Unified Predictive Control.</i>

Sumário

1	INTRODUÇÃO	17
1.1	Revisão da Literatura	18
1.2	Motivação	19
1.3	Objetivos da Pesquisa	20
1.3.1	Objetivo Geral	20
1.3.2	Objetivos Específicos	20
1.4	Trabalhos Publicados	21
1.5	Estrutura do Trabalho	21
2	CONTROLE PREDITIVO	22
2.1	Controle Preditivo Baseado em Modelo Linear	24
2.1.1	Modelo de Predição	24
2.1.1.1	Resposta Impulsiva	24
2.1.1.2	Resposta ao Degrau	25
2.1.1.3	Função de Transferência	25
2.1.1.4	Espaço de Estados	26
2.1.2	Modelo de Perturbação	26
2.1.3	Resposta Livre e Forçada	27
2.1.4	Trajectoria de Referência	27
2.1.5	Função Objetivo e Ajuste dos Parâmetros	28
2.1.6	Obtenção do Sinal de Controle	28
2.1.7	Tratamento das Restrições	29
2.2	Controle Preditivo Baseado em Modelo Não Linear	29
2.2.1	Métodos com Múltiplos Modelos Linearizados	32
2.2.2	Métodos com Modelos de Processo Particulares	34
2.2.3	Métodos com Modelos de Processo Não Linear na Função Objetivo	35
2.2.4	<i>Practical Nonlinear Model Predictive Control</i> (PNMPC)	36
2.2.4.1	Obtenção Numérica das Predições	37
2.3	Conclusões	39
3	MODELAGEM FUZZY EVOLUTIVA	40
3.1	Estrutura de Modelo <i>Fuzzy</i> Takagi-Sugeno	41
3.2	Agrupamento <i>Fuzzy</i>	42
3.2.1	Agrupamento <i>Fuzzy</i> em Batelada	44
3.2.2	Agrupamento <i>Fuzzy</i> Evolutivo	45
3.2.2.1	Agrupamento <i>Fuzzy</i> Evolutivo Baseado em Densidade	45

3.2.2.2	Agrupamento <i>Fuzzy</i> Evolutivo Baseado em Erro	46
3.2.2.3	Agrupamento <i>Fuzzy</i> Evolutivo Baseado em Distância	46
3.3	Estimação dos Parâmetros dos Submodelos	46
3.4	Conclusões	48
4	METODOLOGIA DE CONTROLE PREDITIVO BASEADO EM MODELO <i>FUZZY</i> EVOLUTIVO	49
4.1	Etapas de Treinamento	50
4.1.1	Estimação em Batelada dos Parâmetros do Antecedente	50
4.1.2	Estimação em Batelada dos Parâmetros do Consequente	52
4.2	Etapas de Controle	52
4.2.1	Preparação do Modelo de Predição	53
4.2.2	Estimação das Matrizes de Predição	54
4.2.3	Obtenção do Sinal de Controle	56
4.3	Etapas de Atualização do Modelo <i>Fuzzy</i> Evolutivo	56
4.3.1	Atualização Recursiva dos Parâmetros do Antecedente	57
4.3.2	Criação de Novos Grupos	58
4.3.3	Mecanismo de Fusão de Grupos	59
4.3.4	Estimação Recursiva dos Parâmetros do Consequente	60
4.4	Algoritmo para o Controle Preditivo Baseado em Modelo <i>Fuzzy</i> Evolutivo Proposto	60
5	RESULTADOS COMPUTACIONAIS E EXPERIMENTAIS	62
5.1	Experimento 1- Controle de Processo SISO Não Linear CSTR	62
5.1.1	Identificação do Modelo <i>Fuzzy</i> TS Inicial	63
5.1.2	Avaliação do Controlador	63
5.1.2.1	Cenário 1 - Seguimento de Referência	65
5.1.2.2	Cenário 2 - Seguimento de Referência na Presença de Ruído	65
5.1.2.3	Cenário 3 - Rejeição à Distúrbios	66
5.2	Experimento 2- Controle de Nível de um Processo MIMO Não Linear com Quatro Tanques Acoplados	68
5.2.1	Identificação do Modelo <i>Fuzzy</i> TS Inicial	71
5.2.2	Avaliação do Controlador	72
5.2.2.1	Cenário 1 - Seguimento de Referência	74
5.2.2.2	Cenário 2 - Seguimento de Referência na Presença de Ruído	74
5.2.2.3	Cenário 3 - Rejeição à Distúrbios	75
5.3	Experimento 3 - Processo SISO Não Linear Térmico	77
5.3.1	Identificação do Modelo <i>Fuzzy</i> TS Inicial	79
5.3.2	Avaliação do Controlador	80
5.3.2.1	Cenário 1 - Seguimento de Referência	81

5.3.2.2	Cenário 2 - Rejeição à Distúrbios	82
6	CONSIDERAÇÕES FINAIS	89
6.1	Propostas de Trabalhos Futuros	89
	REFERÊNCIAS	91

1 Introdução

O MPC (*Model Predictive Control*) é uma importante técnica de controle avançado desenvolvido na década de 70, com aplicações iniciais nas indústrias química e petroquímica, atualmente indispensável em diversos setores da indústria de processos (TAVAKOLI; NEGNEVITSKY; MUTTAQI, 2017)(KARAMANAKOS et al., 2014)(MAHMOUDI; ALE-ENEJAD; AHMADI, 2017)(LIU et al., 2017). Isso se deve principalmente porque o MPC permite a solução de problemas de controle complexos, como os casos de controle de processos multivariáveis, processos integradores, processos instáveis, processos não lineares e processos com atraso de transporte. Outra característica importante é a presença de restrições nas variáveis de controle e processo, bem como uma relativa praticidade no ajuste dos parâmetros do controlador, que o torna acessível para pessoas com conhecimentos superficiais na teoria de controle. Dentre as estratégias mais utilizadas na indústria estão aquelas aplicadas ao LMPC (*Linear Model Predictive Control*), como as técnicas DMC (*Dynamic Matrix Control*) e GPC (*Generalized Predictive Control*), que consolidaram-se como as principais estratégias preditivas (CAMACHO; BORDONS; ALBA, 2004).

Naturalmente todos os processos são não lineares e por isso, na prática, é comum a linearização em torno de pontos de operação, o que permite a utilização das estratégias de LMPC. Entretanto, a necessidade de um maior desempenho pode requerer modelos mais realistas. Isso é necessário porque o grau de não linearidade de alguns processos controlados pode dificultar a utilização de modelos linearizados, além de que os pontos de operações podem ser variáveis. Nesses casos, modelos não lineares devem ser utilizados para representar a dinâmica de tais processos, de modo que a aplicação do NMPC (*Nonlinear Model Predictive Control*) surge como uma boa solução para essas questões (FINDEISEN; ALLGÖWER, 2002).

No decorrer do tempo surgiram diversas técnicas de NMPC. Uma das várias classificações (PLUCENIO, 2010) divide-as em três conjuntos:

1. Técnicas que utilizam diretamente um modelo não linear geral e também um algoritmo de otimização não linear genérico. Essas técnicas, apesar de gerais, não apresentam desempenho satisfatório (ZHENG et al., 2018)(WOLF; MARQUARDT, 2016);
2. Técnicas que utilizam modelos não lineares específicos para cada situação e um algoritmo de otimização particular. Essas técnicas são relativamente complexas, o que diminui as possibilidades de aplicação, pois o processo não linear em questão deve se adequar a cada caso específico (AYDIN; BONVIN; SUNDMACHER, 2018)(PEDERSEN; AARSNES; GODHAVN, 2018);

3. Técnicas que usam formas aproximadas para o cálculo das predições do processo não linear de forma tal a utilizar um algoritmo de otimização linear (LIMA; SANTOS; NORMEY-RICO, 2015)(NORMEY-RICO; LIMA; SANTOS, 2015)(LOPEZ; NORMEY-RICO, 2018)(ELIAS et al., 2018).

Dentre as técnicas de NMPC do terceiro grupo acima citado, o PNMPC (*Practical Nonlinear Model Predictive Control*) vem apresentando bom desempenho graças a sua facilidade de implementação e baixo custo computacional, o que a torna interessante quando avalia-se simplicidade (PLUCENIO, 2010).

1.1 Revisão da Literatura

A ideia do MPC foi proposta simultaneamente por diferentes pesquisadores. Desse modo, a tarefa de atribuir a alguém os méritos de originalidade torna-se muito difícil. As datas das publicações fornecem apenas uma ideia da precedência, mas os pioneiros nesse estudo foram os práticos industriais que implementaram técnicas MPC muitos anos antes dos primeiros trabalhos aparecerem, embora os artigos não os incluam na história (MACIEJOWSKI, 2002).

No entanto, existe um consenso de que o MPC tenha surgido no ambiente industrial no final dos anos 70 em aplicações na indústria petroquímica. Nesta época surgiu o MAC (*Model Algorithm Control*) (RICHALET et al., 1978) e o DMC que utilizavam modelos do tipo resposta ao impulso ou degrau para descrever processos e consideravam restrições no processo de otimização. O sucesso destes algoritmos nas aplicações deve-se principalmente à simplicidade das ideias utilizadas.

No início dos anos 80 surgiu uma outra família de algoritmos MPC, relacionados ao controle adaptativo e utilizando geralmente uma função de transferência para processos SISO (DATTA; OCHOA, 1996). Dentre estes podem citar-se o GPC (CLARKE; MOHTADI; TUFFS, 1987), o EPSAC (*Extended Prediction Self Adaptive Control*) (KEYSER; CAUWENBERGHE, 1985), o EHAC (*Extended Horizon Adaptive Control*) (YDSTIE, 1984) e o UPC (*Unified Predictive Control*) (SOETERBOEK, 1992).

Finalmente, com o uso de modelos no espaço de estados diversas formulações foram propostas, permitindo fundamentalmente que toda a teoria de controle associada à essa representação fosse utilizada no contexto MPC (LEE; MORARI; GARCIA, 1994).

Do final dos anos 80 até o final dos anos 90, muitos trabalhos foram publicados analisando questões de estabilidade e robustez dos diversos algoritmos (CLARKE; MOHTADI, 1989)(ROBINSON; CLARKE, 1991)(ANSAY; WERTZ, 1997). Considerando aspectos como restrições de igualdade no horizonte ou pre-estabilização do sistema foi possível encontrar procedimentos para obter a estabilidade em malha fechada (KOUVARITAKIS;

ROSSITER; CHANG, 1992)(CLARKE; SCATTOLINI, 1991). Estudos sobre processos com atraso e os problemas de robustez associados também foram publicados (NORMEY-RICO; CAMACHO, 2001). Este período é marcado pelo aparecimento dos primeiros livros sobre MPC e pela solução do problema de estabilidade do MPC com restrições.

No final da década de 90 até hoje a maioria dos trabalhos de pesquisa tem se dedicado ao estudo de NMPC, de processos híbridos e em aplicações onde exigem-se tempos de amostragem muito pequenos (KOUVARITAKIS; CANNON; ROSSITER, 1999)(JÚNIOR; SERRA, 2017)(JÚNIOR; SERRA, 2018)(NORMEY-RICO, 2007)(ELIAS et al., 2018). O LMPC, por outro lado, consolidou-se como a técnica de controle avançado MIMO mais utilizada na indústria (CAMACHO; BORDONS; ALBA, 2004).

1.2 Motivação

A eficiência do NMPC depende muito da qualidade do modelo não linear do processo. Assim, técnicas de identificação de sistemas desempenham um papel importante no desempenho do controlador (OVIEDO; VANDEWALLE; WERTZ, 2004). Por este motivo, diferentes técnicas de modelagem não linear são utilizadas no contexto NMPC (RASTEGAR; ARAÚJO; MENDES, 2016)(BOULKAIBET et al., 2017)(ESKI; TEMÜRLLENK, 2013). Dentre essas, a modelagem *fuzzy* (desenvolvida por Zadeh (1975)) tornou-se um ativo campo de pesquisa graças a sua capacidade de lidar com incertezas, bem como na descrição de processos com dinâmica não linear (LUGHOFFER, 2011). Consequentemente, a combinação de modelos *fuzzy* com o MPC, FMPC (*Fuzzy Model Predictive Control*), proposto pela primeira vez por Yasunobu e Miyamoto (1984) continua sendo, até hoje, uma técnica muito promissora (KILLIAN; KOZEK, 2018)(KILLIAN; KOZEK, 2017)(BOULKAIBET et al., 2017)(ARIÑO; QUEROL; SALA, 2017)(ZHANG; WU; SHEN, 2017).

De modo geral, existem diferentes métodos para obter um modelo *fuzzy*. Inicialmente, surgiu a suposição de que a estrutura dos modelos *fuzzy* era inteiramente construída com base no conhecimento humano (especialista) (ZADEH, 1975). Posteriormente, foi demonstrado que esta poderia ser desenvolvida com base em um particionamento inteligente do espaço de dados (algoritmos de agrupamento) (BABUŠKA, 1998). Métodos para o projeto automático de modelos *fuzzy* (por exemplo, usando algoritmos genéticos) também foram desenvolvidos (ANGELOV; BUSWELL, 2001)(ROUBOS; SETNES; ABONYI, 2001).

Entretanto, todos esses métodos são aplicáveis de modo *off-line*, ou seja, com fornecimento de dados em batelada. Essas aproximações, por sua vez, podem apresentar um desempenho insatisfatório caso os conjuntos de dados utilizados para obtenção dos modelos não contemplem todos os pontos de operação do processo (ANGELOV; FILEV; KASABOV,

2010). Além disso, por causa de desgaste, ruídos ou perturbações o comportamento dinâmico do processo pode variar com o tempo. Nestes casos, uma abordagem *on-line* para identificação do modelo *fuzzy* é necessária. Dentre as abordagens disponíveis, as técnicas de modelagem evolutivas, que consistem em um mecanismo de aprendizagem baseado na evolução da estrutura do modelo são mais indicadas (ANGELOV, 2013). Estas técnicas, por sua vez, resultaram no desenvolvimento de diversos algoritmos em anos recentes (ANGELOV; FILEV, 2004)(LUGHOFFER; KLEMENT, 2005)(KASABOV; SONG, 2002)(DOVŽAN; ŠKRJANC, 2011)(SOROOSH; KALHOR, 2014).

Até o presente momento, no entanto, poucos autores exploraram a aplicação de técnicas de modelagem evolutivas no contexto MPC. Em Zdesar, Dovzan e Škrjanc (2014) é proposto um algoritmo de controle preditivo com dois graus de liberdade aplicado a uma classe de processos MIMO (*Multiple-Input Multiple-Output*) não lineares. Em Rastegar, Araújo e Mendes (2016) foi proposta uma metodologia de modelagem *fuzzy* evolutiva baseado no NUFGA (*New Unsupervised Fuzzy Algorithm*) aplicado em um esquema de controle preditivo adaptativo em processos SISO (*Single-Input Single-Output*) não lineares. Em Blazic, Dovzan e Škrjanc (2014) é proposta uma metodologia de controle *fuzzy* preditivo adaptativo com antecedente evolutivo aplicada a processos SISO não lineares com dinâmica dominante de primeira ordem.

Embora essas abordagens sejam eficientes, a utilização de técnicas evolutivas no contexto MPC é um campo de pesquisa aberto. Nesse trabalho é proposta uma metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo, diferenciando-se pela utilização do algoritmo PNMPC, que dentre diversas vantagens possui fácil implementação e baixo custo computacional, o que é essencial para aplicações evolutivas (ANGELOV; FILEV; KASABOV, 2010).

1.3 Objetivos da Pesquisa

1.3.1 Objetivo Geral

Desenvolver uma metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo com aplicações em processos não lineares.

1.3.2 Objetivos Específicos

- elaborar um algoritmo para identificação de modelo *fuzzy* evolutivo;
- desenvolver um algoritmo que integre o PNMPC com a identificação do modelo *fuzzy* evolutivo;
- aplicar a metodologia em três processos não lineares.

1.4 Trabalhos Publicados

As publicações produzidas no decorrer desta pesquisa são apresentadas nesta seção. Dois artigos em congressos foram publicados, conforme segue:

- AZEVEDO JÚNIOR, A. P.; SERRA, G. L. O. *Controle Preditivo Baseado em Modelo via Algoritmo Recursivo de Realização de Auto-Sistema*. Conferência Brasileira de Dinâmica, Controle e Aplicações (DINCON 2017). São José do Rio Preto, Brazil.
- AZEVEDO JÚNIOR, A. P.; SERRA, G. L. O. *Metodologia de Controle Preditivo Baseado em Modelo Nebuloso T-S Evolutivo*. XXII Congresso Brasileiro de Automática (CBA 2018). João Pessoa, Brazil.

1.5 Estrutura do Trabalho

Este trabalho, cujo objetivo principal é o desenvolvimento de uma metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo, está organizado da seguinte forma:

- O Capítulo 2 apresenta os principais conceitos associados aos diferentes algoritmos pertencentes à família MPC. Apresenta alguns modelos lineares utilizados para calcular as predições, bem como as características de algumas técnicas NMPC, culminando, por sua vez, no PNMPC, que é utilizado nesta dissertação;
- O capítulo 3 descreve a modelagem *fuzzy* evolutiva, com base na estrutura de modelo *fuzzy* Takagi-Sugeno, evidenciando como estimar os parâmetros do antecedente e consequente. Este capítulo, por sua vez, apresenta os principais conceitos dos algoritmos de agrupamento evolutivo, bem como técnicas para estimação recursiva dos parâmetros dos submodelos;
- O capítulo 4 formula a metodologia de controle preditivo baseado em modelo *fuzzy* utilizada nesta dissertação. Este capítulo é dividido em seções que esclarecem as diferentes etapas do projeto de controle, culminando, ao fim de cada seção, em um algoritmo que detalha os procedimentos matemáticos para a aplicação da estratégia;
- O capítulo 5 apresenta os resultados obtidos na implementação da metodologia proposta no controle de tres processos não lineares bem conhecidos da literatura, a saber: o continuous-stirred tank (CSTR), um processo de controle de nível MIMO com quatro tanques acoplados e um processo SISO não linear térmico;
- O capítulo 6 traz as considerações finais, discussões sobre as principais contribuições e sugestões de trabalhos futuros.

2 Controle Preditivo

Embora se considere que o MPC tenha surgido na década de 70, o uso do conceito de predição em um sistema de controle é bem mais antigo. A ação preditiva de um controlador pode ser interpretada como uma maneira de gerar uma ação capaz de prever um determinado efeito na resposta do processo e evitá-lo ou ao menos diminuí-lo (AGUIRRE, 2007). A forma mais simples de predição pode ser vista em um controlador PD (Proporcional Derivativo), onde a ação de controle $u(t)$, que atua sobre o erro ($e(t) = r(t) - y(t)$) entre a referência ($r(t)$) e a saída ($y(t)$) pode ser vista como uma predição linear de $e(t)$ T_d unidades de tempo à frente de t :

$$u(t) = K_c \left[e(t) + T_d \frac{de(t)}{dt} \right] \cong K_c \hat{e}(t + T_d | t). \quad (2.1)$$

onde K_c é o ganho do controlador, T_d é o tempo derivativo e $\hat{e}(t + T_d | t)$ é a predição do erro $e(t)$ no instante de tempo $t + T_d$ com a informação disponível em t . Esta ação preditiva permite, por exemplo, controlar processos com pequenos atrasos de uma maneira mais eficiente que a ação derivativa (ÅSTRÖM; HÄGGLUND, 1988).

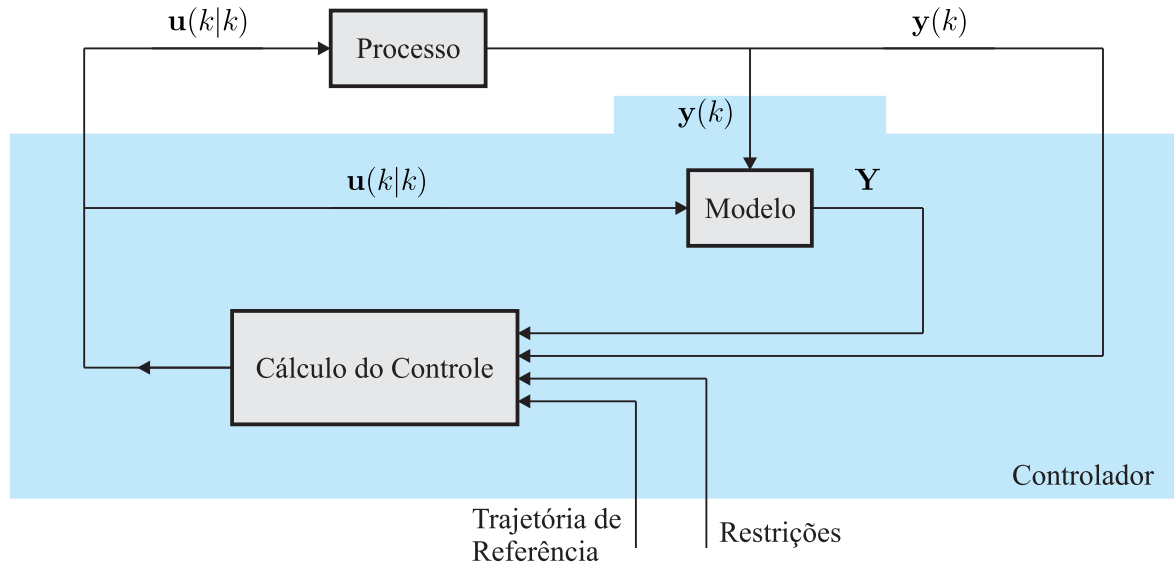
Outro controlador que utiliza a ideia de predição é o PS (Preditor de Smith), proposto por Smith (1957). Este, por sua vez, utiliza um modelo interno para calcular a predição da saída do processo no instante $t + L$ com informações obtidas em t ($\hat{y}(t + L | t)$) com o objetivo de compensar os efeitos do atraso de transporte L . Logo o PS pode ser considerado o primeiro MPC proposto na literatura.

Naturalmente, o MPC não se limita ao estudo de processos com atraso e considera a predição da saída do processo até um tempo $t + j$ ($j > 0$), permitindo a utilização de diferentes modelos do processo e, em geral, calculam a lei de controle utilizando algoritmos de otimização que podem incluir o tratamento de restrições (MACIEJOWSKI, 2002). Nos modelos discretos considerados $t_k = kT_s$ representa o instante da k -ésima amostra (k inteiro) e T_s o período de amostragem. Para simplificar a notação usar-se-á $x(k)$ para representar $x(t_k)$.

A estrutura geral do MPC é apresentada na figura 1 onde se observa seu funcionamento: com o controle atual ($\mathbf{u}(k|k)$)¹, saída atual ($\mathbf{y}(k)$) e um modelo calcula-se primeiro a predição da saída futura do processo $\mathbf{Y}$; com estas informações um bloco calcula o sinal de controle a ser aplicado ao processo com base na minimização de uma função objetivo sujeita às restrições nas variáveis do processo e/ou controle (CAMACHO; BORDONS; ALBA, 2004).

¹ Controle calculado no instante k com base em informações obtidas no instante k .

Figura 1 – Estrutura geral do MPC: a região delimitada em azul representa a estrutura do controlador preditivo baseado em modelo.



Fonte: Autor

O MPC, por sua vez, não designa uma estratégia de controle específica, mas representa uma família de métodos que foram desenvolvidos considerando as seguintes ideias comuns baseadas no conceito de predição (CAMACHO; BORDONS; ALBA, 2004):

- Uso de um modelo explícito do processo e perturbação para predição da saída num determinado horizonte de tempo finito;
- Cálculo de uma sequência de controle para todo o horizonte a partir da minimização de uma determinada função objetivo;
- O horizonte é deslizante, de modo que a cada instante o horizonte é deslocado um passo para frente. Aplica-se unicamente à ação de controle daquele instante e desconsidera-se o resto dos controles dentro do horizonte.

As diferenças entre os diversos algoritmos MPC existentes devem-se basicamente à escolha dos tipos de modelos para o processo e as perturbações, ao tipo de função objetivo, ao procedimento para manipular as restrições e o cálculo do controle (ROSSITER, 2017). Neste capítulo serão analisadas as principais características dos algoritmos MPC. Inicialmente será feita uma introdução sobre o LMPC, seguida de uma análise das principais técnicas NMPC e finalizando, por sua vez, com o detalhamento do algoritmo PNMPCC utilizado neste trabalho.

2.1 Controle Preditivo Baseado em Modelo Linear

Nesta seção serão investigados os principais elementos que compõem um LMPC (em português, Controle Preditivo Baseado em Modelo Linear), a saber: modelo de predição, modelo de perturbação, respostas livre e forçada, trajetória de referência, função objetivo, obtenção do sinal de controle e o tratamento das restrições.

2.1.1 Modelo de Predição

O modelo de predição é provavelmente o elemento mais importante dentro de um MPC, dado que ele deve ser capaz de representar adequadamente a dinâmica do processo para permitir o cálculo das predições da sua saída (AGUIRRE, 2007).

A seguir serão apresentados os principais modelos lineares utilizados para descrição do processo usados em algoritmos de LMPC.

2.1.1.1 Resposta Impulsiva

Utilizado no MAC e em casos especiais no GPC. A relação entrada-saída para processos SISO é expressa por:

$$y(k) = \sum_{i=1}^{\infty} h_i u(k-i) \quad (2.2)$$

onde $y(k)$ e $u(k)$ são os dados de saída e entrada do processo no instante k , respectivamente e sendo h_i (com $i = 1, 2, \dots, \infty$) os coeficientes da resposta ao impulso do processo.

Por possuir infinitos parâmetros, o modelo da equação (2.2) é truncado para os M primeiros valores, ou seja, o modelo somente pode ser aplicado para processos estáveis, nos quais $h_i \rightarrow 0$ quando $i \rightarrow \infty$:

$$y(k) = \sum_{i=1}^M h_i u(k-i). \quad (2.3)$$

sendo a predição da saída j passos a frente dada por:

$$\hat{y}(k+j|k) = \sum_{i=1}^M h_i u(k+j-i|k). \quad (2.4)$$

onde $\hat{y}(k+j|k)$ é a predição da saída no instante $k+j$ com informações obtidas no instante k e $u(k+j|k)$ é o sinal de controle futuro no instante $k+j$ calculado no instante k .

A principal vantagem dessa representação é a descrição direta do comportamento dinâmico do processo como o atraso de transporte e/ou de fase não mínima. As desvantagens

estão na necessidade de um elevado número de parâmetros para o modelo e a impossibilidade de aplicação em processos instáveis.

2.1.1.2 Resposta ao Degrau

Usado pelo DMC (CAMACHO; BORDONS; ALBA, 2004) e suas variantes. É similar ao modelo anterior, mas usa um degrau unitário como sinal de entrada. Para processos SISO estáveis a resposta truncada é dada por:

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

sendo $\Delta u(k) = u(k) - u(k-1)$ o esforço ou incremento de controle e g_i (com $i = 1, 2, \dots, \infty$) os coeficientes da resposta ao degrau do processo. A predição j passos à frente pode ser calculada como

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

onde $\Delta u(k+j|k)$ é o incremento de controle futuro no instante $k+j$ calculado no instante k . Este modelo, por sua vez, possui as mesmas vantagens e desvantagens do anterior.

2.1.1.3 Função de Transferência

Utilizado no GPC (CAMACHO; BORDONS; ALBA, 2004) e baseado no conceito de função de transferência discreta, possui resposta dada por:

$$A(z^{-1})y(k) = B(z^{-1})u(k) \quad (2.7)$$

onde

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n} \\ B(z^{-1}) &= 1 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_{n_b} z^{-n_b} \end{aligned}$$

sendo a_i (com $i = 1, \dots, n$) e b_i (com $i = 1, \dots, n_b$) os parâmetros dos polinômios de saída e entrada, respectivamente. A predição j passos à frente é calculada como:

$$\hat{y}(k+j|k) = \frac{B(z^{-1})}{A(z^{-1})} u(k+j|k) \quad (2.8)$$

A possibilidade de representar processos instáveis e a pouca quantidade de parâmetros necessários para descrição do comportamento dinâmico do processo são algumas das

grandes vantagens desse tipo de representação. No entanto, faz-se necessária uma etapa de identificação da ordem e parâmetros do modelo, o que pode tornar essa representação mais complexa do que quando se utilizam os modelos baseados em resposta impulsiva e baseados em resposta ao degrau.

2.1.1.4 Espaço de Estados

Utiliza o conceito de estado para interpretar a dinâmica do processo como uma transição de estados. Representa-se o modelo MIMO como:

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k)\end{aligned}\tag{2.9}$$

sendo $\mathbf{x}(k)$ o vetor de estado, $\mathbf{y}(k)$ o vetor de saída, $\mathbf{u}(k)$ o vetor de entrada e $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ matrizes de dimensões compatíveis. A predição j passos à frente é calculada como:

$$\begin{aligned}\hat{\mathbf{y}}(k+j|k) &= \mathbf{C}\hat{\mathbf{x}}(k+j|k) \\ &= \mathbf{C}\left[\mathbf{A}^j\mathbf{x}(k) + \sum_{i=1}^j \mathbf{A}^{j-i}\mathbf{B}\mathbf{u}(k+i|k)\right]\end{aligned}\tag{2.10}$$

onde $\hat{\mathbf{x}}(k+j|k)$ e $\hat{\mathbf{y}}(k+j|k)$ são as predições dos vetores de estado e saída, respectivamente, no instante $k+j$ com base em informações obtidas no instante k e $\mathbf{u}(k+i|k)$ é o vetor de entrada futuro no instante $k+i$ calculado no instante k .

A principal vantagem dessa representação é a facilidade na descrição de processos multivariáveis. No entanto, há o inconveniente quando há a necessidade de representar estados não mensuráveis através de observadores, aumentando assim a complexidade do cálculo do controle.

2.1.2 Modelo de Perturbação

A escolha do modelo de perturbação é tão importante quanto a forma de modelar a predição da saída do processo. Os modelos mais utilizados para descrever perturbações determinísticas e estocásticas são os modelos ARIMA (*Auto-Regressive and Integrated Moving Average*) (AGUIRRE, 2015). Nesse caso, a diferença entre a saída do modelo e do processo, representadas por $n(k)$, são modeladas como:

$$n(k) = \frac{C(z^{-1})}{\Delta D(z^{-1})}e(k)\tag{2.11}$$

onde $\Delta = 1 - z^{-1}$ é um integrador e $e(k)$ é um ruído branco de média nula. Os polinômios $C(z^{-1})$ e $D(z^{-1})$ são usados para descrever as características estocásticas de $n(k)$. Este

modelo permite, por exemplo, representar mudanças aleatórias, *offsets* e outros fenômenos normalmente encontrados em ambientes industriais.

2.1.3 Resposta Livre e Forçada

Uma característica comum a todos os algoritmos de LMPC é a utilização dos conceitos de resposta livre e forçada. A ideia é considerar o princípio da superposição para decompor a sequência de controle em duas partes:

$$\mathbf{u}(k) = \mathbf{u}_l(k) + \mathbf{u}_f(k), \quad (2.12)$$

onde $\mathbf{u}_l(k)$ é formada pelos valores passados da ação de controle e pela manutenção dos valores futuros como sendo iguais ao último valor de controle aplicado:

$$\mathbf{u}_l(k+j) = \begin{cases} \mathbf{u}(k+j), & \forall j < 0 \\ \mathbf{u}(k-1), & \forall j \geq 0 \end{cases} \quad (2.13)$$

e $\mathbf{u}_f(k)$ é nulo no passado e igual aos incrementos de controles ($\Delta\mathbf{u}(k)$) a serem aplicados no futuro:

$$\mathbf{u}_f(k+j) = \begin{cases} 0, & \forall j < 0 \\ \Delta\mathbf{u}(k+j), & \forall j \geq 0 \end{cases} \quad (2.14)$$

Desta forma a predição da saída do processo pode ser dividida em duas parcelas: a resposta livre ($\mathbf{y}_l(k)$), que corresponde à predição quando a entrada $\mathbf{u}_l(k)$ é aplicada, e a outra, a resposta forçada ($\mathbf{y}_f(k)$), que corresponde à predição quando o controle é igual a $\mathbf{u}_f(k)$.

2.1.4 Trajetória de Referência

Uma das vantagens do MPC é a utilização do conhecimento prévio da trajetória de referência (quando disponível) para o cálculo do sinal de controle, o que permite, por sua vez que a saída do processo atinja mais rapidamente ou de forma mais suave o novo valor desejado. Esta característica torna-se interessante em algumas aplicações como em robótica, em servoacionamentos e em processos do tipo batelada, onde as referências são conhecidas *a priori* (AGUIRRE, 2007).

Os valores de referência de saída $\omega(k+j)$ utilizados para calcular o controle não são necessariamente coincidentes com a referência de saída real do processo $\mathbf{r}(k+j)$. Normalmente, na prática, utilizam-se estratégias para suavizar as mudanças de referência, de forma similar aos filtros utilizados nas estruturas clássicas de controle com dois graus

de liberdade (CLARKE; MOHTADI; TUFFS, 1987). Uma forma típica para isto é a de um filtro de primeira ordem:

$$\begin{aligned}\omega(k) &= \mathbf{y}(k), \\ \omega(k+j) &= \alpha\omega(k+j-1) + (1-\alpha)\mathbf{r}(k+j), \forall j > 0.\end{aligned}\tag{2.15}$$

onde α ajusta-se entre 0 e 1.

2.1.5 Função Objetivo e Ajuste dos Parâmetros

Em geral, todos os algoritmos MPC consideram como objetivo minimizar um índice J que considera o erro entre as predições da saída ($\hat{\mathbf{y}}(k+j|k)$) e a referência futura conhecida ou predita ($\omega(k+j)$) e penaliza, em alguns casos, o controle $\mathbf{u}(k)$ ou suas variações $\Delta\mathbf{u}(k)$:

$$J = \sum_{j=N_1}^{N_2} \|\hat{\mathbf{y}}(k+j|k) - \omega(k+j)\|_{\mathbf{Q}_\delta}^2 + \sum_{j=1}^{N_u} \|\Delta\mathbf{u}(k+j-1|k)\|_{\mathbf{Q}_\lambda}^2, \tag{2.16}$$

onde N_1 e N_2 são os horizontes de predição mínimo e máximo, N_u é o horizonte de controle, e $\mathbf{Q}_\delta$ (definida positiva) e $\mathbf{Q}_\lambda$ (semi-definida positiva) são, respectivamente, as matrizes de ponderação do erro e do esforço de controle. O horizonte de predição, $N_2 - N_1$, deve ser escolhido para capturar a dinâmica transitória do processo, o que se consegue ajustando-se num valor entre 50% e 95% do tempo de resposta do mesmo. O ajuste de N_1 permite desconsiderar o erro cometido nos primeiros $N_1 - 1$ instantes. Por exemplo, em processos com atraso de d amostras é lógico escolher $N_1 > d$, já que não haverá resposta do processo à entrada $u(k)$ até o tempo $k + d$. N_u não deve ser muito grande para evitar aumentar a dimensão do problema de controle a ser resolvido. Por outro lado, valores muito pequenos tendem a gerar dinâmicas com tempos de resposta muito similares aos de malha aberta. Na prática, sugere-se utilizar valores de N_u entre 1/3 e 1/2 de $N_2 - N_1$ (ROSSITER, 2017).

