# Fuzzy Logic for HVAC Systems

Troy Kleger, *Student, Mercer University*

***Abstract*—This paper will present a fuzzy logic model created to handle HVAC controls for a room. The scope of this paper is small as this is a student paper for a course. The model will attempt to optimize the temperature and humidity of a room based on its environment. Additionally, this paper can serve as a small introduction to fuzzy logic for those who have not previously worked with it. As you will discover later in this paper, not all of the results are good; however, the model presented has specific results that are more than satisfactory. The results that are satisfactory are used to show that the fuzzy logic model is partially working by optimizing the temperature.**

***Index Terms*—fuzzy logic, centroid method, defuzzification, gaussian, mamdani**

## I. Introduction

FUZZY logic was a concept created in 1965 by Lofti Zadeh, in the paper ”Fuzzy Sets” [1]. The concept of ”fuzzy” in terms of fuzzy logic refers to a way of answering a yes/no question with not simply yes or no, but rather with an answer anywhere from yes to no. In fuzzy logic, we apply this concept to sets of statuses. Take temperature, for example, based on the temperature, there will be a degree of membership for ”warm” that is somewhere between 0 and 1. In most cases, multiple sets will be used for a single argument. In this paper, for example, the temperature has four of these sets: freezing, cold, warm, and hot. Once a measured temperature is assigned a degree of membership for all of these sets, rules are made that will use these degrees of membership to get a degree of membership for some output variable, such as power amount. This paper will showcase the various elements of fuzzy logic by showing the elements in the model created.

## II. The System

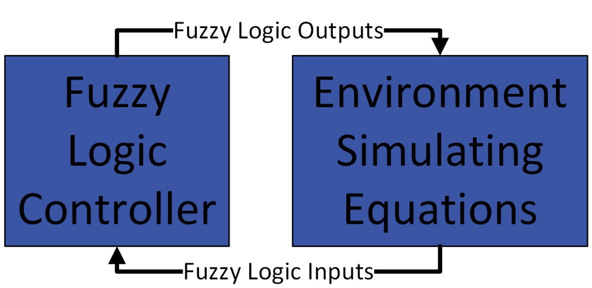


Fig. 1: Simple System Design for the Paper

The system outlined in Fig. 1 is utilized to evaluate the model. The way it works is as follows.

1) The fuzzy logic controller will use the input values to calculate output values.
2) The output values are sent to the environment simulator.
3) The environment simulator uses thermodynamic equations to calculate fuzzy logic inputs.
4) The new fuzzy logic inputs are sent back to the fuzzy logic controller.
5) Repeat.

Data is collected by appending the computed fuzzy logic inputs to a list every time the environment simulator sends them back.

## III. Environment

The papers [2] and [3] were crucial in helping to create these equations. For temperature, we use Fourier's Law of Conduction combined with the heat transfer equation in order to calculate the new temperature. Note the following variables, constants, and equations:

$k$ : Thermal conductivity. Constant $0.03\ W/mK$
$L$ : Wall thickness. Constant $0.2\ m$
$c_{pa}$ : Specific heat of dry air. Constant $1005\ J/kg \cdot K$
$c_{pw}$ : Specific heat of water vapor. Constant $1860\ J/kg \cdot K$
$A$ : Surface area that heat can go through to go in/out the room. Constant $256\ m^2$
$s$ : The number of seconds that a call to the environment simulator will simulate.
$H_{in}$ : Humidity in the room.
$T_{in}$ : Temperature inside the room.
$T_{out}$ : Temperature outside the room.
$w$ : Wall heat loss factor. This is set at the start of a simulation. By default it is one. It simply serves as an easy way to tweak the environment when running the code to get slightly different results.

$$c_p = c_{pa} + \frac{H_{in}}{100} c_{pw} \qquad Q_{loss} = \frac{wkA(T_{in} - T_{out})}{L}$$

$$P_{net} = P_{AC/Heater} - Q_{loss} \qquad \Delta T_{in} = \frac{P_{net} s}{m_{air} c_p}$$

The humidity is much simpler, but less accurate. Note the following variables, constants, and equation.

