**Practical**

**Fuzzy Logic**

**MATLAB Command for Fuzzy Logic Toolbox**

Key Features

* Fuzzy Logic Design app for building fuzzy inference systems and viewing and analyzing results
* Membership functions for creating fuzzy inference systems
* Support for AND, OR, and NOT logic in user-defined rules
* Standard Mamdani and Sugeno-type fuzzy inference systems
* Automated membership function shaping through neuroadaptive and fuzzy clustering learning techniques
* Ability to embed a fuzzy inference system in a Simulink model
* Ability to generate embeddable C code or stand-alone executable fuzzy inference engines

Creating Custom AND and OR Functions

The custom AND and OR inference functions must operate column-wise on a matrix, in

the same way as the MATLAB functions max, min, or prod.

For a row or column matrix x, min(x) returns the minimum element.

x = [1 2 3 4];

min(x)

For a matrix x, min(x) returns a row vector containing the minimum element from each Column.

x = [1 2 3 4;5 6 7 8;9 10 11 12];

min(x)

The function min(x,y) returns an array that is same size as x and y with the minimum elements from x or y. Either of the input arguments can be a scalar. Functions such as max, and prod operate in a similar manner.

In the toolbox, the AND implication methods perform an element by element matrix operation, similar to the MATLAB function min(x,y).

a = [1 2; 3 4];

b = [2 2; 2 2];

min(a,b)

Tipping Problem from the Command Line

To load this system, type:

a = readfis('tipper.fis')

To access the various components of this structure, type the component name after a at the MATLAB prompt. For example:

a.type

The function

getfis(a)

returns almost the same structure information that typing a, alone does:

Similarly, you can obtain structure information using getfis in this manner.

getfis(a,'input',1)

getfis(a,'output',1)

getfis(a,'input',1,'mf',1)

The function setfis acts as the reciprocal to getfis. It allows you to change any property of a FIS. For example, if you wanted to change the name of this system, you could type

a = setfis(a,'name','gratuity');

However, because a is already a MATLAB structure, you can set this information more simply by typing

a.name = 'gratuity';

Now the FIS structure a has been changed to reflect the new name. If you want a little more insight into this FIS structure, try

showfis(a)

Because the variable, a, designates the fuzzy tipping system, you can display any of the GUIs for the tipping system directly from the command line. Any of the following functions will display the tipping system with the associated GUI:

• **fuzzyLogicDesigner(a)** displays the Fuzzy Logic Designer

• **mfedit(a)** displays the Membership Function Editor

• **ruleedit(a)** displays the Rule Editor

• **ruleview(a)** displays the Rule Viewer

• **surfview(a)** displays the Surface Viewer

**System Display Functions**

**Three functions:** plotfis, plotmf, and gensurf

plotfis(a): Shows the whole fuzzy inference System

**plotmf:** Plots all the membership functions associated with a given variable as follows:

plotmf(a,'input',1)

plotmf(a,'output',1)

**gensurf:** plots any one or two inputs versus any one output of a given system

The result is either a two-dimensional curve, or a three-dimensional surface. When there are three or more inputs, gensurf must be generated with all but two inputs fixed, as is described in gensurf.

gensurf(a)

FIS Evaluation

evalfis: To evaluate the output of a fuzzy system for a given input

a = readfis('tipper');

evalfis([1 2],a)

This function can also be used for multiple collections of inputs, because different input vectors are represented in different parts of the input structure.

evalfis([3 5;2 7],a)