



Ingeniare. Revista Chilena de Ingeniería

ISSN: 0718-3291

facing@uta.cl

Universidad de Tarapacá

Chile

Acosta, Gustavo; Gallardo, José; Pérez, Ricardo

Arquitectura de control reactiva para la navegación autónoma de robots móviles

Ingeniare. Revista Chilena de Ingeniería, vol. 24, núm. 1, enero, 2016, pp. 173-181

Universidad de Tarapacá

Arica, Chile

Disponible en: <http://www.redalyc.org/articulo.oa?id=77243535016>

- ▶ Cómo citar el artículo
- ▶ Número completo
- ▶ Más información del artículo
- ▶ Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Arquitectura de control reactiva para la navegación autónoma de robots móviles

Reactive control architecture for autonomous mobile robot navigation

Gustavo Acosta¹ José Gallardo² Ricardo Pérez²

Recibido 5 de noviembre de 2014, aceptado 9 de junio de 2015

Received: November 5, 2014 Accepted: June 9, 2015

RESUMEN

En este artículo se presenta una arquitectura de control basada en el paradigma reactivo para dar solución al problema de la navegación autónoma de robots móviles con capacidades de cómputo y sensórica limitadas. Se destacan como los aportes más relevantes en este trabajo, el encapsulamiento de una arquitectura de control basada en comportamientos en un sistema embebido programable con capacidades de procesamiento limitadas y el uso de la teoría de control fuzzy en la implementación del comportamiento básico de seguimiento de paredes. Las pruebas experimentales se realizaron sobre una plataforma robótica provista de tres sensores ultrasónicos y un módulo de comunicaciones inalámbricas. Este módulo soporta el protocolo 802.15.4 en la banda ISM (Industrial, Scientific and Medical), a través de él se transfieren medidas sensoriales a un PC en el que se elaboran representaciones computacionales del entorno explorado.

Palabras clave: Robot móvil, navegación robótica, arquitectura de control, control reactivo, control fuzzy.

ABSTRACT

In this paper, a control architecture based on the reactive paradigm to solve the problem of autonomous navigation of mobile robots with limited computational capabilities and sensor system is presented. Stand out as the most significant contributions to this work, the encapsulation of a behavior-based control architecture in a programmable embedded system with limited processing capabilities and the use of fuzzy control theory in the implementation of a wall following behavior. Experimental tests were performed on a robotic platform equipped with three ultrasonic sensors and a wireless communication module. This module supports the 802.15.4 protocol in the ISM (Industrial, Scientific and Medical) band, through it sensory measures are transferred to a PC where computational representations of the explored environment are made.

Keywords: Mobile robot, robotic navigation, control architecture, reactive control, fuzzy control.

INTRODUCCIÓN

Una de las tareas que debe llevar a cabo un robot móvil de manera autónoma es la exploración segura de su entorno de trabajo, la cual permite simultáneamente, recabar información sensorial útil

para la elaboración de modelos computacionales o mapas de los sitios visitados [1].

La navegación autónoma se torna un problema complejo por dos razones, la primera es la incertidumbre inherente a la sensórica del robot y

¹ Grupo de Investigación ICARO. Facultad de Ingeniería. Politécnico Colombiano Jaime Isaza Cadavid. Cra 48 N° 7-151. Medellín, Colombia. E-mail: gaacosta@elpoli.edu.co

² Departamento de Ingeniería de Sistemas y Computación (DISC). Universidad Católica del Norte. Av. Angamos 0610. Antofagasta, Chile. E-mail: jgallardo@ucn.cl; rperez@ucn.cl

la segunda es que muy frecuentemente los robots deben explorar entornos desconocidos para los cuales no se dispone, a priori, de mapas confiables y precisos con los cuales trazar trayectorias seguras de navegación [2].

Toda arquitectura de control de navegación incorpora tres funcionalidades básicas que son percepción, planificación y actuación [3]. La función de percepción recoge y procesa los datos suministrados por los sensores y entrega información útil para otras funciones del sistema. En la planificación se usa esta información para generar el conjunto de tareas que el robot debe ejecutar. Finalmente, la función de actuación produce las acciones de control que se deben aplicar a los efectores del robot. De la forma en que se relacionan estas primitivas surgen los tres paradigmas de control de la robótica: el clásico, el reactivo y el híbrido [4].

Para sistemas robóticos SWR (Small Wheeled Robots), usualmente equipados con módulos de percepción, actuación y procesamiento de bajo costo y prestaciones limitadas, el control reactivo basado en comportamientos les permite ejecutar tareas complejas, como por ejemplo la exploración de entornos desconocidos en condiciones de incertidumbre sensorial [5].

En [6-9] se reportan diferentes arquitecturas de control basadas en comportamientos y se emplea el control fuzzy para la síntesis de comportamientos básicos como evasión de obstáculos (*obstacle avoidance*), seguimiento de paredes (*wall following*), búsqueda de objetivos (*goal seeking*) y desarmado de puntos muertos (*deadlock disarming*).

