1. Marco teórico

1.1. Lógica Difusa

La necesidad y uso de lógica de niveles múltiples existe desde los tiempos de Aristóteles, cuando utilizaban la lógica bivaluada, es decir, consideraban sólo dos valores de verdad para las expresiones lógicas: cierto o falso, con los valores numéricos 0 y 1 respectivamente. William de Occam, alrededor de los siglos XIII y XIV apoyaba la lógica de dos estados pero especulaba sobre cual podría ser el valor de verdad de la expresión “si p entonces q” si una de las dos variables no fuera ni verdadera ni falsa. Más adelante, durante el periodo de 1878 a 1956, Lukasewics propuso una lógica de tres niveles como “verdadero” (1), “falso” (0) y un “neutro” (1/2), que representaba medio verdad, medio falso. En tiempos subsecuentes, en China y otras partes del mundo, se continuó con la noción de lógica multivaluada y fue Lofti Zadeh, en su artículo “Fuzzy Sets” de 1965, quien terminó la tarea siguiendo las especulaciones de estudiosos de la lógica anteriores y mostrando que a lo que él llamó “conjuntos difusos” eran las bases de cualquier lógica, independientemente del número de niveles de verdad que se consideraran. [1]

La lógica difusa, es esencialmente una lógica multivaluada que extiende a la clásica. Esta última impone a sus enunciados únicamente valores falso o verdadero.

En el mundo existe mucho conocimiento ambiguo e impreciso por naturaleza. El razonamiento humano con frecuencia actúa con este tipo de información, es decir, el razonamiento humano utiliza valores de verdad que no necesariamente son “tan deterministas”. Las lógicas difusas procuran crear aproximaciones matemáticas en la resolución de ciertos tipos de problemas. Pretenden producir resultados exactos a partir de datos imprecisos, por lo cual son particularmente útiles en aplicaciones electrónicas o computacionales. El adjetivo “difuso” aplicado a ellas se debe a que los valores de verdad no deterministas utilizados en ellas tienen, por lo general, una connotación de incertidumbre. Un vaso medio lleno, independientemente de que también esté medio vacío, no está lleno completamente ni está vacío completamente. Qué tan lleno puede estar es un elemento de incertidumbre, es decir, de difusidad, entendida esta última como una propiedad de indeterminismo.

La lógica difusa fue investigada, por primera vez, a mediados de los años sesenta en la Universidad de Berkeley (California) por el Ingeniero Lotfi A. Zadeh, cuando se dio cuenta de lo que él llamó principio de incompatibilidad: “Conforme la complejidad de un sistema aumenta, nuestra capacidad para ser precisos y construir instrucciones sobre su comportamiento disminuye hasta el umbral más allá del cual, la precisión y el significado son características excluyentes”. Introdujo entonces el concepto de conjunto difuso (Fuzzy Set) bajo el que reside la idea de que los elementos sobre los que se construye el pensamiento humano no son números sino etiquetas lingüísticas. La lógica difusa permite representar el conocimiento común, que es mayoritariamente del tipo lingüístico cualitativo y no necesariamente cuantitativo, en un lenguaje matemático a través de la teoría de los conjuntos difusos y funciones características asociadas a ellos. Permite trabajar a la vez con datos numéricos y términos lingüísticos; los términos lingüísticos son inherentemente menos precisos que los datos numéricos pero en muchas ocasiones aportan una información más útil para el razonamiento humano.

El aspecto central de los sistemas basados en la teoría de la lógica difusa es que, a diferencia de los que se basan en la lógica clásica, tienen la capacidad de reproducir aceptablemente los modos usuales del razonamiento, considerando que la certeza de una proposición es una cuestión de grado. Más formalmente se dice que la lógica es la ciencia de los principios formales del razonamiento aproximado, considerando el razonamiento preciso (lógica clásica) como caso límite. Así pues, las características más atractivas de la lógica difusa son su flexibilidad, su tolerancia con la imprecisión, su capacidad para modelar problemas no lineales y su base en el lenguaje natural. [1] [2]

1.1.1. Teoría de conjuntos difusos.

Un conjunto clásico se define como una colección de elementos que existen dentro de un Universo. Así, si el universo consta de los números enteros no negativos menores que 10:

$$A=\{0,1,2,3,4,5,6,7,8,9\}$$

Entonces se pueden definir algunos conjuntos como, por ejemplo:

$$B=\{0,2,4,6,8\}$$

$$C=\{1,3,5,7,9\}$$

$$D=\{1,4,7\}$$

entre otros.

Con estas definiciones se ha establecido que cada uno de los elementos del Universo pertenecen o no a un determinado conjunto. Por lo tanto, cada conjunto puede definirse completamente por una función de pertenencia, que opera sobre los elementos del Universo, y que le asigna un valor de 1 si el elemento pertenece al conjunto, y de 0 si no pertenece.

La necesidad de trabajar con conjuntos difusos surge del hecho que existen conceptos que no tienen límites claros. Un conjunto difuso se encuentra asociado por un valor lingüístico que está definido por una palabra, etiqueta lingüística o adjetivo. En los conjuntos difusos la función de pertenencia puede tomar valores del intervalo entre 0 y 1, y la transición del valor entre cero y uno es gradual y no cambia de manera instantánea como pasa con los conjuntos clásicos. Un conjunto difuso en un universo en discurso puede definirse como lo muestra la siguiente ecuación:

$$A = \{(x, \mu_A(x)) | x \in U\}$$

Donde $\mu_A(x)$ es la función de pertenencia de la variable *x*, y *U* es el universo. Cuanto más cerca este la pertenencia del conjunto A al valor del 1, mayor será la pertenencia de la variable x al conjunto A, esto se puede ver en la figura 1.1.

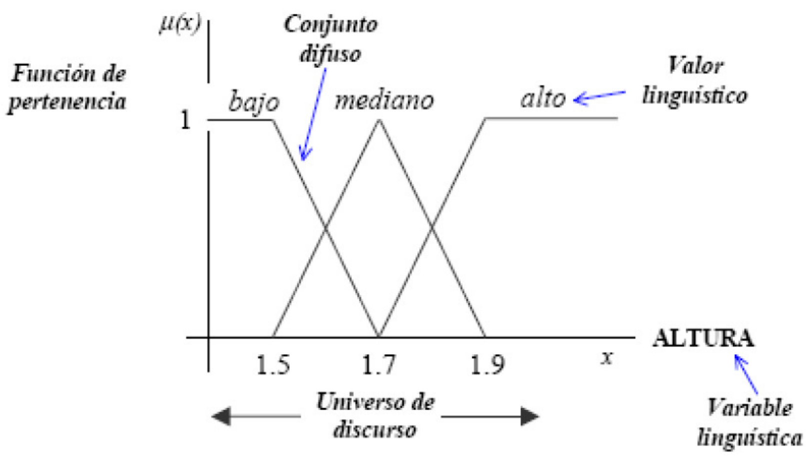


Figura 1.1. Ejemplo de Conjuntos Difusos.

Muchos conceptos de teoría clásica de conjuntos pueden hacerse extensivos a los conjuntos difusos, otros son exclusivamente inherentes a la teoría de los conjuntos difusos. Algunos de los más utilizados son los siguientes:

- El soporte de un conjunto difuso A en el universo de discurso U es un conjunto "crisp" (numérico) que contiene todos los elementos de U que tiene un valor de pertenencia distinto de cero en A, esto es,

$$sop(x) = \{x \in U | \mu_A(x) > 0\}$$

Si el soporte de un conjunto difuso no contiene ningún elemento se tiene un conjunto difuso vacío. Si el soporte de un conjunto difuso es un solo punto se tiene lo que se conoce como “singleton” difuso.