2.1.6 Obtenção do Sinal de Controle

O sinal de controle é obtido com base no *princípio do horizonte deslizando* que é ilustrado (para um processo SISO) na figura 2: o controlador preditivo tem um modelo interno que é utilizado para prever o comportamento da saída do processo, começando do instante atual k , sobre um horizonte de predição ($N_2 - N_1$); a predição da saída depende de uma sequência de controle calculada ($u(k|k), u(k+1|k), \dots, u(k+N_u-1|k)$)² com o objetivo de gerar o melhor comportamento predito de saída, através da minimização do índice J (equação 2.16); aplica-se, por sua vez, o primeiro elemento da sequência de

² Os valores de controle além do horizonte de controle são mantidos iguais ao último elemento da sequência de controle calculada.

controle ($u(k|k)$) ao processo e, após isso, todo o procedimento é repetido para o instante de tempo discreto seguinte.

2.1.7 Tratamento das Restrições

Na prática, todos os processos estão sujeitos a restrições tanto nas variáveis de saída como nas de entrada. Exemplos disso são os limites máximos e mínimos impostos aos atuadores (como por exemplo válvulas), a máxima velocidade de variação de um acionamento (como por exemplo servoacionamentos) ou os valores limites que podem ser atingidos pelas saídas de um processo devido a questões de segurança. Analiticamente, o MPC pode ser formulado como o problema de calcular o mínimo do índice J sujeito a (s.a.) um conjunto de restrições³ do tipo:

$$\begin{aligned} \mathbf{u}_{\min} &\leq \mathbf{u}(k+j|k) \leq \mathbf{u}_{\max}, & \forall j = 0, 1, \dots, N_u - 1. \\ \mathbf{du}_{\min} &\leq \Delta \mathbf{u}(k+j|k) \leq \mathbf{du}_{\max}, & \forall j = 0, 1, \dots, N_u - 1. \\ \mathbf{y}_{\min} &\leq \hat{\mathbf{y}}(k+j|k) \leq \mathbf{y}_{\max}, & \forall j = N_1, N_1 + 1, \dots, N_2. \end{aligned} \quad (2.17)$$

Quando o modelo é linear o problema a ser resolvido a cada período de amostragem é do tipo QP (*Quadratic Programming*) para o qual existem diversos algoritmos eficientes (LUENBERGER; YE, 2008). A presença das restrições, por outro lado, aumenta a complexidade da solução. Ainda assim, a capacidade do MPC de levar as restrições em consideração é o principal motivo do seu sucesso nas aplicações industriais (CAMACHO; BORDONS; ALBA, 2004).

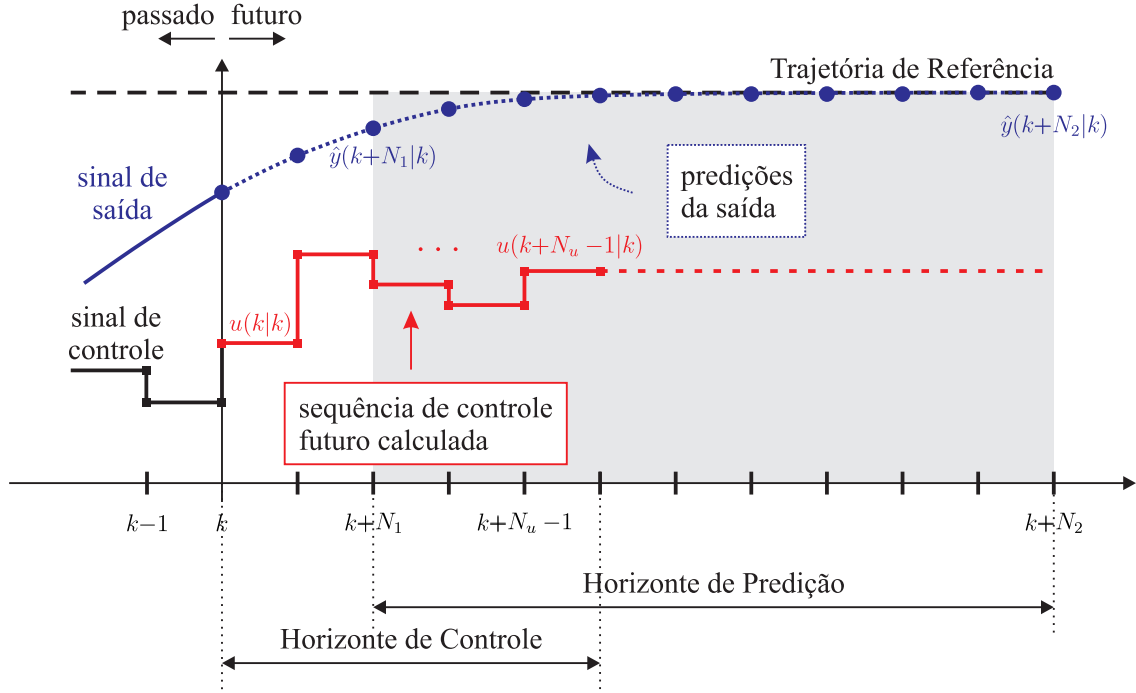
2.2 Controle Preditivo Baseado em Modelo Não Linear

Do ponto de vista conceitual o NMPC (em português, Controle Preditivo Baseado em Modelo Não Linear) não apresenta dificuldades se comparado ao caso linear. Considerando a mesma função objetivo que no caso linear (equação 2.16) e um modelo não linear para o cálculo das predições, o objetivo consiste novamente em encontrar a sequência de controle que minimize o índice J com base no *princípio do horizonte deslizante* (OVIEDO; VANDEWALLE; WERTZ, 2004). Na prática, contudo, são várias as dificuldades encontradas nesse problema: (a) a determinação do modelo não linear do processo quando deve ser obtido por modelagem, (b) a solução do mínimo do índice J que é um problema de otimização não convexo e (c) a análise de estabilidade e robustez da solução (AGUIRRE, 2007).

Por esta razão, uma estratégia comumente utilizada no controle de processos não lineares é a *linearização*, o que reduz significativamente as dificuldades acima citadas pela

³ Esta formulação para restrições é independente do tipo de modelo do processo utilizado.

Figura 2 – Princípio do horizonte deslizante.



Fonte: Autor.

possibilidade da aplicação de algoritmos LMPC. Por exemplo, em variáveis de estado, o modelo não linear MIMO de um processo pode ser escrito como:

$$\begin{aligned}\tilde{\mathbf{x}}(k+1) &= f(\tilde{\mathbf{x}}(k), \mathbf{u}(k)) \\ \tilde{\mathbf{y}}(k) &= g(\tilde{\mathbf{x}}(k))\end{aligned}\quad (2.18)$$

sendo $\tilde{\mathbf{x}}(k)$ e $\tilde{\mathbf{y}}(k)$ os vetores de estado e saída não lineares, respectivamente e $f(\cdot, \cdot)$, $g(\cdot)$ são funções não lineares genéricas (contínuas e diferenciáveis). Considerando um ponto de operação em $(\mathbf{x}^*, \mathbf{u}^*)$, obtém-se o modelo linearizado:

$$\begin{aligned}\bar{\mathbf{x}}(k+1) &= \bar{\mathbf{E}} + \bar{\mathbf{A}}[\bar{\mathbf{x}}(k) - \mathbf{x}^*] + \bar{\mathbf{B}}[\mathbf{u}(k) - \mathbf{u}^*] \\ \bar{\mathbf{y}}(k) &= \bar{\mathbf{F}} + \bar{\mathbf{C}}[\bar{\mathbf{x}}(k) - \mathbf{x}^*]\end{aligned}\quad (2.19)$$

onde $\bar{\mathbf{x}}(k)$ e $\bar{\mathbf{y}}(k)$ são os vetores de estado e saída linearizados, respectivamente e

$$\begin{aligned}\bar{\mathbf{E}} &= f(\mathbf{x}^*, \mathbf{u}^*) \\ \bar{\mathbf{F}} &= g(\mathbf{x}^*, \mathbf{u}^*) \\ \bar{\mathbf{A}} &= \left. \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}^*, \mathbf{u}^*} \\ \bar{\mathbf{B}} &= \left. \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}} \right|_{\mathbf{x}^*, \mathbf{u}^*} \\ \bar{\mathbf{C}} &= \left. \frac{\partial g(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}^*, \mathbf{u}^*}\end{aligned}\quad (2.20)$$

Utilizando uma representação de predição j passos à frente, similar à equação 2.10, obtém-se:

$$\begin{aligned}\hat{\mathbf{x}}(k+j|k) &= \sum_{i=0}^{j-1} \bar{\mathbf{A}}^i \mathbf{L} + \bar{\mathbf{A}}^j \bar{\mathbf{x}}(k) + \sum_{i=1}^k \bar{\mathbf{A}}^{i-1} \bar{\mathbf{B}} \mathbf{u}(k+j-i|k) \\ \hat{\mathbf{y}}(k+j|k) &= \mathbf{M} + \sum_{i=0}^{j-1} \bar{\mathbf{C}} \bar{\mathbf{A}}^i \mathbf{L} + \bar{\mathbf{C}} \bar{\mathbf{A}}^j \bar{\mathbf{x}}(k) + \sum_{i=1}^k \bar{\mathbf{C}} \bar{\mathbf{A}}^{i-1} \bar{\mathbf{B}} \mathbf{u}(k+j-i|k)\end{aligned}\quad (2.21)$$

onde

$$\begin{aligned}\mathbf{L} &= \bar{\mathbf{E}} - \bar{\mathbf{A}} \mathbf{x}^* - \bar{\mathbf{B}} \mathbf{u}^* \\ \mathbf{M} &= \bar{\mathbf{F}} - \bar{\mathbf{C}} \mathbf{x}^*\end{aligned}$$

Considerando o horizonte de predição inicial $N_1 = 1$, final N_2 e de controle N_u , a predição da saída é expressa na forma matricial como:

$$\mathbf{Y} = \underbrace{\mathbf{M}^{N_2} + \mathbf{C}^{N_2} [\Psi + \Phi \bar{\mathbf{x}}(k)]}_{\Gamma} + \mathbf{C}^{N_2} \mathbf{H} \mathbf{U}_l + \underbrace{\mathbf{C}^{N_2} \mathbf{H} \mathbf{T}}_{\mathbf{G}} \mathbf{U}_f \quad (2.22)$$

sendo Γ a resposta livre, $\mathbf{G} \mathbf{U}_f$ a resposta forçada,

$$\begin{aligned}\mathbf{Y} &= \begin{bmatrix} \hat{\mathbf{y}}(k+1|k) \\ \hat{\mathbf{y}}(k+2|k) \\ \vdots \\ \hat{\mathbf{y}}(k+N_2|k) \end{bmatrix} \quad \mathbf{U}_f = \begin{bmatrix} \Delta \mathbf{u}(k|k) \\ \Delta \mathbf{u}(k+1|k) \\ \vdots \\ \Delta \mathbf{u}(k+N_u-1|k) \end{bmatrix} \quad \mathbf{U}_l = \begin{bmatrix} \mathbf{u}(k-1) \\ \mathbf{u}(k-1) \\ \vdots \\ \mathbf{u}(k-1) \end{bmatrix} \\ \Phi &= \begin{bmatrix} \bar{\mathbf{A}} \\ \bar{\mathbf{A}}^2 \\ \vdots \\ \bar{\mathbf{A}}^{N_2} \end{bmatrix} \quad \Psi = \begin{bmatrix} \mathbf{L} \\ \mathbf{L} + \bar{\mathbf{A}} \mathbf{L} \\ \vdots \\ \mathbf{L} + \bar{\mathbf{A}} \mathbf{L} + \dots + \bar{\mathbf{A}}^{N_2-1} \mathbf{L} \end{bmatrix} \quad \mathbf{M}^{N_2} = \begin{bmatrix} \mathbf{M} \\ \mathbf{M} \\ \vdots \\ \mathbf{M} \end{bmatrix} \\ \mathbf{H} &= \begin{bmatrix} \bar{\mathbf{B}} & 0 & \dots & 0 \\ \bar{\mathbf{A}} \bar{\mathbf{B}} & \bar{\mathbf{B}} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbf{A}}^{N_u-1} \bar{\mathbf{B}} & \bar{\mathbf{A}}^{N_u-2} \bar{\mathbf{B}} & \dots & \bar{\mathbf{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbf{A}}^{N_2-2} \bar{\mathbf{B}} & \bar{\mathbf{A}}^{N_2-3} \bar{\mathbf{B}} & \dots & \sum_{i=0}^{N_2-N_u-1} \bar{\mathbf{A}}^i \bar{\mathbf{B}} \\ \bar{\mathbf{A}}^{N_2-1} \bar{\mathbf{B}} & \bar{\mathbf{A}}^{N_2-2} \bar{\mathbf{B}} & \dots & \sum_{i=0}^{N_2-N_u} \bar{\mathbf{A}}^i \bar{\mathbf{B}} \end{bmatrix}\end{aligned}$$

$$\mathbf{C}^{N_2} = \begin{bmatrix} \bar{\mathbf{C}} & 0 & \dots & 0 \\ 0 & \bar{\mathbf{C}} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{\mathbf{C}} \end{bmatrix} \quad \mathbf{T} = \begin{bmatrix} \mathbf{I}^{(m)} & 0 & \dots & 0 \\ \mathbf{I}^{(m)} & \mathbf{I}^{(m)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I}^{(m)} & \mathbf{I}^{(m)} & \dots & \mathbf{I}^{(m)} \end{bmatrix}$$

Uma vez que a representação linear da predição da saída (equação 2.22) é obtida, a sequência de incrementos de controle $\mathbf{U}_f$ é calculada pela minimização do índice J . Isto, por sua vez, resulta em um problema QP. Desse modo, o índice J (equação 2.16) pode ser expresso matricialmente por:

$$J = (\mathbf{W} - \mathbf{Y})^T \mathbf{Q}_\delta (\mathbf{W} - \mathbf{Y}) + \mathbf{U}_f^T \mathbf{Q}_\lambda \mathbf{U}_f \quad (2.23)$$

sendo $\mathbf{W} = [\omega(k+1)^T, \omega(k+2)^T, \dots, \omega(k+N_2)^T]^T$ o vetor de referências futuras. Substituindo a equação 2.22 na equação 2.23, obtém-se:

$$J = J_{\min} + \mathbf{U}_f^T \left(-\mathbf{G}^T \mathbf{Q}_\delta \mathbf{W} + 2\mathbf{G}^T \mathbf{Q}_\delta \Gamma \right) \mathbf{U}_f + \mathbf{U}_f^T \left(\mathbf{G}^T \mathbf{Q}_\delta \mathbf{G} + \mathbf{Q}_\lambda \right) \quad (2.24)$$

onde $J_{\min} = \mathbf{W}^T \mathbf{Q}_\delta \mathbf{W} - \mathbf{W}^T \mathbf{Q}_\delta \Gamma + \Gamma^T \mathbf{Q}_\delta \Gamma - \Gamma^T \mathbf{Q}_\delta \mathbf{W}$ é o valor de J devido à referência e resposta livre que não pode ser modificado por qualquer sinal de controle e que, portanto, não interfere na minimização. A sequência de controle, por sua vez, é calculada como:

$$\mathbf{U} = \mathbf{U}_l + \mathbf{T} \mathbf{U}_f \quad (2.25)$$

onde $\mathbf{U} = [\mathbf{u}(k|k)^T, \mathbf{u}(k+1|k)^T, \dots, \mathbf{u}(k+N_u-1|k)^T]^T$. Dessa forma, com base no *princípio do horizonte deslizante*, aplica-se o primeiro elemento dessa sequência ($\mathbf{u}(k|k)$) ao processo, e após isso todo o procedimento é repetido para o instante de tempo discreto seguinte. Entretanto, como mencionado no capítulo 1, ao passo que a complexidade do processo a ser controlado aumenta, faz-se necessário modelos de predição mais realistas, bem como técnicas que considerem com maior exatidão as características não lineares impostas. As principais estratégias NMPC baseadas nessas características serão apresentadas a seguir.

2.2.1 Métodos com Múltiplos Modelos Linearizados

Naturalmente alguns processos não lineares podem ser linearizados em vários pontos de operação de modo que um modelo não linear resultante pode ser obtido através de

combinações de diferentes modelos linearizados. Por exemplo, o modelo não linear MIMO da equação 2.18 pode ser linearizado como:

$$\begin{aligned}\bar{\mathbf{x}}^i(k+1) &= \bar{\mathbf{E}}^i + \bar{\mathbf{A}}^i [\bar{\mathbf{x}}^i(k) - \mathbf{x}^i] + \bar{\mathbf{B}}^i [\mathbf{u}(k) - \mathbf{u}^i] \\ \bar{\mathbf{y}}^i(k) &= \bar{\mathbf{F}}^i + \bar{\mathbf{C}}^i [\bar{\mathbf{x}}^i(k) - \mathbf{x}^i]\end{aligned}\quad (2.26)$$

onde $\bar{\mathbf{x}}^i(k)$ e $\bar{\mathbf{y}}^i(k)$ são os vetores de estado e saída linearizados no ponto de operação $(\mathbf{x}^i, \mathbf{u}^i)$, respectivamente e

$$\begin{aligned}\bar{\mathbf{E}}^i &= f(\mathbf{x}^i, \mathbf{u}^i) \\ \bar{\mathbf{F}}^i &= g(\mathbf{x}^i, \mathbf{u}^i) \\ \bar{\mathbf{A}}^i &= \left. \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}^i, \mathbf{u}^i} \\ \bar{\mathbf{B}}^i &= \left. \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}} \right|_{\mathbf{x}^i, \mathbf{u}^i} \\ \bar{\mathbf{C}}^i &= \left. \frac{\partial g(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}^i, \mathbf{u}^i}\end{aligned}\quad (2.27)$$

com $i = 1, 2, \dots, c$.

Uma estratégia comumente utilizada é a ponderação entre os modelos linearizados da seguinte forma:

$$\begin{aligned}\bar{\mathbf{x}}(k+1) &= \sum_{i=1}^c \gamma^i \bar{\mathbf{x}}^i(k+1) \\ \bar{\mathbf{y}}(k) &= \sum_{i=1}^c \gamma^i \bar{\mathbf{y}}^i(k)\end{aligned}\quad (2.28)$$

sendo $\sum_{i=1}^c \gamma^i = 1$. Substituindo a equação 2.26 na equação 2.28, obtém-se:

$$\begin{aligned}\bar{\mathbf{x}}(k+1) &= \bar{\mathbf{E}}_t + \bar{\mathbf{A}}_t [\bar{\mathbf{x}}(k) - \mathbf{x}_t] + \bar{\mathbf{B}}_t [\mathbf{u}(k) - \mathbf{u}_t] \\ \bar{\mathbf{y}}(k) &= \bar{\mathbf{F}}_t + \bar{\mathbf{C}}_t [\bar{\mathbf{x}}(k) - \mathbf{x}_t]\end{aligned}\quad (2.29)$$

onde $\bar{\mathbf{E}}_t = \sum_{i=1}^c \gamma^i \bar{\mathbf{E}}^i$, $\bar{\mathbf{F}}_t = \sum_{i=1}^c \gamma^i \bar{\mathbf{F}}^i$, $\bar{\mathbf{A}}_t = \sum_{i=1}^c \gamma^i \bar{\mathbf{A}}^i$, $\bar{\mathbf{B}}_t = \sum_{i=1}^c \gamma^i \bar{\mathbf{B}}^i$, $\bar{\mathbf{C}}_t = \sum_{i=1}^c \gamma^i \bar{\mathbf{C}}^i$, $\mathbf{x}_t = \sum_{i=1}^c \gamma^i \mathbf{x}^i$ e $\mathbf{u}_t = \sum_{i=1}^c \gamma^i \mathbf{u}^i$.

Desse modo, o modelo de predição da equação 2.22 torna-se dependente do ponto de operação γ^i . No entanto, é importante observar que uma aproximação linear ainda é feita ao longo do horizonte de predição o que, por sua vez, pode deteriorar a qualidade da resposta. Por essa razão tal técnica é indicada sempre que as não linearidades do processo forem suaves ao longo do horizonte de predição. Alternativamente, as ponderações podem ser feitas diretamente no sinal de controle, o que corresponde a um LMPC para cada modelo linearizado. Isso, por sua vez, aumenta a carga computacional, pois c problemas de otimização QP devem ser resolvidos a cada instante de amostragem (OVIEDO; VAN-DEWALLE; WERTZ, 2004). Além disso, a escolha de γ^i não é tão simples e já foi estudada

no contexto de RMPC (*Robust Model Predictive Control*) (MANTHANWAR; SAKIZLIS; PISTIKOPOULOS, 2005), AMPC (*Adaptive Model Predictive Control*) (LU; ARKUN, 1999), Min-Max MPC (*Min-Max Model Predictive Control*) (RAMÍREZ et al., 2006) e FMPC (ZHANG; WU; SHEN, 2017).

2.2.2 Métodos com Modelos de Processo Particulares

Quando um processo não linear pode ser modelado por alguns modelos particulares, podem-se utilizar métodos de otimização específicos obtendo soluções mais simples, com melhor desempenho. Considerando, por exemplo, um modelo não linear MIMO de *Hammerstein* que é descrito como

$$\begin{aligned} \mathbf{v}(k) &= f(\mathbf{u}(k)) \\ \tilde{\mathbf{x}}(k+1) &= \mathbf{A}\tilde{\mathbf{x}}(k) + \mathbf{B}\mathbf{v}(k) \\ \tilde{\mathbf{y}}(k) &= \mathbf{C}\tilde{\mathbf{x}}(k) \end{aligned} \quad (2.30)$$

sendo $\mathbf{v}(k)$ o vetor de entrada intermediário. A predição j passos à frente é calculada como:

$$\begin{aligned} \mathbf{v}(k+j|k) &= f(\mathbf{u}(k+j|k)) \\ \hat{\mathbf{y}}(k+j|k) &= \mathbf{C}\hat{\mathbf{x}}(k+j|k) \\ &= \mathbf{C} \left[\mathbf{A}^k \tilde{\mathbf{x}}(k) + \sum_{i=1}^k \mathbf{A}^{i-1} \mathbf{B} \mathbf{v}(k+j-i|k) \right] \end{aligned} \quad (2.31)$$

onde $\mathbf{v}(k+j|k)$ é o vetor de entrada intermediário futuro calculado no instante $k+j$.

O algoritmo NLH-DMC (*Non Linear Hammerstein - Dynamic Matrix Control*) (ZOU et al., 2006), por exemplo, controla o processo não linear modificando o sinal de controle intermediário obtido ($\mathbf{v}(k|k)$) por um compensador de saída da forma

$$\mathbf{u}(k|k) = f^{-1}(\mathbf{v}(k|k)) \quad (2.32)$$

sendo $f^{-1}(\cdot)$ a inversa de $f(\cdot)$. Dessa forma, o algoritmo resulta em um controle pseudo-linear e permite que técnicas de LMPC sejam aplicadas. Outros exemplos de modelos particulares utilizados nesse contexto são as Séries de Volterra e os modelos de *Wiener* (BLOEMEN; BOOM; VERBRUGGEN, 2000)(ŁAWRYŃCZUK, 2016). No entanto, a principal desvantagem desses métodos é a necessidade de que o processo se adeque ao modelo utilizado, pois de outro modo não será possível garantir um bom desempenho do controlador conforme alterem-se os pontos de operação.

2.2.3 Métodos com Modelos de Processo Não Linear na Função Objetivo

Considerando, por exemplo, o modelo não linear MIMO da equação 2.18, horizonte de predição inicial $N_1 = 1$, final N_2 e de controle N_u , as sequências de controles futuros são descritas como:

$$\begin{aligned}
 \mathbf{u}(k|k) &= \mathbf{u}(k-1) + \Delta\mathbf{u}(k|k) \\
 \mathbf{u}(k+1|k) &= \mathbf{u}(k-1) + \Delta\mathbf{u}(k|k) + \Delta\mathbf{u}(k+1|k) \\
 &\vdots = \vdots \\
 \mathbf{u}(k+N_u-1|k) &= \mathbf{u}(k-1) + \Delta\mathbf{u}(k|k) + \Delta\mathbf{u}(k+1|k) + \cdots + \Delta\mathbf{u}(k+N_u-1|k)
 \end{aligned} \tag{2.33}$$

A predição dos estados pode ser obtida por:

$$\begin{aligned}
 \hat{\mathbf{x}}(k+1|k) &= f(\tilde{\mathbf{x}}(k), \mathbf{u}(k|k)) \\
 \hat{\mathbf{x}}(k+2|k) &= f(\hat{\mathbf{x}}(k+1|k), \mathbf{u}(k+1|k)) \\
 &\vdots = \vdots \\
 \hat{\mathbf{x}}(k+N_2|k) &= f(\hat{\mathbf{x}}(k+N_2-1|k), \mathbf{u}(k+N_u-1|k))
 \end{aligned} \tag{2.34}$$

Substituindo as relações obtidas na equação 2.33 na equação 2.34, obtém-se:

$$\begin{aligned}
 \hat{\mathbf{x}}(k+1|k) &= f(\tilde{\mathbf{x}}(k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k)) \\
 \hat{\mathbf{x}}(k+2|k) &= f(\hat{\mathbf{x}}(k+1|k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \Delta\mathbf{u}(k+1|k)) \\
 &\vdots = \vdots \\
 \hat{\mathbf{x}}(k+N_2|k) &= f(\hat{\mathbf{x}}(k+N_2-1|k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \dots, \Delta\mathbf{u}(k+N_u-1|k))
 \end{aligned} \tag{2.35}$$

A predição da saída, por sua vez, é expressa por:

$$\begin{aligned}
 \hat{\mathbf{y}}(k+1|k) &= g(f(\tilde{\mathbf{x}}(k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k))) \\
 &= h_1(\tilde{\mathbf{x}}(k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k)) \\
 \hat{\mathbf{y}}(k+2|k) &= g(f(\hat{\mathbf{x}}(k+1|k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \Delta\mathbf{u}(k+1|k))) \\
 &= h_2(\tilde{\mathbf{x}}(k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \Delta\mathbf{u}(k+1|k)) \\
 &\vdots = \vdots \\
 \hat{\mathbf{y}}(k+N_2|k) &= g(f(\hat{\mathbf{x}}(k+N_2-1|k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \dots, \Delta\mathbf{u}(k+N_u-1|k))) \\
 &= h_{N_2}(\tilde{\mathbf{x}}(k), \mathbf{u}(k-1), \Delta\mathbf{u}(k|k), \dots, \Delta\mathbf{u}(k+N_u-1|k))
 \end{aligned} \tag{2.36}$$

sendo $h_1, \dots, h_{N_2}$ funções não lineares (contínuas e diferenciáveis). Desse modo, o modelo de predição não linear da saída do processo $\tilde{\mathbf{Y}}$ é dado por:

$$\tilde{\mathbf{Y}} = H(\tilde{\mathbf{y}}_p, \mathbf{u}_p, \mathbf{U}_f) \tag{2.37}$$

onde $\tilde{\mathbf{Y}} = [\hat{\mathbf{y}}(k+1|k)^T, \hat{\mathbf{y}}(k+2|k)^T, \dots, \hat{\mathbf{y}}(k+N_2|k)^T]^T$, $H(\dots)$ é uma função não linear (contínua e diferenciável), $\tilde{\mathbf{y}}_p = \tilde{\mathbf{x}}(k)$ e $\mathbf{u}_p = \mathbf{u}(k-1)$ são vetores que contém informações passadas da saída e entrada do processo, respectivamente⁴. A relação da equação 2.37 garante, por sua vez, que as únicas variáveis capazes de modificar o comportamento da predição da saída são os incrementos de controle $\mathbf{U}_f$. Assim o índice J a ser minimizado escreve-se como:

$$J = [\mathbf{W} - H(\tilde{\mathbf{y}}_p, \mathbf{u}_p, \mathbf{U}_f)]^T \mathbf{Q}_\delta [\mathbf{W} - H(\tilde{\mathbf{y}}_p, \mathbf{u}_p, \mathbf{U}_f)] + \mathbf{U}_f^T \mathbf{Q}_\lambda \mathbf{U}_f \quad (2.38)$$

Nesse caso, a otimização é realizada por algoritmos desenvolvidos para minimização de funções não lineares como, por exemplo, o SQP (*Sequential Quadratic Programming*), os quais também baseiam-se no *princípio do horizonte deslizante* (NOCEDAL; WRIGHT, 2006). No entanto, como discutido em Cannon (2004), esses métodos possuem a desvantagem de apresentarem incerteza no tempo de resposta, devido a questões iterativas de convergência, mínimos locais etc.

2.2.4 Pratical Nonlinear Model Predictive Control (PNMPC)

A estratégia PNMPC, proposta por Plucenio (2010), fornece um caminho simples e eficiente para o NMPC. Como visto na seção 2.1.3, todos os algoritmos de LMPC fazem uso do princípio da superposição para obter uma representação linear das predições da saída, $\mathbf{Y}$, em função dos incrementos de controle, $\mathbf{U}_f$ (equação 2.22). Porém, quando um modelo não linear é utilizado para calcular as predições o princípio da superposição não se faz verdadeiro. O PNMPC, contudo, baseia-se na premissa de que, independentemente da linearidade dos processos, as únicas variáveis que podem alterar o valor das predições futuras são os incrementos de controle $\mathbf{U}_f$ (equação 2.36) e por essa razão utiliza-se uma linearização em função dessas variáveis e não a partir do conceito de ponto de operação (seção 2.2.1). Assim, a partir de qualquer modelo (linear ou não), a predição pode ser expressa por:

$$\tilde{\mathbf{Y}} \approx \tilde{\Gamma} + \tilde{\mathbf{G}}\mathbf{U}_f \quad (2.39)$$

sendo $\tilde{\Gamma}$ a resposta livre não linear do processo e $\tilde{\mathbf{G}} = \frac{\partial \tilde{\mathbf{Y}}}{\partial \mathbf{U}_f}$ o Jacobiano de $\tilde{\mathbf{Y}}$. Essa aproximação permite, por exemplo, que algoritmos de otimização QP sejam aplicados no contexto NMPC com maior eficiência que os métodos anteriormente mencionados, pois a não linearidade do modelo é considerada no cálculo da resposta livre $\tilde{\Gamma}$, enquanto que uma aproximação linear é utilizada no cálculo da matriz $\tilde{\mathbf{G}}$. Essa representação nada mais

⁴ A mesma relação pode ser obtida para qualquer tipo de modelo de processo.

é do que uma linearização de primeira ordem da série de Taylor para $\mathbf{U}_f = 0$ do modelo de predição não linear $\tilde{\mathbf{Y}}$. A matriz $\tilde{\mathbf{G}}$, portanto, é dada por:

$$\tilde{\mathbf{G}} = \begin{bmatrix} \frac{\partial \hat{\mathbf{y}}(k+1|k)}{\partial \Delta \mathbf{u}(k|k)} & 0 & \cdots & 0 \\ \frac{\partial \hat{\mathbf{y}}(k+2|k)}{\partial \Delta \mathbf{u}(k|k)} & \frac{\partial \hat{\mathbf{y}}(k+2|k)}{\partial \Delta \mathbf{u}(k+1|k)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{\mathbf{y}}(k+N_u|k)}{\partial \Delta \mathbf{u}(k|k)} & \frac{\partial \hat{\mathbf{y}}(k+N_u|k)}{\partial \Delta \mathbf{u}(k+1|k)} & \cdots & \frac{\partial \hat{\mathbf{y}}(k+N_u|k)}{\partial \Delta \mathbf{u}(k+N_u-1|k)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{\mathbf{y}}(k+N_2|k)}{\partial \Delta \mathbf{u}(k|k)} & \frac{\partial \hat{\mathbf{y}}(k+N_2|k)}{\partial \Delta \mathbf{u}(k+1|k)} & \cdots & \frac{\partial \hat{\mathbf{y}}(k+N_2|k)}{\partial \Delta \mathbf{u}(k+N_u-1|k)} \end{bmatrix}_{\mathbf{U}_f=0} \quad (2.40)$$

onde a forma triangular inferior da matriz $\tilde{\mathbf{G}}$ se deve à causalidade do processo, ou seja

$$\frac{\partial \hat{\mathbf{y}}(k+j|k)}{\partial \Delta \mathbf{u}(k+i|k)} = 0 \quad (2.41)$$

para $i \geq j$. Considerando um processo MIMO com m entradas e p saídas, cada elemento da matriz $\tilde{\mathbf{G}}$ é calculado como:

$$\frac{\partial \hat{\mathbf{y}}(k+j|k)}{\partial \Delta \mathbf{u}(k+i|k)} = \begin{bmatrix} \frac{\partial \hat{y}_1(k+j|k)}{\partial \Delta u_1(k+i|k)} & \cdots & \frac{\partial \hat{y}_1(k+j|k)}{\partial \Delta u_m(k+i|k)} \\ \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}_p(k+j|k)}{\partial \Delta u_1(k+i|k)} & \cdots & \frac{\partial \hat{y}_p(k+j|k)}{\partial \Delta u_m(k+i|k)} \end{bmatrix} \quad (2.42)$$

com

$$\hat{\mathbf{y}}(k+j|k) = [\hat{y}_1(k+j|k), \dots, \hat{y}_p(k+j|k)]^T \quad (2.43)$$

$$\Delta \mathbf{u}(k+i|k) = [\Delta u_1(k+i|k), \dots, \Delta u_m(k+i|k)]^T \quad (2.44)$$

para todo i, j . A matriz $\tilde{\Gamma}$, por outro lado, é obtida via simulação do comportamento do processo na ausência de incrementos de controle futuros, ou seja, fazendo $\mathbf{U}_f = 0$.

