

UNIVERSIDADE FEDERAL DE ITAJUBÁ

PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA

PROJETO DE SISTEMAS DE CONTROLE COM MODELAGEM
TAKAGI-SUGENO E IMPLEMENTAÇÃO DE CONTROLADORES
FUZZY COM RETROAÇÃO DE ESTADOS

TIAGO SOARES VÍTOR

ITAJUBÁ, DEZEMBRO DE 2011

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
EM ENGENHARIA ELÉTRICA

TIAGO SOARES VÍTOR

PROJETO DE SISTEMAS DE CONTROLE COM MODELAGEM
TAKAGI-SUGENO E IMPLEMENTAÇÃO DE CONTROLADORES
FUZZY COM RETROAÇÃO DE ESTADOS

Dissertação submetida ao Programa de Pós-Graduação em
Engenharia Elétrica como parte dos requisitos para obtenção
do Título de Mestre em Ciências em Engenharia Elétrica.

Área de Concentração: Automação e Sistemas Industriais.

Orientador: Prof. Dr. Carlos Alberto Murari Pinheiro.

DEZEMBRO 2011

ITAJUBÁ - MG

Ficha catalográfica elaborada pela Biblioteca Mauá –
Bibliotecária Margareth Ribeiro- CRB_6/1700

V845p

Vítor, Tiago Soares

Projeto de Sistemas de Controle com Modelagem Takagi-Sugeno e Implementação de Controladores FUZZY com Retroação de Estados / Tiago Soares Vítor. -- Itajubá, (MG) : [s.n.], 2011.
86 p. : il.

Orientador: Prof. Dr. Carlos Alberto Murari Pinheiro.
Dissertação (Mestrado) – Universidade Federal de Itajubá.

1. Sistemas não-lineares. 2. Modelagem Fuzzy Takagi-Sugeno.
3. Controlador Fuzzy. 4. Desigualdades lineares matriciais(LMI).
I. Pinheiro, Carlos Alberto Murari, orient. II. Universidade Federal de Itajubá. III. Título.

Dedicatória

Dedico este trabalho aos meus pais, David e Elenice, pelo incentivo constante e encorajador em todas etapas da minha vida.

Agradecimentos

Aos meus pais, pelo amor presente em todos os momentos.

A minha irmã, Fernanda, pela amizade e simpatia.

A minha esposa, Daniela, por fazer parte da minha vida.

A todos os amigos, pelo incentivo e apoio demonstrados durante este tempo.

Ao meu orientador Prof. Carlos Alberto Murari Pinheiro por ter apresentado a diretriz de como realizar o projeto e pela paciente ajuda no decorrer das atividades.

A todos os professores por terem contribuído para o meu desenvolvimento intelectual, possibilitando que este trabalho fosse finalizado.

A CNPQ pela bolsa de mestrado.

A Deus, pois Ele mesmo é quem a todos dá vida, respiração e tudo mais.

*“Boa e bela coisa é comer e beber e gozar cada um do bem de todo o seu trabalho,
com que se afadigou debaixo do sol, durante os poucos dias da vida que Deus lhe deu;
porque esta é a sua porção.”*

Eclesiastes 5:18

Resumo

Este trabalho realiza uma implementação experimental de uma malha de controle através de um sistema computadorizado, onde é utilizada uma metodologia de projeto que visa à estabilização e à compensação de processos com características não-lineares. Será empregado o conceito de compensação paralela distribuída para projetar um controlador *fuzzy* como compensador do sistema. O processo a ser controlado é representado por um modelo *fuzzy* Takagi-Sugeno. Este procedimento é conceitualmente bem fundamentado, consistindo na geração de regras de controle a partir do modelo difuso correspondente do processo a ser controlado. A análise de estabilidade e a obtenção da compensação adequada do sistema podem ser reduzidas a problemas de desigualdades lineares matriciais (LMI), que são resolvidos por ferramentas computacionais específicas. A implantação realizada será ilustrada através de simulações e ensaios em um sistema de nível real.

Palavras-chave: Sistemas não-lineares, modelagem fuzzy Takagi-Sugeno, controlador *fuzzy*, análise de estabilidade por Lyapunov, desigualdades lineares matriciais (LMI).

Abstract

This work presents an experimental implementation of a control loop through a computerized system. It is used a design methodology for stabilization and compensation of processes with nonlinear characteristics. The concept of parallel distributed compensation for designing a fuzzy controller is employed as a compensator of the plant. The process to be controlled is represented by a Takagi-Sugeno fuzzy model. This procedure is conceptually well-founded. The idea is to generate the control rules from the fuzzy model corresponding to the process to be controlled. The stability analysis and appropriate compensation design can be reduced to linear matrix inequality (LMI) problems, which are solved by specific computational tools. The implementation performed will be illustrated through simulations and tests on a real level system.

Keywords: Nonlinear plant, Takagi-Sugeno fuzzy model, fuzzy controller, stability analysis by Lyapunov, linear matrix inequality (LMI).

Lista de Figuras

Figura 1 - Configuração básica dos sistemas fuzzy puros.....	6
Figura 2 - Configuração básica dos sistemas fuzzy com fuzzificador e defuzzificador.....	6
Figura 3 – Configuração básica do sistema fuzzy Takagi-Sugeno.....	7
Figura 4 - Sistema fuzzy TS do Exemplo 1.....	11
Figura 5 - Funções de pertinência da variável de entrada do Exemplo 1.....	12
Figura 6 - Mudança suave de Saída1 para Saída2.....	12
Figura 7 - Systema fuzzy TS do Exemplo 2.....	13
Figura 8 - Funções de pertinência do Exemplo 2.	14
Figura 9 - Operação do sistema fuzzy TS do Exemplo 2 (MathWorks, 2011).	15
Figura 10 - Superfície de saída para Exemplo 2.....	16
Figura 11 – Funções pertinência de entrada do Exemplo 3.....	18
Figura 12 – Exemplo de função ANFIS	19
Figura 13 – Funções de pertinência do Exemplo 4.	23
Figura 14 - (a) Comportamento de $x(k+1) = A1x(k)$. (b) Comportamento $x(k+1) = A2x(k)$	24
Figura 15 – Técnica de projeto PDC – Compensação Paralela Distribuída.....	25
Figura 16 – Estrutura de projeto para controle fuzzy.	28
Figura 17 - Resposta do Exemplo 5 realizado o controle PDC.....	30
Figura 18 – Exemplo de um Sistema de Pêndulo Invertido.	36
Figura 19 - Funções de pertinência do modelo do pêndulo invertido.	37
Figura 20 – Respostas do sistema de pêndulo invertido para diferentes condições iniciais. ...	40
Figura 21 – Respostas do sistema de pêndulo invertido para diferentes condições iniciais. ...	40
Figura 22 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	42
Figura 23 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	42
Figura 24 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	44
Figura 25 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	44
Figura 26 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	46
Figura 27 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.	46
Figura 28 - Processo de Projeto.....	47
Figura 29 - Processo de projeto para projetos experimentais.....	48
Figura 30 - Bancada de ensaio.....	49

Figura 31 – Estrutura típica de um sistema de nível.	50
Figura 32 - Obtenção dos dados de entrada e saída para modelagem do sistema de nível.	51
Figura 34 – Dados do ensaio para modelagem e os resultantes do modelo obtido.	52
Figura 33 - Dados de treinamento para o ANFIS.	52
Figura 35 - Dados de ensaio para validação do modelo.	53
Figura 36 – Dados de ensaio para validação do modelo.	54
Figura 37 - Modelo fuzzy correspondente a planta de nível modelada.	54
Figura 38 – Funções de pertinência da variável $u(k)$	55
Figura 39 – de pertinência da variável $x1(k) = y(k)$	55
Figura 40 - Malha de controle convencional (Ogata, 2003).	60
Figura 42 – Componentes da malha de controle da planta de nível.	61
Figura 41 – Malha de controle fuzzy para a planta de nível.	61
Figura 43 – As leis de controle do controlador fuzzy TS para a planta de nível.	62
Figura 44 - Modelo fuzzy TS da planta de nível.	63
Figura 45 - Diagrama de classes do programa de simulação.	66
Figura 46 - Resposta real da malha de controle fuzzy para dois valores de referência de entrada.	67
Figura 47 - Resposta real do sistema do sistema de controle fuzzy com outros valores de referência.	68
Figura 48 – Respostas reais do sistema de controle fuzzy para uma seqüência de valores de entrada.	69
Figura 49 – Ensaio para o controlador fuzzy e uma malha de controle convencional.	70
Figura 50 - Resposta do sistema para dois valores baixos de referência de entrada.	71
Figura 51 - Resposta do sistema para dois valores de referência de entrada mais altos.	71
Figura 52 - Respostas do sistema de controle para uma seqüência de valores de referência de entrada.	72
Figura 53 - Ensaio para o controlador fuzzy e para uma malha de controle convencional. ...	73
Figura 54 - Sistema de controle fuzzy (programa principal)	84
Figura 55 - Sub-rotina das regras de controle fuzzy (equivalente ao Anexo 4 em MATLAB)	84
Figura 56 - Sub-rotina para leitura de dados em arquivo (ganhos e parâmetros das funções de pertinência)	85
Figura 57 - Sub-rotina das funções de pertinência de entrada.	85

Figura 58 - Subrotina das funções consequentes do controlador	86
Figura 59 - Interface gráfica do sistema de controle	86

Lista de Abreviaturas

PDC	-	<i>Parallel Distributed Compensation</i>
LMI	-	<i>Linear Matrix Inequality</i>
I/O	-	<i>Input/Output</i>
PID	-	Controlador Proporcional+Integral+Derivativo
TS	-	Takagi-Sugeno
ANFIS	-	<i>Adaptive Neuro-Fuzzy Inference Systems</i>
USB	-	<i>Universal Serial Bus</i>
BMI	-	<i>Bilinear Matrix Inequality</i>
DAQ	-	<i>Data Acquisition</i>

Sumário

Dedicatória.....	i
Agradecimentos	ii
Resumo	iv
Abstract.....	v
Lista de Figuras.....	vi
Lista de Abreviaturas	ix
Sumário.....	x
1. Introdução.....	1
2. Revisão Bibliográfica	3
3. Fundamentação Teórica.....	5
3.1 Sistemas fuzzy e modelos fuzzy Takagi-Sugeno.....	5
3.1.1 Modelos fuzzy Takagi-Sugeno	7
3.1.2 Representação dos sistemas fuzzy Takagi-Sugeno	8
3.1.3 Interpretação dos sistemas fuzzy Takagi-Sugeno	10
3.1.4 Exemplos de modelos fuzzy Takagi-Sugeno	11
3.2 Identificação de sistemas e modelagem fuzzy TS	16
3.2.1 Aprendizado neuro-adaptativo para modelagem fuzzy TS.....	17
3.2.2 Exemplo de modelo TS obtido pela ferramenta ANFIS	17
3.3 Análise de Estabilidade.....	19
3.3.1 Análise de estabilidade por Lyapunov	20
3.3.2 Exemplo de Análise de Estabilidade.....	23
3.4 Compensação Paralela Distribuída	25
3.4.1 Sistemas de Controle Fuzzy usando PDC.....	25
3.4.2 Exemplo de projeto PDC	28
3.5 Controladores Convencionais	30

3.5.1	Projeto de Controladores via Alocação de Pólos	30
3.5.2	Projeto de Controladores via Sistemas de Controle Ótimo.....	31
3.6	Projeto e Análise de Estabilidade usando LMIs	32
3.6.1	Desigualdade Linear Matricial.....	32
3.6.2	Projetos de Sistemas de Controle via LMI por Derivadas de Estado	34
3.7	Exemplo de Aplicação da Metodologia em um Sistema de Pêndulo Invertido.....	36
3.7.1	Modelagem fuzzy.....	36
3.7.2	Projeto do Controlador via Alocação de Pólos	38
3.7.3	Projeto do Controlador via Técnica de Controle Ótimo	41
3.7.4	Projeto do controlador via LMI	43
3.8	Resumo da Metodologia de Projeto.....	47
4.	Metodologia.....	49
4.9	Modelagem do Sistema Nível de um Líquido	50
4.9.1	Modelagem Física	50
4.9.2	Construção do modelo Takagi-Sugeno a partir de Ensaios de um Sistema de Nível.....	51
4.10	Projeto do Controlador Fuzzy	56
4.10.1	Erro nulo em Regime Permanente	57
4.10.2	Matriz de Ganhos via Alocação de Pólos	58
4.10.3	Matriz de Ganhos via Sistemas de Controle Ótimo.....	59
4.10.4	Malhas de Controle Correspondentes	60
4.11	Análise de Estabilidade	64
4.11.1	Estabilidade para o Sistema de Controle via Alocação de Pólos.....	64
4.11.2	Estabilidade para Sistema de Controle via Sistema de Controle Ótimo .	64
4.11.3	Questões sobre Instabilidade.....	65
4.12	Programas e Softwares	66

5.	Resultados.....	67
5.1	Controlador Fuzzy via Alocação de Pólos.....	67
5.2	Controlador Fuzzy via Sistema de Controle Ótimo.....	70
6.	Conclusão	74
7.	Referencias Bibliográficas.....	75
	ANEXO 1 – Programa em MATLAB para simulação (programa principal).....	78
	ANEXO 2 – Sub-rotinas para cálculo dos ganhos de retroação	79
	ANEXO 3 – Sub-rotina para análise de estabilidade.....	80
	ANEXO 4 – Sub-rotina do controlador fuzzy	81
	ANEXO 5 – Sub-rotina da planta fuzzy Takagi-Sugeno.....	82
	ANEXO 6 – Sub-rotinas para as funções de pertinência.....	83
	ANEXO 7 – Programa em LABVIEW implementando lei de controle fuzzy	84

1. Introdução

Este trabalho realiza uma implementação experimental de uma malha de controle por intermédio de um sistema computadorizado, onde será utilizada uma metodologia de projeto que visa à estabilização e a compensação de processos não-lineares. O trabalho considera questões relacionadas à estabilidade e projeto de malhas de controle *fuzzy* para sistemas com características não-lineares.

A maioria dos sistemas reais são não-lineares. Um sistema não-linear é caracterizado por não se aplicar o princípio da superposição, i.e., a resposta a duas entradas não pode ser calculada tratando-se uma entrada de cada vez e adicionando-se os resultados (Ogata, 2003). As abordagens tradicionais para sistemas não-lineares utilizam procedimentos de linearização dos modelos dos processos a se controlar em torno de determinados pontos nominais de operação. Modelos linearizados são muito importantes na engenharia de controle por permitirem a utilização de ferramentas lineares para análise e projeto. No entanto, estes tipos de abordagens não garantem um bom desempenho das malhas de controle resultantes em pontos de operação distintos dos nominais.

Existem modelagens apropriadas para representar sistemas com características não-lineares (Aguirre, 2000); os modelos NARX e NARMAX são exemplos. Algumas técnicas de modelagem utilizam modelos *fuzzy*, onde uma representação bastante empregada é aquela que utiliza modelos funcionais *fuzzy*, também conhecidos como modelos Takagi-Sugeno. Neste tipo de modelagem, as dinâmicas locais em diferentes regiões do espaço de estado são representadas por um conjunto de modelos lineares associados. O modelo completo do sistema é obtido pela combinação destes modelos locais, definindo um modelo global.

Obtido o “modelo difuso” de um processo não-linear, uma alternativa muito interessante de malha de controle para compensar o processo, consiste no projeto de um controlador *fuzzy* cujas regras são obtidas através do modelo correspondente. Esta abordagem pode utilizar técnicas convencionais de projeto por meio de métodos de retroação de estados, como aquelas utilizadas para malhas de controle de sistemas lineares com embasamento teórico muito bem fundamentado. Entre tais técnicas pode-se citar: técnicas de alocação de pólos; critério de estabilidade de Lyapunov; métodos de sistemas de controle ótimo; e outros (Ogata, 2003).

Técnicas de controle para sistemas lineares, no entanto, apresentam limitações quando aplicadas a processos não-lineares. O motivo é que a utilização de ganhos fixos nas leis de

controle resultantes, não contempla uma ampla faixa de operação de um sistema não-linear. No entanto, ao se utilizar controladores representados por regras *fuzzy*, pode-se viabilizar a utilização de técnicas de controle convencionais no projeto dos controladores em questão, pois as regras mapeiam adequadamente os vários pontos de operação do sistema a ser controlado. A idéia é que para cada modelo linear local do processo, um controlador *fuzzy* por realimentação de estados seja projetado, por exemplo. O controlador resultante é não-linear globalmente, sendo uma combinação *fuzzy* de cada controlador individual projetado. Esta técnica de controle *fuzzy* é conhecida como “compensação paralela distribuída” (PDC – *Parallel Distributed Compensation*) (Wang, et al., 1996). Análises de estabilidade podem ser realizadas considerando o critério de Lyapunov. As análises de estabilidade e procedimentos de projeto das malhas de controle podem ser reduzidas a problemas de desigualdades lineares matriciais (LMI – *Linear Matrix Inequality*) (Boyd, et al., 1994). Neste trabalho a solução numérica de uma LMI é realizada com o auxílio do software MATLAB. Existem pacotes específicos (*toolboxes*), conhecidos na literatura como “*solvers*” para se resolver LMIs, o *solver* padrão do MATLAB é o “*LMI control toolbox*”.

Neste trabalho foi utilizado um sistema de controle de nível real, que na prática é um sistema não-linear, para ilustrar a aplicação da metodologia citada e implementar experimentalmente uma malha de controle *fuzzy* com as características mencionadas. A realização de testes reais complementar os resultados obtidos por simulações numéricas.

A estrutura deste trabalho está organizada da forma descrita abaixo.

Capítulo 2: Uma revisão bibliográfica sobre temas relacionados com a dissertação desenvolvida, citando os principais trabalhos referenciados na literatura.

Capítulo 3: Apresenta toda a fundamentação teórica necessária para o desenvolvimento do trabalho. Conceitos, formulações e exemplos são apresentados. O capítulo se concretiza através de uma aplicação teórica e de uma compilação dos itens abordados.

Capítulo 4: Apresenta a metodologia de projeto e uma implementação experimental em um sistema computadorizado, foco principal do trabalho.

Capítulo 5: Cita os resultados obtidos no ensaio prático.

Capítulo 6: Encerra o trabalho citando as conclusões obtidas e propostas de trabalhos futuros.

2. Revisão Bibliográfica

A representação de sistemas não-lineares usando modelos *fuzzy* Takagi-Sugeno foi introduzida por Takagi e Sugeno (1985) e desde então muitas pesquisas tem sido feitas para o desenvolvimento de uma metodologia sistemática de sistemas *fuzzy* para processos não-lineares. A principal característica desta modelagem está em representar eficientemente um sistema não-linear através de um conjunto de subsistemas lineares (Taniguchi, et al., 2001).

Os conceitos mais importantes para os sistemas de controle *fuzzy* Takagi-Sugeno estão relacionados à análise de estabilidade e ao projeto de controladores (Tanaka, K.; Sugeno, M., 1990) (Tanaka, K.; Sugeno, M., 1992). Esses temas têm sido extensivamente estudados. No projeto de controladores, o conceito de compensação paralela distribuída (PDC) permite que técnicas convencionais de controle sejam utilizadas (Wang, et al., 1996). Na análise de estabilidade, o foco dos estudos é a aplicação do critério de Lyapunov.

É possível reduzir um problema de estabilidade para os sistemas de controle *fuzzy* de forma a encontrar uma função comum de Lyapunov que satisfaça o conjunto das condições de estabilidade, de modo a englobar todos os subsistemas da modelagem resultante (Tanaka, et al., 1996) (Tanaka, K.; Wang, H.O., 2001). Se as condições de estabilidade são representadas em termos de desigualdade linear de matrizes, o problema pode ser numericamente resolvido por técnicas de otimização convexa implementadas em pacotes computacionais (Gahinet, et al., 1995).

Portanto, uma ampla variedade de problemas da teoria de sistemas de controle pode ser reduzida em problemas de otimização convexa envolvendo LMI. Os resultados destes problemas de otimização podem ser obtidos numericamente com muita eficiência usando métodos de pontos interiores (Boyd, et al., 1994).

Modelos *fuzzy* Takagi-Sugeno para modelagem de sistemas não-lineares e na correspondente compensação, possuem três procedimentos principais (Taniguchi, et al., 2001) (Lian, et al., 2006). O primeiro passo consiste em representar um sistema não-linear através de vários subsistemas lineares locais por meio de regras *fuzzy* e usar a compensação paralela distribuída (Wang, et al., 1995) (Chen, et al., 1999) (Chen, et al., 2000) (Kim, E.; Lee, H., 2000) (Joh, et al., 1998) no projeto do controlador. O segundo estágio consiste na análise de estabilidade do sistema em malha fechada usando o método de Lyapunov através da desigualdade linear de matrizes (Boyd, et al., 1994) (Tanaka, K.; Wang, H.O., 2001). Finalmente, a última etapa consiste em encontrar os ganhos de um controlador

correspondente, solucionando as LMIs do problema em questão por meio de métodos numéricos.

Muitas aplicações bem sucedidas baseadas em modelos Takagi-Sugeno têm sido apresentadas na literatura, revelando um número grande de resultados importantes concernentes a análise de estabilidade e ao projeto de controladores (Fang, et al., 2006).

3. Fundamentação Teórica

Este capítulo apresenta todo o embasamento teórico necessário para o desenvolvimento do trabalho. Cada item abordado será conceituado e exemplificado formando uma base de conhecimentos suficiente para que se compreenda a metodologia em questão.

3.1 Sistemas fuzzy e modelos fuzzy Takagi-Sugeno

A principal característica de um sistema *fuzzy* é a representação de informações (conhecimento) condicionadas por regras *fuzzy* Se-Então:

$$\begin{aligned} & \text{Se } x \text{ é } A \text{ (premissa),} \\ & \text{Então } y \text{ é } B \text{ (consequente).} \end{aligned} \tag{1}$$

Onde x e y são as variáveis de entrada e saída, respectivamente, e A e B são termos linguísticos associados aos conjuntos *fuzzy* que descrevem linguisticamente essas variáveis. Para um dado valor de entrada, a saída correspondente é calculada a partir do conjunto de regras através de um método de inferência.

Em geral há três tipos de sistemas *fuzzy* citados na literatura (Wang, 1997):

- sistemas *fuzzy* puros;
- sistemas *fuzzy* Mamdani;
- sistemas *fuzzy* Takagi-Sugeno (TS).

Estes sistemas serão rapidamente descritos a seguir.

Sistemas fuzzy puros

A Figura 1 ilustra a configuração básica de um sistema *fuzzy* puro. As regras *fuzzy* representam um conjunto de regras Se-Então. A inferência *fuzzy* combina estas regras *fuzzy* em um mapeamento dos conjuntos *fuzzy* no espaço de entrada $U \subset R^n$ para conjuntos *fuzzy* no espaço de saída $V \subset R$. Se a linha tracejada na Figura 1 fizer parte do modelo, o sistema passa a ser denominado sistema *fuzzy* dinâmico. A principal característica de um sistema *fuzzy* puro é que as variáveis de entradas e saídas são conjuntos *fuzzy* (Machado, 2003).

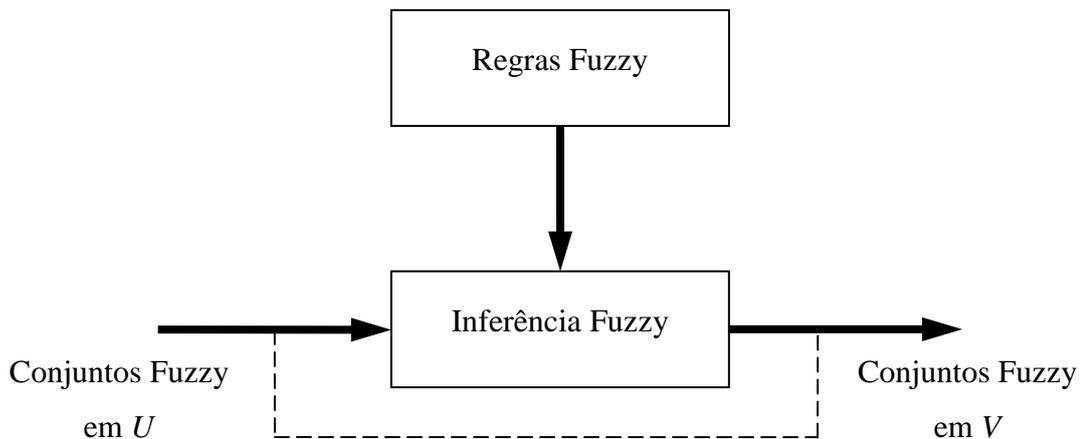


Figura 1 - Configuração básica dos sistemas fuzzy puros.

Sistemas fuzzy Mamdani

A Figura 2 ilustra a configuração básica do sistema *fuzzy* com fuzzificador e defuzzificador. Neste tipo de sistema, um fuzzificador (uma função matemática) na entrada do mesmo transforma a informação de uma variável *não-fuzzy* de valor real, em um grau de pertinência associado a um conjunto *fuzzy* correspondente. Na saída do sistema, um defuzzificador (um procedimento numérico) transforma um (ou mais) conjunto *fuzzy* devidamente processado, em uma variável real *não-fuzzy*. A principal característica do sistema *fuzzy* com fuzzificador e defuzzificador é permitir que o usuário entre com valores reais e obtenha na saída valores reais, mas com as informações internas do sistema computadas em termos associados aos conjuntos nebulosos do modelo (Machado, 2003).

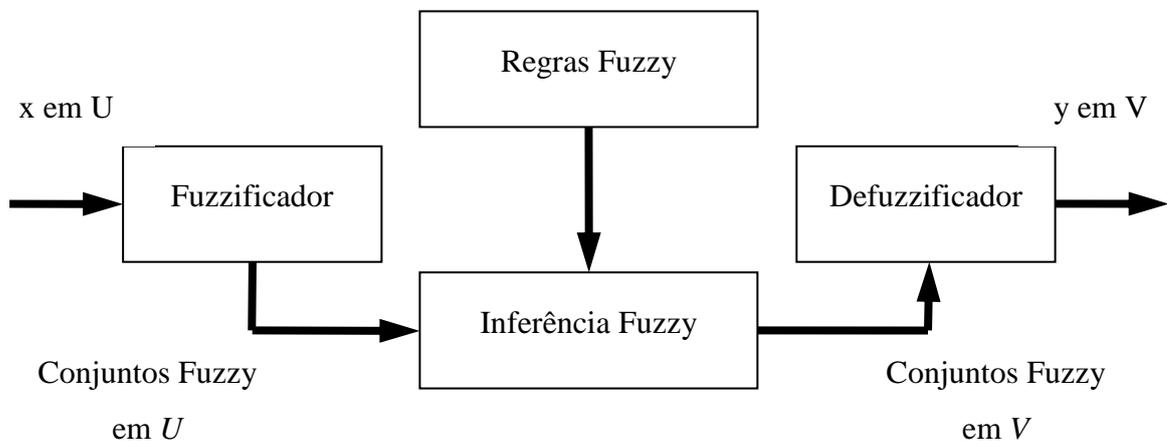


Figura 2 - Configuração básica dos sistemas fuzzy com fuzzificador e defuzzificador.

Sistemas Fuzzy Takagi-Sugeno

Um sistema *fuzzy* alternativo foi proposto por Takagi e Sugeno (1985). Nestes sistemas, as entradas e saídas são variáveis como nos sistemas com fuzzificador e defuzzificador. Entretanto, ao invés de considerar as regras *fuzzy* Se-Então na forma de (1), estes sistemas usam regras na seguinte forma:

$$\begin{array}{l} \text{Se } x \text{ é } A \text{ (premissa),} \\ \text{Então } y = f(x) \text{ (consequente).} \end{array} \quad (2)$$

Comparando (1) e (2), verifica-se que na segunda forma, a parte relativa aos consequentes das regras é expressa agora por uma função polinomial. Esta mudança possibilita representar modelos numéricos e uma forma mais adequada a aplicações numéricas. Assim, no sistema *fuzzy* Takagi-Sugeno é obtido um peso médio dos valores nos consequentes das regras.

A Figura 3 mostra a configuração básica de um sistema *fuzzy* Takagi-Sugeno.

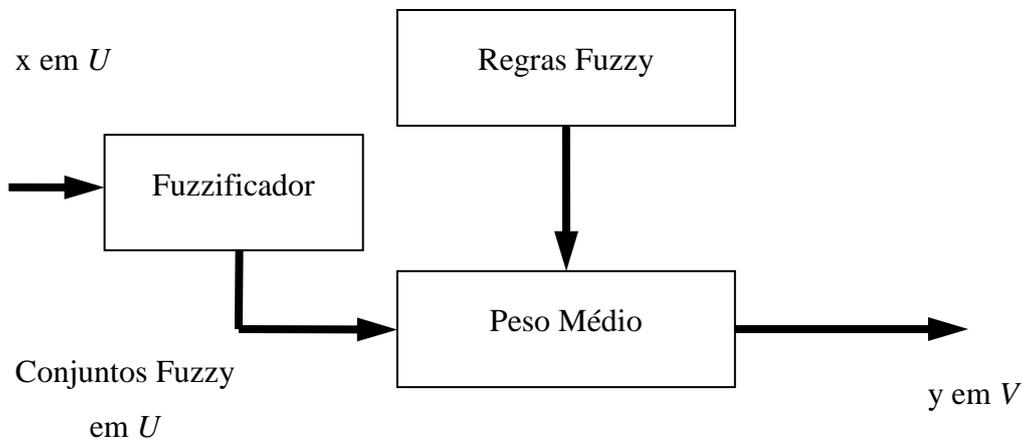


Figura 3 – Configuração básica do sistema fuzzy Takagi-Sugeno.

3.1.1 Modelos fuzzy Takagi-Sugeno

Os modelos *fuzzy* Takagi-Sugeno constituem uma abordagem alternativa para modelagem *fuzzy*. Estes modelos também possuem uma estrutura baseada em regras. Contudo, os consequentes das regras não são conjuntos *fuzzy* como nos modelos linguísticos. Os consequentes das regras *fuzzy* Takagi-Sugeno são formados por funções *crisp* (*não-fuzzy*) que mapeiam as entradas do modelo em relação a sua saída. Essas funções, também

denominadas de modelos locais, possuem usualmente uma forma afim em seus argumentos. Nesse caso o modelo Takagi-Sugeno é linear nos parâmetros das referidas funções.

Estes modelos podem ser considerados como uma versão *fuzzy* dos métodos de aproximação linear por partes, que proporcionam uma relação linear (ou afim) da entrada-saída para cada subespaço pré-determinado do espaço de entrada, possuindo uma habilidade de interpolação considerável. A estrutura dos modelos Takagi-Sugeno é intrinsecamente adequada para a representação de sistemas dinâmicos complexos (não-lineares, etc.) (Filev, D., 1991).

3.1.2 Representação dos sistemas fuzzy Takagi-Sugeno

O modelo *fuzzy* Takagi-Sugeno (TS) pode representar uma classe genérica de sistemas não-lineares (Wang, et al., 1996). Portanto, é possível representar uma dada planta não-linear através de modelo *fuzzy* Takagi-Sugeno (Takagi, T.; Sugeno, M., 1985). As dinâmicas do sistema são capturadas por um conjunto de implicações *fuzzy* que caracterizam relações locais no espaço de estados. A principal característica de um modelo *fuzzy* Takagi-Sugeno é expressar as dinâmicas locais de cada implicação (regra) *fuzzy* por um modelo linear. O modelo *fuzzy* completo do sistema é alcançado pela combinação *fuzzy* dos modelos locais.