- El conjunto de cruce de conjunto difuso es el punto de U cuyo valor de pertenencia al conjunto es igual a 0.5.
- Dos conjuntos difusos A y B son iguales si y sólo si sus funciones características $\mu_A(x)$ y $\mu_B(x)$ son iguales
- El conjunto difuso B contiene al conjunto difuso A, esto es $A \subset B$, si y sólo si $\mu_A(x) \leq \mu_B(x)$ para todo $x \in U$

Variables lingüísticas

Una Variable Lingüística es aquella cuyos valores son palabras o sentencias que van a enmarcarse en un lenguaje predeterminado. Para estas variables lingüísticas se utilizará un nombre y un valor lingüístico sobre un Universo de Discurso. Además, podrán dar lugar a sentencias generadas por reglas sintácticas, a las que se les podrá dar un significado mediante distintas reglas semánticas.

En 1973, el Profesor Lofti Zadeh propuso un concepto de variables lingüísticas o difusas. Se piensa en ellas como los objetos lingüísticos o palabras, más que en números. La entrada del sensor es un sustantivo, por ejemplo, “temperatura”, “velocidad”, “flujo”, “presión”, etc. Dado que un error es sólo la diferencia de valores, no se puede pensar en él de la misma manera. Las mismas variables lingüísticas son adjetivos que modifican las variables (por ejemplo: error “grande y positivo”, error “cero”, error “pequeño y negativo” y error “grande y negativo”). Como mínimo, uno podría simplemente tener variables “positivas”, ”cero” y “negativas” para cada uno de los parámetros. Rangos adicionales como “muy grandes” y “muy pequeñas” podrían también ser agregadas para extender la respuesta a condiciones excepcionales y no lineales, pero no son necesarias en sistemas básicos.

El centro de las técnicas de modelado difuso es la idea de variable lingüística. Desde su raíz, una variable lingüística es el nombre de un conjunto difuso. Si se tiene un conjunto difuso llamado ''largo'' éste es una simple variable lingüística y puede ser empleada como una regla-base en un sistema basado en la longitud de un proyecto en particular. Si la duración-proyecto es larga entonces la terminación de tareas es DECRECIENTE. Una variable lingüística encapsula las propiedades de aproximación o conceptos de imprecisión en un sistema y da una forma de calcular adecuada. Esto reduce la aparente complejidad de describir un sistema que debe concordar con su semántica. Una variable lingüística siempre representa un espacio difuso.

Lo importante del concepto de variable lingüística es su estimación de variable de alto orden más que una variable difusa. En el sentido de que una variable lingüística toma variables difusas como sus valores. En el campo de la semántica difusa cuantitativa al significado de un término "x" se le representa como un conjunto difuso $\mu(x)$ del universo de discusión. Desde este punto de vista, uno de los problemas básicos en semántica es que se desea calcular el significado de un término compuesto.

La idea básica sugerida por Zadeh es que una etiqueta lingüística tal como ''muy'', ''más o menos'', ''ligeramente'', etc... puede considerarse como un operador que actúa sobre un conjunto difuso asociado al significado de su operando. Por ejemplo en el caso de un término compuesto ''muy alto'', el operador ''muy'' actúa en el conjunto difuso asociado al significado del operando ''alto''. Una representación aproximada para una etiqueta lingüística se puede lograr en términos de combinaciones o composiciones de las operaciones básicas. Es importante aclarar que se hará mayor énfasis en que estas representaciones se proponen principalmente para ilustrar el enfoque, más que para proporcionar una definición exacta de las etiquetas lingüísticas.

Zadeh también considera que las etiquetas lingüísticas pueden clasificarse en dos categorías que informalmente se definen como sigue:

- Tipo I: las que pueden representarse como operadores que actúan en un conjunto difuso: ''muy'', ''más o menos'', ''mucho'', ''ligeramente'', ''altamente'', ''bastante'', etc. y,
- Tipo II: las que requieren una descripción de cómo actúan en los componentes del conjunto difuso (operando): ''esencialmente'', ''técnicamente'', ''estrictamente'', ''prácticamente'', ''virtualmente''.

En otras palabras, las variables lingüísticas pueden ser caracterizadas como operadores más que construcciones complicadas sobre las operaciones primitivas de conjuntos difusos.

Funciones de pertenencia

La función de pertenencia proporciona una medida del grado de similaridad de un elemento de U con el conjunto difuso. La forma de la función de pertenencia utilizada depende del criterio aplicado en la resolución de cada problema y cambia en función del punto de vista del usuario. La única condición que debe cumplir una función de pertenencia es que tome valores entre 0 y 1 con continuidad. Las funciones de pertenencia más comúnmente utilizadas por su simplicidad matemática y su manejabilidad son: triangular (Figura 1.2), trapezoidal (Figura 1.3), singleton (Figura 1.4).

$$\mu(x)\begin{cases} 0 & para\ x \leq a \\ \dfrac{x-a}{b-a} & para\ a < x \leq b \\ \dfrac{x-b}{c-b} & para\ b < x \leq c \\ 0 & para\ x > c \end{cases}$$

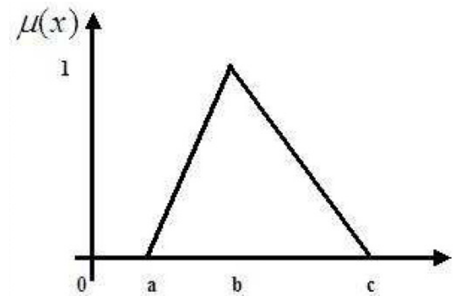


Figura 1.2.Función Triangular

$$\mu(x)\begin{cases} 0 & para\ x \leq a \\ \dfrac{x-a}{b-a} & para\ a < x \leq b \\ 1 & para\ b < x \leq c \\ \dfrac{x-c}{d-c} & para\ c < x \leq d \\ 0 & para\ x > d \end{cases}$$

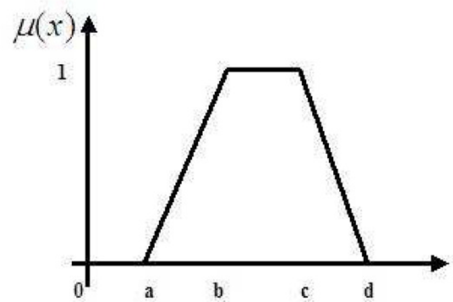


Figura 1.3.Función Trapezoidal

$$\mu(x)\begin{cases} 0 & para\ x < a \\ m & para\ x = a \\ 0 & para\ x > a \end{cases}$$

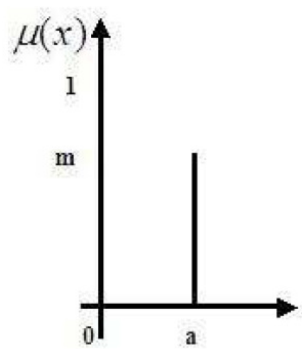


Figura 1.4.Función Singleton

Conceptualmente existen dos aproximaciones para determinar la función de pertenencia asociada a un conjunto: la primera aproximación está basada en el conocimiento humano de los expertos, y la segunda aproximación es utilizar una colección de datos para diseñar una función. El número de funciones de pertenencia asociadas a una misma variable es elegido por el experto: a mayor número de funciones de pertenencia se tiene mayor resolución pero también mayor complejidad computacional; además estas funciones pueden estar superpuestas o no, el hecho de estar una sobre otra pone de manifiesto un aspecto clave de la lógica difusa: una variable puede tener diferentes grados de pertenencia de varios conjuntos difusos, es decir el vaso puede estar medio lleno y medio vacío a la vez.[5]

1.1.2. Operaciones entre conjuntos difusos.

Las tres operaciones básicas entre conjuntos clásicos, Unión, Intersección y Complemento, se definen también para los conjuntos difusos, intentando mantener el significado de tales operaciones. La definición de estas operaciones se hace empleando el concepto de función de pertenencia de los conjuntos.