$Y$ : Max power of the humidifier/dehumidifier. Constant $5\ \%$
$C$ : Humidifier/Dehumidifier control percentage
$s$, $H_{in}$ : Both same as use for temperature.
$H_{out}$ : Humidity outside of the room.
$H_{new}$ : New humidity calculated for the room.

$$\Delta H_{in} = \frac{s}{3600}\left(\frac{C}{100}Y + 0.1(H_{in} - H_{out})\right)$$

## IV. Fuzzy Logic Variables

### A. Inputs

The model uses three fuzzy logic inputs, room temperature (°C), change in room temperature (Δ°C), and humidity (%).

Temperature and humidity are rather self-explanatory as to why we need them; they are the variables that we are trying to keep as close as possible to a desired value. Change in temperature, however, is used largely to help limit overshooting the optimal value as it is being approached ( More on this in section VII ). The intervals for these variables are as follows:

- Temperature : [-50,50]
- Humidity : [0,100]
- Change in temperature: [-10,10]

### *B. Outputs*

The model uses two fuzzy logic outputs, AC/Heater power (%) and Humidifier/Dehumidifier power (%). Both of these can have values anywhere in the interval $[-100, 100]$. For the AC/Heater power, a positive value will push heat into the room, and a negative value will pull heat from the room. For the Humidifier/Dehumidifier power, a positive power will cause humidity to be added to the room, while a negative power will cause humidity to be pulled out of the room. While the theoretical machines that would be used to make these are certainly not normal, using a single power for positives and negatives has greatly simplified the model.

## V. Fuzzy Sets

The fuzzy sets used in the model are rather simple. All inputs and outputs have a high set, a medium set, and a low set, except for temperature. Temperature has four sets to make the model more responsive. If more sets are added, the model becomes more complex. If it becomes more complex, it may give better results, depending on the changes made.