Si bien una amplia variedad de sistemas híbridos inteligentes han sido propuestos para la optimización paramétrica de funciones de pertenencia y el aprendizaje de reglas [10-14], los controladores fuzzy diseñados y sintonizados mediante procedimientos manuales han permitido alcanzar el objetivo de proveer una navegación robusta para diferentes plataformas robóticas. Aunque tales procedimientos siempre darán lugar a implementaciones subóptimas, requiriendo además de procesos experimentales exhaustivos altamente demandantes de tiempo, han permitido sintetizar arquitecturas de control livianas y computacionalmente eficientes, muy conveniente para los sistemas SWR.

En este trabajo se propone una solución al problema de la navegación de pequeños robots móviles con ruedas en entornos interiores. Para tal fin se implementa una arquitectura de control basada en comportamientos que incorpora un comportamiento de evasión de obstáculos frontales y uno de seguimiento de paredes.

Este artículo se organiza de la siguiente manera, después de presentar una introducción al problema de la navegación robótica, el análisis teórico inicia exponiendo las características de la plataforma móvil usada en los experimentos realizados, continúa con la discusión de la arquitectura de control propuesta y presenta el proceso de diseño e implementación de comportamientos. Posteriormente se describen las pruebas de validación adelantadas y los resultados obtenidos, para finalizar con una exposición de las conclusiones surgidas del trabajo realizado y algunos lineamientos sobre trabajo futuro.

ANÁLISIS TEÓRICO

Plataforma Robótica

Por sus características, la plataforma robótica *Walling* clasifica como sistema SWR. El robot cuenta con dos bases en acrílico para la fijación de efectores, tarjeta de control y sensores.

La base circular inferior de 12,9 centímetros de diámetro se usa para fijar los efectores, dos motores DC de 6 V encapsulados en una caja motorreductora doble (Tamiya 70097) y una rueda de soporte *caster ball*. Ruedas de tracción se acoplan a los ejes de salida de la caja. El conjunto base circular, caja reductora, ruedas y *caster ball* configuran un sistema de locomoción diferencial con tercer punto de apoyo. Esta estructura provee locomoción, maniobrabilidad y estabilidad al robot.

Una segunda base cuadrangular de acrílico de 15x15 centímetros soporta los subsistemas de interfaz de efectores, procesamiento, sensórica y comunicaciones. La interfaz de efectores permite controlar la velocidad y sentido de giro de los motores incluidos en la caja motorreductora, está constituida por dos puentes H TA7291SG, cada uno de los cuales puede proporcionar una corriente promedio de 400 mA. El procesador genera las señales PWM (*Pulse-Width Modulation*) que se aplican a los puentes H.

Para el procesamiento de datos sensoriales y la implementación de la arquitectura de control se emplea la placa de desarrollo DEMOQE-128. A esta se incorpora un microcontrolador MCF51QE128 Coldfire de 32 bits que puede operar a una frecuencia de hasta 50 MHz.

El subsistema de percepción cuenta tres módulos ultrasónicos SRF02 para la medida de distancias a obstáculos cercanos localizados al frente y al costado derecho del robot. Asimismo, se acoplaron a las ruedas de tracción encoders ópticos de cuadratura con el fin de proveer información odométrica.

Las distancias a obstáculos, paredes y trayectos recorridos se transfieren del robot a un PC a través de una conexión inalámbrica. Un aplicativo desarrollado en Matlab captura esta información para la elaboración de mapas del entorno.

La comunicación entre PC y robot se da a través de un módulo de comunicaciones inalámbricas Xbee que implementa el protocolo 802.15.4 en la banda ISM (*Industrial, Scientific and Medical*) de 2,4 GHz.

El diseño y construcción del robot permite integrar modularmente los sistemas de actuación, control y percepción exteroceptiva, lo que facilita su expansión y reconfiguración. En la Figura 1 se muestra el robot *Walling*, plataforma en la que fue implementada la arquitectura de control propuesta. **Arquitectura de Control**

En robótica móvil, las arquitecturas de control llevan a cabo las funciones de percepción, representación, cognición y actuación, necesarias para proveer una navegación autónoma y segura. En tal sentido integran elementos de hardware y software necesarios para su implementación, organización e interconexión.

Inicialmente propuesta por Brooks [17], la arquitectura de subsunción surge a mediados de la década de los ochenta como el nuevo paradigma de la inteligencia artificial, en franca oposición hasta el entonces reinante modelo clásico o deliberativo.

Mientras que la aproximación tradicional o deliberativa se basa en un modelo centralizado del mundo sobre el cual se verifica la información sensorial y se generan las acciones que el robot debe ejecutar, el nuevo paradigma busca descomponer una tarea compleja de navegación en un conjunto de

comportamientos simples bien delimitados, razón por la cual también se le conoce como arquitectura de control basada en comportamientos [18].

