



LÓGICA DIFUSA

Daniel Alejandro De Los Santos Cuevas

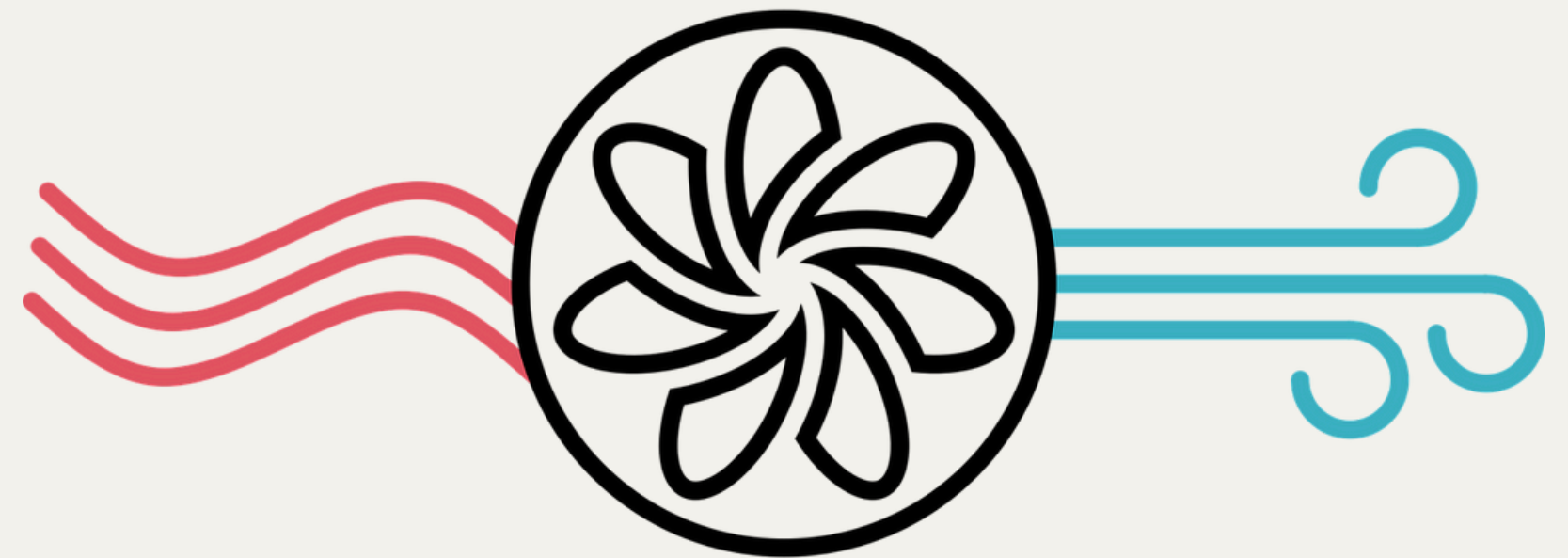
¿QUÉ ES LA LÓGICA DIFUSA?

La lógica difusa es una rama de la inteligencia artificial que le permite a una computadora analizar información del mundo real en una escala entre lo falso y lo verdadero, manipula conceptos vagos, como "caliente" o "húmedo", y permite a los ingenieros construir dispositivos que juzgan la información difícil de definir.



LÓGICA DIFUSA EN UN AIRE ACONDICIONADO

La lógica difusa permite que el control del aire acondicionado sea más suave e inteligente. En lugar de simplemente encender o apagar el compresor, ajusta la potencia y la velocidad del ventilador de manera gradual dependiendo de cuánto difiere la temperatura actual de la deseada.