## VI. Membership Functions

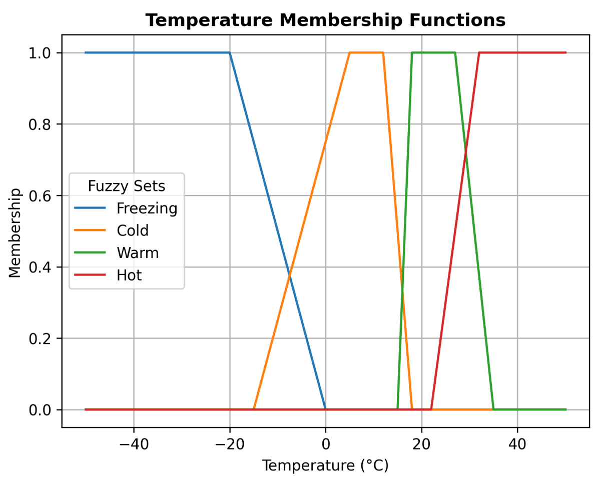


Fig. 2: Temperature Membership Functions

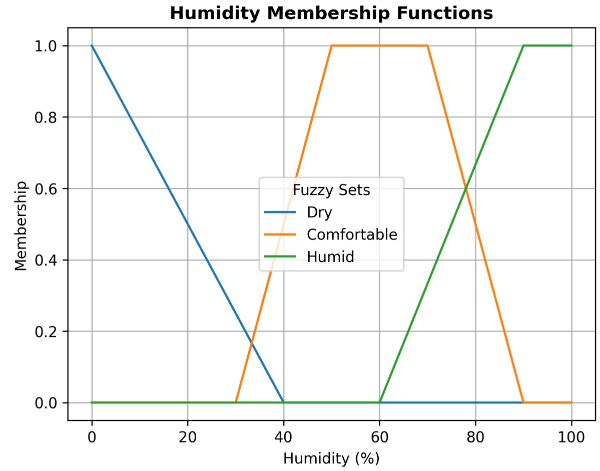


Fig. 3: Humidity Membership Functions

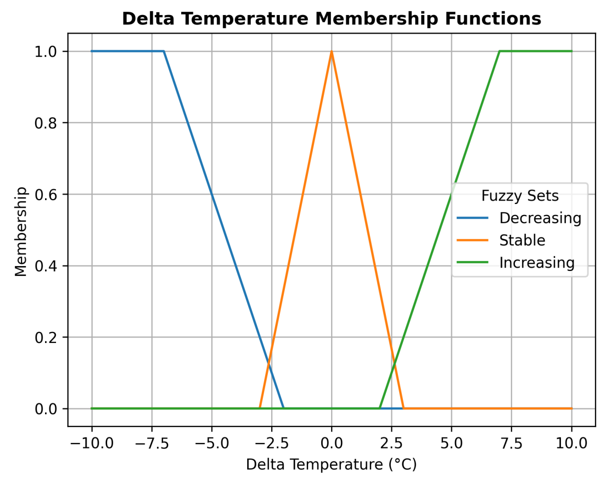


Fig. 4: Delta Temperature Membership Functions

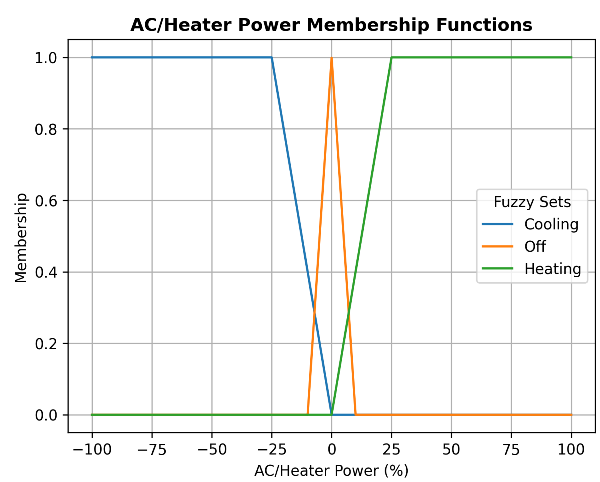


Fig. 5: AC/Heater Power Membership Functions

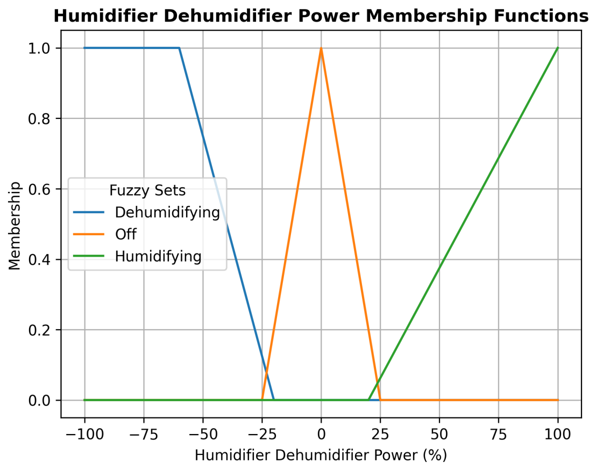


Fig. 6: Humidifier/Dehumidifier Power Membership Functions

The membership functions used are all made with trapezoids or triangles. Some, like the freezing set in Fig. 2, are flipped upside down. The library used allowed the use of a Gaussian curve shape, but this was not used in order to simplify the functions. The functions were easy to make for the inputs; however, the outputs were much trickier. The membership functions for the outputs were tweaked a lot to stop values from exploding. There are a few problems with creating these functions.

- Room ACs and heaters do not typically have the ability to operate at such precise percentages of their power. By forcing this condition, the model unfortunately decreases how realistic it is.
- Deciding where a set's degree of membership had a value of one was rather tricky because, unlike temperature and humidity, there is no common knowledge that gives rough ranges. Temperature and humidity membership functions are counterexamples as they were much easier to create due to common knowledge giving rough ranges for what temperatures are cold, what humidities are humid, etc.
- The small areas on the x-axis where membership functions meet are incredibly important. If there is no overlap, we are essentially allowing one of the fuzzy sets to dominate. For example, when there was no overlap in Fig. 6, the crisp value in the output was very often zero.