Especificamente, um sistema modelo Takagi-Sugeno é descrito por regras *fuzzy* SE-ENTÃO, que localmente representam relações lineares de entrada-saída de um determinado sistema. O modelo *fuzzy* pode ser representado na forma:

$$\text{Regra } i: \quad \begin{array}{l} \text{SE} \quad x_1(k) \text{ é } M_{i1} \cdots \text{ e } x_n(k) \text{ é } M_{in} \\ \text{ENTÃO } \mathbf{x}(k+1) = \mathbf{A}_i \mathbf{x}(k) + \mathbf{B}_i \mathbf{u}(k). \end{array} \quad (3)$$

Onde

$$\begin{array}{l} \mathbf{x}^T(k) = [x_1(k), x_2(k), \dots, x_n(k)]; \\ \mathbf{u}^T(k) = [u_1(k), u_2(k), \dots, u_m(k)]. \end{array} \quad (4)$$

Sendo:

- Regra i ($i = 1, 2, \dots, r$). Denota o número de regras Se-Então. São relações *fuzzy* que descrevem implicações que são calculadas através de uma função de mapeamento;
- $x_j(k)$ ($j = 1, 2, \dots, n$) são variáveis de estado e entrada da regra;

- $u_j(k) (j = 1, 2, \dots, m)$ são variáveis de entrada do modelo local;
- $\mathbf{x}(k+1) = \mathbf{A}_i \mathbf{x}(k) + \mathbf{B}_i \mathbf{u}(k)$ é a saída da implicação (regra) i , resultado das funções que relacionam as entradas do modelo com a saída;
- M_{ij} são as variáveis *fuzzy* onde cada função de pertinência associada pode ser do tipo trapezoidal, triangular, gaussiana, ou outras funções que representem um subespaço *fuzzy* no qual a implicação *Regra i* pode ser aplicada.

Dado um par de $(\mathbf{x}(k), \mathbf{u}(k))$, a saída final do modelo *fuzzy* é calculada como a média ponderada das saídas individuais de cada implicação, ou seja:

$$\mathbf{x}(k+1) = \frac{\sum_{i=1}^r w_i(k) \{ \mathbf{A}_i \mathbf{x}(k) + \mathbf{B}_i \mathbf{u}(k) \}}{\sum_{i=1}^r w_i(k)}. \quad (5)$$

Onde o nível de ativação da i -ésima implicação é dada por

$$w_i(k) = \prod_{j=1}^n M_{ij} (x_j(k)). \quad (6)$$

Sendo $M_{ij} (x_j(k))$ o grau de pertinência de $x_j(k)$ em M_{ij} .

A saída final para a representação do sistema em malha aberta é expresso por (7):

$$\mathbf{x}(k+1) = \frac{\sum_{i=1}^r w_i(k) \{ \mathbf{A}_i \mathbf{x}(k) \}}{\sum_{i=1}^r w_i(k)} \quad (7)$$

Onde é assumido que

$$\sum_{i=1}^r w_i(k) > 0,$$

(8)

$$w_i(k) \geq 0 \quad i = 1, 2, \dots, r$$

para qualquer amostra k . Cada componente linear $A_i x(k)$ é chamado de subsistema local.

Sugeno e seus cooperadores (Takagi, T.; Sugeno, M., 1985) (Sugeno, M.; Kang, G. T., 1986) (Sugeno, M.; Kang, G. T., 1988) (Sugeno, M.; Tanaka, K., 1991) propuseram a utilização de funções afins nas partes consequentes das implicações. Esta escolha permite uma interpretação matemática simples do modelo com uma interpolação de diferentes modelos afins, e implica que a saída é linear em seus parâmetros. Estes parâmetros podem ser estimados utilizando qualquer algoritmo de estimação convencional (Ogata, 2003).

Assim, o modelo *fuzzy* Takagi-Sugeno é capaz de aproximar um sistema não-linear com uma combinação de vários sistemas lineares afins pela decomposição de todo o espaço de entrada em vários espaços parciais e representar cada espaço entrada/saída com uma equação linear.

Geralmente o modelo *fuzzy* de um sistema físico pode ser construído por:

- conversão direta da equação não-linear do modelo físico;
- identificação usando um observador de dados de entrada/saída.

Um objetivo comum em modelagem de sistemas é obter uma parametrização mínima dos sistemas dinâmicos representados. Considerações práticas geralmente necessitam manter o número das variáveis das premissas tão pequeno quanto possível para reduzir os efeitos do dimensionamento das regras de modelagem resultantes.

3.1.3 Interpretação dos sistemas *fuzzy* Takagi-Sugeno

A principal característica do modelo *fuzzy* Takagi-Sugeno em relação aos modelos *fuzzy* linguísticos é que as partes dos consequentes constituem um modelo dinâmico afim ou linear (homogêneo), no lugar de um conjunto *fuzzy*. Esta propriedade apresenta várias vantagens computacionais (Johansen, et al., 1998):

- Da perspectiva de engenharia de controle, o uso de modelos locais lineares preenche a lacuna entre sistemas de controle *fuzzy* e a teoria de sistemas de controle lineares. Existem

muitas ferramentas na teoria de sistemas lineares que podem ser parcialmente aplicadas para modelos Takagi-Sugeno *fuzzy* e os controladores baseados em regras resultantes;

- Os consequentes das regras *fuzzy* Takagi-Sugeno (modelos locais) possibilitam um número reduzido de regras em muitas aplicações. Nas regiões onde as não-linearidades de um modelo são mais acentuadas, os consequentes podem ter funções mais elaboradas para representarem estas não-linearidades e, conforme a necessidade, mais regras podem ser incluídas nestas regiões. Por outro lado, nas regiões onde as não-linearidades não são tão fortes, é possível utilizar funções de pertinência mais simples e o número de regras pode ser reduzido. Esta distribuição das regras leva a uma melhor representação do sistema com um número mínimo de modelos locais;

- A estrutura do modelo (partição do espaço de estados e estrutura da modelo local) e propriedades do modelo local, permite em determinadas aplicações, ser facilmente relacionada com o modelo físico. Isto simplifica o desenvolvimento do modelo e a sua validação.

Destaques deste tipo de modelagem:

- Eficiente computacional;
- Permite trabalhar com técnicas lineares de projeto;
- Permite trabalhar com métodos adaptativos e de otimização;
- Garante continuidade no espaço de saída do modelo representado;
- Permite análises matemáticas conhecidas para garantia de estabilidade.

3.1.4 Exemplos de modelos *fuzzy* Takagi-Sugeno

Exemplo 1

Como ilustração, seja um modelo de sistema de apenas uma entrada e uma única saída (Figura 4) (MathWorks, 2011):

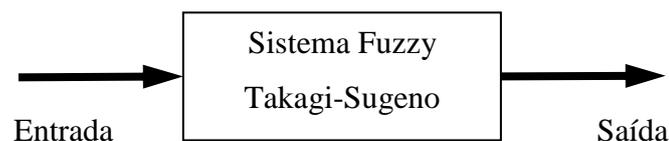


Figura 4 - Sistema fuzzy TS do Exemplo 1.

A variável Entrada possui duas funções de pertinência (Negativa e Positiva) do tipo gaussianas (Figura 5). Negativa se refere a valores de entrada menores que zero, e Positiva refere a valores maiores que zero.

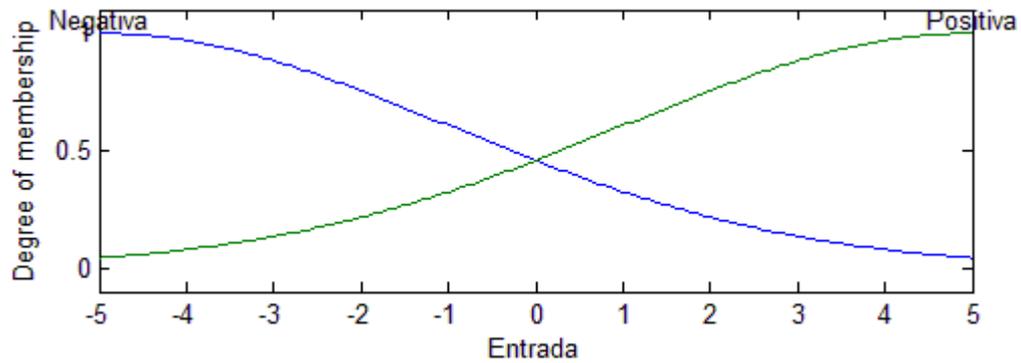


Figura 5 - Funções de pertinência da variável de entrada do Exemplo 1.

A variável Saída possui duas funções de pertinência que são funções lineares em relação à variável de entrada. De acordo com os valores de entrada e segundo as regras *fuzzy* indicadas abaixo, cada função de saída (implicação) será ativada com uma determinada ponderação.

Regra 1: SE Entrada é Negativa ENTÃO Saída₁ = -1 × Entrada - 1

Regra 2: SE Entrada é Positiva ENTÃO Saída₂ = 1 × Entrada - 1

A Figura 6 mostra como a saída deste modelo *fuzzy* muda continuamente entre faixas de valores do domínio [-5, 5] da variável de entrada.

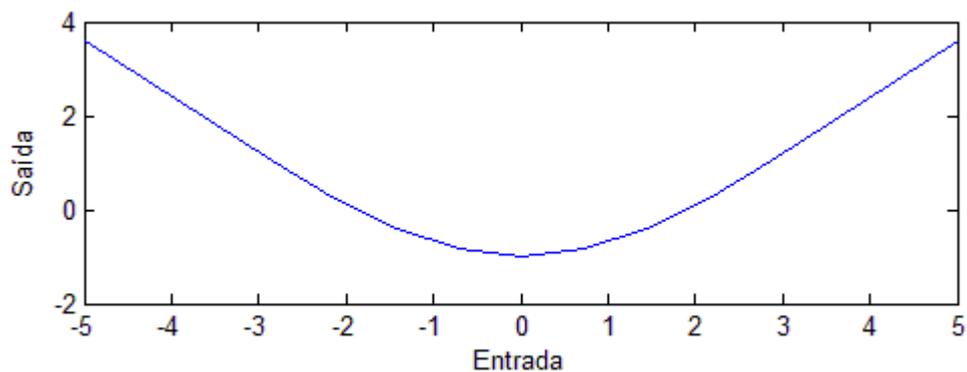


Figura 6 - Mudança suave de Saída1 para Saída2

Como foi visto, modelos *fuzzy* Takagi-Sugeno permitem incorporar a modelagem de informações dentro de um sistema *fuzzy*. Por extensão, conclui-se que um sistema *fuzzy* pode ser construído a partir de vários sistemas lineares representando um sistema altamente não-linear movendo no seu espaço de operação.

Exemplo 2

Considere como exemplo de um modelo *fuzzy* para se decidir a quantidade de gorjeta a ser dada em um determinado restaurante (MathWorks, 2011). A gorjeta será de acordo com o serviço prestado e com a qualidade da comida. Portanto o sistema terá duas entradas (Serviço e Comida) e uma saída (Gorjeta) conforme ilustrado na Figura 7.

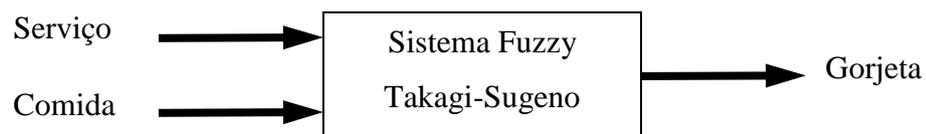


Figura 7 - Systema fuzzy TS do Exemplo 2.

A qualidade do serviço será representada através de três funções de pertinência do tipo gaussiana, podendo o serviço se enquadrar nos conjuntos *fuzzy* Pobre, Médio e Bom. Enquanto a qualidade da comida poderá se enquadrar nos conjuntos *fuzzy* Detestável e Saborosa, que são do tipo trapezoidal. As funções de pertinência em questão estão representadas na Figura 8.

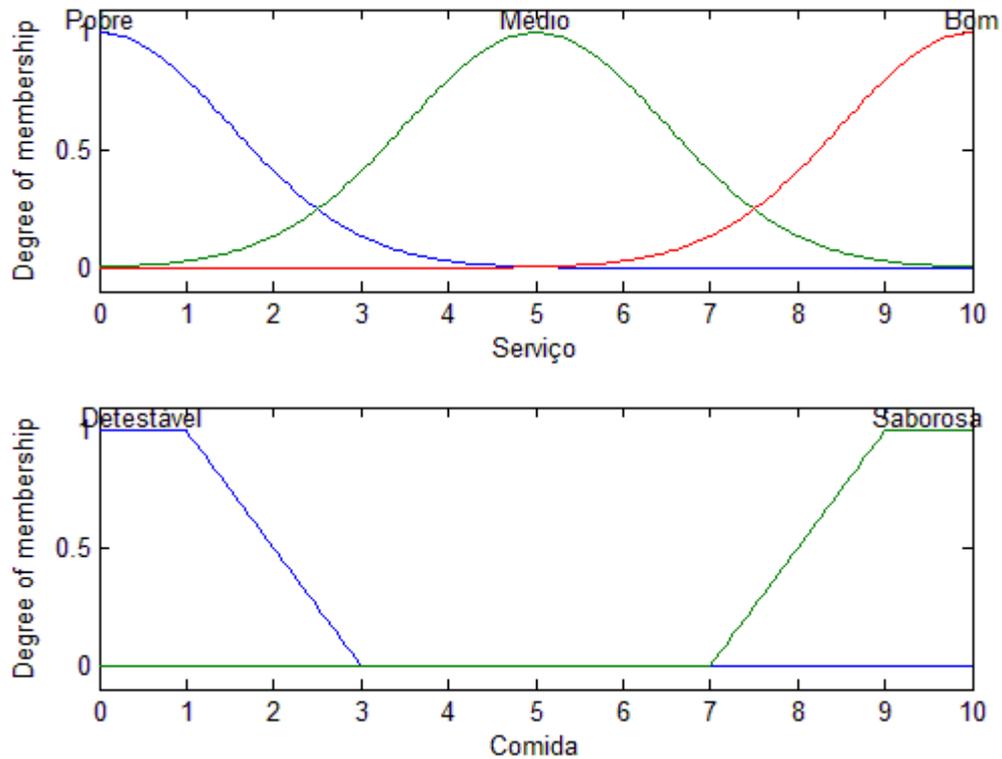


Figura 8 - Funções de pertinência do Exemplo 2.

Para a variável de saída Gorjeta, denominaremos em: Econômica, Média e Generosa. Cada informação da variável de saída é uma função linear constante, sendo:

$$\begin{aligned} \text{Econômica} &= 5, \\ \text{Média} &= 15, \\ \text{Generosa} &= 25. \end{aligned}$$

As regras são dadas por:

- Regra 1: SE Serviço é Pobre ou Comida é Detestável ENTÃO Gorjeta é Econômica;*
Regra 2: SE Serviço é Médio ENTÃO Gorjeta é Média;
Regra 3: SE Serviço é Bom ou Comida é Saborosa ENTÃO Gorjeta é Generosa.

A Figura 9 mostra que para uma avaliação de Serviço = 3 e Comida = 8, a Gorjeta será de 18.6%.

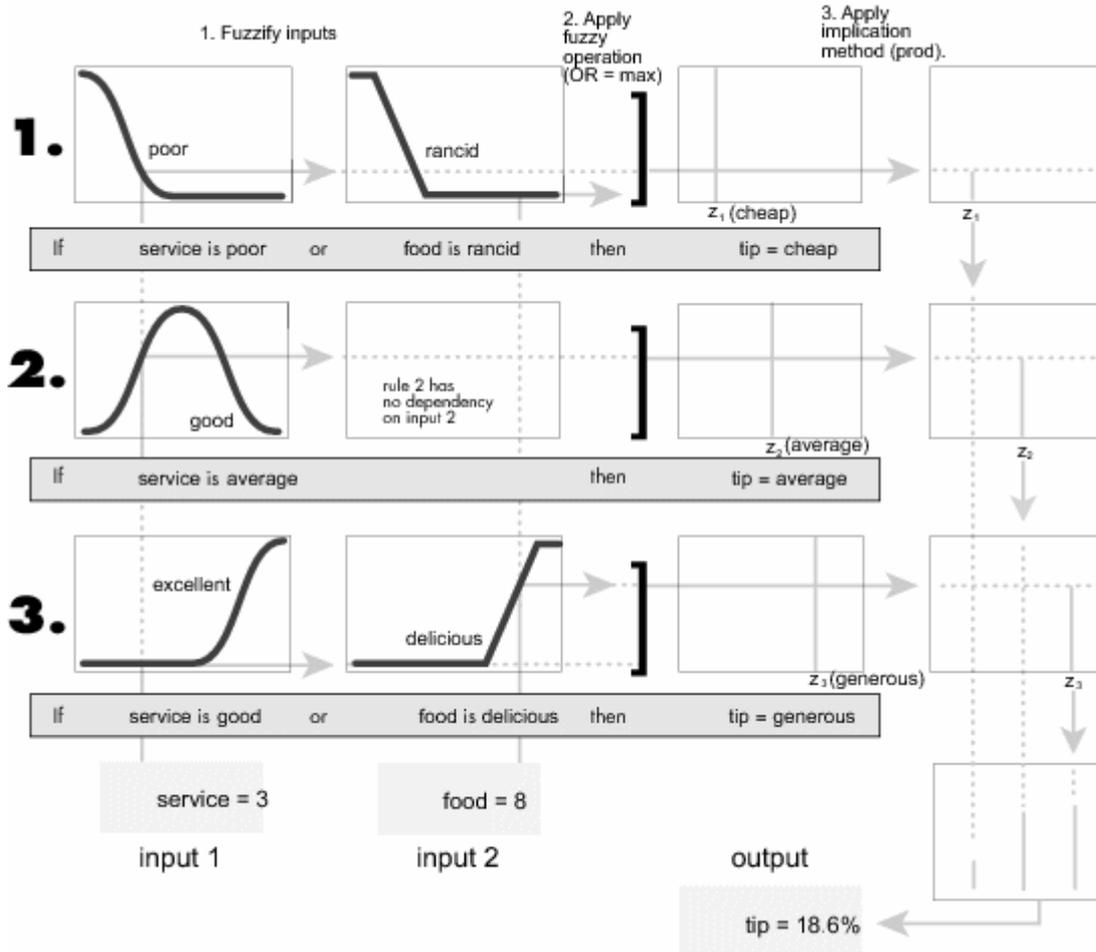


Figura 9 - Operação do sistema fuzzy TS do Exemplo 2 (MathWorks, 2011).

Para a Regra 1 da Figura 9 tem-se os seguintes resultados:

- grau de pertinência da entrada Serviço no conjunto Pobre é 0,126;
- grau de pertinência da entrada Comida no conjunto Detestável é 0;
- aplicando operação OR (valor máximo) resulta na ponderação 0,126.

Para a Regra 2 da Figura 9 tem-se os seguintes resultados:

- grau de pertinência da entrada Serviço no conjunto Médio é 0,398;
- resultando na ponderação 0,398.

Para a Regra 3 da Figura 9 tem-se os seguintes resultados:

- grau de pertinência da entrada Serviço no conjunto Bom é 0;
- grau de pertinência da entrada Comida no conjunto Saborosa é 0,491;
- aplicando operação OR (valor máximo) resulta na ponderação 0,491.

Portanto aplicando (7), tem-se:

$$Gorjeta = \frac{0,126 \times Econ\acute{o}mica + 0,398 \times M\acute{e}dia + 0,491 \times Generosa}{0,126 + 0,398 + 0,491} = 18,6\%.$$

A Figura 10 mostra a superf3cie de sa3ida do sistema *fuzzy*, deixando claro o comportamento da sa3ida gorjeta de acordo com as entradas servi3o e comida. Se a qualidade do servi3o e da comida aumentar, tamb3m a gorjeta aumentar3. Se a qualidade do servi3o e da comida diminuir, tamb3m a gorjeta diminuir3.

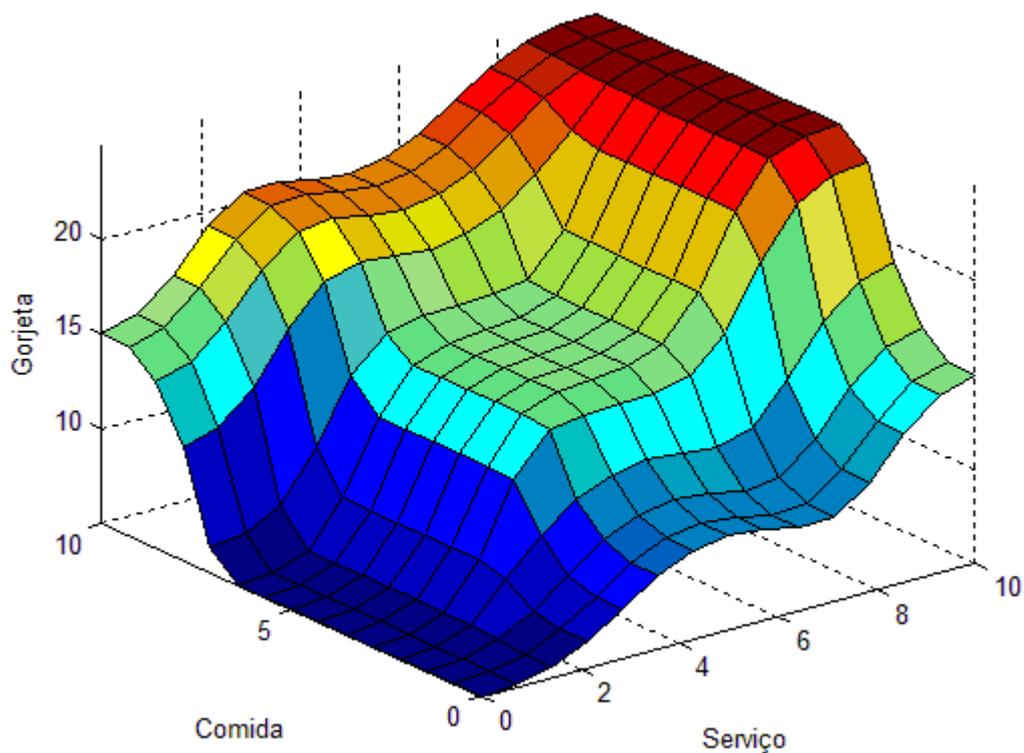


Figura 10 - Superf3cie de sa3ida para Exemplo 2.

3.2 Identifica3o de sistemas e modelagem fuzzy TS

A identifica3o de sistemas pode ser definida como sendo a utiliza3o de procedimentos num3ricos, cuja finalidade 3 obter modelos de sistemas din3micos a partir de medidas das suas entradas e a sa3idas.

Modelos matem3ticos de sistemas din3micos podem ser obtidos utilizando conceitos f3sicos da natureza dos mesmos. Esta modelagem 3 chamada fenomenol3gica ou “caixa branca”, em que geralmente, 3 necess3rio conhecer certas caracter3sticas ou propriedades f3sicas do sistema a ser modelado.

Para sistemas complexos ou do tipo “caixa preta”, nos quais não se tem acesso fácil das partes constituintes dos mesmos, pode ser oneroso obter a modelagem fenomenológica. Nestes casos, a utilização de métodos de identificação de sistemas facilita a obtenção de modelos de sistemas reais. Não é necessário conhecimento prévio do sistema, bastando ter dados de medidas das suas entradas e saídas. A partir do processamento adequado dos dados medidos, obtêm-se os coeficientes ou parâmetros das equações que modelam o sistema em questão. Esta modelagem é conhecida como modelagem empírica ou “caixa preta”. Mesmo para sistemas dinâmicos facilmente modelados fisicamente, o emprego de técnicas de identificação pode ser interessante em termos práticos, uma vez que modelos matemáticos poder ser obtidos de maneira relativamente rápida e precisa (Souza, A. C. Z.; Pinheiro, C. A. M.;, 2008).

3.2.1 *Aprendizado neuro-adaptativo para modelagem fuzzy TS*

Uma das maneiras de construir um modelo *fuzzy* Takagi-Sugeno é através de procedimentos de identificação de sistemas. O software Matlab possui uma rotina de treinamento chamada ANFIS para modelagem de tais sistemas. Este recurso usa um algoritmo híbrido de aprendizado para identificar os parâmetros do sistema *fuzzy* Takagi-Sugeno, que combina o método de mínimos quadrados e o método de aprendizado por *backpropagation*.

Essas técnicas de aprendizado neuro-adaptativa fornecem um método para o processo de modelagem *fuzzy* para mapear informações sobre um conjunto de dados. O nome ANFIS deriva de “sistema de inferência neuro-*fuzzy* adaptativa”. Usando um conjunto de dados de entrada/saída, a função ANFIS constrói um sistema de inferência *fuzzy* dos quais os parâmetros das funções de pertinência são ajustados usando os métodos de *backpropagation* e mínimos quadrados. Este ajuste permite que o sistema *fuzzy* aprenda com os dados de modelagem (MathWorks, 2011).

3.2.2 *Exemplo de modelo TS obtido pela ferramenta ANFIS*

Exemplo 3

Suponha modelar a função abaixo com dados de entrada no intervalo [1, 10]:

$$y = \frac{\text{sen}(2x)}{e^{\frac{x}{5}}}$$

Usando a função ANFIS com as opções de 5 funções de pertinência (*memberships* na forma de sino) de entrada e 20 períodos de treinamento da rede neural de ajuste dos parâmetros, têm-se na Figura 11 os valores resultantes das funções associadas aos cinco conjuntos nebulosos (*in1mf2* a *in1mf5*) correspondentes do modelo em questão.

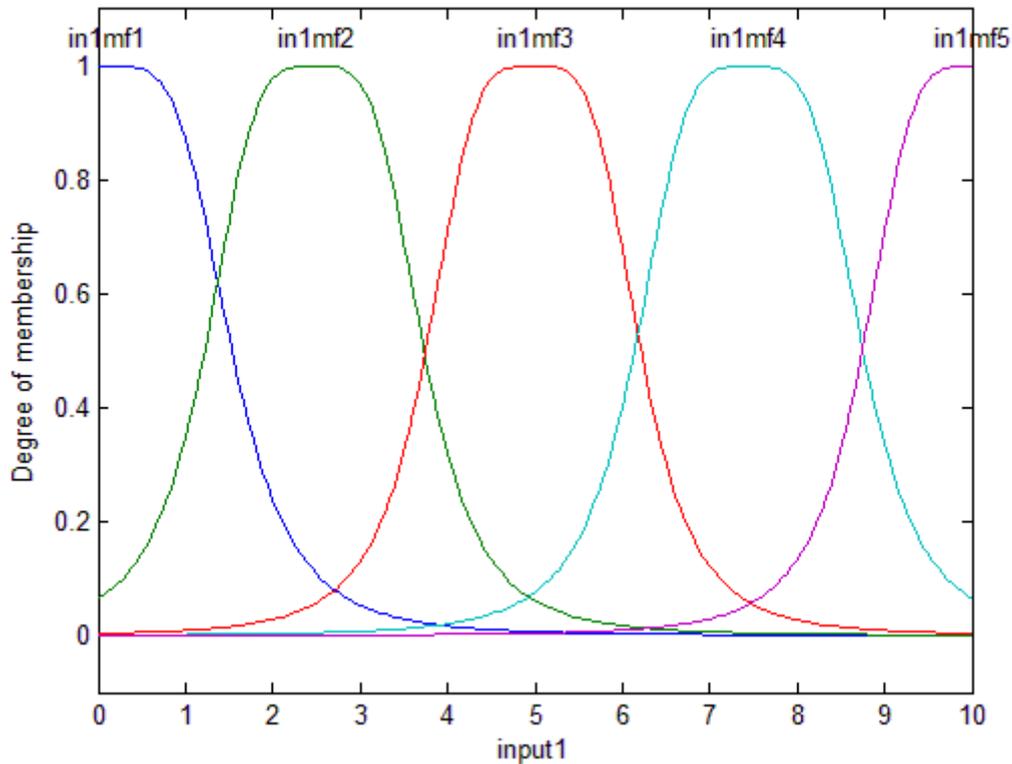


Figura 11 – Funções pertinência de entrada do Exemplo 3.

As regras geradas pelo software são:

- Regra 1: SE input1 é in1mf1 ENTÃO output é output1;*
- Regra 2: SE input1 é in1mf2 ENTÃO output é output2;*
- Regra 3: SE input1 é in1mf3 ENTÃO output é output3;*
- Regra 4: SE input1 é in1mf4 ENTÃO output é output4.*

A variável *input1* do modelo *fuzzy* corresponde à variável *x* da função a ser modelada, e a variável *output* equivale à variável *y*. Os coeficientes resultantes das funções lineares do modelo TS obtido são:

```

output1 = [1,979  0,3108];
output2 = [1,115  -4,011];
output3 = [-0,61  2,636];
output4 = [-0,4502  3,594];
output5 = [0,1557  -1,357].

```

A Figura 12 mostra a simulação do modelo *fuzzy* obtido (gráfico na cor verde) e os dados de treinamento (em cor azul) utilizados no treinamento. Dos gráficos da figura nota-se que a modelagem *fuzzy* da função não-linear apresentou uma aproximação razoável.

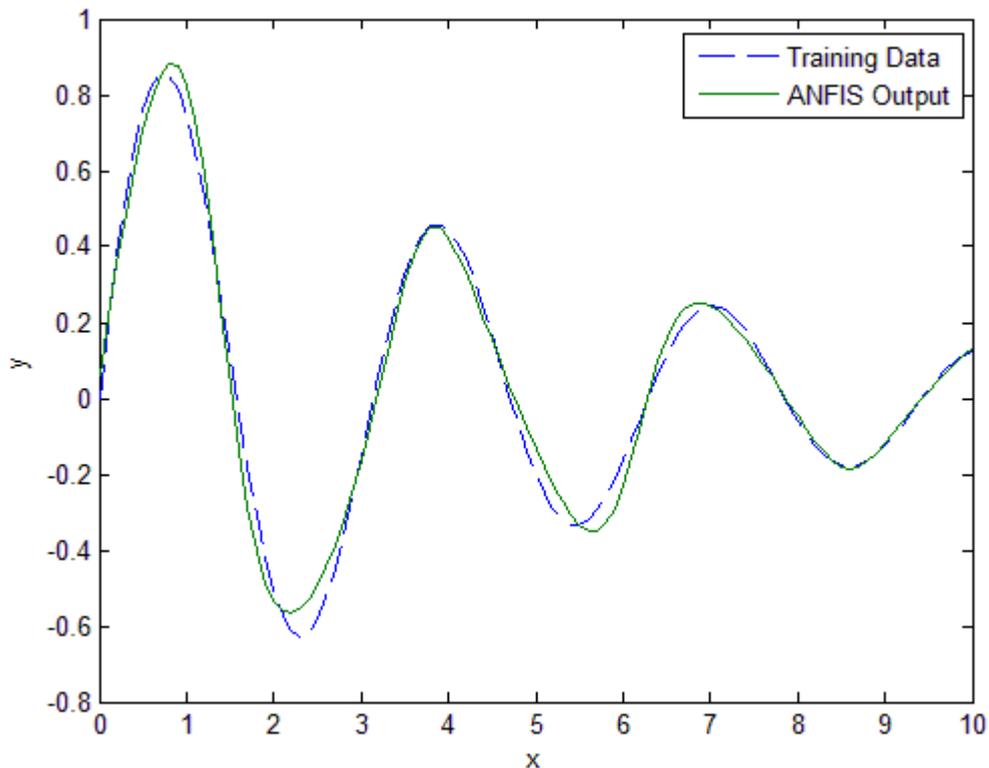


Figura 12 – Exemplo de função ANFIS

3.3 Análise de Estabilidade

Muitos pesquisadores dedicaram-se à busca de condições de análise de estabilidade para sistemas *fuzzy* TS em aplicações relacionadas a sistemas de controle. Durante os anos 90 vários trabalhos foram publicados destacando-se as pesquisas com a utilização de LMIs em sistemas *fuzzy* (Tanaka, K.; Sugeno, M., 1990), onde foi abordado o tratamento de condições

de estabilidade básicas para sistemas de controle *fuzzy* utilizando o método direto de Lyapunov.

3.3.1 Análise de estabilidade por Lyapunov

Um dos mais importantes conceitos concernente às propriedades dos sistemas de controle diz respeito à estabilidade. Teoremas foram derivados (Tanaka, K.; Sugeno, M., 1992) para a estabilidade de um sistema *fuzzy* em conformidade com a definição de estabilidade na concepção de Lyapunov. Uma condição suficiente que garanta a estabilidade de um sistema *fuzzy* é dada em termos do método direto de Lyapunov.