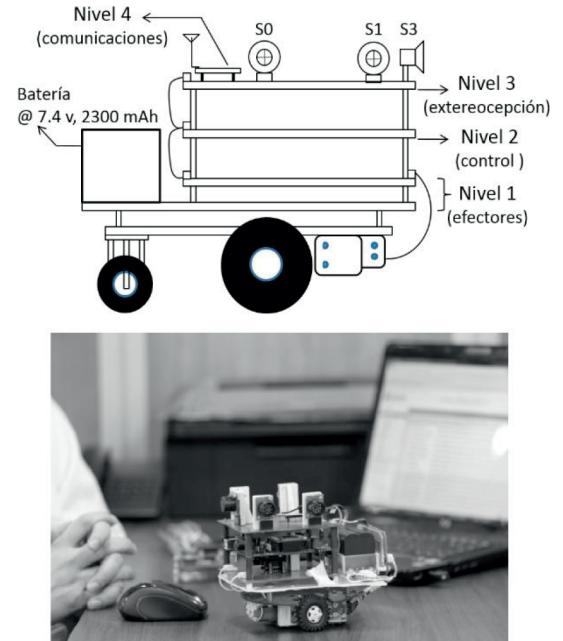


Figura 1. Plataforma de prueba (robot *Walling*).

La arquitectura de control reactiva basada en comportamientos ofrece una serie de ventajas como: 1) extensibilidad, en el sentido de que nuevos comportamientos pueden ser agregados como módulos a una arquitectura ya existente, 2) respuesta en tiempo real a obstáculos no modelados, en virtud de que la relación estímulo-acción es más directa que en el caso de la arquitectura deliberativa, que requiere del mantenimiento y actualización de un modelo del mundo y de un planificador y, 3) bajo costo computacional, precisamente por no requerir de modelos complejos del entorno para la planificación de tareas. En la Figura 2 se presentan los dos paradigmas de control.

Por otra parte, una de las principales críticas que se hace al control basado en comportamientos es el surgimiento, no predecible, de comportamientos complejos no programados que resultan de las interacciones de los comportamientos básicos a través del entorno.

Buscando contar con lo mejor de cada una de las arquitecturas mencionadas, algunos investigadores han adoptado esquemas de control híbrido para dar

solución al problema de la navegación autónoma [3, 12-14].

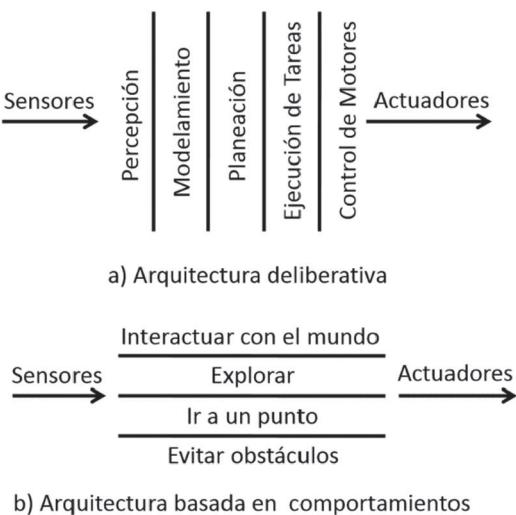


Figura 2. Paradigma deliberativo y reactivo.

Puesto que en este trabajo se considera el problema de la navegación en entornos interiores para sistemas SWR, caracterizados por sus limitaciones de procesamiento y sensórica, la arquitectura propuesta se acoge al paradigma reactivo a través del sistema de control basado en comportamientos que se presenta en la Figura 3.

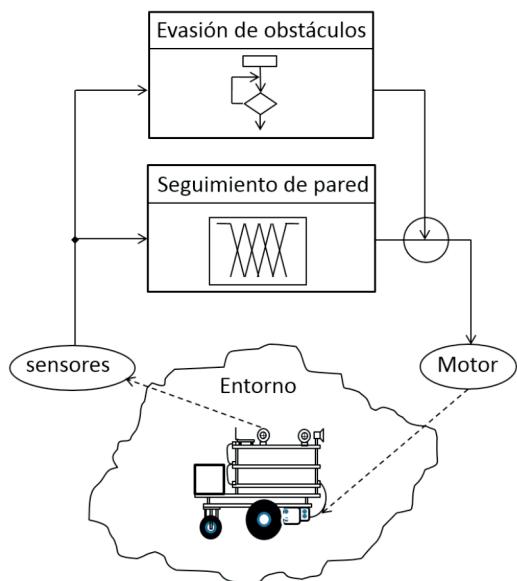


Figura 3. Arquitectura de control propuesta para sistemas SWR.

Diseño de Comportamientos

Se consideraron dos comportamientos básicos jerarquizados, evasión de obstáculos y seguimiento de pared derecha, el primero con mayor nivel de prioridad que el segundo. Ante la presencia de obstáculos frontales a distancias menores o iguales a 50 cm (distancia de seguridad), el comportamiento de evasión de obstáculos se activa, inhibiendo el seguimiento de paredes y asumiendo el control de los efectores del robot hasta alejarlo de la trayectoria de colisión.

Una vez dadas las condiciones de desactivación de la evasión de obstáculos, el comportamiento de seguimiento de pared se activa nuevamente y, mediante el control de los efectores, buscará alcanzar y mantener una distancia de referencia de 50 cm entre el robot y los objetos localizados a su costado derecho (paredes, columnas, muebles, etc.).

Solo un comportamiento podrá estar activo en un instante de tiempo dado y su activación o desactivación únicamente dependerá del estado del entorno captado mediante los sensores del robot.