## VII. Fuzzy Rules

The fuzzy rules used in this paper are very simple. Most consist of just one of the three inputs. The only exceptions are the rules that take a temperature set and a change in temperature set. One such rule is "If the temperature is cold and the change in temperature is increasing, the AC/Heater is off". This rule is helpful since it helps slow the temperature rise down, so it is not at full blast when it crosses the ideal temperature.

## VIII. Data

A 14-day hourly weather forecast for Macon, GA, starting on October 11, 2025, was used. The only data that was used was the hourly temperature and humidity. A cubic spline was created for both variables to allow access to the values at any time, not just on an hour mark. The temperature spline can be seen in Fig. 7. Note that the dots on the graph are the measured temperatures. This spline was created with just the temperature measurements, allowing access to the temperature at any time, not just on an hour mark. The data came from an API call made to weatherapi.com

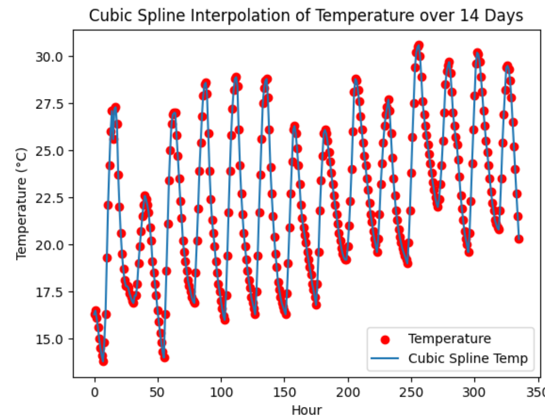


Fig. 7: Cubic spline interpolation of gathered temperature.

## IX. Results

### A. Parameters

The results in this section were computed with the following parameters.

- Step Size: 300 seconds. This is how long each call to the environment simulator simulates. Represented by $s$ in the environment simulating equations. This has a negative linear relationship with how long the code will take to finish the simulation
- Max AC/Heater Power: 3000 $W$. This is higher than is realistic for the situation, but it allows for a better demonstration of the fuzzy logic.
- Simulated time: 72 hours. This was chosen for two reasons. Firstly, there is a positive linear relationship with how long the code takes to finish. Secondly, the data would go all over the place a little while after 72 hours.

### B. Temperature

Overall, the results for the temperature were really good.

- If you look at Fig. 8, you will notice that the temperature quickly shoots up to get closer to a temperature where the degree of membership in the temperature's warm fuzzy set is one. After the initial jump in temperature, the temperature is staying within 2.5°C of 20°C. This demonstrates that the fuzzy logic is able to keep the temperature within an acceptable range.
- The peaks and valleys in the temperature come a little bit after the same peak or valley occurs in the outdoor temperature. This demonstrates the reaction to the outside environment.

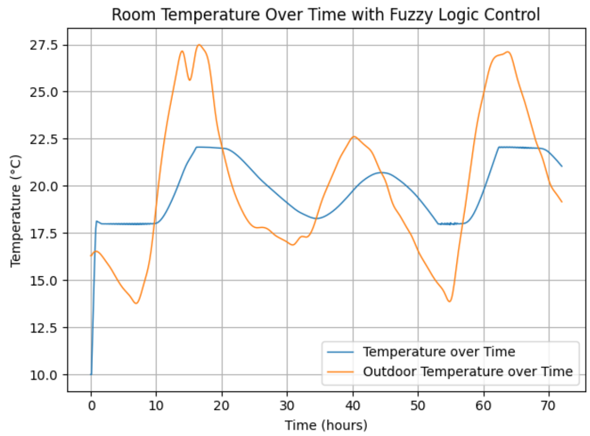


Fig. 8: Temperature over time.