Dados dos conjuntos difusos A y B en el mismo universo X, con funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$ respectivamente, se pueden definir las siguientes operaciones básicas:

Unión. La función de pertenencia de la unión de A y B se define como:

$$\boldsymbol{\mu_{A\cup B} = max\{(\mu_A(x), \mu_B(x))\}}$$

Intersección. La función de pertenencia de la intersección de A y B es:

$$\boldsymbol{\mu_{A\cap B} = min\{(\mu_A(x), \mu_B(x))\}}$$

Complemento. La función de pertenencia del complemento de A se define como:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)$$

1.1.3. Principios de lógica difusa.

Es sabido que la teoría de conjuntos, el álgebra booleana y la lógica clásica son isomorfas, bajo transformaciones adecuadas. Esto significa que tienen una estructura subyacente similar, y que, por tanto, las definiciones que se hagan en una cualquiera de las tres teorías se puede llevar a las otras dos, mediante transformaciones adecuadas. La Tabla 1 muestra la correspondencia de algunos operadores.

Teoría de Conjuntos	Álgebra Boleana	Lógica Tradicional
Intersección	Conjunción	AND
Unión	Disyunción	OR
Complemento	Negación	NOT

Tabla 1. Correspondencia entre operadores de la Teoría de Conjuntos, el Álgebra Booleana y la Lógica Tradicional.

Ahora bien, el razonamiento lógico consiste en la combinación de proposiciones para producir nuevas proposiciones; así, la combinación de las proposiciones "X es A" y "Y es B" mediante el operador AND da como resultado la proposición "X es A AND Y es B". La tabla 1 sugiere que puede representarse esta combinación mediante un operador análogo a la Intersección de Conjuntos.

Lo anterior es posible porque en la lógica tradicional toda proposición puede tener uno de dos valores: verdadero o falso, lo que corresponde en la teoría de conjuntos clásicos a los únicos dos valores que puede tomar la función de pertenencia para cualquier conjunto: 1 ó 0. Ahora bien, en lógica difusa una proposición puede representarse por un conjunto difuso: "X es A" corresponde a un conjunto A con función de pertenencia $\mu_A(x)$, mientras que "Y es B" corresponde a un conjunto B con función de pertenencia $\mu_B(x)$, y la combinación de estas dos proposiciones con el operador AND, es decir la proposición "X es A AND Y es B" corresponde a un nuevo conjunto difuso A AND B con función de pertenencia.

$$\mu_{AandB}(x,y) = \min(\mu(x), \mu_B(y))$$

En donde se ha utilizado el operador *min* para efectuar la intersección de los dos conjuntos. [5]

1.1.4. Operador de implicación.

Debe Hacerse un análisis especial con el operador lógico de implicación =>, que combina dos proposiciones con la expresión *SI... ENTONCES... (IF ...THEN...)*, y que es el fundamento de las inferencias realizadas en sistemas de lógica difusa.

Ante todo, conviene precisar la importancia del operador => consiste en encontrar una forma de interpretar proposiciones semejantes a las utilizadas en la experiencia común para describir conocimientos. Es decir, encontrar un camino matemático para evaluar proposiciones como las siguientes: *"Si las vibraciones son altas Entonces el rodamiento está desgastado"*, o *"Si los ingresos del cliente son bajos Entonces su capacidad de endeudamiento es poca"*.

Ahora bien, la implicación => de la lógica tradicional tiene una tabla de verdad que se muestra en la Tabla 2.

p	**q**	**P=>q**
Verdad	Verdad	Verdad
Verdad	Falso	Falso
Falso	Verdad	Verdad
Falso	Falso	Verdad

Tabla 2 Tabla de verdad de la implicación lógica tradicional

Esta tabla de verdad puede obtenerse también con los operadores básicos Conjunción, Disyunción y Negación.

Para combinar dos proposiciones "X es A" y "Y es B" en la forma "IF X es A THEN Y es B", debe representarse a cada una de dichas proposiciones por conjuntos difusos con funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$ respectivamente, y entonces la proposición combinada estará representada por un conjunto difuso A=>B, cuya función de pertenencia estará dada por:

$$\mu_{A\Rightarrow B}(x,y) = 1 - \min\left(\mu_A(x), \left(1 - \mu_B(y)\right)\right) \quad \text{ó } bien$$

$$\mu_{A\Rightarrow B}(x,y) = \max(1 - \mu_A(x), \mu_B(y))$$

No obstante, las expresiones anteriores (llamadas implicaciones lógicas o implicaciones IF-THEN) no son necesariamente las más útiles para efectuar inferencias, particularmente en aplicaciones de ingeniería. La razón puede hallarse revisando la Tabla 2: La implicación de la lógica tradicional es verdadera en tres condiciones, y sólo es falsa si la primera proposición es verdadera y la segunda es falsa, lo que puede interpretarse con la máxima "La verdad nunca implica falsedad".

La tabla de verdad de la implicación indica en qué condiciones un razonamiento es formalmente correcto, pero no necesariamente útil. Por ejemplo:

"Si 1=2 Entonces 3=3" es una implicación formalmente correcta, porque una falsedad (1=2) puede implicar una verdad (3=3), y para ello basta con sumar al lado izquierdo 2 y al lado derecho 1 (recordando que se partió de 1=2).

De igual forma la proposición "Si 1=2 Entonces 2=3" también es formalmente correcta, porque una falsedad puede implicar una falsedad, y para ello basta con sumar 1 a cada lado de la igualdad.

Los dos ejemplos anteriores son formalmente correctos, pero qué utilidad puede extraerse de ellos en aplicaciones de ingeniería. En realidad sólo implicaciones en las que ambas proposiciones sean verdaderas pueden tener utilidad práctica, y esto es así porque las relaciones causa-efecto son las que interesan en Ingeniería, y no el formalismo de una implicación

1.1.5 Inferencia en Lógica difusa

La Inferencia lógica consiste en la combinación de proposiciones para producir nuevas proposiciones. Así, al combinar la proposición *"X es A"* con la proposición *"IF X es A THEN Y es B"*, se puede inferir la proposición *"Y es B"* (ver figura 1.5).

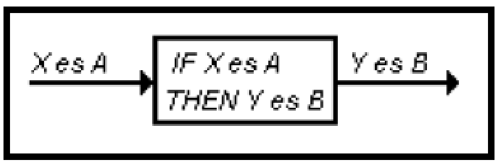


Figura 1.5. Inferencia en Lógica Tradicional

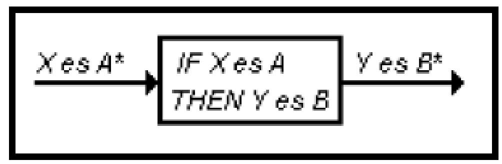


Figura 1.6. Inferencia en Lógica Difusa

Una inferencia como la mencionada en el párrafo anterior sólo es posible en la lógica tradicional si la primera proposición (*"X es A"*) es idéntica a la primera parte de la segunda proposición ("*(IF) X es A"*); sin embargo, en la lógica difusa estas dos proposiciones no necesariamente deben ser idénticas, ya que las fronteras de los conjuntos no son precisas. Así, al combinar la proposición *"X es A*"* con la proposición *"IF X es A THEN Y es B"*, puede obtenerse la proposición *"Y es B*"* (ver figura 1.6).

La combinación de estas proposiciones para efectuar la inferencia tiene su soporte matemático en la *Extensión Cilíndrica* y en la *Composición de Relaciones*, la figura 1.7 muestra gráficamente cómo puede interpretarse esta inferencia. [4]

