

# NAVEGACIÓN DE ROBOTS MÓVILES MEDIANTE COMPORTAMIENTOS UTILIZANDO LÓGICA DIFUSA

## RESUMEN

Este artículo describe la implementación de comportamientos básicos de navegación de robots móviles utilizando lógica difusa. Entre los comportamientos implementados se describen el comportamiento SEGUIR PARED y el comportamiento SEGUIR PASILLO.

Para la implementación de dichos comportamientos se utilizó el toolbox de lógica difusa de Matlab y el modelo cinemático de la plataforma P-METIN del Grupo GIROPS. Una de las principales características de dicha implementación es la información de distancia obtenida de los sensores de dicha plataforma.

**PALABRAS CLAVES:** Lógica Difusa, P-METIN Plataforma Móvil, para el Estudio de Técnicas Inteligentes de Navegación, Sensores.

## ABSTRACT

*This article describes the implementation of basic behaviors of navigation of mobile robots using fuzzy logic. The behavior TO FOLLOW WALL and the behavior TO FOLLOW CORRIDOR are described.*

*For the implementation of these behaviors was used toolbox of fuzzy logic of Matlab and the cinematic model of platform P-METIN of GIROPS. One of the main characteristics of this implementation is the information of distance obtained from the sensors of this platform.*

**KEYWORDS:** Fuzzy logic, P-METIN: Mobile Platform, for the Study of Intelligent Techniques of Navigation, Sensors.

## 1. INTRODUCCIÓN

Los robots móviles deben ser capaces de desenvolverse en un entorno real, construir su propio mapa y navegar de manera autónoma para ello se requiere la implementación de estrategias de control inteligente, que puedan manejar la incertidumbre presentada por el entorno de trabajo, al mismo tiempo que se desempeñen en tiempo real con una relativa baja carga computacional[1].

En lo que respecta a la navegación de robots móviles, es necesario tener en cuenta que:

-En condiciones iniciales no se conoce la localización del robot móvil en el entorno de trabajo.

-El conocimiento a priori del entorno de trabajo, es en general incompleto, incierto y aproximado. La información métrica puede ser imprecisa e inadecuada.

-La información perceptual adquirida en la mayoría de los casos es poco fiable, el limitado rango de los sensores combinado con las características del ambiente (oclusión) y las condiciones adversas de observación induce a ruido y a datos imprecisos y errores en el proceso de interpretación de las medidas.

## HERNANDO PARRA

Ingeniero Mecánico; Esp.  
Profesor Asistente  
Universidad Tecnológica de Pereira  
hparra@utp.edu.co

## LUIS HERNANDO RÍOS G

Ingeniero Electrónico, Ms.C  
Profesor Titular  
Universidad Tecnológica de Pereira  
lhgonza@utp.edu.co

## MAXIMILIANO BUENO L.

Ingeniero Electricista  
Universidad Tecnológica de Pereira  
maxibu@gmail.com

El principal reto de la robótica móvil es la construcción de prototipos versátiles que de manera confiable realicen complejas tareas a pesar de la incertidumbre del ambiente.

La figura 1 muestra un escenario típico en el cual se desplaza la plataforma móvil.

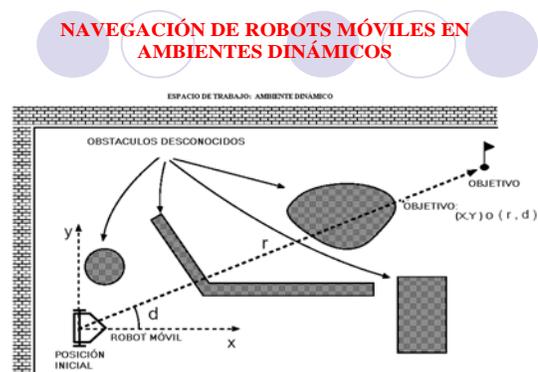


Figura 1. Escenario Típico donde se desplaza una plataforma móvil.