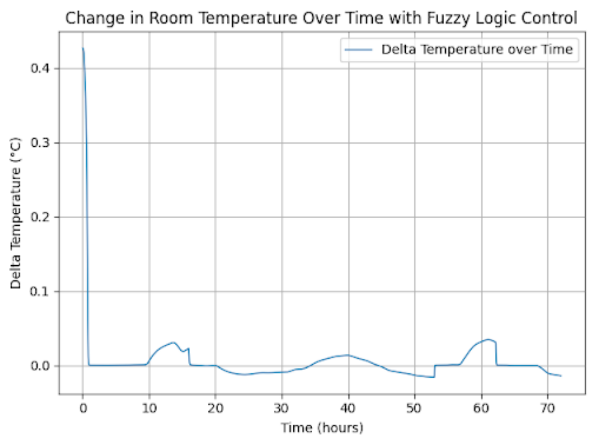


Fig. 9: Change in Temperature over time.

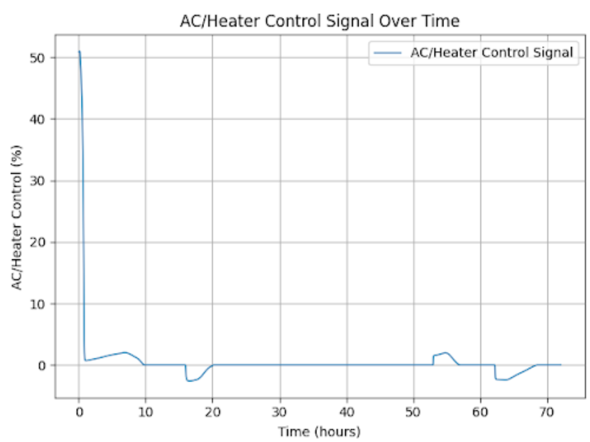


Fig. 10: AC/Heater Power over time.

### C. *AC/Heater Power*

Notice Figs 8 and 10. When analyzing both graphs at the same time, we make some great discoveries.

- The power is very high at first and quickly drops. This is happening since the initial temperature is far from the desired range.
- The power is 0 most of the time, leading to massive savings in energy. It only leaves zero when the temperature is far from the desired range.
- Where there is a sharp turn in the temperature over time graph, the AC/Heater power over time graph either becomes zero again or stops being zero at that exact moment. This shows the effect that turning on and off the AC/Heater has on the temperature.
- The AC/Heater power goes back to zero when the indoor and outdoor temperatures are equal.

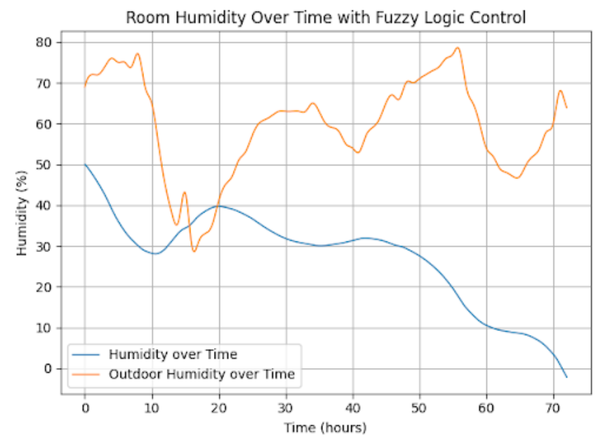


Fig. 11: Humidity over time.

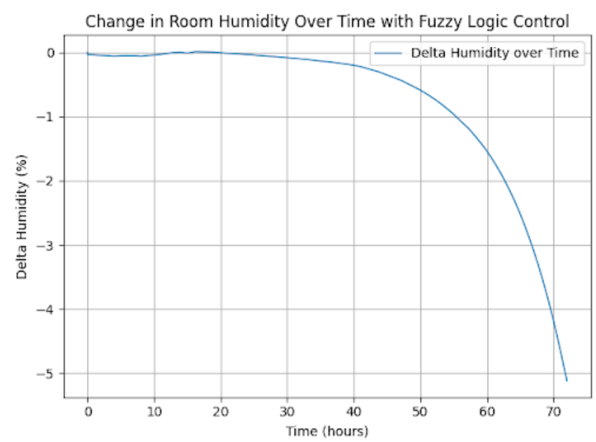


Fig. 12: Change in Humidity over time.

