|  |  |
| --- | --- |
| **EX.NO : 12** | **FUZZY INFERENCE SYSTEM** |
| **DATE : 22.05.2024** |

**AIM:**

To implement a fuzzy inference system in Python using the scikit-fuzzy library. Here you might create a fuzzy inference system for controlling the speed of a car based on the distance to an obstacle.

**ALGORITHM:**

Step 1: Define the input and output variables and their linguistic terms.

Step 2: Define the membership functions for each linguistic term.

Step 3: Define the fuzzy rules based on the input and output variables.

Step 4: Fuzzify the input variables using the defined membership functions.

Step 5: Apply the fuzzy operator (AND or OR) in the antecedent of the rules.

Step 6: Calculate the implication from the antecedent to the consequent of the rules.

Step 7: Aggregate the consequents of all the rules.

Step 8: Defuzzify the aggregated output to get a crisp output.

Step 9: Choose an appropriate defuzzification method (centroid, bisector, middle of maximum, largest of maximum, or smallest of maximum).

Step 10: Implement the algorithm using a programming language or a fuzzy logic toolbox.

**PROGRAM:**

import numpy as np

import skfuzzy as fuzz

from skfuzzy import control as ctrl

*# Create fuzzy variables*

distance = ctrl.Antecedent(np.arange(0, 11, 1), 'distance')

speed = ctrl.Consequent(np.arange(0, 101, 1), 'speed')

*# Define membership functions for distance*

distance['near'] = fuzz.trimf(distance.universe, [0, 0, 5])

distance['medium'] = fuzz.trimf(distance.universe, [0, 5, 10])

distance['far'] = fuzz.trimf(distance.universe, [5, 10, 10])

*# Define membership functions for speed*

speed['slow'] = fuzz.trimf(speed.universe, [0, 0, 50])

speed['medium'] = fuzz.trimf(speed.universe, [0, 50, 100])

speed['fast'] = fuzz.trimf(speed.universe, [50, 100, 100])

*# Define rules*

rule1 = ctrl.Rule(distance['near'], speed['slow'])

rule2 = ctrl.Rule(distance['medium'], speed['medium'])

rule3 = ctrl.Rule(distance['far'], speed['fast'])

*# Create the control system*

speed\_ctrl = ctrl.ControlSystem([rule1, rule2, rule3])

car\_speed = ctrl.ControlSystemSimulation(speed\_ctrl)

*# Input distance and compute speed*

car\_speed.input['distance'] = 7

car\_speed.compute()

*# Print the computed speed*

print("Computed speed:", car\_speed.output['speed'])

**OUTPUT:**



**RESULT:**

The implement a fuzzy inference system in Python using the scikit-fuzzy library is executed successfully.