Trabalho Final de Sistemas Nebulosos

Thiago Malta Coutinho – 2014123335

1. Introdução

A teoria de conjuntos nebulosos parte do princípio de que um elemento possui um **grau de pertencimento** à um ou outro conjunto, diferente da teoria de conjuntos clássica em que o elemento **pertence** ou não à um conjunto. O conceito de pertencimento permite que conhecimento humano seja implementado em máquina de maneira simples através de regras linguísticas. As variáveis de processo são linguísticas, como *BAIXO, GRANDE, QUENTE, FRIO*, implementadas em forma de funções de pertinência.

O objetivo desse trabalho são duas implementações utilizando mecanismos de inferência nebulosa. O primeiro, um Sistema Fuzzy Adaptativo, é construído utilizando o mecanismo SUGENO e é aplicado à três problemas: aproximar uma função quadrática, modelagem de uma função não-linear de três entradas e predição de uma série temporal caótica. O segundo consiste em um controlador nebuloso, utilizado para controlar uma planta complexa.

2. Desenvolvimento

2.1 – SISTEMA FUZZY ADAPTATIVO

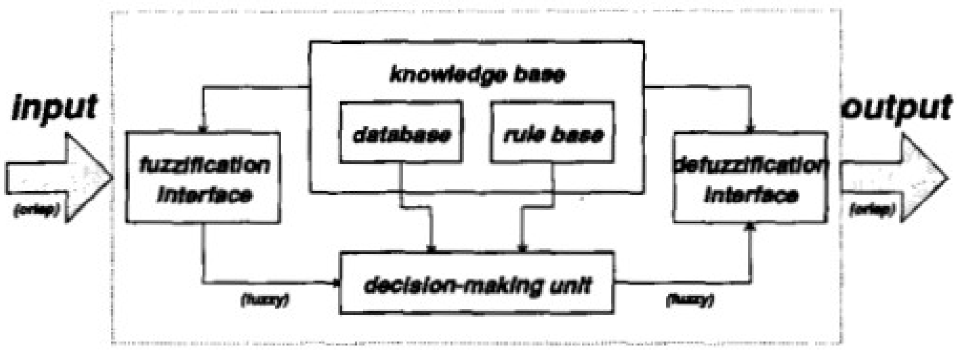

Fig. 1. Fuzzy inference system.

Um sistema fuzzy adaptativo(anfis) foi implementado pelos alunos em linguagem MATLAB. O sistema é baseado no método de inferência SUGENO, possui funções de pertinência gaussianas e utiliza o método do gradiente descendente para otimizar/atualizar seus parâmetros. O mecanismo de SUGENO de ordem 1 é basicamente n retas com m coeficientes, onde n é o número de entradas do problema e m é o número de regras fuzzy. A inferência feita será a soma ponderada das retas pelos seus respectivos pesos, produtório das m funções de pertinência pertencentes à i-ésima entrada, dividido pela soma dos pesos.

Assim temos a definição do sistema adaptativo com seus respectivos pesos e funções de atualização de erro:

ym = $\sum_{i=1}^{n} P_{im}.x_i+q_m$

Wj = $\prod_{i=1}^{n} \mu_{Aij}(x_i)$

$$\mu_{Aij}(x_i)=e^{\left[\frac{-1}{2}\left(\frac{x_i-c_{ij}}{\sigma_{ij}}\right)^2\right]}$$

ys = $ys=\frac{\sum_{j=1}^{m} w_j.y_j}{\sum_{j=1}^{m} w_j}=\frac{a}{b}$

Para atualizarmos o erro quadrático, utilizamos o método do gradiente descendente e derivamos os parâmetros em respeito ao erro(utilizando a regra da cadeia):

$$\min\left(e=\frac{1}{2}.(ys-yd)^2\right)$$

$$c_{ij^{k+1}}=c_{ij^k}-\alpha.\frac{\partial e}{\partial c_{ij}}|k$$

$$P_{ij^{k+1}}=P_{ij^k}-\alpha.\frac{\partial e}{\partial P_{ij}}|k$$

$$q_{j^{k+1}}=q_{j^k}-\alpha.\frac{\partial e}{\partial q_j}|k$$

$$\frac{\partial e}{\partial ys}=(ys-yd)$$

$$\frac{\partial ys}{\partial w_j}=\left(\frac{y_j-ys}{b}\right)$$

$$\frac{\partial w_j}{\partial c_{ij}}=\left[w_j\left(\frac{x_i-c_{ij}}{\sigma_{ij^2}}\right)\right]$$