La evasión de obstáculos corresponde a un comportamiento reactivo puro que se activa una vez el sensor frontal S3 (Figura 1) detecta un obstáculo a una distancia menor o igual a 50 cm. El módulo de control procesa la medida y genera una secuencia de acciones para evadir el obstáculo, evitando así una colisión. Estas acciones son: 1) inhibe el seguimiento de pared derecha y detiene ambos motores, 2) activa el motor derecho hasta contar ocho pulsos del encoder acoplado a la rueda derecha, de esta manera se consigue cambiar la orientación del robot en aproximadamente 90 grados en sentido anti-horario y, 3) si no se detectan nuevos obstáculos a distancias menores o iguales a 50 cm se reactiva el seguimiento de pared, en caso contrario se repite esta misma secuencia.

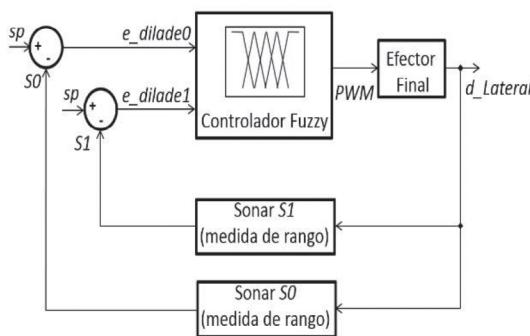
El seguimiento de pared derecha se implementó con base en un controlador fuzzy proporcional redundante. Se identificaron como variables lingüísticas de entrada los errores de distancia de los sonares S0 y S1 con respecto a una de referencia de 50 cm. Los valores crisp de estas variables se calculan mediante las ecuaciones (1) y (2).

$$e_dilade0 = sp - S0 \quad (1)$$

$$e_dilade1 = sp - S1 \quad (2)$$

Donde $e_dilade0$, $e_dilade1$ son los errores de distancia lateral derecha relativos a las medidas de los sonares $S0$ y $S1$ respectivamente (Figura 1) y sp es la distancia de referencia o set-point (50 cm).

El diagrama de bloques del controlador fuzzy usado en la implementación del seguimiento de pared derecha se presenta en la Figura 4.



sp : set-point o consigna (50 cm)

$e_dilade0$, $e_dilade1$: errores de distancia relativos a $S0$ y $S1$

PWM: señal de control aplicada al efecto derecho

$d_Lateral$: distancia del robot a la pared

Figura 4. Controlador fuzzy para la síntesis del comportamiento de seguimiento de pared.

Se consideró como variable lingüística de salida el ciclo de trabajo (*Duty Cycle*) de la señal PWM aplicada al motor derecho. El nivel DC aplicado al motor izquierdo no fue variado. Siendo así, un incremento del nivel DC aplicado al motor derecho provocará un aumento de su velocidad con respecto al motor izquierdo, lo que alejará al robot de la pared localizada a su costado derecho. Asimismo, el decremento del nivel DC de este motor provocará que el robot se acerque a la pared, puesto que su velocidad relativa al motor izquierdo será menor.

La caracterización completa de las variables lingüísticas consideradas en la implementación del controlador fuzzy se presenta en la Tabla 1. El arreglo FAM (Fuzzy Associative Matrix) de la figura 5 expresa el conjunto de reglas que conforman la base de conocimiento del controlador. Para la entrada encerrada en el óvalo se tiene la regla:

IF $e_dilade0$ IS E-NME0 AND $e_dilade1$ IS E-PPEQ1 THEN PWM IS PWM-ag

Tabla 1. Caracterización de variables lingüísticas del controlador fuzzy.