Actualmente las investigaciones en robótica móvil tienden al desarrollo de numerosas y nuevas arquitecturas que integran Percepción y Acción. La

tendencia general se orienta al planeamiento de pequeñas suposiciones sobre el ambiente encontrado, en la fase de ejecución y que la ejecución sea sensible al ambiente y se adapte a las contingencias encontradas.

Para lograr esto, los datos de percepción han sido incluidos en la capa de ejecución (Arquitecturas Híbridas).

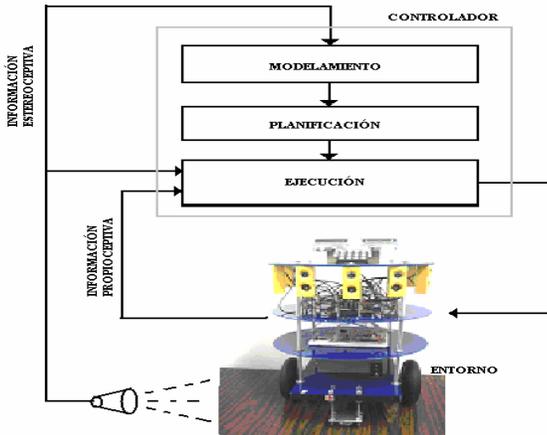


Figura 2. Arquitectura Híbrida. La capa de bajo nivel utiliza la percepción para adaptar dinámicamente el plan de ejecución a las contingencias del ambiente. El módulo de ejecución puede simultáneamente considerar solicitudes del plan de viaje como del ambiente.

Muchos investigadores han optado por implementar estrategia menos complejas, como “Divide y Conquistarás”, descomponiendo la capa de ejecución en pequeñas unidades de Decisión-Acción o comportamientos (Arquitectura Híbrida Basada en Comportamientos) [2],[3].

Un comportamiento es una pequeña unidad de control hecha para alcanzar un objetivo simple en un restringido conjunto de situaciones. Los robots autónomos necesitan para poder realizar tareas complejas, de la activación y cooperación de un número de comportamientos. Cada comportamiento implementa en detalle un control seguro para una subtarea específica como seguir una trayectoria, evitar objetos sensados o cruzar puertas.

En los actuales momentos en el diseño de controladores para robots móviles la lógica difusa juega un papel muy importante para el control de robots móviles basado en comportamientos. La lógica difusa permite el diseño de controladores que son robustos a la incertidumbre, los ruidos o las perturbaciones en las medidas de los sensores. Además, los comportamientos difusos pueden ser sintetizados en un conjunto de reglas de tipo if-then, de forma que el conocimiento experto queda reflejado en términos lingüísticos fáciles de entender.

Algunas arquitecturas de control de robots móviles [5] y [6], se basan en la descomposición de las tareas

complejas en comportamientos. Estos comportamientos están codificados usando una base de reglas difusas del tipo *if Antecedente then Consecuente*. Tanto el antecedente como el consecuente son conjuntos difusos. El valor de verdad del antecedente determina la "deseabilidad" de aplicar esa regla en el estado actual.

En [4] y [5], han definido tres comportamientos básicos (aproximación a la pared más cercana, seguir la pared, evitar obstáculos). Cada comportamiento se ha implementado como una base de reglas difusas.

La salida de cada comportamiento es un conjunto difuso.

Los comportamientos se encargan del control del movimiento del robot. Para combinar estos tres comportamientos se han definido unas metareglas difusas que aseguran una transición suave entre los estados.

## 2. CONCEPTOS BÁSICOS DE LOGICA DIFUSA.

La lógica difusa es básicamente una lógica multivaluada que proporciona un medio para enfrentar situaciones del mundo real, situaciones complejas y dinámicas, que son más fácilmente caracterizadas por palabras que por modelos matemáticos.

Los sistemas difusos son esquemas basados en el conocimiento, el cual a su vez está sintetizado en una base de reglas. El corazón del sistema difuso es la base de conocimiento que consiste en reglas del tipo IF Condición 1-THEN Acción 1. La condición 1 es denominada antecedente y la condición 2 es denominada consecuente.

El esquema básico de un sistema difuso se presenta en la figura 3.