### D. *Humidity*

Figs 11, 12, and 13 show that the control for the humidity failed completely. A big reason is that there is no proper humidity calculation in the environment simulation. Attempts to find an equation bore no fruit, unfortunately. The following methods of fixing this were attempted.

- Adding a "change in humidity" input variable, similar to the "change in temperature" variable.
- Small changes to the humidity equation used.
- Tweaking the membership functions for humidity and humidifier/dehumidifier power.

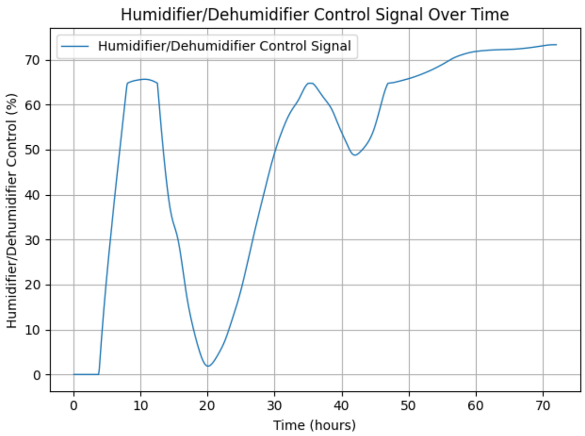


Fig. 13: Humidifier/Dehumidifier Control over time.

While it is disappointing that the humidity controls did not work properly, we still received good results from the temperature.

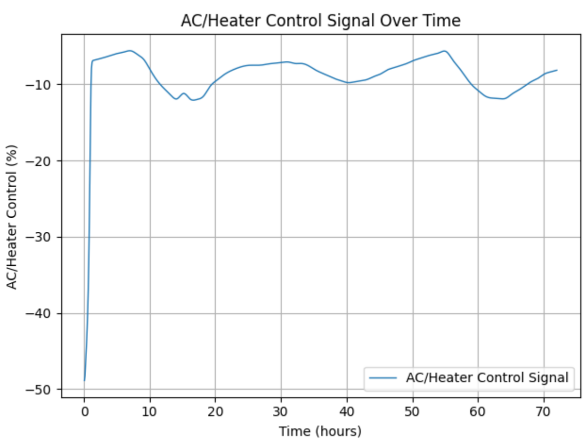


Fig. 14: Shifted AC/Heater Power over time.

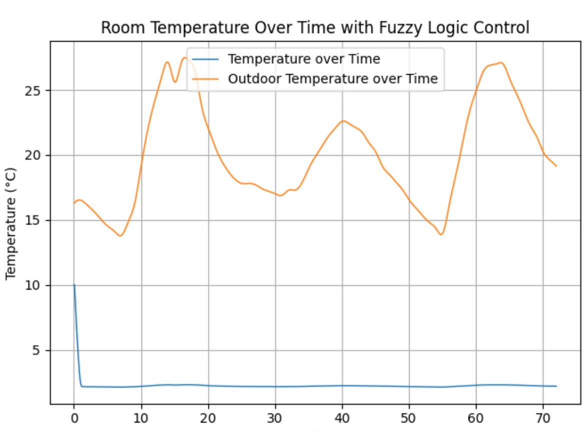


Fig. 15: Shifted Temperature over time.

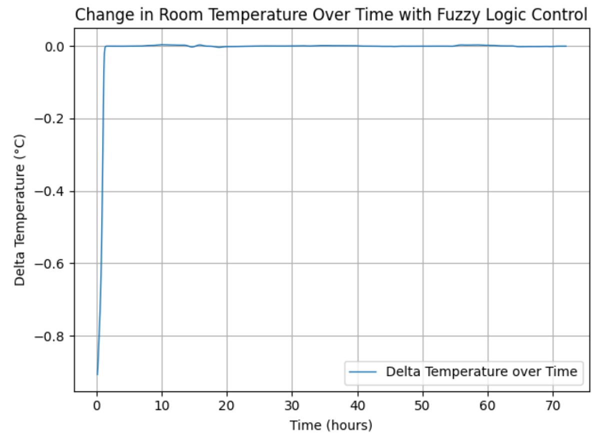


Fig. 16: Shifted Change in Temperature over time.

