IOP Conf. Series: Materials Science and Engineering **732** (2020) 012039 doi:10.1088/1757-899X/732/1/012039

Design of the early fire detection based fuzzy logic using multisensor

F Z Rachman*, **N Yanti, H Hadiyanto, S Suhaedi, Q Hidayati, M E P Widagda and B A Saputra**

Electrical Engineering Department, State Polytechnic of Balikpapan Jl. Soekarno Hatta km.8, Balikpapan 76126, Indonesia

*fathur.zaini@poltekba.ac.id

Abstract. The purpose of this study is to create an early fire detection system based on fuzzy logic. This system provides early warning of fire hazards, reduces the risk of casualties, and is able to be implemented to a wider scale or scope. This system consists of a multisensor to detect fire, smoke and temperature in the room. KY-026 sensor as fire detection, MQ-9 sensor as smoke detection and DS18b20 sensor as temperature detection. The results of the sensor readings are processed by a microcontroller implanted with a Sugeno fuzzy logic method. Sensor input is processed through several stages, namely fuzzification, rule evaluation and deffuzification. The output from this system is a condition value between 1 and 5. The average error of testing between Arduino modules with Matlab is 0.99%.

1. Introduction

According to the Republic of Indonesia PU Regulation No. 26 / PRT / M / 2008, fire hazard is a hazard caused by a potential threat and the degree of exposure to fire from the beginning of the fire to the spreading of the fire that causes smoke and gas [1].

Some research has been done to create a system that aims to detect a threat that can cause a fire if left unchecked. An example of a form of fire detection solution is the creation of a study entitled Design of a IoT-Based Fire Detection System (Internet of Things) and SMS Gateway Using Arduino [2]. Here the researcher uses the GSM module as a long distance communication system to send SMS messages to the user. Then in the next research that is the Design of Residential House Fire Detection Systems in Urban Areas Based on Microcontrollers is also one of the forms of existing solutions [3]. In this study researchers used the ESP8266 (WiFi) module as a long-distance communication system.

There are also studies with similar principles but with different methods. An example is the research on Android-Based Raspberry Pi Fire Detection System [4]. In that study, researchers used the Raspberry Pi as the main microcontroller in processing sensor data. Then there is the research that is closest to the concept of this Final Project, namely research with the title Design of Fire Point Detection System Using Naive Bayes Method Using Arduino-Based Temperature Sensors and Fire Sensors [5], here the researchers use LM35 temperature sensors and fire sensors to search the point where the fire occurred. And to process sensor reading data, the researchers used the Naïve Bayes method which was embedded in the Arduino as a microcontroller.

However, in this study there are still some shortcomings such as the performance of the fire sensor that has not been able to detect a flame from a distance that is not close to the sensor because the number of sensors used is only one and its placement is only at a certain point so that the sensor only focuses on the closest point. Therefore, this research was made to add new features and some advantages from previous studies. Some examples of these new feature shapes are the use of a multisensor system so that in the reading of hotspots it will be more accurate because at every corner of the design there is a series of sensors that read the state of temperature, smoke / gas and fire and the addition of fuzzy methods in processing the sensor data so processing output has many variations of actions or conclusions based on fuzzy rules that are made with wireless sensor networks [6,7].

The use of a multisensor system in this study aims to improve accuracy and sensitivity so that fire threats can be detected as early as possible. The design will be done by placing three sensors namely a fire sensor, temperature sensor and smoke sensor every 8 points in the room, in this study the room will be replaced with a design or prototype. For the placement of the sensor system will be in every corner of the room. So that in this case the total sensor usage is 24 units. The design for the room to be used is planned to be square or rectangular, so that it will be easier to place the sensor system in each corner of the room.

2. System design

The system is built through the hardware and software design stages.

2.1. Hardware design

The hardware design is carried out based on block diagrams as shown in figure 1.