CÓDIGO

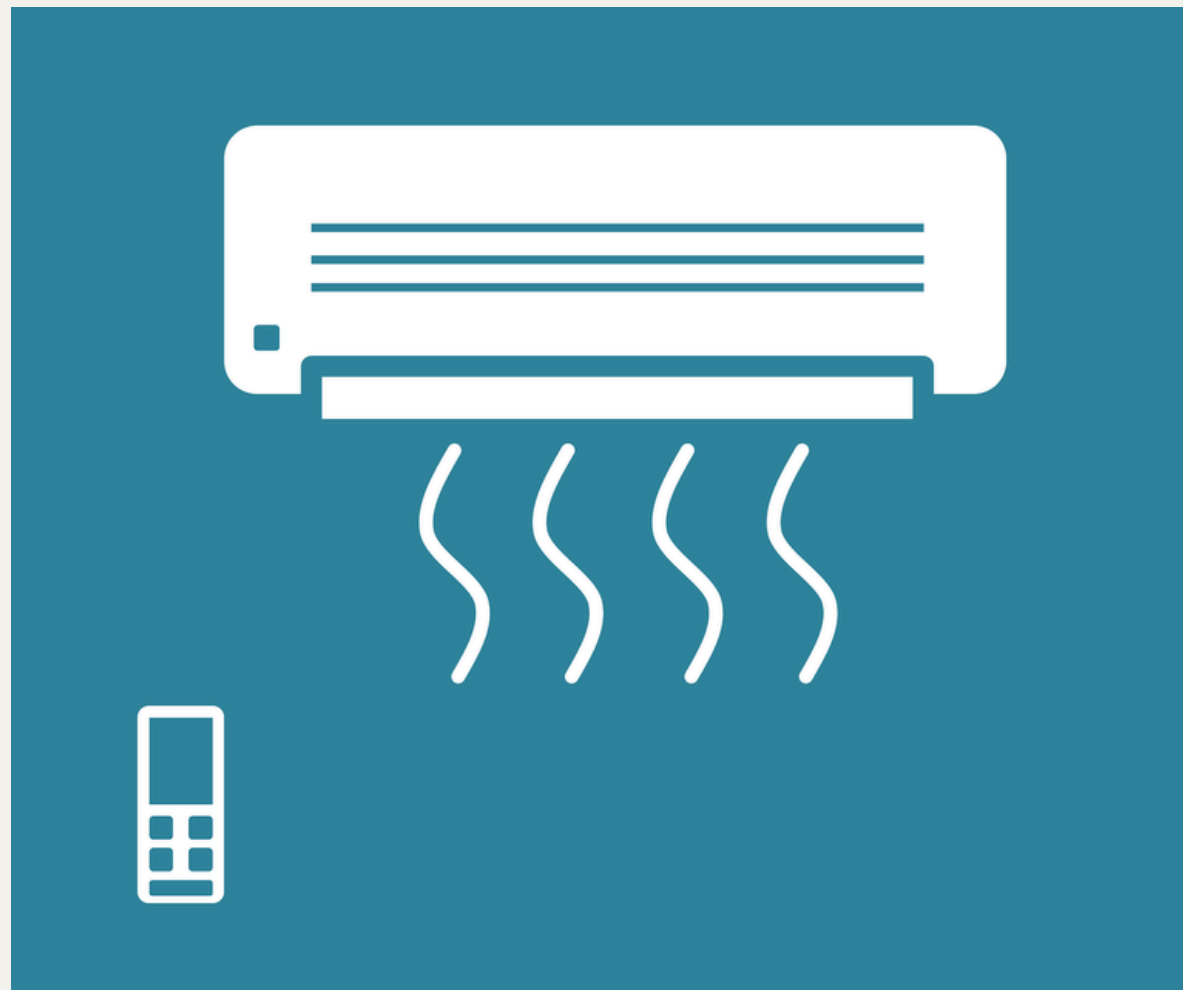
```
1  #include <DHT.h>
2  #include <Fuzzy.h>
3
4  #define DHTPIN 2          // Pin del sensor DHT11
5  #define DHTTYPE DHT11
6  #define RELAY_COMPRESOR 3 // Pin del relé para el compresor
7  #define FAN_PIN 5         // Pin del ventilador
8
9  DHT dht(DHTPIN, DHTTYPE);
10 Fuzzy fuzzy;
11 float setTemperature = 24.0; // Temperatura deseada en °C
12
13 // Variables de la lógica difusa
14 FuzzySet muyFrio(0, 0, 15, 20);
15 FuzzySet frio(15, 20, 22, 24);
16 FuzzySet ideal(22, 24, 26, 28);
17 FuzzySet caliente(26, 28, 30, 35);
18 FuzzySet muyCaliente(30, 35, 40, 40);
19
20 FuzzySet ventiladorBajo(0, 0, 50, 100);
21 FuzzySet ventiladorMedio(50, 100, 150, 200);
22 FuzzySet ventiladorAlto(150, 200, 255, 255);
23
24 void setup() {
25     Serial.begin(9600);
26     dht.begin();
27     pinMode(RELAY_COMPRESOR, OUTPUT);
28     pinMode(FAN_PIN, OUTPUT);
29
30     // Entrada: Diferencia de temperatura
31     FuzzyInput *tempDiff = new FuzzyInput(1);
32     tempDiff->addFuzzySet(&muyFrio);
33     tempDiff->addFuzzySet(&frio);
34     tempDiff->addFuzzySet(&ideal);
35     tempDiff->addFuzzySet(&caliente);
36     tempDiff->addFuzzySet(&muyCaliente);
37     fuzzy.addFuzzyInput(tempDiff);
```

```
38
39     // Salida: Velocidad del ventilador
40     FuzzyOutput *fanSpeed = new FuzzyOutput(1);
41     fanSpeed->addFuzzySet(&ventiladorBajo);
42     fanSpeed->addFuzzySet(&ventiladorMedio);
43     fanSpeed->addFuzzySet(&ventiladorAlto);
44     fuzzy.addFuzzyOutput(fanSpeed);
45
46     // Reglas difusas
47     FuzzyRuleAntecedent *siMuyFrio = new FuzzyRuleAntecedent();
48     siMuyFrio->joinSingle(&muyFrio);
49     FuzzyRuleConsequent *entoncesVentiladorBajo = new FuzzyRuleConsequent();
50     entoncesVentiladorBajo->addOutput(&ventiladorBajo);
51     FuzzyRule *regla1 = new FuzzyRule(1, siMuyFrio, entoncesVentiladorBajo);
52     fuzzy.addFuzzyRule(regla1);
53
54     FuzzyRuleAntecedent *siFrio = new FuzzyRuleAntecedent();