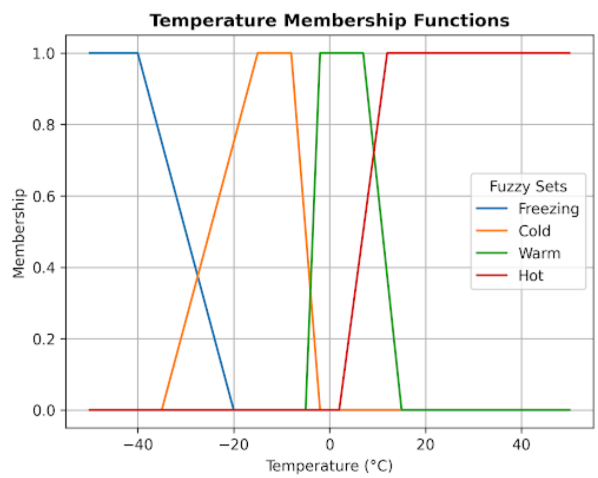


Fig. 17: Shifted Temperature Membership Functions.

## X. Shifting Temperature

To properly test the model, the simulation was run again with different membership functions for temperature. The only change was shifting the membership functions to the left by 20 °C ( Seen in Fig 17 ). The computed temperature, change in temperature, and AC/Heater power graphs can be seen in Figs 15, 16, and 14. A few things can be noticed here.

- The top of the warm membership function for temperature is now around 0°C. This is likely because the new stable temperature is far from the outdoor temperature, so there is a much larger influx of heat from the outside.
- The AC/Heater power is no longer staying at zero for large periods of time.
- The AC/Heater power is still holding around a certain point ( roughly $-10\%$ ).
- The change in room temperature over time is almost exactly zero the entire time after reaching the desired temperature originally.

## XI. Fuzzy Logic Options

Mamdani fuzzy logic was used in this model due to its easy implementation and use in the Python library being used to handle the fuzzy logic. Centroid defuzzification is used for the same reasons as Mamdani fuzzy logic.

## XII. Coding Environment

The simulation was run in a dev container built on a Python 3.13 image. The library that did the fuzzy logic is called skfuzzy. Beyond that, the libraries numpy, matplotlib, and scipy were used. The link to the repo for the project is HERE. The main program is in the main branch, while the version with the temperature membership functions shifted by $-20°\mathrm{C}$ is in the negative20 branch. In order to run the simulation, simply run the system.py script.

## XIII. Conclusion

The model in this paper has successfully shown fuzzy logic being used to control something. While the results are not all good, the results for the temperature are promising, showing a simple way to control temperature while minimizing energy consumption. Changes such as using professional environment-simulating software would be massive, allowing much more accurate calculations for new temperatures and other inputs. Overall, using fuzzy logic can be a simple way to easily control systems, and others should consider using it as an alternative to traditional methods.

A huge thank you goes to Mr. Gill from Mercer's Mechanical Engineering department for meeting with me on two separate occasions in order to assist with the environment simulation. The results would have been much less accurate without his guidance.

## References

[1] S. S. Izquierdoa and L. R. Izquierdob, "Mamdani fuzzy systems for modelling and simulation: A critical assessment," *SSRN*, 1 2017. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2900827

[2] J. W. Pouria Bahramnia, Seyyed Mohammad Hosseini Rostami and G. jun Kim, "Modeling and controlling of temperature and humidity in building heating, ventilating, and air conditioning system using model predictive control," *Energies*, vol. 12, no. 24, 12 2019. [Online]. Available: https://www.mdpi.com/1996-1073/12/24/4805

[3] A. Afram and F. Janabi-Sharifi, "Review of modeling methods for hvac systems," *Applied Thermal Engineering*, vol. 67, no. 1, pp. 507–519, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1359431114002348