



**Universidade de São Paulo
Escola de Engenharia de São Carlos**

Lógica fuzzy utilizando toolbox do MATLAB

Tutorial do toolbox, exemplos e aplicações

Matheus Vitorasso Zanetti - 8006460

Universidade de São Paulo
Escola de Engenharia de São Carlos

4 de abril de 2016

- 1 Toolbox Fuzzy MATLAB com exemplo gorjeta
- 2 Gerando vetor de superfície
- 3 Exercício

Para abrir o recurso no Matlab, basta digitar "fuzzy" na linha de comando

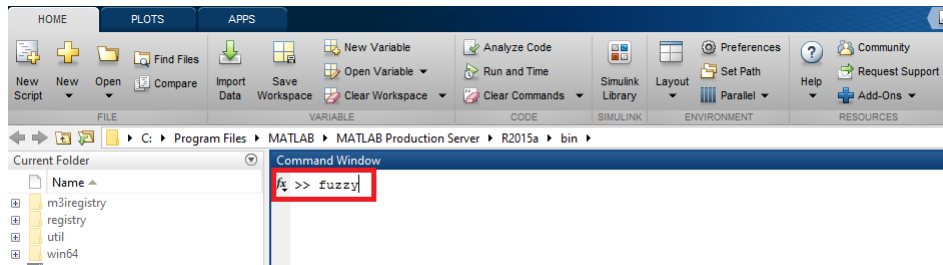


Figura 1: Comando para abrir toolbox

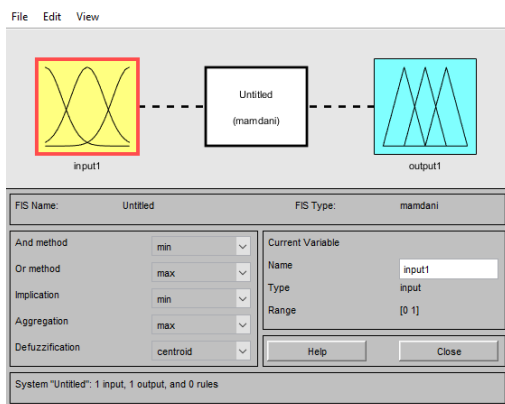


Figura 2: Janela da toolbox

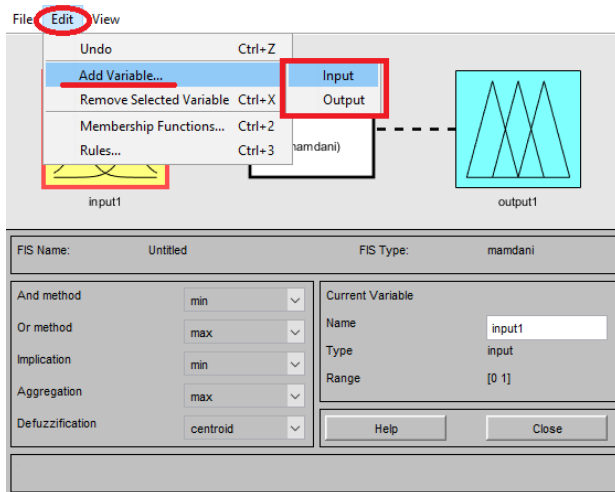
The image shows the MATLAB Fuzzy Logic Toolbox interface with several components annotated by red boxes and lines:

- Nome do sistema**: Points to the central box containing "Untitled" and "(mamdani)".
- Variáveis de entrada**: Points to the yellow box on the left labeled "input1" containing two overlapping sine waves.
- Variáveis de saída**: Points to the cyan box on the right labeled "output1" containing three overlapping triangular waves.
- Implicação**: Points to the central box, specifically to the "(mamdani)" text.
- Tipos de conectivos "E", "OU", Implicação, Agregação e Defuzzificação**: Points to the configuration panel on the left, which includes:
 - And method: min
 - Or method: max
 - Implication: min
 - Agregation: max
 - Defuzzification: centroid
- Informações da variável selecionada**: Points to the "Current Variable" panel on the right, which shows:
 - Name: input1
 - Type: input
 - Range: [0 1]
- Informações gerais**: Points to the bottom status bar, which displays: "System 'Untitled': 1 input, 1 output, and 0 rules".