Considerando o seguinte sistema *fuzzy* livre:

$$\text{Regra } i: \begin{array}{ll} \text{SE} & x_1(k) \text{ é } M_{i1} \cdots \text{ e } x_n(k) \text{ é } M_{in} \\ \text{ENTÃO} & \mathbf{x}(k+1) = \mathbf{A}_i \mathbf{x}(k). \end{array} \quad (9)$$

Os subsistemas lineares correspondentes a i -ésima implicação da regra podem ser representados na forma matricial $\mathbf{A}_i \mathbf{x}(k)$, onde $\mathbf{x}(k) \in \mathbb{R}^n$, $\mathbf{A}_i \in \mathbb{R}^n \times \mathbb{R}^n$. A saída do sistema *fuzzy* é representada por (7).

O Teorema 1 e o Lema 1 descritos a seguir são necessários para o Teorema 2 que é um importante teorema a respeito da estabilidade de um sistema *fuzzy*. O Teorema 1 é conhecido como teorema de estabilidade de Lyapunov.

Teorema 1 (Kuo, B. C., 1980). Considere um sistema discreto descrito por:

$\mathbf{x}(k+1) = f(\mathbf{x}(k))$, onde $\mathbf{x}(k) \in \mathbb{R}^n$, $f(\mathbf{x}(k))$ é um vetor de funções $n \times 1$ com a seguinte propriedade: $f(\mathbf{0}) = \mathbf{0}$ para qualquer k .

Suponha que exista uma função escalar $V(\mathbf{x}(k))$ contínua em $\mathbf{x}(k)$ tal que

- (a) $V(\mathbf{0}) = 0$;
- (b) $V(\mathbf{x}(k)) > 0$ p/ $\mathbf{x}(k) \neq \mathbf{0}$;
- (c) $V(\mathbf{x}(k))$ tende ao infinito quando $\|\mathbf{x}(k)\| \rightarrow \infty$;
- (d) $\Delta V(\mathbf{x}(k)) < 0$ p/ $\mathbf{x}(k) \neq \mathbf{0}$.

Então o estado de equilíbrio $\mathbf{x}(k) = \mathbf{0}$ para todo k é assintoticamente estável e $V(\mathbf{x}(k))$ é uma função de Lyapunov.

Lema 1. Se \mathbf{P} é uma matriz definida positiva tal que

$$\mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{P} < 0 \quad \text{e} \quad \mathbf{B}^T \mathbf{P} \mathbf{B} - \mathbf{P} < 0,$$

onde $\mathbf{A}, \mathbf{B}, \mathbf{P} \in R^{n \times n}$, então

$$\mathbf{A}^T \mathbf{P} \mathbf{B} + \mathbf{B}^T \mathbf{P} \mathbf{A} - 2\mathbf{P} < 0.$$

Prova:

$$\begin{aligned} \mathbf{A}^T \mathbf{P} \mathbf{B} + \mathbf{B}^T \mathbf{P} \mathbf{A} - 2\mathbf{P} &= -(\mathbf{A} - \mathbf{B})^T \mathbf{P} (\mathbf{A} - \mathbf{B}) + \mathbf{A}^T \mathbf{P} \mathbf{A} + \mathbf{B}^T \mathbf{P} \mathbf{B} - 2\mathbf{P} \\ &= -(\mathbf{A} - \mathbf{B})^T \mathbf{P} (\mathbf{A} - \mathbf{B}) + \mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{P} + \mathbf{B}^T \mathbf{P} \mathbf{B} - \mathbf{P} \end{aligned}$$

Uma vez que \mathbf{P} é uma matriz positiva definida,

$$-(\mathbf{A} - \mathbf{B})^T \mathbf{P} (\mathbf{A} - \mathbf{B}) \leq \mathbf{0}.$$

Portanto, a conclusão do lema confere.

Teorema 2 (Tanaka, K.; Sugeno, M., 1992). O equilíbrio de um sistema *fuzzy* (7) é assintoticamente estável globalmente se existir uma matriz definida positiva comum \mathbf{P} para todos os subsistemas do modelo tal que

$$\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} < 0 \quad p/ i \in \{1, 2, \dots, r\}. \quad (10)$$

Prova. Considere a função escalar $V(\mathbf{x}(k))$ tal que

$V(\mathbf{x}(k)) = \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k)$, onde \mathbf{P} é uma matriz positiva definida. Esta função satisfaz as seguintes propriedades:

- (a) $V(\mathbf{0}) = 0$;
- (b) $V(\mathbf{x}(k)) > 0$ p/ $\mathbf{x}(k) \neq \mathbf{0}$;
- (c) $V(\mathbf{x}(k))$ tende ao infinito quando $\|\mathbf{x}(k)\| \rightarrow \infty$.

Seja a condição indicada a seguir:

$$\begin{aligned} \Delta V(\mathbf{x}(k)) &= V(\mathbf{x}(k+1)) - V(\mathbf{x}(k)) \\ &= \mathbf{x}^T(k+1) \mathbf{P} \mathbf{x}(k+1) - \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k) \\ &= \left(\sum_{i=1}^r w_i \mathbf{A}_i \mathbf{x}(k) / \sum_{i=1}^r w_i \right)^T \mathbf{P} \left(\sum_{i=1}^r w_i \mathbf{A}_i \mathbf{x}(k) / \sum_{i=1}^r w_i \right) - \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k) \\ &= \mathbf{x}^T(k) \left\{ \left(\sum_{i=1}^r w_i \mathbf{A}_i^T / \sum_{i=1}^r w_i \right) \mathbf{P} \left(\sum_{i=1}^r w_i \mathbf{A}_i / \sum_{i=1}^r w_i \right) - \mathbf{P} \right\} \mathbf{x}(k) \\ &= \sum_{i,j=1}^r w_i w_j \mathbf{x}^T(k) \{ \mathbf{A}_i^T \mathbf{P} \mathbf{A}_j - \mathbf{P} \} \mathbf{x}(k) / \sum_{i,j=1}^r w_i w_j \end{aligned}$$

$$= \left[\sum_{i=1}^r (w_i)^2 \mathbf{x}^T(k) \{ \mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} \} \mathbf{x}(k) + \sum_{i < j}^r w_i w_j \mathbf{x}^T(k) \{ \mathbf{A}_i^T \mathbf{P} \mathbf{A}_j + \mathbf{A}_j^T \mathbf{P} \mathbf{A}_i - 2\mathbf{P} \} \mathbf{x}(k) \right] / \sum_{i,j=1}^r w_i w_j,$$

onde $w_i \geq 0$ para $i \in \{1, 2, \dots, r\}$ e $\sum_{i=1}^r w_i > 0$.

Do Lema 1 e (14), obtém-se:

(d) $\Delta V(\mathbf{x}(k)) < 0$.

Pelo Teorema 1, $V(\mathbf{x}(k))$ é uma função de Lyapunov e o sistema (7) é assintoticamente estável. Este teorema é reduzido para o teorema de estabilidade de Lyapunov para sistemas lineares discretos quando $r = 1$. A condição de estabilidade do Teorema 2 é produzida, portanto, usando uma função quadrática $V(\mathbf{x}) = \mathbf{x}^T \mathbf{P} \mathbf{x}$. Se existir um $\mathbf{P} > 0$ tal que V demonstre a estabilidade do sistema (7), o mesmo também é dito ser quadraticamente estável e V é chamada função quadrática de Lyapunov.

Este teorema pode ser aplicado para análise de estabilidade de um sistema não-linear aproximado por uma função linear por partes (*piecewise*) quando a condição (10) for satisfeita em $w_i \geq 0$ e $\sum_{i=1}^r w_i > 0$. Uma função linear por partes pode ser descrita como um caso especial de (3) se for utilizado conjuntos *crisp* em vez de conjuntos *fuzzy* nas partes das premissas do sistema *fuzzy*. É possível dividir um sistema não-linear em alguns subsistemas linearizados no espaço de estado do sistema. Isso significa que o mesmo é aproximado por uma função *piecewise*. Como muitos sistemas não-lineares podem ser aproximados por funções lineares por partes, este teorema pode ser largamente aplicado não somente em sistema *fuzzy*, mas também em sistemas não-lineares em geral.

O Teorema 2 apresenta uma condição suficiente para assegurar a estabilidade de um modelo representado por (7). Pode parecer intuitivo achar que um sistema não-linear aproximado seja estável, caso todos os subsistemas locais lineares sejam estáveis. No entanto, não é este o caso geral.

Todas as matrizes \mathbf{A}_i 's são estáveis se existir uma matriz comum definida positiva \mathbf{P} . Nem sempre existirá uma matriz comum definida positiva \mathbf{P} mesmo que todas as matrizes \mathbf{A}_i 's sejam estáveis. Vale ressaltar que um sistema *fuzzy* pode ser assintoticamente estável globalmente, mesmo que não se encontre uma matriz comum definida positiva \mathbf{P} . Assim, deve-se considerar que um sistema *fuzzy* nem sempre é assintoticamente estável globalmente, mesmo que todas as matrizes \mathbf{A}_i 's sejam estáveis.

A obtenção da matriz \mathbf{P} pode ser resolvida eficientemente via técnicas convexas de otimização por LMI (Boyd, et al., 1994). Isso significa expressar a condição de estabilidade do Teorema 2 em LMI. Para checar a estabilidade é preciso encontrar uma matriz comum \mathbf{P} ou determinar que tal matriz não exista.

3.3.2 Exemplo de Análise de Estabilidade

Exemplo 4

Considere o seguinte sistema *fuzzy*:

$$\begin{aligned} \text{Regra 1: SE } x_2(k) \text{ é } M_1 \text{ ENTÃO } \mathbf{x}(k+1) &= \mathbf{A}_1 \mathbf{x}(k) \\ \text{Regra 2: SE } x_2(k) \text{ é } M_2 \text{ ENTÃO } \mathbf{x}(k+1) &= \mathbf{A}_2 \mathbf{x}(k) \end{aligned}$$

onde $\mathbf{x}(k) = [x_1(k) \quad x_2(k)]^T$ e

$$\mathbf{A}_1 = \begin{bmatrix} 1 & -0,5 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} -1 & -0,5 \\ 1 & 0 \end{bmatrix}.$$

A Figura 13 mostra as funções de pertinência de M_1 e M_2 .

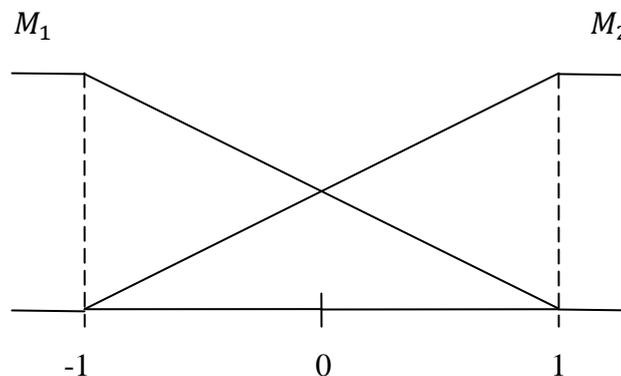


Figura 13 – Funções de pertinência do Exemplo 4.

Sejam as condições iniciais do modelo: $x_1(0) = 0,9$ e $x_2(0) = -0,7$. A Figura 14 ilustra o comportamento dos seguintes sistemas lineares associados ao modelo:

$$\mathbf{x}(k+1) = \mathbf{A}_1 \mathbf{x}(k);$$

$$\mathbf{x}(k+1) = \mathbf{A}_2 \mathbf{x}(k).$$

Os sistemas lineares associados são estáveis, pois A_1 e A_2 são matrizes estáveis, e isto pode está ilustrado na Figura 14(a) e na Figura 14(b), respectivamente. No entanto, o sistema global *fuzzy*, que consiste na combinação *fuzzy* destes sistemas lineares, é instável como indicado na Figura 14(c) para as variáveis de estado x_1 e x_2 . Obviamente, neste exemplo, não existe uma matriz comum P que garante a estabilidade global do sistema *fuzzy*.

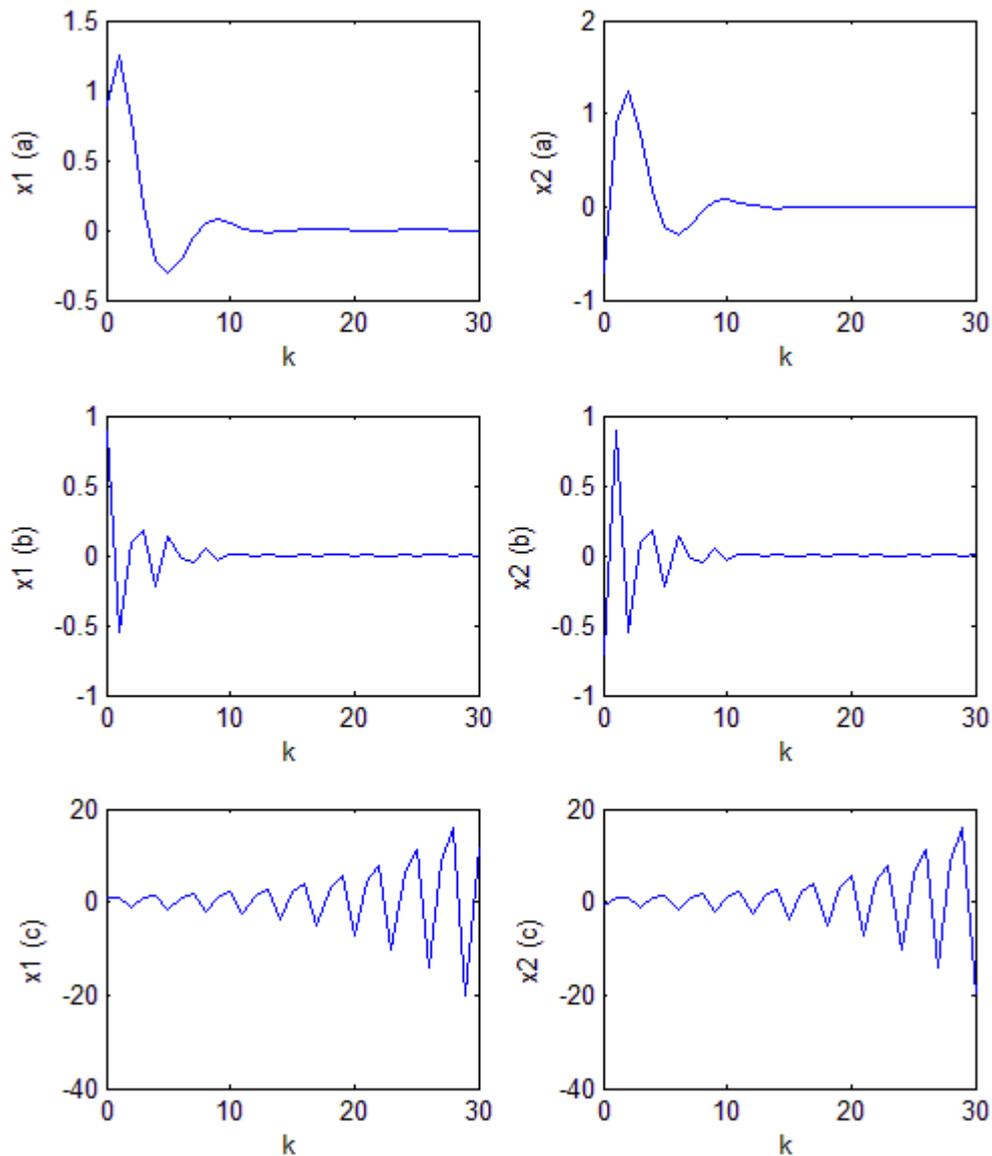


Figura 14 - (a) Comportamento de $x(k+1) = A_1x(k)$. (b) Comportamento $x(k+1) = A_2x(k)$. (c) Comportamento do Sistema Fuzzy global associado.

3.4 Compensação Paralela Distribuída

O conceito de compensação paralela distribuída (PDC) é utilizado para projetar controladores *fuzzy* a fim de estabilizar sistemas *fuzzy* (Wang, et al., 1996). A idéia é projetar um compensador associado a cada regra do modelo *fuzzy* em questão. A Figura 15 ilustra o conceito do PDC. Para cada regra pode-se usar técnicas de projeto de controle de retroação de estados. O resultado completo do controlador *fuzzy*, que em geral é não-linear, é uma combinação *fuzzy* de cada controlador individual.

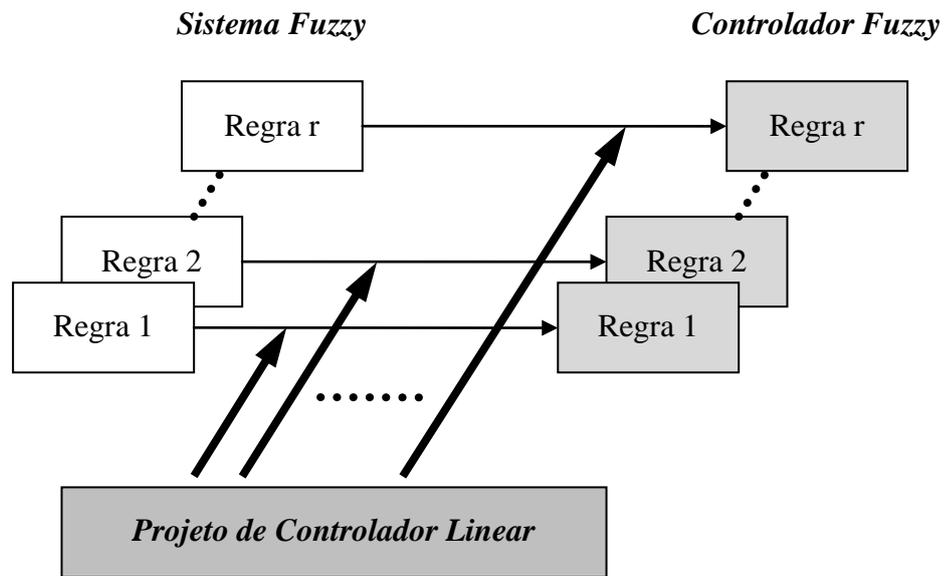


Figura 15 – Técnica de projeto PDC – Compensação Paralela Distribuída.

3.4.1 Sistemas de Controle Fuzzy usando PDC

O controlador *fuzzy* com o conceito PDC compartilha os mesmos conjuntos *fuzzy* do modelo utilizado, e possui a estrutura indicada por (11).

$$\text{Regra } i: \quad \begin{array}{l} \text{SE } x_1(k) \text{ é } M_{i1} \cdots \text{ e } x_n(k) \text{ é } M_{in} \\ \text{ENTÃO } \mathbf{u}(k) = -\mathbf{F}_i \mathbf{x}(k) \end{array} \quad (11)$$

A informação de comando do controlador é dada por (12).

$$\mathbf{u}(k) = \frac{-\sum_{i=1}^r w_i(k)\{\mathbf{F}_i \mathbf{x}(k)\}}{\sum_{i=1}^r w_i(k)} \quad (12)$$

Substituindo (12) em (5) obtém-se (13).

$$\mathbf{x}(k+1) = \frac{\sum_{i=1}^r \sum_{j=1}^r w_i(k)w_j(k)\{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j\} \mathbf{x}(k)}{\sum_{i=1}^r \sum_{j=1}^r w_i(k)w_j(k)} \quad (13)$$

Quando for aplicado o Teorema 2, tem-se a seguinte condição suficiente para a estabilidade quadrática do sistema de controle dado pelo Teorema 3 abaixo.

Teorema 3 (Wang, et al., 1996): O equilíbrio de um sistema *fuzzy* (13) é assintoticamente estável se existir uma matriz comum positiva definida \mathbf{P} tal que:

$$\begin{aligned} & \{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j\}^T \mathbf{P} \{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j\} - \mathbf{P} < \mathbf{0} \\ & p/ w_i(k) \cdot w_j(k) \neq 0, \quad \forall k, \quad i, j = 1, 2, \dots, r. \end{aligned} \quad (14)$$

Note que o sistema (13) pode ser escrito como:

$$\begin{aligned} \mathbf{x}(k+1) = & \frac{1}{W} \left[\sum_{i=1}^r w_i(k)w_j(k)\{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_i\} \mathbf{x}(k) \right. \\ & \left. + 2 \sum_{i < j} w_i(k)w_j(k) \mathbf{G}_{ij} \mathbf{x}(k) \right]. \end{aligned} \quad (15)$$

Onde

$$\mathbf{G}_{ij} = \frac{\{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j\} + \{\mathbf{A}_j - \mathbf{B}_j \mathbf{F}_i\}}{2} \quad i < j$$

e

$$W = \sum_{i=1}^r \sum_{j=1}^r w_i(k) w_j(k).$$

Assim, tem-se a seguinte condição suficiente:

Teorema 4 (Wang, et al., 1996): O equilíbrio de um sistema de controle *fuzzy* (13) é assintoticamente estável se existir uma matriz comum \mathbf{P} definida positiva tal que as duas seguintes condições sejam satisfeitas:

$$\begin{aligned} \{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_i\}^T \mathbf{P} \{\mathbf{A}_i - \mathbf{B}_i \mathbf{F}_i\} - \mathbf{P} < 0 \\ i = 1, 2, \dots, r \end{aligned} \quad (16)$$

$$\begin{aligned} \mathbf{G}_{ij} \mathbf{P} \mathbf{G}_{ij} - \mathbf{P} < 0 \\ i < j < r \end{aligned} \quad (17)$$

Observação: As condições do Teorema 3 são mais relaxadas do que o Teorema 2.

O objetivo do projeto de um sistema de controle *fuzzy* com a abordagem em questão é selecionar \mathbf{F}_i ($i = 1, 2, \dots, r$) tal que as condições de (16) e (17) do Teorema 4 sejam satisfeitas. Usando a notação de estabilidade quadrática, pode-se também pensar no problema do projeto de controle como em achar \mathbf{F}_i 's tal que o sistema (13) em malha fechada seja estável quadraticamente. Se existe tais \mathbf{F}_i 's, o sistema (5) é também dito ser estabilizável quadraticamente via projeto PDC.

Em geral, primeiramente projeta-se um controlador para cada regra e verifica-se se as condições de estabilidade são satisfeitas. Como mencionado, pode-se utilizar técnicas LMI para resolver o problema de análise de estabilidade.

Este procedimento de projeto representa uma forma sistemática de modelagem e controle *fuzzy* para uma vasta classe de sistemas não-lineares. A Figura 16 ilustra a metodologia de projeto em questão.

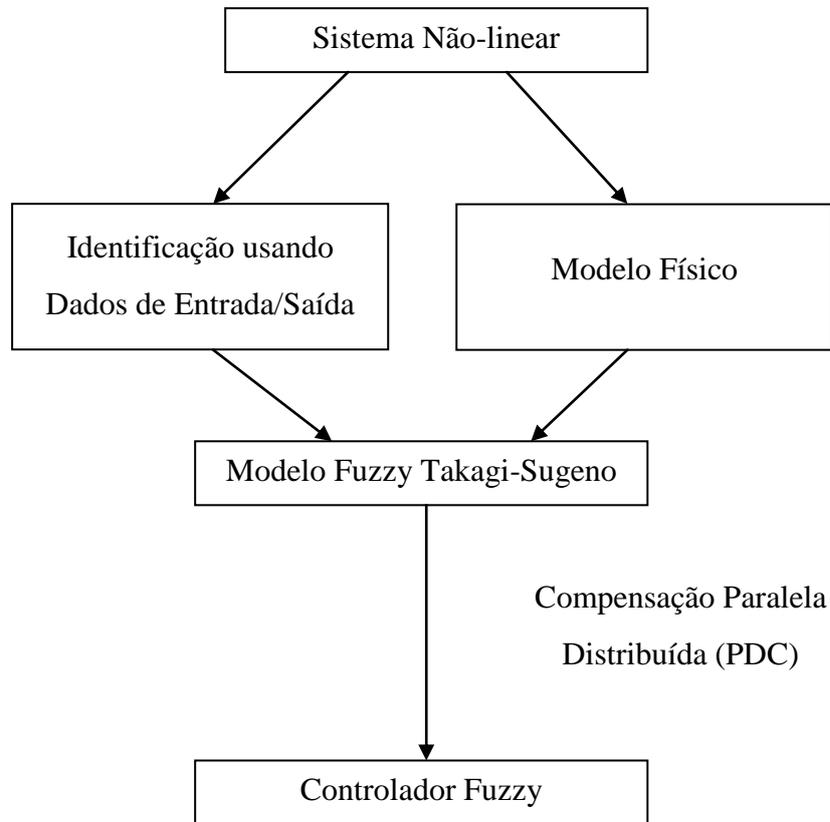


Figura 16 – Estrutura de projeto para controle fuzzy.

3.4.2 Exemplo de projeto PDC

Exemplo 5

Considere o sistema *fuzzy* abaixo onde as matrizes \mathbf{A}_1 e \mathbf{A}_2 são as mesmas do Exemplo 4 e cujas funções de pertinência encontram-se na Figura 13:

Regra 1: SE $x_2(k)$ é M_1 ENTÃO $\mathbf{x}(k+1) = \mathbf{A}_1\mathbf{x}(k) + \mathbf{B}_1\mathbf{u}(k)$

Regra 2: SE $x_2(k)$ é M_2 ENTÃO $\mathbf{x}(k+1) = \mathbf{A}_2\mathbf{x}(k) + \mathbf{B}_2\mathbf{u}(k)$.

Onde:

$$B_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, B_2 = \begin{bmatrix} -2 \\ 1 \end{bmatrix}.$$

Escolhendo os autovalores $[0,5 \quad 0,35]$ para malha fechada, tem-se

$$\mathbf{F}_1 = [0,65 \quad -0,5],$$

$$\mathbf{F}_2 = [0,87 \quad -0,11],$$

e

$$\mathbf{A}_1 - \mathbf{B}_1\mathbf{F}_1 = \begin{bmatrix} 0,35 & 0 \\ 0,35 & 0,5 \end{bmatrix}, \mathbf{A}_2 - \mathbf{B}_2\mathbf{F}_2 = \begin{bmatrix} 0,74 & -0,72 \\ 0,13 & 0,11 \end{bmatrix},$$

$$\mathbf{G}_{12} = \begin{bmatrix} 0,210 & -0,945 \\ 0,24 & 0,305 \end{bmatrix}.$$

O controlador PDC correspondente será:

Regra 1: SE $x_2(k)$ é M_1 ENTÃO $\mathbf{u}(k) = -\mathbf{F}_1\mathbf{x}(k)$;

Regra 2: SE $x_2(k)$ é M_2 ENTÃO $\mathbf{u}(k) = -\mathbf{F}_2\mathbf{x}(k)$.

Encontrando a matriz definida positiva \mathbf{P} abaixo, as condições de estabilidade de (16) e (17) são satisfeitas. Em outras palavras, o sistema de controle *fuzzy* em malha fechada é assintoticamente estável. A matriz \mathbf{P} é obtida utilizando um algoritmo de otimização LMI.

$$\mathbf{P} = \begin{bmatrix} 1,181 & -0,0614 \\ -0,0614 & 2,3044 \end{bmatrix}.$$

A Figura 17 mostra o comportamento do sistema de controle *fuzzy* para as mesmas condições iniciais do exemplo relacionado com a Figura 14. Observe que o sistema tende a se acomodar em um determinado valor devido à estabilidade obtida com a utilização do controlador *fuzzy* considerado.

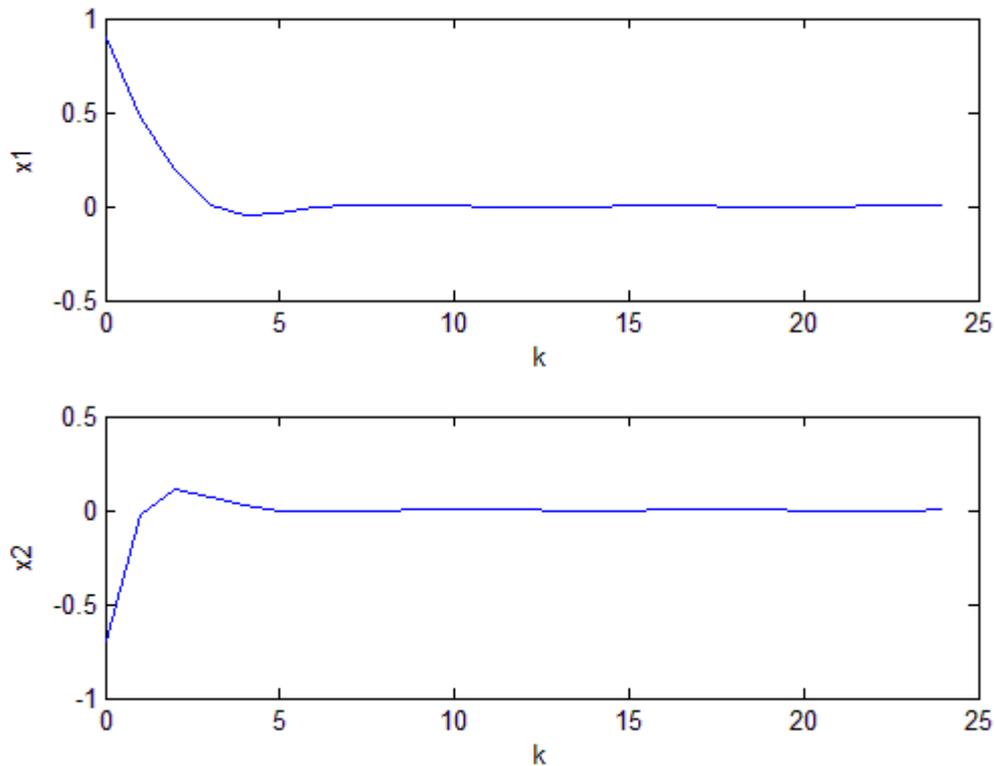


Figura 17 - Resposta do Exemplo 5 realizado o controle PDC.

3.5 Controladores Convencionais

A principal vantagem do controle PDC é possibilitar a utilização de técnicas de projeto de controladores convencionais. Serão abordadas duas técnicas por retroação de estados.

3.5.1 Projeto de Controladores via Alocação de Pólos

Se o sistema considerado for completamente controlável, então os pólos do sistema em malha fechada podem ser localizados em quaisquer posições desejadas por meio de retroação de estado através de uma matriz de ganho de retroação adequada.

Considere um sistema modelado em variáveis de estado discretas (18).

$$\mathbf{x}(n + 1) = \mathbf{A}\mathbf{x}(n) + \mathbf{B}u(n); \quad (18)$$

Onde \mathbf{x} = vetor de estado (n -dimensional); u = sinal de controle (escalar); \mathbf{A} = matriz constante $n \times n$; \mathbf{B} = matriz constante $n \times 1$.

A lei de controle é definida por (19). Isso significa que o sinal de controle é determinado pelo estado instantâneo das variáveis de estado.

$$u = -Kx \quad (19)$$

A partir da especificação desejada dos autovalores em malha fechada é possível calcular os ganhos K de retroação de estados dados por (20) para modelos discretos, e de forma similar para modelos contínuos.

$$|zI - A + BK| = \lambda \quad (20)$$

As matrizes A e B são conhecidas do modelo do sistema, I é uma matriz identidade, λ constitui os autovalores especificados ou alocados, e K são os ganhos de retroação de estados. Então, equaciona-se o sistema de modo a calcular K . Os pólos ou autovalores alocados podem ser representados na forma (21).

$$\lambda = (z + p_1)(z + p_2) \dots (z + p_n) \quad (21)$$

3.5.2 Projeto de Controladores via Sistemas de Controle Ótimo

Um sistema de controle ótimo possibilita que o mesmo tenha um determinado desempenho definido por uma função de custo. As especificações são dadas por parâmetros como a matriz de peso Q relacionada com a dinâmica das variáveis de estado x , e o peso R relativo à entrada u do sistema, que geralmente está associada à energia gasta pelo processo.

