

AISC

Class: B.E COMP

Experiment 07: Apply concepts of Fuzzy to develop a Fuzzy Controller system.

Learning Objective:

Basic Experiments

Solve a reasoning problem using unification

Tools: Python

Theory:

Fuzzy logic is applied with great success in various control application. Almost all the consumer products have fuzzy control. Some of the examples include controlling your room temperature with the help of air-conditioner, anti-braking system used in vehicles, control on traffic lights, washing machines, large economic systems, etc.

Why Use Fuzzy Logic in Control Systems

A control system is an arrangement of physical components designed to alter another physical system so that this system exhibits certain desired characteristics. Following are some reasons of using Fuzzy Logic in Control Systems –

While applying traditional control, one needs to know about the model and the objective function formulated in precise terms. This makes it very difficult to apply in many cases.

By applying fuzzy logic for control we can utilize the human expertise and experience for designing a controller.

The fuzzy control rules, basically the IF-THEN rules, can be best utilized in designing a controller.

Assumptions in Fuzzy Logic Control (FLC) Design

While designing fuzzy control system, the following six basic assumptions should be made –

The plant is observable and controllable – It must be assumed that the input, output as well as state variables are available for observation and controlling purpose.

Existence of a knowledge body – It must be assumed that there exist a knowledge body having linguistic rules and a set of input–output data set from which rules can be extracted.

Existence of solution – It must be assumed that there exists a solution.

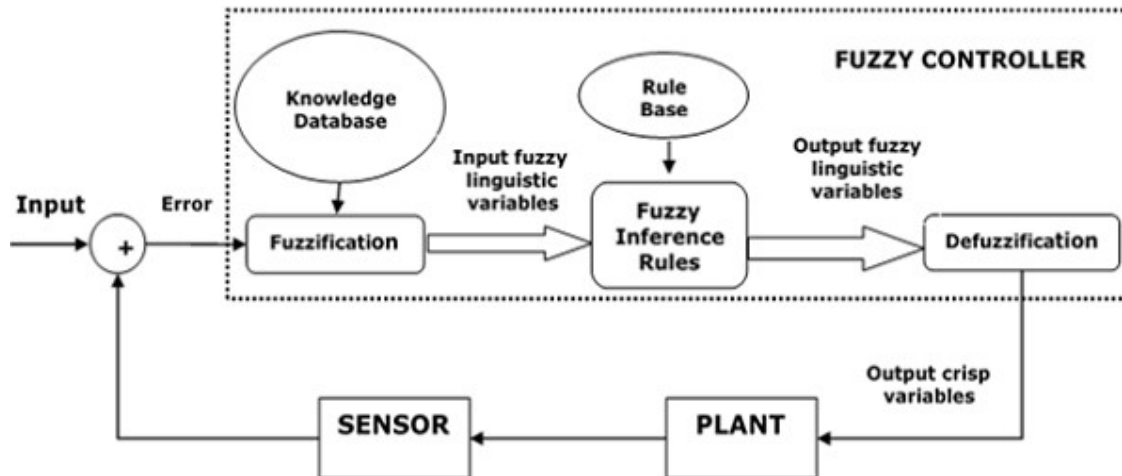
‘Good enough’ solution is enough – The control engineering must look for ‘good enough’ solution rather than an optimum one.

Range of precision – Fuzzy logic controller must be designed within an acceptable range of precision.

Issues regarding stability and optimality – The issues of stability and optimality must be open in designing Fuzzy logic controller rather than addressed explicitly.

Architecture of Fuzzy Logic Control

The following diagram shows the architecture of Fuzzy Logic Control (FLC).



Fuzzy Logic Control Architecture

Major Components of FLC

Followings are the major components of the FLC as shown in the above figure –

Fuzzifier – The role of fuzzifier is to convert the crisp input values into fuzzy values.

Fuzzy Knowledge Base – It stores the knowledge about all the input-output fuzzy relationships. It also has the membership function which defines the input variables to the fuzzy rule base and the output variables to the plant under control.

Fuzzy Rule Base – It stores the knowledge about the operation of the process of domain.

Inference Engine – It acts as a kernel of any FLC. Basically it simulates human decisions by performing approximate reasoning.

Defuzzifier – The role of defuzzifier is to convert the fuzzy values into crisp values getting from fuzzy inference engine.

Steps in Designing FLC

Following are the steps involved in designing FLC –

Identification of variables – Here, the input, output and state variables must be identified of the plant which is under consideration.

Fuzzy subset configuration – The universe of information is divided into number of fuzzy subsets and each subset is assigned a linguistic label. Always make sure that these fuzzy subsets include all the elements of universe.