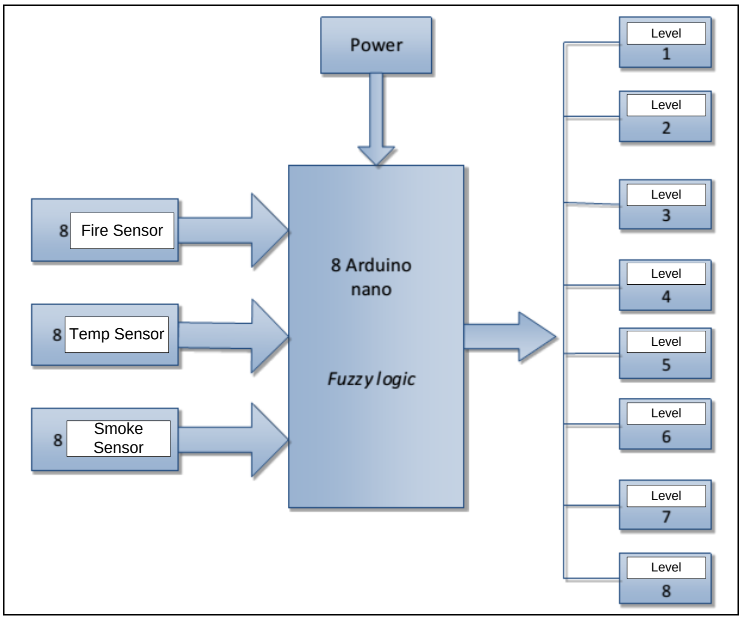

Figure 1. Fire detection system block diagram.

This system consists of 8 KY-026 fire sensors, 8 DS18B20 temperature sensors, 8 MQ-9 smoke sensors and 8 Arduino Nano. So that each sensor and Arduino will be placed at 8 corner points of the design, so that each corner point has its own system that is processed using the Sugeno fuzzy method. With a system like this, the point of reading the hotspot will be wider. This system consists of inputs, processes and outputs. The input section consists of the KY-026 fire sensor, the DS18B20 temperature sensor and the MQ-9 smoke sensor [8-10]. In the process part of this system using Arduino Nano with ATmega328 microcontroller base. And for the output of this system in the form of a firm value of 1 to 5 from the result of defuzzification.

2.2. Software design

The design of the software in question is the design of the system using the Sugeno fuzzy method. The design of the fuzzy method itself requires several stages in the form of fuzzification, rulemaking, inference and defuzzification. In figure 2 shows that the sub-process in fuzzy control has a function that is interconnected with other sub-processes so that the resulting sub-process will be input from the next sub-process until it becomes the final output of the system. Flow chart of fuzzy control design can be seen in figure 2.

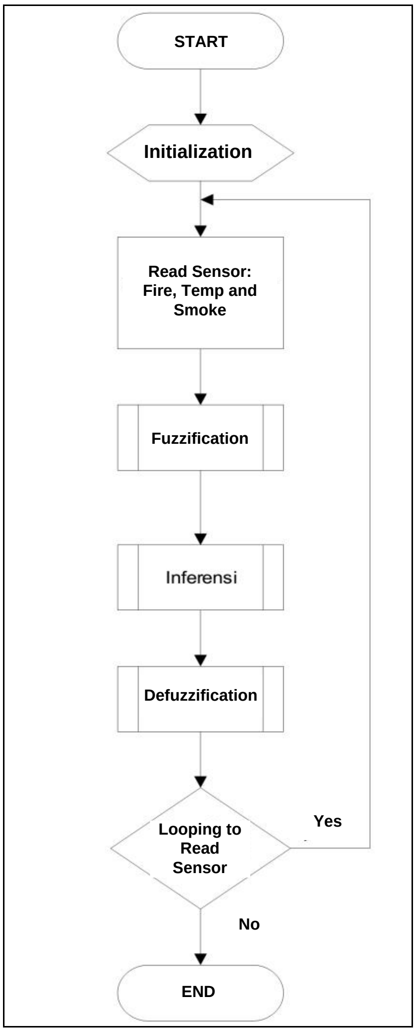

Figure 2. Flow chart of fuzzy design.

In figure 2, the fuzzification sub-system will process the input data obtained when reading. The data is in the form of firm or crisp values. The fuzzification sub-process will change the explicit values that exist into the membership function or degree of membership. The system built has 3 inputs in the form of fire data, temperature data and smoke content in the design. In the fire data are classified into 4 criteria, namely: close, somewhat close, far and not detected. In the temperature data are classified into 5 criteria, namely: cold, cool, comfortable, warm and hot. And the smoke data is classified into 5 criteria, namely: no smoke, tenuous, moderate, concentrated, and very concentrated. Each input data will be checked for membership value to determine the input class. Fuzzy set design of the three inputs can be seen in figure 3, figure 4 and figure 5.