O balanceamento entre energia e dinâmica é representado matematicamente pela função de custo (22).

$$J = \int_{-\infty}^{\infty} [(X^T * Q * X) + (u^T * R * u)] dt \quad (22)$$

O mínimo da função de custo é obtido por (23).

$$\frac{dJ}{dx} = 0 \quad (23)$$

Com a equação de Riccati (24) é possível encontrar a matriz P levando em consideração o conceito de estabilidade de Lyapunov, e obter a matriz de ganhos ótimos (25) garantindo a estabilidade do sistema de controle resultante (Ogata, 2003).

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (24)$$

$$K = R^{-1}B^T P \quad (25)$$

3.6 Projeto e Análise de Estabilidade usando LMIs

Recentemente, problemas de otimização de LMIs tem recebido muita atenção (Boyd, et al., 1994). Estes problemas de otimização podem ser resolvidos em tempo polinomial. O desenvolvimento de métodos de pontos interiores para estes problemas tem se evidenciado na resolução eficiente de problemas relacionados (Nesterov, Y.; Nemirovsky, A., 1994). Para sistemas e controle, a otimização LMI tem uma importante vantagem de possibilitar a remodelagem de uma vasta variedade destes problemas como problemas LMI (Boyd, et al., 1994). Exceto para um pequeno número de casos especiais, a maioria dos problemas práticos não tem solução analítica. Entretanto, o ponto principal é que através da estrutura LMI estes problemas podem ser em geral eficientemente resolvidos de forma numérica. Assim, remodelar um problema de controle como um problema LMI é equivalente a encontrar uma solução para um problema de otimização.

3.6.1 Desigualdade Linear Matricial

Definição (Boyd, et al., 1994): Uma desigualdade linear matricial (LMI – *Linear Matrix Inequality*) é uma desigualdade da seguinte forma:

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0. \quad (26)$$

Onde $\mathbf{x}^T = (x_1, x_2, \dots, x_m)$ é um vetor de variáveis e as matrizes simétricas $F_i = F_i^T \in R^{n \times n}$, $i = 0, \dots, m$ são dadas. O símbolo de desigualdade > 0 significa que $F(x)$ é definida positiva.

A LMI representada por (26) é uma restrição convexa em \mathbf{x} , i.e., o conjunto $\{\mathbf{x} | \mathbf{F}(\mathbf{x}) > \mathbf{0}\}$ é convexo. Uma LMI pode representar uma vasta variedade de restrições convexas em \mathbf{x} . Em particular, desigualdades lineares, desigualdades quadráticas convexas, desigualdades na norma de matrizes, e restrições que se originam na teoria de controle, tais como estabilidade por Lyapunov e desigualdades de matrizes quadráticas convexas, podem ser representadas na forma de LMIs. Múltiplas LMIs $\mathbf{F}^i > \mathbf{0}, i = 1, \dots, p$ podem ser expressadas como uma única LMI $\text{diagonal}(\mathbf{F}^{(1)}, \dots, \mathbf{F}^{(p)}) > \mathbf{0}$.

É muito comum em LMIs as variáveis serem matrizes, por exemplo, na desigualdade de Lyapunov

$$\mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{P} < \mathbf{0},$$

onde $\mathbf{A} \in R^{n \times n}$ é dado e $\mathbf{P} = \mathbf{P}^T$ é variável. Neste caso a LMI não será escrita explicitamente na forma $\mathbf{F}(\mathbf{x}) > \mathbf{0}$. Além de economizar notação, isso pode conduzir a uma computação mais eficiente. Entretanto, a desigualdade de Lyapunov pode ser representada facilmente na forma (26) tomando $\mathbf{F}_0 = \mathbf{0}, \mathbf{F}_i = -\mathbf{A}^T \mathbf{P}_i \mathbf{A} + \mathbf{P}_i$, onde $\mathbf{P}_1, \dots, \mathbf{P}_m$ formam uma base para matrizes simétricas $n \times n$.

Problemas LMIs (Boyd, et al., 1994): Dada uma LMI $\mathbf{F}(\mathbf{x}) > \mathbf{0}$, o problema LMI é encontrar $\mathbf{x}^{factive}$ tal que $\mathbf{F}(\mathbf{x}^{factive}) > \mathbf{0}$. Este é um problema de factibilidade convexa.

Como exemplo, a simultânea condição de estabilidade de Lyapunov no Teorema 2 é exatamente um problema LMI: Dado $\mathbf{A}_i \in R^{n \times n}, i = 1, \dots, r$, é preciso encontrar \mathbf{P} que satisfaça a LMI

$$\mathbf{P} > \mathbf{0}, \quad \mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} < \mathbf{0}, \quad i = 1, 2, \dots, r$$

ou determinar que não existe \mathbf{P} . As condições de estabilidade encontradas neste trabalho são expressadas na forma de LMIs. Esta remodelagem é significativa no sentido de que algoritmos de otimização convexa podem ser usados para análise de estabilidade e problemas de projeto de controle. Esta remodelagem, portanto, constitui soluções para a análise de estabilidade e problemas de projeto de controle no arcabouço de modelos *fuzzy* Takagi-Sugeno, e de síntese de controladores via PDC.

O procedimento de projeto envolve um processo iterativo. Para cada regra, um controlador é designado baseado na consideração de um desempenho local estabelecido. Análises baseadas em LMIs são realizadas para verificar se as condições de estabilidade são satisfeitas. Caso não sejam satisfeitas, o controlador para cada regra será re-projetado. O procedimento iterativo de projeto é efetivo, entretanto, do ponto de vista de projeto de controle, é mais desejável projetar diretamente um controle que garanta a estabilidade de um sistema em malha fechada.

3.6.2 Projetos de Sistemas de Controle via LMI por Derivadas de Estado

É possível utilizar LMI's para resolver diretamente o problema do projeto de um sistema de controle *fuzzy*, satisfazendo as condições de estabilidade de Lyapunov (Faria, F. A., 2009). Seja um sistema controlável descrito por (27).

$$E\dot{x}(t) = Ax(t) + Bu(t) \quad (27)$$

Sendo $E \in \mathbf{R}^{n \times n}$ inversível, pode-se passar (27) para o formato padrão:

$$\dot{x}(t) = A_F x(t) + B_F u(t) = AE^{-1}x(t) + BE^{-1}u(t).$$

O objetivo é encontrar as condições necessárias e suficientes para que (27) seja assintoticamente estável usando realimentação da derivada dos estados. Então, o problema pode ser formulado como encontrar uma matriz constante K tal que:

- a matriz $(E + BK)$ tenha inversa;
- o sistema em malha fechada, obtido com a realimentação da derivada dos estados $u(t) = -K\dot{x}(t)$ seja assintoticamente estável.

Logo, o sistema em malha fechada pode ser representado por:

$$\begin{aligned} E\dot{x}(t) &= Ax(t) - BK\dot{x}(t) \leftrightarrow (E + BK)\dot{x}(t) = Ax(t) \\ \dot{x}(t) &= (E + BK)^{-1}Ax(t) = A_E x(t). \end{aligned}$$

Observações:

- A matriz $(E + BK)$ possui inversa quando $\text{rank}[E, B] = n$;

- Uma condição necessária para a aplicação da realimentação da derivada de estados é que o modelo do sistema representado por (27) não possua pólos na origem.

Teorema 5 (Faria, F. A., 2009): Supondo que o sistema dado por (27) tenha matriz de estados com inversa, ele será estabilizável se e somente se, existir uma matriz simétrica $Q \in \mathbf{R}^{n \times n}$ e uma matriz $Y \in \mathbf{R}^{m \times n}$, tais que:

$$Q > 0, \\ AQE^T + EQA^T + BYA^T + AY^T B^T > 0.$$

Neste caso, um ganho para o controlador pode ser obtido por:

$$K > YQ^{-1}.$$

No entanto, nem sempre isto garante uma solução suficiente por não levar em consideração restrições de desempenho do sistema de controle considerado. Uma restrição na taxa de decaimento da malha de controle resultante pode ser usada para diminuir o tempo de estabelecimento do sistema, sendo definida como o maior número real positivo γ tal que:

$$\lim_{t \rightarrow \infty} e^{\gamma t} \|x(t)\| = 0$$

para toda solução $x(t)$ do sistema em $t > 0$.

Teorema 6 (Faria, F. A., 2009): Supondo que o sistema definido por (27) tenha matriz de estados inversível e dado $\gamma > 0$, deve existir uma matriz simétrica $Q \in \mathbf{R}^{n \times n}$ e uma matriz $Y \in \mathbf{R}^{m \times n}$, tais que:

$$Q > 0, \\ \begin{bmatrix} AQE^T + EQA^T + BYA^T + AY^T B^T & EQ + BY \\ QE^T + Y^T B^T & \frac{-Q}{2\gamma} \end{bmatrix} < 0,$$

onde o sistema é estabilizável com taxa de decaimento, e um ganho para o controlador pode ser obtido por:

$$K > YQ^{-1}.$$

3.7 Exemplo de Aplicação da Metodologia em um Sistema de Pêndulo Invertido

Para ilustração da metodologia de projeto apresentada, é aplicada a compensação paralela distribuída (PDC) em um problema de balanço e equilíbrio de um pêndulo invertido (Figura 18). As equações de movimento do pêndulo em representação de variáveis de estado contínuas são expressas por (28).

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{g \cdot \sin(x_1) - amlx_2^2 \sin(2x_1)/2 - a \cdot \cos(x_1) u}{4l/3 - aml \cdot \cos^2(x_1)} \end{aligned} \quad (28)$$

Onde x_1 denota o ângulo (em radianos) do pêndulo com a vertical, e x_2 é a velocidade angular. $g = 9,8 \text{ m/s}^2$ é a constante de gravidade, m é a massa do pêndulo, M é a massa do carro, l é o comprimento do pêndulo, $a = 1/(m + M)$, e u é a força aplicada no carro (em Newtons). Os parâmetros do modelo são $m = 2 \text{ [Kg]}$, $M = 8 \text{ [Kg]}$, $l = 0,5 \text{ [m]}$ (Wang, et al., 1996).

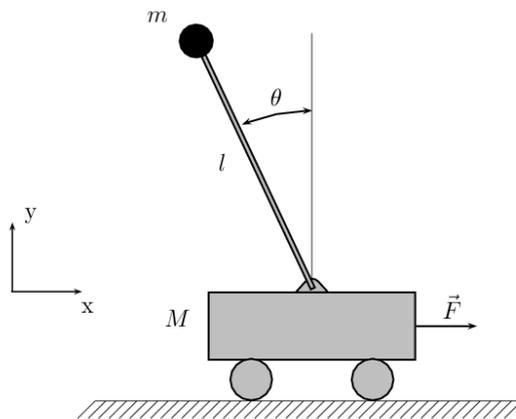


Figura 18 – Exemplo de um Sistema de Pêndulo Invertido.

3.7.1 Modelagem fuzzy

O objetivo final é obter um sistema de controle para equilibrar o pêndulo invertido quando o ângulo estiver dentro dos limites $x_1 \in (-\pi/2, \pi/2)$. Para usar a abordagem PDC,

primeiramente deve-se encontrar o modelo *fuzzy* Takagi-Sugeno que represente a dinâmica deste sistema. Um modelo baseado em regras *fuzzy* pode ser considerado abaixo

$$\begin{aligned} \text{Regra 1: } & \text{SE } x_1 \text{ ZE} \\ & \text{ENTÃO } \dot{x} = A_1 x + B_1 u \\ \text{Regra 2: } & \text{SE } x_1 \text{ é N (ou P)} \\ & \text{ENTÃO } \dot{x} = A_2 x + B_2 u \end{aligned}$$

onde

$$\begin{aligned} A_1 &= \begin{bmatrix} 0 & 1 \\ \frac{g}{4l/3 - aml} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 17,2941 & 0 \end{bmatrix}; \\ B_1 &= \begin{bmatrix} 0 \\ \frac{-a}{4l/3 - aml} \end{bmatrix} = \begin{bmatrix} 0 \\ -0,1765 \end{bmatrix}; \\ A_2 &= \begin{bmatrix} 0 & 1 \\ \frac{2g}{\pi(4l/3 - aml\beta^2)} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 9,36 & 0 \end{bmatrix}; \\ B_2 &= \begin{bmatrix} 0 \\ \frac{-a\beta}{4l/3 - aml\beta^2} \end{bmatrix} = \begin{bmatrix} 0 \\ -0,0052 \end{bmatrix}. \end{aligned}$$

O parâmetro auxiliar de modelagem possui o valor $\beta = \cos(88^\circ)$. As funções de pertinência para as regras de modelagem estão ilustradas na Figura 19.

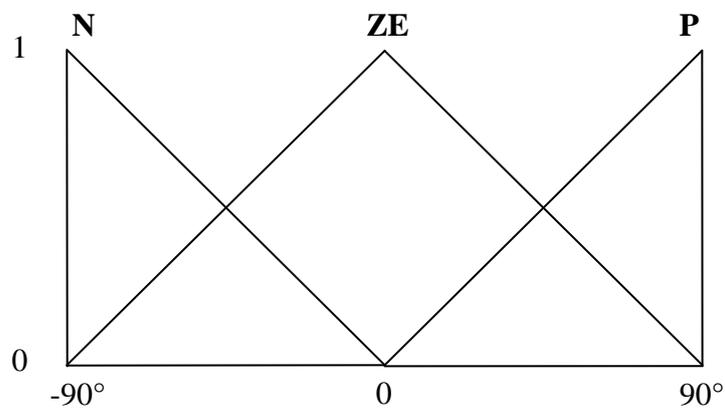


Figura 19 - Funções de pertinência do modelo do pêndulo invertido.

3.7.2 Projeto do Controlador via Alocação de Pólos

A partir da especificação desejada dos autovalores em malha fechada é possível calcular os ganhos \mathbf{F} de retroação de estados por meio de (20). Equaciona-se o sistema de modo a calcular $\mathbf{F} = -\mathbf{K}$, ou seja:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$\mathbf{F} = [F_1 \quad F_2], \quad \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\mathbf{A}_S = \begin{bmatrix} -b_1 a_{22} + b_1 \lambda_1 + b_2 a_{12} & -a_{11} b_2 + \lambda_1 b_2 + a_{21} b_1 \\ -b_1 a_{22} + b_1 \lambda_2 + b_2 a_{12} & -a_{11} b_2 + \lambda_2 b_2 + a_{21} b_1 \end{bmatrix},$$

$$\mathbf{B}_S = - \begin{bmatrix} a_{11} a_{22} - a_{11} \lambda_1 - \lambda_1 a_{22} + \lambda_1^2 - a_{21} a_{12} \\ a_{11} a_{22} - a_{11} \lambda_2 - \lambda_2 a_{22} + \lambda_2^2 - a_{21} a_{12} \end{bmatrix},$$

$$\mathbf{F}^T = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}.$$

Escolhendo os autovalores (para modelo contínuo) em $[-2, -2]$ para a malha de controle resultante, os seguintes ganhos obtidos são:

$$\mathbf{F}_1 = [-120,6667 \quad -22,6667];$$

$$\mathbf{F}_2 = [-2551,6 \quad -764].$$

Aplicando-se o Teorema 3, inicialmente se calcula a matriz \mathbf{G}_{12} :

$$\mathbf{G}_{12} = \begin{bmatrix} 0 & 1 \\ -212,1325 & -67,4675 \end{bmatrix}.$$

Usando o algoritmo de otimização LMI do MATLAB, encontra-se:

$$\mathbf{P} = \begin{bmatrix} 0,1889 & 0,0269 \\ 0,0269 & 0,0126 \end{bmatrix}.$$

Sendo a matriz comum \mathbf{P} definida positiva, satisfazendo as condições de estabilidade do Teorema 4, portanto, o sistema controlado é assintoticamente estável. O controle PDC resultante é dado por:

$$\begin{aligned} \text{Regra 1: } & \text{SE } x_1 \text{ é ZE} \\ & \text{ENTÃO } u = -\mathbf{F}_1 \mathbf{x} \\ \text{Regra 2: } & \text{SE } x_1 \text{ é N (ou P)} \\ & \text{ENTÃO } u = -\mathbf{F}_2 \mathbf{x} \end{aligned}$$

Onde

$$u = -w_1 \mathbf{F}_1 \mathbf{x} - w_2 \mathbf{F}_2 \mathbf{x}.$$

Sendo w_1 e w_2 os valores das Regras 1 e 2, respectivamente. Esta lei de controle garante a estabilidade do sistema fuzzy controlado (modelo *fuzzy* + controle PDC). Portanto, aplica-se este controlador PDC ao sistema original (28) para efetivação do controle e realização de simulações computacionais.

A Figura 20 mostra a resposta do sistema controlado para as condições iniciais $x_1 = 1^\circ, 5^\circ, 10^\circ$, e $x_2 = 0$. A Figura 21 mostra a resposta para condições iniciais envolvendo os ângulos $x_1 = 15^\circ, 30^\circ, 45^\circ$, e $x_2 = 0$. A partir das simulações (Figura 20 e Figura 21), conclui-se que o sistema de controle é efetivo, equilibrando o pêndulo para as condições operacionais consideradas.

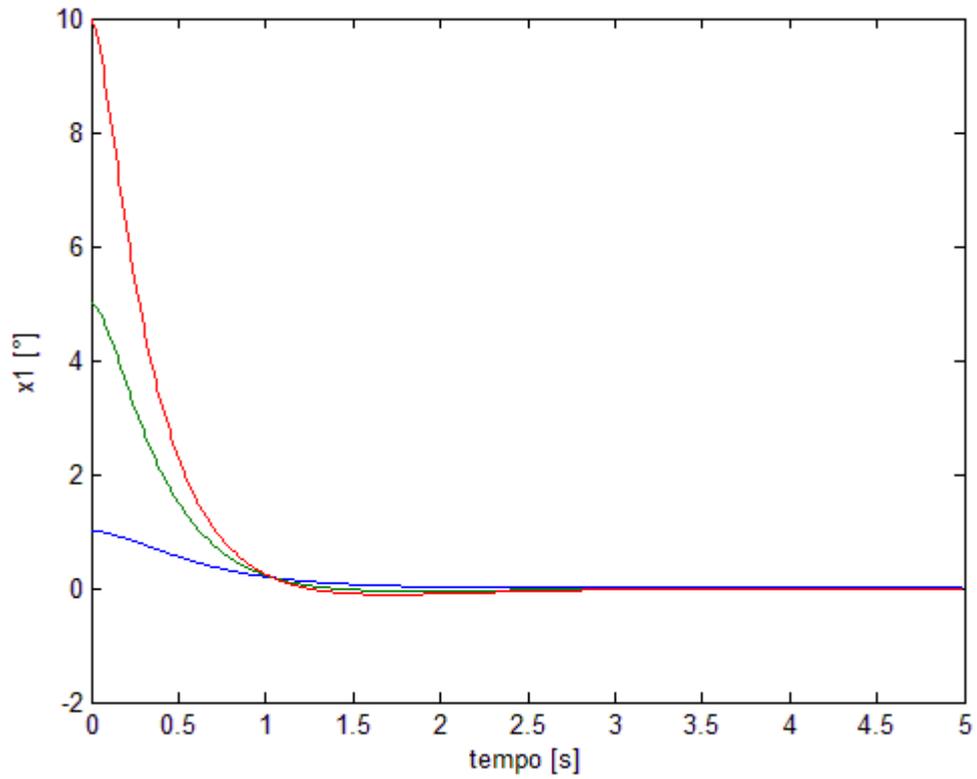


Figura 20 – Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

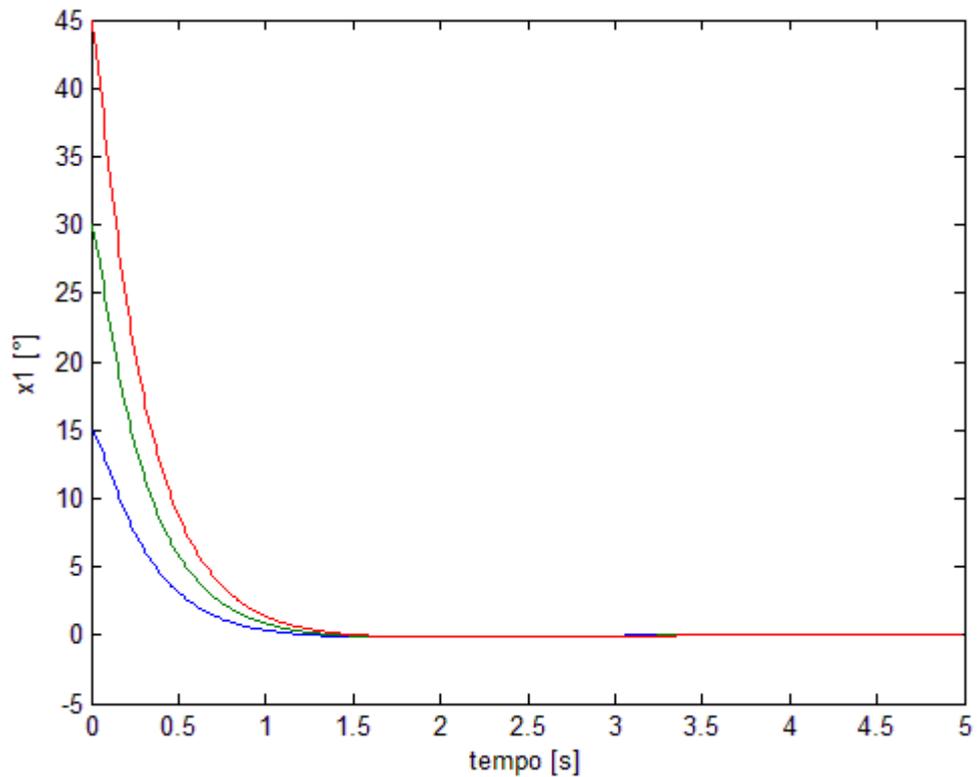


Figura 21 – Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

3.7.3 Projeto do Controlador via Técnica de Controle Ótimo

Um sistema de controle ótimo possibilita que o processo tenha um determinado desempenho definido por uma função de custo (22).

Adotando os parâmetros $\mathbf{Q} = [0,25 \ 0; 0 \ 1]$ e $R = 1$, os seguintes valores de $\mathbf{F} = \mathbf{K}$ são obtidos pela equação de Riccati (24) e (25):

$$\mathbf{F}_1 = [-196,0013 \quad -47,1418];$$

$$\mathbf{F}_2 = [-3575,3 \quad -1168,6].$$

Assim:

$$\mathbf{G}_{12} = \begin{bmatrix} 0 & 1 \\ -302,657 & -103,2382 \end{bmatrix};$$

$$\mathbf{P} = \begin{bmatrix} 0,171 & 0,0094 \\ 0,0094 & 0,0055 \end{bmatrix}.$$

A matriz \mathbf{P} definida positiva satisfaz as condições de estabilidade do Teorema 4. Portanto, o sistema controlado é assintoticamente estável. As regras de controle são as mesmas do exemplo anterior e representam o sistema de controle PDC resultante.

A Figura 22 mostra a resposta do sistema controlado para condições iniciais envolvendo os ângulos $x_1 = 1^\circ, 5^\circ, 10^\circ$, e $x_2 = 0$. A Figura 23 mostra a resposta para condições iniciais envolvendo os ângulos $x_1 = 15^\circ, 30^\circ, 45^\circ$, e $x_2 = 0$.

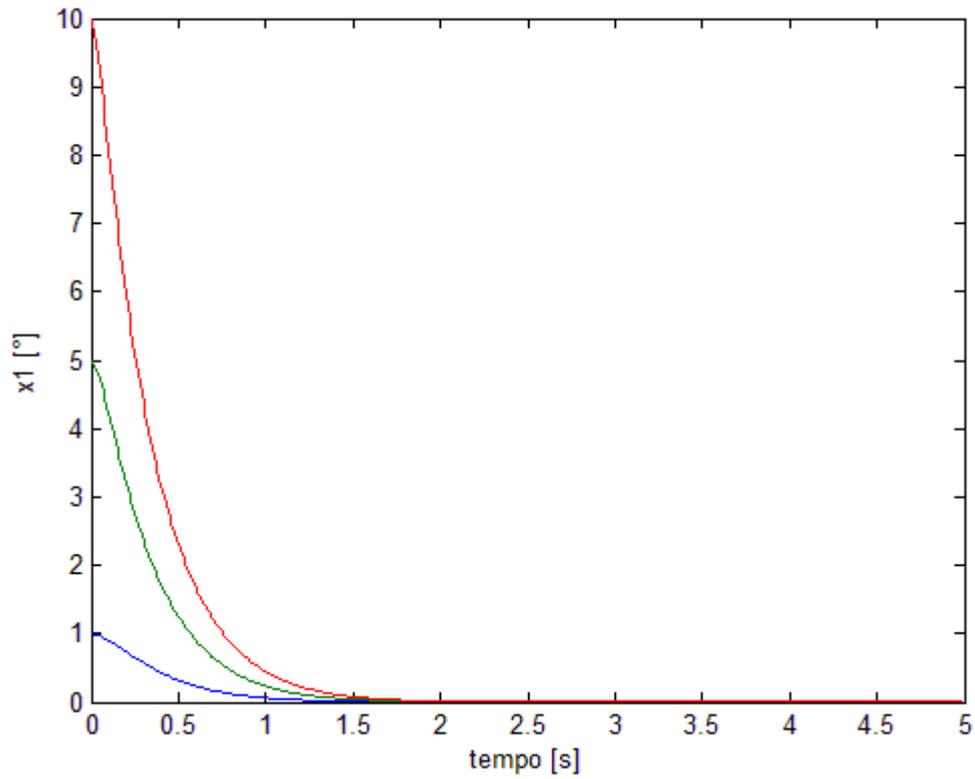


Figura 22 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

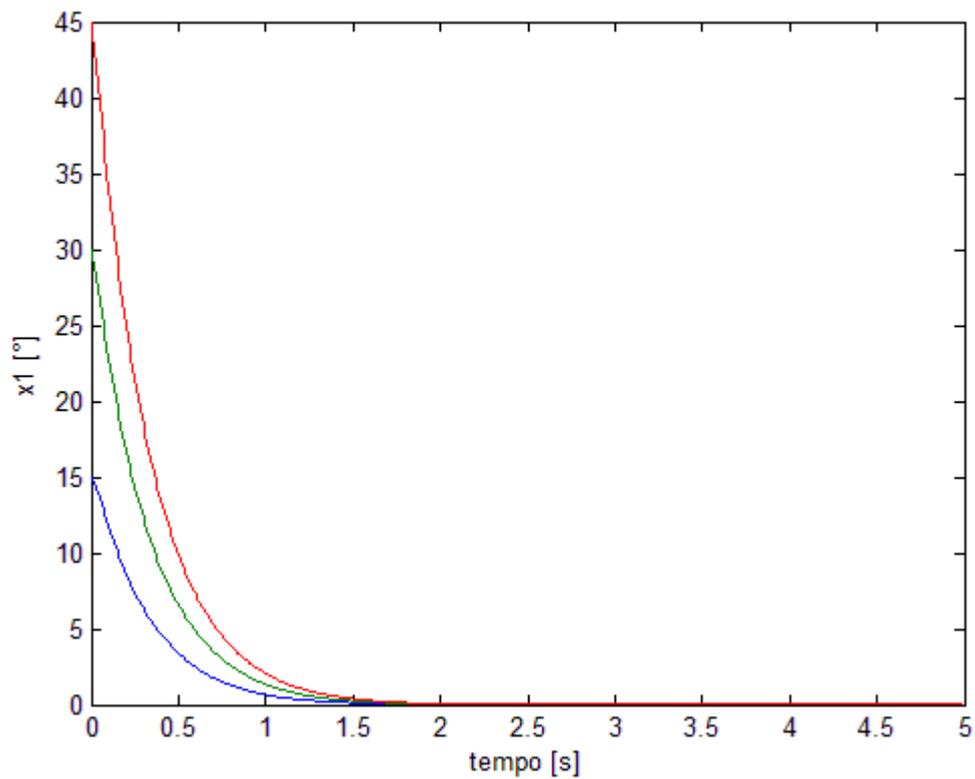


Figura 23 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

A partir das simulações (Figura 22 e Figura 23), conclui-se que o controle é efetivo equilibrando o pêndulo para as condições operacionais consideradas.

3.7.4 Projeto do controlador via LMI

O Teorema 5 utiliza LMI's para resolver diretamente o problema do projeto de controle *fuzzy*, satisfazendo as condições de estabilidade de Lyapunov. Aplicando-se inicialmente o Teorema 5, os seguintes ganhos são obtidos:

$$\begin{aligned} \mathbf{F}_1 &= [-36,9343 \quad -115,9781]; \\ \mathbf{F}_2 &= [-681,900 \quad -2205,90]. \end{aligned}$$

Assim:

$$\mathbf{G}_{12} = \begin{bmatrix} 0 & 1 \\ -46,9332 & -194,9457 \end{bmatrix}$$

e

$$\mathbf{P} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Os elementos da matriz \mathbf{P} resultante tendem a zero não satisfazendo as condições de estabilidade do Teorema 4. A Figura 24 ilustra a resposta do sistema para os ângulos $x_1 = 1^\circ, 5^\circ, 10^\circ$, e $x_2 = 0$. A Figura 25 mostra a resposta para condições iniciais envolvendo os ângulos $x_1 = 15^\circ, 30^\circ, 45^\circ$, e $x_2 = 0$.

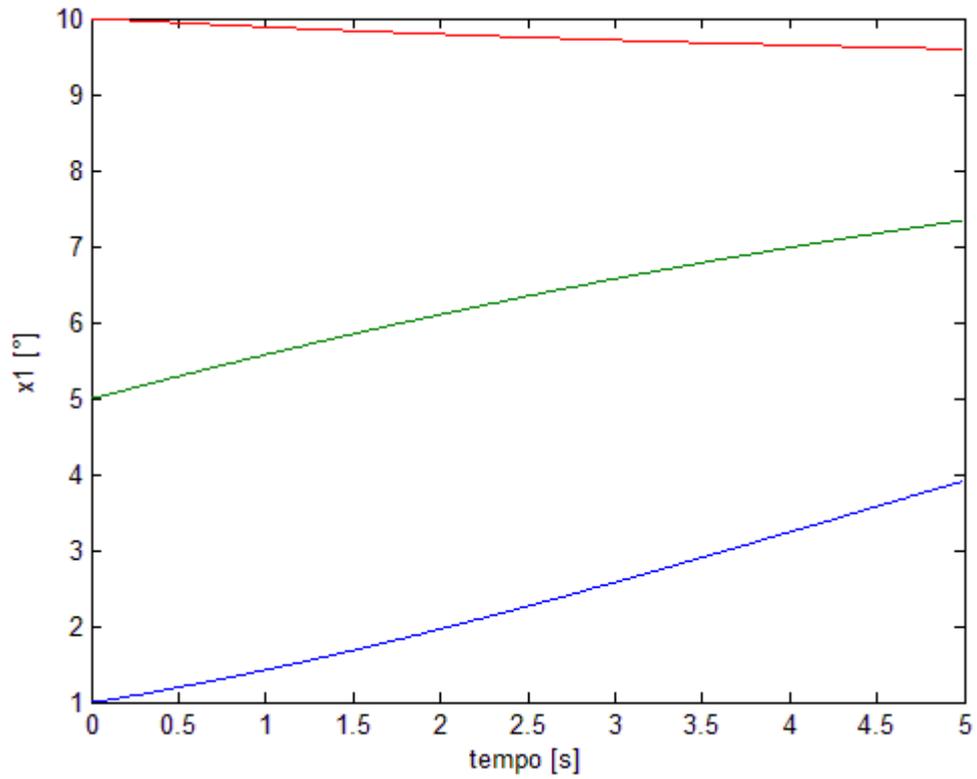


Figura 24 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

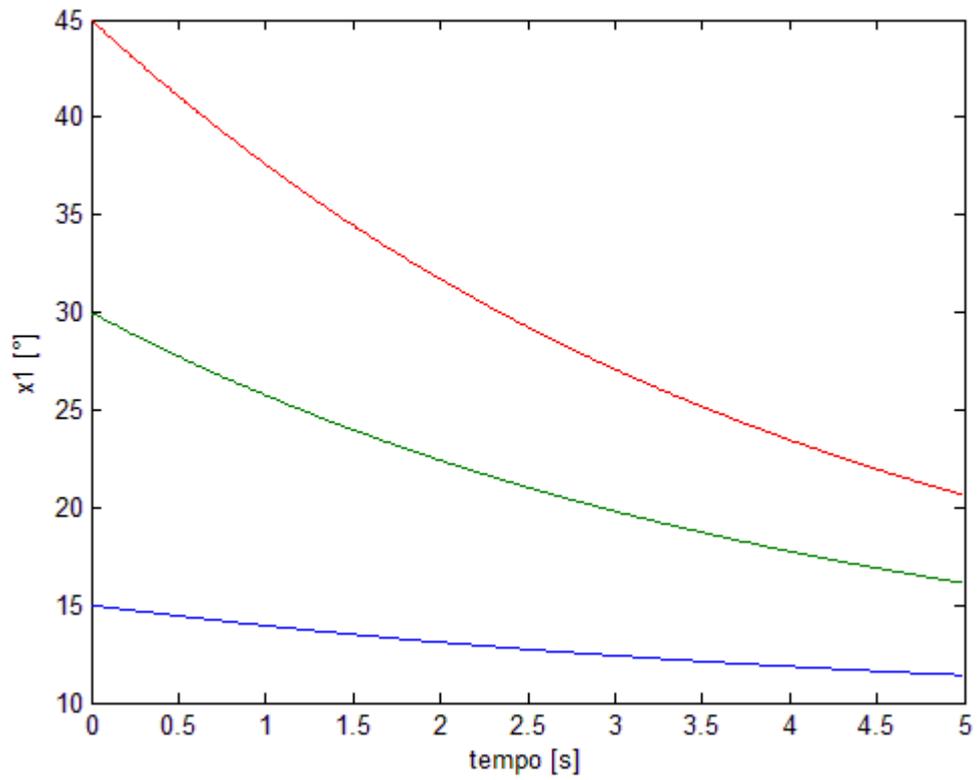


Figura 25 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

Observa-se que neste exemplo o sistema de controle não foi adequado, comprovando o não cumprimento das condições de estabilidade. Logo, o Teorema 5, que considera apenas a estabilidade, é insuficiente para estabilizar o sistema.

