Proyecto de Lógica Difusa de Simulación

Loraine Monteagudo García (C-411)

I. Sistema de Inferencia Propuesto

Un sistema de inferencia está constituido por 4 componentes principales: una base de conocimiento, sistemas de difusificación, sistemas de inferencia y sistemas de desdifusificación.

i. Difusificación

Este componente le asigna a las variables de entrada distintos grados de pertenencia en las distintas clases. Para esto, se establece una correspondencia entre los valores de entrada a valores de 0 a 1 usando un conjunto de funciones de pertenencia.
Las funciones de pertenencia implementadas en nuestro sistema fueron:

1. Triangular (Figura I.1): Está definido por sus límites superior (a) y superior (b) y el valor modal (m) tal que $a < m < b$.

$$f(x) = \begin{cases} 0 & \text{si} \quad x \leq a \\ \dfrac{x-a}{m-a} & \text{si} \quad x \in (a, m] \\ \dfrac{b-x}{b-m} & \text{si} \quad x \in (m, b) \\ 0 & \text{si} \quad x \geq b \end{cases}$$

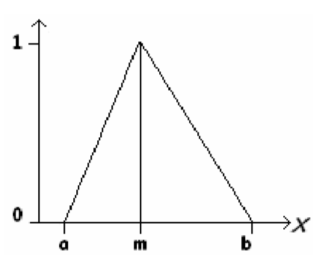

Figura I.1: Función Triangular

2. Función Trapezoidal (Figura I.2): Definida por sus límites superior (a) y superior (d) y sus límites de soporte, b y c, inferior y superior respectivamente.

$$f(x) = \begin{cases} 0 & \text{si} \quad x \leq a \text{ o } x \geq d \\ \dfrac{x-a}{b-a} & \text{si} \quad x \in (a,b] \\ 1 & \text{si} \quad x \in (b,c) \\ \dfrac{d-x}{d-c} & \text{si} \quad x \in (b,d) \end{cases}$$

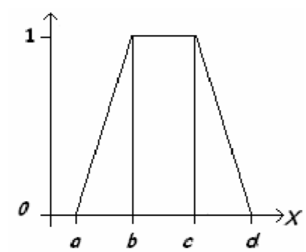

Figura I.2: Función Trapezoidal

3. Función S (Figura I.3): Definida por sus límites inferior (a) y superior (b), y el valor m, tal que $a < m < b$.

$$f(x) = \begin{cases} 0 & \text{si} \quad x \leq a \\ 2(\dfrac{x-a}{b-a})^2 & \text{si} \quad x \in (a,m] \\ 1-2(\dfrac{x-b}{b-a})^2 & \text{si} \quad x \in (m,b) \\ 1 & \text{si} \quad x \geq b \end{cases}$$

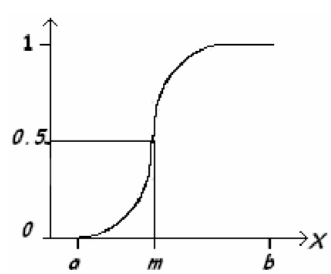

Figura I.3: Función S

Cada una de las funciones de pertenencia fue implementada en `membership.py`, donde cada función es una clase que recibe como parámetro cada una de las variables expuestas anteriormente. Después se evalúan para cada uno de los puntos distintos de x.

II. Operaciones con conjuntos difusos

Se definieron las tres operaciones básicas de los conjuntos difusos:

- Unión: $A \vee B \implies A \cup B \implies max\{\mu_A(x), \mu_B(x)\}$
- Intersección: $A \wedge B \implies A \cap B \implies min\{\mu_A(x), \mu_B(x)\}$
- Complemento: $\overline{A} \implies \neg A \implies \mu_A = 1 - \mu_A$

Donde μ_A es la función de pertenencia de A.
Cada una de estas operaciones está implementada en `operators.py`

III. Sistemas de inferencia

Los mecanismos de inferencia relacionan los conjuntos difusos de entrada y salida, y representa a las reglas que definen el sistema. Las entradas a este componente son conjuntos difusos (grados de pertenencia) y las salidas son también conjuntos difusos, asociados a la variable de salida. Para la implementación de este sistema de inferencia se implementaron 2 métodos de agregación. Para la creación del sistema este recibe como entrada el nombre de uno de estos métodos.