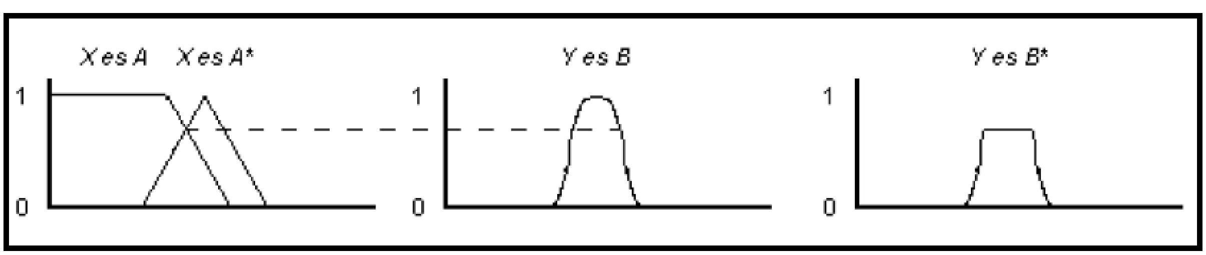
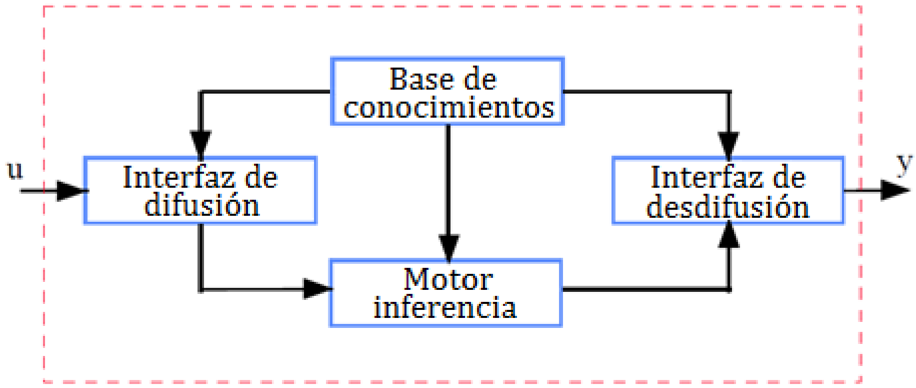


Figura 1.7. Representación gráfica de los mecanismos de Inferencia en Lógica Difusa

1.1.6. Modelos difusos lingüísticos

Estos modelos se basan en un conjunto de reglas heurísticas donde las variables lingüísticas de las entradas y salidas se representan por conjuntos difusos.

La siguiente figura muestra los principales componentes de un modelo difuso lingüístico: interfaz de difusión, base de conocimiento, motor de inferencia e interfaz de desdifusión.



Interfaz de difusión. Este elemento transforma las variables de entrada del modelo *(u)* en variables difusas. Para esta interfaz se deben tener definidos los rangos de variación de las variables de entrada y los conjuntos difusos asociados con sus respectivas funciones de pertenencia.

Base de conocimientos. Contiene las reglas lingüísticas del control y la información referente a las funciones de pertenencia de los conjuntos difusos.

Estas reglas lingüísticas, tienen típicamente la siguiente forma:

Si u_1 es *A* y u_2 es *B* entonces y es *C*

donde A, B y C son los conjuntos difusos de las variables de entrada *u1* y *u2*, y de la variable de salida y respectivamente.

Existen varias formas de derivar las reglas, entre las que destacan las basadas en:

- La experiencia de expertos y el conocimiento de ingeniería de control. La base de reglas se determina a partir de entrevistas con el operador o a través del conocimiento de la dinámica del proceso.
- La modelación del proceso. Los parámetros de la base de conocimiento se obtienen a partir de datos de entrada y salida del proceso.

Motor de inferencia. Una vez obtenidos los valores difusos, el objetivo de la inferencia difusa es transportarlos de la premisa a la conclusión de cada regla. Esto resulta en un conjunto difuso para cada variable de salida de cada regla. En este proceso, los métodos más usados son Mínimo y Producto.

- Producto

En este método, la función de pertenencia de salida graduada por el grado de pertenencia calculado para la premisa de la regla.

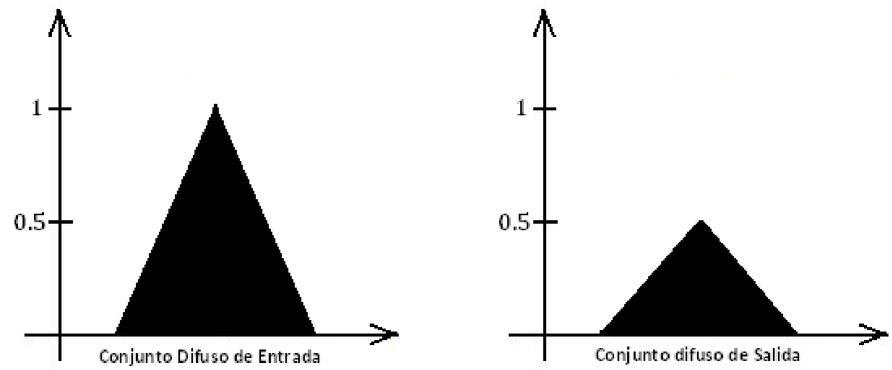


Figura 1.8.Conjuntos difusos de entrada y salida aplicando el método de inferencia difuso: Producto

Por ejemplo si el grado de pertenencia es de 0.5 y el conjunto difuso es triangular la figura que se obtendrá es la siguiente:

- Mínimo

En el método de inferencia Mínimo, la función de pertenencia de salida es recortada a la altura correspondiente al grado de verdad calculado para la premisa de la regla, La forma de la salida resultante se afecta por este método formando un trapecio.

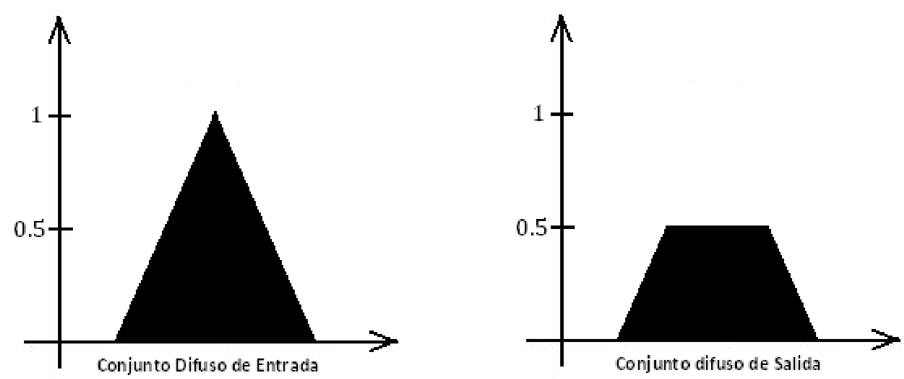


Figura 1.9.Conjuntos difusos de entrada y salida aplicando el método de inferencia difuso: Mínimo

Esta nueva figura del conjunto difuso de salida es la forma cómo influye específicamente esta regla en la variable de salida.

Combinación de reglas. En éste paso todos los subconjuntos difusos de salida asignados para cada variable de salida de las diferentes reglas se combinan en un solo conjunto difuso.

Para determinar la forma del conjunto difuso de salida a partir del conjunto de respuestas obtenidas de las diversas reglas, Lógica Difusa cuenta con los métodos Máximo y Suma.

- Máximo

Este método evalúa el contorno de todos los conjuntos de salida formando un solo contorno de salida.

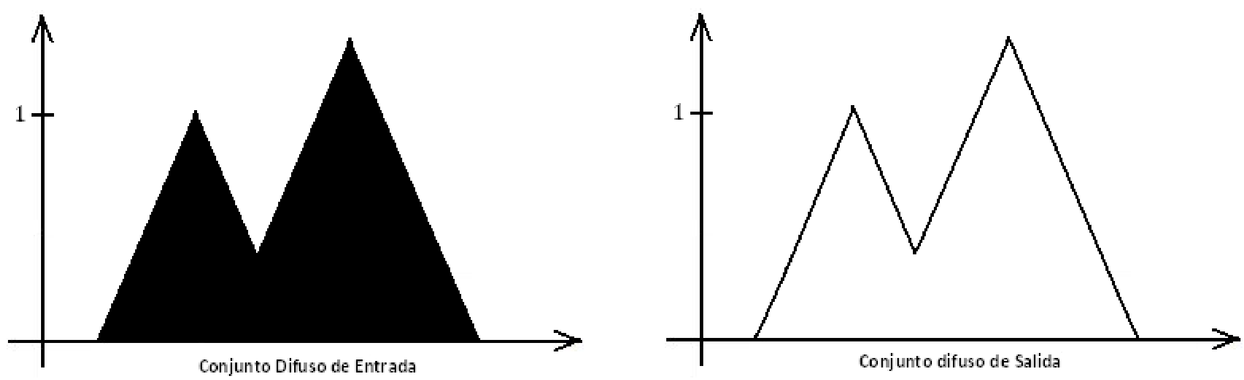


Figura 1.10.Conjuntos difusos de entrada y salida aplicando el método de combinación de reglas máximo

En este método, la combinación de los subconjuntos difusos se construye tomando el punto máximo de todos los subconjuntos difusos asignados por las reglas de inferencia utilizando el operador OR.

