Aim – To implement Fuzzy controller for Washing machine.

Theory –

Fuzzy Controller:-

A fuzzy control system is a control system based on fuzzy logic—a mathematical system that analyzes analog input values in terms of logical variables that take on continuous values between 0 and 1, in contrast to classical or digital logic, which operates on discrete values of either 1 or 0.

Fuzzy set:

1. Fuzzy set is a set having degrees of membership between 1 and 0. Fuzzy sets are represented with tilde character (~). For example, Number of cars following traffic signals at a particular time out of all cars present will have membership value between [0,1].
2. Partial membership exists when member of one fuzzy set can also be a part of other fuzzy sets in the same universe.
3. The degree of membership or truth is not same as probability, fuzzy truth represents membership in vaguely defined sets.

Mathematical logic:

A fuzzy set A~ in the universe of discourse, U, can be defined as a set of ordered pairs and it is given by $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in X\}$

When the universe of discourse, U, is discrete and finite, fuzzy set A~ is given by where “n” is a finite value.

$$\tilde{A} = \sum_{i=1}^{n} \frac{\mu_{\tilde{A}}(x_i)}{x_i} = \frac{\mu_{\tilde{A}}(x_1)}{x_1} + \frac{\mu_{\tilde{A}}(x_2)}{x_2} + \ldots + \frac{\mu_{\tilde{A}}(x_n)}{x_n}$$

$$\tilde{A} = \int \frac{\mu_{\tilde{A}}(x)}{x}$$

Fuzzy sets also satisfy every property of classical sets.

Operations:

Common Operations on fuzzy sets: Given two Fuzzy sets A~ and B~

1. Union: Fuzzy set C~ is union of Fuzzy sets A~ and B~

$$\tilde{C} = \tilde{A} \cup \tilde{B}, \quad \mu_{\tilde{C}}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$$

2. Intersection: Fuzzy set D~ is intersection of Fuzzy sets A~ and B~ :

$$\tilde{D} = \tilde{A} \cap \tilde{B} \qquad \mu_{\tilde{D}}(x) = \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$$

3. Complement: Fuzzy set E~ is complement of Fuzzy set A~:

$$\tilde{E} = \mathbb{C}_{\tilde{A}} X \qquad \mu_{\tilde{E}}(x) = 1 - \mu_{\tilde{A}}(x)$$

Code Working Explaination:-

Prerequisites

When it comes to computations NumPy is the fundamental package in terms of performing mathemetical computations and n-dimensional matrix manipulations.

First Step is to install NumPy using pip:

pip install numpy

This will install latest running version of NumPy in your system.

Latest version of NumPy: 1.14

scikit-fuzzy is fuzzy-logic toolkit provided by scipy. It provides a robust toolkit for performing fuzzy-logic algorithms.

Make sure if you're using NumPy then version must be >= 1.6 Otherwise scikit-fuzzy won't allow the installation. Latest stable version is avaialble on PyPi!

Again using pip we will install the latest version of scikit-fuzzy:

pip install -U scikit-fuzzy

If you already have scikit-fuzzy installed then this will upgrade it to a more stable version.

Rule Application

Using scikit-fuzzy we will generate a Control System that will estimate how long will it take to wash one load of clothes?Our inputs will be known as Antecednets and Outputs will be known as Consequents in a scikit-fuzzy controller.

Antecednets(Inputs):

type_of_dirtiness:

Universe (ie, crisp value range): Determine type of dirtiness in terms of percentage 1 to 100.Fuzzy set (ie, fuzzy value range): NonFat, medium, Fat

degree_of_dirtiness:

Universe (ie, crisp value range): Determine the degree of dirtiness in terms of percentage 1 to 100

Fuzzy set (ie, fuzzy value range): Low,Medium,Fat

Consequents(Outputs):

wash_time:

Universe: According to type_of_dirtiness and degree_of_dirtiness program will determine how long it would take to wash one load of clothes. (Output is generated in the format of minutes between (1 to 60))

Fuzzy set (ie, fuzzy value range): VeryShort,Short,Medium,Long,VeryLong

will upgrade it to a more stable version.

rule1 = ctrl.Rule(degree_dirt['High'] | type_dirt['Fat'], wash_time['VeryLong'])

rule2 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['Fat'], wash_time['long'])

rule3 = ctrl.Rule(degree_dirt['Low'] | type_dirt['Fat'], wash_time['long'])

rule4 = ctrl.Rule(degree_dirt['High'] | type_dirt['Medium'], wash_time['long'])

rule5 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['Medium'], wash_time['medium'])

rule6 = ctrl.Rule(degree_dirt['Low'] | type_dirt['Medium'], wash_time['medium'])

rule7 = ctrl.Rule(degree_dirt['High'] | type_dirt['NonFat'], wash_time['medium'])

rule8 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['NonFat'], wash_time['short'])

rule9 = ctrl.Rule(degree_dirt['Low'] | type_dirt['NonFat'], wash_time['very_short'])