2.2.4.1 Obtenção Numérica das Predições

O enfoque dado por [Plucenio \(2010\)](#) foi a obtenção numérica das predições de $\tilde{\mathbf{Y}}$. Sendo assim, bastante útil, pois existem diferentes técnicas para modelagem de processos

não lineares. Com base no modelo de predição da equação 2.37 calcula-se matriz $\tilde{\Gamma}$ fazendo $\mathbf{U}_f = 0$, ou seja

$$\tilde{\Gamma} = H(\tilde{\mathbf{y}}_p, \mathbf{u}_p) \quad (2.45)$$

Para o cálculo da matriz $\tilde{\mathbf{G}}$ divide-se a mesma como:

$$\tilde{\mathbf{G}} = [\tilde{\mathbf{G}}_1, \dots, \tilde{\mathbf{G}}_{N_u}] \quad (2.46)$$

onde

$$\tilde{\mathbf{G}}_i = \begin{bmatrix} \frac{\partial \hat{\mathbf{y}}(k+1|k)}{\partial \Delta \mathbf{u}(k+i-1|k)} \\ \vdots \\ \frac{\partial \hat{\mathbf{y}}(k+N_2|k)}{\partial \Delta \mathbf{u}(k+i-1|k)} \end{bmatrix} = \begin{bmatrix} \frac{\partial \hat{y}_1(k+1|k)}{\partial \Delta u_1(k+i-1|k)} & \cdots & \frac{\partial \hat{y}_1(k+1|k)}{\partial \Delta u_m(k+i-1|k)} \\ \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}_p(k+1|k)}{\partial \Delta u_1(k+i-1|k)} & \cdots & \frac{\partial \hat{y}_p(k+1|k)}{\partial \Delta u_m(k+i-1|k)} \\ \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}_1(k+N_2|k)}{\partial \Delta u_1(k+i-1|k)} & \cdots & \frac{\partial \hat{y}_1(k+N_2|k)}{\partial \Delta u_m(k+i-1|k)} \\ \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}_p(k+N_2|k)}{\partial \Delta u_1(k+i-1|k)} & \cdots & \frac{\partial \hat{y}_p(k+N_2|k)}{\partial \Delta u_m(k+i-1|k)} \end{bmatrix} \quad (2.47)$$

com $i = 1, 2, \dots, N_u$. A matriz $\tilde{\mathbf{G}}_i$ também pode ser dividida como:

$$\tilde{\mathbf{G}}_i = [\tilde{\mathbf{G}}_i^1, \dots, \tilde{\mathbf{G}}_i^m] \quad (2.48)$$

sendo $\tilde{\mathbf{G}}_i^j = \left[\frac{\partial \hat{y}_1(k+1)}{\partial \Delta u_j(k+i-1|k)}, \dots, \frac{\partial \hat{y}_p(k+1)}{\partial \Delta u_j(k+i-1|k)}, \dots, \frac{\partial \hat{y}_1(k+N_2)}{\partial \Delta u_j(k+i-1|k)}, \dots, \frac{\partial \hat{y}_p(k+N_2)}{\partial \Delta u_j(k+i-1|k)} \right]^T$ que pode ser aproximada por:

$$\tilde{\mathbf{G}}_i^j = \lim_{\Delta u_j(k+i-1|k) \rightarrow 0} \frac{H(\tilde{\mathbf{y}}_p, \mathbf{u}_p, \Delta u_j(k+i-1|k)) - \tilde{\Gamma}}{\Delta u_j(k+i-1|k)} \quad (2.49)$$

para todo $j = 1, \dots, m$. A equação 2.49 pode ser lida como o limite da diferença entre a resposta do modelo de predição $\tilde{\mathbf{Y}}$, quando um pequeno incremento na j -ésima entrada ($\Delta u_j(k+i-1|k)$) é aplicado, pelo comportamento das predições quando nenhum incremento é aplicado (resposta livre $\tilde{\Gamma}$) dividido pela variação da j -ésima entrada ($\Delta u_j(k+i-1|k)$). De modo que a equação 2.49 pode ser reescrita como:

$$\tilde{\mathbf{G}}_i^j = \frac{H(\tilde{\mathbf{y}}_p, \mathbf{u}_p, \epsilon_{(i-1),j}) - \tilde{\Gamma}}{\epsilon_{(i-1),j}} \quad (2.50)$$

onde $\epsilon_{(i-1),j}$ é um pequeno incremento de controle aplicado no instante $k+i-1$ na j -ésima entrada com $i = 1, 2, \dots, N_u$ e $j = 1, 2, \dots, m$.

Uma vez que as matrizes $\tilde{\Gamma}$ e $\tilde{\mathbf{G}}$ são calculadas, o incremento de controle $\mathbf{U}_f$ é obtido pela minimização do índice J , expresso por:

$$J = J_{\min} + \mathbf{U}_f^T \left(-\tilde{\mathbf{G}}^T \mathbf{Q}_\delta \mathbf{W} + 2\tilde{\mathbf{G}}^T \mathbf{Q}_\delta \tilde{\Gamma} \right) \mathbf{U}_f + \mathbf{U}_f^T \left(\mathbf{G}^T \mathbf{Q}_\delta \mathbf{G} + \mathbf{Q}_\lambda \right) \quad (2.51)$$

onde $J_{\min} = \mathbf{W}^T \mathbf{Q}_\delta \mathbf{W} - \mathbf{W}^T \mathbf{Q}_\delta \tilde{\Gamma} + \tilde{\Gamma}^T \mathbf{Q}_\delta \tilde{\Gamma} - \tilde{\Gamma}^T \mathbf{Q}_\delta \mathbf{W}$ é o valor de J que não interfere na minimização. Desse modo, o problema a ser resolvido novamente corresponde a um *quadratic programming*.

2.3 Conclusões

Neste capítulo apresentaram-se os principais elementos que compõem um MPC. De modo geral destacaram-se os modelos utilizados para predição do comportamento futuro do processo, bem como as aproximações baseadas nos mesmos (linear e não linear). Salientou-se que a *linearização* é uma técnica prática para o controle de processos não lineares, mas que pode ser melhorada quando utilizam-se múltiplos modelos linearizados. Observou-se também que em casos especiais, onde a modelagem é feita por meio de modelos particulares, o cálculo do controle também pode ser relativamente simples, mas existe a desvantagem de nem todos os processos serem modelados de tal modo. Em seguida, apresentaram-se técnicas que utilizam um modelo não linear diretamente na função objetivo, mas que apresentam incerteza no tempo de resposta. Finalmente, apresentou-se o PNMPC, que também baseia-se na ideia de *linearização*, mas não a partir do conceito de ponto de operação e sim através de uma linearização em relação às únicas variáveis que alteram o comportamento predito do processo que são, respectivamente, os incrementos de controle.

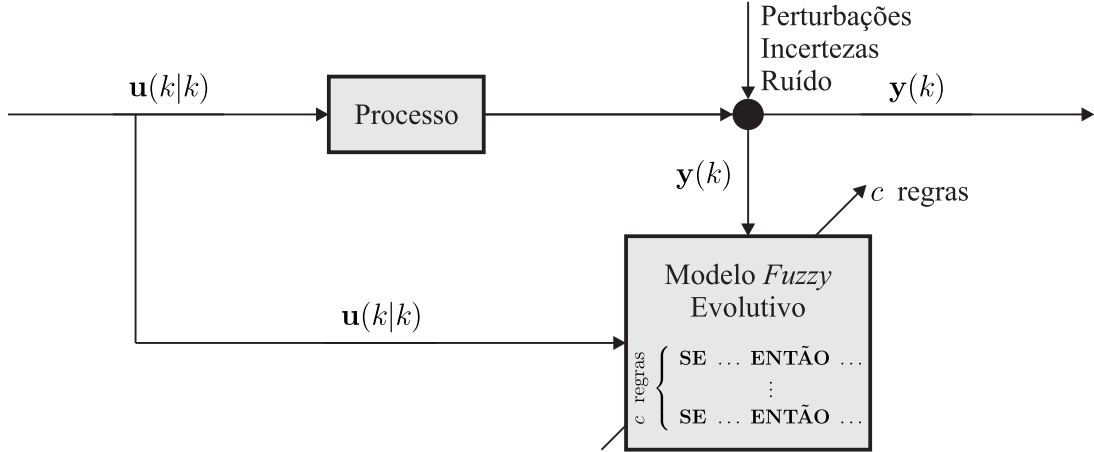
3 Modelagem *Fuzzy* Evolutiva

Os primeiros trabalhos publicados sobre modelos capazes de adaptar sua estrutura surgiram na década de 90. Essas abordagens, inicialmente baseadas em redes neurais, impulsionaram o desenvolvimento de metodologias cada vez mais eficientes e interpretáveis (FILHO, 2011). Dentre estas, destacam-se as pesquisas de Angelov (2002), Kasabov (2002) e Angelov e Kasabov (2005), que apresentaram modelos capazes de expandir ou simplificar sua estrutura a fim de se adaptar às alterações do ambiente ou mesmo mudanças internas. Essas pesquisas, por sua vez, resultaram nas que vieram a ser conhecidas como técnicas de modelagem evolutiva, cujos modelos não são apenas adaptativos, mas também evoluem suas funcionalidades e atualizam sua inteligência (LUGHOFFER, 2011).

As técnicas de modelagem evolutiva baseiam-se no conceito de Aprendizado Autônomo (*Autonomous Learning*) que é muito útil na prática, visto que informações dinâmicas disponíveis de processos ou plantas industriais são imprevisíveis (perturbações, incertezas e ruído) e/ou não lineares. Por este motivo, modelos evolutivos devem ser capazes de extrair rapidamente conhecimento desses ambientes (entradas e saídas) por meio de sensores em tempo real, de modo que essas técnicas frequentemente estão associadas à algoritmos que trabalham recursivamente, o que é apropriado para aplicações *on-line* (ANGELOV, 2013).

O conhecimento extraído pode ser representado de forma aproximada quando estruturas de modelos *fuzzy* são utilizados. Estes modelos são capazes de representar declarações vagas, incertezas e conhecimento aproximado através de um conjunto regras na forma **SE-antecedente-ENTÃO-consequente**, onde o número de regras, o tipo de antecedente e o tipo consequente são os seus principais parâmetros. O tipo de consequente, por sua vez, define as duas principais estruturas de modelos *fuzzy* encontradas na literatura: Mandani e Takagi-Sugeno (WANG, 1997).

Consequentemente, a combinação de técnicas de modelagem evolutiva com estruturas de modelos *fuzzy*, denominada modelagem *fuzzy* evolutiva, possui a habilidade de aumentar ou diminuir a quantidade de regras de acordo com alguma medida de qualidade, bem como a capacidade de adaptar seus parâmetros (LUGHOFFER, 2011). Um esquema ilustrativo, em diagrama de blocos, do funcionamento da modelagem *fuzzy* evolutiva é mostrado na figura 3. Este capítulo, por sua vez, dedica-se a apresentar os principais conceitos associados ao bloco modelo *fuzzy* evolutivo, apresentado no diagrama, a saber: estrutura de modelo *fuzzy* Takagi-Sugeno, técnicas para estimação do antecedente (agrupamento) e a estimação dos parâmetros do consequente.

Figura 3 – Estrutura geral da Modelagem *Fuzzy* Evolutiva.

Fonte: Autor.

3.1 Estrutura de Modelo *Fuzzy* Takagi-Sugeno

Dentre os diferentes tipos de modelos *fuzzy*, a estrutura TS (Takagi-Sugeno) está presente na maior parte das metodologias de modelagem evolutiva propostas na literatura (FILHO, 2017)(LUGHOFER et al., 2015)(LUGHOFER, 2008)(ANGELOV; KASABOV, 2005)(FILHO, 2011). Sua estrutura baseada em regras é definida como:

$$\text{Regra } i: \underbrace{\text{SE } \mathbf{z}(k) \text{ É } Z_i}_{\text{antecedente}} \underbrace{\text{ENTÃO } \mathbf{y}^i(k) = \Theta_i \mathbf{z}(k)}_{\text{consequente}} \quad (3.1)$$

onde $\mathbf{z}(k) = [z_1(k), z_2(k), \dots, z_l(k)]^T$ contém as variáveis linguísticas do antecedente e Θ_i é uma matriz de dimensões compatíveis. Cada variável linguística $z_j(k)$ é associada ao conjunto Z_i por uma função de pertinência $\mu_j^i(z_j(k)) \in [0,1]$, com $i = 1, 2, \dots, \tilde{c}$ e $j = 1, 2, \dots, l$. Cada regra, por sua vez, possui um valor numérico correspondente, chamado de grau de ativação $\beta^i(\mathbf{z}(k))$, que é expresso como:

$$\beta^i(\mathbf{z}(k)) = \mu_1^i(z_1(k)) \star \mu_2^i(z_2(k)) \star \dots \star \mu_l^i(z_l(k)) \quad (3.2)$$

sendo o operador $\star$ uma norma- t (WANG, 1997). Normalizando o valor de $\beta^i(\mathbf{z}(k))$, obtém-se:

$$\gamma^i(\mathbf{z}(k)) = \frac{\beta^i(\mathbf{z}(k))}{\sum_{i=1}^c \beta^i(\mathbf{z}(k))} \quad (3.3)$$

onde $\gamma^i(\mathbf{z}(k))$ é o grau de ativação normalizado ($\sum_{i=1}^c \gamma^i(\mathbf{z}(k)) = 1$). A saída do modelo

fuzzy TS, por sua vez, é dada por:

$$\tilde{\mathbf{y}}(k) = \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \mathbf{y}^i(k) = \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \Theta_i \mathbf{z}(k) \quad (3.4)$$

Modelos *fuzzy* TS tem como principal aplicação a representação de relações entrada-saída de processos com características complexas como, por exemplo, dinâmicas não lineares, através da ponderação de submodelos lineares locais (Θ_i) (FILHO, 2017). A identificação de sua estrutura consiste, basicamente, na determinação das variáveis do antecedente e consequente que pode ser feita de modo automático quando algoritmos especializados são utilizados para obter os conjuntos Z_j^i , a partir de um particionamento inteligente dos dados (agrupamento), e técnicas de identificação de sistemas são aplicadas na determinação de Θ_i (BABUŠKA, 1998). Estes algoritmos, por sua vez, serão analisadas em detalhes nas próximas seções.

3.2 Agrupamento Fuzzy

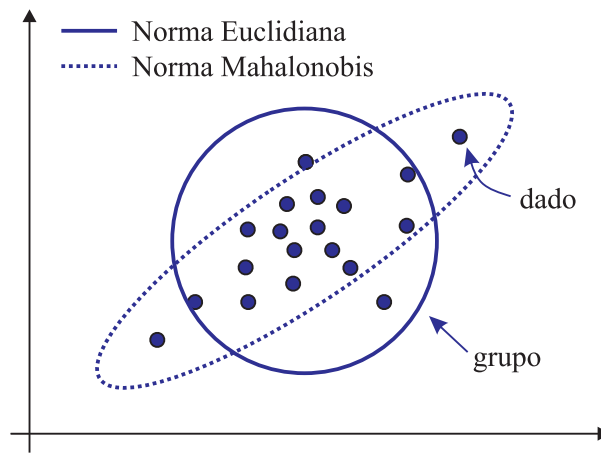
Agrupamento é uma técnica de aprendizagem de máquina não supervisionada, em que o espaço de dados (histórico de entradas e saídas de um processo, por exemplo) é particionado com base em algum padrão. A partir desse particionamento, os modelos podem ser construídos sem a intervenção humana, ou seja, de modo automático. Agrupar dados, por sua vez, consiste em encontrar uma distribuição ótima de grupos de modo que os dados pertencentes a um mesmo grupo possuam alta similaridade entre si e de grupos distintos não. Por este motivo, uma variável importante, principalmente para modelagem de processos complexos, é o número de grupos (ANGELOV, 2013).

Naturalmente, uma abordagem de agrupamento pode ser considerada adequada quando o número de grupos é determinado somente a partir da distribuição de dados, tornando até mesmo o ajuste dessa variável automático (evolutivo) (MACIEL; GOMIDE; BALLINI, 2012). Outras características importantes dos grupos são os pontos focais e os seus limites dentro do espaço de dados. O ponto focal de um grupo é o seu centro que, por sua vez, não precisa ser o centro espacial, mas pode ser um ponto escolhido para fazer esse papel (protótipo). Os limites, por outro lado, definem o formato dos grupos e podem ser hipercúbicos, hiperesféricos, hiper elipsoidais, dentre outros (ANGELOV, 2013)(AGGARWAL; REDDY, 2013).

Além disso, é necessário selecionar alguma medida de similaridade para que o espaço de dados seja particionado corretamente. Esta medida define o grau de homogeneidade (semelhança) das amostras pertencentes a um mesmo grupo e pode ser calculada de diferentes formas como, por exemplo, utilizando medidas de distância que podem ser a Euclidiana, Mahalonobis, Cosseno, etc. A distância (norma) Euclidiana, por exemplo,

produz grupos circulares ou hiperesféricos, enquanto que a distância (norma) Mahalonobis apresenta formato de elipse ou hiper elipsoidal, como pode ser visto na figura 4. De modo que a escolha do melhor formato está intimamente associado ao conhecimento que se possui acerca do processo a ser modelado (ANGELOV, 2013).

Figura 4 – Formato dos grupos criados a partir de um conjunto de dados usando-se diferentes métricas de distância: grupo demarcado pela linha sólida resultante de um agrupamento baseado na norma Euclidiana; grupo demarcado pela linha tracejada resultante de um agrupamento baseado na norma Mahalonobis.



Fonte: Adaptado de Angelov (2013).

Quando o agrupamento está inserido no contexto de modelagem *fuzzy*, cada grupo corresponde a um submodelo e, conseqüentemente, a uma regra. Como visto na equação 3.4, a saída ($\tilde{y}(k)$) de um modelo *fuzzy* TS é construída a partir de uma soma ponderada das saídas de cada submodelo ($y^i(k)$). Portanto, definir o número de grupos consiste, também, em determinar o tamanho da estrutura do modelo *fuzzy*, ou seja, o número de regras. Além disso, nessa abordagem, os submodelos não são mutuamente exclusivos, como ocorre em metodologias baseadas em agrupamento clássico (AGGARWAL; REDDY, 2013), mas sim cooperativos, permitindo que o modelo *fuzzy* resultante tolere a sobreposição de grupos (ANGELOV, 2013).

Agrupamentos *fuzzy* podem ser realizados em batelada ou de forma evolutiva, como será discutido nas próximas seções. Na abordagem batelada, todo o espaço de dados é particionado em um número fixo de grupos (regras) que, normalmente, é definido *a priori*. Na abordagem evolutiva, porém, o número de grupos (regras) e outras variáveis do agrupamento são modificados à medida que novas informações do processo são adquiridas.

3.2.1 Agrupamento *Fuzzy* em Batelada

Considerando um espaço de dados com N amostras coletadas de um processo que deseja-se modelar, obtém-se:

$$\mathbf{Z} = \begin{bmatrix} z_1(1) & z_1(2) & \dots & z_1(N) \\ z_2(1) & z_2(2) & \dots & z_2(N) \\ \vdots & \vdots & \ddots & \vdots \\ z_l(1) & z_l(2) & \dots & z_l(N) \end{bmatrix} \quad (3.5)$$

onde $\mathbf{Z}$ é chamada matriz de dados.

O objetivo do agrupamento *fuzzy* batelada é particionar $\mathbf{Z}$ em um número fixo de c grupos ou subconjuntos. Primeiramente, supõe-se que cada amostra $\mathbf{z}(n)$ de $\mathbf{Z}$ possui um valor de pertinência $\mu_{i,n}$ associado¹ ao i -ésimo grupo, com $n = 1, 2, \dots, N$ e $i = 1, 2, \dots, \tilde{c}$. Assim, a partição *fuzzy* de $\mathbf{Z}$ é definida como uma família de subconjuntos que possuem as seguintes propriedades:

$$\begin{aligned} \mu_{i,n} &\in [0,1], \quad 1 \leq i \leq \tilde{c}, \quad 1 \leq n \leq N \\ \sum_{i=1}^{\tilde{c}} \mu_{i,n} &= 1, \quad 1 \leq n \leq N \\ 0 < \sum_{n=1}^N \mu_{i,n} &< N, \quad 1 \leq i \leq \tilde{c} \end{aligned} \quad (3.6)$$

Algoritmos especializados em agrupamento *fuzzy* fazem uso destas propriedades e particionam o espaço de dados com base na minimização do seguinte índice:

$$J_a = \sum_{i=1}^{\tilde{c}} \sum_{n=1}^N (\mu_{i,n})^{m_f} D^2(i, n, \mathbf{A}_f^i) \quad (3.7)$$

onde $m_f \in [1, \infty)$ é chamado coeficiente de *fuzzificação* e $D(i, n, \mathbf{A}_f^i)$ é uma medida de distância (norma) expressa por:

$$D^2(i, n, \mathbf{A}_f^i) = \|\mathbf{z}(n) - \mathbf{z}^i\|_{\mathbf{A}_f^i}^2 = [\mathbf{z}(n) - \mathbf{z}^i]^T \mathbf{A}_f^i [\mathbf{z}(n) - \mathbf{z}^i] \quad (3.8)$$

sendo $\mathbf{z}^i$ e $\mathbf{A}_f^i$ o ponto focal e a matriz norma induzida do i -ésimo grupo, respectivamente, com $n = 1, 2, \dots, N$ e $i = 1, 2, \dots, \tilde{c}$. A minimização do índice J_f , por sua vez, consiste em um problema de otimização não linear, que é resolvido por diferentes técnicas de agrupamento *fuzzy* batelada.

A escolha da matriz $\mathbf{A}_f^i$, por outro lado, diferencia dois dos principais métodos de agrupamento *fuzzy* batelada encontrados na literatura. Considerando, por exemplo, $\mathbf{A}_f = \mathbf{I}^{(l)}$, a distância $D(i, n, \mathbf{I}^{(l)})$ é do tipo Euclidiana e o método de agrupamento *fuzzy*

¹ Não confundir o valor de pertinência $\mu_{i,n}$ com a função de pertinência $\mu_j^i(z_j(k))$.

batelada correspondente é chamado FCM (*Fuzzy c-means*) (TSAO; BEZDEK; PAL, 1994). No entanto, a matriz $\mathbf{A}_f^i$ pode ser escrita alternativamente como:

$$\mathbf{A}_f^i = [\rho^i \det(\Sigma_i)]^{1/l} \Sigma_i^{-1} \quad (3.9)$$

onde $\rho^i > 0$, chamado volume do i -ésimo grupo, e Σ_i , chamado matriz de covariância *fuzzy* do i -ésimo grupo, é definida como:

$$\Sigma_i = \frac{\sum_{n=1}^N (\mu_{i,n})^{m_f} [\mathbf{z}(n) - \mathbf{z}^i]^T [\mathbf{z}(n) - \mathbf{z}^i]}{\sum_{n=1}^N (\mu_{i,n})^{m_f}} \quad (3.10)$$

com $i = 1, 2, \dots, \tilde{c}$. Nesse caso a distância $D(i, n, \mathbf{A}_f^i)$ torna-se do tipo Mahalanobis e o método de agrupamento *fuzzy* batelada associado é chamado GK (*Gustafson-Kessel*) (GUSTAFSON; KESSEL, 1978).

Portanto, fica evidente que a melhor escolha para $D(i, n, \mathbf{A}_f^i)$ e, conseqüentemente, para o método de agrupamento *fuzzy* associado, depende muito do conhecimento sobre o processo a ser modelado. Isso, por sua vez, abre uma gama de possibilidades, pois, técnicas de agrupamento *fuzzy* baseadas em distância, como os já mencionados FCM, GK e o FMLE (*Fuzzy Maximum Likelihood Estimates Clustering*) (BEZDEK; DUNN, 1975), vêm sendo constantemente modificados e melhorados para se adequar a uma demanda cada vez maior por qualidade e desempenho (SOROOSH; KALHOR, 2014)(FILHO, 2017).

3.2.2 Agrupamento Fuzzy Evolutivo

Grande parte das técnicas de modelagem *fuzzy* evolutiva utilizam algum conceito referente à organização do espaço de dados. Isto, por sua vez, resulta na capacidade de modificar grupos existentes pelo aumento ou diminuição do seu número, ou seja, modificando a quantidade de regras (PEDRYCZ, 2005). Dentre os diferentes conceitos utilizados, destacam-se a densidade, o erro e a distância (SOROOSH; KALHOR, 2014). Os algoritmos de agrupamento *fuzzy* evolutivo baseados nesses conceitos serão apresentados a seguir.

3.2.2.1 Agrupamento Fuzzy Evolutivo Baseado em Densidade

A família eTS (*evolving Takagi-Sugeno*) são os mais notáveis exemplos de técnicas cujo agrupamento *fuzzy* evolutivo baseia-se no conceito de densidade. A primeira versão do eTS, proposta em Angelov e Filev (2004), utilizou o conceito de potencial (como medida de densidade) e uma versão *on-line* do então conhecido *Subtrative Clustering*. Posteriormente, em Angelov e Filev (2005), foi publicado uma versão simplificada desse

método, chamada SimpleTS (*Simplified Method for Learning Evolving Takagi-Sugeno*), que era computacionalmente mais eficiente, pois utilizava o conceito de *scatter* como medida de densidade. Em [Angelov e Zhou \(2006\)](#), porém, foi proposta uma versão estendida do eTS, chamada exTS (*Extended Evolving Takagi-Sugeno*) que apresentou uma fórmula para adaptação dos formatos (raios) e duas métricas para monitoramento da qualidade dos grupos, idade e suporte. Finalmente, uma nova medida de qualidade, chamada utilidade, foi introduzida no que veio a ser conhecido como eTS+ ([ANGELOV; ZHOU, 2008](#)).

3.2.2.2 Agrupamento Fuzzy Evolutivo Baseado em Erro

Estes métodos particionam o espaço de dados com base em uma função do erro. Um marco nessa área foi o desenvolvimento do AHLTMN (*Adaptive Habitually Linear and Nonlinear Model*), proposto por [Kalhor, Araabi e Lucas \(2010\)](#). Esta técnica intenciona manter o modelo o mais simples possível, no entanto, quando o erro de modelagem atinge certo valor limítrofe adaptativo, um novo grupo é criado temporariamente. Em [Kalhor, Araabi e Lucas \(2012\)](#) foi proposto um método sistemático para projeto do AHLTMN com base em três teoremas para garantir que o erro seja pequeno. Além desses, em [Kalhor, Araabi e Lucas \(2013\)](#) foi proposto um modelo *fuzzy* evolutivo que opera através do chaveamento de modelos vizinhos. Neste último, as operações de divisão e fusão de grupos são aplicados.

3.2.2.3 Agrupamento Fuzzy Evolutivo Baseado em Distância

Estes métodos particionam o espaço de dados com base em uma função da distância. De modo simples, se a distância de uma nova amostra de dado para todos os grupos for maior que um certo valor preestabelecido então um novo grupo é criado. Dois trabalhos de destaque que baseiam-se nessa ideia são o FLEXFIS (*Fuzzy Inference System*) ([LUGHO-FER, 2008](#)) e o DENFIS (*Dynamic Evolving Neural-Fuzzy Inference System*) ([KASABOV; SONG, 2002](#)). Além destes, porém, existem métodos que aprimoram as capacidades das técnicas de agrupamento *fuzzy* em batelada baseados em distância tornando-os evolutivos. Um exemplo é a versão evolutiva do algoritmo GK, proposto em [Soroosh e Kalhor \(2014\)](#), que, por sua vez, diferencia-se dos demais pela capacidade de criar grupos com formatos complexos (norma Mahalonobis), bem como por admitir diferentes operações entre grupos como, por exemplo, fusão e divisão.

3.3 Estimação dos Parâmetros dos Submodelos

Uma vez que os grupos estejam definidos em um instante de tempo discreto k , a parte do consequente dos modelos *fuzzy* TS (submodelos) precisa ser calculada. Visto que para fazê-lo é necessário conhecer as expressões para $\mathbf{y}^i(k)$ ou $\tilde{\mathbf{y}}(k)$, o aprendizado

desses parâmetros é chamado de supervisionado. Os submodelos, por outro lado, podem ser de diferentes tipos: polinomiais, ARX (*Auto Regressive with eXogenous input*), além de todos os modelos lineares mencionados na seção 2.1.1. Porém, independente do tipo de submodelo escolhido, é necessário optar entre as abordagens global e local para estimação desses parâmetros.

Na abordagem global os parâmetros dos submodelos são calculados simultaneamente pela utilização de $\tilde{\mathbf{y}}(k)$. Esta, porém, não é indicada para aplicações evolutivas, visto que as dimensões das matrizes utilizadas para o cálculo dos submodelos dependem do número de regras. A abordagem local, por outro lado, calcula cada submodelo individual e sequencialmente, por utilizar a expressão de $\mathbf{y}^i(k)$. Esta, conseqüentemente, possui maior flexibilidade quando a estrutura do modelo *fuzzy* TS é modificada durante o aumento ou diminuição de regras (LUGHOFFER, 2011). A partir dessa última e considerando um processo com p saídas que deseja-se modelar, deve-se minimizar o seguinte índice:

$$J_i = \sum_{n=1}^k \gamma^i(\mathbf{z}(n)) \mathbf{e}_i(n)^2 \quad (3.11)$$

onde $\mathbf{e}_i(n) = \mathbf{y}(n) - \mathbf{y}^i(n)$ representa o erro do i -ésimo submodelo linear, com $n = 1, 2, \dots, k$ e $i = 1, 2, \dots, c$. Substituindo a equação 3.1 na equação 3.11, obtém-se:

$$J_i = \sum_{n=1}^k \gamma^i(\mathbf{z}(n)) [\mathbf{y}(n) - \Theta_i \mathbf{z}(n)]^2 \quad (3.12)$$

Os parâmetros Θ_i obtidos através da minimização de J_i podem ser estimados de duas formas: batelada e recursiva. Como mencionado no início do capítulo 3, técnicas evolutivas frequentemente estão associadas à algoritmos que trabalham recursivamente. No entanto, geralmente é necessário inicializar procedimentos evolutivos e/ou recursivos com alguma estimação inicial em batelada. Considerando, por exemplo, uma batelada de N amostras de dados a estimação dos submodelos consiste em um problema, chamado FWLS (*Fuzzily Weighted Least Squares*), cuja solução é dada por:

$$\Theta_i = (\mathbf{ZQ}_i \vec{\mathbf{Y}} \mathbf{Q}_i^T \mathbf{Z}^T) (\mathbf{ZQ}_i \mathbf{Q}_i^T \mathbf{Z}^T)^{-1} \quad (3.13)$$

com $i = 1, 2, \dots, c$ e

$$\mathbf{Q}_i = \begin{bmatrix} \gamma^i(\mathbf{z}(1)) & 0 & \dots & 0 \\ 0 & \gamma^i(\mathbf{z}(2)) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \gamma^i(\mathbf{z}(N)) \end{bmatrix} \quad \vec{\mathbf{Y}} = \begin{bmatrix} \mathbf{y}(1)^T \\ \mathbf{y}(2)^T \\ \vdots \\ \mathbf{y}(N)^T \end{bmatrix}$$

Alternativamente, os parâmetros dos submodelos Θ_i podem ser atualizados recursivamente pela solução do RFWLS (*Recursive Fuzzily Weighted Least Squares*) dado por:

$$\Theta_i(k+1) = \Theta_i(k) + \kappa(k) [\mathbf{y}(k+1) - \mathbf{z}(k+1)^T \Theta_i(k)] \quad (3.14)$$

$$\kappa(k) = \mathbf{P}_i(k+1) \mathbf{z}(k+1) = \frac{\mathbf{P}_i(k) \mathbf{z}(k+1)}{\frac{1}{\gamma^i(\mathbf{z}(k+1))} + \mathbf{z}(k+1)^T \mathbf{P}_i(k) \mathbf{z}(k+1)} \quad (3.15)$$

$$\mathbf{P}_i(k+1) = [\mathbf{I}^{(l)} - \kappa(k) \mathbf{z}(k+1)^T] \mathbf{P}_i(k) \quad (3.16)$$

onde $\Theta_i(k)$ é a matriz de parâmetros do i -ésimo submodelo estimado no instante k e $\mathbf{P}_i(k)$ é chamada matriz de covariância do erro do i -ésimo submodelo também estimada no instante de tempo k (LUGHOFER, 2011).

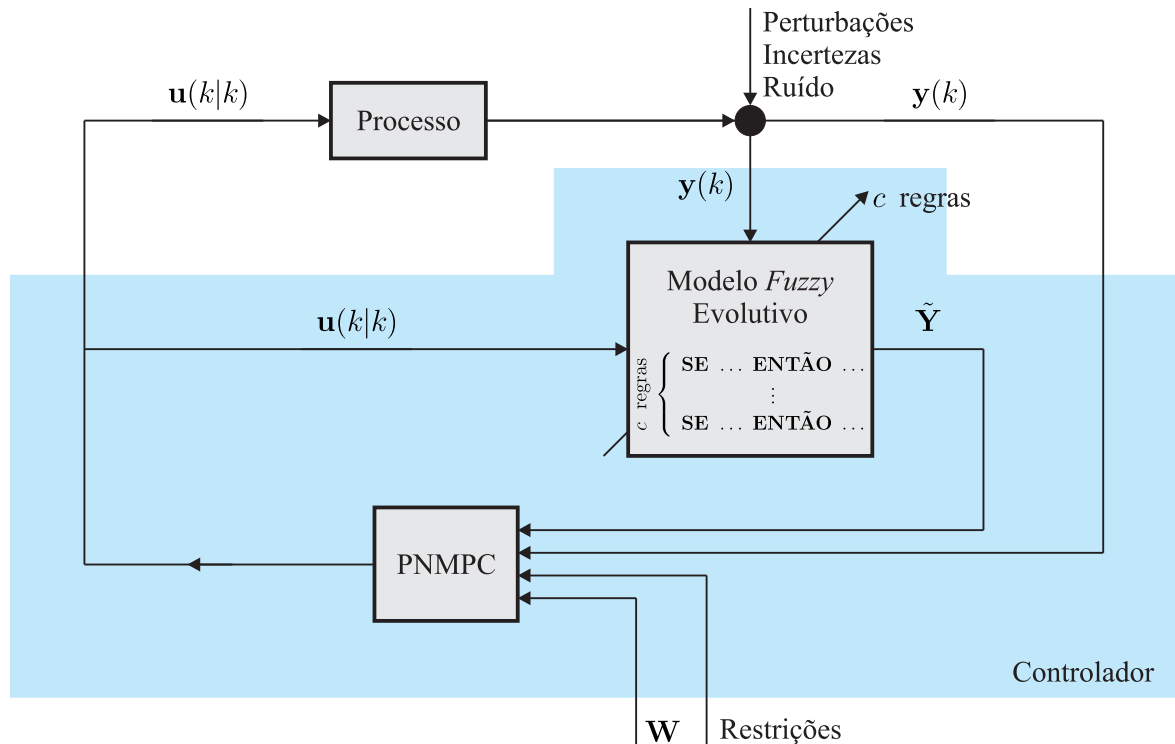
3.4 Conclusões

Neste capítulo discutiu-se a modelagem *fuzzy* evolutiva sob os seguintes aspectos: estrutura de modelo *fuzzy* TS, estimação dos parâmetros do antecedente e consequente. Para estimar os parâmetros do antecedente, apresentaram-se técnicas para o particionamento inteligente dos dados na forma de algoritmos de agrupamento de duas formas: batelada e evolutiva. Na abordagem batelada, salientou-se a necessidade de escolher uma boa medida de similaridade entre os dados, enquanto que na abordagem evolutiva destacaram-se os diferentes conceitos utilizados para criação de grupos, a saber: densidade, erro e distância. Finalmente, técnicas para estimação dos parâmetros do consequente foram apresentadas nas formas batelada e recursiva.