Obtaining membership function – Now obtain the membership function for each fuzzy subset that we get in the above step.

Fuzzy rule base configuration – Now formulate the fuzzy rule base by assigning relationship between fuzzy input and output.

Fuzzification – The fuzzification process is initiated in this step.

Combining fuzzy outputs – By applying fuzzy approximate reasoning, locate the fuzzy output and merge them.

Defuzzification – Finally, initiate defuzzification process to form a crisp output.

Advantages of Fuzzy Logic Control

Let us now discuss the advantages of Fuzzy Logic Control.

Cheaper – Developing a FLC is comparatively cheaper than developing model based or other controller in terms of performance.

Robust – FLCs are more robust than PID controllers because of their capability to cover a huge range of operating conditions.

Customizable – FLCs are customizable.

Emulate human deductive thinking – Basically FLC is designed to emulate human deductive thinking, the process people use to infer conclusion from what they know.

Reliability – FLC is more reliable than conventional control system.

Efficiency – Fuzzy logic provides more efficiency when applied in control system.

Disadvantages of Fuzzy Logic Control

We will now discuss what are the disadvantages of Fuzzy Logic Control.

Requires lots of data – FLC needs lots of data to be applied.

Useful in case of moderate historical data – FLC is not useful for programs much smaller or larger than historical data.

Needs high human expertise – This is one drawback as the accuracy of the system depends on the knowledge and expertise of human beings.

Needs regular updating of rules – The rules must be updated with time.

Implementation:

SOURCE CODE:

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl

# New Antecedent/Consequent objects hold universe variables and membership
# functions
Food_Quality = ctrl.Antecedent(np.arange(0, 11, 1), 'Food Quality')
Service_Quality = ctrl.Antecedent(np.arange(0, 11, 1), 'Service Quality')
Tip = ctrl.Consequent(np.arange(0, 26, 1), 'Tip')

# Auto-membership function population is possible with .automf(3, 5, or 7)
Food_Quality.automf(3)
Service_Quality.automf(3)

# Custom membership functions can be built interactively with a familiar,
# Pythonic API
Tip['low'] = fuzz.trimf(Tip.universe, [0, 0, 13])
Tip['medium'] = fuzz.trimf(Tip.universe, [0, 13, 25])
Tip['high'] = fuzz.trimf(Tip.universe, [13, 25, 25])

rule1 = ctrl.Rule(Food_Quality['poor'] | Service_Quality['poor'], Tip['low'])
rule2 = ctrl.Rule(Service_Quality['average'], Tip['medium'])
rule3 = ctrl.Rule(Service_Quality['good'] | Food_Quality['good'], Tip['high'])

tipping_ctrl = ctrl.ControlSystem([rule1, rule2, rule3])

tipping = ctrl.ControlSystemSimulation(tipping_ctrl)

