CO542 – Neural Networks and Fuzzy Systems
Lab 1 – Fuzzy Logic

Designing Fuzzy Control Systems

1. Identify the variables (inputs, states and outputs) of the system
2. Partition the universe of discourse or the interval spanned by each variable into a number of fuzzy subsets, assigning each a linguistic label (subsets include all the elements in the universe).
3. Assign or determine a membership function for each fuzzy subset.
4. Assign the fuzzy relationships between the inputs or state's fuzzy subsets on the one hand and the outputs fuzzy subsets on the other hand forming the rule base.
5. Choose appropriate scaling factors for the input and output variables in order to normalize the variables to the [0 1] or the [-1 1] interval.
6. Fuzzify the inputs to the controller.
7. Use fuzzy approximate reasoning to infer the output contributed from each rule.
8. Aggregate the fuzzy outputs recommended by each rule.
9. Apply defuzzification to form a crisp output.

Using MATLAB Fuzzy Fuzzy Logic Toolbox to Design Fuzzy Systems

Fuzzy Logic Toolbox provides functionalities for analyzing, designing, and simulating systems based on fuzzy logic. Functions are provided for many common methods, including fuzzy clustering and adaptive neurofuzzy learning.

Using this toolbox you can model complex system behaviors using simple logic rules, and then implement these rules in a fuzzy inference system. There is inbuilt functionality to simulate your designed fuzzy system.

For more details refer :

1. https://ww2.mathworks.cn/help/fuzzy/getting-started-with-fuzzy-logic-toolbox.html
2. https://www.youtube.com/watch?v=wBrHEXkTero

Simple Tutorial

- Open MATLAB and type 'Fuzzy' in the command prompt. This will open the FIS (Fuzzy Inference System)
- Select File→ New FIS → Mamdani
- You can add more input using Edit → Add Variable → Input
- To edit and add membership functions Edit → Membership Functions
- To edit rules Edit → Rules