$$\frac{\partial w_j}{\partial \sigma_{ij}}=\left\{w_j\left[\frac{\left(x_i-c_{ij}\right)^2}{\sigma_{ij^3}}\right]\right\}$$

$$\frac{\partial ys}{\partial y_j}=\left(\frac{w_j}{b}\right)$$

$$\frac{\partial y_j}{\partial P_{ij}}=x_i$$

$$\frac{\partial y_j}{\partial q_j}=1$$

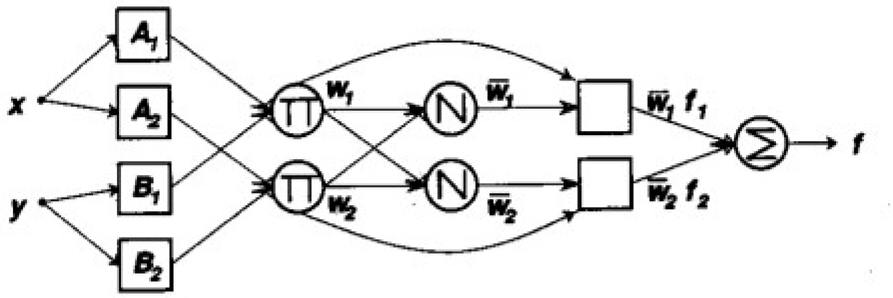

Fig2 - Estrutura ANFIS

Cada problema foi abordado utilizando a estrutura desenvolvida em sala e as funções genfis1 e genfis2 do matlab. A função genfis1 gera as funções de pertinência a partir de grids feitos no espaço de variáveis de entrada e pode acabar sofrendo do “mal da dimensionalidade”, quando o número de parâmetros aumenta exponencialmente com a dimensão do problema. A função genfis2 é mais otimizada e procura gerar funções de pertinência a partir de clusters de dados. Um parâmetro de raio deve ser passado a função para que os clusters possam ser gerados.

2.2.1 – Problema 1: aproximar a função y = x^2

O número de regras utilizadas por todos os métodos foi 5. Para o modelo adaptativo a taxa de aprendizado utilizada foi 0.05 com 20 epocas de treinamento. Para o modelo genfis2 o raio utilizado foi 0.8. Na figura abaixo podemos visualizar os resultados obtidos nos dados de validação dos modelos.