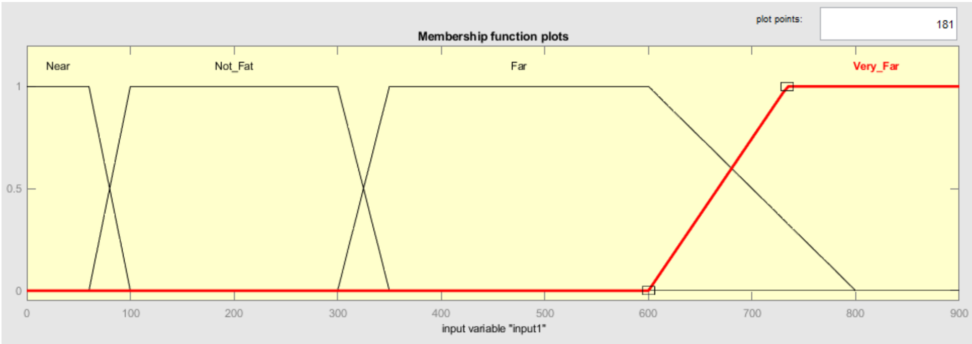

Figure 3. Fire variable membership function.

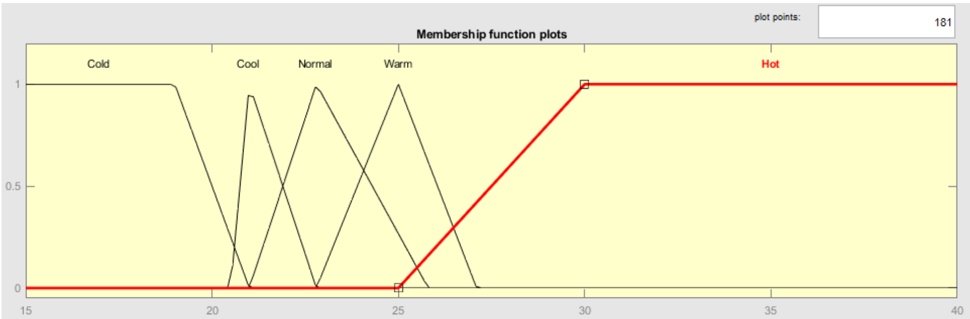

Figure 4. Temperature variable membership function.

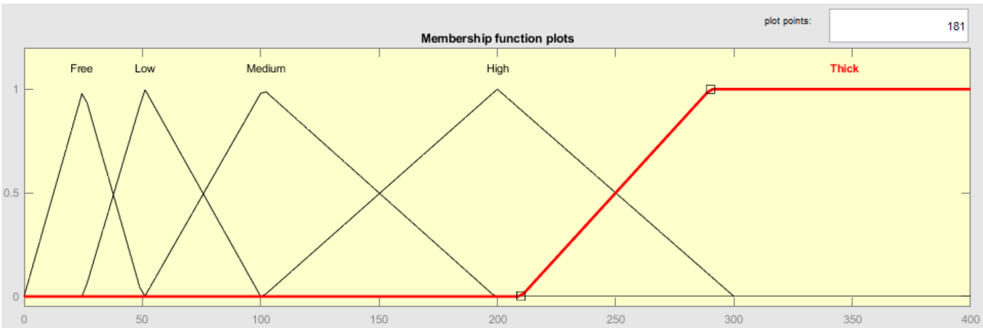

Figure 5. Smoke variable membership function.

The range of values given to each membership function can be seen in table 1, table 2 and table 3.

$$\mu\,[x] = \begin{cases} \mathbf{0}\,; & x \leq a \\ \dfrac{b-x}{b-a}\,; & a \leq x \leq b \\ \mathbf{1}\,; & x \geq a \\ \mathbf{0}\,; & x \leq a \\ \dfrac{x-a}{b-a}\,; & a \leq x \leq b \\ \mathbf{1}\,; & x \geq b \end{cases} \qquad \ldots\ldots (1)$$

Table 1. Fire variable membership function.