No entanto, quando se impõe restrições ao sistema, soluções podem ser encontradas. Por isso, de posse do Teorema 6, que leva em consideração restrições de desempenho, os seguintes ganhos são obtidos para taxa de decaimento $\gamma = 0,0006068$ adotada após várias tentativas para obtenção de melhores respostas:

$$\begin{aligned} F_1 &= [-224,1219 \quad -91,7925], \\ F_2 &= [-13038 \quad -2749]. \end{aligned}$$

Assim,

$$G_{12} = \begin{bmatrix} 0 & 1 \\ -1137,7 & -242,8 \end{bmatrix}$$

e

$$P = \begin{bmatrix} 0,0703 & 0,0019 \\ 0,0019 & 0,0007 \end{bmatrix}.$$

A taxa de decaimento utilizada foi obtida após algumas simulações que apresentaram um melhor desempenho para o sistema. Sendo a matriz comum P definida positiva, satisfazendo as condições de estabilidade do Teorema 4, portanto, o sistema controlado é assintoticamente estável.

A Figura 26 mostra a resposta do sistema controlado para condições iniciais envolvendo os ângulos: $x_1 = 1^\circ, 5^\circ, 10^\circ$, e $x_2 = 0$. A Figura 27 mostra a resposta para condições iniciais com os ângulos $x_1 = 15^\circ, 30^\circ, 45^\circ$, e $x_2 = 0$.

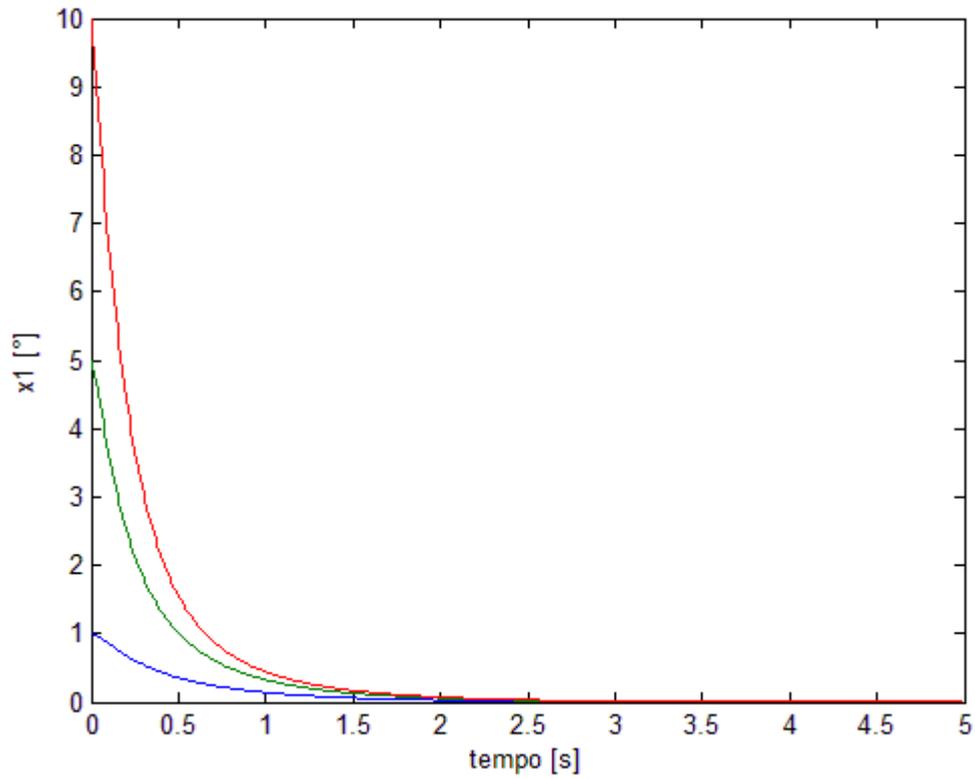


Figura 26 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

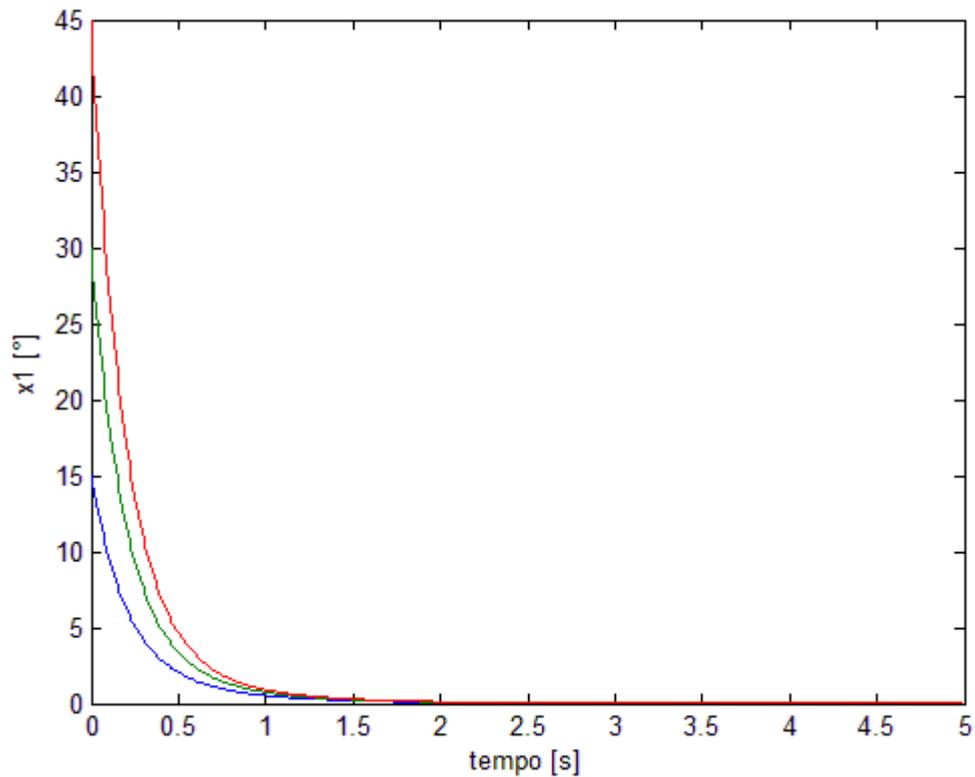


Figura 27 - Respostas do sistema de pêndulo invertido para diferentes condições iniciais.

3.8 Resumo da Metodologia de Projeto

Conforme visto nas aplicações anteriores, as etapas da metodologia apresentada consistem em:

- 1º) Construir o modelo Takagi-Sugeno;
- 2º) Projetar o controle PDC usando técnicas de retroação de estado utilizando técnicas convencionais de projeto para cada equação linear do modelo obtido;
- 3º) Verificar estabilidade por Lyapunov usando LMI's;
- 4º) Validar controle através da realização de simulações e/ou ensaios.

Para o exemplo apresentado, partiu-se do modelo físico para a construção do modelo *fuzzy* Takagi-Sugeno e por fim foram realizadas simulações computacionais para comprovação de metodologia aplicada. Isto pode ser visto na Figura 28 através dos procedimentos indicados em cor vermelha na ilustração.

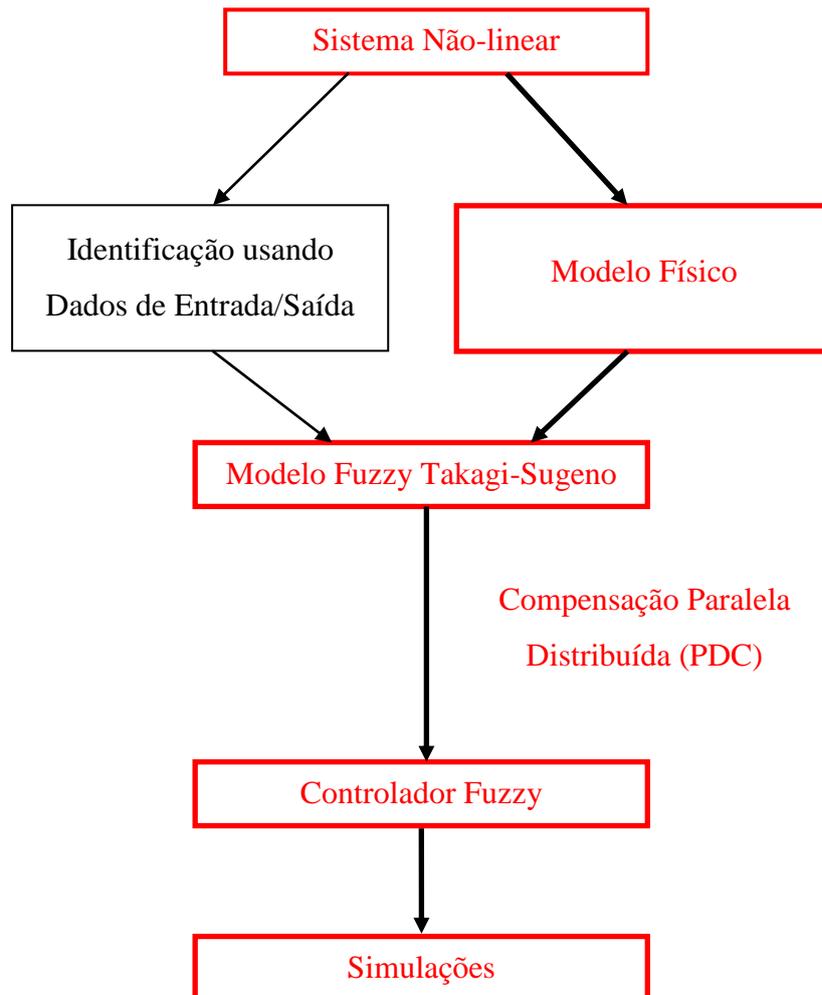


Figura 28 - Processo de Projeto

Um dos objetivos deste trabalho é validar a metodologia de projeto apresentada através de experimentos práticos. Na compilação bibliográfica, não se encontrou ensaios práticos realizados neste contexto, mas somente simulações computacionais.

Neste sentido outro objetivo desta dissertação é realizar a implementação real de um controlador com a estrutura PDC mostrada. Esta pesquisa se iniciou através de simulações computacionais, e por fim se concretizará através de ensaios experimentais na implantação real de um controlador *fuzzy* projetado conforme os procedimentos mostrados.

Para a aplicação da metodologia, um modelo *fuzzy* Takagi-Sugeno será obtido por meio de processos de identificação de sistemas, usando dados de entrada/saída medidos de um processo real. As demais etapas de projeto se seguirão conforme ilustrado na Figura 29 (caminho em cor vermelha), finalizando com ensaios experimentais realizados em um laboratório da universidade.

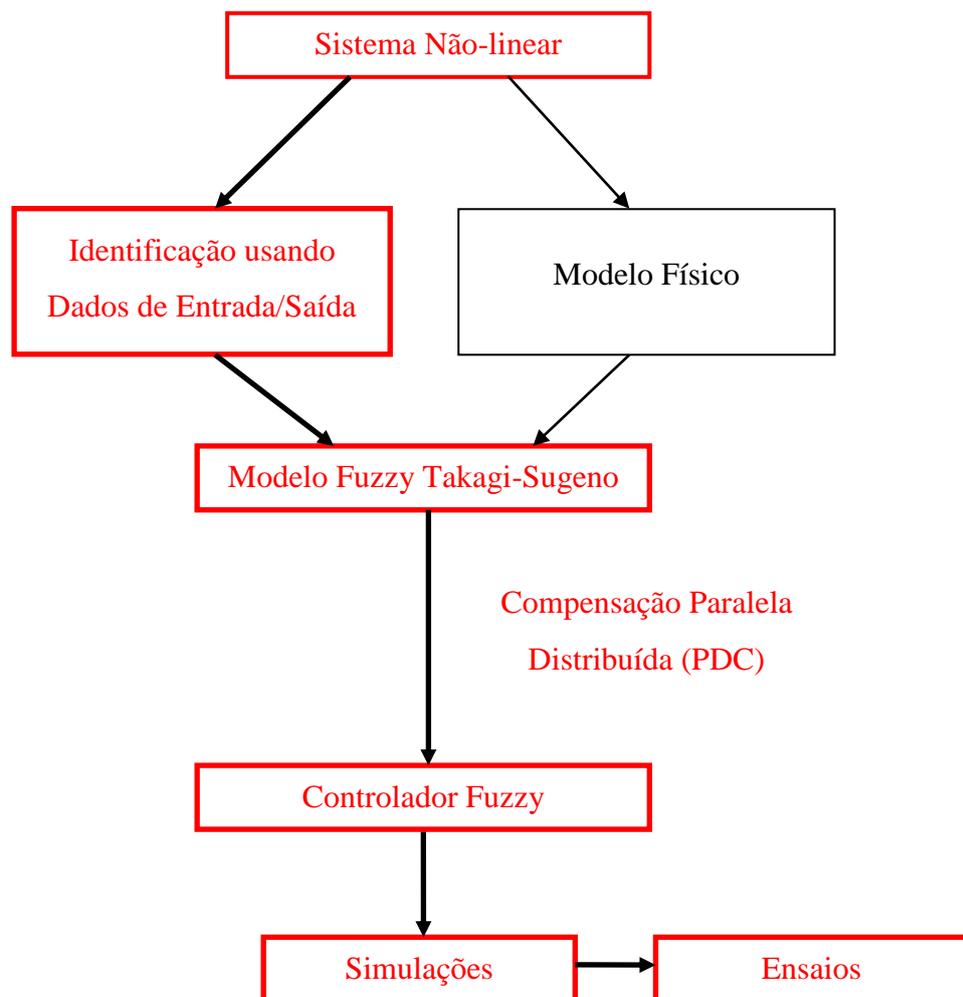


Figura 29 - Processo de projeto para projetos experimentais.

4. Metodologia

Toda a metodologia apresentada é aplicada a um sistema de controle de nível real, sendo este a base para simulações computacionais e ensaios práticos. O processo em questão é representado em escala reduzida por um processo de nível que possui as seguintes partes constituintes:

- um módulo eletrônico responsável pela instrumentação do processo;
- um sistema de aquisição de dados;
- um computador responsável por executar o algoritmo de controle.

A bancada de ensaio utilizada está ilustrada na foto da Figura 30.

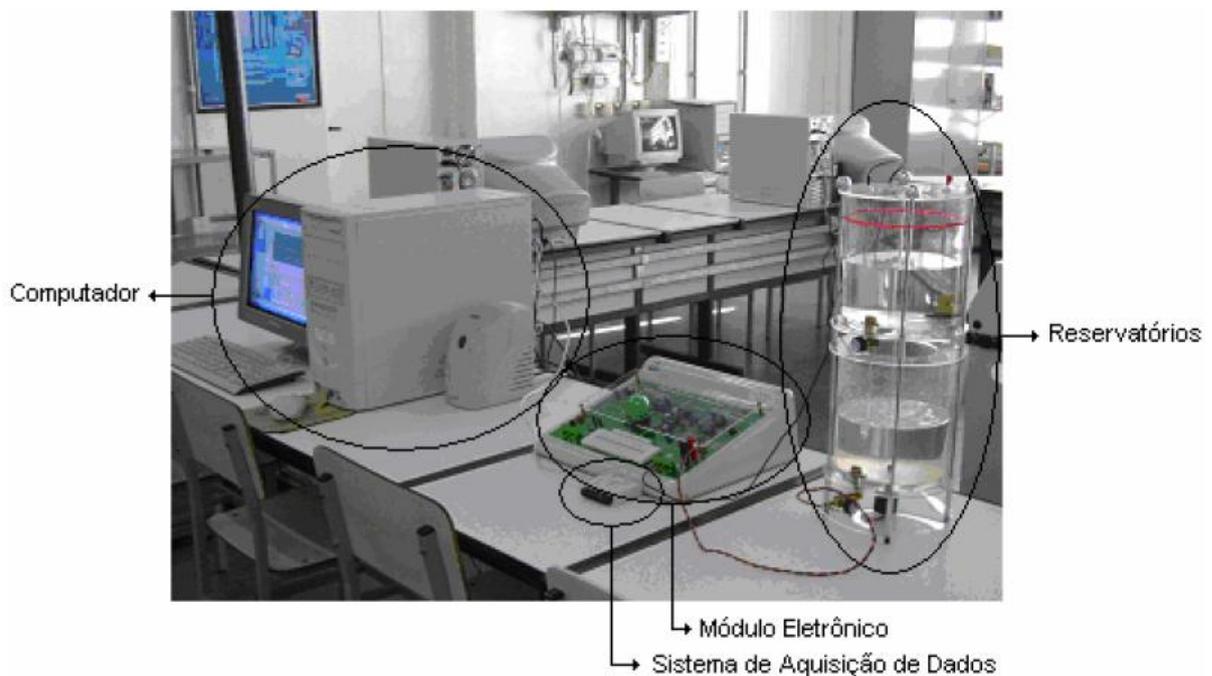


Figura 30 - Bancada de ensaio.

O sistema de nível é composto por dois reservatórios sobrepostos e interligados. Sem a atuação do sistema de controle, o líquido (água) do processo é acumulado no reservatório inferior. A atuação no sistema é realizada por uma servo-bomba elétrica que transfere o líquido do reservatório inferior para o reservatório superior, onde se deseja controlar o nível no mesmo. O reservatório superior possui uma escala para inspeção visual do nível controlado.

O nível é medido por meio de duas hastes metálicas em contato com o fluido que representa um transdutor capacitivo. A variação do nível será percebida como uma variação no dielétrico do capacitor associado às hastes. O módulo eletrônico condiciona o sinal do transdutor capacitivo. A informação resultante é obtida através de um sistema de aquisição de dados (DAQ) acoplado a um computador. A informação de comando responsável pela atuação da servo-bomba do processo, também é fornecida pelo DAQ. Essas informações são computadas pelo algoritmo de controle processado no computador, gerando um sinal de controle para o *driver* de acionamento da servo-bomba. O sistema de aquisição de dados utilizado (modelo NI USB-6008 da National Instruments) possui 8 entradas analógicas e 2 saídas analógicas de 12 bits, 12 entradas/saídas digitais programáveis, um contador de 32 bits, além de uma interface USB para comunicação com o computador utilizado para implementar o algoritmo de controle.

4.9 Modelagem do Sistema Nível de um Líquido

4.9.1 Modelagem Física

Vários processos industriais envolvem o escoamento de líquidos através de reservatórios e tubulações Figura 31 (Ogata, 2003). O fluxo nestes processos geralmente é turbulento e não-laminar. Os regimes de escoamento são classificados de acordo com o valor do número de Reynolds. Sistemas envolvendo escoamento turbulento, na maioria das vezes, têm de ser representados por equações diferenciais não-lineares, enquanto sistemas envolvendo escoamento laminar podem ser representados por equações diferenciais lineares.

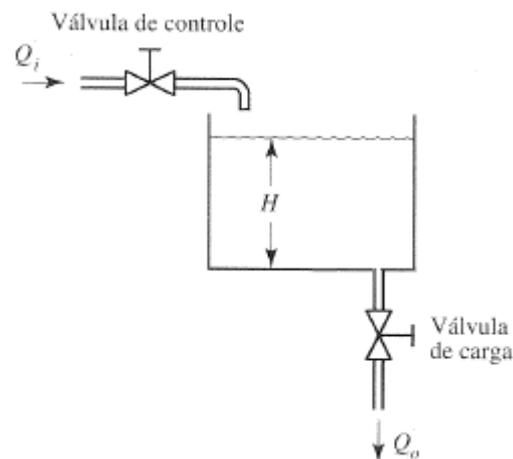


Figura 31 – Estrutura típica de um sistema de nível.

Considerando um reservatório com uma seção reta de área A , a lei de conservação de massa expressa a relação:

$$A \frac{dH}{dt} = Q_i - Q_o.$$

É conhecido que a vazão de um fluido através de um orifício é normalmente uma função da raiz quadrada da queda de pressão através do mesmo, ou seja:

$$Q_o = K_t \sqrt{H}.$$

Substituindo as duas expressões acima e reagrupando os termos, obtém o seguinte modelo:

$$A \frac{dH}{dt} + K_t \sqrt{H} = Q_i.$$

Constituindo uma equação diferencial não-linear (Martins de Carvalho, 2000).

4.9.2 Construção do modelo Takagi-Sugeno a partir de Ensaios de um Sistema de Nível

Dados de ensaio

Foram realizados, inicialmente, ensaios na planta efetuando-se medidas das informações de entrada e saída do processo, objetivando a modelagem do sistema em questão (Figura 32).

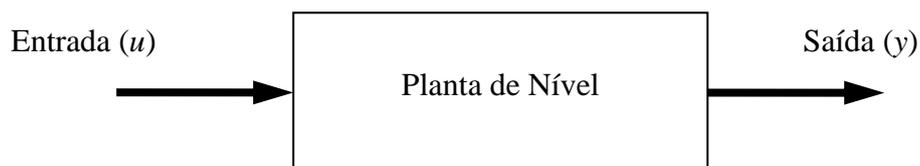


Figura 32 - Obtenção dos dados de entrada e saída para modelagem do sistema de nível.

O procedimento consistiu em se variar a entrada u , registrando a saída y (nível). Foram coletados e armazenados pares de informações (u, y) variando no tempo.

Foi utilizado o *toolbox* ANFIS do software MATLAB para encontrar o modelo Takagi-Sugeno correspondente do processo. As informações $u(k)$ e $y(k)$ farão parte dos dados de entrada do sistema *fuzzy*, enquanto $y(k + 1)$ fará parte dos dados de saída para o treinamento ANFIS. A Figura 33 ilustra as informações consideradas na modelagem do sistema. A Figura 34 mostra dados do ensaio realizado no processo de nível utilizado.

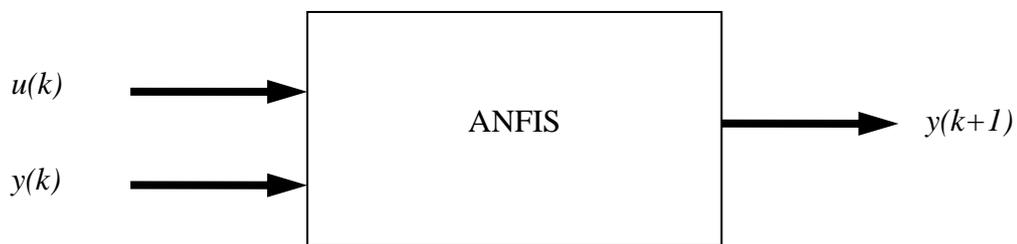


Figura 33 - Dados de treinamento para o ANFIS.

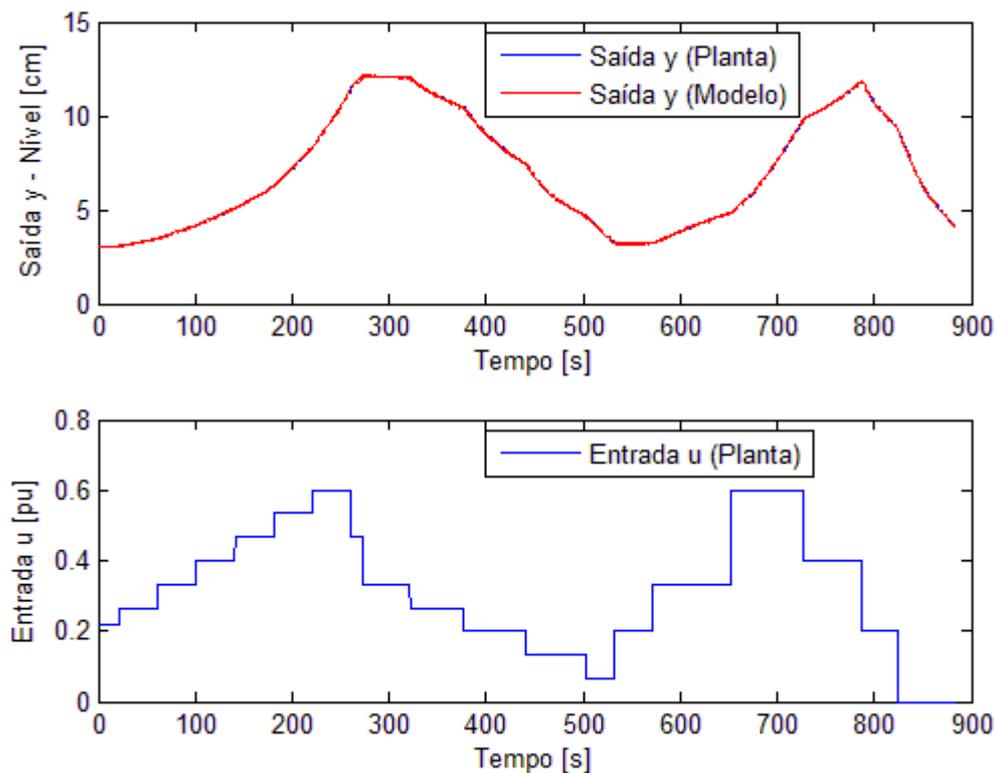


Figura 34 – Dados do ensaio para modelagem e os resultantes do modelo obtido.

A Figura 34 mostra outro conjunto de dados de entrada/saída medidos e utilizados para validar a modelagem *fuzzy* obtida, objetivando validar o modelo resultante. Dos gráficos da Figura 35 é possível notar que o sistema difuso obtido apresenta uma boa aproximação do processo real. Idem para outro conjunto de dados mostrados na Figura 36.

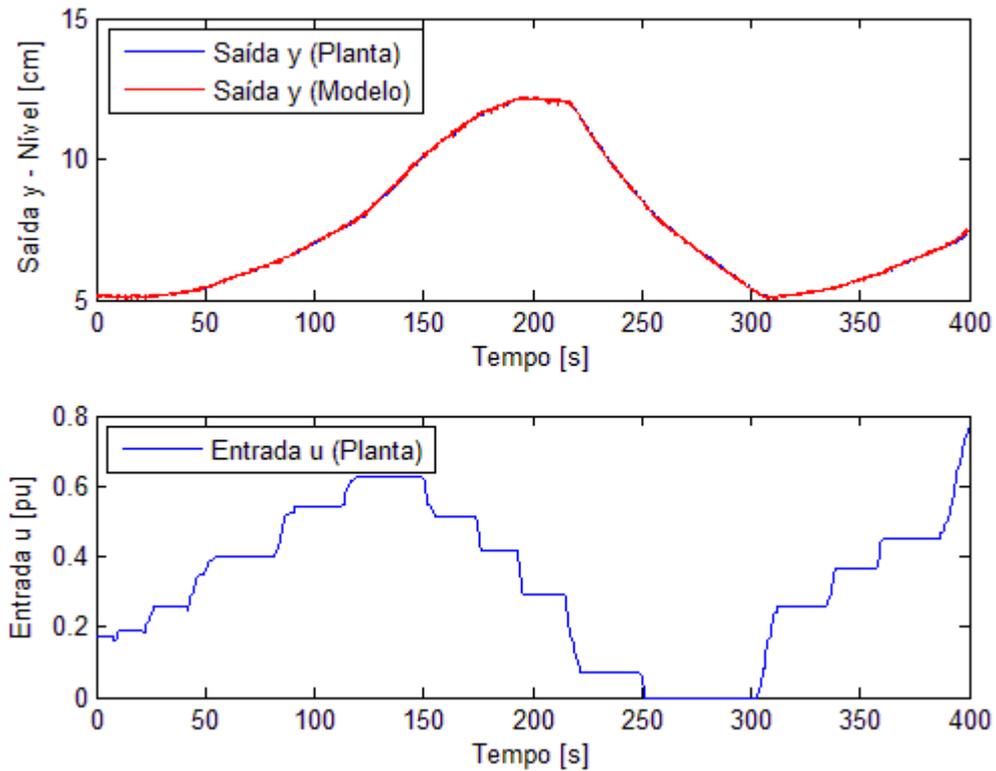


Figura 35 - Dados de ensaio para validação do modelo.

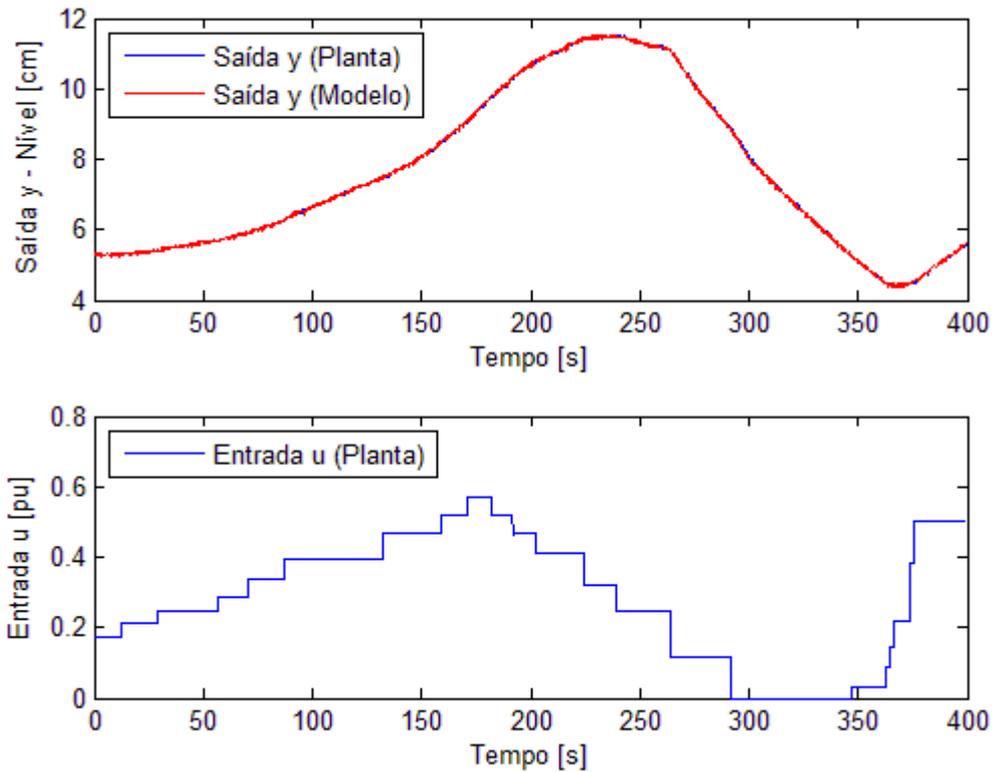


Figura 36 – Dados de ensaio para validação do modelo.