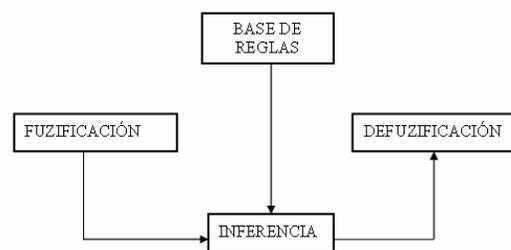


Figura 3. Esquema Básico sistema difuso

Los datos que entran al sistema difuso están expresados en lenguaje común, debido a esto es necesario trasladar el conocimiento inicial, es decir los datos a un lenguaje que sea entendido por el sistema difuso (fuzificación). Al terminar el proceso los datos son llevados nuevamente a su forma original a través de la defuzificación.

Las reglas creadas por el usuario describen el comportamiento del sistema, estas pueden ser del tipo Mandani o Takagi-Sugeno.

La lógica difusa o borrosa (Fuzzy logic) se fundamenta en la idea, que en un instante dado, no es posible precisar el valor de una variable X, sino tan solo conocer el grado de pertenencia a cada uno de los conjuntos en que se ha dividido el rango de variación de la variable.

El grado de pertenencia se cuantifica mediante la función de pertenencia f, que puede tener diferentes formas. En la figura 4 se muestran algunas de estas.

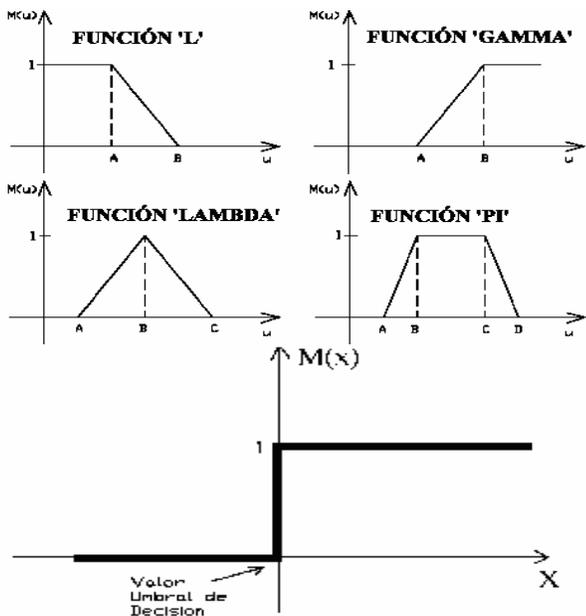


Figura 4. Funciones de pertenencia Típicas

La figura 5 muestra una función de pertenencia cualquiera asociada a una variable (para este ejemplo la variable es altura), además se muestran los valores lingüísticos y el universo de discurso.

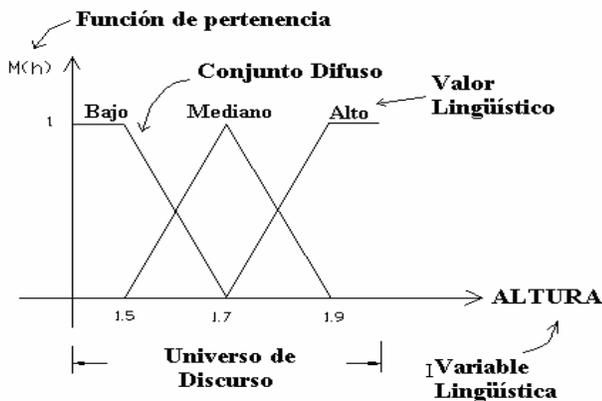


Figura 5. Algunos parámetros de importancia

El proceso de combinación de las diferentes reglas se realiza a partir del motor de inferencia.