55     siFrio->joinSingle(&frio);
56     FuzzyRuleConsequent *entoncesVentiladorMedio = new FuzzyRuleConsequent();
57     entoncesVentiladorMedio->addOutput(&ventiladorMedio);
58     FuzzyRule *regla2 = new FuzzyRule(2, siFrio, entoncesVentiladorMedio);
59     fuzzy.addFuzzyRule(regla2);
60
61     FuzzyRuleAntecedent *siCaliente = new FuzzyRuleAntecedent();
62     siCaliente->joinSingle(&caliente);
63     FuzzyRuleConsequent *entoncesVentiladorAlto = new FuzzyRuleConsequent();
64     entoncesVentiladorAlto->addOutput(&ventiladorAlto);
65     FuzzyRule *regla3 = new FuzzyRule(3, siCaliente, entoncesVentiladorAlto);
66     fuzzy.addFuzzyRule(regla3);
67
68     FuzzyRuleAntecedent *siMuyCaliente = new FuzzyRuleAntecedent();
69     siMuyCaliente->joinSingle(&muyCaliente);
70     FuzzyRuleConsequent *entoncesVentiladorAlto2 = new FuzzyRuleConsequent();
71     entoncesVentiladorAlto2->addOutput(&ventiladorAlto);
72     FuzzyRule *regla4 = new FuzzyRule(4, siMuyCaliente, entoncesVentiladorAlto2);
73     fuzzy.addFuzzyRule(regla4);
74 }
75
76 void loop() {
77     float currentTemp = dht.readTemperature();
78     if (isnan(currentTemp)) {
79         Serial.println("Error al leer temperatura!");
80         return;
81     }
```

CÓDIGO

```
82
83     float diffTemp = currentTemp - setTemperature;
84     Serial.print("Diferencia de temperatura: ");
85     Serial.println(diffTemp);
86
87     fuzzy.setInput(1, diffTemp);
88     fuzzy.fuzzify();
89
90     int fanSpeed = fuzzy.defuzzify(1);
91     analogWrite(FAN_PIN, fanSpeed);
92
93     Serial.print("Velocidad del ventilador: ");
94     Serial.println(fanSpeed);
95
96     delay(2000);
97 }
98
```

¿POR QUÉ USAR LÓGICA DIFUSA?



Mejor confort térmico: No hay cambios bruscos en la temperatura.

Mayor eficiencia energética: El compresor y el ventilador trabajan con menor consumo.

Mayor duración del equipo: Evita encendidos y apagados bruscos.



MUCHAS
GRACIAS