4 Metodologia de Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo

A estrutura geral da metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo proposta nesta dissertação é apresentada na figura 5 onde se observa seu funcionamento: a partir de informações dinâmicas das entradas ($\mathbf{u}(k|k)$) e saídas ($\mathbf{y}(k)$) de um processo MIMO não linear, obtidas através de sensores em tempo real, atualizam-se a estrutura e os parâmetros do modelo *fuzzy* evolutivo; o modelo *fuzzy* evolutivo, por sua vez, é utilizado para calcular as previsões das saídas $\tilde{\mathbf{Y}}$; essas previsões, uma trajetória de referência ($\mathbf{W}$) e um conjunto de restrições são utilizadas pelo algoritmo PNMPC para calcular o próximo sinal de controle a ser aplicado ao processo.

Figura 5 – Estrutura geral da Metodologia de Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo proposta: a região delimitada em azul representa a estrutura do controlador proposto.



Fonte: Autor

No entanto, para garantir que o controlador proposto (região delimitada em azul na figura 5) não apresente desempenho inicial insatisfatório (instabilidade, por exemplo), um conjunto de dados em batelada é utilizado para estimar os parâmetros iniciais do modelo *fuzzy* em uma etapa de treinamento *off-line*. Posteriormente, o controlador entrará

no modo *on-line* onde cumprirá dois objetivos principais: calcular o sinal de controle a partir do modelo *fuzzy* evolutivo utilizando o PNMPC; atualizar os parâmetros e estrutura do modelo *fuzzy* evolutivo. Assim, este capítulo inicia com a seção 4.1, que descreve um procedimento em batelada para estimar o modelo *fuzzy* inicial. Em seguida, na seção 4.2, detalha-se a aplicação do algoritmo PNMPC para obter o sinal de controle a partir de um modelo *fuzzy* evolutivo. Finalmente, na seção 4.3, é descrito todo o mecanismo para atualização dos parâmetros e estrutura do modelo *fuzzy* evolutivo.

4.1 Etapa de Treinamento

A etapa de treinamento consiste na estimação dos parâmetros do modelo *fuzzy*. Para obtê-lo, portanto, é realizado um procedimento em batelada para calcular os parâmetros do antecedente e consequente. Considerando um processo MIMO não linear com p saídas e m entradas a estrutura da regra *fuzzy* TS, utilizada nesse trabalho, possui a seguinte forma:

$$\text{Regra } i: \text{SE } \underbrace{\mathbf{z}(k) \text{ É } Z_i}_{\text{antecedente}} \text{ ENTÃO } \underbrace{\mathbf{y}^i(k) = \Theta_i \mathbf{z}(k)}_{\text{consequente}} \quad (4.1)$$

onde $\mathbf{z}(k) = [\mathbf{u}(k-1)^T, \dots, \mathbf{u}(k-n_u)^T, \mathbf{y}(k-1)^T, \dots, \mathbf{y}(k-n_y)^T]^T \in \mathbb{R}^{l \times 1}$ contém as variáveis linguísticas do antecedente¹, $\Theta_i \in \mathbb{R}^{p \times l}$ é a matriz de parâmetros do i -ésimo submodelo, $\mathbf{y}^i(k) \in \mathbb{R}^{p \times 1}$ é o vetor de saída do i -ésimo submodelo, $\mathbf{u}(k) \in \mathbb{R}^{m \times 1}$ é o vetor de entrada, com $i = 1, 2, \dots, \tilde{c}$ e $l = m.n_u + p.n_y$. A matriz de dados, formada a partir de um conjunto em batelada com N amostras, é expressa como:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{u}(0) & \mathbf{u}(1) & \dots & \mathbf{u}(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{u}(-n_u) & \mathbf{u}(-n_u+1) & \dots & \mathbf{u}(N-n_u) \\ \mathbf{y}(0) & \mathbf{y}(1) & \dots & \mathbf{y}(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}(-n_y) & \mathbf{y}(-n_y+1) & \dots & \mathbf{y}(N-n_y) \end{bmatrix} \quad (4.2)$$

sendo $\mathbf{Z} \in \mathbb{R}^{m.n_u + p.n_y \times N}$.

4.1.1 Estimação em Batelada dos Parâmetros do Antecedente

Os parâmetros do antecedente são estimados pelo Algoritmo de Agrupamento *fuzzy* Batelada GK. Este, por sua vez, particiona a matriz $\mathbf{Z}$ através do seguinte procedimento iterativo (BABUŠKA, 1998):

¹ n_u e n_y são os maiores atrasos da saída e entrada, respectivamente.

1. Escolher um número fixo de c grupos a serem criados, inicializar os centros $\mathbf{z}^i$ e as matrizes norma induzida $\mathbf{A}_f^i$, com $i = 1, 2, \dots, \tilde{c}$.
2. Calcular a distância $D(i, n, \mathbf{A}_f^i)$ para todos os pontos usando:

$$D^2(i, n, \mathbf{A}_f^i) = \|\mathbf{z}(n) - \mathbf{z}^i\|_{\mathbf{A}_f^i}^2 = [\mathbf{z}(n) - \mathbf{z}^i]^T \mathbf{A}_f^i [\mathbf{z}(n) - \mathbf{z}^i] \quad (4.3)$$

onde $n = 1, 2, \dots, N$ e $i = 1, 2, \dots, \tilde{c}$.

3. Calcular o valor de pertinência para todas as amostras de dados usando:

$$\mu_{i,n} = \frac{\left[\frac{1}{D(i,n,\mathbf{A}_f^i)} \right]^{\frac{2}{m_f-1}}}{\sum_{j=1}^c \left[\frac{1}{D(j,n,\mathbf{A}_f^j)} \right]^{\frac{2}{m_f-1}}} \quad (4.4)$$

onde $n = 1, 2, \dots, N$ e $i = 1, 2, \dots, \tilde{c}$ e m_f é o coeficiente de *fuzzificação*.

4. Calcular os centros dos grupos usando:

$$\mathbf{z}^i = \frac{\sum_{n=1}^N (\mu_{i,n})^{m_f} \mathbf{z}(n)}{\sum_{n=1}^N (\mu_{i,n})^{m_f}} \quad (4.5)$$

onde $\mathbf{z}^i \in \mathbb{R}^{l \times 1}$, com $i = 1, 2, \dots, \tilde{c}$.

5. Calcular a matriz de covariância *fuzzy* Σ_i e a matriz norma induzida $\mathbf{A}_f^i$ usando:

$$\Sigma_i = \frac{\sum_{n=1}^N (\mu_{i,n})^{m_f} [\mathbf{z}(n) - \mathbf{z}^i]^T [\mathbf{z}(n) - \mathbf{z}^i]}{\sum_{n=1}^N (\mu_{i,n})^{m_f}} \quad (4.6)$$

$$\mathbf{A}_f^i = [\rho^i \det(\Sigma_i)]^{1/l} \Sigma_i^{-1} \quad (4.7)$$

onde $\Sigma_i \in \mathbb{R}^{l \times l}$, $\mathbf{A}_f^i \in \mathbb{R}^{l \times l}$ e ρ^i é o volume do i -ésimo grupo, com $i = 1, 2, \dots, \tilde{c}$.

6. Voltar para o passo 2 até que algum critério seja satisfeito.

4.1.2 Estimação em Batelada dos Parâmetros do Consequente

Ao término do Algoritmo de Agrupamento *fuzzy* GK, os centros $\mathbf{z}^i$ e as matrizes norma induzida $\mathbf{A}_f^i$ são utilizados para calcular o grau de ativação normalizado (equação 3.3) de uma amostra $\mathbf{z}(n)$ da seguinte forma:

$$\gamma^i(\mathbf{z}(n)) = \frac{\left[\frac{1}{D(i,n,\mathbf{A}_f^i)} \right]^{\frac{2}{m_f-1}}}{\sum_{j=1}^c \left[\frac{1}{D(j,n,\mathbf{A}_f^j)} \right]^{\frac{2}{m_f-1}}} \quad (4.8)$$

com $i = 1, 2, \dots, \tilde{c}$. Desse modo, os parâmetros do consequente são estimados em batelada utilizando o FWLS (ver seção 3.3), cuja solução é:

$$\Theta_i = (\mathbf{ZQ}_i \vec{\mathbf{Y}} \mathbf{Q}_i^T \mathbf{Z}^T) (\mathbf{ZQ}_i \mathbf{Q}_i^T \mathbf{Z}^T)^{-1} \quad (4.9)$$

onde

$$\mathbf{Q}_i = \begin{bmatrix} \gamma^i(\mathbf{z}(1)) & 0 & \dots & 0 \\ 0 & \gamma^i(\mathbf{z}(2)) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \gamma^i(\mathbf{z}(N)) \end{bmatrix} \in \mathbb{R}^{N \times N} \quad (4.10)$$

$$\vec{\mathbf{Y}} = \begin{bmatrix} \mathbf{y}(1)^T \\ \mathbf{y}(2)^T \\ \vdots \\ \mathbf{y}(N)^T \end{bmatrix} \in \mathbb{R}^{N \times p} \quad (4.11)$$

com $i = 1, 2, \dots, c$. O procedimento para estimar o modelo *fuzzy* TS inicial apresentado nesta seção é detalhado no Algoritmo 1.

4.2 Etapa de Controle

Ao passo que o controlador proposto baseia-se em um modelo *fuzzy* evolutivo, onde os parâmetros e estrutura são continuamente modificados, reescreve-se a equação 3.4, do seguinte modo:

$$\tilde{\mathbf{y}}(k) = \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \mathbf{y}^i(k) = \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \Theta_i(k) \mathbf{z}(k) \quad (4.12)$$

Algoritmo 1 Etapa de Treinamento

```

1: Variáveis de Entrada
2: Indicar valores para  $n_u, n_y, \tilde{c}, \mathbf{z}^i, \mathbf{A}_f^i, m_f$  e  $\rho^i$ , com  $i = 1, 2, \dots, \tilde{c}$ ;
3: Estimação em Batelada dos Parâmetros do Antecedente
4: enquanto Algum critério de parada não for satisfeito faça
5:   para  $n \leftarrow 1$  até  $N$  faça
6:     para  $i \leftarrow 1$  até  $\tilde{c}$  faça
7:       Calcular  $D(i, n, \mathbf{A}_f^i)$  (equação 4.3);
8:     fim para
9:   fim para
10:  para  $n \leftarrow 1$  até  $N$  faça
11:    para  $i \leftarrow 1$  até  $\tilde{c}$  faça
12:      Calcular  $\mu_{i,n}$  (equação 4.4);
13:    fim para
14:  fim para
15:  para  $i \leftarrow 1$  até  $\tilde{c}$  faça
16:    Calcular  $\mathbf{z}^i$  (equação 4.5);
17:  fim para
18:  para  $i \leftarrow 1$  até  $\tilde{c}$  faça
19:    Calcular  $\Sigma_i$  (equação 4.6);
20:    Calcular  $\mathbf{A}_f^i$  (equação 4.7);
21:  fim para
22: fim enquanto
23: Estimação em Batelada dos Parâmetros do Consequente
24: Construir  $\tilde{\mathbf{Y}}$  (equação 4.11);
25: para  $i \leftarrow 1$  até  $\tilde{c}$  faça
26:   para  $n \leftarrow 1$  até  $N$  faça
27:     Calcular  $\gamma^i(\mathbf{z}(n))$  (equação 4.8);
28:   fim para
29:   Construir  $\mathbf{Q}_i$  (equação 4.10);
30:   Calcular  $\Theta_i$  (equação 4.9);
31: fim para

```

onde $\tilde{c}$ é o número de grupos e $\Theta_i(k)$ é a matriz de parâmetros do i -ésimo submodelo calculada no instante k , com $i = 1, 2, \dots, \tilde{c}$. A seguir, serão descritos os procedimentos para obtenção do sinal de controle através da técnica PNMPC.

4.2.1 Preparação do Modelo de Predição

Diferente da aproximação usada no PNMPC (equação 2.39), neste trabalho as predições não baseiam-se em um modelo não linear fixo. Por esta razão, o vetor de predições da saída é reescrito como:

$$\tilde{\mathbf{Y}}(k) \approx \tilde{\Gamma}(k) + \tilde{\mathbf{G}}(k)\mathbf{U}_f \quad (4.13)$$

Além disso, a fim de suprimir os efeitos de erros de modelagem, bem como pertur-

bações na saída, adiciona-se um modelo de perturbações $\frac{\mathbf{e}(k)}{\Delta}$ à saída $\tilde{\mathbf{y}}(k)$ da equação 4.12. Desse modo, tem-se que:

$$\tilde{\mathbf{y}}(k) = \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \Theta_i(k) \mathbf{z}(k) + \frac{\mathbf{e}(k)}{\Delta} \quad (4.14)$$

onde $\mathbf{e}(k) \in \mathbb{R}^{p \times 1}$ é um vetor de ruído branco e média nula. Fazendo manipulações algébricas na equação 4.14, obtém-se:

$$\tilde{\mathbf{y}}(k) = \tilde{\mathbf{y}}(k-1) + \sum_{i=1}^{\tilde{c}} \gamma^i(\mathbf{z}(k)) \Theta_i(k) \Delta \mathbf{z}(k) + \mathbf{e}(k) \quad (4.15)$$

sendo $\Delta \mathbf{z}(k) = [\Delta \mathbf{u}(k-1)^T, \dots, \Delta \mathbf{u}(k-n_u)^T, \Delta \mathbf{y}(k-1)^T, \dots, \Delta \mathbf{y}(k-n_y)^T]^T \in \mathbb{R}^{1 \times l}$. Para calcular as predições da saída j passos a frente utiliza-se o valor esperado da variável $\tilde{\mathbf{y}}(k+j|k)$, que possui melhor valor, no sentido estocástico, quando o ruído futuro ($\mathbf{e}(k+j)$) é nulo (AGUIRRE, 2007), ou seja,

$$\begin{aligned} \hat{\mathbf{y}}(k+j|k) &= \hat{\mathbf{y}}(k+j-1|k) + \sum_{i=1}^{\tilde{c}} \gamma^i(\hat{\mathbf{z}}(k+j)) \Theta_i(k) \Delta \hat{\mathbf{z}}(k+j) \\ &= \tilde{f}(k, \hat{\mathbf{z}}(k+j|k), \Delta \hat{\mathbf{z}}(k+j|k)) \end{aligned} \quad (4.16)$$

onde $\tilde{f}(\dots)$ é uma função não linear (contínua e diferenciável) que representa o modelo de predição *fuzzy* TS evolutivo. Este modelo, por sua vez, será utilizado para estimar numericamente as matrizes $\tilde{\Gamma}(k)$ e $\tilde{\mathbf{G}}(k)$ seguindo o método descrito na seção 2.2.4.

4.2.2 Estimação das Matrizes de Predição

Considerando um horizonte de predição inicial $N_1 = 1$, final N_2 e de controle N_u , estima-se a matriz $\tilde{\Gamma}(k)$ por simular o comportamento do modelo (equação 4.16) ao longo do horizonte de predição quando nenhum incremento futuro de controle é aplicado ($\mathbf{U}_f = 0$), ou seja,

$$\tilde{\Gamma}(k) = \begin{bmatrix} \hat{\mathbf{y}}_l(k+1|k) \\ \hat{\mathbf{y}}_l(k+2|k) \\ \vdots \\ \hat{\mathbf{y}}_l(k+N_2|k) \end{bmatrix} \in \mathbb{R}^{p \cdot N_2 \times 1} \quad (4.17)$$

onde

$$\hat{\mathbf{y}}_l(k+j|k) = \tilde{f}(k, \hat{\mathbf{z}}_l(k+j|k), \Delta \hat{\mathbf{z}}_l(k+j|k)) \quad (4.18)$$

com

$$\begin{aligned}\hat{\mathbf{z}}_l(k+j|k)^T &= [\mathbf{u}_l(k+j-1)^T, \dots, \mathbf{u}_l(k+j-n_u)^T, \hat{\mathbf{y}}_l(k+j-1)^T, \dots, \hat{\mathbf{y}}_l(k+j-n_y)^T] \\ \Delta \hat{\mathbf{z}}_l(k+j|k)^T &= [\Delta \mathbf{u}_l(k+j-1)^T, \dots, \Delta \mathbf{u}_l(k+j-n_u)^T, \Delta \hat{\mathbf{y}}_l(k+j-1)^T, \dots, \Delta \hat{\mathbf{y}}_l(k+j-n_y)^T]\end{aligned}\quad (4.19)$$

desde que

$$\Delta \mathbf{u}_l(k+n) = \begin{cases} \Delta \mathbf{u}(k+n), & \forall n < 0 \\ 0, & \forall n \geq 0 \end{cases} \quad (4.20)$$

$$\hat{\mathbf{y}}_l(k+n|k) = \mathbf{y}(k+n|k) \quad \forall n < 0 \quad (4.21)$$

com $j = 1, 2, \dots, N_2$.

Para calcular a matriz $\tilde{\mathbf{G}}(k) \in \mathbb{R}^{p.N_2 \times m.N_u}$ utiliza-se o procedimento iterativo descrito na seção 2.2.4. Este, por sua vez, inicia com a subdivisão de $\tilde{\mathbf{G}}(k)$ como:

$$\tilde{\mathbf{G}}(k) = [\tilde{\mathbf{G}}_1(k), \dots, \tilde{\mathbf{G}}_{N_u}(k)] \quad (4.22)$$

onde cada matriz $\tilde{\mathbf{G}}_{i_1}(k) \in \mathbb{R}^{p.N_2 \times m}$ também pode ser subdividida como:

$$\tilde{\mathbf{G}}_{i_1}(k) = [\tilde{\mathbf{G}}_{i_1}^1(k), \dots, \tilde{\mathbf{G}}_{i_1}^m(k)] \quad (4.23)$$

sendo cada coluna $\tilde{\mathbf{G}}_{i_1}^{i_2}(k) \in \mathbb{R}^{p.N_2 \times 1}$ calculada por:

$$\tilde{\mathbf{G}}_{i_1}^{i_2}(k) = \frac{\tilde{\mathbf{Y}}_f(\epsilon_{(i_1-1), i_2}) - \tilde{\Gamma}(k)}{\epsilon_{(i_1-1), i_2}} \quad (4.24)$$

onde

$$\tilde{\mathbf{Y}}_f(\epsilon_{(i_1-1), i_2}) = \begin{bmatrix} \hat{\mathbf{y}}_f(k+1|k) \\ \hat{\mathbf{y}}_f(k+2|k) \\ \vdots \\ \hat{\mathbf{y}}_f(k+N_2|k) \end{bmatrix} \in \mathbb{R}^{p.N_2 \times 1} \quad (4.25)$$

é o vetor de predições da saída quando um pequeno incremento de controle $\epsilon_{(i_1-1), i_2}$ é aplicado no instante $k+i_1-1$ na i_2 -ésima entrada e

$$\hat{\mathbf{y}}_f(k+j|k) = \tilde{f}(k, \hat{\mathbf{z}}_f(k+j|k), \Delta \hat{\mathbf{z}}_f(k+j|k)) \quad (4.26)$$

sendo

$$\begin{aligned}\hat{\mathbf{z}}_f(k+j|k)^T &= [\mathbf{u}_f(k+j-1)^T, \dots, \mathbf{u}_f(k+j-n_u)^T, \hat{\mathbf{y}}_f(k+j-1)^T, \dots, \hat{\mathbf{y}}_f(k+j-n_y)^T] \\ \Delta \hat{\mathbf{z}}_f(k+j|k)^T &= [\Delta \mathbf{u}_f(k+j-1)^T, \dots, \Delta \mathbf{u}_f(k+j-n_u)^T, \Delta \hat{\mathbf{y}}_f(k+j-1)^T, \dots, \Delta \hat{\mathbf{y}}_f(k+j-n_y)^T]\end{aligned}\quad (4.27)$$

onde $\Delta \mathbf{u}_f(k+j) = [\Delta u_1^f(k+j) \dots \Delta u_{i_2}^f(k+j) \dots \Delta u_m^f(k+j)]^T \in \mathbb{R}^{m \times 1}$ e

$$\Delta u_{n_2}^f(k+n_1) = \begin{cases} \epsilon_{(i_1-1), i_2}, & \text{se } n_1 = i_1 - 1 \text{ e } n_2 = i_2 \\ 0, & \text{caso contrário} \end{cases} \quad (4.28)$$

$$\hat{\mathbf{y}}_f(k+n_1|k) = \mathbf{y}(k+n_1|k) \quad \forall n < 0 \quad (4.29)$$

com $i_1 = 1, 2, \dots, N_u$, $i_2 = 1, 2, \dots, m$, $j = 1, 2, \dots, N_2$.

4.2.3 Obtenção do Sinal de Controle

Uma vez que as matrizes $\tilde{\Gamma}(k)$ e $\tilde{\mathbf{G}}(k)$ estejam calculadas no instante k , o problema de controle pode ser formulado como um QP que consiste na minimização de um índice dado por:

$$J = \mathbf{U}_f^T [-\tilde{\mathbf{G}}(k)^T \mathbf{Q}_\delta \mathbf{W} + 2\tilde{\mathbf{G}}(k)^T \mathbf{Q}_\delta \tilde{\Gamma}(k)] \mathbf{U}_f + \mathbf{U}_f^T [\mathbf{G}(k)^T \mathbf{Q}_\delta \mathbf{G}(k) + \mathbf{Q}_\lambda] \quad (4.30)$$

onde $\mathbf{W} \in \mathbb{R}^{p \cdot N_2 \times 1}$ é o vetor de referências futuras, $\mathbf{Q}_\delta \in \mathbb{R}^{p \cdot N_2 \times p \cdot N_2}$ é a matriz de ponderação do erro e $\mathbf{Q}_\lambda \in \mathbb{R}^{m \cdot N_u \times m \cdot N_u}$ é a matriz de ponderação do esforço de controle.

Além disso, a minimização do índice J (equação 4.30) está sujeita a um conjunto de restrições da forma:

$$\begin{aligned} \mathbf{u}_{\min} &\leq \mathbf{u}(k+j|k) \leq \mathbf{u}_{\max}, & \forall j = 0, 1, \dots, N_u - 1. \\ \mathbf{du}_{\min} &\leq \Delta \mathbf{u}(k+j|k) \leq \mathbf{du}_{\max}, & \forall j = 0, 1, \dots, N_u - 1. \\ \mathbf{y}_{\min} &\leq \hat{\mathbf{y}}(k+j|k) \leq \mathbf{y}_{\max}, & \forall j = 1, 2, \dots, N_2. \end{aligned} \quad (4.31)$$

onde $\mathbf{u}_{\min} \in \mathbb{R}^{m \times 1}$, $\mathbf{u}_{\max} \in \mathbb{R}^{m \times 1}$, $\mathbf{du}_{\min} \in \mathbb{R}^{m \times 1}$, $\mathbf{du}_{\max} \in \mathbb{R}^{m \times 1}$, $\mathbf{y}_{\min} \in \mathbb{R}^{p \times 1}$ e $\mathbf{y}_{\max} \in \mathbb{R}^{p \times 1}$ são os valores máximo e mínimo do controle, esforço de controle e saída, respectivamente. Finalmente, uma vez que sequência de incrementos de controles futuros $\mathbf{U}_f$ é obtida, o sinal de controle aplicado ao processo é calculado como:

$$\mathbf{u}(k|k) = \mathbf{u}(k-1) + \Delta \mathbf{u}(k|k) \quad (4.32)$$

Todo o procedimento descrito nesta seção é detalhado no Algoritmo 2.

4.3 Etapa de Atualização do Modelo Fuzzy Evolutivo

Neste trabalho utilizou-se uma versão modificada do Algoritmo de Agrupamento fuzzy GK evolutivo, proposto em Soroosh e Kalhor (2014), em conjunto com o RFWLS

Algoritmo 2 Etapa de Controle

-
- 1: **Variáveis de Entrada**
 - 2: Indicar valores para N_2 e N_u ;
 - 3: Indicar valor do vetor de referências futuras $\mathbf{W}$;
 - 4: Indicar valor para a matriz de ponderação do erro $\mathbf{Q}_\delta$;
 - 5: Indicar valor para a matriz de ponderação do esforço de controle $\mathbf{Q}_\lambda$;
 - 6: Indicar valores para as restrições $\mathbf{u}_{\min}$, $\mathbf{u}_{\max}$, $\mathbf{du}_{\min}$, $\mathbf{du}_{\max}$, $\mathbf{y}_{\min}$ e $\mathbf{y}_{\max}$ (equação 4.31);
 - 7: Escolher $\epsilon_{(i_1-1),i_2} = 10^{-2} \sim 10^{-5} \forall i_1, i_2$;
 - 8: **Preparação do Modelo de Predição**
 - 9: Formular $\hat{\mathbf{y}}(k+j|k)$ (equação 4.16);
 - 10: **Estimação das Matrizes de Predição**
 - 11: **para** $j \leftarrow 1$ **até** N_2 **faça**
 - 12: Calcular $\hat{\mathbf{z}}_l(k+j|k)$ e $\Delta\hat{\mathbf{z}}_l(k+j|k)$ (equação 4.19);
 - 13: Calcular $\hat{\mathbf{y}}_l(k+j|k)$ (equação 4.18);
 - 14: **fim para**
 - 15: Construir $\tilde{\Gamma}(k)$ (equação 4.17);
 - 16: **para** $i_1 \leftarrow 1$ **até** N_u **faça**
 - 17: **para** $i_2 \leftarrow 1$ **até** m **faça**
 - 18: **para** $j \leftarrow 1$ **até** N_2 **faça**
 - 19: Calcular $\hat{\mathbf{z}}_f(k+j|k)$ e $\Delta\hat{\mathbf{z}}_f(k+j|k)$ (equação 4.27);
 - 20: Calcular $\hat{\mathbf{y}}_f(k+j|k)$ (equação 4.26);
 - 21: **fim para**
 - 22: Construir $\tilde{\mathbf{Y}}_f(\epsilon_{(i_1-1),i_2})$ (equação 4.25);
 - 23: Calcular $\tilde{\mathbf{G}}_{i_1}^{i_2}(k)$ (equação 4.24);
 - 24: **fim para**
 - 25: Construir $\tilde{\mathbf{G}}_{i_1}(k)$ (equação 4.23);
 - 26: **fim para**
 - 27: Construir $\tilde{\mathbf{G}}(k)$ (equação 4.22);
 - 28: **Obtenção do Sinal de Controle**
 - 29: Calcular $\mathbf{U}_f$ (equação 4.30);
 - 30: Calcular $\mathbf{u}(k|k)$ (equação 4.32);
-

para atualizar os parâmetros e estrutura do modelo *fuzzy* TS evolutivo. Este método supera as limitações do Algoritmo de Agrupamento *fuzzy* Batelada GK por atualizar recursivamente os parâmetros do antecedente e utilizar uma medida de similaridade para criação e fusão de grupos.

4.3.1 Atualização Recursiva dos Parâmetros do Antecedente

Os parâmetros do Agrupamento *fuzzy* GK, por sua vez, são atualizados recursivamente com base na estratégia proposta por Dovžan e Škrjanc (2011) de acordo com a solução do seguinte conjunto de equações:

$$\mathbf{z}^i(k+1) = \mathbf{z}^i(k) + \frac{\gamma^i(\mathbf{z}(k+1))^{m_f}(\mathbf{z}(k+1) - \mathbf{z}^i(k))}{N_i(k+1)} \quad (4.33)$$

$$\Sigma_i(k+1) = \frac{N_i(k)}{N_i(k+1)} \left\{ \Sigma_i(k) + \frac{\gamma^i(\mathbf{z}(k+1))^{m_f}}{N_i(k+1)} [\mathbf{z}(k+1) - \mathbf{z}^i(k)] [\mathbf{z}(k+1) - \mathbf{z}^i(k)]^T \right\} \quad (4.34)$$

onde $\mathbf{z}^i(k)$ é o i -ésimo centro do grupo calculado no instante k , $\Sigma_i(k)$ é a i -ésima matriz de covariância *fuzzy* do grupo calculada no instante k e

$$N_i(k+1) = N_i(k) + \gamma^i(\mathbf{z}(k+1))^{m_f} \quad (4.35)$$

com $i = 1, 2, \dots, \tilde{c}$. Para calcular as matrizes norma induzida, é necessário computar a inversa e o determinante da matriz de covariância *fuzzy* (Σ_i) recursivamente. Para isso, utiliza-se a identidade de Woodbury e o lema do determinante (PETERSEN; PEDERSEN, 2012) como:

$$\Sigma_i^{-1}(k+1) = \frac{N_i(k+1)}{N_i(k)} \left\{ \Sigma_i^{-1}(k) - \frac{\Sigma_i^{-1}(k) [\mathbf{z}(k+1) - \mathbf{z}^i(k)] [\mathbf{z}(k+1) - \mathbf{z}^i(k)]^T \Sigma_i^{-1}(k)}{1 + \frac{N_i(k+1)}{\gamma^i(\mathbf{z}(k))^{m_f}} [\mathbf{z}(k+1) - \mathbf{z}^i(k)] \Sigma_i^{-1}(k) [\mathbf{z}(k+1) - \mathbf{z}^i(k)]} \right\} \quad (4.36)$$

$$\det(\Sigma_i(k+1)) = \left[\frac{N_i(k)}{N_i(k+1)} \right]^l \left\{ 1 + \frac{\gamma^i(\mathbf{z}(k+1))^{m_f}}{N_i(k+1)} [\mathbf{z}(k+1) - \mathbf{z}^i(k)] \Sigma_i^{-1}(k) [\mathbf{z}(k+1) - \mathbf{z}^i(k)] \right\} \quad (4.37)$$

onde $\Sigma_i^{-1}(k)$ e $\det(\Sigma_i(k))$ são a i -ésima inversa da matriz de covariância *fuzzy* e o i -ésimo determinante da matriz de covariância *fuzzy*, respectivamente, calculados no instante k e usados para atualizar a matriz norma induzida por:

$$\mathbf{A}_f^i(k+1) = \sqrt[l]{\rho^i \det(\Sigma_i(k+1))} \Sigma_i^{-1}(k+1) \quad (4.38)$$

com $i = 1, 2, \dots, \tilde{c}$.

4.3.2 Criação de Novos Grupos

Para criação de novos grupos, utiliza-se o critério de similaridade, proposto em Soroosh e Kalhor (2014), expresso por:

$$S_i(\mathbf{z}(k+1)) = \frac{1}{1 + \|\mathbf{z}(k+1) - \mathbf{z}^i(k)\|_{\mathbf{A}_f^i(k)}} \quad (4.39)$$

sendo que um novo grupo será criado sempre que a seguinte condição for satisfeita:

$$\max_{i=1}^{\tilde{c}} S_i(\mathbf{z}(k+1)) < \eta_1 \quad (4.40)$$

onde $\eta_1 \in [0,1]$ é uma constante, tal que valores grandes da mesma resultam na criação de poucos grupos. Embora a escolha ideal para η_1 dependa da aplicação, em [Soroosh e Kalhor \(2014\)](#) sugere-se utilizar $\eta_1 \in [0,6 \ 0,8]$. Os parâmetros iniciais do novo $\tilde{c}$ -ésimo grupo criado serão:

$$\mathbf{z}^{\tilde{c}}(k+1) = \mathbf{z}(k+1) \quad (4.41)$$

$$\Sigma_{\tilde{c}}(k+1) = \zeta_1 \mathbf{I}^{(l)} \quad (4.42)$$

$$\det(\Sigma_{\tilde{c}}(k+1)) = \zeta_1^l \quad (4.43)$$

$$\mathbf{A}_f^{\tilde{c}} = \mathbf{I}^{(l)} \quad (4.44)$$

onde $\zeta_1 = 10^{-2} \sim 10^{-5}$.