Métodos de agregación

- Método de Mamdani: Supongamos que una regla tiene las siguiente forma:

  R_i = if u is A_i and v is B_i then w is C_i, i = 1,2, ..., n
  Para $u \in U$, $c \in V$, y $x \in W$.

  El método de Mamdani usa el operador de mínimo ($\wedge$) para la implicación difusa ($\implies$). Para la regla i la salida es:

  $$\mu_{C_i'}(w) = \alpha_i \wedge \mu_{C_i}(w)$$
  Donde $\alpha_i = \mu_{A_i}(u) \wedge \mu_{B_i}(v)$

  De manera general, si se tienen n reglas:

  $$\mu_{C'}(w) = \vee_{i=1}^{n}[\alpha_i \wedge \mu_{C_i}(w)] = \vee_{i=1}^{n}\mu_{C_i'}(w)$$

  $$C' = \cup_{i=1}^{n} C_i^{'}$$

- Método de Larsen: Para la siguiente regla:

  R_i: if u is A_i and v is B_i then w is C_i, i = 1, 2, ..., n

  En este caso se usa el operador producto ($\cdot$) para la implicación $\implies$. Para una regla i se tiene que:

  $$\mu_{C_i'}(w) = \alpha_i \cdot \mu_{C_i}(w)$$
  Donde $\alpha_i = \mu_{A_i}(u) \wedge \mu_{B_i}(v)$

Para n reglas:

$$\mu_{C'}(w) = \vee_{i=1}^{n}[\alpha_i \cdot \mu_{C_i}(w)] = \vee_{i=1}^{n}\mu_{C_i'}(w)$$

$$C' = \cup_{i=1}^{n} C_i^{'}$$

Estos métodos se encuentran implementados en `aggregation.py` como clases que reciben el conjunto de reglas a analizar, así como el dominio y la precisión con que se analizan los distintos valores del conjunto difuso de salida.

iv. Desdifusificación

En esta etapa se realiza la función contraria al difusor. La entrada que recibe es el conjunto difuso de salida, resultado de la etapa de inferencia y la salida es un valor concreto de la variable de salida. Para obtener, a partir del conjunto difuso de salida que resulta de la agregación de todas las reglas, un resultado escalar, se aplican métodos matemáticos. Estos algoritmos fueron implementados en `defuzzification.py`. A continuación, se realiza un resumen de ellos:

- *Mean of Maximum (MOM)*: Esta estrategia retorna el valor medio de todas las acciones de control, cuya función de pertenencia alcanza el máximo. Se usa la versión discreta, cuyo resultado se puede obtener de la siguiente forma:

  $$z_o = \sum_{j=1}^{k} \frac{z_j}{k}$$

  z_j : valores para los cuales las funciones de pertenencia alcanzan el máximo.

  k : cantidad de dichos valores.

- *Left of Maximum (LOM)*: De todos los posibles acciones cuya función de pertenencia alcanza el máximo, se retorna el primer valor en alcanzarlo, es decir, el que se encuentra a la izquierda.
- *Right of Maximum (ROM)*: De todos los posibles acciones cuya función de pertenencia alcanza el máximo, se retorna el último valor en alcanzarlo, es decir, el que se encuentra a la derecha.
- *Median of Maximum (MMOM)*: De todos los valores que alcanzan el máximo en su función de pertenencia, se retorna el que se encuentra en la mitad.
- *Centroid of Area (COA)*: Este método genera el centro de gravedad de la posible distribución del conjunto difuso C de salida. Para nuestro caso discreto se usa la siguiente fórmula:

  $$z_0 = \frac{\sum_{j=1}^{n} \mu_C(z_j) \cdot z_j}{\sum_{j=1}^{n} \mu_C(z_j)}$$

- *Bisector of Area (BOA)*: El método del bisector encuentra la línea vertical que divide el conjunto difuso en dos subregiones de igual área.