Variables Lingüísticas de Entrada	
<u>Variable:</u> $e_dilade0$ (error de distancia)	<u>Variable:</u> $e_dilade1$ (error de distancia)
<u>Conjunto de términos:</u>	<u>Conjunto de términos:</u>
E-ZERO: Error igual a cero ($S0$), distancia cercana al set-point	E-ZER1: Error igual a cero ($S1$), distancia cercana al set-point
E-PPEQ0: Error positivo pequeño ($S0$)	E-PPEQ1: Error positivo pequeño ($S1$)
E-PME0: Error positivo mediano ($S0$)	E-PME1: Error positivo mediano ($S1$)
E-PGR0: Error positivo grande ($S0$), robot muy cerca de la pared	E-PGR1: Error positivo grande ($S1$), robot muy cerca de la pared
E-NPEQ0: Error negativo pequeño ($S0$)	E-NPEQ1: Error negativo pequeño ($S1$)
E-NME0: Error negativo mediano ($S0$)	E-NME1: Error negativo mediano ($S1$)
E-NGR0: Error negativo grande ($S0$), robot muy alejado de la pared	E-NGR1: Error negativo grande ($S1$), robot muy alejado de la pared
<u>Universo del discurso:</u>	<u>Universo del discurso:</u>
$U=[-6,6]$, en cm	$V=[-6,6]$, en cm
<u>Reglas sintácticas:</u> se utiliza la expresión “Error” y los modificadores “igual a cero”, “pequeño”, “mediano” y “grande”	<u>Reglas sintácticas:</u> se utiliza la expresión “Error” y los modificadores “igual a cero”, “pequeño”, “mediano” y “grande”
<u>Reglas semánticas:</u> de acuerdo con la figura 6	<u>Reglas semánticas:</u> de acuerdo con la figura 6
Variable Lingüísticas de Salida	
<u>Variable:</u> PWM (ciclo de trabajo señal PWM motor derecho)	
<u>Conjunto de términos:</u>	
PWM-z: PWM cero, el robot no se aleja ni se acerca a la pared	
PWM-fp: PWM disminuye poco, el robot se acerca lentamente a la pared	
PWM-fm: PWM disminuye medianamente, el robot se acerca a la pared	
PWM-fg: PWM disminuye grande, el robot se acerca rápidamente a la pared	
PWM-ap: PWM aumenta poco, el robot se aleja lentamente de la pared	
PWM-am: PWM aumenta medianamente, el robot se aleja de la pared	
PWM-ag: PWM aumenta grande, el robot se aleja rápidamente de la pared	
<u>Universo del discurso:</u> $Z=[20,80]$ en % del ciclo de trabajo de la señal PWM	
<u>Reglas sintácticas:</u> se utiliza la expresión “PWM” y los modificadores “cero”, “disminuye/aumenta poco”, “disminuye/aumenta medianamente”, “disminuye/aumenta grande”	
<u>Reglas sintácticas:</u> de acuerdo con la figura 6	

e_dilade0							
E_ZERO	E_PPEQ0	E_PEM0	E_PGRO	E_NPEQ0	E_NME0	E_NGR0	
PWM-z	PWM-fp	PWM-fm	PWM-fg	PWM-ap	PWM-am	PWM-ag	
PWM-ap	PWM-ap	PWM-fp	PWM-fp	PWM-ap	PWM-ag		
PWM-am	PWM-am	PWM-am	PWM-z	PWM-ag			
PWM-ag	PWM-ag	PWM-ag	PWM-ag				
PWM-fp	PWM-fp	PWM-fm		PWM-fp	PWM-fm	PWM-fm	
PWM-fm	PWM-fm			PWM-fm	PWM-fp	PWM-fm	
PWM-fg				PWM-fg	PWM-fm	PWM-fg	

Figura 5. Base de reglas del controlador fuzzy.

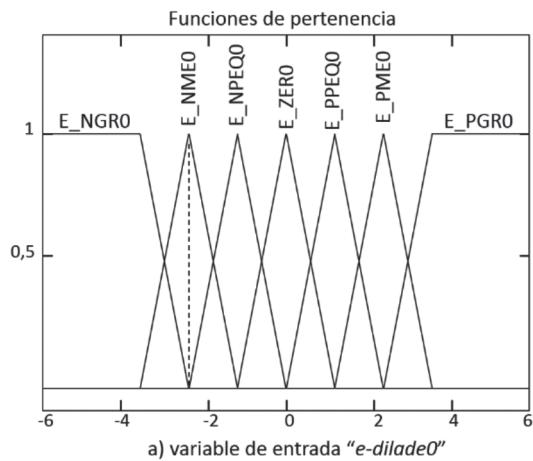
En la Figura 6 se muestran los conjuntos fuzzy asociados a las variables lingüísticas consideradas, $e_{dilade0}$, $e_{dilade1}$ y PWM , y se señalan valores específicos adoptados por las variables de entrada y el valor que asigna el controlador a la variable de salida. Esto en correspondencia con la regla tomada como ejemplo.

RESULTADOS

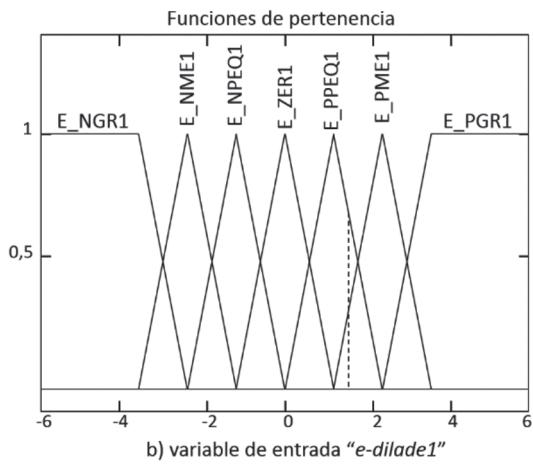
Las pruebas de validación de la arquitectura de control se llevaron a cabo en entornos interiores estructurados sin la presencia de objetos o personas en movimiento (entornos estáticos). Se consideraron tres tipos de escenarios:

- Entorno tipo I: oficina con dos paredes de ladrillo separadas por una puerta de madera.
- Entorno tipo II: pasillo con dos paredes de ladrillo no colineales separadas por una columna de concreto.
- Entorno tipo III: pasillo con dos paredes de ladrillo separadas por una puerta abierta que conduce a un baño.