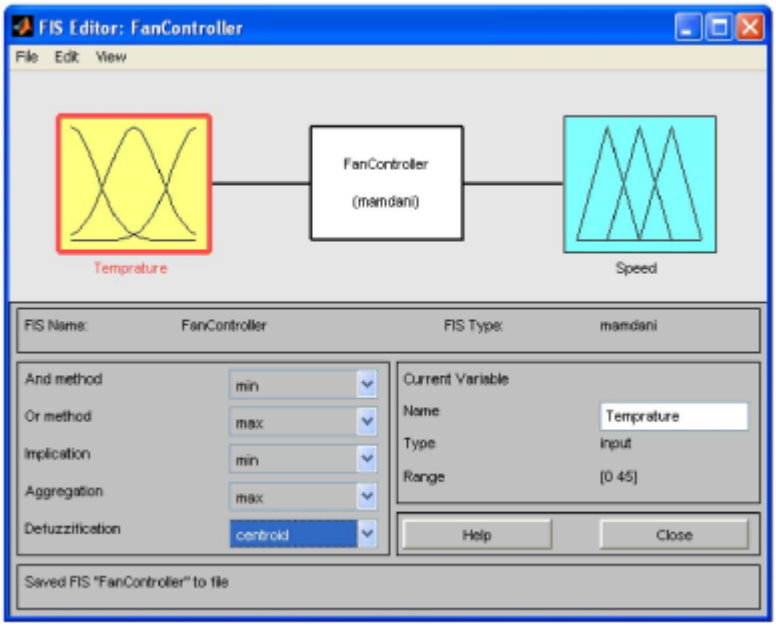

Figure 1: FIS Editor in MATLAB Fuzzy Toolbox

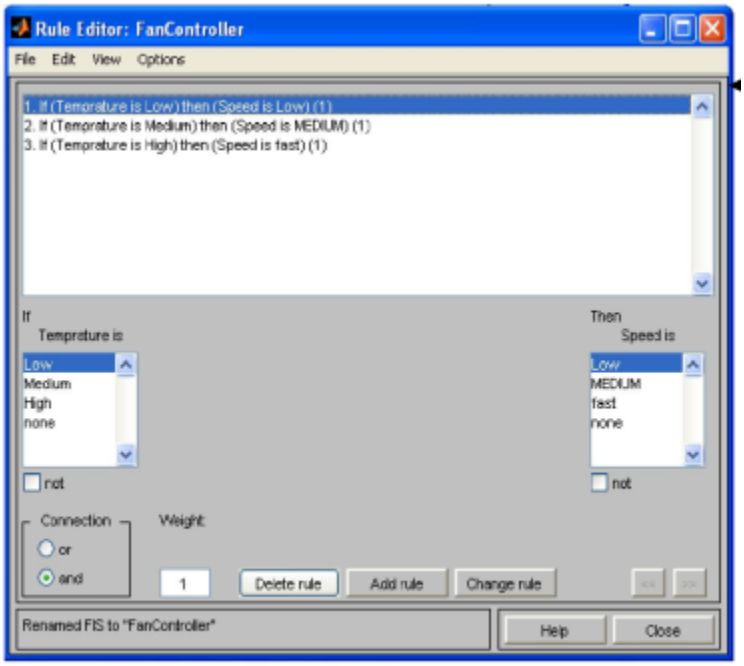

Figure 2: Rule Editor Window

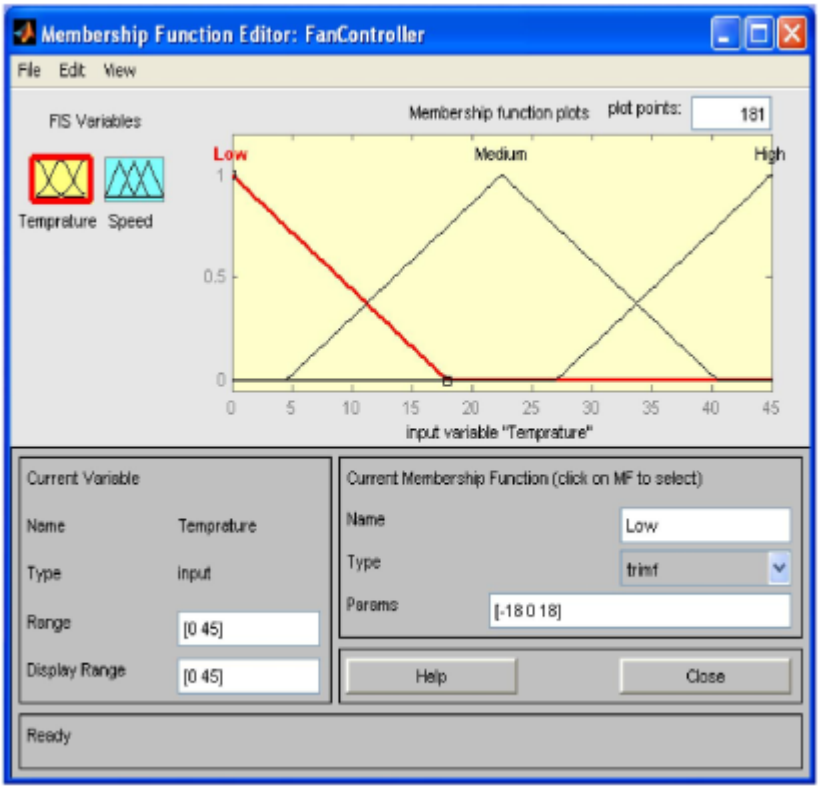

Figure 3: Membership Function Editor Window

Scenario – One Input One Output

Suppose you need to control the speed of a motor by changing the input voltage. When a set point is defined, if for some reason, the motor runs faster, we need to slow it down by reducing the input voltage. If the motor slows below the set point, the input voltage must be increased so that the motor speed reaches the set point.

Use the following as input and output action words:

Input:	Output:
Too slow	Less voltage (Slow down)
Just right	No change
Too fast	More voltage (Speed up)

Lab Tasks

1. Define the rule-base for the scenario (Refer Figure 4).
2. Suppose, the speed increases from the set point of 2420 to 2437.4 rpm. This is depicted on the membership function as shown in Figure 5. Calculate required voltage to maintain a rpm at **set speed**.
   - Use Mamdani model and maximum defuzzification method