- Por Suma

Otra forma es la combinación por suma de los puntos máximos de los conjuntos de salida difusos. Por ejemplo es la figura 1.11, el punto que se encuentra señalado se obtiene sumando el grado de pertenencia en ese punto para cada conjunto difuso, dando como resultado 0.6+0.2=0.8.

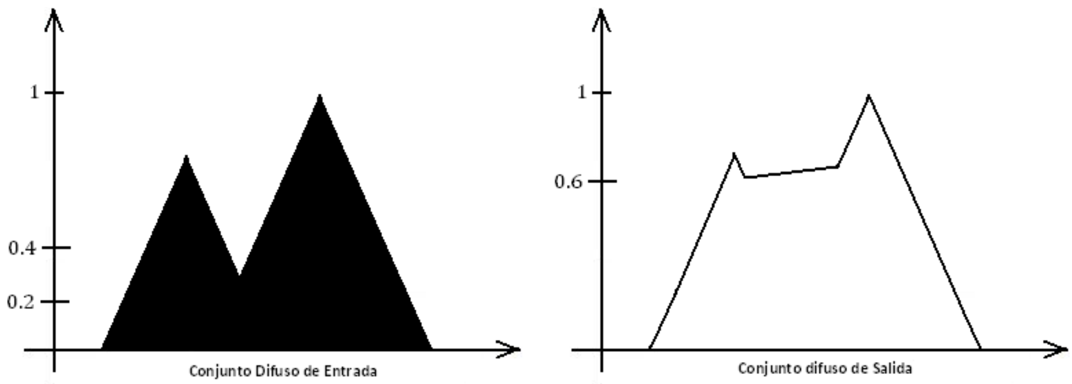


Figura 1.11.Conjuntos difusos de entrada y salida aplicando el método de combinación de reglas por suma

Interfaz de desdifusión. Este elemento provee salidas discretas y determinísticas a partir de los conjuntos difusos C' obtenidos como resultado de la inferencia.

- **Método del centroide.** Consiste en determinar el punto de equilibrio de la figura formada por los conjuntos difusos de salida, con la expresión:

$$z^* = \frac{\int x\mu(x)dx}{\int \mu(x)dx}$$

El punto de equilibrio obtenido se utiliza para representar a los conjuntos difusos de salida.

- **Método del centro de gravedad.** Este método determina el centro del área perteneciente a la combinada función de membresía. Esta operación es computacionalmente compleja y por lo tanto resultan ciclos lentos de inferencia. El punto representativo de los conjuntos de salida se determina con la expresión:

$$z^* = \frac{\sum z_i\mu(z_i)}{\sum \mu(z_i)}$$

- **Método de altura máxima.** Consiste en realizar un barrido de todos los elementos de los conjuntos difusos de salida para localizar el elemento con mayor grado de pertenencia. El elemento obtenido se utiliza para representar los conjuntos difusos de salida. Al realizar un barrido de todos los elementos de los conjuntos difusos de la figura 1.12 se determina que el elemento con mayor grado de pertenencia es *z**.

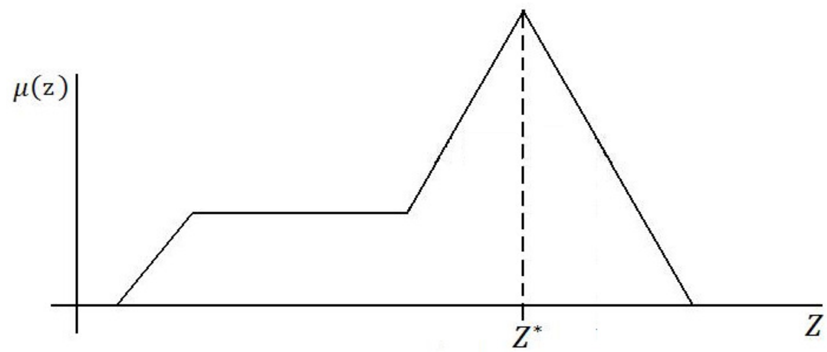


Figura 1.12. Desdifusión con el método de altura máxima.

- **Método de primero de los máximos.** Se realiza un barrido de izquierda a derecha, de los elementos de los conjuntos difusos, hasta obtener el primer elemento con valor de pertenencia máximo, el cual se utiliza para representar a los conjuntos difusos de salida. Al realizar un barrido de los elementos de los conjuntos difusos de salida en la figura 1.13, se determina que el primer valor de pertenencia máximo se encuentra en el elemento *z**.

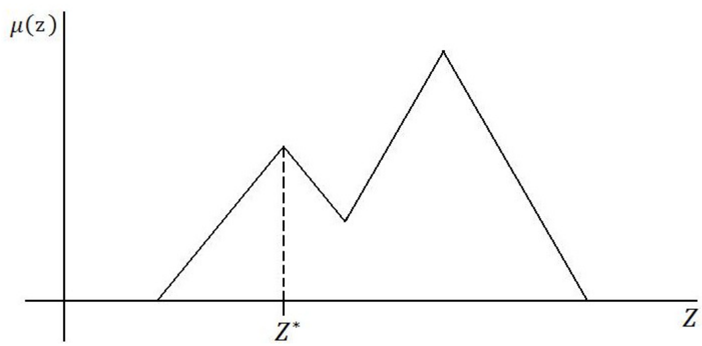


Figura 1.13. Desdifusión con el método de primero de los máximos.

- **Método de último de los máximos**. Se realiza un barrido de izquierda a derecha, de los elementos de los conjuntos difusos hasta obtener el último valor de pertenencia máximo, el cual se utiliza para representar a los conjuntos difusos de salida. Al realizar un barrido de los elementos de los conjuntos difusos de salida de la figura 1.14, se determina que el último valor de pertenencia máximo se encuentra en el elemento *z**.

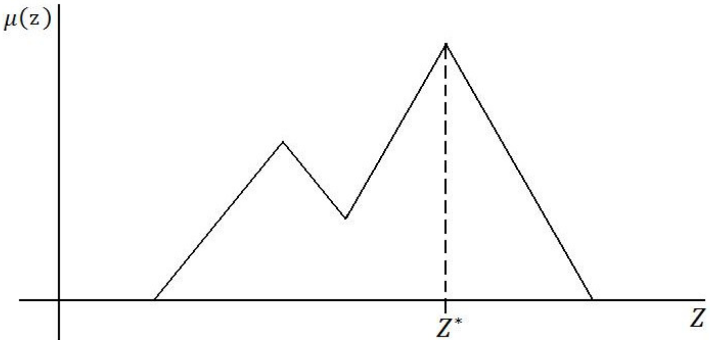


Figura 1.14. Desdifusión con el método de último de los máximos.

- **Método de promedio pesado.** Se aplica cuando los conjuntos difusos de salida están formados por figuras simétricas. Para aplicarlo se siguen los siguientes pasos:
  1. Se identifican las figuras simétricas.
  2. Se obtiene el punto (*z1, μ(z)*), localizado sobre el eje de simetría de cada figura.
  3. Se aplica la expresión:

$$z^* = \frac{\sum z_i \mu(z_i)}{\sum \mu(z_i)}$$

En los conjuntos difusos de salida de la figura 1.15 se tienen dos figuras simétricas, las cuales se representan con los puntos (z_1, $\mu(z_1)$) y (z_2, $\mu(z_2)$) ubicados en los ejes de simetría de cada una. El valor representativo de los conjuntos difusos se calcula como:

$$z^* = \frac{z_1\mu(z_1) + z_2\mu(z_2)}{\mu(z_1) + \mu(z_2)}$$

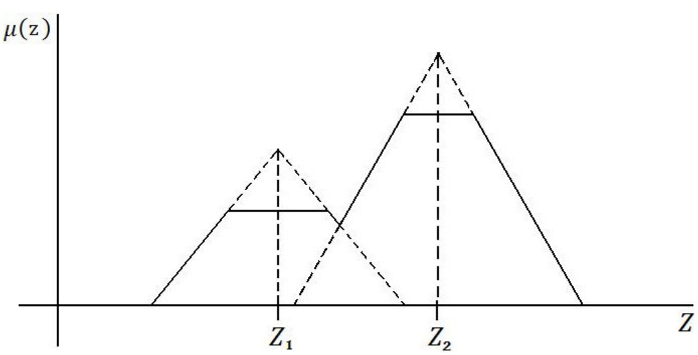


Figura 1.15. Desdifusión con el método de promedio pesado.