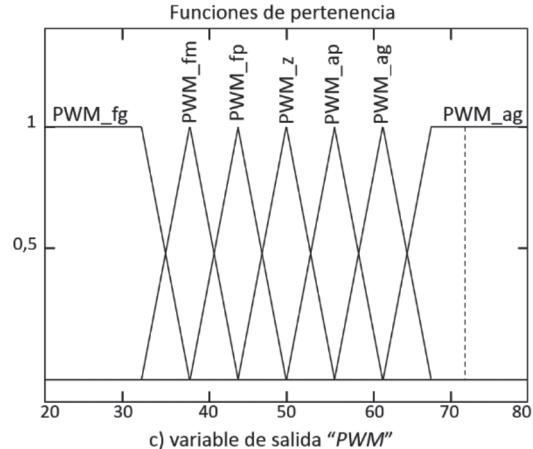
En la Figura 7 se aprecia el resultado de una de las pruebas de navegación llevadas a cabo en un entorno de oficinas, sobre un tramo de pared de 560 cm localizado a la derecha del robot y a una distancia inicial de 55 cm, cinco centímetros por encima de la referencia (entorno tipo I). Durante el seguimiento de pared el robot transmitió lecturas de distancia de los sonares S0 y S1 a una frecuencia de muestreo de 10 Hz (10 Sa/s).



a) variable de entrada "e_dilade0"



b) variable de entrada "e_dilade1"



c) variable de salida "PWM"

Figura 6. Base de reglas del controlador fuzzy.

Se transmitieron 500 muestras por recorrido con un tiempo total de navegación dado por la ecuación (3).

$$t = \frac{500 \text{ Sa}}{10 \text{ Sa/s}} = 50 \text{ s} \quad (3)$$

Las pruebas se llevaron a cabo a una velocidad lineal promedio de la plataforma de 11,2 cm/s. La distancia lineal recorrida por Walling durante las pruebas de navegación se calcula mediante la ecuación (4).

$$d_{nav} = v \times t = 50 \text{ s} \times 11,20 \text{ cm/s} = 560 \text{ cm} \quad (4)$$

Se aprecia como el controlador acerca el robot a la distancia de referencia de 50 cm y trata de mantener esta separación a lo largo del recorrido. El error promedio es de aproximadamente 1,7 cm y se debe principalmente a las características del sensor empleado. El sensor SRF02 solo entrega distancias enteras, si la distancia real es de por ejemplo 50,5 cm, entregará una lectura de 50 ó 51 cm, además su error máximo es de 1 cm para distancias cercanas.

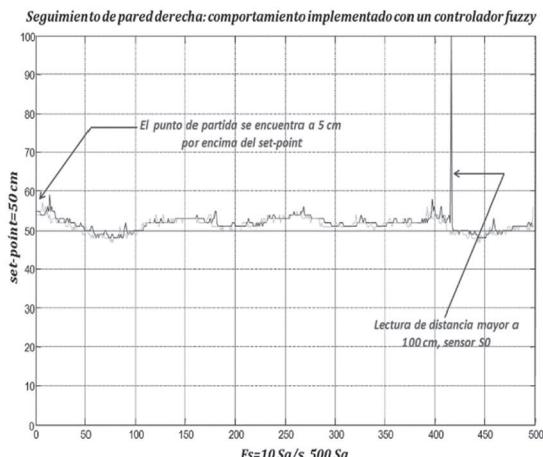


Figura 7. Prueba de seguimiento de pared derecha. Distancia de los sensores S0 y S1 a la pared derecha.

La Figura 7 muestra también una falsa lectura en el sensor S0, producto del fenómeno de reflexión especular asociado a los sensores ultrasónicos. En este sentido el esquema redundante propuesto, en el que se ingresan dos medidas de distancia al controlador fuzzy, permitió reducir los efectos negativos asociados a la incertidumbre en la medida.

También se incorporó un mecanismo de saturación por software. Cuando se reciben medidas menores

o iguales a 20 cm o mayores o iguales a 100 cm en cualquiera de los sensores, la variable que almacena dicha medida se coloca en 50 (centímetros).

En la Figura 8 se observa una prueba libre de reflexiones especulares, lo que significa que el robot no registró durante su recorrido cambios excesivos de orientación respecto a la pared, manteniéndose muy cerca de la distancia de referencia de 50 cm.

Los datos de odometría y distancia transmitidos por el robot durante la navegación, permitieron elaborar pequeños mapas de celdas de ocupación mediante un aplicativo desarrollado en Matlab.

En la Figura 9 se puede observar uno de los entornos de oficinas explorado y el mapa elaborado para una resolución de celda de 5 cm.

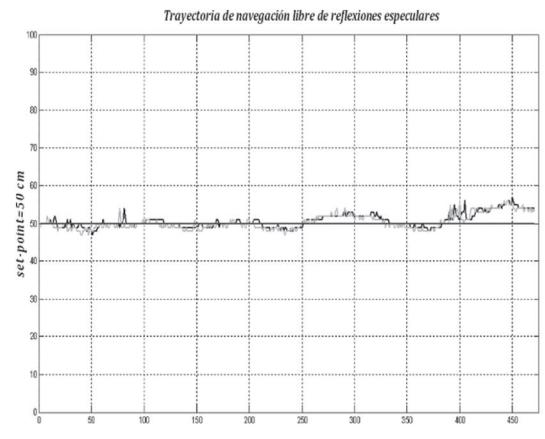


Figura 8. Recorrido libre de reflexiones especulares. Distancia de los sensores S0 y S1 a la pared derecha.