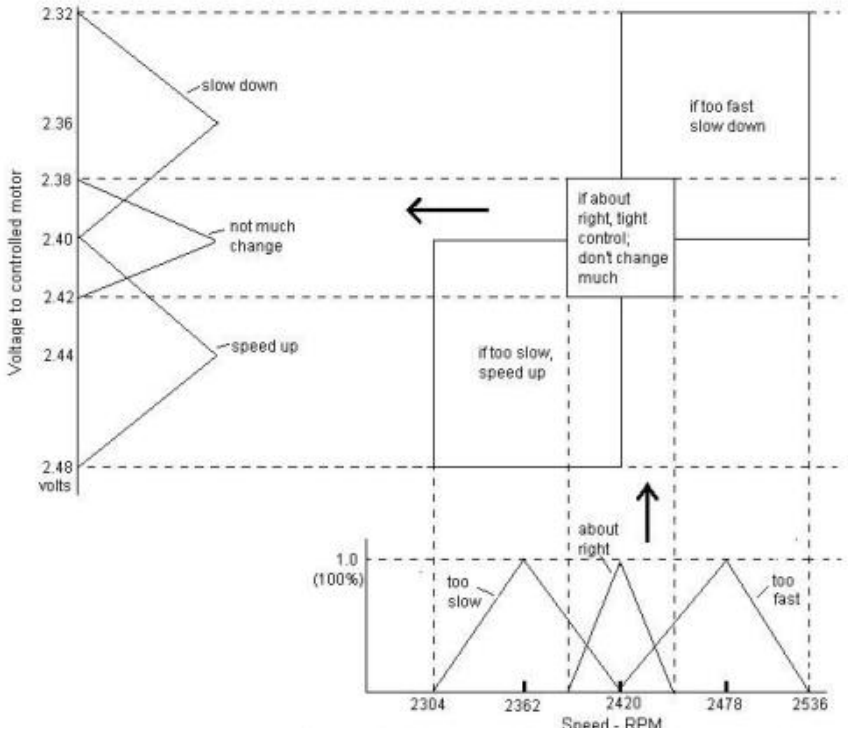

Figure 4. Membership Functions (I/P: Speed O/P: Voltage)

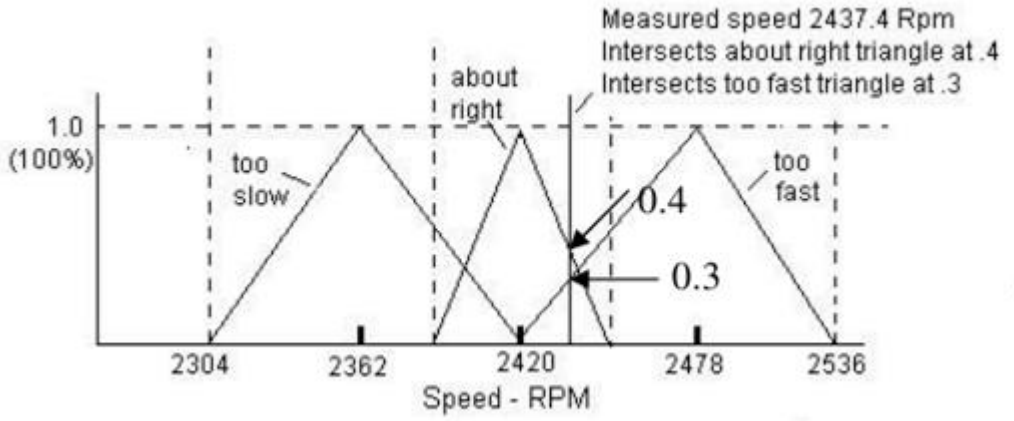

Figure 5: Speed above set point (set point=2420rpm)

3. Solve the same using Fuzzy Logic tool box in Matlab.
   - Steps
     1. Draw the appropriate membership functions using "Membership Function Editor"
        ~ Start the Matlab and type " Fuzzy" in command Prompt to open Fuzzy Interference System (FIS) Editor window
        ~ Input: Speed
        ~ Output: Voltage
     2. Use "Rule Editor" to create rules(Hints: three rules)
        ~ Select New " FIS -> Mamdani" from the File Menu
     3. Save the file as "FZ_Motor.fis" and run following commands to calculate appropriate in put voltage
        » fis = readfis(' FZ_Motor ')
        » out=evalfis(2437.4,fis)

Note: You have to submit Matlab files along with snapshots of each below (e.g. Figure 1, 2, and 3).

~ Input Membership Function
~ Output Membership Function
~ Rule Editor

Submission

You can submit a single ZIP file as e15XXXlab01.zip including all:

~ A PDF file with rule base, tables, numerical calculations, graphical calculations, results, and snapshots of Matlab simulation as indicated in the labs, any descriptions, etc.

~ MatLab Files

~ All images (graphical calculations and outputs from MatLab)

Note: XXX indicates your registration number in all cases. **Those who have plagiarized content will be heavily penalized.**