4.3.3 Mecanismo de Fusão de Grupos

Um mecanismo útil para o desenvolvimento de métodos de agrupamento é a fusão de grupos similares para aprimorar a generalização e reduzir a redundância ([ANGELOV, 2013](#)). O primeiro estágio da etapa de fusão consiste em encontrar grupos similares usando a medida de similaridade da equação 4.39. Desse modo, a similaridade de um par (p, q) de grupos distintos é dada por:

$$\text{sim}_{pq} = \sqrt{S_p(\mathbf{z}^p(k+1))S_q(\mathbf{z}^q(k+1))} \quad (4.45)$$

onde sim_{pq} é uma medida de similaridade entre um par (p, q) de grupos distintos. A fusão ocorrerá sempre que um par (p^*, q^*) de grupos satisfizer a seguinte condição:

$$\text{sim}_{p^*q^*} > \eta_2 \quad (4.46)$$

onde $\eta_2 \in [0,1]$ é uma constante que deve sempre ser maior que η_1 , tal que valores grandes para η_2 provocam uma diminuição no número de fusões de grupos e, conforme [Soroosh e Kalhor \(2014\)](#), sugere-se $\eta_2 \in [0,85 \ 1]$. Os parâmetros do r -ésimo grupo resultante da fusão do par de grupos (p^*, q^*) são calculados como ([SOLEIMANI-B; LUCAS; ARAABI, 2010](#)):

$$N_r(k+1) = N_{p^*}(k+1) + N_{q^*}(k+1) \quad (4.47)$$

$$\mathbf{z}^r(k+1) = \frac{N_{p^*}(k+1)\mathbf{z}^{p^*}(k+1) + N_{q^*}(k+1)\mathbf{z}^{q^*}(k+1)}{N_{p^*}(k+1) + N_{q^*}(k+1)} \quad (4.48)$$

$$\begin{aligned} \Sigma_r(k+1) = & \frac{1}{N_{p^*}(k+1) + N_{q^*}(k+1)} \left\{ N_{p^*}(k+1)\Sigma_{p^*}(k+1) + N_{q^*}(k+1)\Sigma_{q^*}(k+1) \right. \\ & \left. + \left(\frac{N_{p^*}N_{q^*}}{N_{p^*} + N_{q^*}} \right) \left[\mathbf{z}^{p^*}(k+1) - \mathbf{z}^{q^*}(k+1) \right] \left[\mathbf{z}^{p^*}(k+1) - \mathbf{z}^{q^*}(k+1) \right]^T \right\} \end{aligned} \quad (4.49)$$

4.3.4 Estimação Recursiva dos Parâmetros do Consequente

Os parâmetros do consequente são calculados com base no algoritmo RFWLS, que consiste na solução do seguinte conjunto de equações:

$$\Theta_i(k+1) = \Theta_i(k) + \kappa(k) [\mathbf{y}(k+1) - \mathbf{z}(k+1)^T \Theta_i(k)] \quad (4.50)$$

$$\kappa(k) = \frac{\mathbf{P}_i(k) \mathbf{z}(k+1)}{\frac{1}{\gamma^i(\mathbf{z}(k+1))} + \mathbf{z}(k+1)^T \mathbf{P}_i(k) \mathbf{z}(k+1)} \quad (4.51)$$

$$\mathbf{P}_i(k+1) = [\mathbf{I}^{(l)} - \kappa(k) \mathbf{z}(k+1)^T] \mathbf{P}_i(k) \quad (4.52)$$

onde $\mathbf{P}_i(k) \in \Re^{l \times l}$, chamada matriz de covariância do erro do i -ésimo grupo e a matriz de parâmetro $\Theta_i(k)$, calculados no instante k são inicializados, com $i = 1, 2, \dots, \tilde{c}$. Quando um novo grupo é criado ($\tilde{c}$), estes parâmetros são inicializados com os seguintes valores:

$$\Theta_{\tilde{c}}(k+1) = \sum_{i=1}^{\tilde{c}-1} \gamma^i(\mathbf{z}(k+1)) \Theta_i(k) \quad (4.53)$$

$$\mathbf{P}_{\tilde{c}}(k+1) = \zeta_2 \mathbf{I}^{(l)} \quad (4.54)$$

com $\zeta_2 = 10^2 \sim 10^5$.

O procedimento para atualização do modelo *fuzzy* TS evolutivo, descrito nesta seção, é resumido no Algoritmo 3.

4.4 Algoritmo para o Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo Proposto

Ao passo que cada estágio da metodologia já foi devidamente esclarecida, apresenta-se todo o procedimento para controle de processos MIMO não lineares baseado na metodologia proposta através do Algoritmo 4.

Algoritmo 3 Etapa de Atualização do Modelo *Fuzzy* Evolutivo

```

1: Variáveis de Entrada
2: Indicar valores para  $\eta_1, \eta_2, \zeta_1$  e  $\zeta_2$ ;
3: Ler o valor de  $\mathbf{z}(k+1)$ ;
4: Criação de Novos Grupos
5: Calcular  $S_i(\mathbf{z}(k+1))$  (equação 4.39);
6: se Condição para criação de grupo é verdadeira (equação 4.40) então
7:    $\tilde{c} = \tilde{c} + 1$ ;
8:   Inicializar  $\mathbf{z}^{\tilde{c}}(k+1)$  (equação 4.41);
9:   Inicializar  $\Sigma_{\tilde{c}}(k+1)$  (equação 4.42);
10:  Inicializar  $\det(\Sigma_{\tilde{c}}(k+1))$  (equação 4.43);
11:  Inicializar  $\mathbf{A}_f^{\tilde{c}}$  (equação 4.44);
12:  Inicializar  $\Theta_{\tilde{c}}(k+1)$  (equação 4.53);
13:  Inicializar  $\mathbf{P}_{\tilde{c}}(k+1)$  (equação 4.54);
14: senão
15:   Atualização Recursiva dos Parâmetros do Antecedente
16:   para  $i \leftarrow 1$  até  $\tilde{c}$  faça
17:     Calcular  $\mathbf{z}^i(k+1)$  (equação 4.33);
18:     Calcular  $\Sigma_i(k+1)$  (equação 4.34);
19:     Calcular  $\Sigma_i^{-1}(k+1)$  (equação 4.36);
20:     Calcular  $\det(\Sigma_i(k+1))$  (equação 4.37);
21:     Calcular  $\mathbf{A}_f^i(k+1)$  (equação 4.38);
22:   fim para
23:   Estimação Recursiva dos Parâmetros do Consequente
24:   para  $i \leftarrow 1$  até  $\tilde{c}$  faça
25:     Calcular  $\Theta_i(k+1)$  (equação 4.50);
26:     Calcular  $\mathbf{P}_i(k+1)$  (equação 4.52);
27:   fim para
28:   Mecanismo de Fusão de Grupos
29:   Calcular  $\text{sim}_{pq}$  (Equação 4.45);
30:   se Condição para fusão é verdadeira (equação 4.46) então
31:     Calcular  $N_r(k+1)$  (equação 4.47);
32:     Calcular  $\mathbf{z}^r(k+1)$  (equação 4.48);
33:     Calcular  $\Sigma_r(k+1)$  (equação 4.49);
34:      $\tilde{c} = \tilde{c} - 1$ 
35:   fim se
36: fim se

```

Algoritmo 4 Algoritmo para o Controle Preditivo Baseado em Modelo *Fuzzy* Evolutivo Proposto

```

1: Etapa de Treinamento
2: Calcular parâmetros iniciais do modelo fuzzy TS (Algoritmo 1);
3: para  $k \leftarrow 1$  até ... faça
4:   Etapa de Controle
5:   Calcular sinal de controle  $\mathbf{u}(k|k)$  (Algoritmo 2);
6:   Etapa de Atualização do Modelo Fuzzy Evolutivo;
7:   Atualizar parâmetros do modelo fuzzy TS evolutivo (Algoritmo 3);
8: fim para

```

5 Resultados Computacionais e Experimentais

Neste capítulo, a metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo é aplicada em três processos não lineares com características distintas. O primeiro é um processo SISO não linear CSTR (*Continuous-Stirred Tank Reactor*). Este, por sua vez, é um conhecido *benchmark* para avaliar metodologias de NMPC e têm sido amplamente explorado em diversos trabalhos como, por exemplo, em [Oviedo, Vandewalle e Wertz \(2004\)](#), [Su e Ma \(2012\)](#), [Ghaffari, Naghavi e Safavi \(2013\)](#), [Boulkaibet et al. \(2017\)](#) e [Xie et al. \(2018\)](#). O segundo consiste em um controle nível de um processo MIMO não linear com quatro tanques acoplados. Este, por sua vez, é estudado no meio acadêmico para avaliar estratégias de controle multivariável, a exemplos dos trabalhos de [Johansson \(2000\)](#), [Alavi e Hayes \(2006\)](#), [Roinila, Vilkkio e Jaatinen \(2008\)](#), [Gaaloul e M'Sahli \(2009\)](#) e [Rosinová e Kozáková \(2012\)](#). O terceiro é um processo laboratorial SISO não linear térmico que, por sua vez, foi utilizado para a aplicação de diferentes técnicas de identificação e controle em anos recentes ([FILHO, 2017](#))([COSTA, 2016](#)).

5.1 Experimento 1- Controle de Processo SISO Não Linear CSTR

O diagrama esquemático do processo é apresentado na figura 6. Este, por sua vez, é responsável pela conversão de um produto A_a em um novo produto B_b . A variável de saída $C_a(t)$ [mol/l] é a concentração do produto A_a , enquanto que $T(t)$ [K] é a temperatura da mistura. A reação resultante é exotérmica e a taxa de fluxo do refrigerante $q_c(t)$ [l/min] (variável de entrada) é usada para controlar essa reação. A constante C_{a0} é a concentração de alimentação da entrada e q é uma constante que representa a taxa de fluxo do processo.

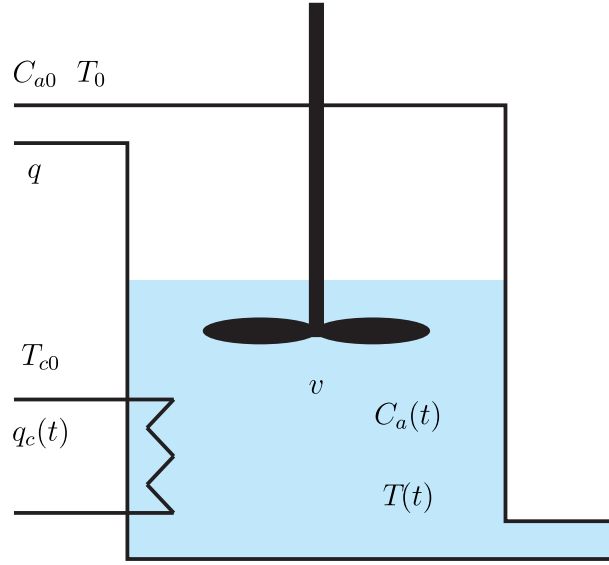
O modelo não linear do processo é descrito pelo seguinte conjunto de equações:

$$\frac{dC_a(t)}{dt} = \frac{q}{v} (C_{a0} - C_a(t)) - k_0 C_a(t) e^{-\frac{E}{RT(t)}} \quad (5.1)$$

$$\frac{dT(t)}{dt} = \frac{q}{v} (T_0 - T(t)) + k_1 C_a(t) e^{-\frac{E}{RT(t)}} + k_2 q_c(t) \left(1 - e^{-\frac{k_3}{q_c(t)}}\right) (T_{c0} - T(t)) \quad (5.2)$$

onde $k_1 = \frac{\Delta H k_0}{\rho C_p}$, $k_2 = \frac{\rho_c C_{pc}}{\rho C_p v}$ e $k_3 = \frac{h_a}{\rho_c C_{pc}}$. A Tabela 1 fornece as descrições e os valores das constantes utilizadas neste experimento.

Figura 6 – Diagrama esquemático de um processo SISO não linear CSTR.



Fonte: Adaptado de Boulkaibet et al. (2017).

Tabela 1 – Parâmetros do processo SISO não linear CSTR.

Parâmetros	Descrição	Valor Nominal
q	Taxa de fluxo do processo	100 l/min
k_0	Constante da taxa de reação	$7,2 \times 10^{10} \text{ min}^{-1}$
v	Volume do reator	100 l
T_0	Temperatura de alimentação	350 K
E/R	Energia de ativação	$1 \times 10^4 \text{ K}$
T_{c0}	Temperatura refrigerante interna	350 K
ΔH	Calor de reação	$2 \times 10^5 \text{ cal/mol}$
ρ, ρ_c	Densidade dos líquidos	$1 \times 10^3 \text{ g/l}$
C_p, C_{pc}	Calores específicos	1 cal/g/K
C_{a0}	Concentração da alimentação interna	1 mol/l
h_a	coeficiente de transferência de calor	$7 \times 10^5 \text{ cal/min/K}$

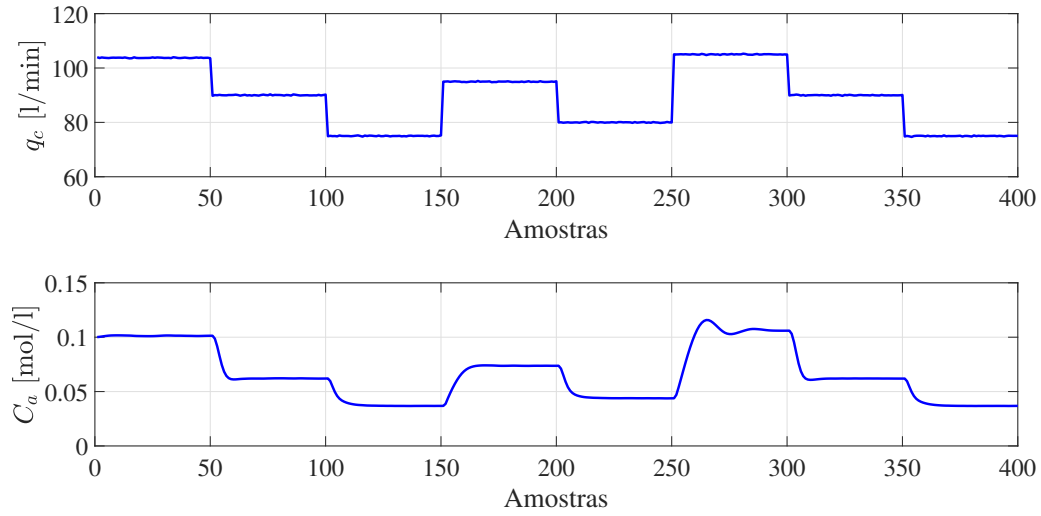
5.1.1 Identificação do Modelo *Fuzzy* TS Inicial

Para identificar o modelo *fuzzy* TS inicial utilizaram-se os seguinte parâmetros: $n_u = 3$, $n_y = 5$, $c = 25$, $N = 400$, $T_s = 0,1 \text{ min}$, $\rho^i = 1 \ (\forall i)$ e $m_f = 2$. A figuras 7 apresenta os valores de entrada e saída utilizados para identificação do modelo inicial. A figura 8 apresenta a validação do modelo $\tilde{y}$ obtido.

5.1.2 Avaliação do Controlador

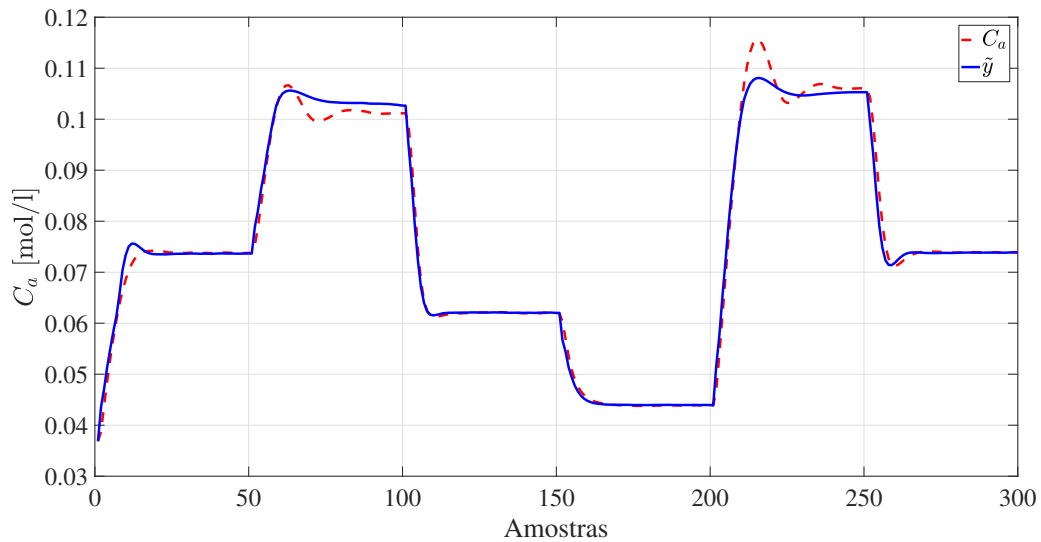
Os parâmetros de entrada do controlador foram os seguintes: $N_2 = 10$, $N_u = 5$, $\epsilon_{(i_1-1), i_2} = 10^{-3} \ (\forall i_1, i_2)$, $\mathbf{Q}_\delta = \mathbf{I}^{(N_2)}$, $\mathbf{Q}_\lambda = 0,0001\mathbf{I}^{(N_u)}$, $\mathbf{u}_{\min} = 75$, $\mathbf{u}_{\max} = 110$, $\mathbf{du}_{\min} = -10$, $\mathbf{du}_{\max} = 10$, $\mathbf{y}_{\min} = 0$, $\mathbf{y}_{\max} = 0,12$, $\eta_1 = 0,8$, $\eta_2 = 0,85$, $\zeta_1 = 10^{-2}$ e $\zeta_2 = 10^4$.

Figura 7 – Experimento 1: dados de entrada q_c [l/min] e saída C_a [mol/l] do processo utilizados para identificação do modelo *fuzzy* TS inicial.



Fonte: Autor.

Figura 8 – Experimento 1: validação do modelo *fuzzy* TS inicial, onde a linha contínua em azul indica a saída do modelo $\tilde{y}$ [mol/l], enquanto que a linha tracejada vermelha representa a saída real C_a [mol/l].



Fonte: Autor.

Assim, o desempenho do controlador proposto foi avaliado em três cenários distintos. O primeiro consiste no rastreamento de uma trajetória de referência ω constante. O segundo consiste no mesmo rastreamento com a inserção de ruído na saída. O terceiro avalia a capacidade do controlador proposto de rejeitar perturbações constantes na saída.

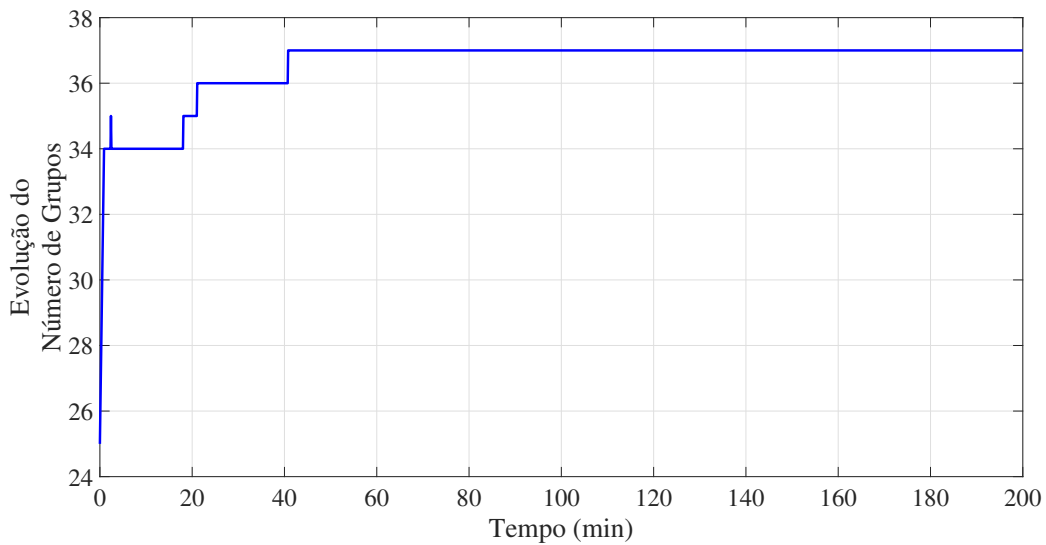
5.1.2.1 Cenário 1 - Seguimento de Referência

Neste cenário, o controlador preditivo baseado em modelo *fuzzy* evolutivo foi submetido à tarefa de rastrear uma trajetória de referência constante expressa matematicamente pela seguinte função periódica $\omega(t)$:

$$\omega(t + nT) = \begin{cases} 0,05, & 0 \leq t \leq 20 \text{ s}, \\ 0,1, & 20 \leq t \leq 40 \text{ s} \end{cases} \quad (5.3)$$

onde $n = 1, \dots, 5$, $T = 20$ min é o período de repetição. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros neste cenário são apresentados nas figuras 9, 10 e 11, respectivamente. O comportamento da saída e sinal de controle são apresentados na figura 12.

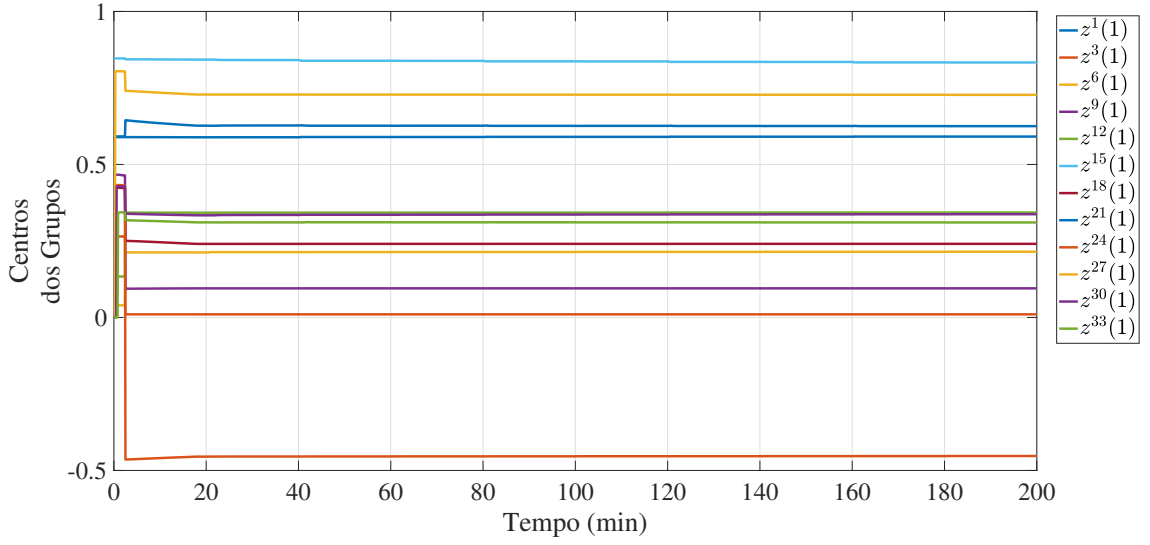
Figura 9 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).



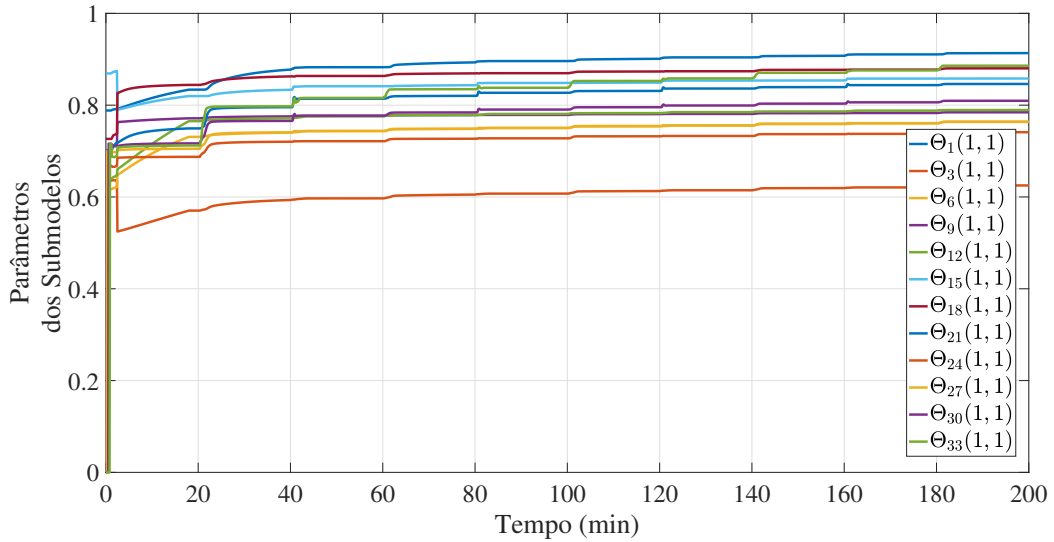
Fonte: Autor.

5.1.2.2 Cenário 2 - Seguimento de Referência na Presença de Ruído

Neste cenário, o controlador preditivo baseado em modelo *fuzzy* evolutivo foi submetido à mesma trajetória de referência do cenário anterior (equação 5.3) com a inclusão de um ruído branco com média nula e variância igual a 0,0001 adicionado à saída do processo com o intuito de emular ruído de medição. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros nesse cenário são apresentados nas figuras 13, 14 e 15, respectivamente. O comportamento da saída e sinal de controle são apresentados na figura 16.

Figura 10 – Experimento 1: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 1).

Fonte: Autor.

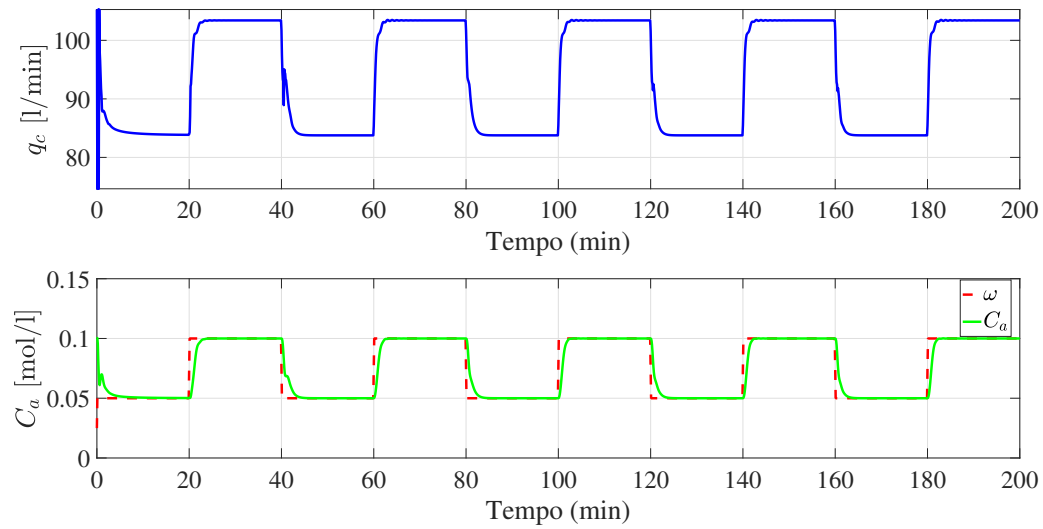
Figura 11 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$. (cenário 1).

Fonte: Autor.

5.1.2.3 Cenário 3 - Rejeição à Distúrbios

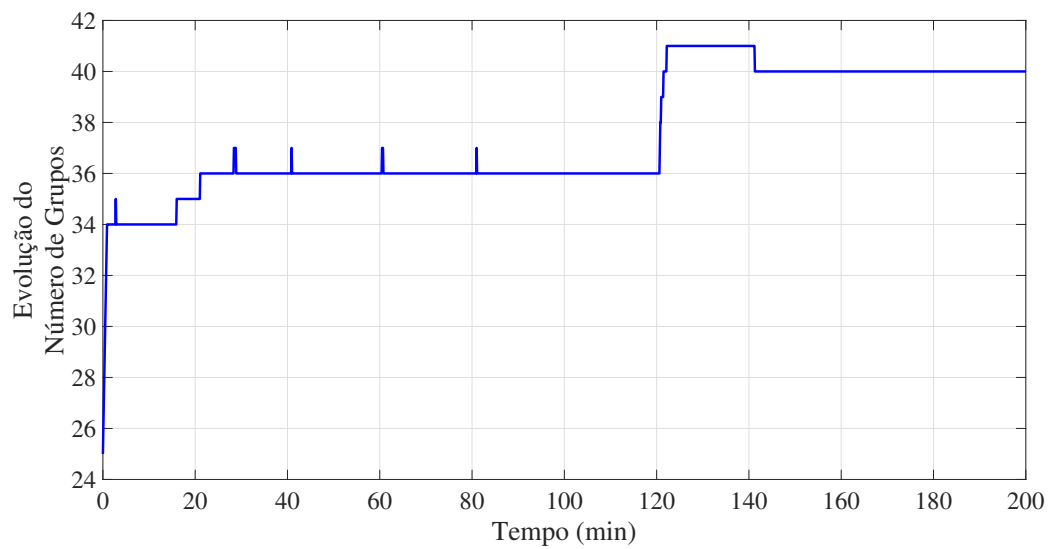
Neste cenário, a capacidade do controlador preditivo baseado em modelo *fuzzy* evolutivo é avaliada quanto à capacidade de rejeitar distúrbios constantes de saída. Para isso submete-se o controlador ao rastreo de uma trajetória de referência constante $\omega(k) = 0,1$ e aplica-se um distúrbio constante de saída com valor igual a 0,001 no instante $t = 100$ min. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros são apresentados nas figuras 17, 18 e 19, respectivamente. O comportamento

Figura 12 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 1).



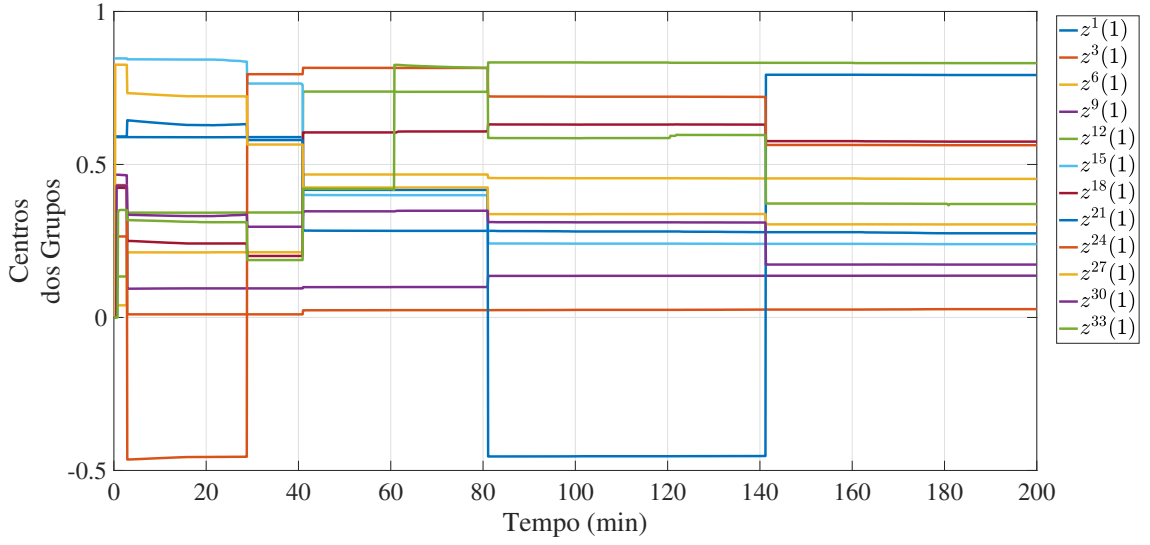
Fonte: Autor.

Figura 13 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).

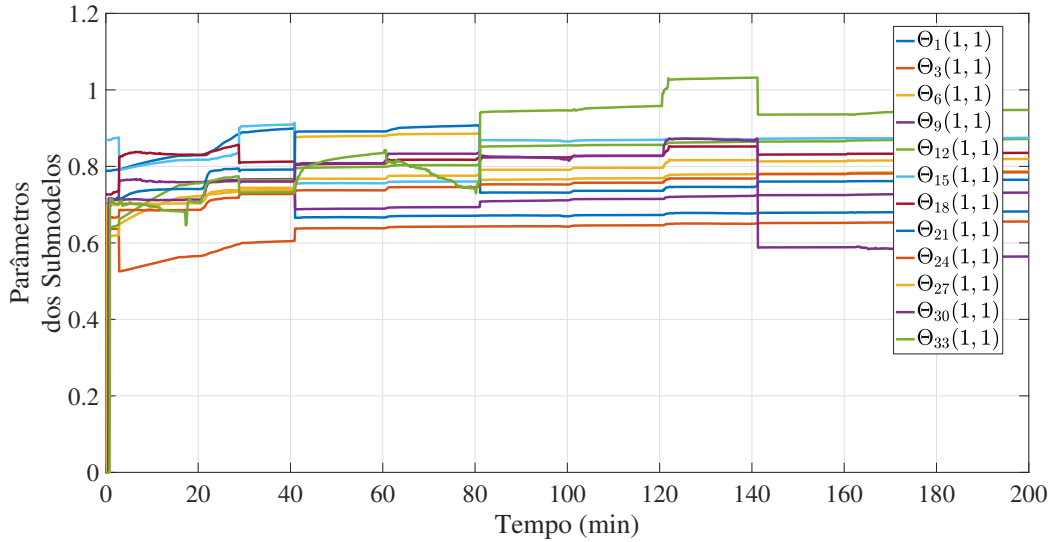


Fonte: Autor.

da saída e sinal de controle são apresentados na figura 20.

Figura 14 – Experimento 1: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 2).

Fonte: Autor.

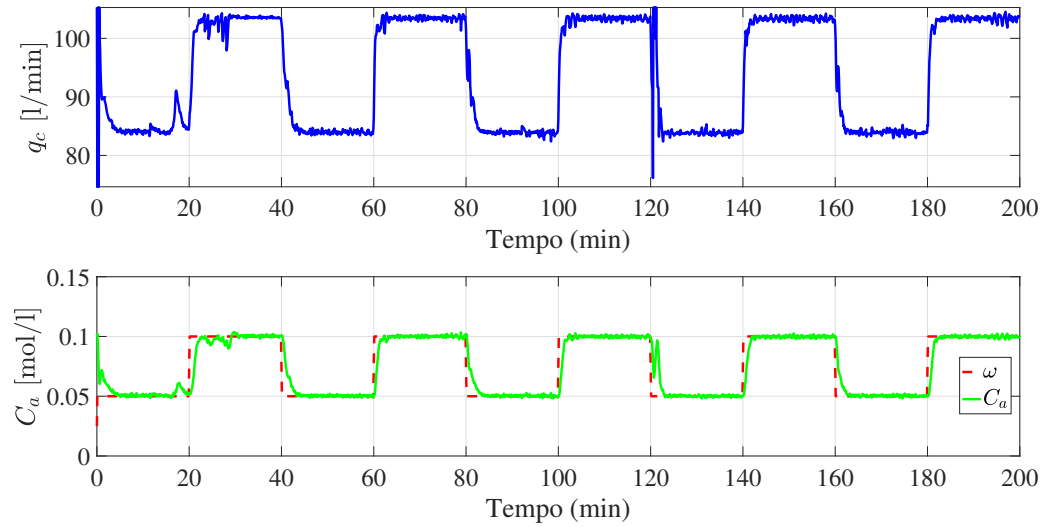
Figura 15 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).