# Pass inputs to the ControlSystem using Antecedent labels with Pythonic API
# Note: if you like passing many inputs all at once, use .inputs(dict_of_data)
tipping.input['Food Quality'] = float(input("Rate the quality of Food (1 to 10): "))
tipping.input['Service Quality'] = float(input("Rate the quality of Service (1 to 10): "))
```

Crunch the numbers

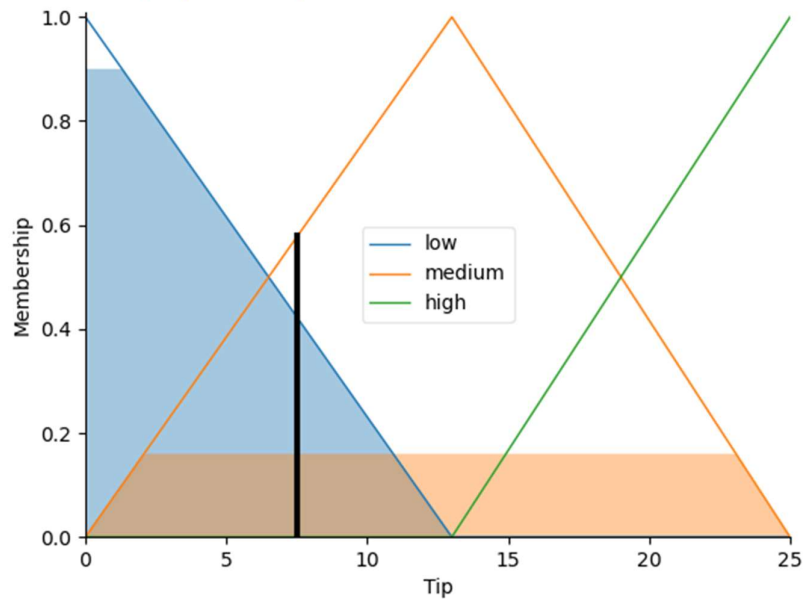
tipping.compute()

print ("Final Tip (0 to 25): ",tipping.output['Tip'])

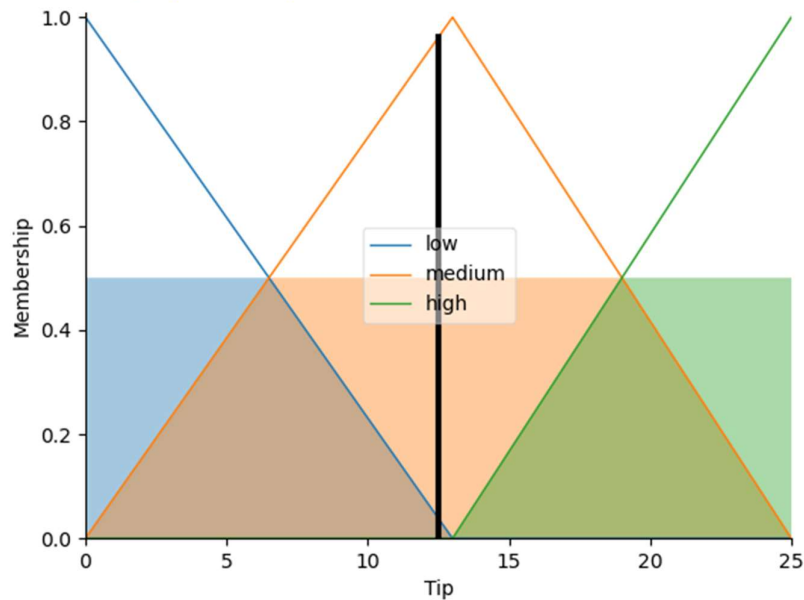
Tip.view(sim=tipping)

OUTPUT:

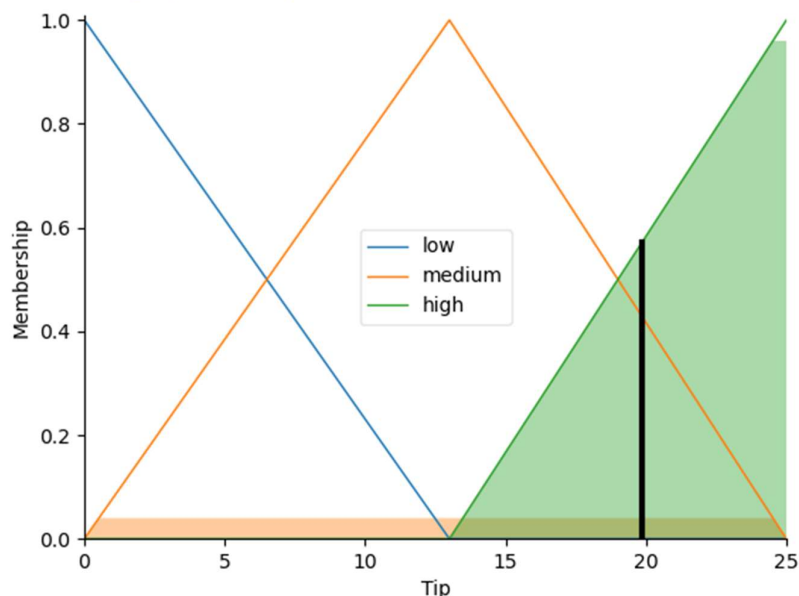
Rate the quality of Food (1 to 10): 0.5
Rate the quality of Service (1 to 10): 0.8
Final Tip (0 to 25): 7.520909051365793



Rate the quality of Food (1 to 10): 2.5
Rate the quality of Service (1 to 10): 7.5
Final Tip (0 to 25): 12.5



Rate the quality of Food (1 to 10): 6.5
Rate the quality of Service (1 to 10): 9.8
Final Tip (0 to 25): 19.847607361963192



Learning Outcomes: Students should have the ability to FLC

LO1: Understand the problem formulation

Course Outcomes: Upon completion of the course students will be able to understand FLC

Conclusion: Hence we have successfully understood the concept of Fuzzy controller system and also implemented the same for Tipping problem at any restaurant.

Viva Questions:

Ques.1 What do you understand by FLC?

Ques. 2. Explain the basic steps in FLC?

Ques. 3. Write types of problem discussed in detail.

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				