

INTELIGÊNCIA COMPUTACIONAL**TRABALHO COMPUTACIONAL –4****- Enunciado:**

Os sistemas de inferência fuzzy podem ser utilizados em diversas aplicações. Uma delas é na modelagem de sistemas. Nessa aplicação, o sistema de inferência fuzzy é projetado para receber as variáveis de entrada do sistema e fornecer na saída uma função de saída que represente o modelo do sistema em questão.

- Questões:

1) Com base nos códigos de exemplo apresentados em anexo, implemente (A) um sistema de inferência fuzzy tipo Mandani e (B) um sistema de inferência tipo Takagi-Sugeno de primeira ordem para aproximar a função SINC no intervalo de 0 a 2π .

$$\text{Sinc}(x) = \frac{\sin(x)}{x}$$

2) Calcular a função SINC e calcular o erro quadrático médio de cada sistema de inferência implementado na aproximação da função.

$$EQM = \frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2$$

3) Elaborar um relatório contendo a descrição dos sistemas de inferência implementados, dos testes realizados e comentários sobre os resultados obtidos.

ANEXO 1: Função sistema de inferência de Mandani

```
function [ y ] = mandani_1(x)
%
% mandani_1
% Implementação do sistema de inferencia fuzzy tipo Mandani para 1 entrada
% (baseado no exemplo 4.1 do livro Neuro-fuzzy and Soft Computing)
%
% (c) Prof. Maurílio J. Inácio 2017
% Eng. de Computação / FACIT
%
% [ y ] = mandani_1(x)
%
% y = saída do sistema
% x = variável de entrada
%
% % Implementado em MATLAB ver. 2016a
%

% Parametros das funcoes de pertinencia dos antecedentes
A1 = [-20 -15 -6 -3]; % pequeno
A2 = [-6 -3 3 6]; % medio
A3 = [3 6 15 20]; % grande

% Parametros das funcoes de pertinencia dos consequentes
B1 = [-2.46 -1.46 1.46 2.46]; % pequeno
B2 = [1.46 2.46 5 7]; % medio
B3 = [5 7 13 15]; % grande

% Geracao das funcoes de pertinencia dos consequentes
y = 1 : 0.1 : 10;
for i = 1 : size(y,2);
    fp_B1(i) = trapmf(y(i), B1);
    fp_B2(i) = trapmf(y(i), B2);
    fp_B3(i) = trapmf(y(i), B3);
end

% Fuzzyficacao
pA1 = trapmf(x, A1); % pertinencia na funcao A1
pA2 = trapmf(x, A2); % pertinencia na funcao A2
pA3 = trapmf(x, A3); % pertinencia na funcao A3

% Calculo das regras fuzzy (composicao max-min)
% regra 1: Se X eh pequeno entao Y e pequeno
w1 = pA1; % forca de disparo da regra 1
fp_B11 = min(w1, fp_B1); % funcao de pertinencia B1 qualificada
% regra 2: Se X eh medio entao Y e medio
w2 = pA2; % forca de disparo da regra 2
fp_B22 = min(w2, fp_B2); % funcao de pertinencia B1 qualificada
% regra 3: Se X eh grande entao Y e grande
w3 = pA3; % forca de disparo da regra 3
fp_B33 = min(w3, fp_B3); % funcao de pertinencia B1 qualificada
```



FUNDAÇÃO EDUCACIONAL MONTES CLAROS
FACULDADE DE CIÊNCIA E TECNOLOGIA DE MONTES CLAROS
ENGENHARIA DE COMPUTAÇÃO



```
% agregacao das regras  
fp_y = max([fp_B11; fp_B22; fp_B33]); % funcao de pertinencia de saida  
  
% Defuzzyficacao  
y = defuzz(y, fp_y, 'centroid'); % saida
```

ANEXO 2: Script para teste da função sistema de inferência de Mandani

```
% Teste do Sistema Fuzzy tipo Mandani

clear;
close all;
clc;

x = -10 : 0.1 : 10;

for i = 1 : size(x,2)
    [ y(i) ] = mandani_1(x(i));
end

plot(x,y)
xlabel('X'); ylabel('Y');
title('Saida do Sistema Fuzzy tipo Mandani');
```

ANEXO 3: Função sistema de inferência de Takagi-Sugeno

```
function [ z ] = takagi_sugeno_1(x)
```

```
% _____ takagi_sugeno_1 _____  
% Implementação do sistema de inferência fuzzy tipo Takagi-Sugeno para 1  
% entrada  
% (baseado no exemplo 4.3 do livro Neuro-fuzzy and Soft Computing)  
%  
% (c) Prof. Maurílio J. Inácio 2017  
% Eng. de Computação / FACIT  
%  
% [ z ] = takagi_sugeno_1(x)  
%  
% z = saída do sistema  
% x = variável de entrada  
%  
% Implementado em MATLAB ver. R2016a  
%  
  
% Parametros das funcoes de pertinencia dos antecedentes  
A1 = [6 4 -10]; % pequeno  
A2 = [4 4 0]; % medio  
A3 = [6 4 10]; % grande  
  
% Fuzzyficacao  
pA1 = gbellmf(x, A1); % pertinencia na funcao A1  
pA2 = gbellmf(x, A2); % pertinencia na funcao A2  
pA3 = gbellmf(x, A3); % pertinencia na funcao A3  
  
% Calculo das regras fuzzy (operador produto)  
% regra 1: Se X eh pequeno entao Z eh z1  
w1 = pA1; % forca de disparo da regra 1  
z1 = 0.1*x + 6.4; % funcao linear do consequente 1  
% regra 2: Se X eh medio entao Z eh z2  
w2 = pA2; % forca de disparo da regra 2  
z2 = -0.5*x + 4; % funcao linear do consequente 2  
% regra 3: Se X eh grande entao Z eh z3  
w3 = pA3; % forca de disparo da regra 3  
z3 = x - 2; % funcao linear do consequente 3  
  
% Defuzzyficacao  
z = (w1*z1 + w2*z2 + w3*z3) / (w1 + w2 + w3); % saida
```

ANEXO 4: Script para teste da função sistema de inferência de Takagi-Sugeno

```
% Teste do Sistema Fuzzy tipo Takagi-Sugeno

clear;
close all;
clc;

x = linspace(-10, 10, 200);

for i = 1 : size(x,2)
    [ z(i) ] = takagi_sugeno_1(x(i));
end

plot(x,z)
xlabel('X'); ylabel('Z');
title('Saída do Sistema Fuzzy tipo Takagi-Sugeno');
```