Fonte: Autor.

5.2 Experimento 2- Controle de Nível de um Processo MIMO Não Linear com Quatro Tanques Acoplados

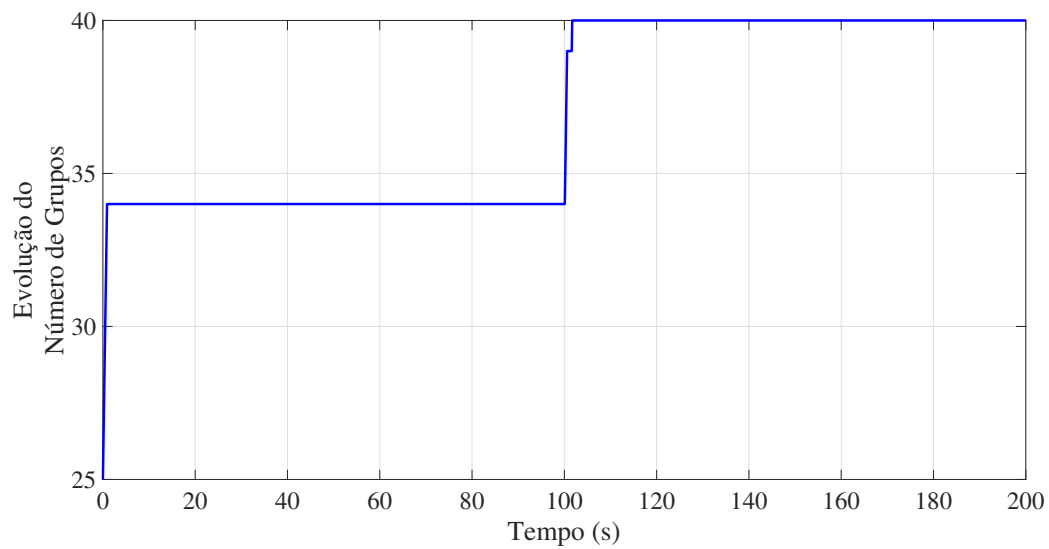
O diagrama esquemático do processo MIMO não linear com quatro tanques acoplados é apresentado na figura 21. O objetivo é controlar os níveis dos dois tanques inferiores com duas bombas. As entradas do processo $u_1(t)$ ([V]) e $u_2(t)$ ([V]) são os sinais de tensão das bombas 1 e 2, respectivamente. As saídas $y_1(t)$ ([cm]) e $y_2(t)$ ([cm]) são as alturas dos tanques 1 e 2, respectivamente. O modelo do processo, por sua vez, é descrito pelo

Figura 16 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 2).



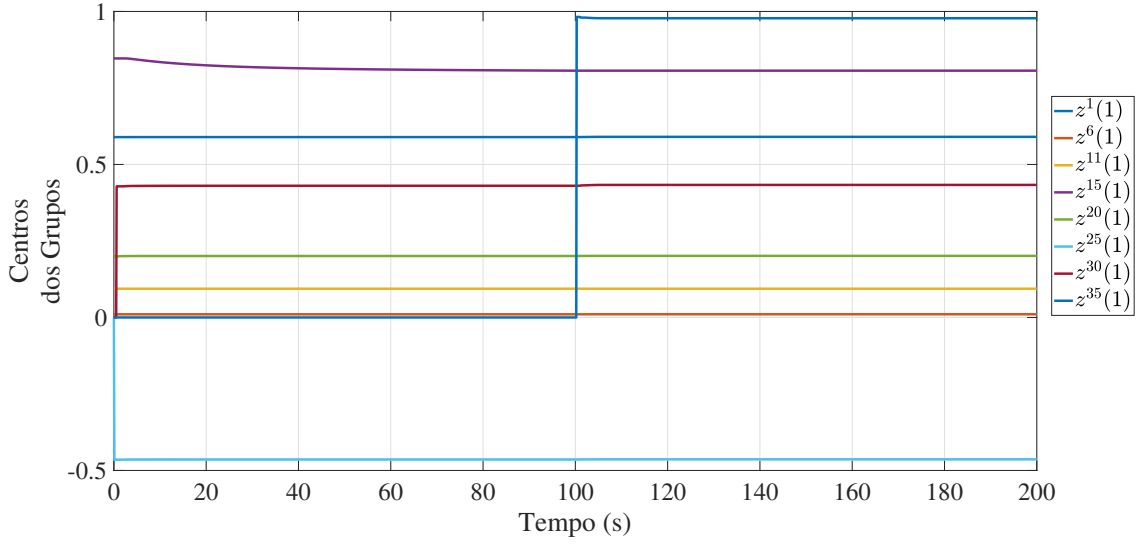
Fonte: Autor.

Figura 17 – Experimento 1: evolução do número de grupos (regras) $\tilde{c}$ (cenário 3).

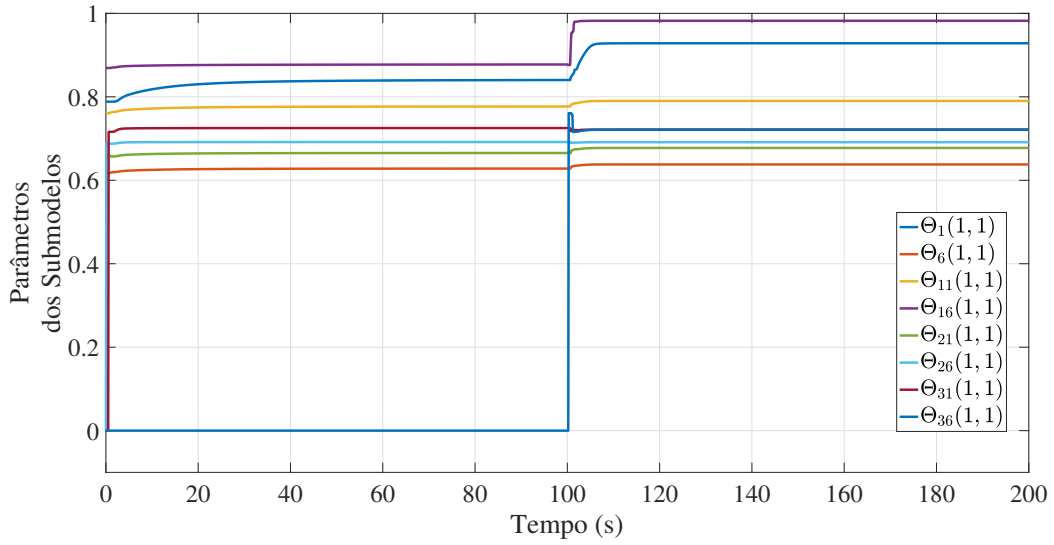


Fonte: Autor.

seguinte conjunto de equações:

Figura 18 – Experimento 1: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 3).

Fonte: Autor.

Figura 19 – Experimento 1: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 3).

Fonte: Autor.

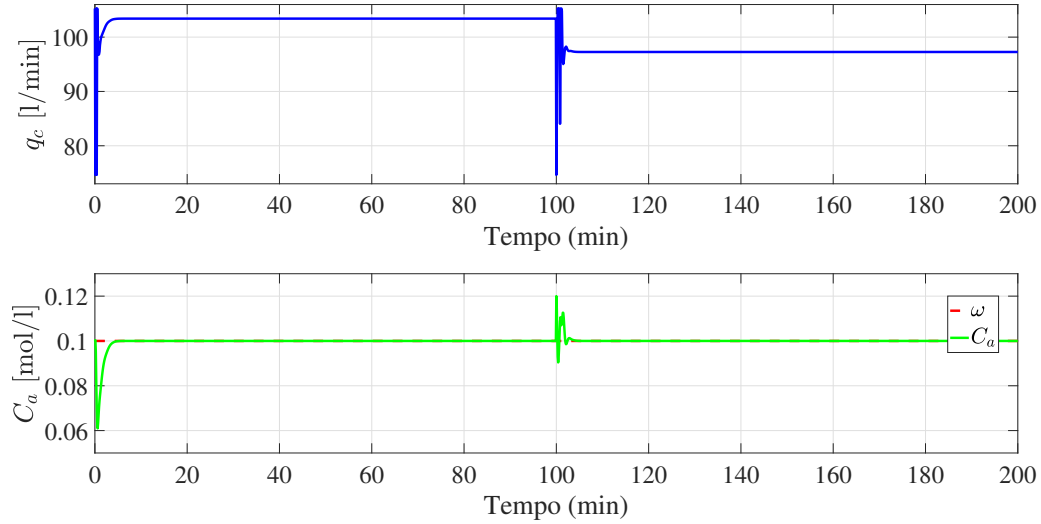
$$\frac{dy_1(t)}{dt} = -\frac{a_1}{A_1}\sqrt{2gy_1(t)} + \frac{a_3}{A_1}\sqrt{2gy_3(t)} + \frac{\gamma_1 k_1}{A_1}u_1(t) \quad (5.4)$$

$$\frac{dy_2(t)}{dt} = -\frac{a_2}{A_2}\sqrt{2gy_2(t)} + \frac{a_4}{A_2}\sqrt{2gy_3(t)} + \frac{\gamma_2 k_2}{A_2}u_2(t) \quad (5.5)$$

$$\frac{dy_3(t)}{dt} = -\frac{a_3}{A_3}\sqrt{2gy_3(t)} + \frac{(1-\gamma_2)k_2}{A_3}u_2(t) \quad (5.6)$$

$$\frac{dy_4(t)}{dt} = -\frac{a_4}{A_4}\sqrt{2gy_4(t)} + \frac{(1-\gamma_1)k_1}{A_4}u_1(t) \quad (5.7)$$

Figura 20 – Experimento 1: resposta da saída C_a [mol/l] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado q_c [l/min] (cenário 3).



Fonte: Autor.

Tabela 2 – Parâmetros do processo MIMO não linear com quatro tanques acoplados.

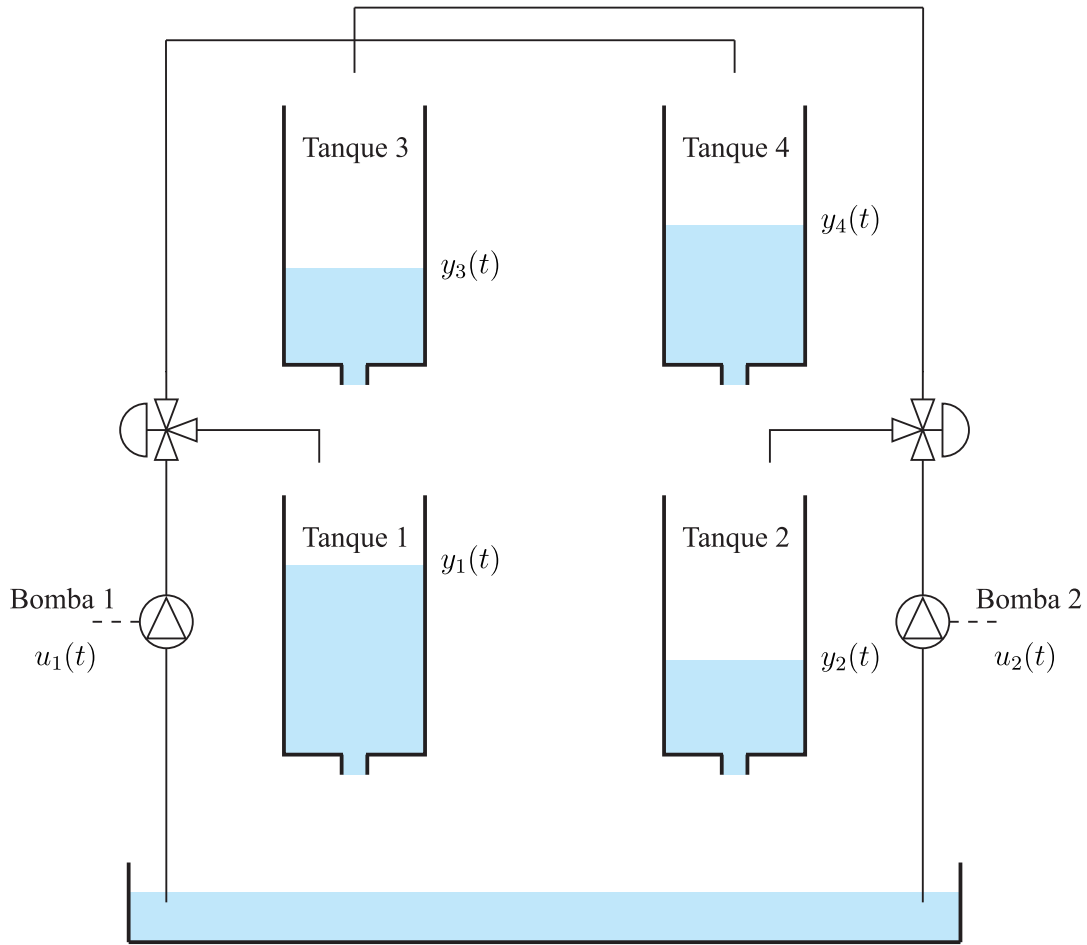
Parâmetros	Descrição	Valor Nominal
A_1, A_2	Área da seção transversal dos tanques 1 e 2	28 cm ²
A_2, A_4	Área da seção transversal dos tanques 2 e 4	32 cm ²
a_1, a_3	Área da seção transversal dos orifícios dos tanques 1 e 3	0,071 cm ²
a_2, a_4	Área da seção transversal dos orifícios dos tanques 2 e 4	0,057 cm ²
g	Aceleração da gravidade	981 cm/s ²
k_1	Constante da válvula 1	3,33 cm ³ /Vs
k_2	Constante da válvula 2	3,33 cm ³ /Vs
γ_1	Abertura da válvula 1	0,70
γ_2	Abertura da válvula 2	0,60

onde A_n e a_n são a área da seção transversal e do orifício de saída do n -ésimo tanque, respectivamente, com $n = 1, \dots, 4$; as constantes γ_1 e $\gamma_2 \in (0,1)$ correspondem as aberturas das válvulas 1 e 2, respectivamente; as constantes k_1 e k_2 estão associadas às válvulas; a constante g é a aceleração da gravidade. A Tabela 1 apresenta os valores dos parâmetros utilizados no experimento.

5.2.1 Identificação do Modelo *Fuzzy* TS Inicial

Para identificar o modelo *fuzzy* TS inicial utilizaram-se os seguinte parâmetros: $n_u = 1$, $n_y = 2$, $c = 4$, $N = 2000$, $T_s = 0,5s$, $\rho^i = 1$ ($\forall i$) e $m_f = 2$. As figuras 22 e 23 apresentam os valores das entradas e saídas, respectivamente, utilizados para identificação do modelo inicial. A figura 24 apresenta a validação das saídas $\tilde{y}_1$ e $\tilde{y}_2$ do modelo obtido.

Figura 21 – Diagrama esquemático de um processo MIMO não linear com quatro tanques acoplados. A altura dos níveis de água nos tanques 1 e 2 são controlados por duas bombas.

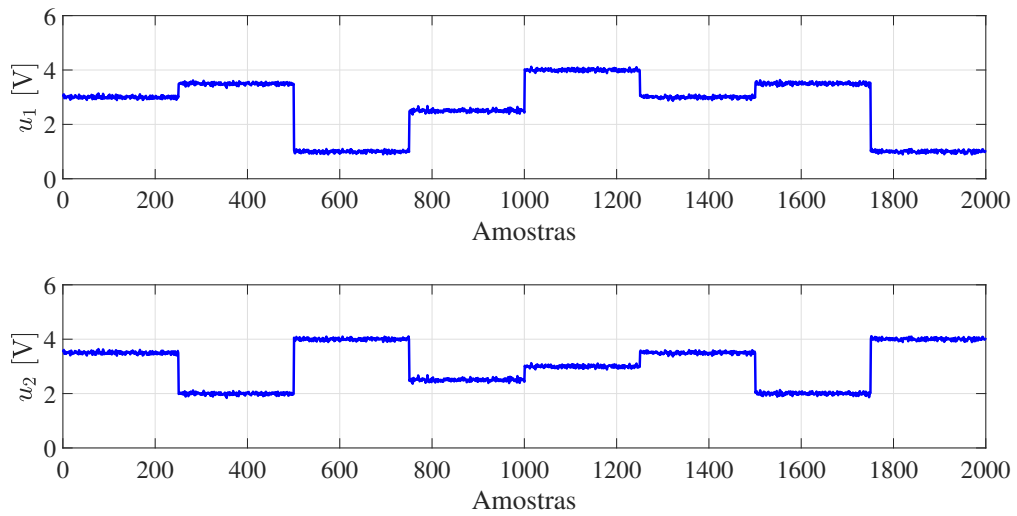


Fonte: Adaptado de Johansson (2000).

5.2.2 Avaliação do Controlador

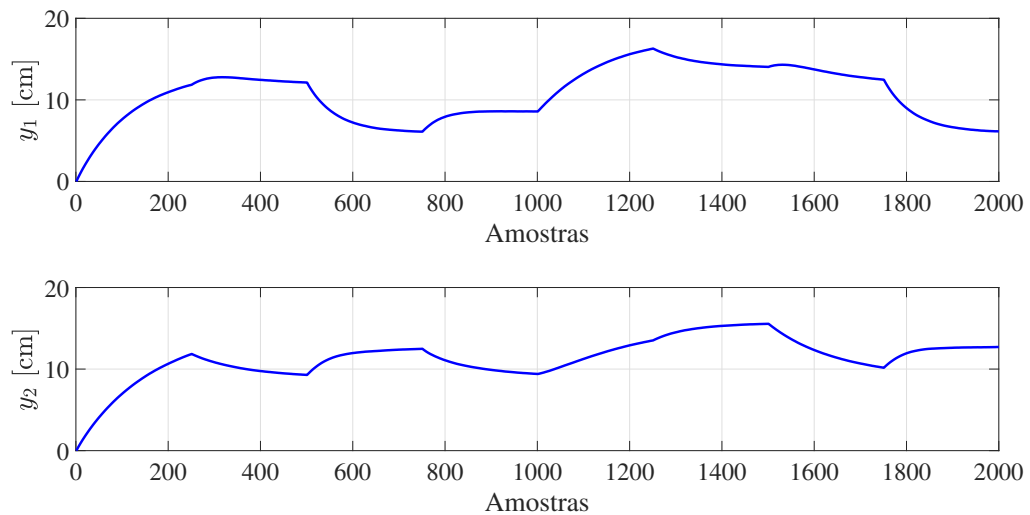
Os parâmetros de entrada do controlador foram os seguintes: $N_2 = 7$, $N_u = 4$, $\epsilon_{(i_1-1), i_2} = 10^{-2}$ ($\forall i_1, i_2$), $\mathbf{Q}_\delta = \mathbf{I}^{(2N_2)}$, $\mathbf{Q}_\lambda = 0,01\mathbf{I}^{(2N_u)}$, $\mathbf{u}_{\min} = [0,0]^T$, $\mathbf{u}_{\max} = [10,10]^T$, $\mathbf{du}_{\min} = [-5, -5]^T$, $\mathbf{du}_{\max} = [5,5]^T$, $\mathbf{y}_{\min} = [0,0]^T$, $\mathbf{y}_{\max} = [20,20]^T$, $\eta_1 = 0,8$, $\eta_2 = 0,85$, $\zeta_1 = 10^{-2}$ e $\zeta_2 = 10^4$. Dessa forma, avaliou-se o desempenho do controlador proposto de modo análogo à seção anterior.

Figura 22 – Experimento 2: entradas u_1 [V] e u_2 [V] aplicadas ao processo para identificação do modelo *fuzzy* TS inicial.



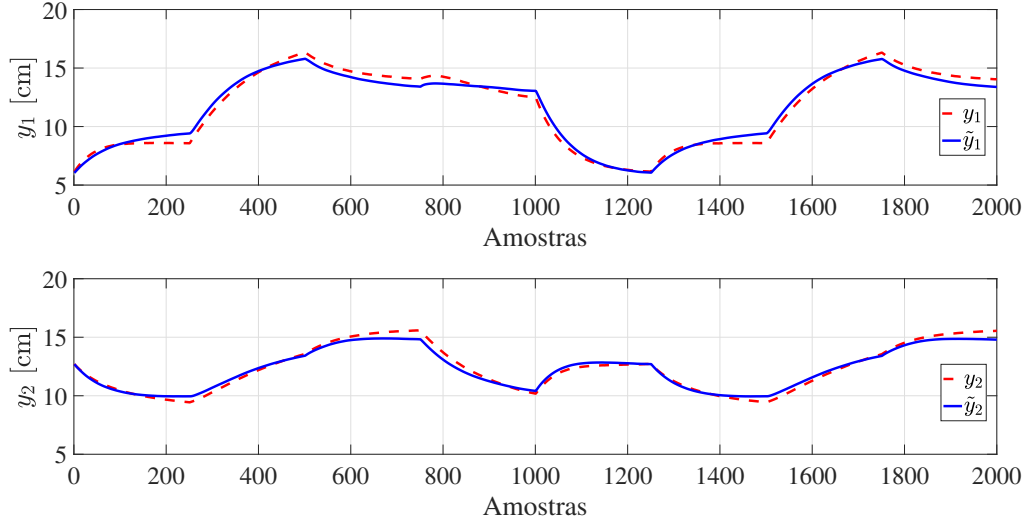
Fonte: Autor.

Figura 23 – Experimento 2: saídas y_1 [cm] e y_2 [cm] medidas do processo para identificação do modelo *fuzzy* TS inicial.



Fonte: Autor.

Figura 24 – Experimento 2: validação do modelo *fuzzy* TS inicial, onde as linhas contínuas são as saídas de validação y_1 [cm] e y_2 [cm], enquanto que as linhas tracejadas são as saídas do modelo $\tilde{y}_1$ [cm] e $\tilde{y}_2$ [cm].



Fonte: Autor.

5.2.2.1 Cenário 1 - Seguimento de Referência

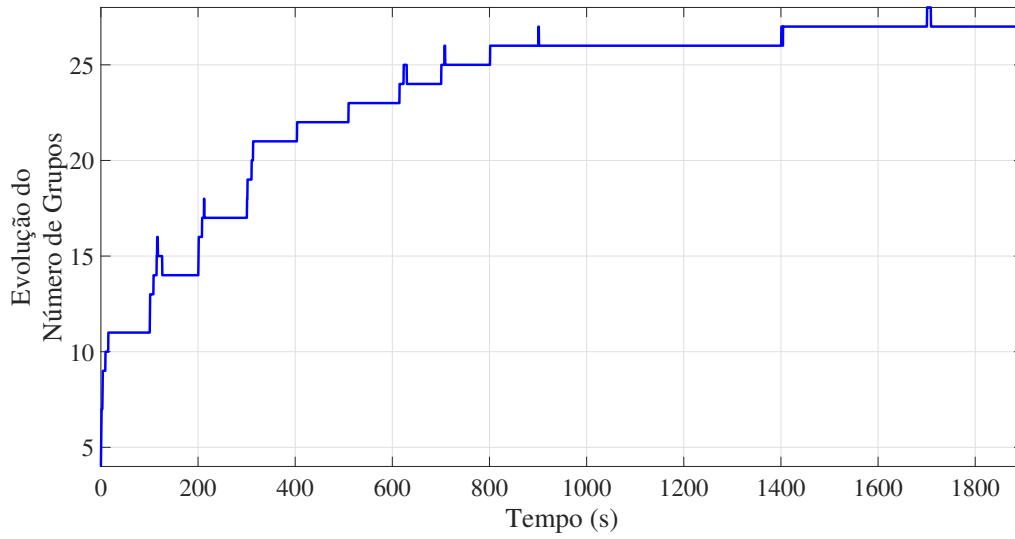
Neste cenário, o controlador foi submetido à tarefa de rastrear as trajetórias de referência constantes $\omega_1(t)$ e $\omega_2(t)$ dadas por:

$$\omega_1(t + nT) = \omega_2(t + nT) = \begin{cases} 5, & 0 \leq t \leq 100 \text{ s}, \\ 10, & 100 \text{ s} < t \leq 200 \text{ s}, \\ 15, & 200 \text{ s} \leq t \leq 300 \text{ s}, \\ 20, & 300 \text{ s} < t \leq 400 \text{ s}, \\ 15, & 400 \text{ s} \leq t \leq 500 \text{ s}, \\ 10, & 500 \text{ s} < t \leq 600 \text{ s} \end{cases} \quad (5.8)$$

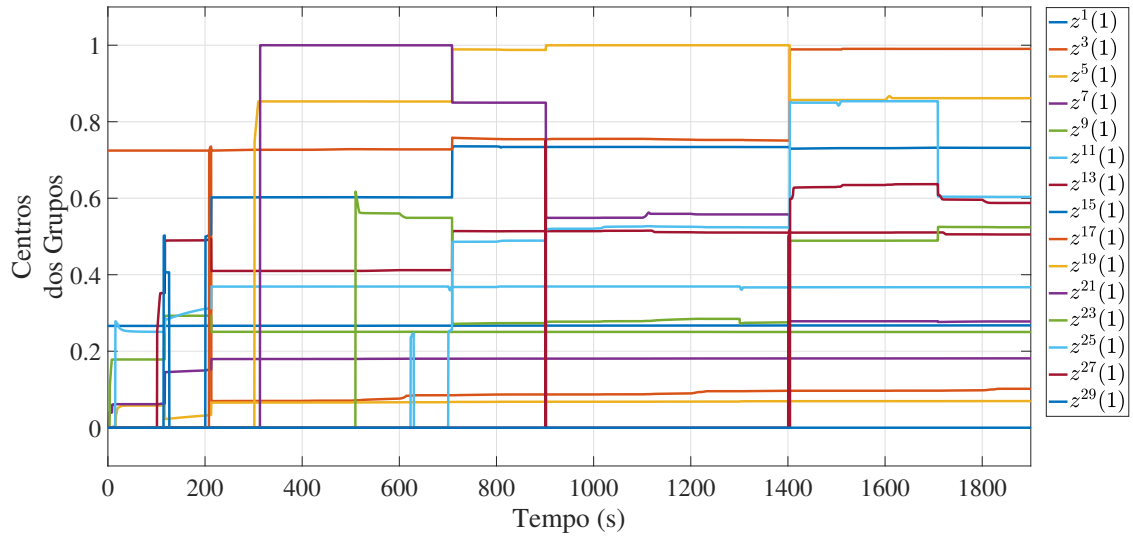
onde $n = 1, 2, 3$, $T = 600 \text{ s}$ é o período de repetição. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros nesse cenário são apresentados nas figuras 25, 26 e 27, respectivamente. O comportamento das saídas e sinais de controle são apresentados nas figuras 25, 28 e 29, respectivamente.

5.2.2.2 Cenário 2 - Seguimento de Referência na Presença de Ruído

Neste cenário, o controlador foi submetido à mesma trajetória de referência do cenário anterior (equação 5.8) com a inclusão de um ruído branco com média nula e variância igual a 0,2 adicionado às saídas do processo com o intuito de emular ruídos de medição. Os resultados obtidos para evolução das regras, evolução dos centros e evolução

Figura 25 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).

Fonte: Autor.

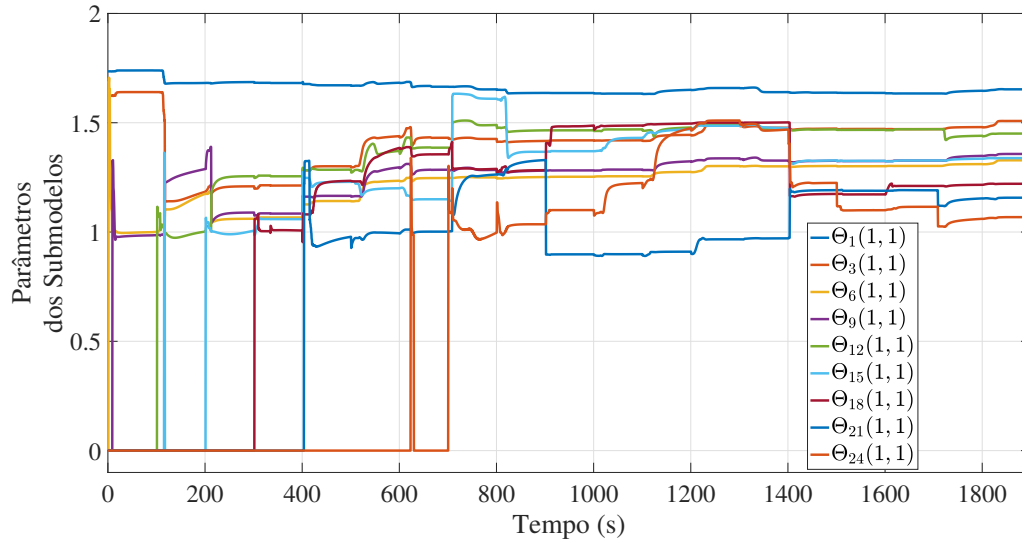
Figura 26 – Experimento 2: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 1).

Fonte: Autor.

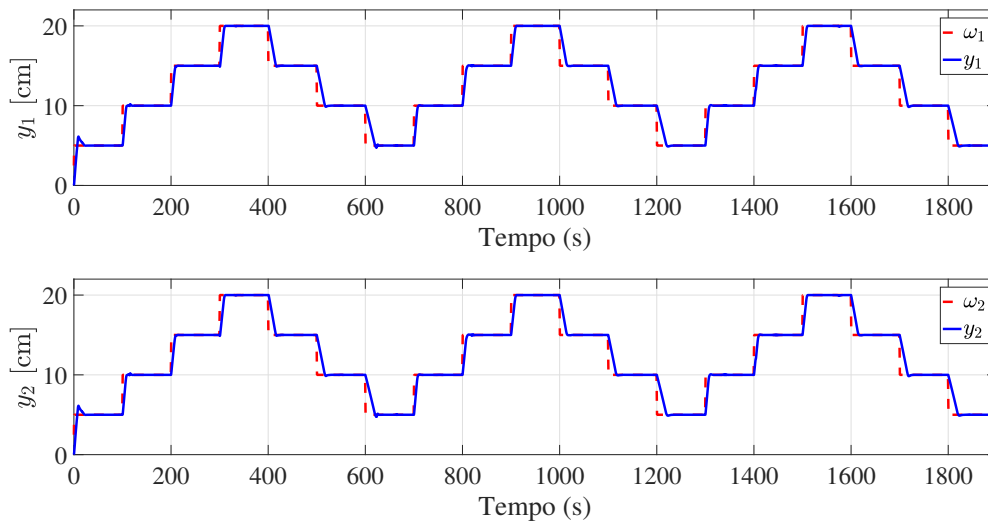
dos parâmetros são apresentados nas figuras 30, 31 e 32, respectivamente. O comportamento das saídas e sinais de controle são apresentados nas figuras 33 e 34, respectivamente.

5.2.2.3 Cenário 3 - Rejeição à Distúrbios

Neste cenário, o controlador preditivo baseado em modelo *fuzzy* evolutivo é avaliada quanto à capacidade de rejeitar distúrbios de carga constantes. Para isso submete-se o controlador ao rastreamento de trajetórias de referências constantes $\omega_1(t) = \omega_2(t) = 10$ e aplica um distúrbio na saída y_1 de valor igual a -1 no instante $t = 100$ s e outro na saída y_2

Figura 27 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 1).

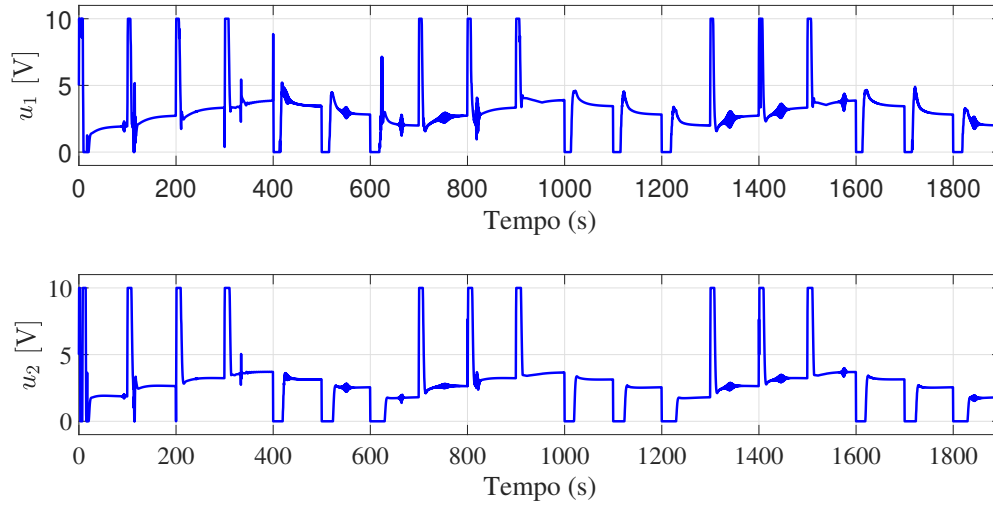
Fonte: Autor.

Figura 28 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 1).

Fonte: Autor.

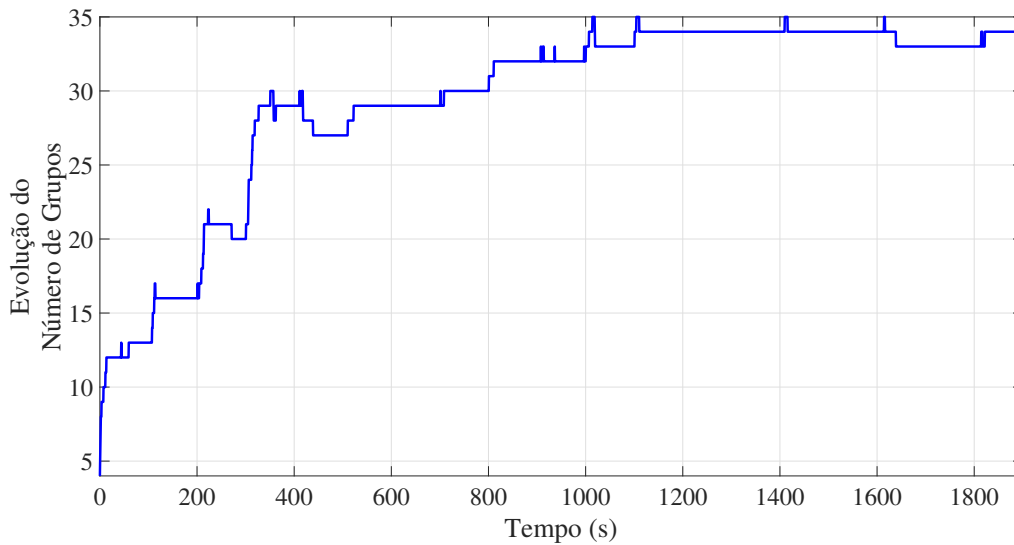
de valor igual a 1 no instante $t = 200$ s. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros nesse cenário são apresentados nas figuras 35, 36 e 37, respectivamente. O comportamento das saídas e sinais de controle aplicados na entrada do processo são apresentados nas figuras 38 e 39, respectivamente.

