

**University Of Engineering And Technology, Mardan**

**Knowledge Representation And Reasoning**

**Assignment No 3**

**Title:** Implementation of an Intelligent System

Using Knowledge Representation Techniques

**Department Of Computer Science**

**Section (Artificial Intelligence)**

**Submitting To**

**Mr Bilal Khan**

**By**

**Hassan Khitab #22MDBCS297**

**Submission Deadline:** 31st December, 2024

**Introduction:**

The Intelligent Room Temperature Controller is a system designed using fuzzy logic, a powerful

knowledge representation technique. This system dynamically adjusts the room temperature by

controlling the heater level based on the input parameters: temperature and humidity. The goal of

the system is to mimic human reasoning for maintaining an optimal and comfortable environment.

**Objectives:**

1. To designan intelligent system that uses fuzzy logic for reasoning.

2. To implement membership functions and fuzzy rules for effective control.

3. To create a user-friendly interface for easy interaction with the system.

4. To simulate real-life scenarios and provide actionable outputs.

**System Overview:**

**3.1. Inputs**

- Temperature (°C): Ranges from 0°C to 40°C.

- Humidity (%): Ranges from 0% to 100%.

**3.2. Output**

- Heater Level (%): Determines the required intensity of the heater, ranging from 0% (off) to 100%

(full power).

**3.3. Fuzzy Logic Implementation**

The system uses fuzzy logic to process the input variables (temperature and humidity) and generate

the appropriate output (heater level). This is achieved using:

- Membership functions for representing fuzzy sets.

- Fuzzy rules to define the relationship between inputs and the output.

- A defuzzification process to convert fuzzy results into a crisp value.

1. **Implementation Details**

**4.1. Tools and Technologies**

- Programming Language: Python

- Libraries:

- numpy for numerical operations.

- scikit-fuzzy for fuzzy logic implementation.

- tkinter for the graphical user interface (GUI).

**4.2. Steps in Development**

1. Define Fuzzy Variables:

- Temperature: Divided into "Cold," "Warm," and "Hot."

- Humidity: Divided into "Low," "Medium," and "High."

- Heater Level: Divided into "Low," "Medium," and "High."

2. Membership Functions:

- Represented as triangular functions to define the degree of membership for each variable.

3. Fuzzy Rules:

- Rule 1: IF Temperature is Cold AND Humidity is Low THEN Heater Level is High.

- Rule 2: IF Temperature is Cold AND Humidity is Medium THEN Heater Level is Medium.

- Rule 3: IF Temperature is Warm THEN Heater Level is Low.

- Rule 4: IF Temperature is Hot THEN Heater Level is Low.

4. Defuzzification:

- Converts fuzzy output into a crisp value for actionable results.

5. GUI Implementation:

- Created a user-friendly interface using tkinter for entering temperature and humidity values.

- Displayed the calculated heater level dynamically.

**5. Example Simulation**

**Input:**

- Temperature: 18°C

- Humidity: 40%

**Output:**

- Heater Level: 85.00%

The system successfully calculated a high heater level due to low temperature and moderate

humidity, providing a realistic response to the input scenario.

**6. User Guide**

**6.1. System Requirements**

- Python 3.8 or higher.

- Required Libraries: scikit-fuzzy, tkinter, numpy.

**6.2. Steps to Run the System**

1. Install required libraries using:

**bash**

pip install scikit-fuzzy

2. Run the Python script.

3. Enter the temperature (°C) and humidity (%) values in the GUI.

4. Click the "Calculate Heater Level" button to see the result.

**6.3. Example Usage**

- Input: Temperature = 25°C, Humidity = 50%.

- Output: Heater Level = 30.00%.

**7. Conclusion**

The Intelligent Room Temperature Controller demonstrates the effective use of fuzzy logic for

reasoning and decision-making. By integrating a GUI, the system ensures usability and practical

application. This project highlights the importance of knowledge representation techniques in

building real-world intelligent systems.

**8. Future Enhancements**

1. Incorporate additional parameters like time of day or occupancy for enhanced control.

2. Use advanced machine learning models to fine-tune fuzzy rules dynamically.

3. Extend the system to control cooling devices like air conditioners.

**9. References**

1. "Fuzzy Logic with Engineering Applications" by Timothy J. Ross.

2. Scikit-Fuzzy Documentation: https://pythonhosted.org/scikit-fuzzy/

3. Tkinter Documentation: https://docs.python.org/3/library/tkinter.html