Los dos tipos de motor de inferencia más utilizados en sistemas difusos y de control son:

**Motor de inferencia mínimo.** Dado el conjunto difuso A' en U, el motor de inferencia mínimo entrega B' como:

$$\mu_{B'}(y) = \max_{l=1} \left[ \sup \min(\mu_{A'}(x), \mu_{A'}(x_1), \dots, \mu_{A'}(x), \mu_{A'}(x)) \right] \tag{1}$$

**Motor de inferencia producto.** Dado el conjunto difuso A' en U, el motor de inferencia producto entrega B' en V de la forma:

$$\mu_{B'}(y) = \max_{l=1}^M \left[ \sup X \in U \left( \mu_{A'}(x) \prod_{i=1}^n \mu_{A'_i}(x_i) \mu_{B'}(y) \right) \right] \tag{2}$$

Siendo X y Y las variables lingüísticas de entrada y salida del sistema difuso y M el número de reglas en la base de reglas difusas.

Los procesos que operan en el mundo real usualmente requieren de acciones de control no difusas o sea señales que estén bien definidas dentro de los números reales.

Por ello se hace necesario recurrir a la *interfaz de defuzificación*.

La estrategia de defuzificación más utilizada es el **método del centro de gravedad**. El defuzificador por centro de gravedad especifica a y\* como el centro del área cubierta por la función de pertenencia de B', esto es:

$$y^* = \frac{\int y \mu_{B'}(y) dy}{\int \mu_{B'}(y) dy} \tag{3}$$

### 3. IMPLEMENTACIÓN DE LOS COMPORTAMIENTOS EN EL ENTORNO PROPUESTO.

Para la implementación de los comportamientos se ha propuesto un entorno conocido. El robot de pruebas tiene 8 sensores de ultrasonido ubicados en diferentes regiones de la plataforma, están ubicados de tal manera que entregan información de la distancia de los obstáculos, que lo rodean, en este caso los obstáculos son las paredes del entorno seleccionado.

Los sensores utilizados tienen un rango mínimo de 40 cm. y máximo de 300 cm.

Las dimensiones del espacio a utilizar son 500 cm. de largo y 200 cm. de ancho, la plataforma debe desplazarse en línea recta desde un extremo al otro del lugar donde se encuentra, si en algún momento pierde la trayectoria, el rumbo debe ser corregido y regresar a la trayectoria óptima, es decir, en línea recta.

La figura 6 muestra un grafico del entorno utilizado.

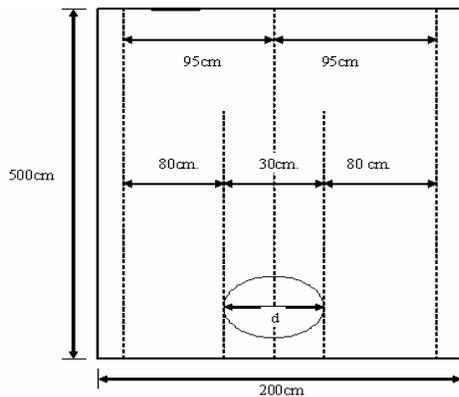


Figura 6. Entorno Utilizado

La disposición de los sensores se muestra en la figura 7.

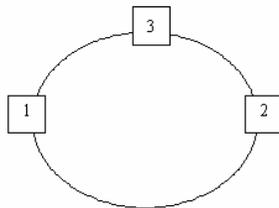


Figura 7. Disposición de los sensores

Los cuadros designados como 1, 2 y 3 indican los sensores agrupados, esto se hace con el fin de tener menor cantidad de datos. El grupo 1 y 2 contienen 2 sensores y el grupo 3 contiene 3 sensores.

Cada una de las mediciones de los sensores es promediada, de esta manera se obtiene el valor de la lectura del grupo. Por ejemplo, si los sensores que conforman el grupo 1, entregan dos lecturas L1 y L2, la lectura del grupo será:

$$lectura = \frac{L1 + L2}{2} \quad (4)$$

**4. UTILIZACIÓN DEL TOOLBOX DE LOGICA DIFUSA DE MATLAB**

Utilizando el toolbox de fuzzy disponible en Matlab 7.0 se implementan cada uno de los comportamientos propuestos.