Code –

Main.py

```
from fuzzify import *

def compute_washing_parameters(type_of_dirt,degree_of_dirt):

    if type_of_dirt < 0.0 or type_of_dirt > 100.0:

        raise Exception("Invalid Type of Dirtiness: %lf" %type_of_dirt)

    if degree_of_dirt < 0.0 or type_of_dirt > 100.0:

                raise  Exception("Invalid  Degree  of  Dirtiness:  %lf"
%degree_of_dirt)

    type_fuzzy = fuzzify_laundry(type_of_dirt,degree_of_dirt)

    return type_fuzzy

if __name__ == "__main__":

    type_of_dirt = float(input("Enter Type of Dirtiness [0-100]"))

    degree_of_dirt = float(input("Enter Degree of Dirtiness [0-100]"))
```

```
                                        washing_parameters          =
compute_washing_parameters(type_of_dirt,degree_of_dirt)

    print(washing_parameters)

#pip install -U scikit-fuzzy
```

fuzzify.py

```
from skfuzzy import control as ctrl

import skfuzzy as fuzz

import numpy as np

class washing_machine:

    degree_dirt = ctrl.Antecedent(np.arange(0, 101, 1), 'degree_dirt')

    type_dirt = ctrl.Antecedent(np.arange(0, 101, 1), 'type_dirt')

    wash_time = ctrl.Consequent(np.arange(0, 61, 1), 'wash_time')

    degree_names = ['Low', 'Medium', 'High']

    type_names = ['NonFat', 'Medium', 'Fat']

    #Outputing them into auto-membership functions

    degree_dirt.automf(names=degree_names)

    type_dirt.automf(names=type_names)

    # Washing Time Universe

      wash_time['very_short'] = fuzz.trimf(wash_time.universe, [0, 8,
12])

    wash_time['short'] = fuzz.trimf(wash_time.universe, [8, 12, 20])

    wash_time['medium'] = fuzz.trimf(wash_time.universe, [12, 20, 40])

    wash_time['long'] = fuzz.trimf(wash_time.universe, [20, 40, 60])

      wash_time['VeryLong'] = fuzz.trimf(wash_time.universe, [40, 60,
60])

    # Rule Application

         rule1 = ctrl.Rule(degree_dirt['High'] | type_dirt['Fat'],
wash_time['VeryLong'])

        rule2 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['Fat'],
wash_time['long'])
```

```
        rule3 = ctrl.Rule(degree_dirt['Low'] | type_dirt['Fat'],
wash_time['long'])

       rule4 = ctrl.Rule(degree_dirt['High'] | type_dirt['Medium'],
wash_time['long'])

      rule5 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['Medium'],
wash_time['medium'])

       rule6 = ctrl.Rule(degree_dirt['Low'] | type_dirt['Medium'],
wash_time['medium'])

       rule7 = ctrl.Rule(degree_dirt['High'] | type_dirt['NonFat'],
wash_time['medium'])

      rule8 = ctrl.Rule(degree_dirt['Medium'] | type_dirt['NonFat'],
wash_time['short'])

       rule9 = ctrl.Rule(degree_dirt['Low'] | type_dirt['NonFat'],
wash_time['very_short'])

    # Washing Control Simulation

      washing_ctrl = ctrl.ControlSystem([rule1, rule2, rule3, rule4,
rule5, rule6, rule7, rule8, rule9])

    washing = ctrl.ControlSystemSimulation(washing_ctrl)

def fuzzify_laundry(fuzz_type,fuzz_degree):

    washing_machine.washing.input['type_dirt'] = fuzz_type

    washing_machine.washing.input['degree_dirt'] = fuzz_degree

    washing_machine.washing.compute()

    washing_machine.wash_time.view(sim=washing_machine.washing)

    return washing_machine.washing.output['wash_time']
```

Output –

```
C:\Users\nick_pc\Desktop\CI PRACS\fuzzycontroller>python main.py
Enter Type of Dirtiness [0-100]60
Enter Degree of Dirtiness [0-100]50
31.660678642714554

C:\Users\nick_pc\Desktop\CI PRACS\fuzzycontroller>
```

Conclusion – In this experiment I learnt what are fuzzy controller, how it is used to, I used my own dataset for the experiment, I was able to perform various operations on fuzzy sets. and apply rules on the controller and learned defuzzification too.