Membership Function	Value (ADC)
Near	≤ 100
Not Far	≥ 60 and ≤ 350
Far	≥ 300 dan ≤ 800
very far	≥ 600

Table 1 is a function of association of fire variables. Based on the input variables, the collection function is divided into 4, namely: Near, Not Far, far, Very Far.

Table 2. Temperature variable membership function.

Membership Function	Value (°C)
Cold	≤ 21
Cool	≥ 20,5 and ≤ 22,8
Normal	≥ 21 and ≤ 25,8
Warm	≥ 22,8 and ≤ 27,1
Hot	≥ 25

Table 2 is a table of temperature variable membership functions. Based on the input variables, the fuzzy membership function is divided into 5, namely: cold, cool, normal, warm and hot.

Table 3. Smoke variable membership function.

Membership Function	Value ((ppm)
Smoke free	≥0 and ≤50
low smoke	≥25 and ≤100
Medium Smoke	≥51 and ≤199
High Smoke	≥101 and ≤300
Thick smoke	≥ 300

Table 3 is a membership table function of smoke input variable. Variable smoke input is divided into 5 parts namely Free, Low, Medium, High and Thick.

After the fuzzification process is finished, proceed with the inference process. Inference is the process of combining many rules based on available data. Inference engine is a process to convert fuzzy input into fuzzy output by following the rules (IF-THEN Rules) that have been set on the fuzzy knowledge base using the MIN implication function to get the α-predicate value of each rule (α1, α2, α3 , ... αn,) [11].

The result of inference is the set of fuzzy logic or fuzzy logic membership functions. Information in fuzzy logic must be converted into explicit values so that it can be used to control a process. The firm value must reflect the information contained in the fuzzy logic set. The process used to convert the results of fuzzy inference into firm value output is called defuzzification. In the Sugeno fuzzy method the implication function used is MIN (Minimum) to get the α-predicate of each rule. Then each α-predicate value is used to calculate the output of the results of the inference explicitly (crips) of each rule (Z_1, Z_2, ... Z_a). After that, the final process is defuzzification. In this system applying the defuzzy weighted average method to find the value of the system output.

The fuzzy output set in the rule mechanism is singleton output with a range of 1 to 5 mapped to 5 linguistic variables. The five variables are A with a value of singleton 1, B with a value of singleton 2, C with a value of singleton 3, D with a value of singleton 4 and E with a value of singleton 5.

3. Results and Discussion

After the next design phase the implementation stage. Implementation is carried out in accordance with the design that has been done before. The hardware implementation can be seen in figure 6.

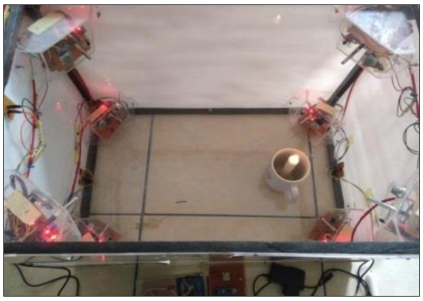

Figure 6. Hardware implementation.

Table 4. Test results of fire locations in the lower right without gas.

No	Multisensor	Fire (ADC)	Temp(°C)	Smoke (ppm)	*Output Fuzzy*
1	A	51.00	27.37	0.36	3.00
2	B	52.00	29.06	1.75	3.00
3	C	36.00	27.06	1.75	2.71
4	D	56.00	29.50	3.22	3.00
5	E	37.00	29.31	0.05	3.00
6	F	43.00	29.06	1.75	3.00
7	G	45.00	26.69	0.80	2.50
8	H	47.00	27.5	2.25	3.00

Table 5. Test results of fire locations in the lower right with gas.

No	Multisensor	Fire (ADC)	Temp(°C)	Smoke (ppm)	O*utput Fuzzy*
1	A	227	27.50	33.60	3.50
2	B	219	29.25	160.25	4.50
3	C	34	27.37	53.30	4.00
4	D	61	29.75	94.99	4.00
5	E	39	29.56	134.28	4.45
6	F	45	28.62	45.10	3.61
7	G	43	26.62	211.26	4.26
8	H	44	27.69	38.44	3.50

From the results of multisensor testing with or without being given gas originating from matches, the results of multisensor testing without gas as shown in table 4 shows the results of the outputs worth 2.50 to 3.00. There are several factors that can affect the reading results of a multisensor. For example, when taking multisensory data the existing fire sensor in the system experiences a slight disturbance due to the sunlight factor and lights from a very bright room, thus affecting the reading of the fire sensor which should not have detected the presence of fire but when testing the sensor has detected the presence of light from the sun and lights. Whereas in table 5 the test is carried out by giving a source of fire and gas. The results of the test produce a value of more than 3.00, this is because in this test there is an additional input of gas coming from matches. Based on the fuzzy rules that have been made, then if the system detects a fire and there is a large enough gas content, the fuzzy output will usually produce a value of more than 3.00.