Figura 29 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 1).



Fonte: Autor.

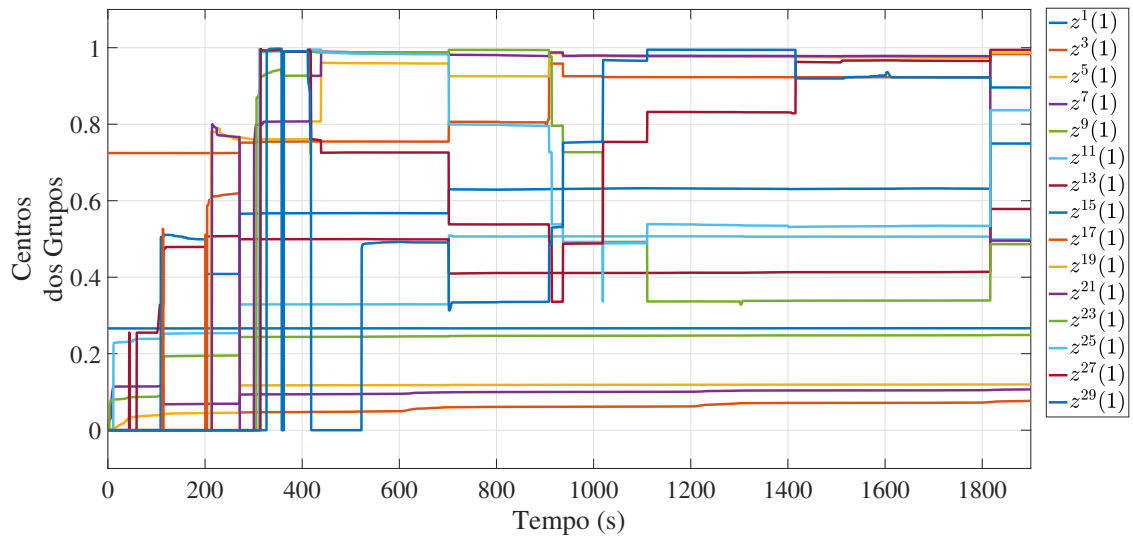
Figura 30 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).



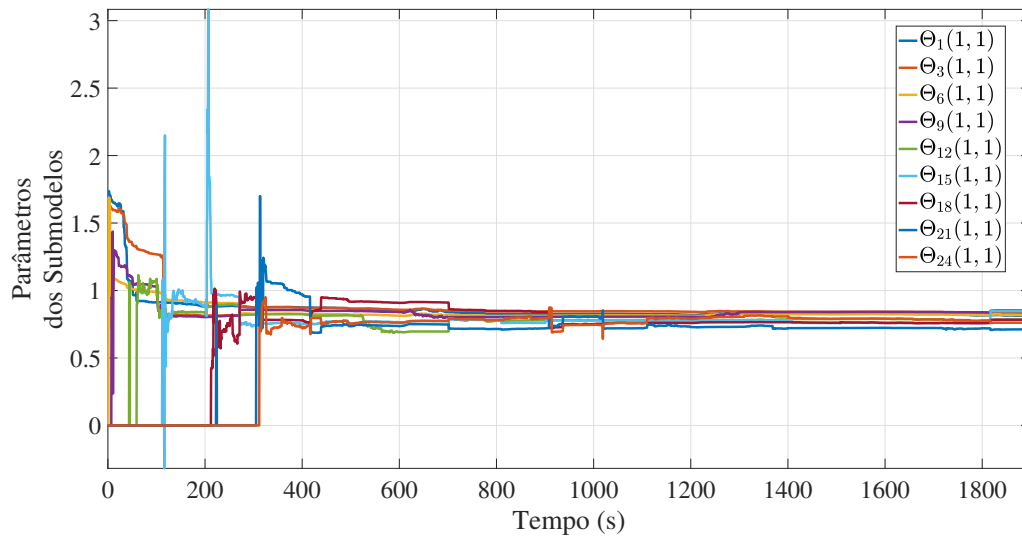
Fonte: Autor.

5.3 Experimento 3 - Processo SISO Não Linear Térmico

A plataforma de controle utilizada neste experimento, apresentada na figura 40, é composta de uma processo SISO não linear térmico, o *software LabVIEWTM* (*Laboratory Virtual Instrument Engineering Workbench*), o CompactRIO 9073, o módulo para entrada analógica NI 9219, o módulo para saída analógica NI 9263, o sensor de temperatura LM 35 e o atuador CI TCA 785. O processo térmico é uma torradeira monofásica de 220 [V]

Figura 31 – Experimento 2: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 2).

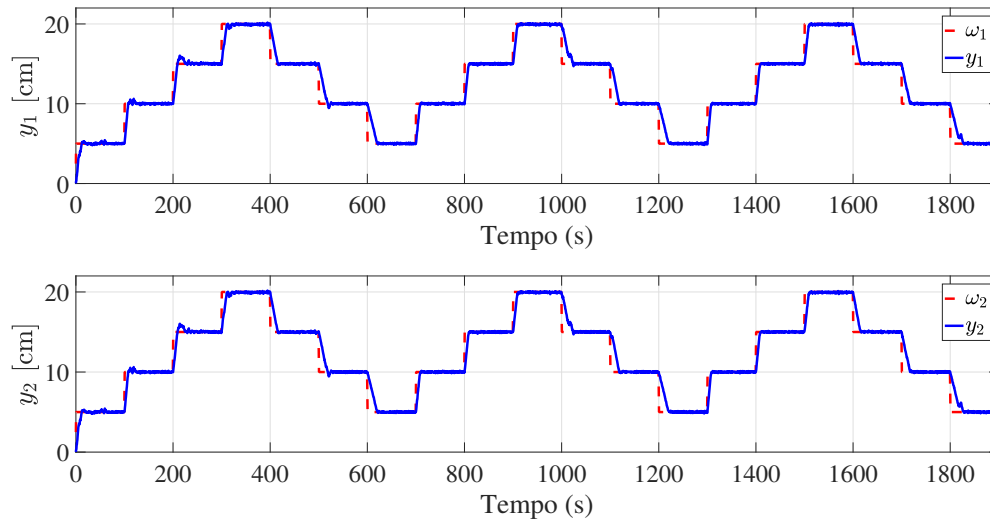
Fonte: Autor.

Figura 32 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).

Fonte: Autor.

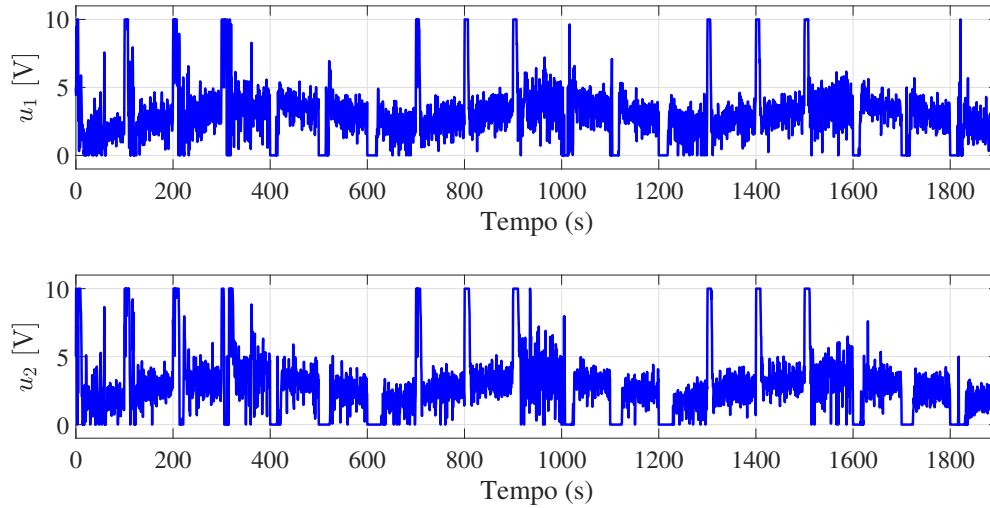
em corrente alternada, com faixa de temperatura no intervalo de 25 [°C] a 200 [°C]. O *LabVIEWTM* é um ambiente de desenvolvimento flexível que contém todas as ferramentas para projetar e implementar sistemas de medição e controle. O objetivo deste experimento é controlar a saída y [°C] de temperatura com o sinal de controle u [V].

Figura 33 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 2).



Fonte: Autor.

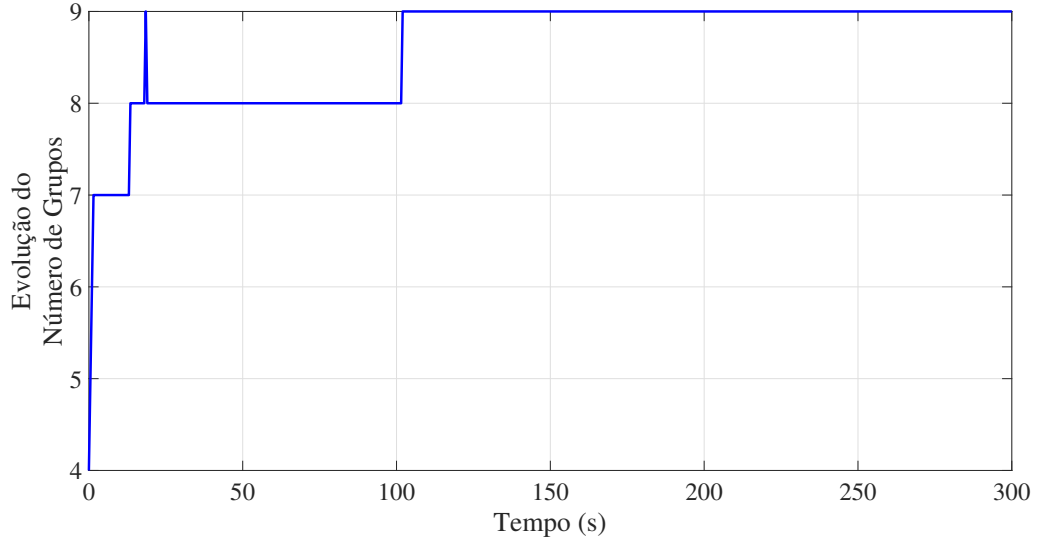
Figura 34 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 2).



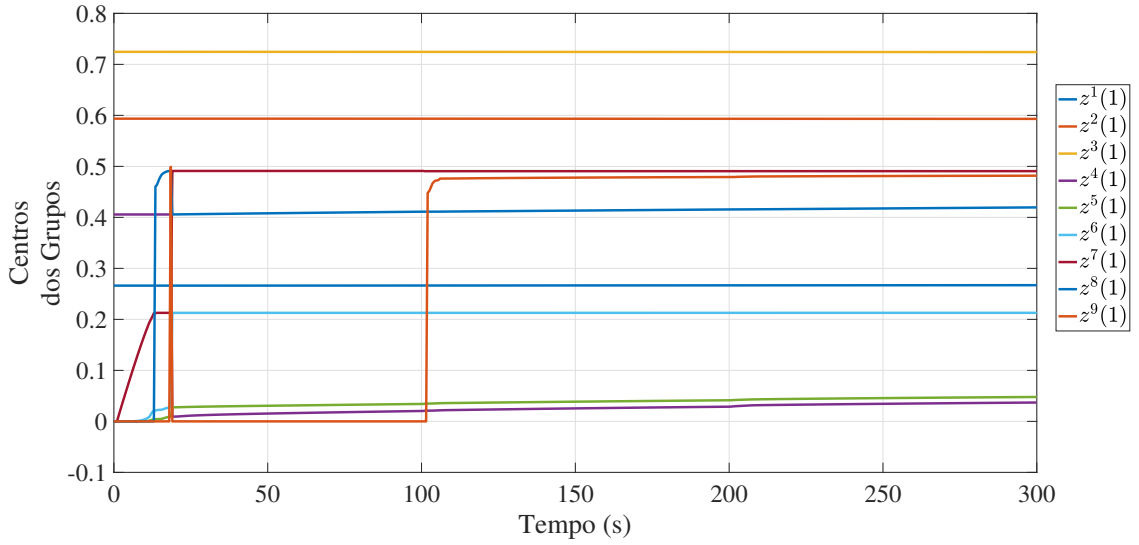
Fonte: Autor.

5.3.1 Identificação do Modelo *Fuzzy* TS Inicial

Para identificar o modelo *fuzzy* TS inicial utilizaram-se os seguinte parâmetros: $n_u = 2$, $n_y = 2$, $c = 12$, $N = 12000$, $T_s = 1$ s, $\rho^i = 1$ ($\forall i$) e $m_f = 2$. A figuras 41 apresenta os valores de entrada e saída utilizados para identificação do modelo inicial. A figura 42 apresenta a validação do modelo $\tilde{y}$ obtido.

Figura 35 – Experimento 2: evolução do número de grupos (regras) $\tilde{c}$ (cenário 3).

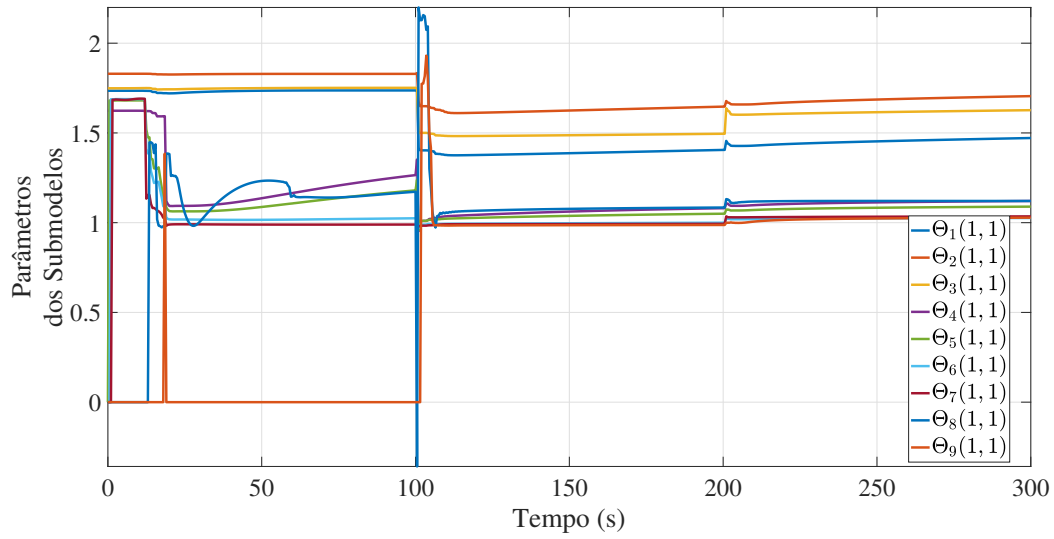
Fonte: Autor.

Figura 36 – Experimento 2: evolução dos centros dos grupos (regras) $\mathbf{z}^i(1)$ (cenário 3).

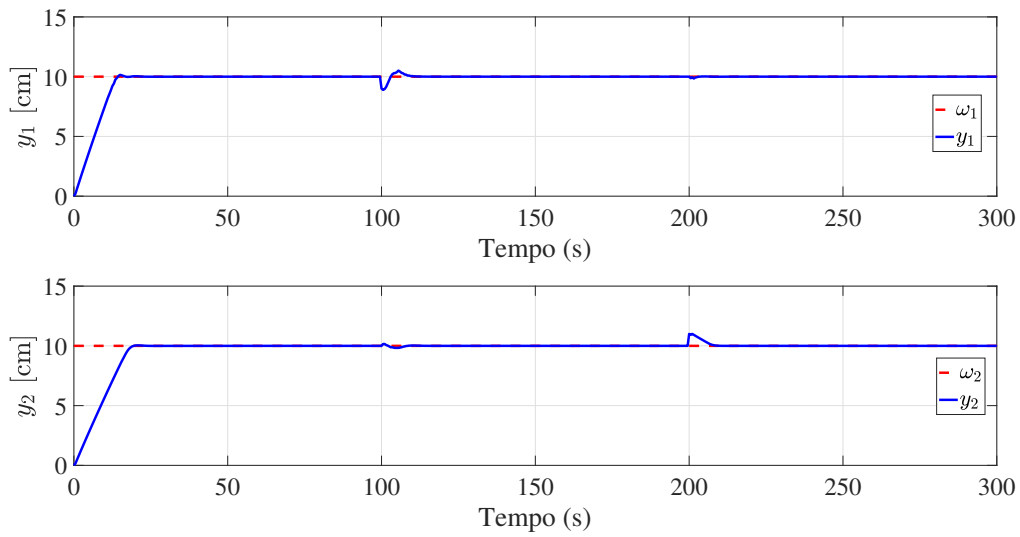
Fonte: Autor.

5.3.2 Avaliação do Controlador

Os parâmetros de entrada do controlador foram os seguintes: $N_2 = 7$, $N_u = 4$, $\epsilon_{(i_1-1), i_2} = 10^{-2}$ ($\forall i_1, i_2$), $\mathbf{Q}_\delta = \mathbf{I}^{(2N_2)}$, $\mathbf{Q}_\lambda = 0,01\mathbf{I}^{(2N_u)}$, $\mathbf{u}_{\min} = 0$, $\mathbf{u}_{\max} = 140$, $\mathbf{du}_{\min} = -220$, $\mathbf{du}_{\max} = 220$, $\mathbf{y}_{\min} = 0$, $\mathbf{y}_{\max} = 150$, $\eta_1 = 0,6$, $\eta_2 = 0,85$, $\zeta_1 = 10^{-2}$ e $\zeta_2 = 10^4$. Dessa forma, avaliou-se o desempenho do controlador em dois cenários: seguimento de referência ω constante e rejeição à distúrbio constante na saída.

Figura 37 – Experimento 2: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 3).

Fonte: Autor.

Figura 38 – Experimento 2: resposta das saídas y_1 [cm] e y_2 [cm] no rastreo das referências ω_1 e ω_2 , respectivamente (cenário 3).

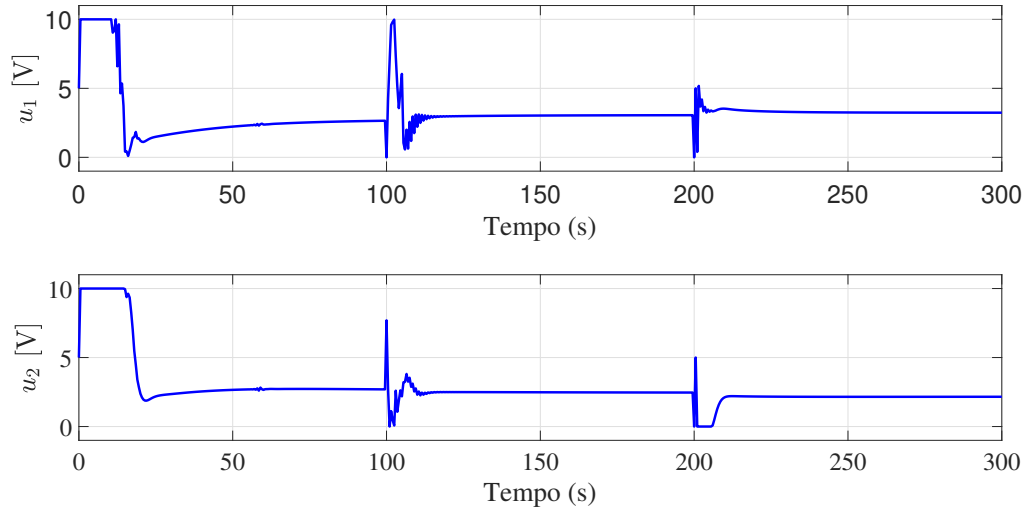
Fonte: Autor.

5.3.2.1 Cenário 1 - Seguimento de Referência

Neste cenário, o controlador preditivo baseado em modelo *fuzzy* evolutivo foi submetido à tarefa de rastrear uma trajetória de referência constante expressa matematicamente pela seguinte função periódica $\omega(t)$:

$$\omega(t + nT) = \begin{cases} 100, & 0 \leq t \leq 500 \text{ s}, \\ 60, & 500 \text{ s} < t \leq 1000 \text{ s} \end{cases} \quad (5.9)$$

Figura 39 – Experimento 2: sinais de controle u_1 [V] e u_2 [V] aplicados ao processo (cenário 3).



Fonte: Autor.

onde $n = 1, 2, 3$, $T = 100$ s é o período de repetição. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros neste cenário são apresentados nas figuras 43, 44 e 45, respectivamente. O comportamento da saída e sinal de controle são apresentados na figura 46.

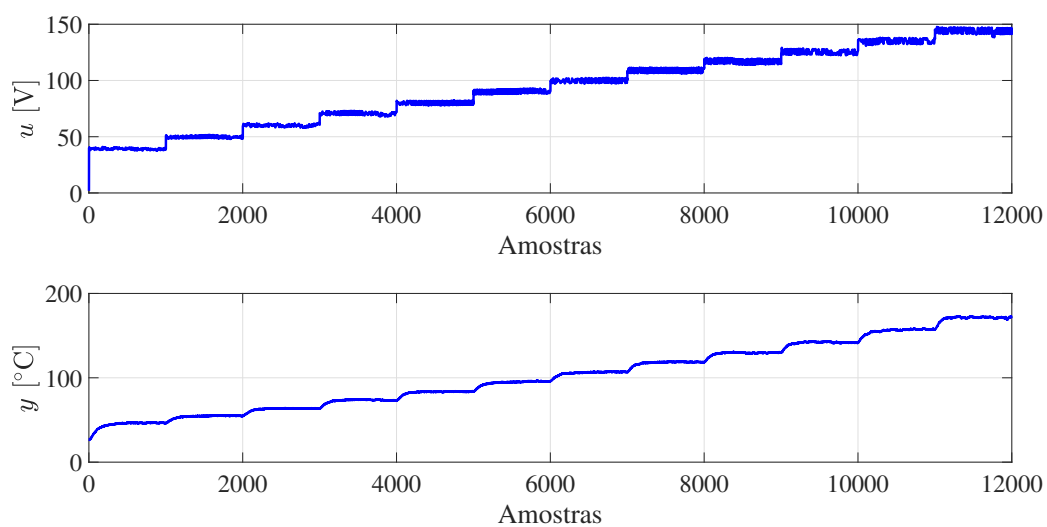
5.3.2.2 Cenário 2 - Rejeição à Distúrbios

Neste cenário, a capacidade do controlador preditivo baseado em modelo *fuzzy* evolutivo é avaliada quanto à capacidade de rejeitar distúrbios constantes de saída. Para isso submete-se o controlador ao rastreo de uma trajetória de referência constante $\omega(t) = 100$ e aplica-se um distúrbio de saída na forma de um confinando térmico do processo no instante $t = 100$ s. Os resultados obtidos para evolução das regras, evolução dos centros e evolução dos parâmetros são apresentados nas figuras 47, 48 e 49, respectivamente. O comportamento da saída e sinal de controle são apresentados na figura 50.

Figura 40 – Experimento 3: Processo SISO Não Linear Térmico.

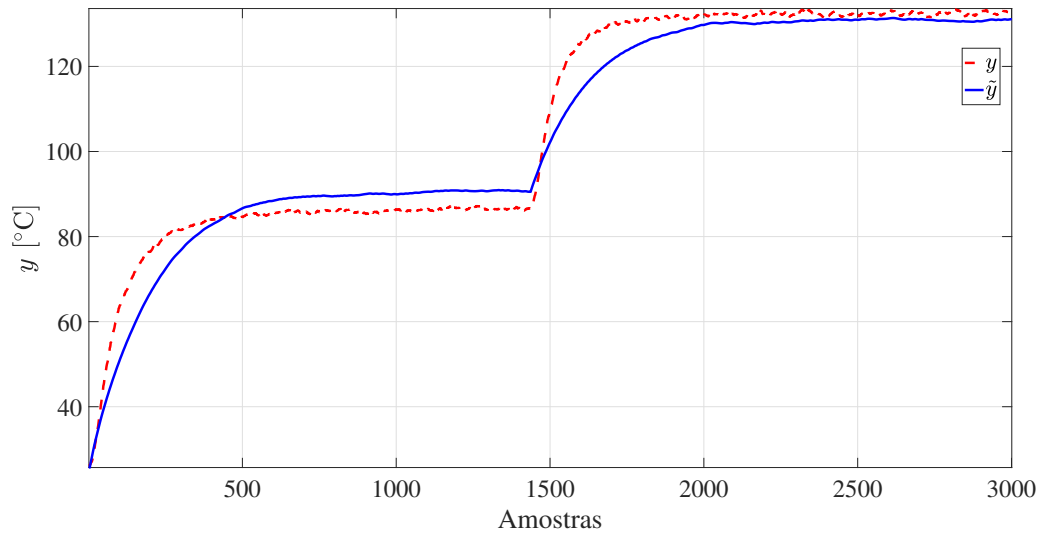


Fonte: Filho (2017).

Figura 41 – Experimento 3: dados de entrada u [V] e saída y [°C] do processo utilizados para identificação do modelo *fuzzy* TS inicial.

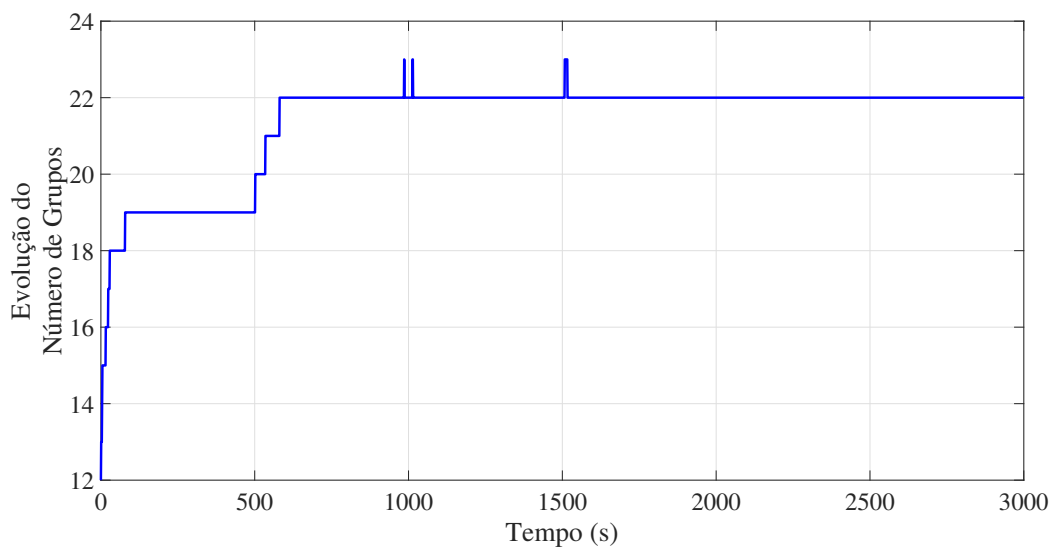
Fonte: Autor.

Figura 42 – Experimento 3: validação do modelo *fuzzy* TS inicial, onde a linha contínua em azul indica a saída do modelo $\tilde{y}$ [°C], enquanto que a linha tracejada vermelha representa a saída real y [°C].

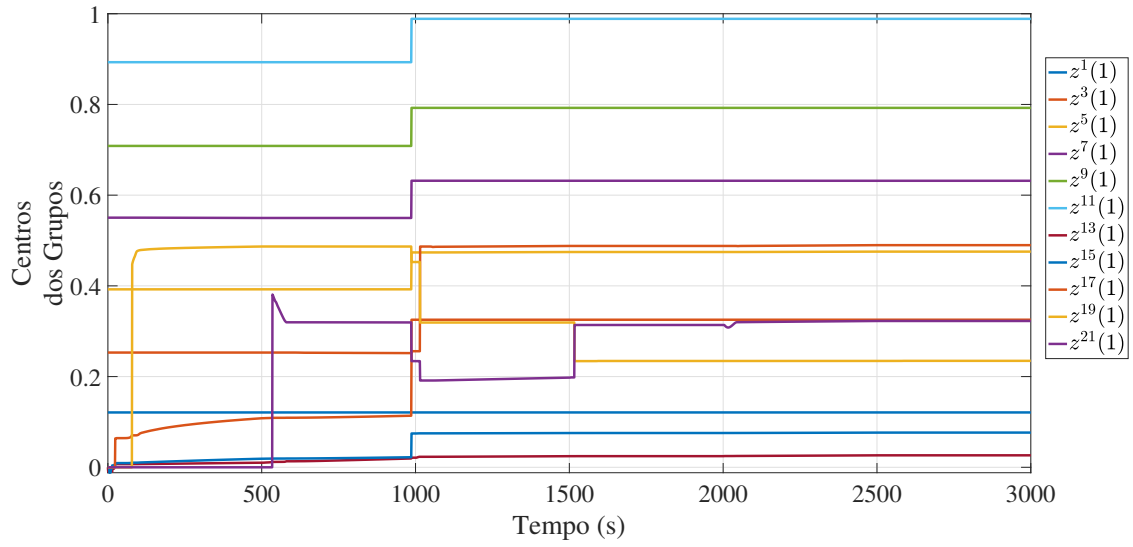


Fonte: Autor.

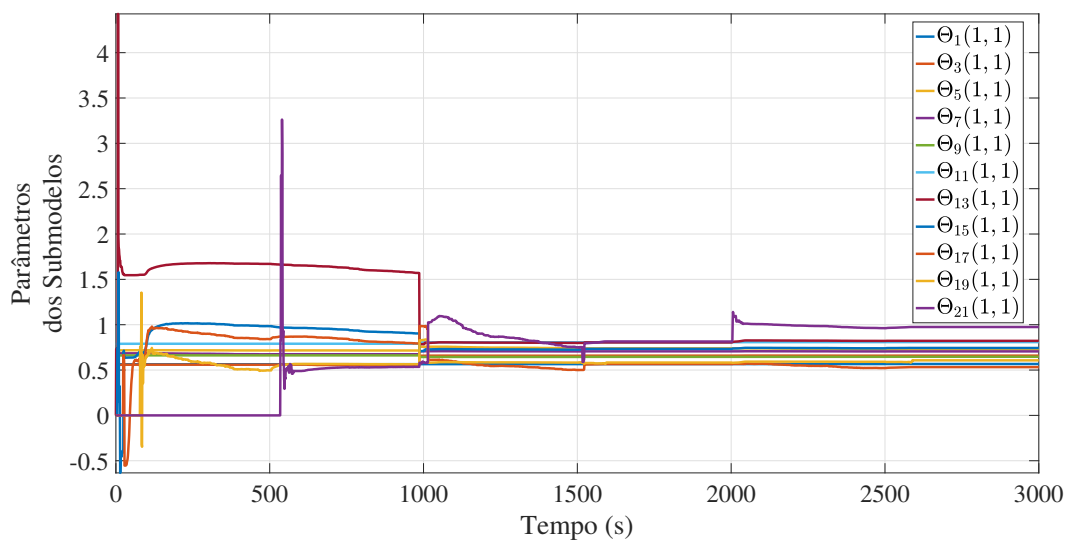
Figura 43 – Experimento 3: evolução do número de grupos (regras) $\tilde{c}$ (cenário 1).



Fonte: Autor.

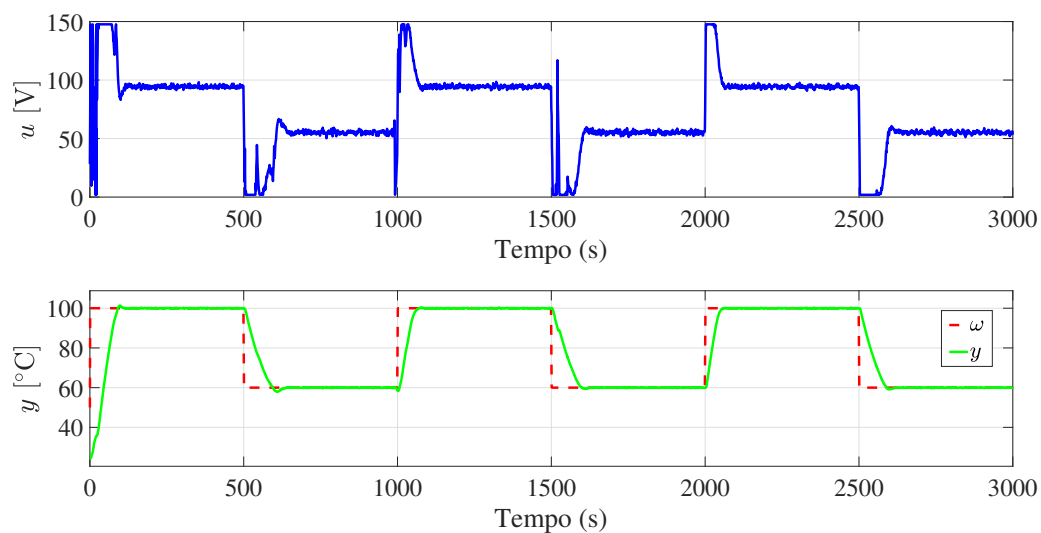
Figura 44 – Experimento 3: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 1).

Fonte: Autor.

Figura 45 – Experimento 3: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 1).