**4.1 Comportamiento Seguir Pared**

El comando `readfis` entrega una descripción general del sistema.

```
a=readfis('pared.fis')
name: 'pared'
type: 'mamdani'
andMethod: 'min'
orMethod: 'max'
defuzzMethod: 'centroid'
impMethod: 'min'
aggMethod: 'max'
input: [1x3 struct]
output: [1x2 struct]
rule: [1x22 struct]
```

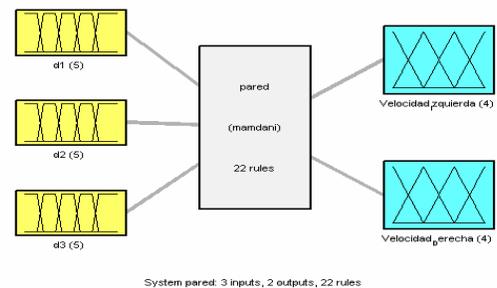


Figura 8. Entradas y salidas del sistema

Las funciones de pertenencia utilizadas son del tipo trapezoidal, esta selección se ha realizado debido a la facilidad de expresar estas funciones a través de un modelo matemático o conjunto de ecuaciones. Estas funciones de pertenencia representan de manera adecuada la naturaleza de los datos.

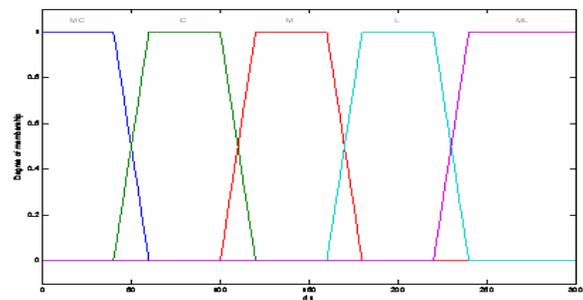


Figura 9. Función de pertenencia para las entradas del sistema

Las etiquetas lingüísticas empleadas para las funciones de pertenencia en las variables de entrada son las siguientes:

- MC: muy cerca
- C: cerca
- NR: normal

L: lejos  
ML: muy lejos

Las etiquetas lingüísticas empleadas para las funciones de pertenencia en las variables de salida son las siguientes:

MC: muy cerca  
C: cerca  
NR: normal  
L: lejos  
ML: muy lejos

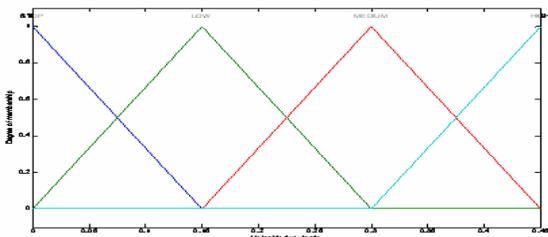


Figura 10. Función de pertenencia para las salidas del sistema

Se obtienen todas las reglas planteadas. Aquí se muestran algunas de ellas.

1. If (d1 is MC) and (d2 is ML) and (d3 is MC) then (Velocidad\_Izquierda is HIGH)(Velocidad\_Derecha is LOW) (1)
2. If (d1 is MC) and (d2 is ML) and (d3 is C) then (Velocidad\_Izquierda is MEDIUM)(Velocidad\_Derecha is MEDIUM) (1)
3. If (d1 is MC) and (d2 is ML) and (d3 is M) then (Velocidad\_Izquierda is HIGH)(Velocidad\_Derecha is HIGH) (1)
4. If (d1 is MC) and (d2 is ML) and (d3 is L) then (Velocidad\_Izquierda is HIGH)(Velocidad\_Derecha is HIGH) (1)

**4.2 Explicación de las reglas planteadas**

Para comprender el planteamiento de las reglas se presenta el siguiente ejemplo basados en una de las reglas planteadas.

1. If (d1 is MC) and (d2 is ML) and (d3 is MC) then (Velocidad Izquierda is HIGH) (Velocidad Derecha is LOW) (1)

Esta regla plantea un caso en el que la plataforma se encuentra cerca de una pared ubicada al frente, detectada por el grupo 3 de sensores, además el registro del grupo 1 indica que a la derecha se encuentra una pared cerca, en cambio a la izquierda según el grupo 2 se tiene el espacio necesario para realizar un giro, esto indica que la plataforma cambiara su rumbo hacia este lado.