II. Principales Ideas

Para la creación del sistema se analizó la estructura de las distintas componentes que las conforman, así como el concepto de algunos elementos de la lógica difusa.
En primer lugar, se considera lo que constituye una variable lingüística. Una variable lingüística constituye un concepto con un conjunto asociado de valores lingüísticos, siendo cada uno de

estos valores un conjunto difuso definido por una correspondiente función de membresía. Para representar estos conjuntos difusos se emplea la noción de adjetivo, estos están compuestos por una función de membresía que determina, además del grado de pertenencia a dicho conjunto difuso, cuán aplicable es dicho adjetivo a la variable. Por lo tanto, se tienen variables que tienen un nombre como forma de identificarlas y un conjunto de adjetivos. Estos adjetivos poseen también un nombre como medio de identificación y una función de pertenencia para describir el comportamiento de este adjetivo. Estas definiciones se encuentran en `linguistic_var.py`.
Luego, otra cuestión que considerar antes de analizar la estructura del sistema de inferencia propuesto es la representación de las reglas de inferencia. Las reglas tienen la forma general de:

Si [variable1] is [adjective1] entonces [variable2] es [adjective2]

En la cual x *is* A es el antecedente o premisa, que puede ser todo lo complicado que se quiera, con la interacción de varias proposiciones al estilo: $x_i is A_i$ a través de las operaciones entre los conjuntos difusos que fueron explicadas en la subsección II. De manera general, el antecedente es aquello que se encuentra entre *si* y *entonces*. Por otro lado *y es B* es el consecuente o conclusión de la regla. El consecuente también puede ser más complejo; cuando se tienen varias variables de salida se crean múltiples reglas con el mismo antecedente.
Una vez teniendo claro estos conceptos se tiene la estructura de lo que constituye una regla: un antecedente y un consecuente.
Se notó que en ambos métodos descritos en III se calcula el valor de α_i como la aplicación del antecedente. Por lo tanto, de forma precisa, dado un conjunto de valores que le corresponden a cada una de las variables que se encuentran en el antecedente este es capaz de calcular el valor de α_i.
Para la creación de las reglas se espera una oración del estilo: *Si `[antecedente]` entonces `[consequente]`*.
A partir del conjunto de variables y de adjetivos que se definen en el problema esta oración se parsea obteniendo una instancia real de las variables y los adjetivos que se encuentran en el antecedente y en el consecuente.
Después de tener representado estos conceptos, se procedió a la implementación del sistema de inferencia difuso. El sistema espera como primera entrada el método de agregación que se va a usar y el método de desdifusicación.
Para el inicio de la inferencia se añaden el conjunto de reglas, las variables y los adjetivos. Se espera además un dominio, que es el intervalo en el que se define el conjunto difuso de salida, así como una precisión para determinar cada uno de los elementos a analizar. Es decir, si se tiene que $dominio = (start, end)$ se analizan los valores x_0 formados de aplicar la secuencia: $(x = start, x < end, x+ = precision)$. Es en este primer momento, donde se establecen las reglas que se van a aplicar que estas se parsean.
Luego el sistema tiene otro método en el que realiza la evaluación del sistema: se reciben valores para cada uno de las variables de entrada, y es cuando se realiza la inferencia. El método de agregación elegido es aplicado, recibiendo como respuesta un conjunto difuso discreto, aplicado solo a los valores de la muestra seleccionada. Por último, es realizado el método de desdifusificación para obtener un resultado escalar. Este proceso de evaluación además de retornar dicho resultado devuelve en conjunto difuso representado y la muestra para la cual está definida este conjunto para cada variable de salida especificada en el conjuntos de reglas.

III. Problema a Solucionar

El problema elegido para analizar y validar el sistema propuesto es un sistema de autofoco.
Para ejecutar este ejemplo, simplemente se ejecuta el fichero `sample.py`, en donde se pedirán los valores de las variables de entrada, el método de agregación a usar y el método de desdifusificación.

Como salida se generarán varios gráficos ubicados en `img` y se imprimirá la plausibilidad de las variables de salida.
Las cámaras usualmente están equipadas con una característica de auto-foco que estima la distancia al centro de la vista del buscador. Sin embargo, este método puede que no funcione todo el tiempo, como el objeto de interés puede que no siempre esté en el centro de la vista. Mejores resultados de enfoque pueden ser obtenidos usando varias medidas de distancia a partir de la escena. Luego, se tiene como objetivo principal determinar la distancia del objeto usando 3 medidas de distancia para un sistema de enfoque automático de una cámara.