Figura 2: Informações da janela

Por onde começar?

Começe adicionando a quantidade adequada de variáveis de entrada e saída



The screenshot displays the MATLAB Fuzzy Logic Designer interface. The 'Edit' menu is open, with 'Add Variable...' selected. A sub-menu shows 'Input' and 'Output' options. The 'input1' variable is highlighted in yellow, and the 'output1' variable is shown as a cyan box with three triangles. The FIS Name is 'Untitled' and the FIS Type is 'mamdani'. The 'And method' is 'min', 'Or method' is 'max', 'Implication' is 'min', 'Aggregation' is 'max', and 'Defuzzification' is 'centroid'. The 'Current Variable' section shows 'Name: input1', 'Type: input', and 'Range: [0 1]'. Buttons for 'Help' and 'Close' are visible.

FIS Name:	Untitled	FIS Type:	mamdani
And method	min	Current Variable	
Or method	max	Name	input1
Implication	min	Type	input
Aggregation	max	Range	[0 1]
Defuzzification	centroid		

Como editar funções de pertinência

The image shows the MATLAB Fuzzy Inference System (FIS) editor interface. At the top, the 'Edit' menu is open, with the 'Membership Functions...' option highlighted in blue and a red box around it. The workspace contains an input variable 'input2' with two bell-shaped membership functions, a rule block labeled 'Untitled (mamdani)', and an output variable 'output1' with three triangular membership functions. The bottom panel shows the FIS settings: FIS Name: Untitled, FIS Type: mamdani, And method: min, Or method: max, Implication: min, Aggregation: max, Defuzzification: centroid, and Current Variable: input1.

Figura 5: Como acessar funções de pertinência

Como editar funções de pertinência

The screenshot displays the MATLAB Fuzzy Logic Designer interface. At the top, there is a menu bar with 'File', 'Edit', and 'View'. Below it, the 'FIS Variables' section shows three variables: 'Comida' (highlighted with a red box), 'Gorgeta', and 'Serviço'. Each variable has a corresponding membership function icon. The 'Comida' variable is associated with a red line on the plot.

The main plot, titled 'Membership function plots', shows three membership functions: 'mf2' (a triangular function peaking at 0.5), 'mf3' (a right-shoulder function), and 'mf1' (a left-shoulder function, highlighted with a red box). The x-axis is labeled 'input variable "Comida"' and ranges from 0 to 1. The y-axis represents the membership degree, ranging from 0 to 1. A red line is drawn across the plot, starting at (0, 1) and ending at (0.4, 0).

Below the plot, the 'Current Variable' section shows 'Comida' as the selected variable, with an 'input' type and a range of [0 1]. The 'Current Membership Function' section shows 'trimf' as the selected function type, with parameters [-0.4 0 0.4].

Four red callout boxes with white text point to specific elements: 'Variáveis Fuzzy' points to the FIS Variables section; 'Funções de pertinência e discretização' points to the membership function plot; 'Parâmetros das Variáveis' points to the 'Current Variable' section; and 'Parâmetros das funções de pertinência' points to the 'Current Membership Function' section.