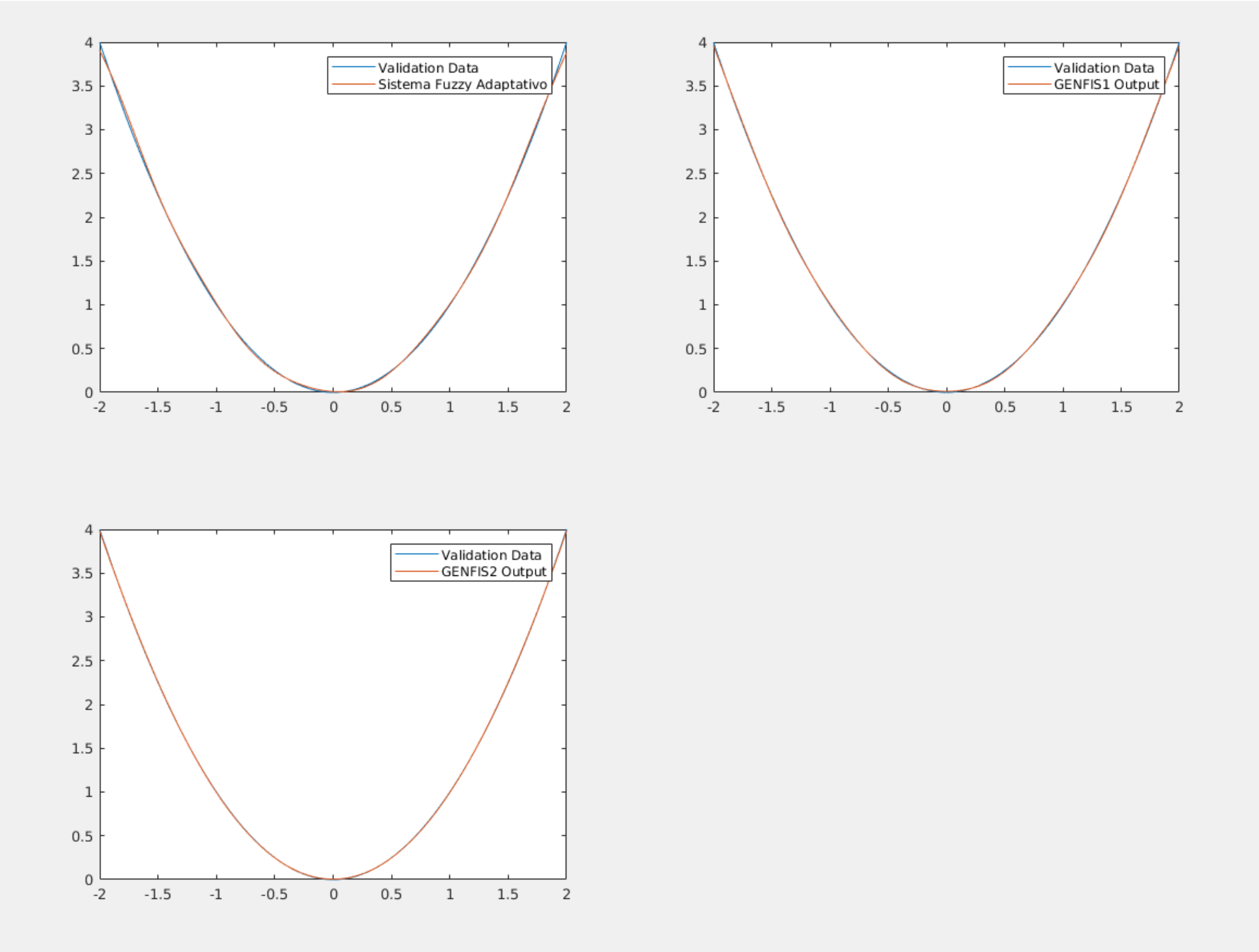

Fig 3 – Aproximações da função y=x^2 para os modelos Fuzzy Adaptativo, GENFIS1 e GENFIS2

Os erros finais(erro quadrático médio) obtidos por cada modelo foram:

MSE(Sistema Fuzzy Adaptativo): **3.5729e-04**

MSE(GENFIS1): **9.6223e-05**

MSE(GENFIS2): **1.9619e-05**

2.2.2 – Problema 2: aproximação de uma função não-linear com três entradas

No problema atual os parâmetros anteriores foram mantidos, apenas a taxa de aprendizado que foi aumentada para 0.01(necessário para que o modelo encontre uma solução). Os resultados obtidos estão expressos abaixo:

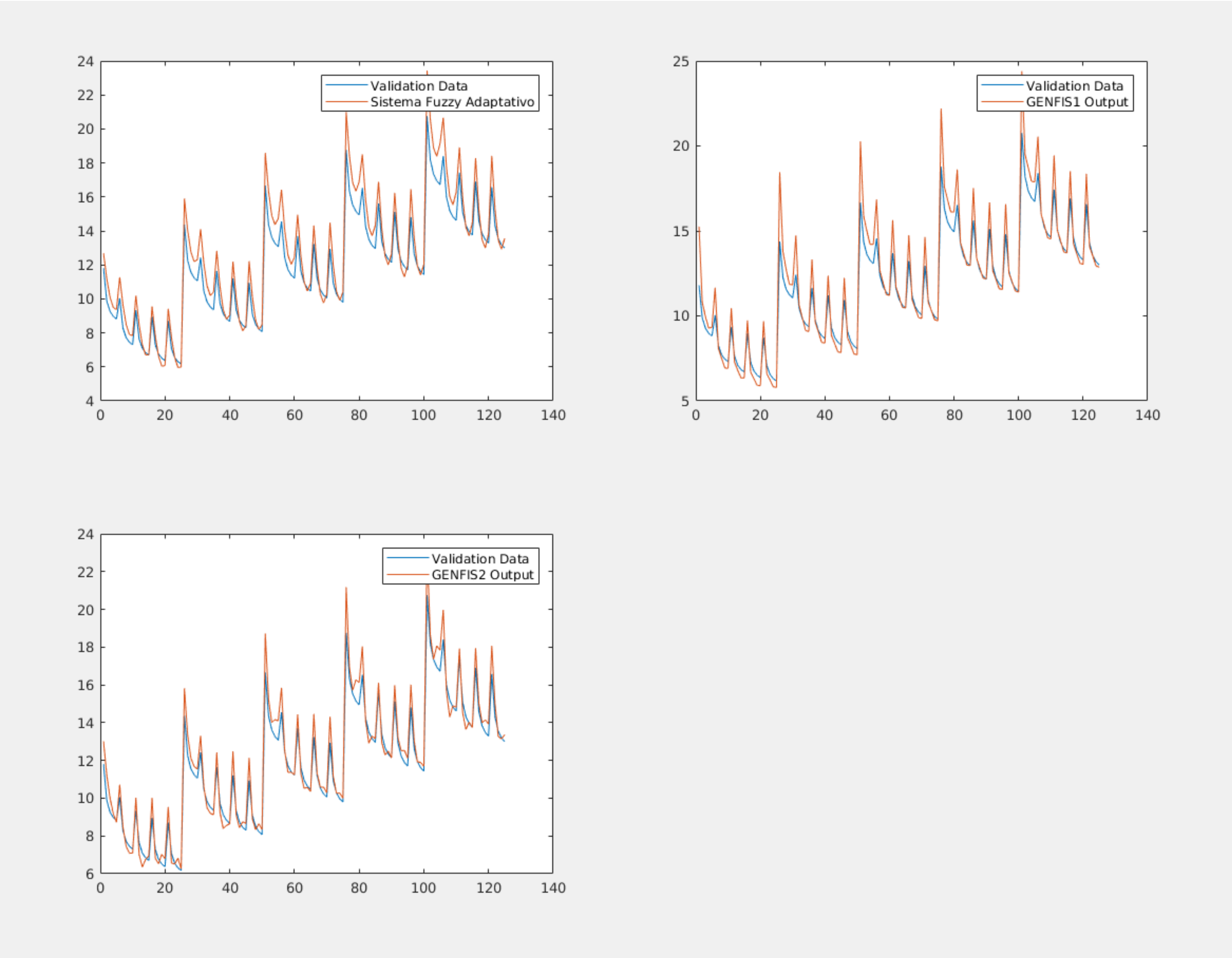

Fig 4 – Aproximações de função não-linear para os modelos Fuzzy Adaptativo, GENFIS1 e GENFIS2.

Os erros finais(erro quadrático médio) obtidos por cada modelo foram:

MSE(Sistema Fuzzy Adaptativo): **1.1774**

MSE(GENFIS1): **1.2489**

MSE(GENFIS2): **0.5335**

2.2.3 – Problema 3: Predição de uma série temporal caótica

A série a ser prevista é definida por:

$$\dot{x} = \frac{0.2x\left(t-\tau\right)}{1+x^{10}\left(t-\tau\right)} - 0.1x\left(t\right)$$

Com entradas x(t), x(t-6), x(t-12) e x(t-18) para a predição de x(t+6). Como os dados de entrada possuem dimensão maior, foram utilizadas 2 regras para a função genfis1 de forma a evitar o "mal da dimensionalidade". Os resultados obtidos para a predições estão expressos abaixo:

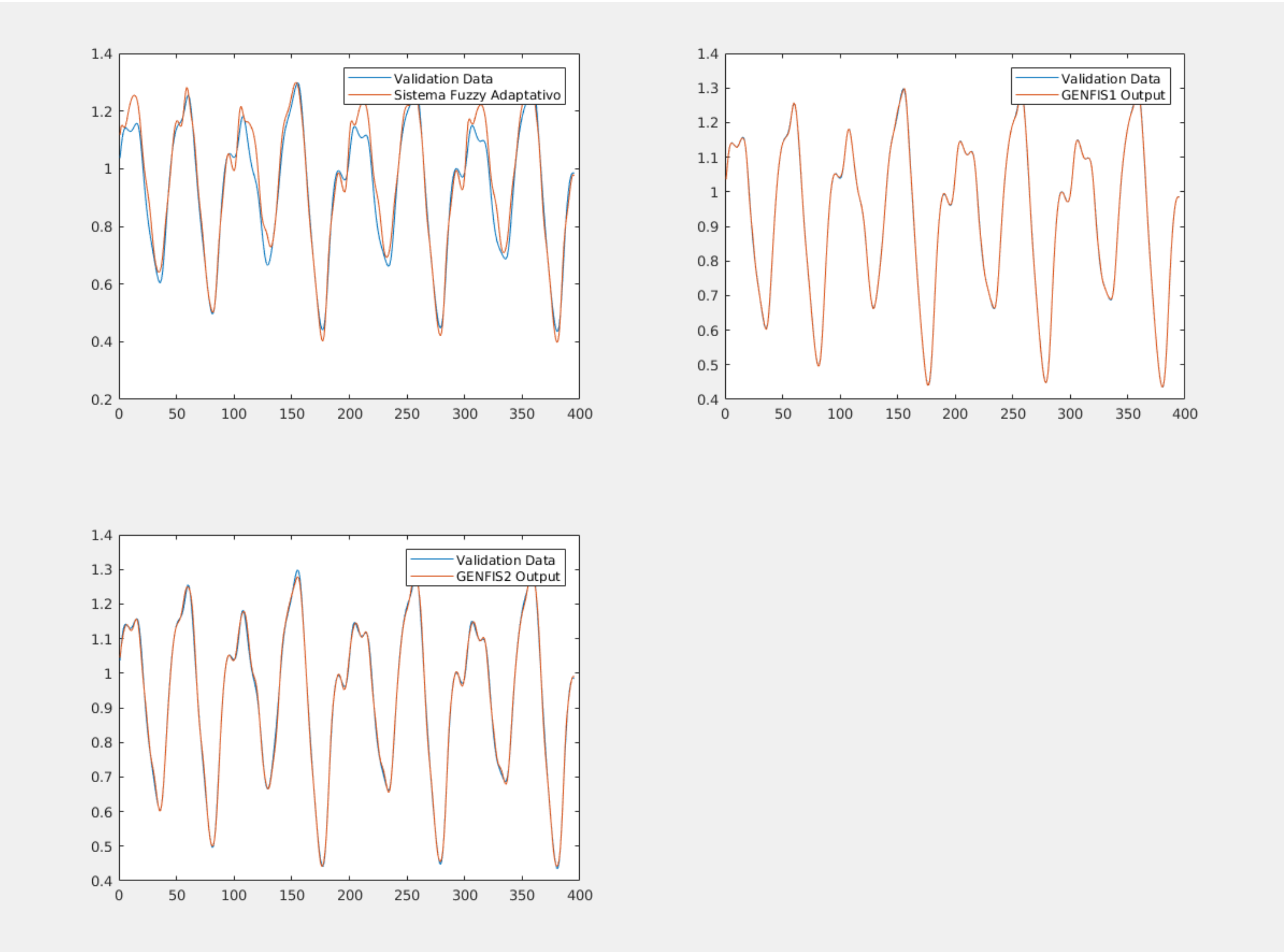

Fig 5 – Aproximações de série temporal caótica para os modelos Fuzzy Adaptativo, GENFIS1 e GENFIS2.

Os erros finais(erro quadrático médio) obtidos por cada modelo foram:

MSE(Sistema Fuzzy Adaptativo): **0.0024**

MSE(GENFIS1): **7.3181e-06**

MSE(GENFIS2): **9.2252e-05**

2.2 – CONTROLADOR NEBULOSO

Um controlador nebuloso permite a trasmissão de conhecimento do operador para o controlador. O conhecimento do operador sobre a planta que ele opera é uma informação muito imporatante e que ao ser incorporada ao controlador pode gerar resultados mais estáveis, rápidos e de menor complexidade.