I. Definición de las variables de entrada

Las entradas del sistema de inferencia difuso serán 3 medidas de distancia puntos ubicados a la izquierda (`left`), centro (`center`) y la derecha (`right`) de la vista del buscador.
Cada variable de entrada, representando la distancia, tiene 3 conjuntos difusos:

- *Near*
- *Medium*
- *Far*

Como se puede observar en la Figura III.1 fueron usadas las funciones triangulares y trapezoidales como funciones de pertenencia.

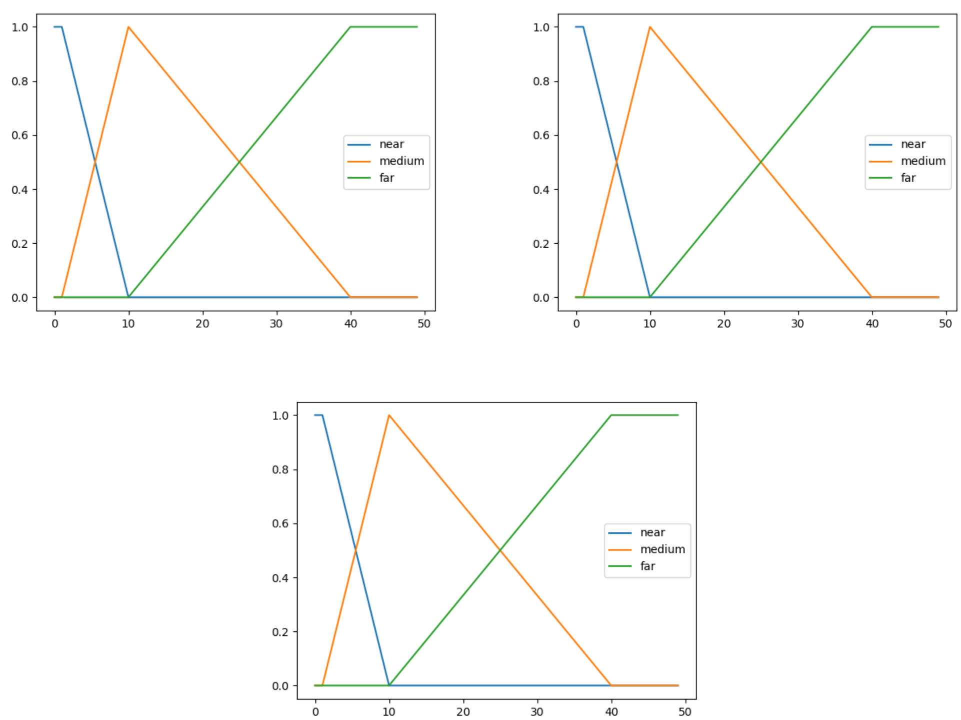

Figura III.1: Conjuntos difusos de las variables de entrada

II. Definición de las variables de salida

Como salida se tienen la plausibilidad de los valores asociados con estos 3 puntos: `plausability of left`, `plausibility of center` y `plausability of right`. El punto con la mayor plausibilidad es considerado como el punto de interés.
Para cada una de estas variables de salida, representando la plausibilidad, se tienen 4 conjuntos difusos:

- *Low*
- *Medium*
- *High*
- *Very High*

Sus conjuntos difusos se muestran en la figura III.2.