Figura 6: Informações da janela

Exemplo de funções

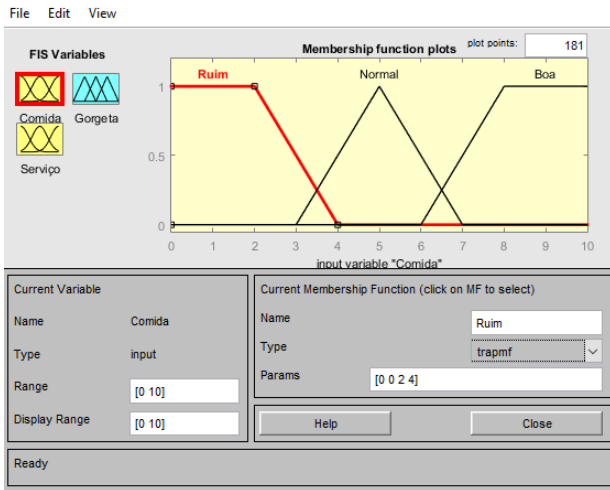


Figura 7: Exemplo

Adicionando regras

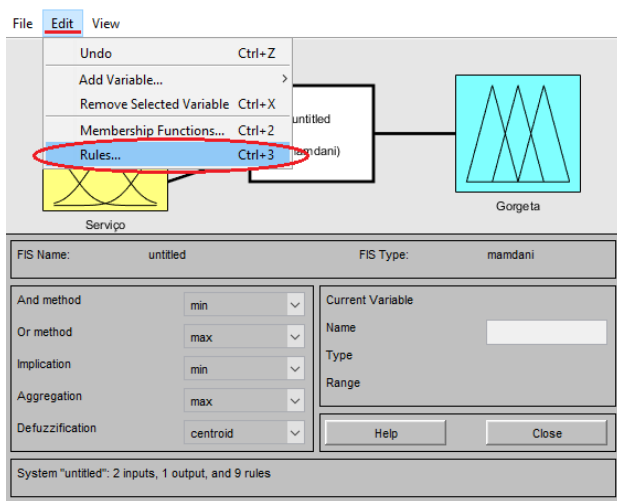


Figura 8: Como adicionar regras

Adicionando regras

The screenshot shows the MATLAB Fuzzy Logic Designer interface. At the top, there is a menu bar with 'File', 'Edit', 'View', and 'Options'. Below the menu bar is a list of rules:

1. If (Comida is Ruim) and (Serviço is Ruim) then (Gorgeta is Pequena) (1)
2. If (Comida is Normal) and (Serviço is Ruim) then (Gorgeta is Pequena) (1)
3. If (Comida is Boa) and (Serviço is Ruim) then (Gorgeta is Razoável) (1)
4. If (Comida is Ruim) and (Serviço is Médio) then (Gorgeta is Pequena) (1)
5. If (Comida is Normal) and (Serviço is Médio) then (Gorgeta is Razoável) (1)
6. If (Comida is Boa) and (Serviço is Médio) then (Gorgeta is Razoável) (1)
7. If (Comida is Ruim) and (Serviço is Bom) then (Gorgeta is Razoável) (1)
8. If (Comida is Normal) and (Serviço is Bom) then (Gorgeta is Grande) (1)
9. If (Comida is Boa) and (Serviço is Bom) then (Gorgeta is Grande) (1)

Below the list is a rule editor. The 'If' section contains two dropdown menus: 'Comida is' with 'Ruim' selected and 'Serviço is' with 'Ruim' selected. The 'Then' section contains a dropdown menu: 'Gorgeta is' with 'Pequena' selected. The connection is set to 'and'. A central box contains the text: 'Selecionar variáveis de entrada e mapear a saída de acordo com as regras'. At the bottom, there are buttons for 'Delete rule', 'Add rule', and 'Change rule', along with navigation arrows. The status bar at the bottom left says 'Renamed FIS to "untitled"' and the bottom right has 'Help' and 'Close' buttons.