Se indican sobre el mapa algunas medidas que se usan para comparar las medidas reales con las calculadas sobre el mapa. Se consideraron, d1: distancia de la trayectoria seguida por el robot respecto al muro, d2: profundidad a la que se encuentra la puerta respecto al muro y d3: ancho de la puerta. Las medidas tomadas sobre el mapa se aproximan a las reales en un 93,5 %.

En este caso se modificó el controlador fuzzy para que realizara cambios automáticos de set-point ante cambios bruscos en las lecturas de distancia, como por ejemplo al detectar la profundidad de la puerta después de estar siguiendo el muro a 50 cm y, posteriormente, al detectar nuevamente el muro

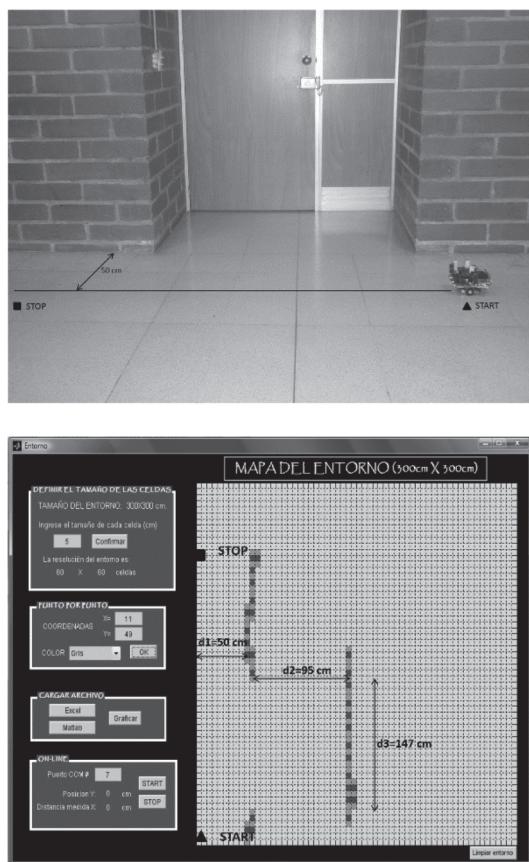


Figura 9. Mapas de celdas de ocupación de un entorno de oficinas.

(Figura 9). De esta forma se pudo mantener una trayectoria de navegación rectilínea.

CONCLUSIONES Y TRABAJO FUTURO

La incorporación de dos comportamientos básicos caracterizados por su bajo costo computacional, en una arquitectura basada en el paradigma reactivo, permitió la navegación segura de un pequeño robot móvil en entornos interiores de trabajo.

Al aplicar un esquema redundante de medida junto con un mecanismo de saturación, fue posible reducir los efectos negativos de la reflexión especular sobre el comportamiento de seguimiento de paredes.

La implementación del seguimiento de pared con base en un controlador fuzzy otorgó robustez adicional al sistema ante la incertidumbre en la medida. Esto, dada la capacidad de los sistemas

fuzzy de evitar transiciones bruscas en los valores de pertenencia de las variables lingüísticas ante los cambios en la medida, ya sea por ruido asociado al sensor o por ruido debido a las condiciones en que se llevó a cabo la medida. Asimismo, el esquema secuencial adoptado para la evasión de obstáculos permitió obtener una navegación libre de colisiones con obstáculos frontales.

Los datos recopilados durante la exploración del entorno permitieron construir representaciones computacionales simples con base en el concepto de celdas de ocupación. Los mapas elaborados se ajustaron en buena medida al entorno real, reflejando sus propiedades geométricas más relevantes.

La arquitectura propuesta y los algoritmos desarrollados pueden ser fácilmente adaptados a diferentes tipos de robots con limitaciones en computación y sensórica. Para plataformas más complejas que disponen de un mejor hardware, se recomienda optar por una arquitectura de control híbrida capaz de dar solución de manera simultánea al problema de la navegación local y global.

En este trabajo el ajuste paramétrico de las funciones de pertenencia se realizó de manera manual, lo que implica un proceso de experimentación exhaustivo que demanda mucho tiempo. En tal sentido es posible realizar, a futuro, la automatización de este proceso mediante métodos de búsqueda y optimización basados en algoritmos genéticos.

Es posible implementar algunas variantes del controlador fuzzy incorporando las acciones derivativa e integral y comparar su desempeño respecto a la solución propuesta en este artículo. También se puede hacer la implementación del comportamiento de evasión de obstáculos a través de lógica fuzzy e incorporar otros paradigmas de la inteligencia artificial como las redes neuronales con el fin de mejorar el grado adaptabilidad y cognición del sistema.

AGRADECIMIENTOS

Los autores agradecen el apoyo brindado por sus instituciones de origen, el Politécnico Colombiano Jaime Isaza Cadavid y la Universidad Católica del Norte, en el desarrollo de este trabajo.