Funções de Pertinência de Entrada do Modelo Fuzzy

As funções de pertinência para a entrada do modelo *fuzzy* obtidas pelo ANFIS foram definidas como sendo do tipo trapezoidais. No modelo do sistema *fuzzy* considerado (Figura 37) adotou-se as informações $u(k)$ e $y(k)$ como variáveis de entrada, e $y(k+1)$ como informação de saída. Para cada variável de entrada foram obtidas duas funções de pertinência (Figura 37 e Figura 39).

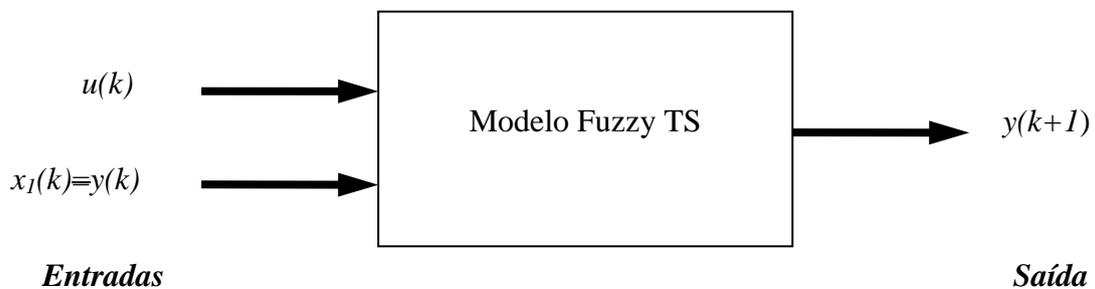


Figura 37 - Modelo fuzzy correspondente a planta de nível modelada.

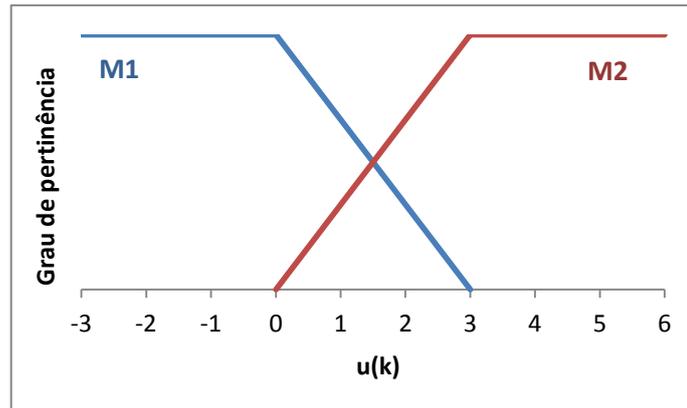


Figura 38 – Funções de pertinência da variável $u(k)$.

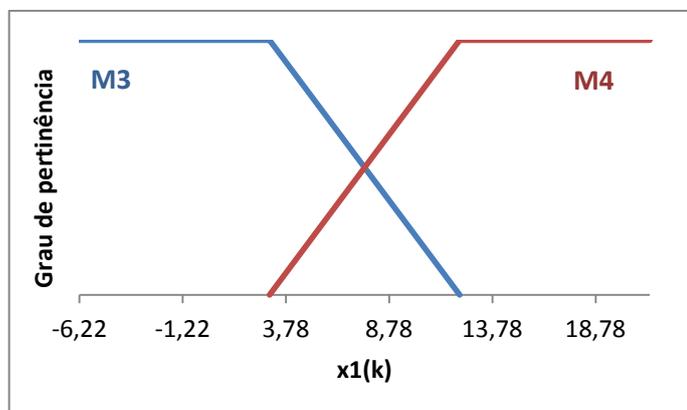


Figura 39 – de pertinência da variável $x_1(k) = y(k)$.

Funções polinomiais e regras fuzzy

As regras obtidas pelo ANFIS e os parâmetros das funções polinomiais de saída são:

$$\text{Regra 1: } \begin{array}{l} SE \ x_1(k) = M3 \ e \ u(k) = M1 \\ ENT\tilde{A}O \ x_1(k+1) = 0,9856 \times x_1(k) + 0,0536 \times u(k) + 0,0396; \end{array}$$

$$\text{Regra 2: } \begin{array}{l} SE \ x_1(k) = M3 \ e \ u(k) = M2 \\ ENT\tilde{A}O \ x_1(k+1) = 0,9847 \times x_1(k) + 0,2241 \times u(k) + 0,1519; \end{array}$$

$$\text{Regra 3: } \begin{array}{l} SE \ x_1(k) = M4 \ e \ u(k) = M1 \\ ENT\tilde{A}O \ x_1(k+1) = 0,9373 \times x_1(k) + 0,0519 \times u(k) + 0,0351; \end{array}$$

$$\text{Regra 4: } \begin{array}{l} SE \ x_1(k) = M4 \ e \ u(k) = M2 \\ ENT\tilde{A}O \ x_1(k+1) = 0,9369 \times x_1(k) + 0,2112 \times u(k) + 0,1451. \end{array}$$

Os parâmetros resultantes do modelo são:

$$A_1 = 0,9856;$$

$$A_2 = 0,9847;$$

$$A_3 = 0,9373;$$

$$A_4 = 0,9369;$$

$$B_1 = 0,0536;$$

$$B_2 = 0,2241;$$

$$B_3 = 0,0519;$$

$$B_4 = 0,2112;$$

$$C_1 = 0,0396;$$

$$C_2 = 0,1519;$$

$$C_3 = 0,0351;$$

$$C_4 = 0,1451.$$

Destaca-se que foram obtidos termos C_i não nulos nas funções de modelagem. Este parâmetro é chamado de *bias*, uma constante das funções polinomiais do modelo. O modelo *fuzzy* original proposto por Takagi e Sugeno (1985) é um modelo afim, possui a parte dos consequentes com termos *bias* não nulos. Entretanto, a maioria dos trabalhos que adotam sistemas fuzzy TS, utilizam o “modelo homogêneo”. Um sistema *fuzzy* TS homogêneo não possui termos com *bias*.

Para o projeto do sistema de controle é utilizado o modelo homogêneo no procedimento de cálculo dos ganhos da retroação de estado. Assim, o projeto do controlador pode ser feito com LMIs, simplificando o equacionamento do problema. O uso de modelos afins, em geral, é descrito por Desigualdades Bilineares Matriciais (BMIs) cuja solução computacional é muito mais complexa.

4.10 Projeto do Controlador Fuzzy

O conceito de projeto de um sistema de controle com estrutura PDC é bem direto. Para cada regra do modelo *fuzzy* TS projeta-se um controlador usando técnicas convencionais por retroação de estados. No entanto, antes de se calcular os ganhos é preciso preparar as

equações do sistema para que se obtenha erro nulo em regime permanente na atuação da malha de controle resultante.

4.10.1 Erro nulo em Regime Permanente

Para obter erro nulo em regime permanente em malhas de controle por retroação de estados, deve-se introduzir um integrador (com um ganho k_i) na malha de controle em questão, caso o modelo do sistema não contenha nenhum pólo na origem. As matrizes do sistema tomarão a seguinte forma para efeito de cálculo dos ganhos de retroação para modelos discretos:

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} * T & 1 \end{bmatrix};$$

$$\hat{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix}.$$

Onde T é o tempo de amostragem adotado na malha de controle em questão. Os ganhos de retroação resultantes ficam na forma:

$$\hat{\mathbf{K}} = [\mathbf{K} \quad -k_i].$$

Considerado $T = 0,1$ [s] nas simulações e nos ensaios realizados, as matrizes e vetores resultantes são:

$$\hat{\mathbf{A}}_1 = \begin{bmatrix} 0,9856 & 0 \\ -0,1 & 1 \end{bmatrix},$$

$$\hat{\mathbf{A}}_2 = \begin{bmatrix} 0,9847 & 0 \\ -0,1 & 1 \end{bmatrix},$$

$$\hat{\mathbf{A}}_3 = \begin{bmatrix} 0,9373 & 0 \\ -0,1 & 1 \end{bmatrix},$$

$$\hat{\mathbf{A}}_4 = \begin{bmatrix} 0,9369 & 0 \\ -0,1 & 1 \end{bmatrix},$$

$$\hat{\mathbf{B}}_1 = \begin{bmatrix} 0,0536 \\ 0 \end{bmatrix},$$

$$\hat{\mathbf{B}}_2 = \begin{bmatrix} 0,2241 \\ 0 \end{bmatrix},$$

$$\hat{\mathbf{B}}_3 = \begin{bmatrix} 0,0519 \\ 0 \end{bmatrix},$$

$$\hat{\mathbf{B}}_4 = \begin{bmatrix} 0,2212 \\ 0 \end{bmatrix}.$$

4.10.2 Matriz de Ganhos via Alocação de Pólos

A partir da especificação da equação característica desejada para o sistema, é possível determinar a matriz de ganhos $\hat{\mathbf{K}}_i$ para o controlador de cada conseqüente linear do modelo TS por meio da técnica de alocação de pólos.

Tratando-se de cada conseqüente linear, se os autovalores desejados para a matriz $\hat{\mathbf{A}} - \hat{\mathbf{B}}\hat{\mathbf{K}}$ (os pólos em malha fechada) forem especificados como sendo p_1, p_2, \dots, p_n , então a matriz de ganho de retroação \mathbf{K} e a constante do ganho integral k_i podem ser determinadas. Num projeto prático, é necessário considerar várias matrizes $\hat{\mathbf{K}}$ diferentes (o que corresponde a vários conjuntos distintos de autovalores desejados) e proceder simulações de modo a encontrar uma condição que conduza ao melhor desempenho global do sistema. Após alguns testes e simulações, foram escolhidos os seguintes valores para os pólos (autovalores) em malha fechada $[-0,1 \quad -0,1]$. Estes pólos se encontram dentro do círculo unitário ao serem mapeados no plano z para o sistema discreto, assegurando a estabilidade (Phillips, C. L.; Nagle, H. T., 1995).

Estes pólos serão alocados para cada conseqüente linear resultante das matrizes $\hat{\mathbf{A}}_i$ e $\hat{\mathbf{B}}_i$ ($i = 1, 2, \dots, r$) correspondentes a modelo *fuzzy* em questão. Como resultado de (20) e (21), foram obtidos os ganhos:

$$\hat{\mathbf{K}}_1 = [k_1 \quad -k_{i1}] = [22,1319 \quad 1,8668];$$

$$\hat{\mathbf{K}}_2 = [k_2 \quad -k_{i2}] = [5,2863 \quad 0,4462];$$

$$\hat{\mathbf{K}}_3 = [k_3 \quad -k_{i3}] = [21,9169 \quad 1,9271];$$

$$\hat{\mathbf{K}}_4 = [k_4 \quad -k_{i4}] = [5,3818 \quad 0,4734].$$

Um importante resultado que será discutido posteriormente é a utilização da média dos ganhos para um controle por retroação de estados convencional. A matriz de ganhos médios resultante é:

$$\hat{\mathbf{K}}_{\text{médio}} = [k_{1\text{médio}} \quad -k_{i\text{médio}}] = [13,6792 \quad 1,1784].$$

4.10.3 Matriz de Ganhos via Sistemas de Controle Ótimo

Considerando o projeto de sistemas de controle estáveis baseados em índices de desempenho quadráticos, há o interesse de se escolher um vetor de controle tal que um dado índice de desempenho seja minimizado. Portanto, o projeto de sistemas de controle ótimo baseados em tais índices de desempenho se reduz à determinação dos elementos da matriz \mathbf{K} . No caso do sistema *fuzzy* TS, este conceito aplica-se para cada consequente linear das regras de modelagem.

Para a minimização de índices quadráticos de desempenho há necessidade de se resolver a equação de Riccati. Foram escolhidos os seguinte parâmetros da função de custo após alguns testes a fim de se obter melhores resultados:

$$R = 1 ;$$

$$\mathbf{Q}_1 = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix};$$

$$\mathbf{Q}_2 = \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix};$$

$$\mathbf{Q}_3 = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix};$$

$$\mathbf{Q}_4 = \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}.$$

Com (24) e (25) foram obtidos:

$$\hat{\mathbf{K}}_1 = [k_1 \quad -k_{i1}] = [37,38 \quad 3,1623];$$

$$\hat{\mathbf{K}}_2 = [k_2 \quad -k_{i2}] = [9,8892 \quad 1];$$

$$\hat{\mathbf{K}}_3 = [k_3 \quad -k_{i3}] = [36,7295 \quad 3,1623];$$

$$\hat{\mathbf{K}}_4 = [k_4 \quad -k_{i4}] = [9,9683 \quad 1].$$

A matriz média dos ganhos será:

$$\hat{\mathbf{K}}_{médio} = [k_{1médio} \quad -k_{imédio}] = [23,4918 \quad 2,0812].$$

4.10.4 Malhas de Controle Correspondentes

Em um sistema de controle convencional, o diagrama de blocos da malha de controle por retroação de estados resultante tem a forma básica descrita na Figura 40. É citada esta representação para comparação com a malha de controle *fuzzy* (Figura 41).

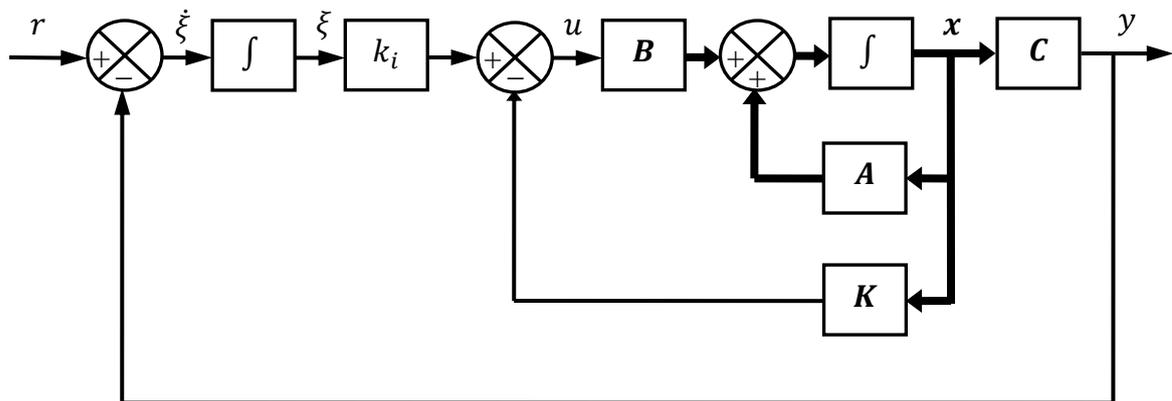


Figura 40 - Malha de controle convencional (Ogata, 2003).

A partir da Figura 40 é possível escrever (Ogata, 2003):

$$\begin{aligned}
 \dot{x} &= Ax + Bu \\
 y &= Cx \\
 u &= -Kx + k_i \xi \\
 \dot{\xi} &= r - y = r - Cx
 \end{aligned} \tag{29}$$

onde x = vetor de estado do processo a controlar, u = sinal de controle, y = saída, ξ = sinal de saída do integrador, r = sinal de entrada de referência, A , B e C = matrizes do sistema.

O sistema de controle *fuzzy* para a planta de nível em questão é representado pelo diagrama de blocos da Figura 41. Como a planta só tem uma variável de estado, esta será a própria saída y do sistema.

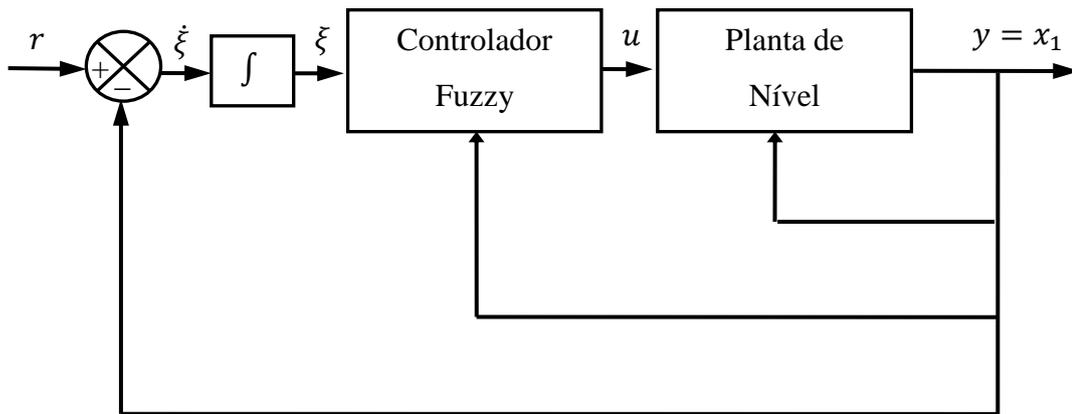


Figura 41 – Malha de controle fuzzy para a planta de nível.

Observa-se na Figura 41, a retroação do estado x_1 . Assim como na Figura 40, haverá a integração do erro da malha de controle visando à obtenção de erro nulo em regime permanente.

Comparando a malha de controle *fuzzy* (Figura 41) com a malha de controle convencional (Figura 40), pode-se concluir através das equações em (29) que a forma de se calcular o erro $\xi = r - y = r - x_1$ é a mesma para os dois tipos de sistema. No entanto, os cálculos da variável de estado x_1 e da informação de controle u são processados de forma diferente. A malha convencional lida diretamente com equações lineares, enquanto a malha de controle *fuzzy* realiza inferências *fuzzy*.

A Figura 42 ilustra a malha de controle através dos componentes do sistema. A referência de entrada é dada através de um computador que processa as informações de leitura do nível já condicionadas, gerando um sinal de controle. Este sinal é amplificado e aplicado ao atuador da planta, controlando o sistema. Observa-se que o controlador *fuzzy* é implementado através de um algoritmo computacional.

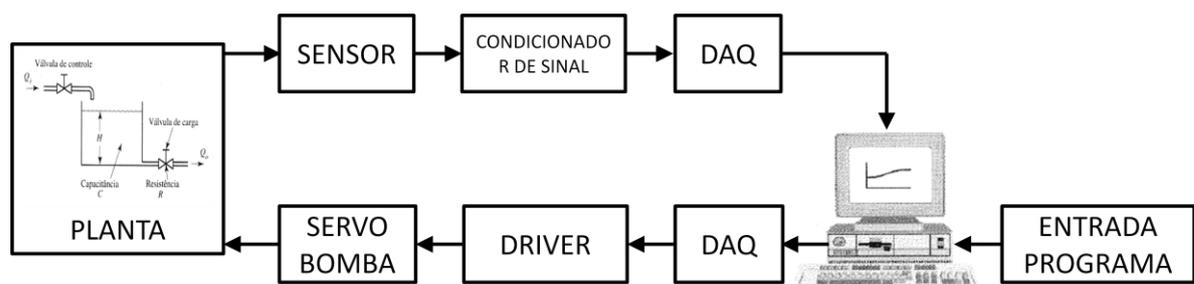


Figura 42 – Componentes da malha de controle da planta de nível.

Controlador fuzzy TS

A Figura 43 mostra como o valor da informação de controle u é obtida.

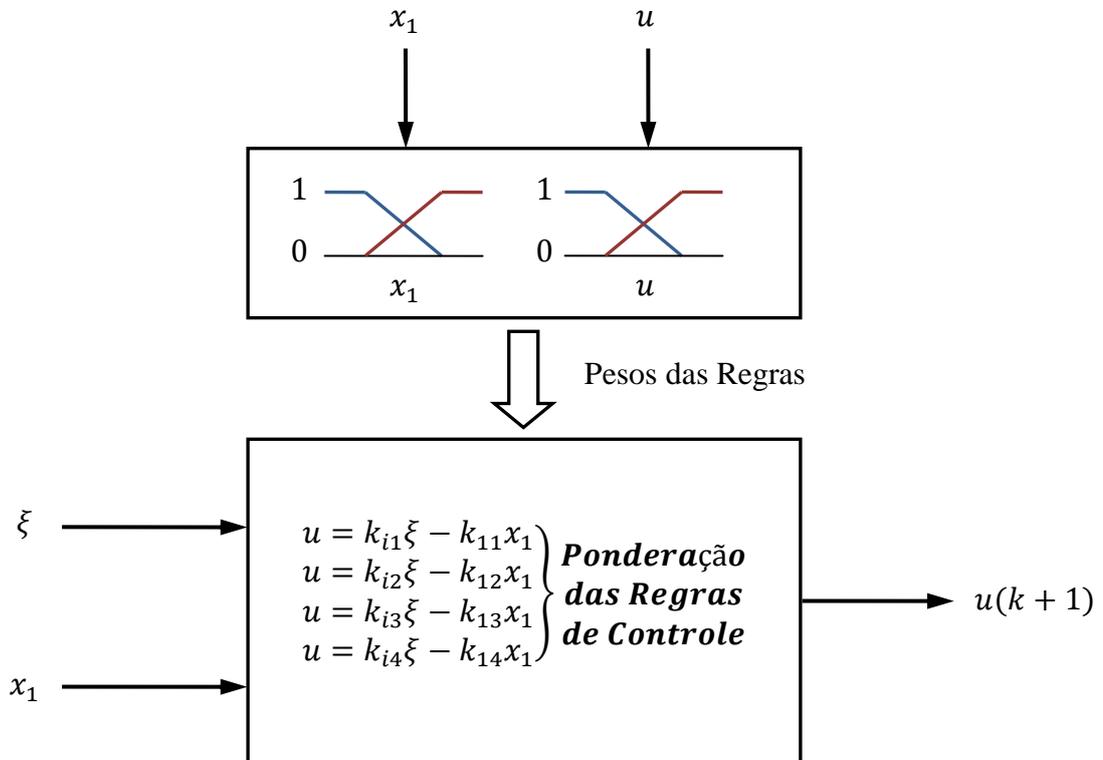


Figura 43 – As leis de controle do controlador fuzzy TS para a planta de nível.

A integral do erro ξ e a retroação do estado x_1 são as entradas do controlador *fuzzy*. Estes dados são processados através das regras do controlador, cujos conseqüentes são equações lineares. Cada informação de u de cada regra do controlador é ponderada de acordo com os seus correspondentes pesos. Os pesos das regras são dados pela operação AND das funções de pertinência das premissas das regras. O resultado é um dado u que será aplicado na atuação do sistema.

Funções de Pertinência das Variáveis de Entrada

O controlador *fuzzy* utiliza os mesmos conjuntos *fuzzy* (funções de pertinência de entrada) do modelo *fuzzy* obtido (Figura 38 e Figura 39). Isso facilita o projeto da técnica PDC.

Funções Polinomiais e Regras fuzzy

As regras obtidas do controlador são:

$$\text{Regra 1: } \begin{array}{l} \text{SE } x_1(k) = M3 \text{ e } u(k) = M1 \\ \text{ENTÃO } u(k+1) = k_{i1}\xi - k_1x_1; \end{array}$$

$$\text{Regra 2: } \begin{array}{l} \text{SE } x_1(k) = M3 \text{ e } u(k) = M2 \\ \text{ENTÃO } u(k+1) = k_{i2}\xi - k_2x_1; \end{array}$$

$$\text{Regra 3: } \begin{array}{l} \text{SE } x_1(k) = M4 \text{ e } u(k) = M1 \\ \text{ENTÃO } u(k+1) = k_{i3}\xi - k_3x_1; \end{array}$$

$$\text{Regra 4: } \begin{array}{l} \text{SE } x_1(k) = M4 \text{ e } u(k) = M2 \\ \text{ENTÃO } u(k+1) = k_{i4}\xi - k_4x_1. \end{array}$$

Os ganhos em questão foram obtidos anteriormente via técnicas de alocação de pólo e métodos de controle ótimo.

Modelo fuzzy TS da Planta

A Figura 44 mostra como o valor da variável de estado x_1 de saída é obtida.

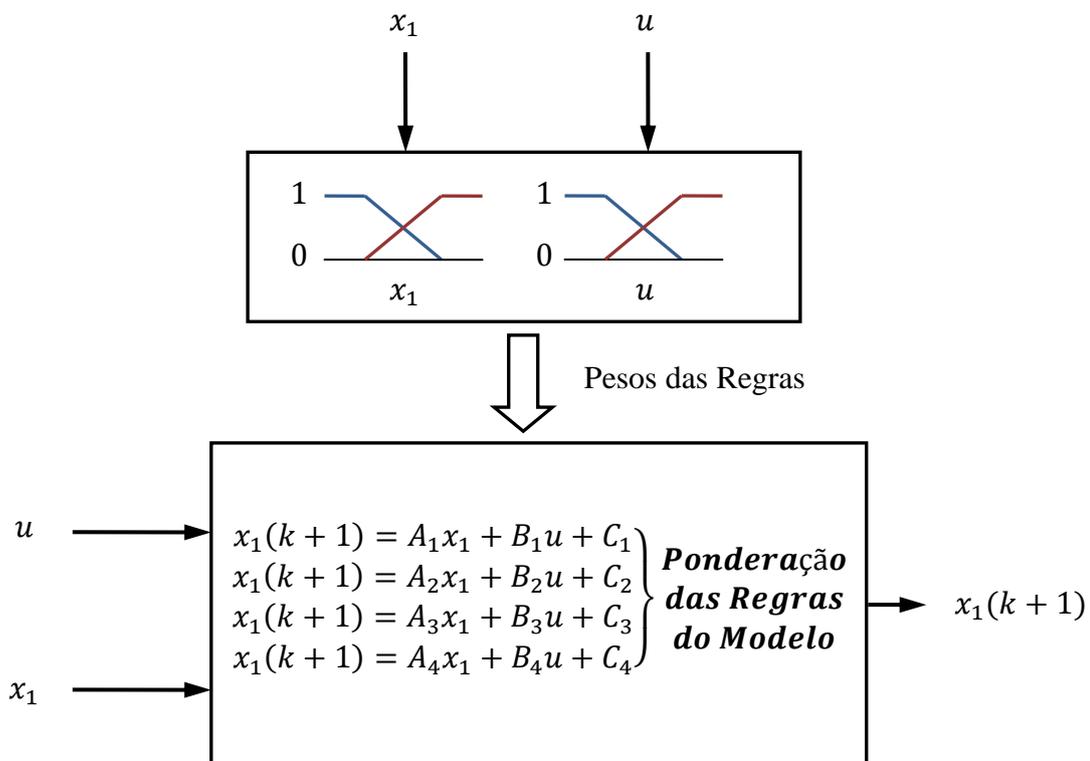


Figura 44 - Modelo fuzzy TS da planta de nível.

O sinal de controle u e a retroação do estado x_1 são entradas do modelo fuzzy TS da planta. Estes dados são processados através das regras do modelo, cujos conseqüentes são

equações lineares. Cada $x_1(k + 1)$ resultante de cada regra é ponderado de acordo com os seus correspondentes pesos. Os pesos das regras são encontrados pela operação AND com os valores das funções de pertinências das premissas das regras. O resultado é um valor que representa a saída do modelo *fuzzy* da planta.

Observa-se que o termo C_i foi desprezado durante o procedimento de cálculo dos ganhos de retroação, e também o será na análise de estabilidade através de LMIs. Esta aproximação permitirá simplificar a solução dos problemas de estabilidade via LMI.

4.11 Análise de Estabilidade

Neste trabalho, a solução numérica das LMIs foi realizada com o auxílio do software MATLAB. Para resolver LMIs é necessária a instalação de pacotes específicos (*toolboxes*), conhecidos na literatura como “*solvers*” de LMIs. O solver padrão do MATLAB é o “LMI control toolbox” (Gahinet, et al., 1995).

4.11.1 Estabilidade para o Sistema de Controle via Alocação de Pólos

Usando o algoritmo de otimização LMI do MATLAB, foi testado o Teorema 3 considerando os ganhos obtidos por alocação de pólos, encontra-se:

$$\mathbf{P} = \begin{bmatrix} 2,6036 & -1,0994 \\ -1,0994 & 6,0821 \end{bmatrix}.$$

Segundo o critério de Sylvester (Ogata, 2003), para que uma matriz \mathbf{P} seja definida positiva, é necessário que todos os determinantes menores principais sucessivos de \mathbf{P} sejam positivos.

Como $P_{11} = 2,6036 > 0$ e $|\mathbf{P}| = 14,6266 > 0$, logo a matriz comum \mathbf{P} é definida positiva, satisfazendo as condições de estabilidade do Teorema 4. Portanto, o sistema controlado é assintoticamente estável para os ganhos de retroação calculados por alocação de pólos.

4.11.2 Estabilidade para Sistema de Controle via Sistema de Controle Ótimo

Agora, usando o algoritmo de otimização LMI do MATLAB para testar o Teorema 3, considerando os ganhos obtidos por técnica de controle ótimo, encontra-se:

$$\mathbf{P} = \begin{bmatrix} 0,1975 & -0,1878 \\ -0,1878 & 3,0669 \end{bmatrix}.$$

Similar a análise anterior, como $P_{11} = 0,1975 > 0$ e $|\mathbf{P}| = 0,5704 > 0$, logo a matriz comum \mathbf{P} é definida positiva, satisfazendo as condições de estabilidade do Teorema 4. Portanto, o sistema controlado é também assintoticamente estável para os ganhos de retroação calculados por controle ótimo.

4.11.3 Questões sobre Instabilidade

Um rápido teste pode ser feito para conferir se o procedimento de teste de estabilidade apresentado é realmente fidedigno. Um exemplo pode ser ilustrado ao se projetar um controlador impondo pólos instáveis, como [2 2]. Estes pólos estão fora do círculo unitário do plano z para o sistema discreto assegurando a instabilidade (Phillips, C. L.; Nagle, H. T., 1995). Isto resulta nos seguintes ganhos de retroação:

$$\hat{\mathbf{K}}_1 = [k_1 \quad -k_{i1}] = [-56,2738 \quad 746,7213],$$

$$\hat{\mathbf{K}}_2 = [k_2 \quad -k_{i2}] = [-13,4542 \quad 178,4803],$$

$$\hat{\mathbf{K}}_3 = [k_3 \quad -k_{i3}] = [-59,0215 \quad 770,8416],$$

$$\hat{\mathbf{K}}_4 = [k_4 \quad -k_{i4}] = [-14,4999 \quad 189,3496].$$

Considerando os ganhos obtidos por alocação de pólos instáveis, encontra-se:

$$\mathbf{P} = \begin{bmatrix} -0,0087 & -0,0075 \\ -0,0075 & -5,5376 \end{bmatrix}.$$

Pelo critério de Sylvester, nota-se que os determinantes menores principais sucessivos de \mathbf{P} são: $P_{11} = -0,0087 < 0$; $|\mathbf{P}| = 0,0481 > 0$. Como existe um resultado negativo, a matriz \mathbf{P} não é positiva definida. Logo, as condições de estabilidade do Teorema 4 não são satisfeitas. Portanto, o sistema controlado não é assintoticamente estável para os ganhos de retroação adotados.

4.12 Programas e Softwares

Para a realização das simulações e dos ensaios deste trabalho, foram desenvolvidos dois programas principais:

- Em MATLAB foi elaborado um programa de simulação abordando toda a metodologia teórica aplicada à planta de nível. A Figura 45 apresenta o diagrama de classes deste programa.