- **Método de centro de sumas.** Para de todos los conjuntos difusos con la expresión aplicar este método se siguen los siguientes pasos:

1. Se determina un punto representativo z_i para cada conjunto difuso.

2. Se calcula el área A_i de cada conjunto difuso.

3. Se obtiene un punto representativo de todos los conjuntos con la expresión:

$$\boldsymbol{z = \frac{\sum_{i=1}^{n} z_i A_i}{\sum_{i=0}^{n} A_i}}$$

En la figura 1.16 se tiene que z_1 y z_2 son los puntos representativos de los conjuntos difusos, donde A_1 y A_2 representan las áreas de cada conjunto, al aplicar el método de centro de sumas se obtiene que

$$z^* = \frac{z_1 A_1 + z_2 A_2}{A_1 + A_2}$$

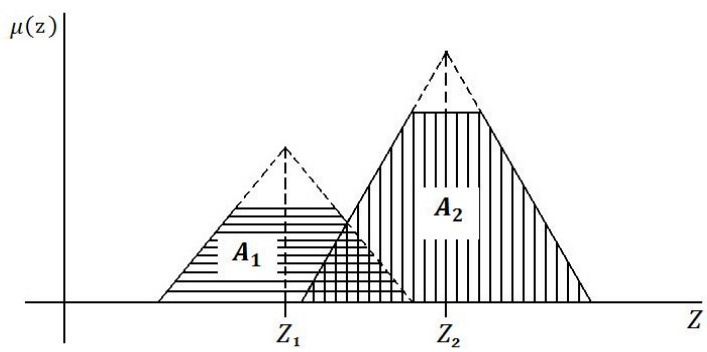


Figura 1.16. Desdifusión con el método de centro de sumas.

- **Método de promedio de máximos.** Consiste en realizar un barrido de todos los elementos de los conjuntos difusos para localizar el primero y el último elemento con mayor grado de pertenencia, representados como z_1 y z_2 respectivamente. El valor representativo z^* de los conjuntos difusos se calcula con la expresión:

$$z^* = \frac{z_1 + z_2}{2}$$

Al realizar un barrido de todos los elementos del conjunto difuso de la figura 1.17, se encuentra el primer valor máximo de pertenencia en el elemento z_1 y el último valor máximo de pertenencia en el elemento z_2. El valor representativo de los conjuntos se calcula como el valor promedio de estos dos elementos.

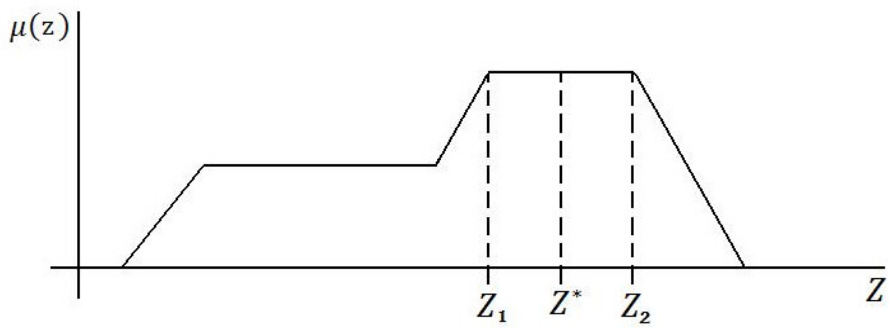


Figura 1.17. Desdifusión con el método de promedio de máximos.

- **Método de centro de área mayor.** Para aplicar este método se siguen los siguientes pasos:
  1. Se identifican los conjuntos convexos que forma la figura.
  2. Se calcula el área de cada conjunto convexo.
  3. Se comparan las áreas.
  4. Se obtiene el centroide del área mayor.

En la figura 1.18 se tienen los conjuntos convexos *A* y *B*, cuyas áreas son A_1 y A_2 respectivamente. Si al comparar las áreas se obtiene que $A_1 > A_2$, entonces el punto representativo de los conjuntos difusos se obtiene con el centroide del conjunto convexo de área mayor con la expresión:

$$z^* = \frac{\int z_{A1}\mu_{A1}(z)dz}{\int \mu(z)_{A1}dz}$$

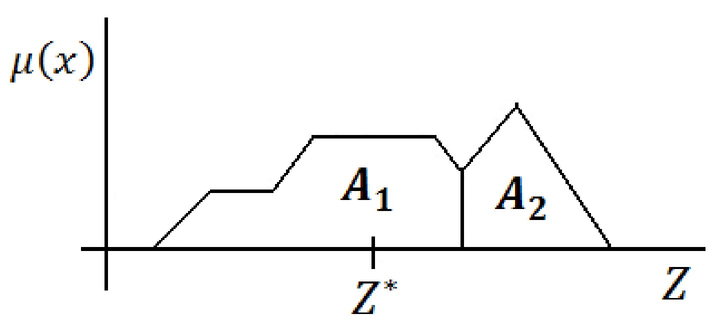


Figura 1.18. Desdifusión con el método de centro de área mayor. [4] [5] [6]

1.2. Ejemplo de Caso Resuelto con Lógica Difusa

La meta es la de ubicar el vehículo en un estacionamiento específico, en medio de otros vehículos y en forma paralela. El vehículo puede empezar de cualquier posición y orientación en el parque de estacionamiento, como puede apreciarse en la figura 1.19

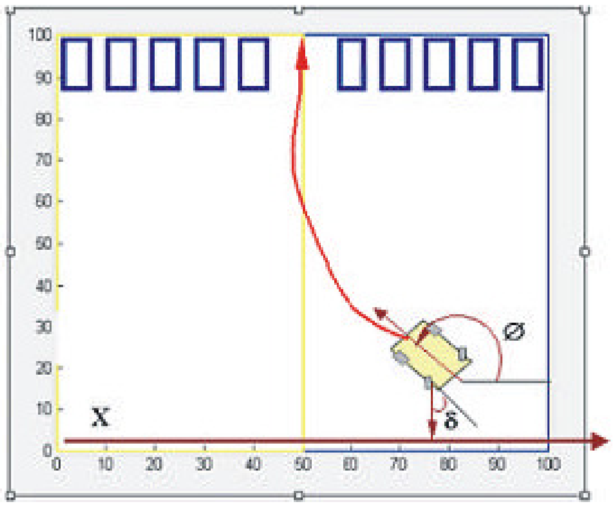


Figura 1.19. Posición Inicial del móvil.

Paso 1: Definición de las variables lingüísticas:

Según la figura 1.19, se consideran las siguientes variables:

X : Espacio horizontal donde el móvil podrá desplazarse.

Φ : Ángulo referencial del eje del vehículo, cuya dirección y sentido va de la parte delantera del vehículo hacia la parte posterior, con relación al eje horizontal; se asume que el vehículo sólo puede retroceder.

δ : Ángulo de giro de timón del vehículo, que guarda estrecha relación con el desplazamiento del vehículo, es decir, a mayor giro del timón, mayor será la curvatura en el desplazamiento del vehículo.

Del análisis de las variables se concluye que X y Φ son independientes, mientras δ, que es el ángulo de giro del timón dependerá de X y Φ. Ver figura 1.20.