REFERENCIAS

- [1] Y. Zhang, J. Liu, G. Hoffmann, M. Quilling, K. Payne, P. Bose and A. Zimdars. "Real-time indoor mapping for mobile robots with limited sensing". The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2010). San Francisco, USA. 2010.
- [2] M.S. Guzel and R. Bicker. "A behaviour-based architecture for mapless navigation using vision". International Journal of Advanced Robotic Systems. Vol. 9 Nº 2, pp. 1-13. 2012. ISSN: 1729-8806. DOI: 10.5772/46200.
- [3] C. Urdiales, E.J. Pérez, F. Sandoval, J. Vázquez-Salceda. "A hybrid architecture for autonomous navigation in dynamic environments". Proc. of the IEEE/WIC International Conference on Intelligent Agent Technology. Halifax, Canada. 2003.
- [4] J. Simpson, C.L. Jacobsen and M.C. Jadud. "Mobile robot control: the subsumption architecture and occam-pi". Editores: P. Welch, J. Kerridge and F. Barnes. Communicating Process Architectures 2006. IOS Press, pp. 225-236. Amsterdam, the Netherlands. 2006.
- [5] D. Nakhaeinia, S.H. Tang, S.B. Mohd Noor and O. Motlagh. "A review of control architectures for autonomous navigation of mobile robots". International Journal of the Physical Sciences. Vol. 6 Nº 2, pp. 169-174. 2011. ISSN: 1992-1950. DOI: 10.5897/IJPS10.540.
- [6] S. Tongchai, S. Suksakulchai, D.M. Wilkes and N. Sarkar. "Sonar behavior-based fuzzy control for mobile robot". Proc. the IEEE International Conference on Systems, Man and Cybernetics. Nashville, USA. 2000.
- [7] P. Vadakkepat, O.C. Miin, X. Peng and T.H. Lee. "Fuzzy behavior-based control of mobile robots". IEEE Transactions on Fuzzy Systems. Vol. 12 Nº 4, pp. 559-565. 2004. ISSN: 1063-6706. DOI: 10.1109/TFUZZ.2004.832536.
- [8] B.Q. Yong, L.S Ming, S.W. Yan and A.M. Jin. "A fuzzy behavior-based architecture for mobile robot navigation in unknown environments". Proc. of International Conference on Artificial Intelligence and Computational Intelligence (AICI'09). Shanghai, China. 2009.
- [9] N. Rahman, A.R. Jafri and M.U. Keerio. "Fuzzy behaviour based navigation of a mobile robot for tracking multiple targets in an unstructured environment". Proc. of the 6th WSEAS International Conference on Robotics, Control and Manufacturing Technology. Hangzhou, China. 2006.
- [10] M. Algabr, H. Mathkour and H. Ramdane. "Mobile robot navigation and obstacle-avoidance using ANFIS in unknown environment". International Journal of Computer Applications. Vol. 91 Nº 14. 2014. ISSN 0975-8887. DOI: 10.5120/15952-5400.
- [11] M.-R. Akbarzadeh-T, K. Kumbla, E. Tunstel and M. Jamshidi. "Soft computing for autonomous robotic systems". Computers and Electrical Engineering. Vol. 26 Nº 1, pp. 5-32. 2000. ISSN: 0045-7906. DOI: 10.1016/S0045-7906(99)0027-0.
- [12] F. Hoffmann. "Evolutionary algorithms for fuzzy control system design". Proc. of the IEEE. Vol. 89 Nº 9, pp. 1318–1333. 2001. ISSN: 0018-9219. DOI: 10.1109/5.949487.
- [13] M. Subramaniam, A. Manju and J. Nigamc. "A Study of parameter tuning of weighted fuzzy rule base using genetic algorithm for trajectory control of PUMA 560". International Journal of Engineering Science and Technology (IJEST). Vol. 3 Nº 9, pp. 7016-7028. 2011. ISSN: 0975-5462.
- [14] F. Halal and I. Dumitache. "Evolutionary learning of a fuzzy controller for a mobile robot". U.P.B. Sci. Bull. Series C. Vol. 69 Nº 3. 2007. ISSN: 1454-234x.
- [15] W. Gueaieb and S. Miah. "A Modular cost-effective mobile robot navigation system using RFID technology". Journal of Communications. Vol. 4 Nº 2. 2009. ISSN: 1796-2021. DOI: 10.4304/jcm.4.2.89-95.
- [16] S. Parasuraman, V. Ganapathy and B. Shirinzadeh. "Behavior based mobile robot navigation technique using AI system: experimental investigations". International Conference on Automation, Robotics and Autonomous Systems (ARAS 05). Cairo, Egypt. 2005.
- [17] R.A. Brooks. "A robust layered control system for a mobile robot". IEEE Journal of Robotics and Automation. Technical Report. Massachusetts Institute of Technology. Cambridge, USA. 1986.
- [18] M. Carreras. "A Proposal of a behavior-based control architecture with reinforcement learning for an autonomous underwater robot". Tesis para optar al grado de doctor. Universidad de Girona. Girona, España. 2003.