El sistema de inferencia difuso empleando el toolbox de Matlab 7.0, se realiza de la siguiente forma:

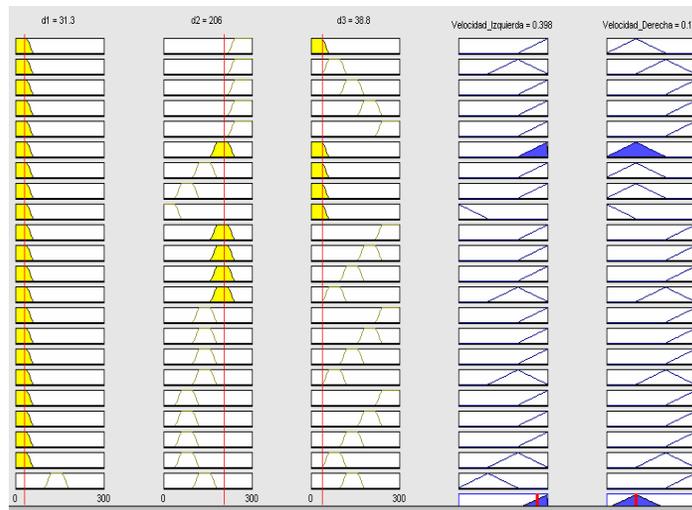


Figura 11. Sistema de Inferencia del Toolbox de Lógica Difusa.

**4.3 Pruebas realizadas: Comportamiento Seguir Pared.**

DATO	ENTRADAS			SALIDAS	
	d1(cm)	d2(cm)	d3(cm)	Vm1(m/s)	Vm2(m/s)
1	28,8	48,8	279	0,389	0,389
2	134	126	154	0,15	0,389
3	36,3	126	56,3	0,302	0,266
4	36,3	176	214	0,397	0,397
5	56,3	149	26,3	0,381	0,15
6	56,3	149	121	0,381	0,381
7	31,3	274	121	0,398	0,398
8	33,8	98,8	33,8	0,398	0,15
9	33,8	204	63,8	0,3	0,3
10	33,8	204	93,8	0,3	0,3
11	33,8	204	121	0,398	0,398
12	33,8	204	151	0,398	0,398
13	149	154	159	0,15	0,398
14	41,3	200	174	0,395	0,395
15	18,8	149	109	0,313	0,313
16	41,3	176	21,3	0,397	0,15
17	43,8	136	96,3	0,3	0,3
18	28,8	131	239	0,398	0,398
19	124	116	116	0,15	0,397
20	33,8	189	169	0,392	0,392

Tabla 1. Tabla con la información obtenida de los grupos de sensores para el comportamiento Seguir Pared.

**4.4 Explicación del Funcionamiento.**

Se asume que la plataforma móvil se encuentra en un punto inicial Xo en estado de reposo. La primera lectura evalúa las distancias de los tres grupos de sensores. A partir de la información obtenida se toma la decisión sobre el ángulo de giro.

La plataforma compara las distancias entregadas por los grupos 1 y 2 y gira hacia el lado que registra la menor lectura, si las lecturas son iguales gira hacia el lado izquierdo según el diseño elaborado.

Cada 30 cm se registran lecturas y se toma decisión sobre la variación en las velocidades de las ruedas. Cuando la

lectura registrada por el grupo 3 (parte delantera) disminuye, la velocidad también disminuye.

La regla inicial es planteada de la siguiente forma:

*If  $d1 < d2$  o  $d1 = d2$  THEN Velocidad izquierda es baja y velocidad derecha es alta (1)*

La regla (1) indica que hay un giro hacia la derecha.

Para permitir que la plataforma continúe siguiendo la pared, la regla es planteada de la siguiente forma:

*If  $d1 = MC$  y  $d2 = M$  y  $d3 = M$  THEN Velocidad izquierda es alta y velocidad derecha es alta (2)*

#### 4.5 Comportamiento Seguir Pasillo Explicación del Funcionamiento.

Se asume que la plataforma móvil se encuentra en un punto inicial  $X_0$  en estado de reposo. La primera lectura evalúa las distancias de los tres grupos de sensores. A partir de la información obtenida se toma la decisión que rumbo tomar.