Fuzzy testing by comparing the results of fuzzy output on Arduino with fuzzy output on MATLAB. The test results are calculated the difference between the two test results, so it can be determined how many errors there are between the reading of fuzzy outputs on modules with fuzzy output using MATLAB software. The data for other experimental results can be seen in table 6.

Table 6. Fuzzy logic testing results.

No	*Input*			*Output*		Differ	*Error*
	Temp(°C)	Fire(ADC)	Smoke(ppm)	Arduino	Matlab		
1	23.40	48.8	26.5	2.00	1.28	0.72	5.62%
2	29.25	774	1.31	2.00	2.00	0	0%
3	24.00	50	20	2.00	1.40	0.6	4.28%
4	25.00	25	280	4.00	4.00	0	0%
5	35.00	25	300	5.00	5.00	0	0%
6	15.00	500	25	1.00	1.00	0	0%
7	25.00	200	100	2.73	2.73	0	0%
8	25.00	350	200	3.00	3.00	0	0%
9	10.00	800	20	1.00	1.00	0	0%
10	15.00	200	500	4.00	4.00	0	0%
Average						**0.13**	**0.99%**

4. Conclusion

Based on the discussion and testing of the previous tools MQ-9 sensor testing found an average error of 7.29%. From the test data 1 to 10 experience a difference that is not so large. The difference is due to the unstable MQ-9 sensor readings compared to the gas / smoke readings on the measuring instrument. There is a fuzzy testing program between Arduino and fuzzy that has been made in MATLAB, there are some differences that are not large. This is as shown in the test data. From 10 test data, there are 2 test data that experience the difference. Difference found as in test 1 found an error of 5.62% with a difference value of 0.72 and in test 3 found an error of 4.28% with a difference value of 0.6. So that found an average error of 0.99%. The output of this system is the defuzzification value of each module from the eight corner points. So that the final output can only be seen on the display interface that has been created. Deffuzification output is a value from 1 to 5.

References

[1] Permen PU RI No. 26/PRT/M/2008 Tentang Persyaratan Teknis Sistem Proteksi Kebakaran Pada Bangunan Gedung Dan Lingkungan.

[2] Sasmoko D and Mahendra A 2017 Rancang Bangun Sistem Pendeteksi Kebakaran Berbasis IOT dan SMS Gateway Menggunakan Arduino *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer* **8**(2) 469-476

[3] Yendri D, Wildian W and Tiffany A 2017 Perancangan Sistem Pendeteksi Kebakaran Rumah Penduduk Pada Daerah Perkotaan Berbasis Mikrokontroler *Prosiding Semnastek*

[4] Putra I W P A, Piarsa I N, and Wibawa K S 2018 Sistem Pendeteksi Kebakaran Menggunakan Raspberry Pi Berbasis Android *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)* 167-173

[5] Dana M M 2018 Rancang Bangun Sistem Deteksi Titik Kebakaran Dengan Metode Naive Bayes Menggunakan Sensor Suhu dan Sensor Api Berbasis Arduino *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer* 3384-3390

[6] Rachman F Z 2015 Prototype development of monitoring system in patient infusion with wireless sensor network *Proc. International Seminar on Intelligent Technology and Its Applications (ISITIA) IEEE* 397-402

[7] Rachman F Z, Yanti N, Jamal N, Purwanto E 2018 Wireless Sensor Network With Mesh Topology To Control Room Devices *Proc. International Conference on Applied Science and Technology (iCAST) IEEE* 5-9

[8] Datasheet KY-026

[9] Datasheet DS18B20

[10] Datasheet MQ-9

[11] Pratama A S 2018 *Rancang Bangun Penyiram Air Otomatis Dan Proteksi hama Tanaman Menggunakan Fuzzy Logic Control* (Yogyakarta: Tesis UNY)