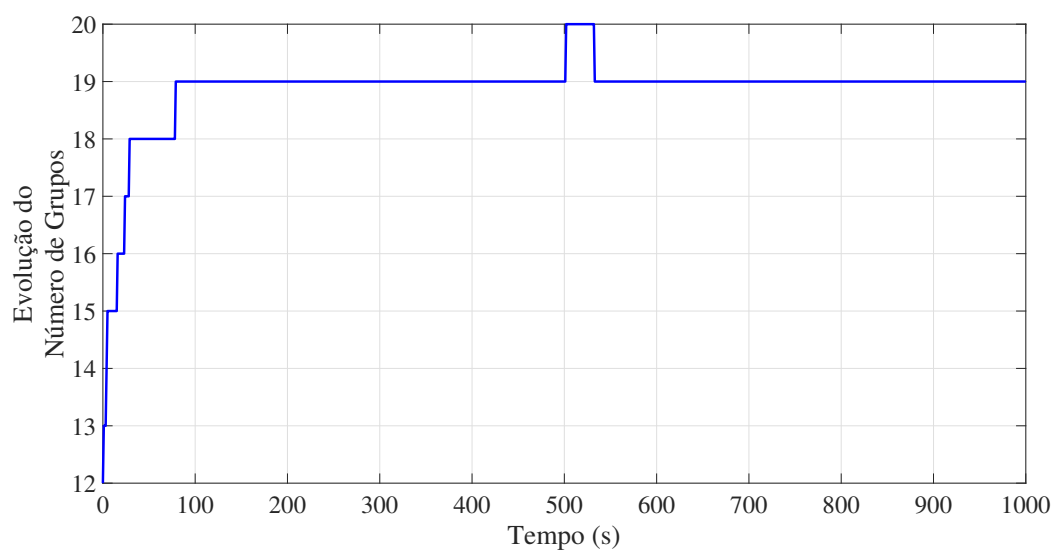
Fonte: Autor.

Figura 46 – Experimento 3: resposta da saída y [°C] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado u [V] (cenário 1).

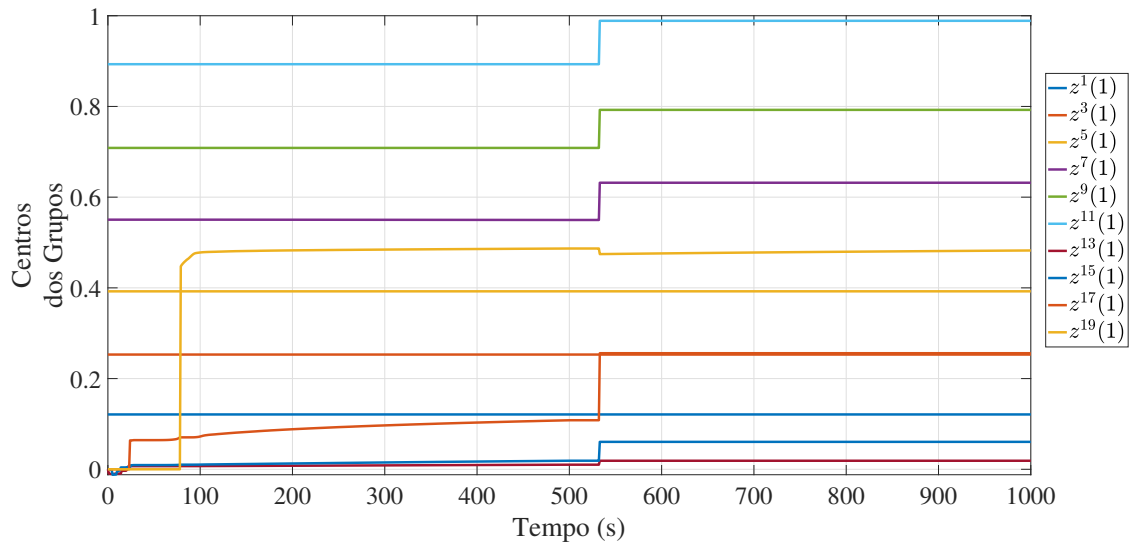


Fonte: Autor.

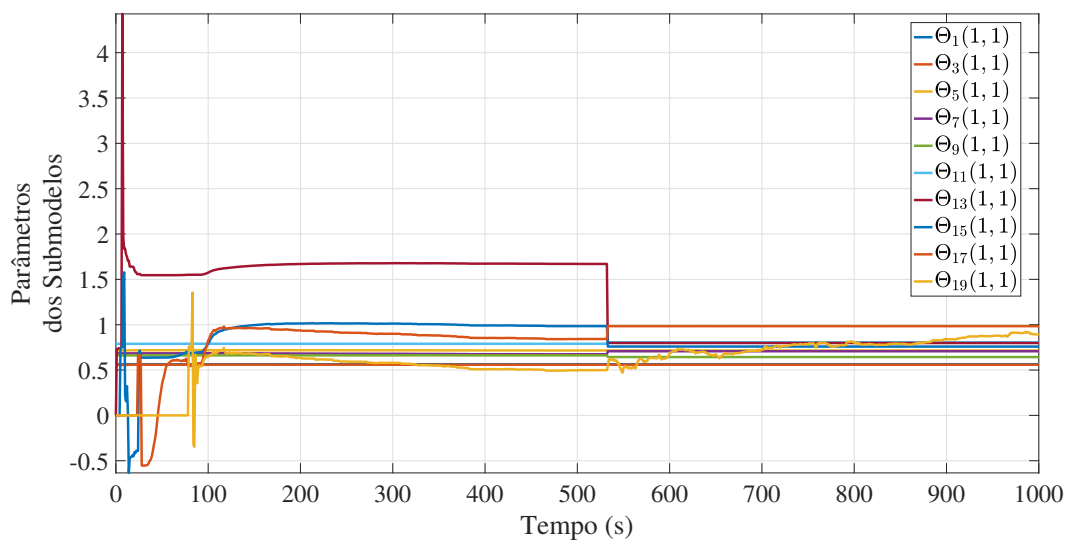
Figura 47 – Experimento 3: evolução do número de grupos (regras) $\tilde{c}$ (cenário 2).



Fonte: Autor.

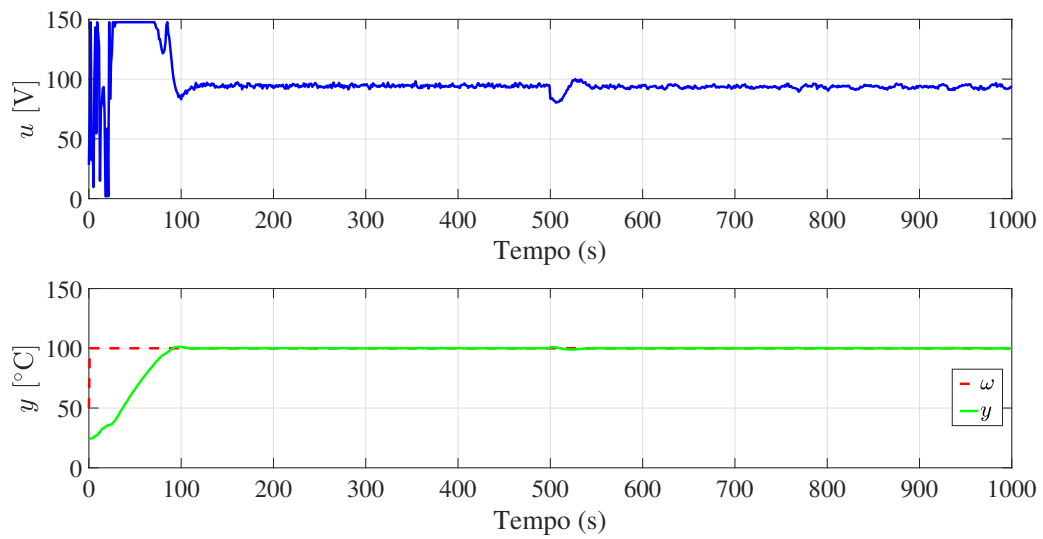
Figura 48 – Experimento 3: evolução dos centros dos grupos (regras) $z^i(1)$ (cenário 2).

Fonte: Autor.

Figura 49 – Experimento 3: evolução dos parâmetros dos submodelos $\Theta_i(1,1)$ (cenário 2).

Fonte: Autor.

Figura 50 – Experimento 3: resposta da saída y [°C] (linha verde contínua) no seguimento da trajetória de referência ω (linha vermelha tracejada) e sinal de controle aplicado u [V] (cenário 2).



Fonte: Autor.

6 Considerações Finais

Nesta dissertação foi proposta uma metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo. A técnica de controle utilizada foi o PNMPC que estima as matrizes de predição com base em uma aproximação do modelo de predição não linear do processo e calcula o sinal de controle solucionando um QP. O modelo de predição, por sua vez, é atualizado a cada instante de amostragem a partir de uma versão evolutiva do algoritmo de agrupamento *fuzzy* GK e da técnica RFWLS. Contudo, para obter o modelo *fuzzy* inicial são realizados procedimentos em batelada com as técnicas de agrupamento *fuzzy* GK e de identificação FWLS.

A importância de uma boa estimativa de modelo *fuzzy* inicial encontra-se na necessidade que o controlador proposto possui de utilizar o modelo no cálculo do sinal de controle. Esse modelo inicial, por sua vez, deve ser capaz de garantir a estabilidade nos instantes iniciais da operação em tempo real. Uma vez que o modelo *fuzzy* evolutivo tende a representar com maior exatidão a dinâmica do processo que deseja-se controlar, o desempenho do controlador proposto no seguimento de referências também é aprimorado ao longo da operação. Os resultados obtidos na aplicação da metodologia em processos *benchmarks* garantem, por exemplo, que mesmo com mudanças na dinâmica do processo durante a operação em tempo real, o modelo *fuzzy* evolutivo atualiza seus parâmetros e estrutura de modo a compensar isto. As principais contribuições deste trabalho são listadas a seguir:

- Proposta de uma nova metodologia de controle preditivo baseado em modelo *fuzzy* evolutivo, através da combinação das técnicas PNMPC, de uma versão evolutiva do agrupamento *fuzzy* GK e do RFWLS;
- Contribuição ao estudo do controle evolutivo, ou seja, controladores inteligentes cujos parâmetros e estrutura são modificados a partir do conhecimento extraído de processos que desejam-se controlar;
- Contribuição ao estudo do controle preditivo evolutivo.

6.1 Propostas de Trabalhos Futuros

O desenvolvimento de técnicas de controle avançado é um trabalho contínuo e árduo. Graças a diversidade de aplicações, bem como as dificuldades oriundas de ambientes industriais, técnicas existentes e já consolidadas vêm sendo aprimoradas a fim de atender novas demandas por qualidade e desempenho. Um exemplo claro deste

fato é o desenvolvimento desta dissertação. Essa, por sua vez, se insere em um contexto relativamente novo, onde questões importantes acerca do controlador ainda devem ser exploradas. Portanto, esse trabalho finaliza com as seguintes propostas e recomendações para pesquisas futuras:

- Garantir a estabilidade teórica do controlador preditivo baseado em modelo *fuzzy* evolutivo;
- Avaliar e comparar o desempenho de diferentes técnicas NMPC com o controlador proposto;
- Avaliar e comparar o desempenho de diferentes técnicas de modelagem *fuzzy* evolutiva com o controlador proposto;
- Considerar características de atrasos de transporte nos processos controlados e utilizar estratégias para compensação robusta dos mesmos;
- Garantir a robustez teórica do controlador preditivo baseado em modelo *fuzzy* evolutivo.

Referências

AGGARWAL, C. C.; REDDY, C. K. *Data clustering: algorithms and applications*. [S.l.]: CRC press, 2013. Citado 2 vezes nas páginas 42 e 43.

AGUIRRE, L. *Enciclopédia de automática: Controle e automação*. [S.l.]: Blucher, 2007. Citado 5 vezes nas páginas 22, 24, 27, 29 e 54.

AGUIRRE, L. *Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. [S.l.]: Editora UFMG, 2015. Citado na página 26.

ALAVI, S. M. M.; HAYES, M. J. Quantitative feedback design for a benchmark quadruple-tank process. In: *2006 IET Irish Signals and Systems Conference*. [S.l.: s.n.], 2006. p. 401–406. Citado na página 62.

ANGELOV, P. *Evolving Rule-based Models: A Tool for Design of Flexible Adaptive Systems*. Springer, 2002. Citado na página 40.

ANGELOV, P. *Autonomous Learning Systems: From Data Streams to Knowledge in Real-Time*. [S.l.]: Wiley, 2013. Citado 5 vezes nas páginas 20, 40, 42, 43 e 59.

ANGELOV, P.; BUSWELL, R. Evolving rule-based models: A tool for intelligent adaptation. In: *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*. [S.l.: s.n.], 2001. v. 2, p. 1062–1067 vol.2. Citado na página 19.

ANGELOV, P.; FILEV, D. Simplets: a simplified method for learning evolving takagi-sugeno fuzzy models. In: *The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05*. [S.l.: s.n.], 2005. p. 1068–1073. Citado na página 45.

ANGELOV, P.; FILEV, D. P.; KASABOV, N. *Evolving Intelligent Systems: Methodology and Applications*. [S.l.]: Wiley-IEEE Press, 2010. Citado na página 20.

ANGELOV, P.; KASABOV, N. Evolving Computational Intelligence Systems. *International Workshop on Genetic Fuzzy Systems*, Granada, Spain, p. 76–82, 2005. Citado 2 vezes nas páginas 40 e 41.

ANGELOV, P.; ZHOU, X. Evolving fuzzy systems from data streams in real-time. In: *2006 International Symposium on Evolving Fuzzy Systems*. [S.l.: s.n.], 2006. p. 29–35. Citado na página 46.

ANGELOV, P. P.; FILEV, D. P. An approach to online identification of takagi-sugeno fuzzy models. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 34, n. 1, p. 484–498, Feb 2004. Citado 2 vezes nas páginas 20 e 45.

ANGELOV, P. P.; ZHOU, X. Evolving fuzzy-rule-based classifiers **From Data** streams. *IEEE Transactions on Fuzzy Systems*, v. 16, n. 6, p. 1462–1475, Dec 2008. Citado na página 46.

- ANSAY, P.; WERTZ, V. Model uncertainties in gpc: A systematic two-step design. In: *1997 European Control Conference (ECC)*. [S.l.: s.n.], 1997. p. 2057–2062. Citado na página 18.
- ARIÑO, C.; QUEROL, A.; SALA, A. Shape-independent model predictive control for takagi–sugeno fuzzy systems. *Engineering Applications of Artificial Intelligence*, v. 65, p. 493 – 505, 2017. Citado na página 19.
- ÅSTRÖM, K.; HÄGGLUND, T. *Automatic Tuning of PID Controllers*. [S.l.]: Instrument Soc. of America, 1988. Citado na página 22.
- AYDIN, E.; BONVIN, D.; SUNDMACHER, K. Computationally efficient nmppc for batch and semi-batch processes using parsimonious input parameterization. *Journal of Process Control*, v. 66, p. 12 – 22, 2018. Citado na página 17.
- BABUŠKA, R. *Fuzzy Modeling for Control*. [S.l.]: Springer Netherlands, 1998. (International Series in Intelligent Technologies). Citado 3 vezes nas páginas 19, 42 e 50.
- BEZDEK, J. C.; DUNN, J. C. Optimal fuzzy partitions: A heuristic for estimating the parameters in a mixture of normal distributions. *IEEE Transactions on Computers*, C-24, n. 8, p. 835–838, Aug 1975. Citado na página 45.
- BLAZIC, S.; DOVZAN, D.; SKRJANC, I. Robust evolving fuzzy adaptive control with input-domain clustering. *IFAC Proceedings Volumes*, v. 47, n. 3, p. 5387 – 5392, 2014. 19th IFAC World Congress. Citado na página 20.
- BLOEMEN, H. H. J.; BOOM, T. J. J. van den; VERBRUGGEN, H. B. Model-based predictive control for hammerstein systems. In: *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*. [S.l.: s.n.], 2000. v. 5, p. 4963–4968 vol.5. Citado na página 34.
- BOULKAIBET, I. et al. A new t-s fuzzy model predictive control for nonlinear processes. *Expert Systems with Applications*, v. 88, p. 132 – 151, 2017. Citado 3 vezes nas páginas 19, 62 e 63.
- CAMACHO, E.; BORDONS, C.; ALBA, C. *Model Predictive Control*. [S.l.]: Springer London, 2004. (Advanced Textbooks in Control and Signal Processing). Citado 6 vezes nas páginas 17, 19, 22, 23, 25 e 29.
- CANNON, M. Efficient nonlinear model predictive control algorithms. *Annual Reviews in Control*, v. 28, n. 2, p. 229 – 237, 2004. Citado na página 36.
- CLARKE, D.; MOHTADI, C. Properties of generalized predictive control. *Automatica*, v. 25, n. 6, p. 859 – 875, 1989. Citado na página 18.
- CLARKE, D.; MOHTADI, C.; TUFFS, P. Generalized predictive control—part i. the basic algorithm. *Automatica*, v. 23, n. 2, p. 137 – 148, 1987. Citado 2 vezes nas páginas 18 e 28.
- CLARKE, D. W.; SCATTOLINI, R. Constrained receding-horizon predictive control. *IEE Proceedings D - Control Theory and Applications*, v. 138, n. 4, p. 347–354, July 1991. Citado na página 19.

- COSTA, E. B. M. *Controle PID Fuzzy Adaptativo com Estabilidade Robusta por Enxame de Partículas Multiobjetivo*. Dissertação (Mestrado) — Universidade Federal do Maranhão, São Luís, 2016. Citado na página 62.
- DATTA, A.; OCHOA, J. Adaptive internal model control: Design and stability analysis. *Automatica*, v. 32, n. 2, p. 261 – 266, 1996. Citado na página 18.
- DOVŽAN, D.; ŠKRJANC, I. Recursive clustering based on a gustafson–kessel algorithm. *Evolving Systems*, v. 2, n. 1, p. 15–24, Mar 2011. Citado 2 vezes nas páginas 20 e 57.
- ELIAS, T. de A. et al. Optimal solar collectors defocusing based on maximum temperature. In: *2018 9th International Renewable Energy Congress (IREC)*. [S.l.: s.n.], 2018. p. 1–5. Citado 2 vezes nas páginas 18 e 19.
- ESKI, İ.; TEMÜRLenk, A. Design of neural network-based control systems for active steering system. *Nonlinear Dynamics*, v. 73, n. 3, p. 1443–1454, Aug 2013. Citado na página 19.
- FILHO, O. D. R. *Modelagem Nebulosa Evolutiva: novas topologias e algoritmos de aprendizagem*. Tese (Doutorado) — Universidade Federal de Minas Gerais, São Luís, 2011. Citado 2 vezes nas páginas 40 e 41.
- FILHO, O. D. R. *Modelagem baseada em agrupamento nebuloso evolutivo de máxima verossimilhança aplicada a sistemas dinâmicos operando em ambiente não-estacionário*. Tese (Doutorado) — Universidade Federal do Maranhão, São Luís, 2017. Citado 5 vezes nas páginas 41, 42, 45, 62 e 83.
- FINDEISEN, R.; ALLGÖWER, F. *An Introduction to Nonlinear Model Predictive Control*. 2002. Citado na página 17.
- GAALOUL, A.; M'SAHLI, F. Output feedback controller of a multivariable process. In: *2009 6th International Multi-Conference on Systems, Signals and Devices*. [S.l.: s.n.], 2009. p. 1–6. Citado na página 62.
- GHAFFARI, V.; NAGHAVI, S. V.; SAFAVI, A. Robust model predictive control of a class of uncertain nonlinear systems with application to typical cstr problems. *Journal of Process Control*, v. 23, n. 4, p. 493 – 499, 2013. Citado na página 62.
- GUSTAFSON, D. E.; KESSEL, W. C. Fuzzy clustering with a fuzzy covariance matrix. In: *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*. [S.l.: s.n.], 1978. p. 761–766. Citado na página 45.
- JOHANSSON, K. H. The quadruple-tank process: a multivariable laboratory process with an adjustable zero. *IEEE Transactions on Control Systems Technology*, v. 8, n. 3, p. 456–465, May 2000. Citado 2 vezes nas páginas 62 e 72.
- JÚNIOR, A. P. A.; SERRA, G. L. O. Controle preditivo baseado em modelo via algoritmo recursivo de realização de auto-sistema. *Conferência Brasileira de Dinâmica, Controle e Aplicações (DINCON)*, 2017. Citado na página 19.
- JÚNIOR, A. P. A.; SERRA, G. L. O. Metodologia de controle preditivo baseado em modelo fuzzy evolutivo. *Congresso Brasileiro de Automática (CBA)*, 2018. Citado na página 19.

- KALHOR, A.; ARAABI, B.; LUCAS, C. An online predictor model as adaptive linear model and evolving takagi-sugeno model. p. 1–6, 01 2010. Citado na página 46.
- KALHOR, A.; ARAABI, B.; LUCAS, C. A new systematic design for habitually linear evolving ts fuzzy model. v. 39, p. 1725–1736, 02 2012. Citado na página 46.
- KALHOR, A.; ARAABI, B. N.; LUCAS, C. Evolving takagi–sugeno fuzzy model based on switching to neighboring models. *Applied Soft Computing*, v. 13, n. 2, p. 939 – 946, 2013. Citado na página 46.
- KARAMANAKOS, P. et al. Variable switching point predictive torque control of induction machines. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, v. 2, n. 2, p. 285–295, June 2014. Citado na página 17.
- KASABOV, N. Evolving connectionist systems for adaptive learning and knowledge discovery: method, tools, applications. *IEEE International Conference on Intelligent Systems*, v. 1, p. 24–28, 2002. Citado na página 40.
- KASABOV, N. K.; SONG, Q. Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, v. 10, n. 2, p. 144–154, Apr 2002. Citado 2 vezes nas páginas 20 e 46.
- KEYSER, R. D.; CAUWENBERGHE, A. V. Extended prediction self-adaptive control. *IFAC Proceedings Volumes*, v. 18, n. 5, p. 1255 – 1260, 1985. 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation, York, UK, 3-7 July. Citado na página 18.
- KILLIAN, M.; KOZEK, M. Optimal partitioning of a boiler-turbine unit for fuzzy model predictive control. *IFAC-PapersOnLine*, v. 50, n. 1, p. 2011 – 2016, 2017. 20th IFAC World Congress. Citado na página 19.
- KILLIAN, M.; KOZEK, M. Implementation of cooperative fuzzy model predictive control for an energy-efficient office building. *Energy and Buildings*, v. 158, p. 1404 – 1416, 2018. Citado na página 19.
- KOUVARITAKIS, B.; CANNON, M.; ROSSITER, J. A. Non-linear model based predictive control. *International Journal of Control*, Taylor and Francis, v. 72, n. 10, p. 919–928, 1999. Citado na página 19.
- KOUVARITAKIS, B.; ROSSITER, J. A.; CHANG, A. O. T. Stable generalised predictive control: an algorithm with guaranteed stability. *IEE Proceedings D - Control Theory and Applications*, v. 139, n. 4, p. 349–362, July 1992. Citado na página 19.
- ŁAWRYŃCZUK, M. Nonlinear predictive control of dynamic systems represented by wiener–hammerstein models. *Nonlinear Dynamics*, v. 86, n. 2, p. 1193–1214, Oct 2016. Citado na página 34.
- LEE, J. H.; MORARI, M.; GARCIA, C. E. State-space interpretation of model predictive control. *Automatica*, v. 30, n. 4, p. 707 – 717, 1994. Citado na página 18.
- LIMA, D. M.; SANTOS, T. L. M.; NORMEY-RICO, J. E. Robust nonlinear predictor for dead-time systems with input nonlinearities. *Journal of Process Control*, v. 27, p. 1 – 14, 2015. Citado na página 18.

- LIU, Z. et al. A model-based predictive direct power control for traction line-side converter in high-speed railway. *IEEE Transactions on Industry Applications*, v. 53, n. 5, p. 4934–4943, Sept 2017. Citado na página 17.
- LOPEZ, L. G.; NORMEY-RICO, J. E. Practical nonlinear model predictive control of a 5 mw wind turbine. In: *2018 9th International Renewable Energy Congress (IREC)*. [S.l.: s.n.], 2018. p. 1–6. Citado na página 18.
- LU, Y.; ARKUN, Y. A scheduling quasi-minmax mpc for lpv systems. In: *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*. [S.l.: s.n.], 1999. v. 4, p. 2272–2276 vol.4. Citado na página 34.
- LUENBERGER, D.; YE, Y. *Linear and Nonlinear Programming*. [S.l.]: Springer US, 2008. (International Series in Operations Research & Management Science). Citado na página 29.
- LUGHOFFER, E. *Evolving Fuzzy Systems - Methodologies, Advanced Concepts and Applications*. [S.l.]: Springer Berlin Heidelberg, 2011. (Studies in Fuzziness and Soft Computing). Citado 4 vezes nas páginas 19, 40, 47 e 48.
- LUGHOFFER, E. et al. Generalized smart evolving fuzzy systems. *Evolving Systems*, Springer, v. 6, n. 4, p. 269–292, 2015. Citado na página 41.
- LUGHOFFER, E.; KLEMENT, E. P. Flexfis: A variant for incremental learning of takagi-sugeno fuzzy systems. In: *The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05*. [S.l.: s.n.], 2005. p. 915–920. Citado na página 20.
- LUGHOFFER, E. D. Flexfis: A robust incremental learning approach for evolving takagi-sugeno fuzzy models. *IEEE Transactions on fuzzy systems*, IEEE, v. 16, n. 6, p. 1393–1410, 2008. Citado 2 vezes nas páginas 41 e 46.
- MACIEJOWSKI, J. *Predictive Control: With Constraints*. [S.l.]: Prentice Hall, 2002. (Pearson Education). Citado 2 vezes nas páginas 18 e 22.
- MACIEL, L.; GOMIDE, F.; BALLINI, R. MIMO evolving participatory learning fuzzy modeling. In: IEEE. *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*. [S.l.], 2012. p. 1–8. Citado na página 42.
- MAHMOUDI, H.; ALEENEJAD, M.; AHMADI, R. A new multiobjective modulated model predictive control method with adaptive objective prioritization. *IEEE Transactions on Industry Applications*, v. 53, n. 2, p. 1188–1199, March 2017. Citado na página 17.
- MANTHANWAR, A. M.; SAKIZLIS, V.; PISTIKOPOULOS, E. N. Robust parametric predictive control design for polytopically uncertain systems. In: *Proceedings of the 2005, American Control Conference, 2005*. [S.l.: s.n.], 2005. p. 3994–3999 vol. 6. Citado na página 34.
- NOCEDAL, J.; WRIGHT, S. *Numerical Optimization*. [S.l.]: Springer New York, 2006. (Springer Series in Operations Research and Financial Engineering). Citado na página 36.
- NORMEY-RICO, J. *Control of Dead-time Processes*. [S.l.]: Springer London, 2007. (Advanced Textbooks in Control and Signal Processing). Citado na página 19.

- NORMEY-RICO, J. E.; CAMACHO, E. F. A unified approach to design dead-time compensators for stable and integrative processes with dead-time. In: *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)*. [S.l.: s.n.], 2001. v. 3, p. 2267–2272 vol.3. Citado na página 19.
- NORMEY-RICO, J. E.; LIMA, D. M.; SANTOS, T. L. Robustness of nonlinear mpc for dead-time processes**this work was financed by cnpq-brasil (conselho nacional de desenvolvimento científico e tecnológico). *IFAC-PapersOnLine*, v. 48, n. 23, p. 332 – 341, 2015. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015. Citado na página 18.
- OVIEDO, J.; VANDEWALLE, J.; WERTZ, V. *Fuzzy Logic, Identification and Predictive Control*. [S.l.]: Springer London, 2004. (Advances in Industrial Control). Citado 4 vezes nas páginas 19, 29, 33 e 62.
- PEDERSEN, T.; AARSNES, U. J. F.; GODHAVN, J.-M. Flow and pressure control of underbalanced drilling operations using nmpe. *Journal of Process Control*, v. 68, p. 73 – 85, 2018. Citado na página 17.
- PEDRYCZ, W. *Knowledge-based clustering: from data to information granules*. [S.l.]: John Wiley & Sons, 2005. Citado na página 45.
- PETERSEN, K. B.; PEDERSEN, M. S. *The Matrix Cookbook*. [S.l.]: Technical University of Denmark, 2012. Version 20121115. Citado na página 58.
- PLUCENIO, A. *Desenvolvimento de Técnicas de Controle Não Linear para Elevação e Fluidos Multifásicos*. 160 f. Tese (Doutorado em Engenharia de Automação e Sistemas) — Universidade Federal de Santa Catarina, Florianópolis, 2010. Citado 4 vezes nas páginas 17, 18, 36 e 37.
- RAMÍREZ, D. et al. Min-max mpc based on a computationally efficient upper bound of the worst case cost. *Journal of Process Control*, v. 16, n. 5, p. 511 – 519, 2006. Citado na página 34.
- RASTEGAR, S.; ARAÚJO, R.; MENDES, J. A new approach for online t-s fuzzy identification and model predictive control of nonlinear systems. *Journal of Vibration and Control*, v. 22, n. 7, p. 1820–1837, 2016. Citado 2 vezes nas páginas 19 e 20.
- RICHALET, J. et al. Model predictive heuristic control: Applications to industrial processes. *Automatica*, v. 14, n. 5, p. 413 – 428, 1978. Citado na página 18.
- ROBINSON, B. D.; CLARKE, D. W. Robustness effects of a prefilter in generalised predictive control. *IEE Proceedings D - Control Theory and Applications*, v. 138, n. 1, p. 2–8, Jan 1991. Citado na página 18.
- ROINILA, T.; VILKKO, M.; JAATINEN, A. Corrected mathematical model of quadruple tank process. *IFAC Proceedings Volumes*, v. 41, n. 2, p. 11678 – 11683, 2008. 17th IFAC World Congress. Citado na página 62.
- ROSINOVÁ, D.; KOZÁKOVÁ, A. Decentralized robust control of mimo systems: Quadruple tank case study. *IFAC Proceedings Volumes*, v. 45, n. 11, p. 72 – 77, 2012. 9th IFAC Symposium Advances in Control Education. Citado na página 62.

- ROSSITER, J. *Model-Based Predictive Control: A Practical Approach*. [S.l.]: CRC Press, 2017. (Control Series). Citado 2 vezes nas páginas 23 e 28.
- ROUBOS, H.; SETNES, M.; ABONYI, J. Learning fuzzy classification rules from data. In: JOHN, R.; BIRKENHEAD, R. (Ed.). *Developments in Soft Computing*. Heidelberg: Physica-Verlag HD, 2001. p. 108–115. Citado na página 19.
- SMITH, J. M. Closer control of loops with dead time. *Chemical Engineering Progress*, v. 53, n. 5, p. 217 – 219, 1957. Citado na página 22.
- SOETERBOEK, R. *Predictive control: a unified approach*. [S.l.]: Prentice Hall, 1992. (Prentice-Hall international series in systems and control engineering). Citado na página 18.
- SOLEIMANI-B, H.; LUCAS, C.; ARAABI, B. N. Recursive gath-geva clustering as a basis for evolving neuro-fuzzy modeling. In: *International Conference on Fuzzy Systems*. [S.l.: s.n.], 2010. p. 1–7. Citado na página 59.
- SOROOSH, S. A.; KALHOR, A. Evolving takagi-sugeno model based on online gustafson-kessel algorithm and kernel recursive least square method. In: *2014 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*. [S.l.: s.n.], 2014. p. 1–8. Citado 6 vezes nas páginas 20, 45, 46, 56, 58 e 59.
- SU, C.; MA, J. Nonlinear predictive control using fuzzy hammerstein model and its application to cstr process. *AASRI Procedia*, v. 3, p. 8 – 13, 2012. Conference on Modelling, Identification and Control. Citado na página 62.
- TAVAKOLI, A.; NEGNEVITSKY, M.; MUTTAQI, K. M. A decentralized model predictive control for operation of multiple distributed generators in an islanded mode. *IEEE Transactions on Industry Applications*, v. 53, n. 2, p. 1466–1475, March 2017. Citado na página 17.
- TSAO, E. C.-K.; BEZDEK, J. C.; PAL, N. R. Fuzzy kohonen clustering networks. *Pattern Recognition*, v. 27, n. 5, p. 757 – 764, 1994. Citado na página 45.
- WANG, L. *A Course in Fuzzy Systems and Control*. [S.l.]: Prentice Hall PTR, 1997. Citado 2 vezes nas páginas 40 e 41.
- WOLF, I. J.; MARQUARDT, W. Fast nmpc schemes for regulatory and economic nmpc – a review. *Journal of Process Control*, v. 44, p. 162 – 183, 2016. Citado na página 17.
- XIE, S. et al. Weighted-coupling cstr modeling and model predictive control with parameter adaptive correction for the goethite process. *Journal of Process Control*, v. 68, p. 254 – 267, 2018. Citado na página 62.
- YASUNOBU, S.; MIYAMOTO, S. A predictive fuzzy control for automatic train operation. *Japanese Systems and Control*, v. 28, n. 10, p. 605–613, 1984. Citado na página 19.
- YDSTIE, B. Extended horizon adaptive control. *IFAC Proceedings Volumes*, v. 17, n. 2, p. 911 – 915, 1984. 9th IFAC World Congress: A Bridge Between Control Science and Technology, Budapest, Hungary, 2-6 July 1984. Citado na página 18.

- ZADEH, L. The concept of a linguistic variable and its application to approximate reasoning—i. *Information Sciences*, v. 8, n. 3, p. 199 – 249, 1975. Citado na página 19.
- ZDESAR, A.; DOVZAN, D.; ŠKRJANC, I. A 2 dof predictive control based on evolving fuzzy model. In: *2014 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*. [S.l.: s.n.], 2014. p. 1–6. Citado na página 20.
- ZHANG, F.; WU, X.; SHEN, J. Extended state observer based fuzzy model predictive control for ultra-supercritical boiler-turbine unit. *Applied Thermal Engineering*, v. 118, p. 90 – 100, 2017. Citado 2 vezes nas páginas 19 e 34.
- ZHENG, Q. et al. A turboshaft engine nmpc scheme for helicopter autorotation recovery maneuver. *Aerospace Science and Technology*, v. 76, p. 421 – 432, 2018. Citado na página 17.
- ZOU, Z. et al. Design and simulation of nonlinear hammergein systems dynamic matrix control algorithm. In: *Proceedings of the 6th World Congress on Intelligent Control and Automation*. [S.l.: s.n.], 2006. v. 1, n. s.n, p. 1981–1985. Citado na página 34.