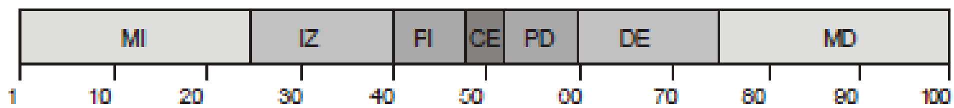
Figura 1.20. Variables Independientes y Dependientes.

Paso 2: Definición de rangos y valores lingüísticos de las variables:

a) Para la variable X (desplazamiento Horizontal del móvil):

El espacio de desplazamiento horizontal varía entre 0 y 100. Este rango se divide en siete regiones simétricamente proporcionales. A cada uno de estos segmentos se les asigna valores Lingüísticos difusos, los mismos que son totalmente simétricos respecto al segmento medio o centro. Es decir, la simetría es tanto en tamaño del segmento como en los valores lingüísticos.

A continuación se describen en detalle los valores y rangos mencionados para la variable X. Ver figura 1.21.



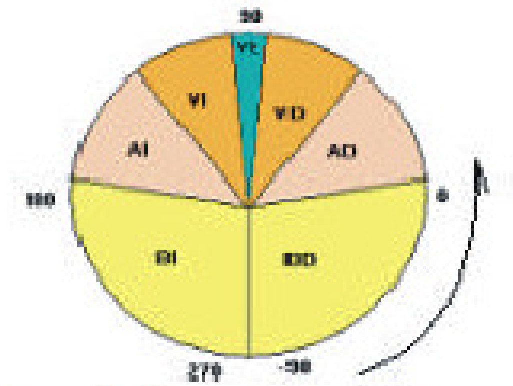
Valores	Rango	Descripción
MI	0.25	Muy a la Izquierda
IZ	25.40	A la Izquierda
PI	40.48	Un poco a la Izquierda
CE	48.52	Al Centro
PD	52.60	Un poco a la Derecha
DE	60.75	A la Derecha
MD	75.100	Muy a la Derecha

Figura 1.21. Valores y rangos para la Variable X.

b) Para la variable Φ(Angulo Referencial del Vehículo):

El móvil podrá tener una libertad de giro total es decir de 360 grados. Pero por consideraciones del problema debe existir una distribución simétrica respecto a la posición vertical del móvil, por lo tanto el rango de esta variable será desde 90 hasta los 270 grados. Este rango también es dividido en siete sectores simétricamente proporcionales, los mismos que se describen a continuación.

Para tener una mejor idea de los rangos mencionados, se pueden graficar de la siguiente manera. Ver figura 1.22.

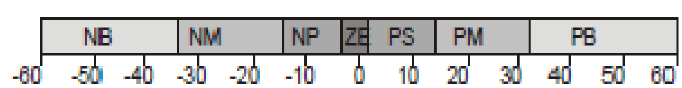


Valores	Rango	Descripción
BD	-90.5	Hacia Abajo a la Derecha
AD	5.45	Hacia Arriba a la Derecha
VD	45.81	Vertical Derecha
VE	81.99	Vertical
VI	99.135	Vertical Izquierda
AI	135.1775	Hacia Arriba a la Izquierda
BI	175.270	Hacia Abajo a la Izquierda

Figura 1.22. Valores y rangos para la Variable Φ.

c) Para la variable d (Ángulo de giro del timón):

El timón tendrá una libertad de giro entre +60 y -60 grados. Este rango también ha sido dividido en siete sectores simétricamente proporcionales, los mismos que se describen a continuación. Ver figura 1.23.



Valores	Valores Numéricos	Rango	Descripción
NB	1	-60 -34	Hacia Abajo a la Derecha
NM	2	-34 -15	Hacia Arriba a la Derecha
NP	3	-15 1.5	Vertical Derecha
ZE	4	1.5 15	Vertical
PP	5	1.5 15	Vertical Izquierda
PM	6	15 34	Hacia Arriba a la Izquierda
PB	7	34 60	Hacia Abajo a la Izquierda

Figura 1.23. Valores y rangos para la Variable δ.

Paso 3: Asignar función de pertenencia a los rangos de las variables:

La función de pertenencia la definimos en forma detallada en la figura 1.24.

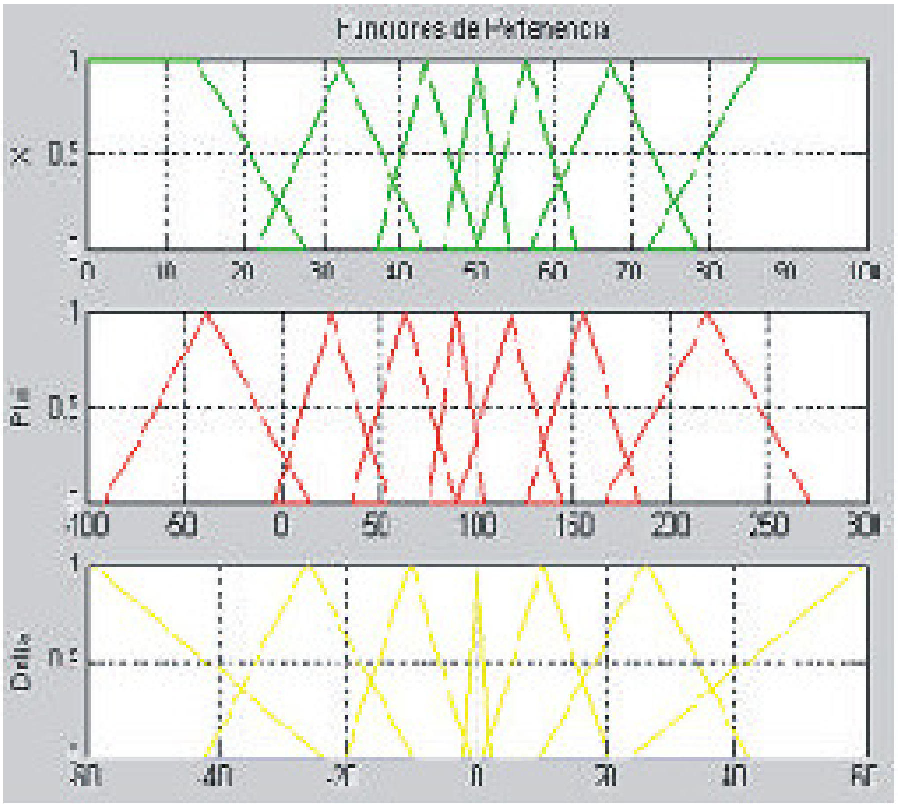


Figura 1.24. Funciones de Pertenencia para las variables X, Φ, δ.

a) Base de reglas con valores lingüísticos

La base de reglas son los comandos u órdenes de movimiento que se le asigna al móvil según la ubicación actual y el ángulo que forma con la base. Estos valores están especificados en la figura 1.25.

	MI	IZ	PI	CE	PD	DE	MD
BD	PM	PM	PM	PM	PB	PB	PB
AD	NS	PM	PB	PM	PB	PB	PB
VD	NB	NM	PB	PB	PM	PB	PB
VE	NB	NB	NM	ZE	PM	PB	PB
VI	NB	NB	NM	NB	NB	PM	PB
AI	NB	NB	NB	NM	NB	NM	NB
BI	NB	NB	NB	NM	NM	NM	NM

Figura 1.25. Matriz de Base de reglas con Valores Lingüísticos.

b) Base de reglas con valores numéricos

Los valores lingüísticos que se aprecian en esta matriz deben ser reasignados a valores numéricos para que pueda ser procesado por la computadora, esta equivalencia se aprecia en la figura 1.26.

	MI	IZ	P1	CE	PD	DE	MD
BD	6	6	6	6	7	7	7
AD	3	6	7	6	7	7	7
VD	1	2	7	7	6	7	7
VE	1	1	2	4	6	7	7
VI	1	1	2	1	1	6	7
AI	1	1	1	2	1	2	5
BI	1	1	1	2	2	2	2

Figura 1.26. Matriz de Base de reglas con Valores Numéricos.