O sistema consituti da seguinte equação:

$$y(k)=1.4*y(k-1) - 0.6*y(k-2) - 3*u(k-1)^3 + 2*u(k-1) - u(k-2)^3 + 2*u(k-2);$$

Com referência igual à:

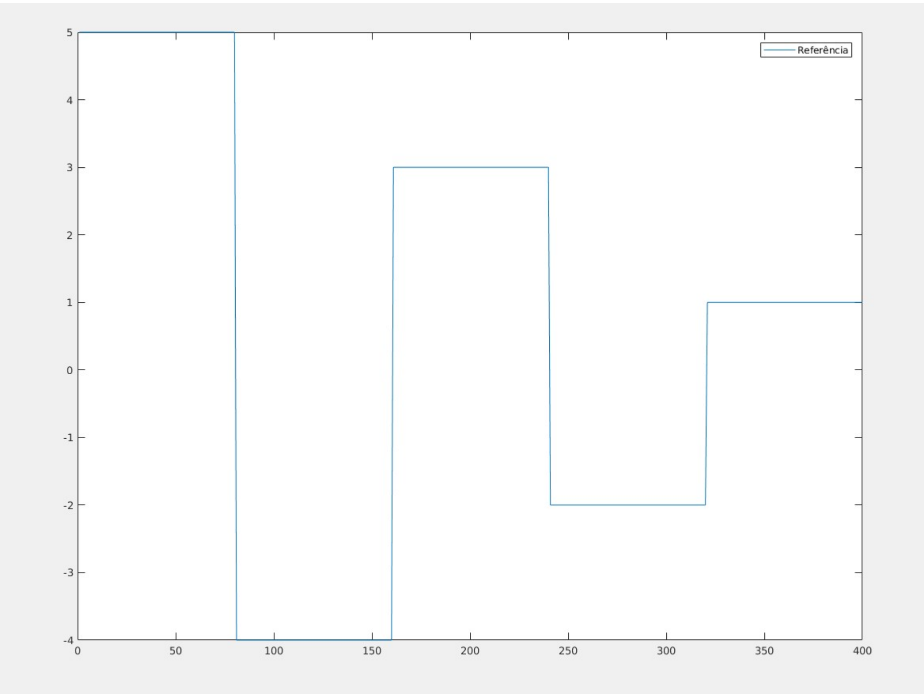

Fig 6 –Referência de resposta ao sistema

O controlador foi implementado utilizando o mecanismo de inferência Mamdani com as seguintes funções de pertinência para entra e ação de controle:

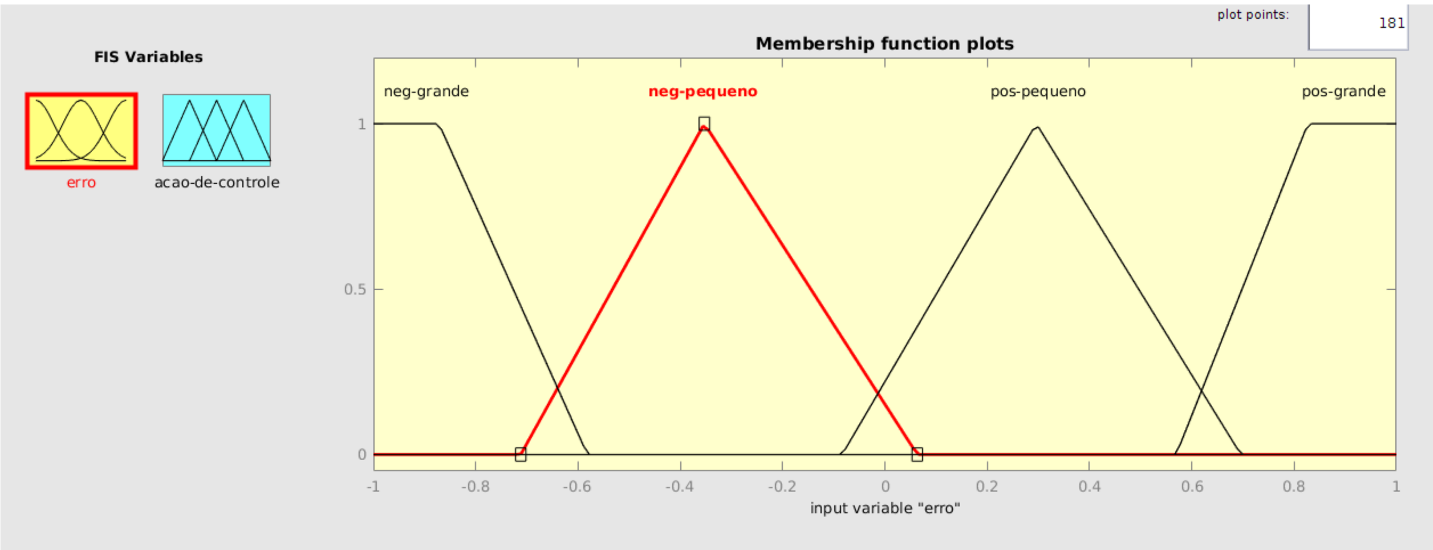

Fig 7 – Funções de pertinência da entrada “erro” do controlador nebuloso

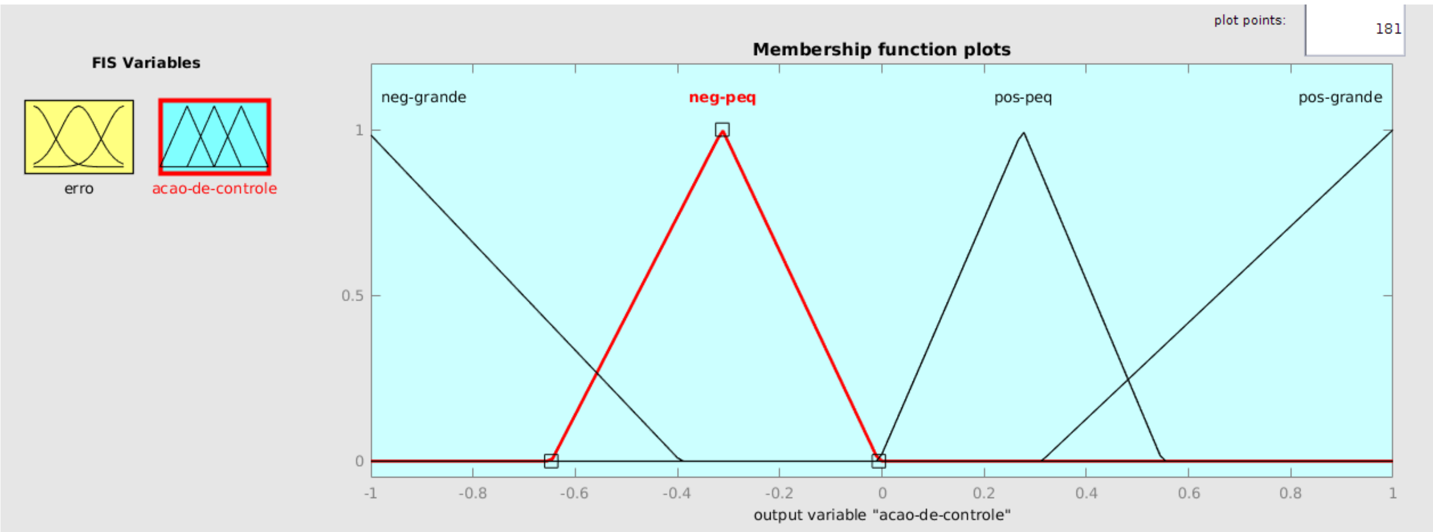

Fig 8 – Funções de pertinência da saída “acao-de-controle” do controlador

1. If (erro is neg-pequeno) then (acao-de-controle is neg-peq) (1)
2. If (erro is neg-grande) then (acao-de-controle is neg-grande) (1)
3. If (erro is pos-grande) then (acao-de-controle is pos-grande) (1)
4. If (erro is pos-pequeno) then (acao-de-controle is pos-peq) (1)

Fig 9 – Regras utilizadas no controlador nebuloso

A resposta obtida pelo controlador foi a seguinte:

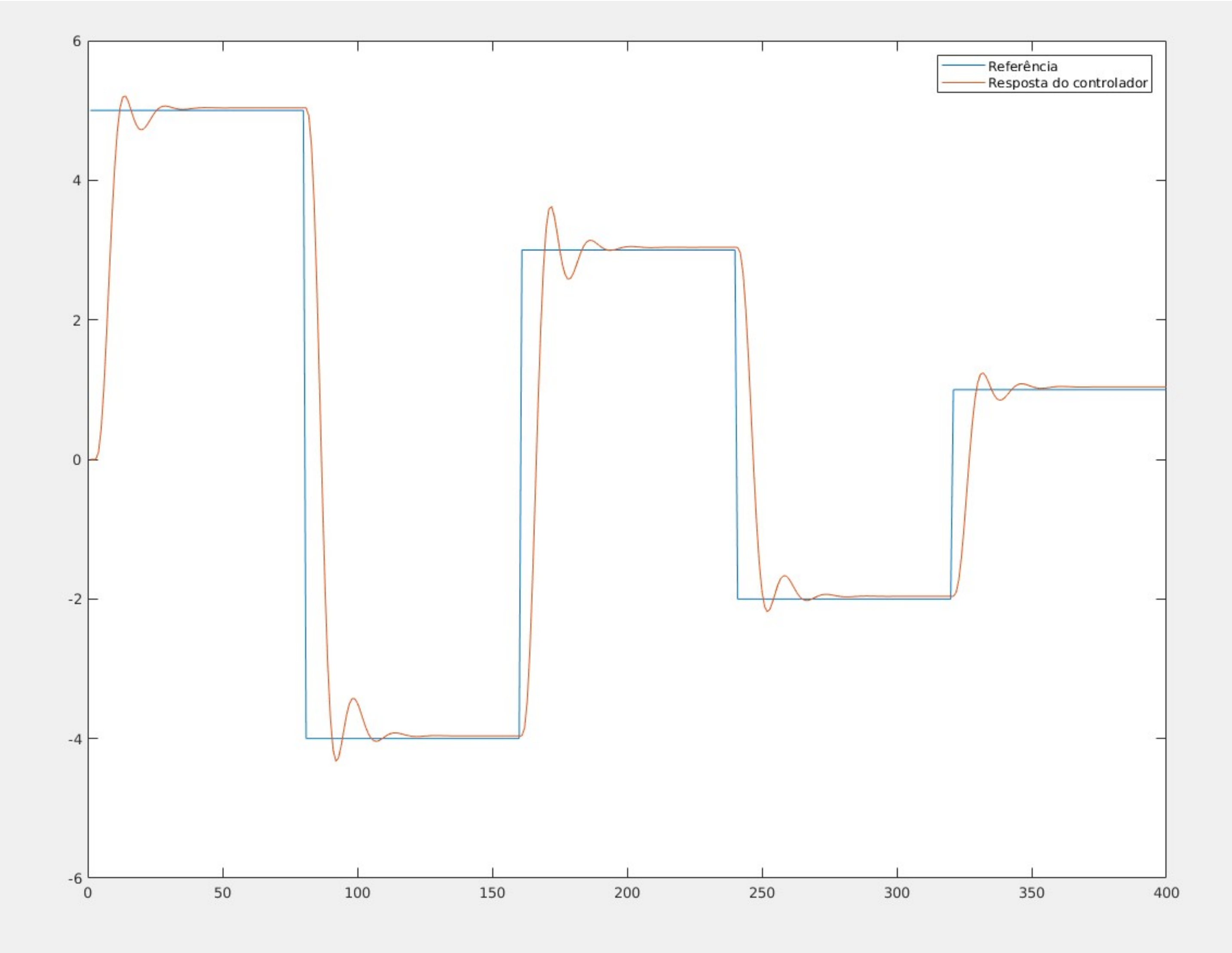

Fig 10 – Resposta de referência e resposta obtida pelo controlador nebuloso

3. Conclusões

No presente trabalho foi possível aplicar a inferência nebulosa para resolver problemas complexos de engenharia. Os mecanismos se mostraram de grande poder e aplicáveis a diversos problemas e àreas.

Ao utilizar o toolbox *Fuzzy* do matlab, foi possível a implementação de funções de pertinência de maneira rápida e eficiente. As funções genfis1 e genfis2 também se mostraram de fácil aprendizado e utilização. O sistema adaptativo implementado foi de grande ajuda para a compreensão do aluno sobre os mecanismos de inferência nebulosa e também se mostrou efieciente para um modelo simples e de fácil implementação.