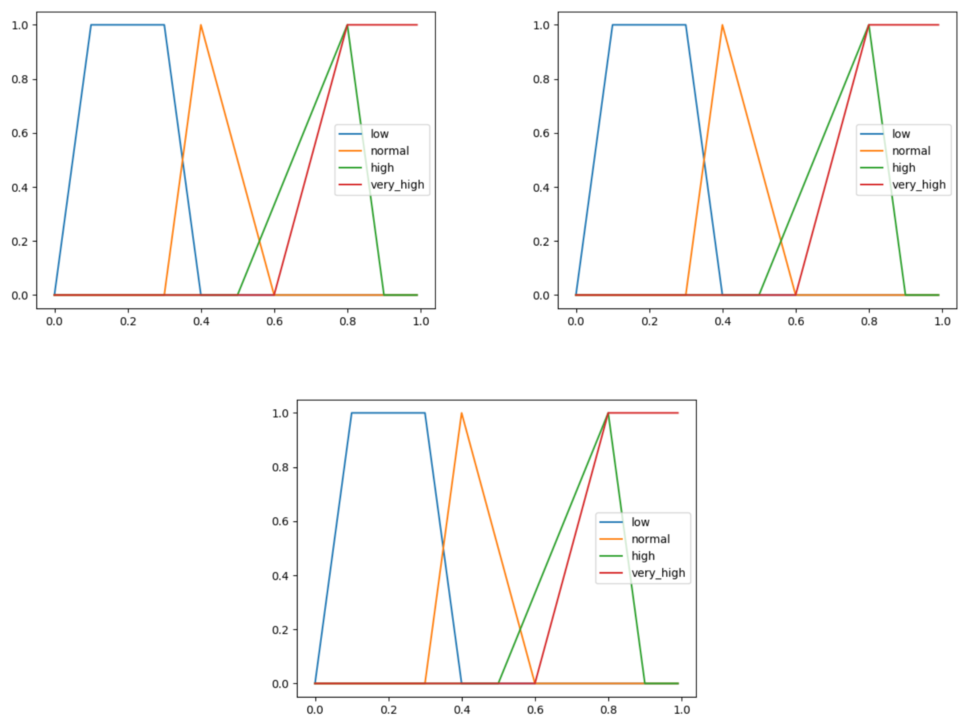

Figura III.2: Conjuntos difusos de las variables de salida

III. Reglas de inferencia

El principio central para establecer las reglas de este sistema de autoenfoque es que la probabilidad de que un objeto esté a media distancia (típicamente 10 metros) es alta, y se convierte muy baja a medida que la distancia aumenta (digamos, más de 40 metros).
Para cada una de las variables de salida, se crea un bloque de reglas, estas se procesan en el sistema de inferencia de forma individual. Luego se elige como objeto de interés aquel que tenga

mayor plausibilidad.

Primer Bloque. Plausibilidad de la derecha:

1. Si *left* es *near* entonces *plausibility of left* is *medium*
2. Si *left* es *medium* entonces *plausibility of left* is *high*
3. Si *left* es *far* entonces *plausibility of left* is *low*
4. Si *left* es *near* y *center* es *near* entonces *plausibility of left* es *low*
5. Si *left* es *medium* y *center* es *medium* entonces *plausibility of left* es *low*

Segundo Bloque. Plausibilidad del centro:

1. Si *center* es *near* entonces *plausibility of center* es *normal*
2. Si *left* es *near* y *center* es *near* y *right* es *near* entonces *plausibility of center* es *high*
3. Si *center* es *far* entonces *plausibility of center* es *low*
4. Si *left* es *far* y *center* es *far* entonces *plausibility of center* es *high*
5. Si *left* es *medium* entonces *plausibility of center* es *high*
6. Si *left* es *medium* y *center* es *far* entonces *plausibility of center* es *low*
7. Si *right* es *medium* y *center* es *far* entonces *pc* es *low*
8. Si *left* es *medium* y *center* es *medium* y *right* es *medium* entonces *plausibility of center* es *very_high*

Tercer Bloque. Plausibilidad de la izquierda:

1. Si *right* es *near* entonces *plausibility of right* es *normal*
2. Si *right* es *medium* entonces *plausibility of right* es *high*
3. Si *right* es *far* entonces *plausibility of right* es *low*
4. Si *right* es *near* y *center* es *near* entonces *plausibility of right* es *low*
5. Si *right* es *medium* y *center* es *medium* entonces *plausibility of right* es *low*

IV. Consideraciones

Después de varias ejecuciones se llegó a la conclusión que la plausibilidad del centro se convierte en alta cuando la distancia en el centro es alrededor de 10 metros, una distancia que definimos ser *Medium* en la definición de las variables de entrada. Se vuelve menor cuando la distancia aumenta, especialmente cuando la distancia a la izquierda es *Medium*.
La plausibilidad de la izquierda se comporta de una manera similar a la de la derecha. Ambas son altas cuando la distancia en ese punto es alrededor de 10 metros, una distancia que está definida como *Medium*, excepto en el caso en el que la distancia en el centro esté además alrededor de los 10 metros. En ese caso, elegimos el centro como el objeto deseado.
Todas estas observaciones pueden derivarse del conjunto de reglas definidas, lo que valida de cierta manera la efectividad de nuestro sistema.