Figura 9: Regras adicionadas

Como visualizar resultados

The screenshot displays the MATLAB Fuzzy Logic Designer interface. At the top, the 'View' menu is open, with 'Rules' (Ctrl+5) and 'Surface' (Ctrl+6) options highlighted. The workspace shows a Mamdani FIS named 'untitled' with two input variables, 'Comida' and 'Serviço', and one output variable, 'Gorgeta'. The 'Comida' and 'Serviço' variables are represented by yellow membership function plots, while 'Gorgeta' is represented by a cyan membership function plot. The configuration panel below the workspace shows the FIS Name as 'untitled' and the FIS Type as 'mamdani'. The 'And method' is set to 'min', 'Or method' to 'max', 'Implication' to 'min', 'Aggregation' to 'max', and 'Defuzzification' to 'centroid'. The 'Current Variable' section is empty. The interface includes 'Help' and 'Close' buttons and a status bar at the bottom that reads 'Ready'.

Figura 10: Visualizando resultados

Janela de resultados



Figura 11: Janela interativa de resultados

Visualizar superfície dos resultados

O Matlab consegue mapear todas as possibilidades de entradas e suas respectivas saídas em um gráfico tridimensional

File Edit **View**

- Rules Ctrl+5
- Surface Ctrl+6

Comida

Serviço

exemplo gorjeta
(mamdani)

Gorjeta

FIS Name: exemplo gorjeta FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

Name: Comida

Type: input

Range: [0 10]

Help Close

Ready

Janela de superfície

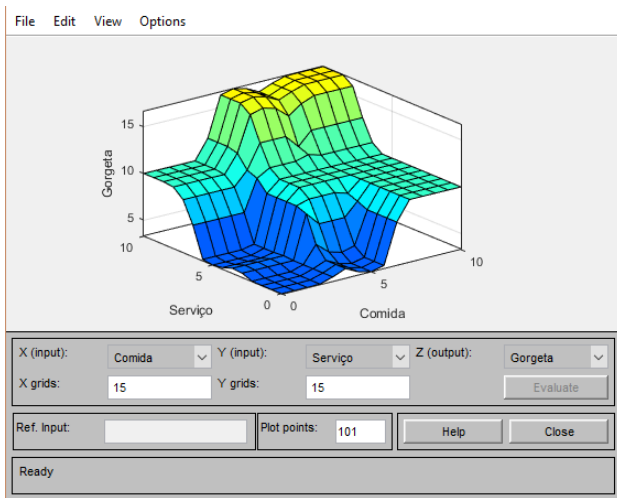


Figura 13: Janela de superfície

Matlab é um software perfeito para o quê?

Matlab é um software perfeito para o quê?

Prototipagem!

Matlab é um software perfeito para o quê?

Prototipagem!

Por isso o utilizamos para fazer simulações e modelagem. Mas como podemos aplicá-lo às respostas de nosso problema?

Gerando um vetor de superfície

Geraremos um vetor de superfície, e salvaremos seus dados em um arquivo de texto, para facilmente carregarmos na memória de um microcontrolador no futuro.

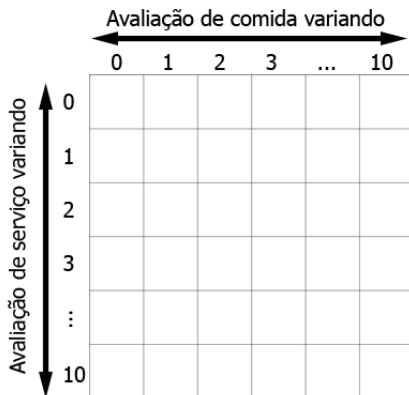


Figura 14: Mapeamento de superfície

Depois de salvar as configurações fuzzy do toolbox (em "file", "export to file"), é de valia conhecer duas funções do Matlab que manipulam o arquivo salvo:

Leitura do Sistema Fuzzy

➤ `var_fis = readfis('arquivo_fis')` % Atribui o sistema fuzzy definido por "arquivo_fis" à variável "var_fis".

Respostas do Sistema Fuzzy

➤ `saida = evalfis(M, var_fis)` % Simula o sistema fuzzy para uma matriz M de entradas, onde cada linha de M é um vetor de entrada.