IV. RESULTADOS

La simulación fue implementada en programa Matlab, se puede apreciar la precisión de como el móvil llega a su destino desde cualquier ubicación. Ver figura 1.27. [7]

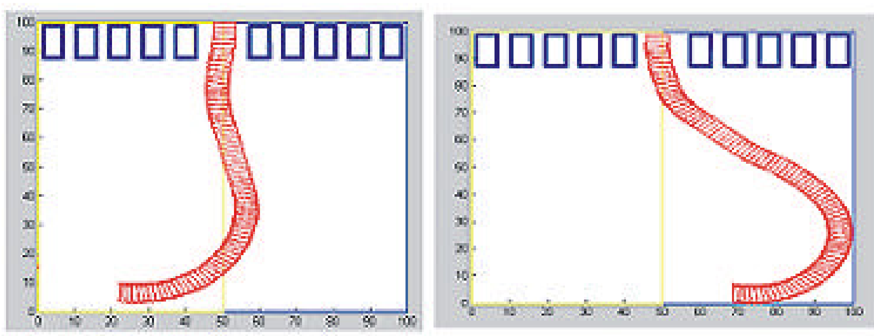


Figura 1.27. Resultado de la simulación en Matlab.

1.3. Software usado

1.3.1. Origen de Java

Sun Microsystems desarrolla el lenguaje Java en un intento de resolver simultáneamente una amplia variedad de problemas que se planteaban a los desarrolladores de software. Estos problemas eran debido a la proliferación de arquitecturas incompatibles, tanto entre las diferentes máquinas como entre los diversos sistemas operativos y sistemas de ventanas que funcionaban sobre una misma máquina; añadiendo la dificultad de crear aplicaciones distribuidas en una red como Internet.

Los lenguajes que se usaban, como C o C++, se compilaban para que los ejecutara un chip, y si se cambiaba el chip, todo el software debía compilarse de nuevo. Esta situación encarecía mucho los desarrollos y el problema era especialmente grave en el campo de la electrónica de consumo. La aparición de un chip más barato y, generalmente, más eficiente, conduce inmediatamente a los fabricantes a incluirlo en las nuevas series de sus cadenas de producción, porque por pequeña que sea la diferencia en precio, si se multiplica por la emisión masiva de los aparatos, supone un ahorro considerable. Por tanto, Gosling decidió mejorar las características de Oak y utilizarlo.

Este lenguaje se aplicó a dos proyectos relacionados con una interfaz de usuario bastante ambiciosa. Ninguno de estos proyectos se convirtió nunca en un sistema comercial, aunque fueron desarrollados enteramente en un Java primitivo y fueron como su bautismo de fuego. Pese a que finalmente FirstPerson cerró en la primavera de 1994, la historia del lenguaje no terminó.

Pese a lo que parecía ya un olvido definitivo, Bill Joy, cofundador de Sun y uno de los desarrolladores principales del Unix de Berkeley, juzgó que Internet podría llegar a ser el campo de juego adecuado para disputar a Microsoft su primacía casi absoluta en el terreno del software, y vio en Oak el instrumento idóneo para llevar a cabo estos planes. Tras un cambio de nombre y algunos cambios de diseño, el lenguaje Java fue presentado en sociedad en agosto de 1995.

1.3.2. Características de Java

El lenguaje Java cuenta con una serie de características que lo hacen interesante, las cuales se pueden resumir como sigue:

Simple

- Simple en su sintaxis, por lo que es más sencillo para programar.
- Tiene un recolector de basura que le permite al programador despreocuparse de liberar memoria.
- Elimina características que provocan errores en otros lenguajes como C y C++.

Orientado a Objetos.

- Soporta encapsulación, herencia y polimorfismo.
- Cuenta con resolución dinámica de métodos al modo de Objetive-C.
- Proporciona soporte para identificar clases en tiempo de ejecución.

Características de interconexión.

- Cuenta con extensas capacidades de interconexión mediante TCP/IP.
- Existen librerías de rutinas para acceder e interactuar con protocolos como http y ftp.

Portable.

- Implementa otros estándares de portabilidad para facilitar el desarrollo. Los enteros se representan con 32 bits en complemento a 2.
- Construye sus interfaces de usuario a través de un sistema abstracto de ventanas de forma que las ventanas puedan ser implantadas en entornos Unix, Pc o Mac.

Seguro.

- Java elimina características como los punteros o el casting implícito, mecanismo que usa para asociar datos el compilador de C y C++, con el fin de prevenir el acceso ilegal a la memoria.
- Los ambientes de ejecución de Java aplican un probador de teoremas a los programas, para detectar fragmentos de código ilegal, como lo sería el tipo de ataque Caballo de Troya.
- Cuenta con firmas digitales, que se verifican antes de crear cualquier objeto.
- Imposibilita, también, abrir algún archivo de la máquina local. Siempre que se realizan operaciones con archivos, éstas trabajan sobre el disco duro de la máquina de donde partió el applet
- No permite ejecutar ninguna aplicación nativa de una plataforma e impide que se utilicen otros ordenadores como puente, es decir, nadie puede utilizar nuestra máquina para hacer peticiones o realizar operaciones con otra.
- Sin embargo, puede ser descompilado.

Neutral a la arquitectura.

- El compilador Java genera su código a un archivo objeto de formato independiente de la arquitectura de la máquina en que se ejecutará.
- Cualquier máquina que tenga el sistema de ejecución (run-time) puede ejecutar ese código objeto, sin importar en modo alguno la máquina en que ha sido generado.

Interpretado.

- El intérprete Java puede ejecutarse en diversas plataformas
- Java es más lento que C.

Multihilo.

- Al ser Multihilo, Java permite muchas actividades simultáneas en un programa. Los hilos, a veces llamados procesos ligeros o hilos de ejecución, son pequeños procesos o piezas independientes de un gran proceso.
- Al estar estos hilos construidos en el mismo lenguaje, son más fáciles de usar y más robustos que sus homólogos en C o C++.
- El beneficio de ser multihilo consiste en un mejor rendimiento interactivo y mejor comportamiento en tiempo real. Aunque el comportamiento en tiempo real está limitado a las capacidades del sistema operativo subyacente (Unix, Windows, etc.) de la plataforma, aún supera a los entornos de flujo único de programa (single-threaded) tanto en facilidad de desarrollo como en rendimiento.

Dinámico.

- No intenta conectar todos los módulos que comprenden una aplicación hasta el mismo tiempo de ejecución. Las librerías nuevas o actualizadas no paralizarán la ejecución de las aplicaciones actuales -siempre que mantengan el API anterior.
- Simplifica el uso de protocolos nuevos o actualizados. Si su sistema ejecuta una aplicación Java sobre la red y encuentra una pieza de la aplicación que no sabe manejar, Java es capaz de traer automáticamente cualquier pieza que el sistema necesite para funcionar.
- Para evitar que los módulos de ByteCode o los objetos o nuevas clases, haya que estar trayéndolas de la red cada vez que se necesiten, Java implementa las opciones de persistencia, para que no se eliminen cuando se limpie la caché de la máquina.

1.3.3. Ventajas

Primero: No se debe volver a escribir el código si se quiere ejecutar el programa en otra máquina. Un solo código funciona para todos los browsers compatibles con Java o donde se tenga una Máquina Virtual de Java (Mac's, PC's, Sun's, etc).

Segundo: Java es un lenguaje de programación orientado a objetos, y tiene todos los beneficios que ofrece esta metodología de programación.

Tercero: Un browser compatible con Java deberá ejecutar cualquier programa hecho en Java, esto ahorra a los usuarios tener que estar insertando "plug-ins" y demás programas que a veces quitan tiempo y espacio en disco.

Cuarto: Java es un lenguaje y por lo tanto puede hacer todas las cosas que puede hacer un lenguaje de programación: Cálculos matemáticos, procesadores de palabras, bases de datos, aplicaciones gráficas, animaciones, sonido, hojas de cálculo, etc.

Quinto: Si lo que interesa son las páginas de Web, ya no tienen que ser estáticas, se le pueden poner toda clase de elementos multimedia y permiten un alto nivel de interactividad, sin tener que gastar en paquetes carísimos de multimedia. [8] [9]