En cada lectura de los sensores de ultrasonido (simulada), se comparan los registros del grupo 1 y 2 (grupos de los extremos), para este comportamiento estas lecturas deben ser iguales ya que la plataforma se esta desplazando en línea recta.

Cuando las lecturas son diferentes el rumbo se corrige variando las velocidades de las ruedas.

DATO	ENTRADAS			SALIDAS	
	d1(cm)	d2(cm)	d3(cm)	Vm1(cm/s)	Vm2(cm/s)
1	80,7	80,7	300	0,395	0,395
2	80,7	80,7	104	0,321	0,321
3	47,3	125	206	0,379	0,15
4	125	46	206	0,15	0,381
5	109	39,3	206	0,15	0,39
6	84,7	58	41,3	0,15	0,398
7	28,7	130	41,3	0,323	0,15
8	28,7	130	216	0,398	0,15
9	146	23,3	259	0,15	0,398
10	111	47,3	104	0,15	0,345
11	43,3	96,7	104	0,321	0,15
12	105	58	166	0,15	0,397
13	80,7	80,7	271	0,395	0,395
14	62,3	116	195	0,382	0,15
15	117	60,2	264	0,15	0,379
16	80,7	81,3	33,2	0,261	0,261
17	42,6	128	114	0,312	0,15
18	144	44,3	114	0,15	0,309
19	134	43	227	0,15	0,386
20	24,3	138	117	0,332	0,15

Tabla 2. Tabla con la información obtenida de los grupos de sensores para el comportamiento Seguir Pasillo.

## 5. CONCLUSIONES

Haciendo uso del toolbox de Lógica Difusa de Matlab se implementaron los comportamientos difusos SEGUIR PARED y SEGUIR PASILLO. Los resultados obtenidos fueron satisfactorios, dichos resultados se

implementarán en la plataforma móvil (P-METIN) del grupo GIROPS.

Como resultado de la utilización del Toolbox se concluye que este facilita una mejor implementación de las reglas ya que a partir de él se tiene una visión mas amplia de los universos de discurso, aunque presenta limitaciones en aplicaciones de comportamientos específicos, donde dicha implementación es mejor realizarla a través de programación en Matlab.

## 6. BIBLIOGRAFÍA

[1] Carrasco Rodrigo. Cipriano Aldo .Sistema de Guiado para un robot móvil basado en lógica difusa. XV Congreso de la Asociación Chilena de Control Automático. Octubre de 2002.

[2] Simarro Raúl. Ricolfe Carlos. Navarro José. Simó José. Herramienta para la prueba de controladores Inteligentes aplicados a la robótica móvil. Universidad Politécnica de Valencia. Departamento de Informática de Sistemas y Computadoras. 2002.

[3] Jasmin Velagic. Bakir Lacevic. Branislava Perunicic. A 3-level autonomous mobile robot navigation system designed by using reasoning/search approaches. Robotics and Autonomous Systems 54 Pages 989–1004. 2006

[4] Cordoba Juan. Ríos Luis. Restrepo Carlos. Descripción de una plataforma móvil implementada para el estudio de técnicas inteligentes de navegación: P-METIN. Scientia Et Technica. Año XI, No.27 Abril de 2005

[5] Driankov Dimiter. A Reminder on Fuzzy Logic. Springer- Physica Verlag. Pages 25-47.2001

[6] Saffiotti Alessandro. The Uses of Fuzzy Logic in Autonomous Robot Navigation. Sofá Computing 1 (4):180-197. 1997.

[7] Saffiotti Alessandro. Handling uncertainty in control of Autonomous robots. Artificial Intelligence Today. Springer-Verlag. Pages 381-408. 1999.

[8] Saffiotti Alessandro.Fuzzy Logic in Autonomous Navigation. Springer-Physica Verlag.Pages 3-24. 2001