Experiment 5

Aim: implement fuzzy controller for washing machine system using python AI

Name: Sejal Santosh Jadhav

RollNo : 2102617

class : B.E

Branch: IT

Batch: T2

Subject: DS LAB

DOP	DOS	mark's	sign

Experiment 5

Page No.:

Date :

Aim:- Implement fuzzy controller for washing machine system using python API.

Theory:

1] What are fuzzy controller?

→ - fuzzy controller are a type of control system used in engineering and various application to manage complex and uncertain situations. They are part of the broader field of fuzzy logic. Which is a mathematical framework designed to model and handle uncertainty in decision-making process.

- fuzzy controller are particularly useful when dealing with systems that have impression or ambiguous input-output relationships or when difficult to model the system using traditional mathematical methods.

- They have been applied in various field in various field, including engineering, robotics, control systems and decision making process where human expertise intuition play a significant role.

2] Explain different stages in fuzzy control system?

→ ① fuzzification:

- This stage involves converting crisp numerical inputs like sensor reading into fuzzy linguistic variables. linguistic terms are used to describe the inputs characteristics, such as "low", "midium", "high"

- fuzzyfication is achived by mapping the crisp input values to membership degrees in the liguistic terms using membership functions

② fuzzy rule base : Contain a set of If-then rules that expresss the control strategy. Each rule consist of a combination of input conditions and an associated output actions

③ Interface Engine : The interface engine evaluates the fuzzy rule based on the fuzzified input values. It determines the degree to which each rule is satisfied by current input values

④ Defuzzification :
- The aggregated fuzzy outputs need to be converted back into crisp control action that can be applied to the system

5] Aggregation :
In this stage the output of the individual rules are combined to form an overall fuzzy output

3] What are mamdani type fuzzy controller ?

→ - The mamdane type fuzzy controller is one of the most common and well-known type of fuzzy controller
- It was introduced by Ebrahim mamdani in the 1970's and has bee widely used in various applications for its simplicity and effectiveness in dealing with uncertain and imprecision information
- The mamdani fuzzy controller follows a specific structure and methodology for making control decision based on fuzzy logic
- mamdani type of fuzzy controllers are particularly suitable for system where human expertise & intuition play a significant role in decision making

4] with arbitary example explain creating mamdani fuzzy control

→ Step1 : Define ligusitis variable and membership functions.
1] Temprature (input) : • linguistic terms: low, medium, high
• membership functions : triangular or trapezial membership.
2] Humidity (input) : • linguistic terms : low, medium, high
• membership functions : Triangular or trapezial membership
Step2 : Define fuzzy rules
1] If temprature is low and humidity is low then low, if medium fan speed medium, if high fan medium
2] If temprature is medium and humidity is low then fan speed medium, if medium fanspeed high
3] If temprature is high and humidity is low the fanspeed is medium, if medium fan speed high
Step3 : fuzzyfication : if ~~from~~ temp is 28°C & humidity lvl is 70%
• Temp : medium (0.6), • Humidity : high (0.7)

Step4 : Inteference engine
• Temprature match degree : 0.6 (medium)
• Humidity match degree - 0.7 (high)
• Degree of activation - min (0.6, 0.7) = 0.6

Step5 : Aggregation → Activation of 0.6 has activation 0.4
the aggregated output for "High" fan spee be max (0.6, 0.4) = 0.6

Step6 : Defuzzification → convert aggregated fuzzy output into crisp value for fan speed using defuzzification.

Step7 : output scaling and activation
Scale the crisp control output to appropriate range for the fan's control mechanism

code.

pip install simpful

install Simpful library using pip, simpful provides a lightweight API that allow to intuitively define fuzzy sets and fuzzy rules and to perform fuzzy inference

from simpful import *
import function/object from simpful library

FS = fuzzy sy FS = FuzzySystem()
create a fuzzy system object

I1 = FuzzySet(function = Triangular_MF(a=0, b=0, c=50), term="small")
I2 = FuzzySet(function = Triangular_MF(a=0, b=50, c=100), term="medium")
I3 = FuzzySet(function = Triangular_MF(a=50, b=100, c=100), term="large")
# Define fuzzy set and linguistic variables
# parameters
# a - universe of discourse cordinate of the leftmost vertex
# b, c - universe of discourse coordinate of the upper, rightmost vertex
# term : string represent the linguistic term to be associated to fuzzy set

FS.add_linguistic_variable("dirt", LinguisticVariable[I1, I2, I3], concept = "amount dirt", universe_of_discourse=[0,100]

FS.add_linguistic_variable("grease", LinguisticVariable[I1, I2, I3], concept = "amount grease", universe_of_discourse[0,100]
Adds a new linguistic variables to the fuzzy system amount dirt and grease
universe of discourse : A list of element specifying min and max of the univers of discourse Here 0 is min and 100 is max

#concept: a string providing a brief description of the concept represented by the linguistic variable

Define fuzzy rules for fuzzy system washing machine controller

R1 = 'IF (dirt IS small) AND (grease IS small) THEN (washtime IS very short)'

R2 = "IF (dirt IS small) AND (grease IS ~~small~~ medium) THEN (washtime IS medium)"

R3 = "IF (dirt IS small) AND (grease IS large) THEN (washtime IS long)"

R4 = "IF (dirt IS ~~small~~ medium) AND (grease IS small) THEN (washtime IS short)"

R5 = "IF (dirt IS medium) AND (grease IS medium) THEN (washtime IS medium)"

R6 = "IF (dirt IS ~~medium~~ medium) AND (grease IS large) THEN (washtime IS long)"

R7 = 'IF (dirt IS ~~medium~~ large) AND (grease IS small) THEN (washtime IS medium)'

R8 = "IF (dirt IS large) AND (grease IS medium) THEN (washtime IS long)"

R9 = 'IF (dirt IS large) AND (grease IS large) THEN (washtime IS very long)'

FS.add_rules([R1, R2, R3, R4, R5, R6, R7, R8, R9])

add Rules to fuzzy system

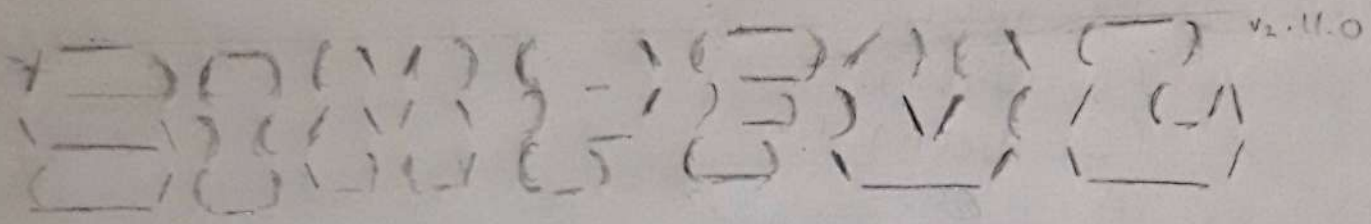
set input values

FS.set_variable ("dirt", 60)

FS.set_variable ("grease", 60)

perform mamdani inference and print output of given print (FS.Mamdani_inference (["washtime"]))

Output



Output

{'washTime': 30.581630542394986}