- Em LABVIEW foi implementado o algoritmo de controle *fuzzy* TS para a planta de nível. Este programa se comunica com o programa de simulação através do arquivo *param.lvm*, possibilitando transferir todos os dados calculados necessários para o sistema de controle *fuzzy*, como os ganhos de retroação e os parâmetros das funções de pertinência utilizadas. Nos anexos no final da dissertação, encontram-se os programas desenvolvidos para este trabalho.

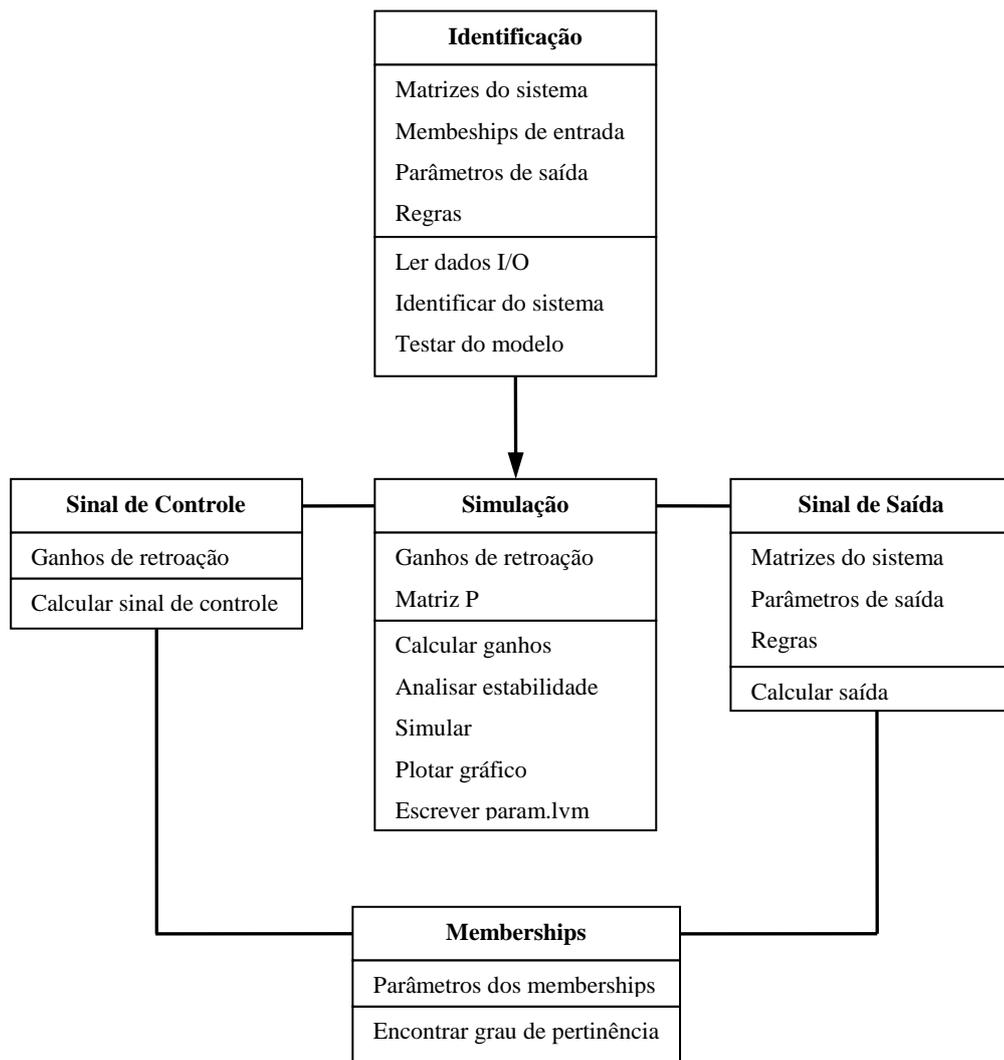


Figura 45 - Diagrama de classes do programa de simulação.

5. Resultados

Até aqui foram realizadas as seguintes etapas:

- O modelo *fuzzy* TS foi obtido representando matematicamente a planta de nível;
- Com o modelo pronto, aplicou-se o conceito de compensação paralela distribuída (PDC) para o projeto do sistema de controle *fuzzy*. Duas técnicas foram utilizadas para obtenção dos ganhos de retroação de estados: alocação de pólos e controle ótimo;

- Usando técnicas LMI, analisou-se a estabilidade da malha de controle para os dois conjuntos de ganhos, concluindo sobre condições de estabilidade assintótica.

Portanto, todas as informações necessárias para a simulação estão prontas, assim como para a aplicação em ensaios práticos da planta reduzida. Logo, a validação de toda metodologia será comprovada através dos gráficos gerados pelas simulações e ensaios, como se segue.

5.1 Controlador Fuzzy via Alocação de Pólos

Observa-se na Figura 46 que a resposta do modelo obtido e a resposta real da planta são bem similares, indicando uma boa modelagem. A resposta da malha de controle é estável e com a dinâmica esperada.

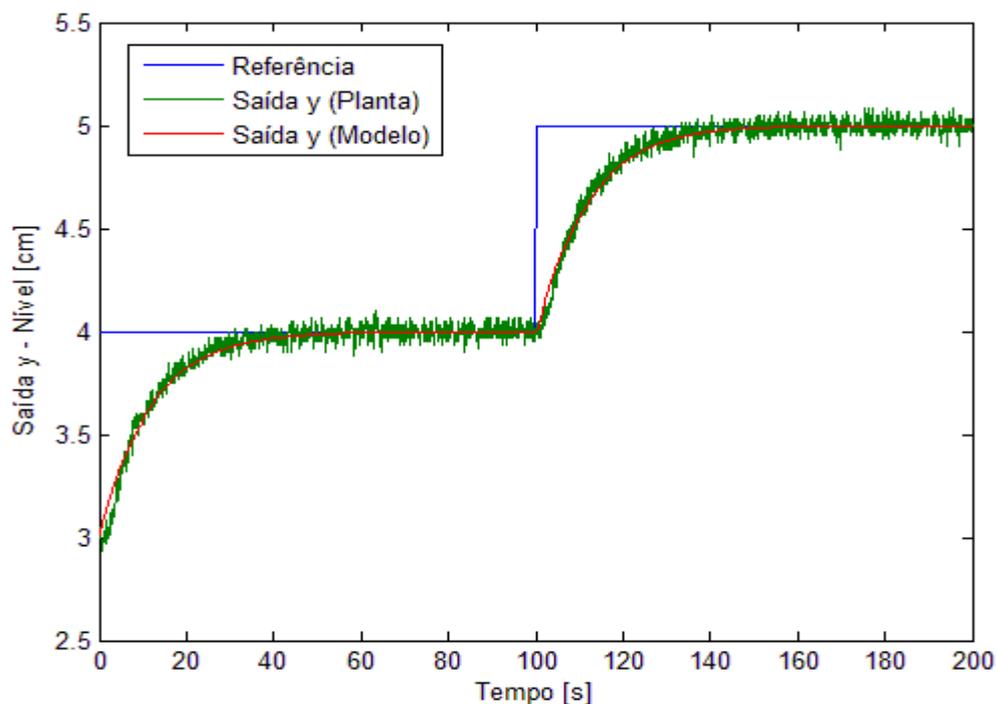


Figura 46 - Resposta real da malha de controle fuzzy para dois valores de referência de entrada.

A Figura 47 mostra a resposta do sistema para valores de referência mais altos. As oscilações observadas na informação de comando do sistema caracterizam os chaveamentos dos ganhos correspondentes a lei de controle *fuzzy* utilizada.

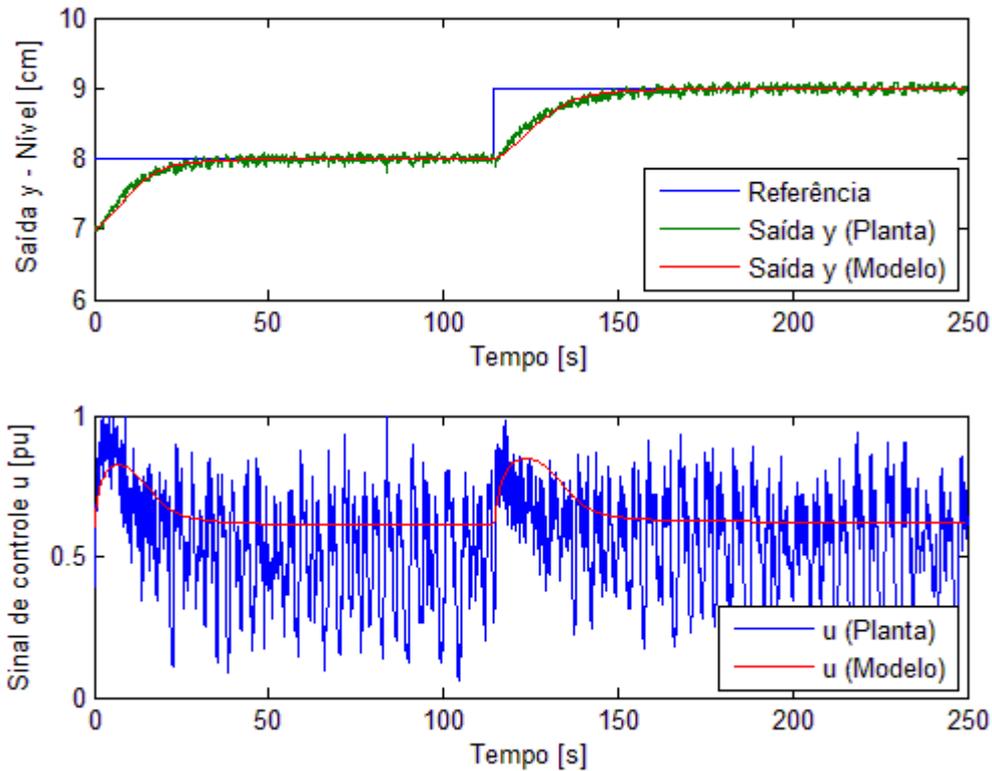


Figura 47 - Resposta real do sistema do sistema de controle fuzzy com outros valores de referência.

A Figura 48 mostra a resposta da malha de controle *fuzzy* para uma sequência de valores de referência de entrada, abrangendo uma faixa de valores considerável de saída. Isso ilustra a característica do controlador *fuzzy* Takagi-Sugeno de poder trabalhar em um vasto espaço de estados, não se restringindo a um ponto de operação.

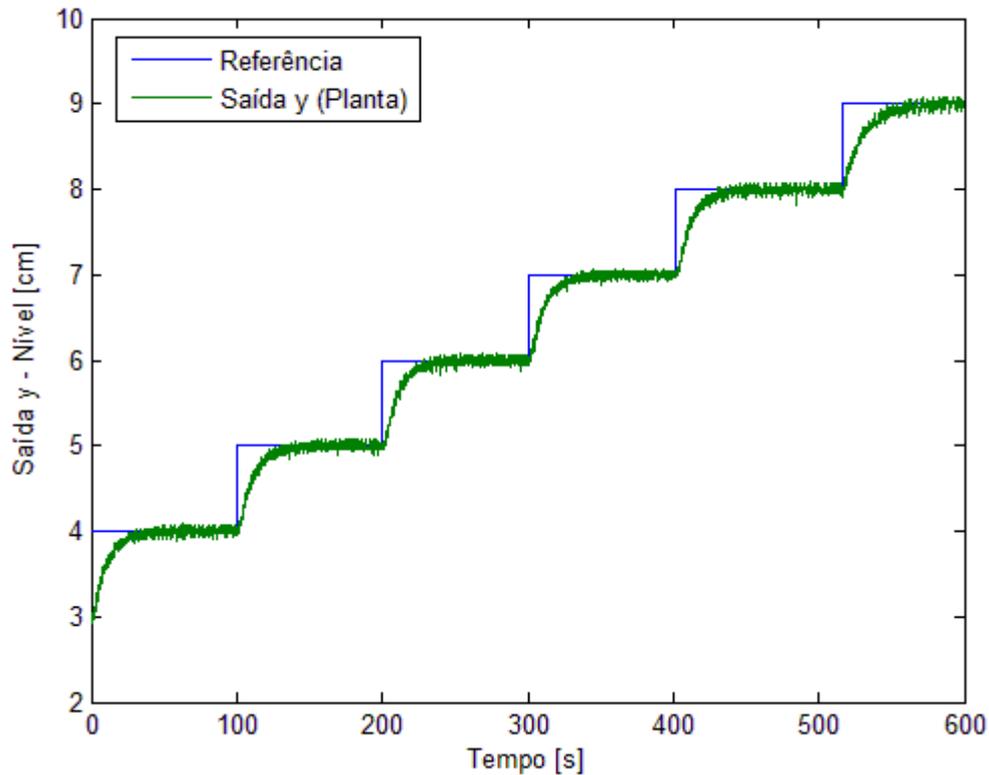


Figura 48 – Respostas reais do sistema de controle fuzzy para uma seqüência de valores de entrada.

Outro resultado interessante a ser destacado é a possibilidade de se utilizar a média dos valores dos ganhos de retroação *fuzzy* para realizar uma malha de controle convencional. A Figura 49 mostra a resposta real da malha de controle convencional com ganhos fixos dados pela média dos ganhos do controlador *fuzzy* utilizado. Observa-se que as respostas do sistema *fuzzy* e da malha convencional estão bem próximas para a faixa de referência de entrada considerada. Mas esta característica não se mantém para toda a faixa de operação possível quando se considera a malha de controle convencional, uma vez que o processo controlado apresenta um modelo não-linear.

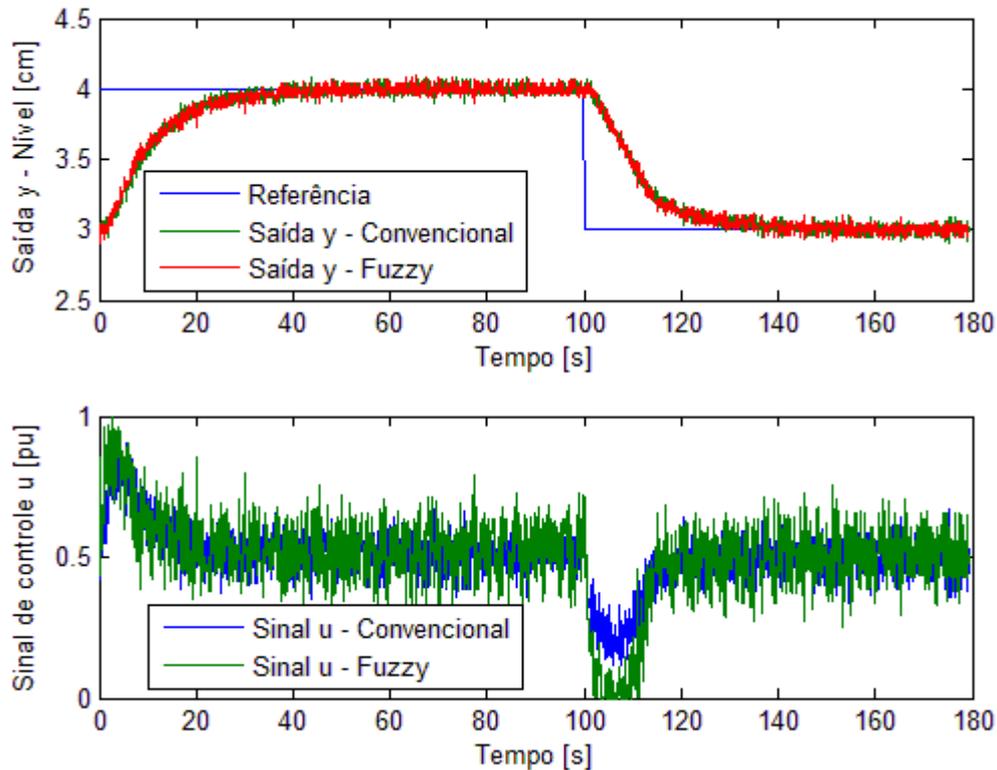


Figura 49 – Ensaio para o controlador fuzzy e uma malha de controle convencional.

5.2 Controlador Fuzzy via Sistema de Controle Ótimo

Simulações e ensaios foram realizados semelhantemente para um sistema de controle ótimo, a fim de se obter resultados comparativos ao sistema de controle via alocação de pólos.

A Figura 50 mostra respostas do sistema para dois valores de referência de entrada. Observa-se que a resposta obtida tem uma coincidência de valores maior que a resposta da Figura 46.

No caso de valores mais altos de saída, aparece uma pequena diferença nos resultados entre a simulação e ensaio real (Figura 51) que se deve a eventuais diferenças entre o sistema real e sua modelagem resultante.

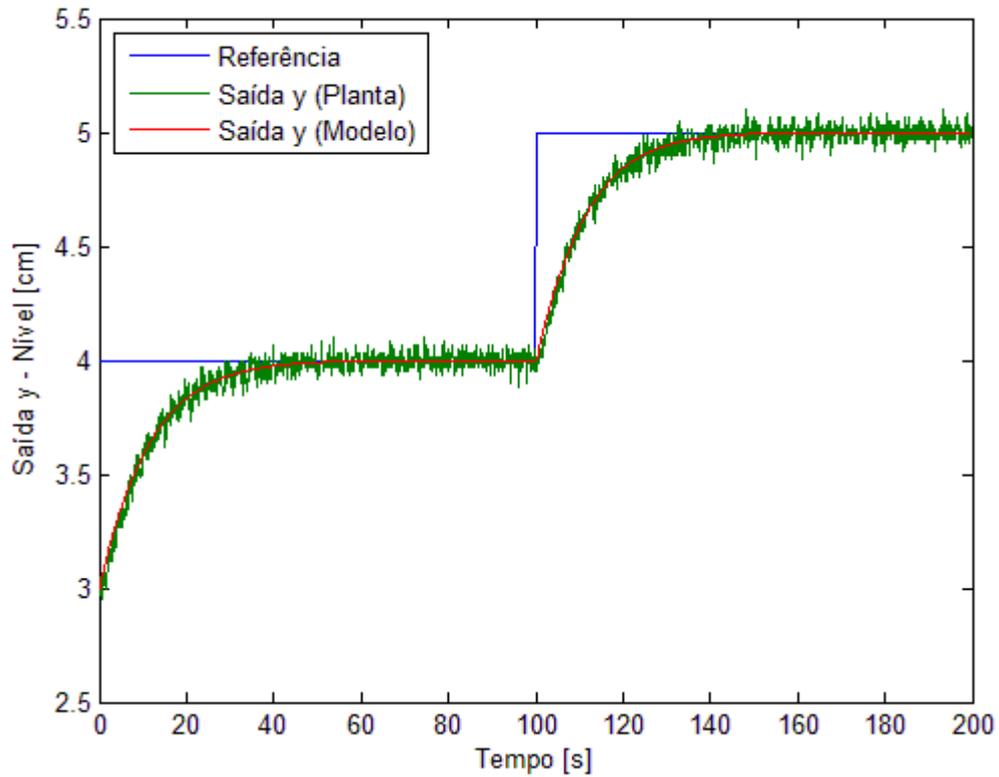


Figura 50 - Resposta do sistema para dois valores baixos de referência de entrada.

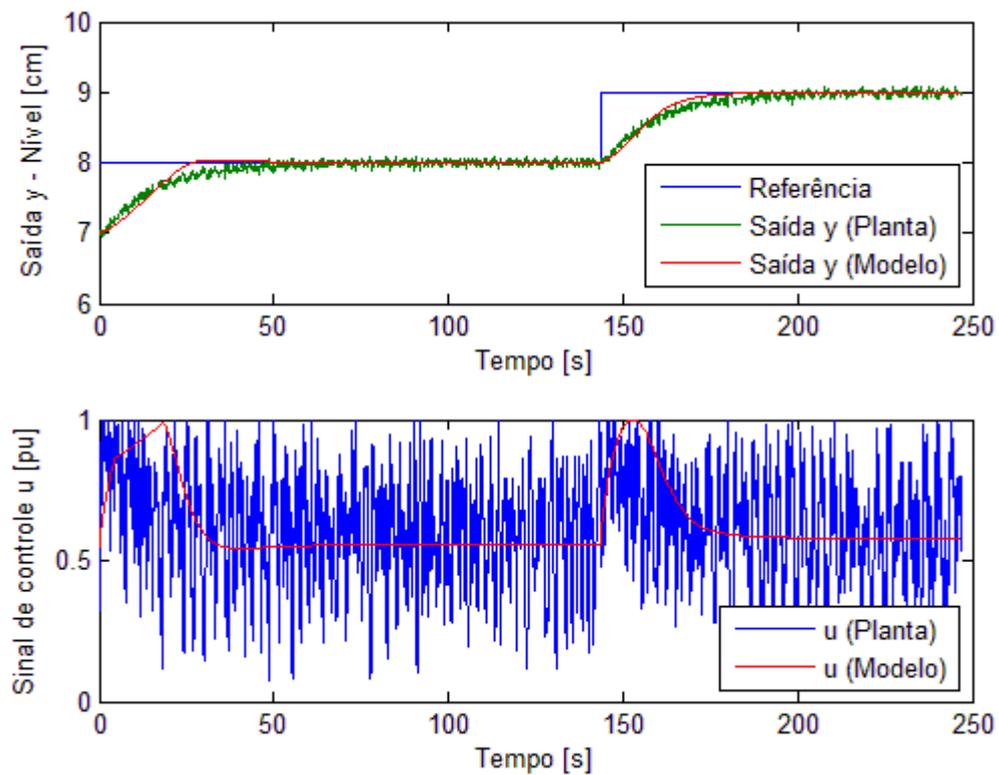


Figura 51 - Resposta do sistema para dois valores de referência de entrada mais altos.

A Figura 52 mostra a resposta do sistema quando também submetido a uma seqüência de valores de entrada, abrangendo uma faixa de valores considerável de saída. Observa-se que o controlador *fuzzy* de modelo Takagi-Sugeno permite que a malha de controle resultante trabalhe em um vasto espaço de estados.

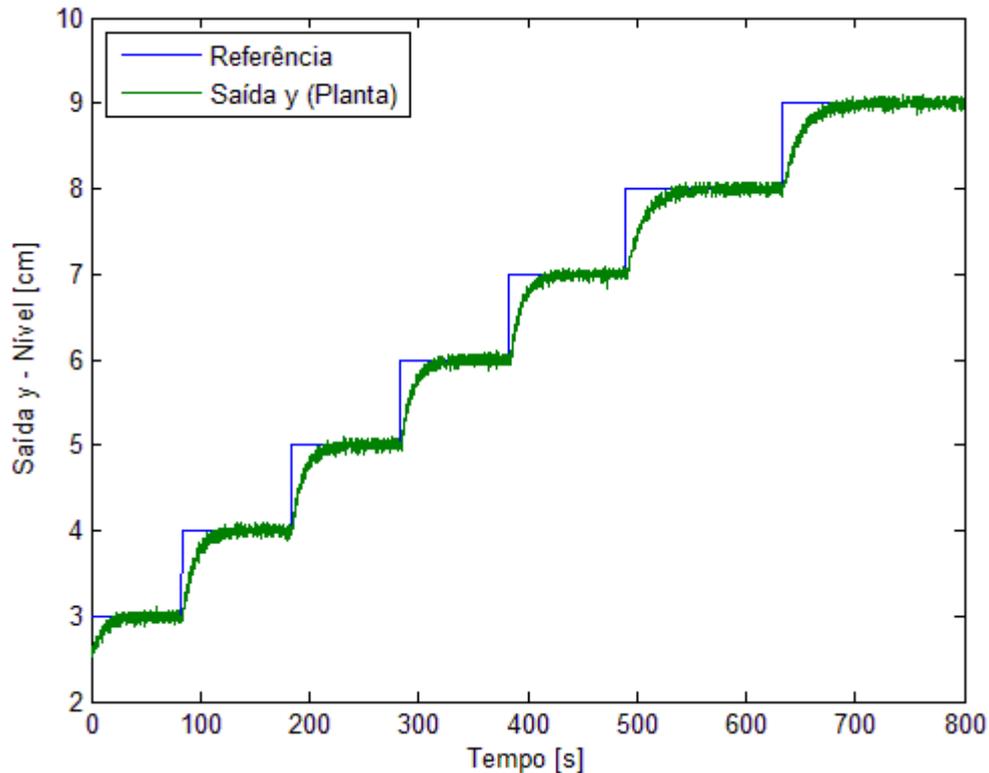


Figura 52 - Respostas do sistema de controle para uma seqüência de valores de referência de entrada.

Por fim, também foi realizado o ensaio com a média dos ganhos *fuzzy* para um controlador convencional por retroação de estados. A Figura 53 ilustra a resposta em questão. Observa-se que a informação de controle da malha convencional possui oscilações menores quando comparada com a malha *fuzzy*. Isto mostra a limitação do controle convencional na compensação de um sistema não-linear devido os ganhos fixos. Esse mesmo resultado é observado na Figura 49.

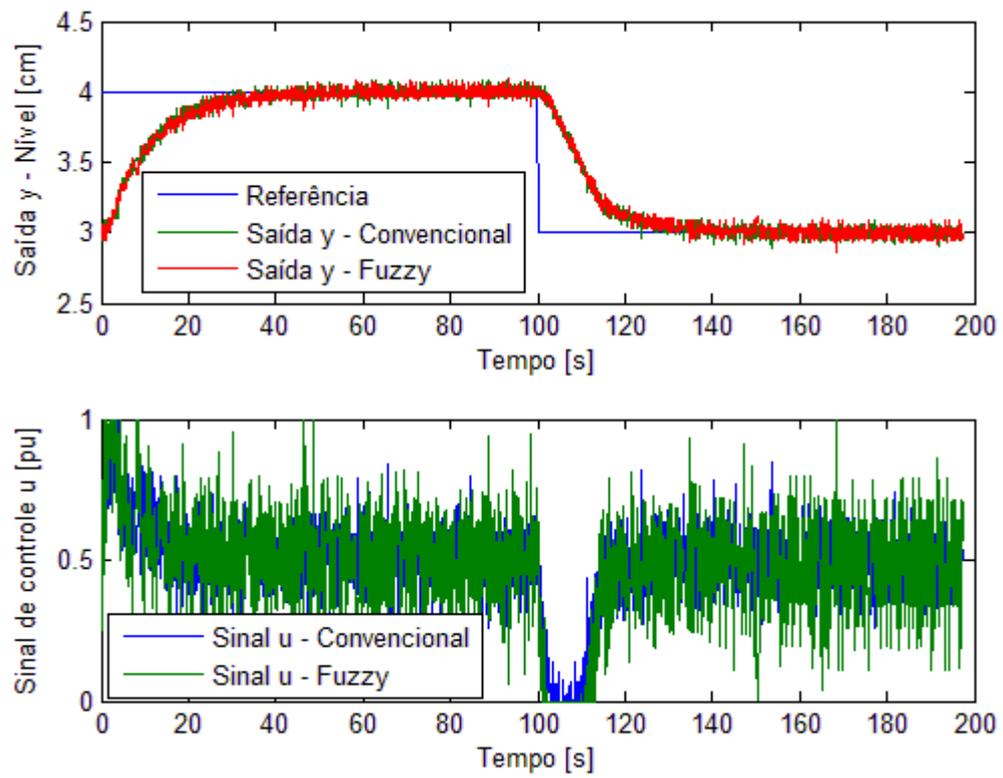


Figura 53 - Ensaio para o controlador fuzzy e para uma malha de controle convencional.

6. Conclusão

Os procedimentos mostrados neste trabalho se mostraram eficientes em análises de estabilidade e no projeto de sistemas de controle *fuzzy*.

Os modelos *fuzzy* Takagi-Sugeno permitem representar processos não-lineares, onde as dinâmicas locais, em diferentes regiões do espaço de estado, são modelada por modelos lineares.

A compensação paralela distribuída (PDC) se mostrou promissora para aplicações reais, permitindo a utilização de técnicas convencionais de projeto de sistemas de controle na síntese de regras de controladores *fuzzy*. Uma das motivações deste trabalho foi utilizar técnicas de retroação de estados para o projeto e realização prática de um sistema de controle *fuzzy* real.

Análises de estabilidade e técnicas de compensação podem ser realizados por métodos de desigualdades lineares matriciais, baseadas no critério de estabilidade quadrática de Lyapunov. Caso as condições de estabilidade não sejam satisfeitas, o controlador para cada regra é re-projetado. As LMIs são recursos poderosos para auxiliar no processo de projeto do controle *fuzzy*. Algoritmos computacionais de otimização convexa são usados para encontrar a solução de problemas de LMI.

O principal objetivo deste trabalho foi comprovar a eficiência da metodologia de projeto de controladores *fuzzy* TS através de um experimento prático. Ensaios foram realizados em uma planta de escala reduzida de um sistema de nível. A abordagem deste trabalho se iniciou através de simulações computacionais e por fim se concretizou através de uma implementação computacional em tempo-real.

Esta abordagem será considerada em trabalhos futuros visando á compensação de sistemas não-lineares com múltiplas entradas e múltiplas saídas. A implantação de malhas de controle *fuzzy* em sistemas embarcados em condições de tempo-real também será pesquisada.

7. Referencias Bibliográficas

Aguirre, L. A. 2000. *Introdução à Identificação de Sistemas*. Belo Horizonte : UFMG, 2000. ISBN 85-7041-220-7.

Boyd, S.; El Ghaoui, L.; Feron, E.; Balakrishnan, V. 1994. *Linear Matrix Inequalities in Systems and Control Theory*. Philadelphia : SIAM books, 1994.

Chen, B. S., Tseng, C. S. e Uang, H. J. 2000. Mixed $H_2=H_\infty$ fuzzy output feedback control design for nonlinear dynamic systems: An LMI approach. *IEEE Trans. Fuzzy Syst.* 2000, Vol. 8, pp. 249-265.

Chen, B. S., Tseng, C. S. e Uang, H. J. 1999. Robustness design of nonlinear dynamic systems via fuzzy linear control. *IEEE Trans. Fuzzy Syst.* 1999, Vol. 7, 5, pp. 571–585.

Fang, C. H.; Liu, Y. S.; Kau, S. W.; Hong, L.; Lee, C. H. 2006. A New LMI-Based Approach to Relaxed Quadratic Stabilization of T–S Fuzzy Control Systems. *IEEE Trans. Fuzzy Systems*. 2006, Vol. 14, 3.

Faria, F. A.; 2009. *Projetos de Controladores Baseados em LMI usando Realimentação da Derivada dos Estados*. UNESP. 2009. Tese de Mestrado.

Filev, D. 1991. Fuzzy modeling of complex systems. *International Journal Approximate Reasoning*. 1991, Vol. 5, pp. 281-290.

Gahinet, P.; Nemirovski, A.; Laub, A. J. e; Chilali, M. 1995. LMI control toolbox - For use with MATLAB. [Online] 1995. [Citado em: 14 de Outubro de 2011.] http://www.mathworks.com/access/helpdesk_r13/help/pdf_doc/lmi/lmi.pdf.

Joh, J., Chen, Y. H. e Langari, R. 1998. On the stability issues of linear Takagi–Sugeno fuzzy models. *IEEE Trans. Fuzzy Syst.* Agosto de 1998, Vol. 6, 3, pp. 402–410.

Johansen, T. A., Shorten, R. e Smith, R. M. 1998. On the interpretation and identification of dynamic Takagi-Sugeno fuzzy models. *IEEE Transactions of Fuzzy and Systems*. 1998, Vol. 8, 3, pp. 297-313.

Kim, E.; Lee, H. 2000. New approaches to relaxed quadratic stability condition of fuzzy control systems. *IEEE Trans. Fuzzy Syst.* Outubro de 2000, Vol. 8, 5, pp. 523–533.

Kuo, B. C. 1980. *Digital Control Systems*. s.l. : Holt-Saunders, 1980.

Lian, K. Y., Tu, H. W. e Liou, J. J. 2006. Stability Conditions for LMI-Based Fuzzy Control From Viewpoint of Membership Functions. *IEEE Trans. Fuzzy Systems*. Dezembro de 2006, Vol. 14, 6.

