

Proyecto de Simulación Lógica Difusa

Dayany Alfaro González-C411

1. Características del Sistema de Inferencia Propuesto

El Sistema de Inferencia propuesto está implementado usando Python como lenguaje. Entre las principales características se encuentra que permite usar los métodos de agregación *Mamdani* o *Larsen*, mientras que para la etapa de defusificación es posible usar los métodos Centroide, Bisección o cualquiera de las variantes de los Máximos. En cuanto a las funciones de pertenencia a definir para los conjuntos difusos ofrece como opciones el uso de funciones triangulares y trapezoidales. Por otra parte para definir las reglas es posible usar los operadores lógicos *and*, *or* y *not*, los cuales tienen asociadas las funciones *min*, *max* y $1 - x$ respectivamente. Soporta múltiples variables de entrada y múltiples variables de salida.

2. Principales Ideas seguidas para la implementación del Sistema

Para el diseño del sistema se implementan las siguientes clases:

FuzzyInputVariable: representa una variable lingüística que se va a usar en el antecedente de las reglas. Tiene un nombre, un dominio y un valor de dicho dominio asociado.

FuzzyOutputVariable: representa una variable lingüística que se va a usar en el consecuente de las reglas. Tiene un nombre y un dominio.

FuzzySet: representa un conjunto difuso, el cual tiene un nombre, una variable lingüística (**FuzzyInputVariable** o **FuzzyOutputVariable**) que está clasificando y una función de pertenencia.

FuzzyRule: representa una regla difusa, la cual tiene una lista de los nombres de los conjuntos difusos y los operadores que forman su antecedente. Además, tiene el nombre del conjunto difuso de salida que es el consecuente de la regla. Tiene una función **Evaluate** que dadas las entradas de cada variable lingüística al sistema se encarga de evaluar la regla, fusificando primero los valores y luego aplicando los operadores entre cada par de términos consecutivos.

FuzzyInferenceSystem: representa el Sistema de Inferencia Difusa. Para crearlo es necesario suministrarle un string que identifique el método de agregación (e.g: mamdani) y el método de defusificación (e.g: centroid) que se quiere utilizar. Para realizar la inferencia esta clase provee del método **infer**, el cual necesita los siguientes parámetros:

- **rules:** es un diccionario donde las llaves van a ser un string que representa los antecedentes y los valores un string que representa los consecuentes. Tienen que tener un formato específico. Los operadores se van a representar por &(and), |(or) y ~(not), mientras que los conjuntos difusos deben estar escritos en la forma Nombre del conjunto concatenado con Nombre de la variable al que está asociado. En el caso de que existan varias variables de salida los nombres de los correspondientes conjuntos difusos tienen que estar separados por comas(.). Un ejemplo extraído del problema que se propone para validar el sistema propuesto:
Si la regla es *If Temperature is Cold and Humidity is Wet Then Speed is Slow* esto se representaría como *ColdTemperature & WetHumidity: SlowSpeed*.
- **fsets:** es un diccionario en el que las llaves son el nombre que va a tener un conjunto difuso al usarse en la regla (*ColdTemperature* en el ejemplo anterior) y los valores van a ser los objetos **FuzzySet**.
- **outvars:** es una lista con las instancias de **FuzzyOutputVariable** que representan las variables lingüísticas de salida del sistema.

3. Propuesta de Problema a Solucionar mediante inferencia difusa

El sistema propuesto controla la velocidad de un ventilador de acuerdo a la temperatura y humedad del ambiente. Por tanto, consiste de 2 entradas, temperatura y humedad, y una sola salida que sería la velocidad del ventilador.

Los conjuntos difusos que van a describir las variables de entrada y salida son los siguientes:

- Temperature: Cold, Medium, Hot
- Humidity: Dry, Normal, Wet
- Speed: Slow, Moderate, Fast

En la figura 1 se muestran las funciones de membresía de cada uno de dichos conjuntos difusos.

Un total de 9 reglas son usadas para describir el conocimiento necesario para operar el ventilador:

- If Temperature is Cold and Humidity is Wet Then Speed is Slow
- If Temperature is Medium and Humidity is Wet Then Speed is Slow
- If Temperature is Cold and Humidity is Normal Then Speed is Slow

- If Temperature is Hot and Humidity is Wet Then Speed is Moderate
- If Temperature is Medium and Humidity is Normal Then Speed is Moderate
- If Temperature is Cold and Humidity is Dry Then Speed is Moderate
- If Temperature is Hot and Humidity is Normal Then Speed is Fast
- If Temperature is Hot and Humidity is Dry Then Speed is Fast
- If Temperature is Medium and Humidity is Dry Then Speed is Fast