Criando uma rotina que gera vetor de superfície

```
clc
clear all
close all
var_fis = readfis('myfile.fis'); %Salva o arquivo do toolbox
%no workspace
serv = linspace(0,10,40); % quarenta pontos de resolucao
alim = linspace(0,10,40);
M = zeros(40,40); %Inicializa o vetor M
for i =1:40 % Loop para mapear todos pontos
    for j =1:40
        M(i,j) = evalfis([serv(i),alim(j)],var_fis);
    end
end

figure
surf(M) %Plotar a superficie
save dados.txt M -ASCII; % Salvar vetor M em arquivo txt
```

Exemplo prático

Imagine que você é engenheiro em uma usina sucroalcooleira, que opta por queimar bagaço de cana para alimentar uma planta de cogeração em sua caldeira. Não há atualmente nenhuma medição na vazão mássica (consumo) do combustível. A diretoria gostaria de uma solução para contabilizar o quanto de bagaço a caldeira está demandando, porém as soluções triviais para pesar dinamicamente o bagaço sendo transportado ou falharam ou são muito caras.

Com este problema em mãos, você decide utilizar lógica Fuzzy para estimar a quantidade de bagaço que entra na fornalha. Você decide utilizar como dados de entrada um valor medido já presente, a vazão de vapor na caldeira, e instalar um sensor barato de umidade para o combustível, ciente de que a umidade afetará o poder calorífico do mesmo.

O exercício é livre para modelagem do aluno, porém segue um exemplo de funções de pertinência e regras para melhor ambientação ao problema:

1 Vazão de vapor [ton/h]

- Baixa: Trapezoidal - $a = 0$; $b = 50$; $c = 60$; $d = 80$
- Normal : Trapezoidal - $a = 60$; $b = 80$; $c = 90$; $d = 110$
- Alta : Trapezoidal - $a = 90$; $b = 110$; $c = 120$; $d = 121$

2 Umidade em base úmida [%]

- Seco: Trapezoidal - $a = 0$; $b = 45$; $c = 49$; $d = 54$
- Normal : Trapezoidal - $a = 50$; $b = 54$; $c = 56$; $d = 60$
- Úmido: Trapezoidal - $a = 56$; $b = 61$; $c = 65$; $d = 66$

3 Consumo de biomassa [ton/h]

- Baixo: Trapezoidal - $a = 0$; $b = 20$; $c = 26$; $d = 33$
- Médio: Trapezoidal - $a = 27$; $b = 33$; $c = 36$; $d = 42$
- Elevado: Trapezoidal - $a = 36$; $b = 43$; $c = 50$; $d = 51$

Regras:

- Se vazão é baixa E umidade é seco ENTÃO consumo é baixo;
- Se vazão é baixa E umidade é normal ENTÃO consumo é baixo;
- Se vazão é baixa E umidade é úmido ENTÃO consumo é médio;
- Se vazão é normal E umidade é seco ENTÃO consumo é médio;
- Se vazão é normal E umidade é normal ENTÃO consumo é médio;
- Se vazão é normal E umidade é úmido ENTÃO consumo é elevado;
- Se vazão é alta E umidade é seco ENTÃO consumo é médio;
- Se vazão é alta E umidade é normal ENTÃO consumo é elevado;
- Se vazão é alta E umidade é úmido ENTÃO consumo é elevado;

Tomem alguns minutos para resolverem o exercício proposto da usina.

- 1 Notas de aula do prof. Dr. Ivan Nunes da Silva - SEL0421 Sistemas Inteligentes
- 2 Documentação Matlab

Convite para usar \LaTeX

Esta apresentação foi redigida em \LaTeX , utilizando o compilador online gratuito sharelatex.

Fica aqui o convite do autor para que o leitor tome a iniciativa para experimentar a redigir suas apresentações, relatórios, e textos científicos em \LaTeX .

Vai lhe salvar de muita dor de cabeça! (Numeração automática de páginas e figuras e equações, citações, equações com símbolos não usuais, etc...) Muito obrigado!

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$