- Machado, E. R. M. D. 2003.** *Modelagem e Controle de Sistemas Fuzzy Takagi-Sugeno*. UNESP. 2003. Tese de Doutorado.
- Martins de Carvalho, J. L. 2000.** *Sistema de Controle Automático*. s.l. : LTC, 2000.
- MathWorks. 2011.** R2011b Documentation. [Online] 2011. [Citado em: 29 de Outubro de 2011.] <http://www.mathworks.com/help/toolbox/fuzzy/fp49243.html>.
- Nesterov, Y.; Nemirovsky, A.; 1994.** *Interior-point polynomial methods in convex programming*. Philadelphia : SIAM, 1994.
- Ogata, K. 2003.** *Engenharia de Controle Moderno*. São Paulo : Prentice Hall, 2003.
- Phillips, C. L.; Nagle, H. T.; 1995.** *Digital Control System Analysis and Design*. Englewood Cliffs, New Jersey : Prentice Hall International, 1995.
- Souza, A. C. Z.; Pinheiro, C. A. M.; 2008.** *Introdução à Modelagem, Análise e Simulação de Sistemas Dinâmicos*. Rio de Janeiro : Interciência, 2008.
- Sugeno, M.; Kang, G. T. 1986.** Fuzzy modeling and control multilayer incinerator. *Fuzzy Sets and Systems*. 1986, 18, pp. 329-346.
- Sugeno, M.; Kang, G. T. 1988.** Structure identification of fuzzy model. *Fuzzy Sets and Systems*. 1988, 28, pp. 15-33.
- Sugeno, M.; Tanaka, K. 1991.** Successive identification of a fuzzy model and its applications to prediction of a complex system. *Fuzzy Sets and Systems*. 1991, 42, pp. 315-334.
- Takagi, T.; Sugeno, M. 1985.** Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Janeiro de 1985, Vols. SMC-15, 1, pp. 116–132*.
- Tanaka, K., Ikeda, T. e Wang, H.O. 1996.** Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: quadratic stabilizability, H_∞ control theory and linear matrix inequalities. *IEEE Trans. Fuzzy Systems*. 1996, Vol. 4, 1, pp. 1–13.
- Tanaka, K.; Sugeno, M. 1990.** Stability analysis of fuzzy systems using Lyapunov's direct method. *Proc. NAFIPS'90*. Junho de 1990, pp. 133–136.
- Tanaka, K.; Sugeno, M.; 1992.** Stability analysis and design of fuzzy control systems. *Fuzzy Sets Syst*. 1992, Vol. 45, 2, pp. 135–156.
- Tanaka, K.; Wang, H.O. 2001.** *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*. New York : Wiley, 2001.
- Taniguchi, T.; Tanaka, K.; Ohatake, H.; Wan, H. O. 2001.** Model construction, rule reduction, and robust compensation for generalized form of Takagi-Sugeno fuzzy systems. *IEEE Transactions on Fuzzy Systems*. 2001, Vol. 9, 4, pp. 525–537.

Wang, H. O., Tanaka, K. e Griffin, M. 1996. An approach to fuzzy control of nonlinear systems: Stability and design issues. *IEEE Trans. Fuzzy Syst.* 1996, Vol. 4, pp. 14–23.

Wang, H. O., Tanaka, K. e Griffin, M. F. 1995. Parallel distributed compensation of nonlinear systems by Takagi-Sugeno fuzzy model. *Proc. FUZZY-IEEE/IFES.* 1995, Vol. 95, pp. 531–538.

Wang, L. 1997. *A course in fuzzy systems and control.* Upper Saddle River, USA : Prentice Hall, 1997.

ANEXO 1 – Programa em MATLAB para simulação (programa principal)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% - Simulacao - %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%-----Calculando Ganhos-----

T = 0.1; %tempo de amostragem [s]
[K1, Ki, Ah, Bh, F] = func_Calc_Ganhos_AP(A, B, T); %calcula ganhos

%-----Parametros de entrada para o LabView-----

ParmGanhos = [Ki(1) K1(1) Ki(2) K1(2) Ki(3) K1(3) Ki(4) K1(4)];
ParmMF = [pts(1,1) pts(1,2) pts(2,1) pts(2,2) pts(3,1) pts(3,2) ...
          pts(4,1) pts(4,2)];

dlmwrite('ParmGanhos.txt', ParmGanhos, '\t');
dlmwrite('ParmMF.txt', ParmMF, '\t');

%-----Simulação para uma dada referência-----

%tamanho dos vetores (tempo de simulação)
N = 2000;

%parametros iniciais para o sistema em regime em 3 cm
ref = 3; x1 = 3; Ierro = 35.5739; u = 1.1535; t = 0;

for n = 1 : N

    ref = 4; %referencia de entrada
    erro = ref - x1; %erro
    Ierro = Ierro + erro*T; %integral do erro
    u = func_Sinal_Controlo(pts, K1, Ki, Ierro, u, x1); %sinal de controle
    x1 = func_Resposta_Fuzzy(pts, A, B, C, u, x1); %resposta fuzzy

    %Vetores para plotagem
    VSt(n) = t;
    VSy(n) = x1;
    VSu(n) = u;
    VSref(n) = ref;

    t = t + T; %incremento de tempo

end

figure, plot(VSt, VSref, VSt, VSu, VSt, VSy)
title('Simulacao do Modelo Fuzzy');
legend('Referência', 'Sinal de Controle u', 'Saída y (Modelo)');
xlabel('Tempo [s]'); ylabel('Saída y - Nivel [cm]');

```

ANEXO 2 – Sub-rotinas para cálculo dos ganhos de retroação

```

function [K1, Ki, Ah, Bh, F] = func_Calc_Ganhos_AP(A, B, T)

%-----Matrizes do Sistema obtido do Modelo Fuzzy-----

A1 = A(1); A2 = A(2); A3 = A(3); A4 = A(4);
B1 = B(1); B2 = B(2); B3 = B(3); B4 = B(4);
c = 1; %y = c*X

%-----Obtendo ganhos por Alocação de Pólos: J = [-0.1 -0.1]-----
J = [-0.1 -0.1];

A1h = [A1 0; -T*c 0]; B1h = [B1; 0]; K = acker(A1h,B1h,J);
K1(1) = K(1); Ki(1) = -K(2);
A2h = [A2 0; -T*c 0]; B2h = [B2; 0]; K = acker(A2h,B2h,J);
K1(2) = K(1); Ki(2) = -K(2);
A3h = [A3 0; -T*c 0]; B3h = [B3; 0]; K = acker(A3h,B3h,J);
K1(3) = K(1); Ki(3) = -K(2);
A4h = [A4 0; -T*c 0]; B4h = [B4; 0]; K = acker(A4h,B4h,J);
K1(4) = K(1); Ki(4) = -K(2);

%Para verificação de estabilidade por LMI
Ah = [A1h A2h A3h A4h];
Bh = [B1h B2h B3h B4h];
F = [K1(1) Ki(1) K1(2) Ki(2) K1(3) Ki(3) K1(4) Ki(4)];

function [K1, Ki, Ah, Bh, F] = func_Calc_Ganhos_CO(A, B, T)

%-----Matrizes do Sistema obtido do Modelo Fuzzy-----

A1 = A(1); A2 = A(2); A3 = A(3); A4 = A(4);
B1 = B(1); B2 = B(2); B3 = B(3); B4 = B(4);
c = 1; %y = c*X

%-----Obtendo ganhos ótimos por Riccati-----
A1h = [A1 0; -T*c 0]; B1h = [B1; 0]; [X,L,K] = care(A1h,B1h,[10 0; 0 10]);
K1(1) = K(1); Ki(1) = -K(2);
A2h = [A2 0; -T*c 0]; B2h = [B2; 0]; [X,L,K] = care(A2h,B2h,[10 0; 0 1]);
K1(2) = K(1); Ki(2) = -K(2);
A3h = [A3 0; -T*c 0]; B3h = [B3; 0]; [X,L,K] = care(A3h,B3h,[10 0; 0 10]);
K1(3) = K(1); Ki(3) = -K(2);
A4h = [A4 0; -T*c 0]; B4h = [B4; 0]; [X,L,K] = care(A4h,B4h,[10 0; 0 1]);
K1(4) = K(1); Ki(4) = -K(2);

%Para verificação de estabilidade por LMI
Ah = [A1h A2h A3h A4h];
Bh = [B1h B2h B3h B4h];
F = [K1(1) Ki(1) K1(2) Ki(2) K1(3) Ki(3) K1(4) Ki(4)];

```

ANEXO 3 – Sub-rotina para análise de estabilidade

```

function [Pout] = func_LMI(Ah, Bh, F)

%-----Matrizes do Sistema-----
A1 = [Ah(1,1) Ah(1,2); Ah(2,1) Ah(2,2)];
A2 = [Ah(1,3) Ah(1,4); Ah(2,3) Ah(2,4)];
A3 = [Ah(1,5) Ah(1,6); Ah(2,5) Ah(2,6)];
A4 = [Ah(1,7) Ah(1,8); Ah(2,7) Ah(2,8)];
B1 = [Bh(1,1); Bh(2,1)]; B2 = [Bh(1,2); Bh(2,2)];
B3 = [Bh(1,3); Bh(2,3)]; B4 = [Bh(1,4); Bh(2,4)];

%-----Ganhos do Controle-----
F1 = [F(1,1) -F(1,2)]; F2 = [F(1,3) -F(1,4)];
F3 = [F(1,5) -F(1,6)]; F4 = [F(1,7) -F(1,8)];

%-----Estabilidade por Hurwitz-----
G12 = ((A1-B1*F2)+(A2-B2*F1))/2; G13 = ((A1-B1*F3)+(A3-B3*F1))/2;
G14 = ((A1-B1*F4)+(A4-B4*F1))/2; G23 = ((A2-B2*F3)+(A3-B3*F2))/2;
G24 = ((A2-B2*F4)+(A4-B4*F2))/2; G34 = ((A3-B3*F4)+(A4-B4*F3))/2;

%-----Declaração de variáveis LMI-----
setlmis([]);
P=lmivar(1,[2 1]);

%-----Equacionamento-----
lmiterm([1 1 1 P],(A1-B1*F1)',1,'s');% LMI #1:(A1-B1*F1)'*P+P*(A1-B1*F1)
lmiterm([2 1 1 P],(A2-B2*F2)',1,'s');% LMI #2:(A2-B2*F2)'*P+P*(A2-B2*F2)
lmiterm([3 1 1 P],(A3-B3*F3)',1,'s');% LMI #3:(A3-B3*F3)'*P+P*(A3-B3*F3)
lmiterm([4 1 1 P],(A4-B4*F4)',1,'s');% LMI #4:(A4-B4*F4)'*P+P*(A4-B4*F4)
lmiterm([5 1 1 P],G12',1,'s');% LMI #5:G12'*P+P*G12
lmiterm([6 1 1 P],G13',1,'s');% LMI #6:G13'*P+P*G13
lmiterm([7 1 1 P],G14',1,'s');% LMI #7:G14'*P+P*G14
lmiterm([8 1 1 P],G23',1,'s');% LMI #8:G23'*P+P*G23
lmiterm([9 1 1 P],G24',1,'s');% LMI #9:G24'*P+P*G24
lmiterm([10 1 1 P],G34',1,'s');% LMI #10:G34'*P+P*G34

Resp_LMI=getlmis;

%-----Resultado-----
[tmin,xfeas] = feasp(Resp_LMI);
Pout = dec2mat(Resp_LMI,xfeas,P);

%Critério de Sylvester: todos determinantes menores
%principais sucessivos de P devem ser positivos (Ogata)
if(Pout(1,1) > 0)
    if(det(Pout)>0)
        disp('O Sistema Controlado é Assintoticamente Estável.')
    end
end
end

```

ANEXO 4 – Sub-rotina do controlador fuzzy

```

function [u_out] = func_Sinal_Controlo(pts, K1, Ki, Ie, u, x1)

%-----Ganhos do Controle-----
k11 = K1(1); k12 = K1(2); k13 = K1(3); k14 = K1(4);
ki1 = Ki(1); ki2 = Ki(2); ki3 = Ki(3); ki4 = Ki(4);

%-----Funções de Pertinência - pontos dos memberships-----
E1 = pts(1,1); E2 = pts(2,1); E3 = pts(3,1); E4 = pts(4,1);
D1 = pts(1,2); D2 = pts(2,2); D3 = pts(3,2); D4 = pts(4,2);

p11 = func_in1mf1(E1,D1,u); %in1mf1 (pertinência em M1)
p12 = func_in1mf2(E2,D2,u); %in1mf2 (pertinência em M2)
p21 = func_in2mf1(E3,D3,x1); %in2mf1 (pertinência em M3)
p22 = func_in2mf2(E4,D4,x1); %in2mf2 (pertinência em M4)

%-----Peso de cada regra - Método and (prod)para regras-----
r1 = p11 * p21; r2 = p11 * p22; r3 = p12 * p21; r4 = p12 * p22;

%-----Saída do Controlador-----
u_out1 = ki1*Ie-k11*x1;
if u_out1 > 5
    u_out1 = 5;
elseif u_out1 < 0
    u_out1 = 0;
end
u_out2 = ki2*Ie-k12*x1;
if u_out2 > 5
    u_out2 = 5;
elseif u_out2 < 0
    u_out2 = 0;
end
u_out3 = ki3*Ie-k13*x1;
if u_out3 > 5
    u_out3 = 5;
elseif u_out3 < 0
    u_out3 = 0;
end
u_out4 = ki4*Ie-k14*x1;
if u_out4 > 5
    u_out4 = 5;
elseif u_out4 < 0
    u_out4 = 0;
end
u_out = (r1*u_out1+r2*u_out2+r3*u_out3+r4*u_out4)/(r1+r2+r3+r4);

```

ANEXO 5 – Sub-rotina da planta fuzzy Takagi-Sugeno

```

function [x1_out] = func_Resposta_Fuzzy(pts, A, B, C, u, x1)

%-----Matrizes do Sistema-----
A1 = A(1); A2 = A(2); A3 = A(3); A4 = A(4);
B1 = B(1); B2 = B(2); B3 = B(3); B4 = B(4);
C1 = C(1); C2 = C(2); C3 = C(3); C4 = C(4);

%-----Funções de Pertinência - pontos dos memberships-----
E1 = pts(1,1); E2 = pts(2,1); E3 = pts(3,1); E4 = pts(4,1);
D1 = pts(1,2); D2 = pts(2,2); D3 = pts(3,2); D4 = pts(4,2);

p11 = func_in1mf1(E1,D1,u); %in1mf1 (pertinência em M1)
p12 = func_in1mf2(E2,D2,u); %in1mf2 (pertinência em M2)
p21 = func_in2mf1(E3,D3,x1); %in2mf1 (pertinência em M3)
p22 = func_in2mf2(E4,D4,x1); %in2mf2 (pertinência em M4)

%-----Peso de cada regra - Método and (prod)para regras-----
r1 = p11 * p21; r2 = p11 * p22; r3 = p12 * p21; r4 = p12 * p22;

%-----Saida do Modelo-----
x1_out1 = A1*x1+B1*u+C1;
x1_out2 = A2*x1+B2*u+C2;
x1_out3 = A3*x1+B3*u+C3;
x1_out4 = A4*x1+B4*u+C4;

x1_out = (r1*x1_out1+r2*x1_out2+r3*x1_out3+r4*x1_out4)/(r1+r2+r3+r4);

```

ANEXO 6 – Sub-rotinas para as funções de pertinência

```

function [w] = func_in1mf1(E, D, u)

a = 1/(E-D); b = -a*D;

if u < E
    w = 1;
elseif u >= E && u < D
    w = a*u + b;
elseif u >= D
    w = 0;
end

function [w] = func_in1mf2(E, D, u)

a = -1/(E-D); b = -a*E;

if u < E
    w = 0;
elseif u >= E && u < D
    w = a*u + b;
elseif u >= D
    w = 1;
end

function [w] = func_in2mf1(E, D, x1)

a = 1/(E-D); b = -a*D;

if x1 < E
    w = 1;
elseif x1 >= E && x1 < D
    w = a*x1 + b;
elseif x1 >= D
    w = 0;
end

function [w] = func_in2mf2(E, D, x1)

a = -1/(E-D); b = -a*E;

if x1 < E
    w = 0;
elseif x1 >= E && x1 < D
    w = a*x1 + b;
elseif x1 >= D
    w = 1;
end

```

ANEXO 7 – Programa em LABVIEW implementando lei de controle fuzzy

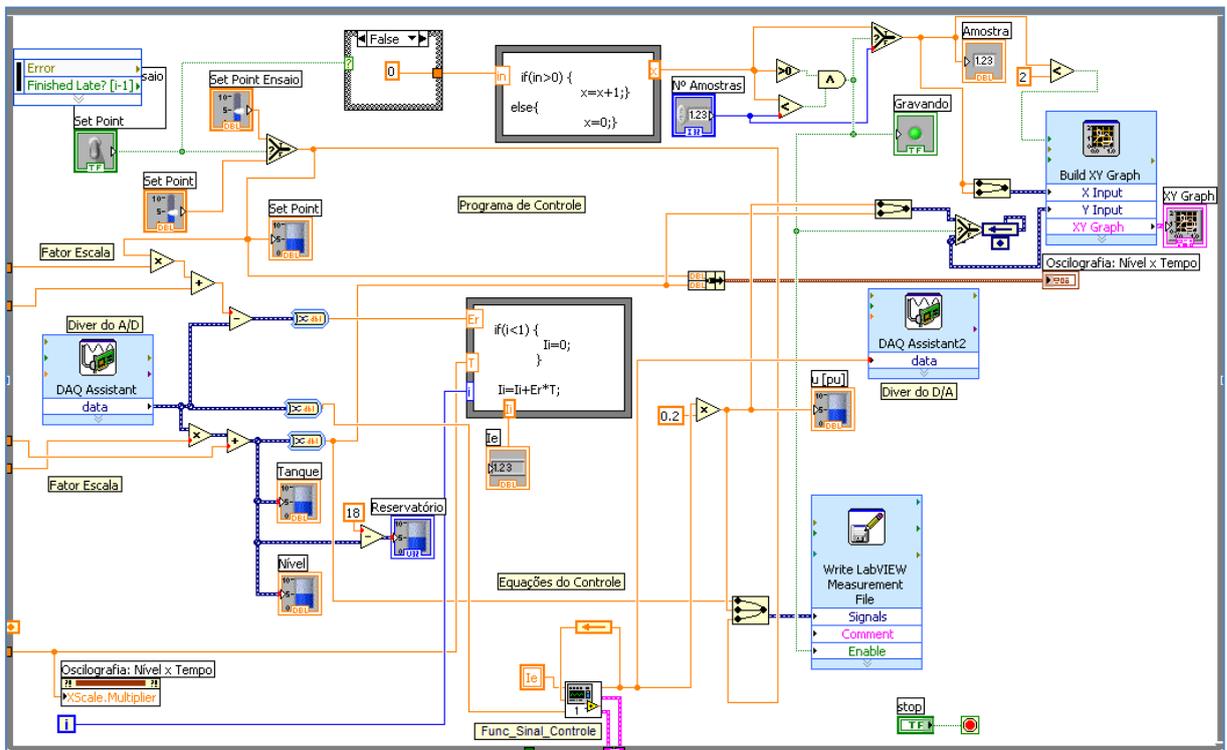


Figura 54 - Sistema de controle fuzzy (programa principal)

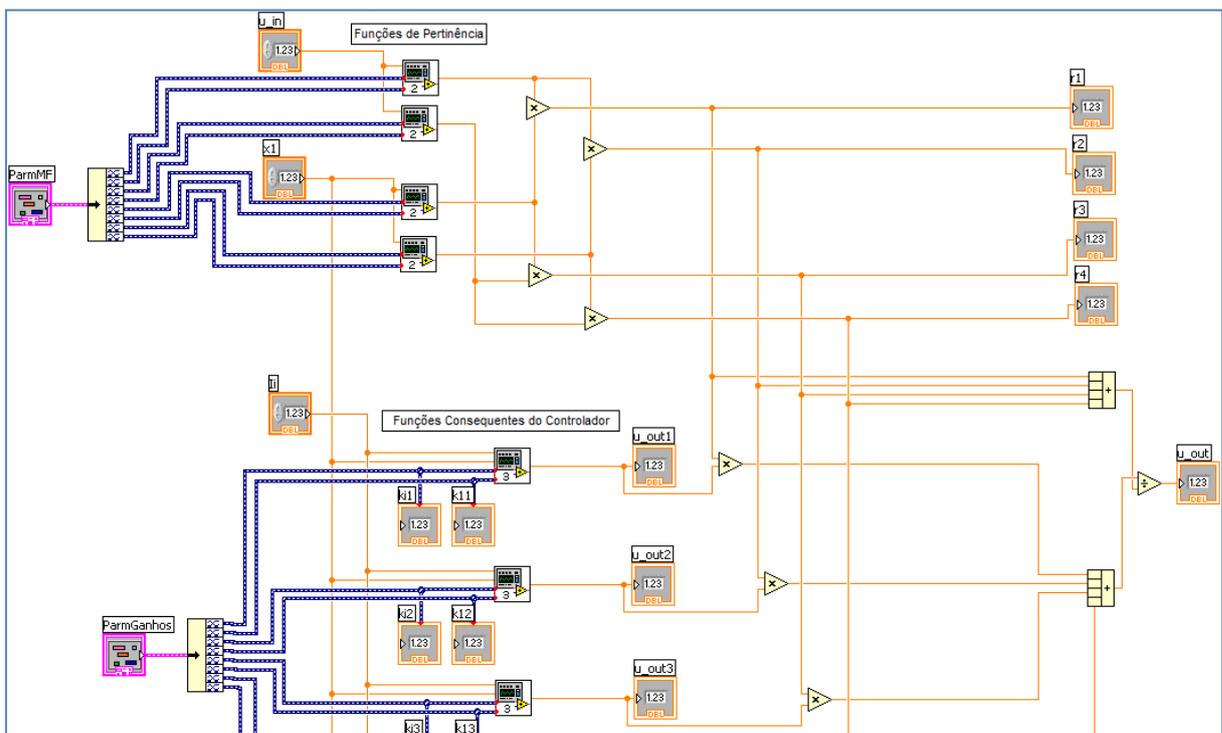


Figura 55 - Sub-rotina das regras de controle fuzzy (equivalente ao Anexo 4 em MATLAB)

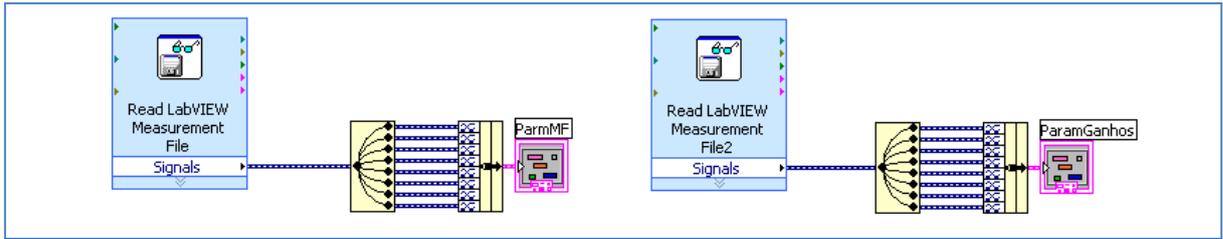


Figura 56 - Sub-rotina para leitura de dados em arquivo (ganhos e parâmetros das funções de pertinência)

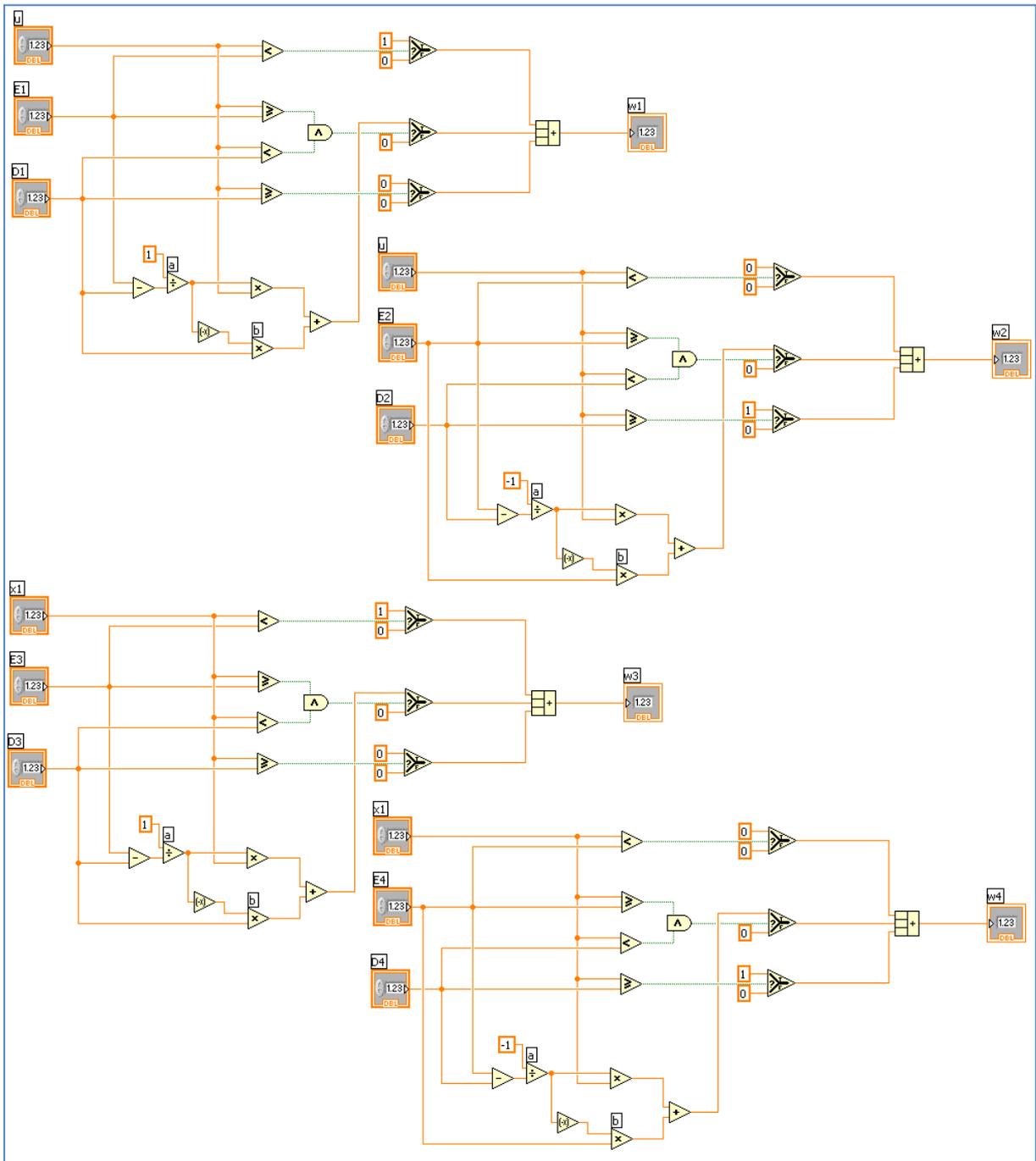


Figura 57 - Sub-rotina das funções de pertinência de entrada

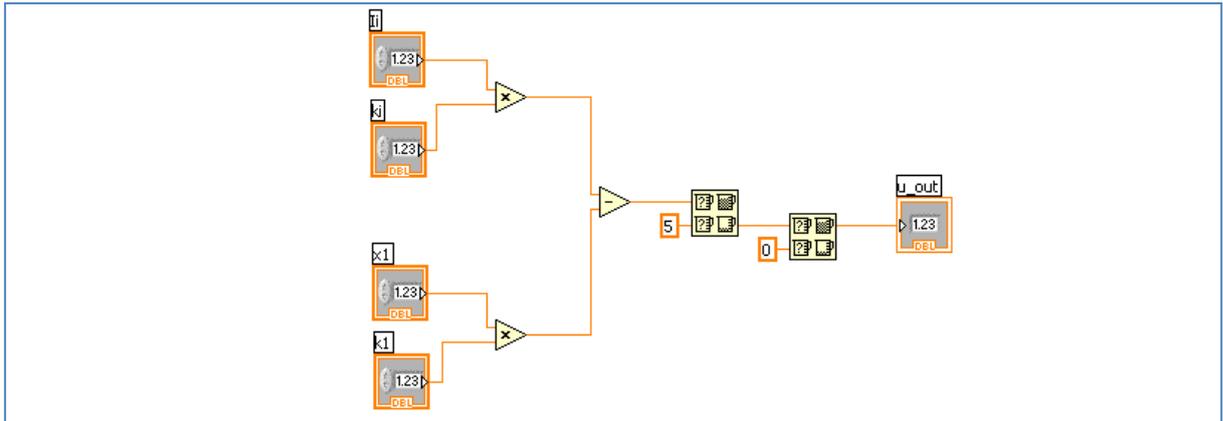


Figura 58 - Subrotina das funções consequentes do controlador

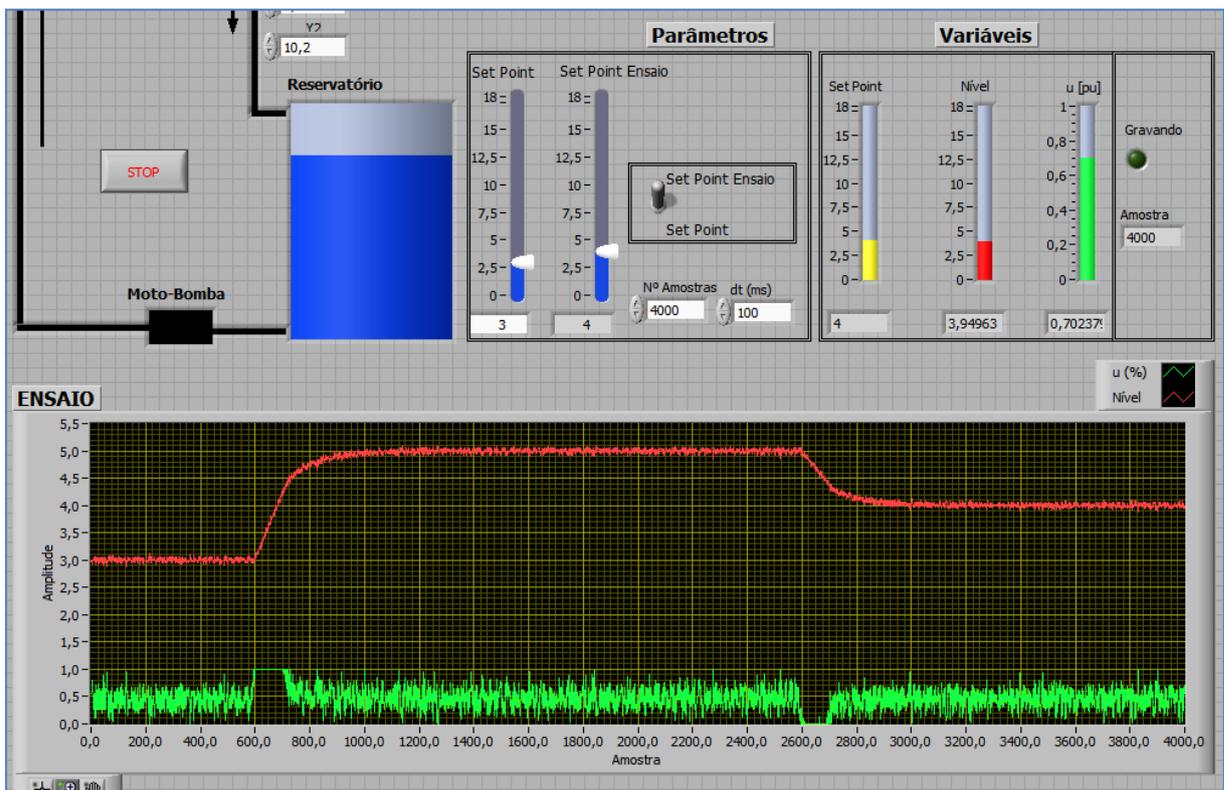


Figura 59 - Interface gráfica do sistema de controle