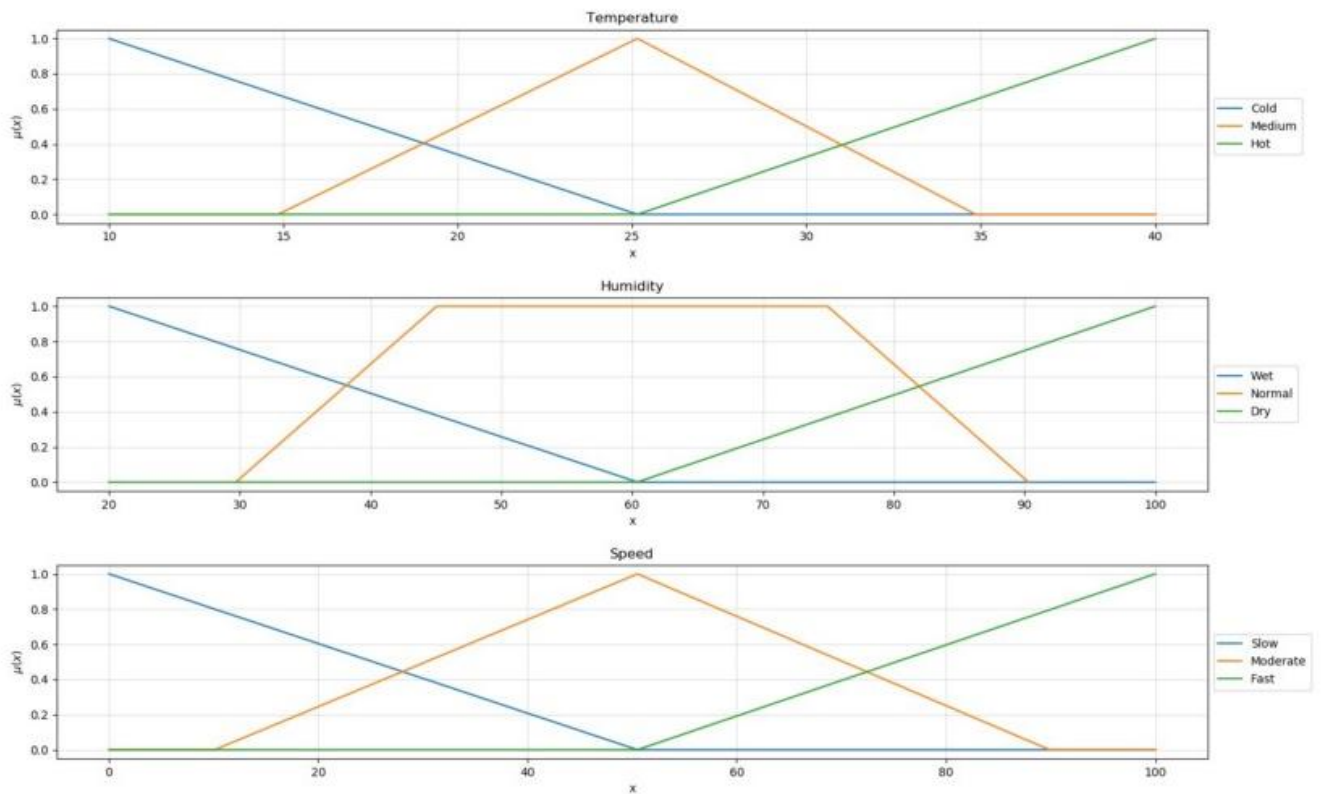


Figura 1: Funciones de membresía.

4. Consideraciones obtenidas a partir de la solución del problema con el sistema de inferencia implementado

Se hizo un análisis de cómo se comporta el sistema para los siguientes valores de las variables de entrada:

- **Temperature:** 30
- **Humidity:** 60

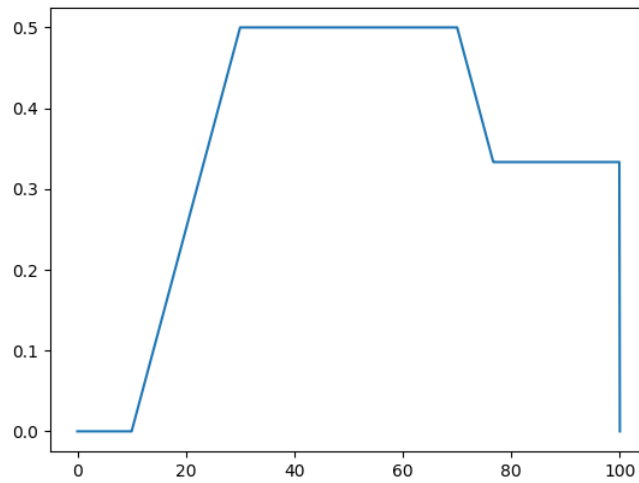


Figura 2: Función de agregación Mamdani.

Al aplicar la función de agregación Mamdani se obtiene como salida la función que se muestra en la figura 2. Mientras que la aplicación de Larsen produce la función que se muestra en la figura 3.

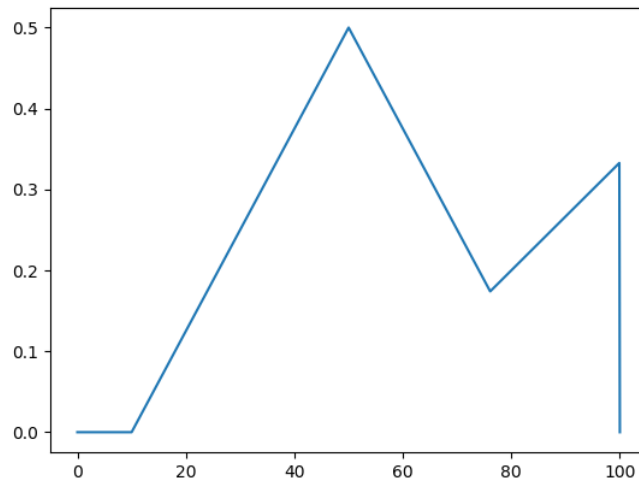


Figura 3: Función de agregación Larsen.

En el cuadro 1 se pueden observar los resultados obtenidos al aplicar todas las posibles combinaciones de métodos de agregación y defusificación. Como se aprecia la mayoría de los resultados son relativamente cercanos exceptuando cuando se hace uso de el Primer y el Último Máximo como métodos de defusificación luego de haber aplicado Mamdani como método de agregación.

	Mamdani	Larsen
Centroide	56.42	58.07
Bisector	55.5	55.3
Máximo Promedio	50.0	50.0
Primer Máximo	30.0	50.0
Último Máximo	70.0	50.0

Cuadro 1: Resultados según método de agregación y defusificación utilizados.