

## Sistema de frenos autónomo para un carrito móvil utilizando lógica difusa

<sup>1,2</sup>Melisa Kuzman; <sup>1,2</sup>Raúl Rivera; <sup>2</sup>Juan Manuel López; <sup>2</sup>Walter Gemín; <sup>2</sup>Juana Fernández; <sup>2</sup>Miguel Revuelta

<sup>1</sup>Universidad CAECE, Argentina.

<sup>2</sup>Universidad Nacional de Mar del Plata, Argentina.

melisakuzman@fi.mdp.edu.ar ; rrivera@fi.mdp.edu.ar

### Eje Temático del Trabajo: I WICEI

#### Resumen

En sistemas de control donde la complejidad del proceso es alta, no existen modelos matemáticos precisos que permitan resolver su transferencia. En estos casos los sistemas expertos permiten desarrollar sistemas de control basados en conocimientos, que con el agregado de hechos a su base de datos adquiere las reglas que le confiere habilidades y eficiencia en el control. Estas reglas representan el proceso de razonamiento de un experto, quien a partir de la información obtenida del sistema tome las acciones apropiadas. Contrario al control lógico, cuando la información del sistema no es exacta, la lógica difusa permite trabajar adecuadamente con este tipo de información basado en reglas determinadas por expertos.

En los conjuntos difusos la función de pertenencia puede tomar valores entre 0 y 1, y la transición del valor entre cero y uno es gradual, evitando un cambio brusco en la salida propia de los conjuntos clásicos. Para definir un conjunto difuso, se puede utilizar cualquier función de pertenencia, sin embargo son más frecuentes las del tipo triangular, trapezoidal, parabólica y gaussiana.

Este trabajo se enfoca en el uso de funciones triangulares y trapezoidales, ya que son las más simples de implementar y adaptables al conjunto de control propuesto, que consiste en el desarrollo de un sistema de frenos autónomo para un móvil experimental de tracción independiente en dos ruedas. Todo esto permite implementarlo en un microcontrolador de bajo costo, en un módulo Arduino, con resultados apropiados para todo el rango de funcionamiento, obteniendo un error menor al 4% respecto al esperado teóricamente.

**Palabras clave:** Mamdani, Microcontrolador, Control Difuso, Autónomo.

#### Introducción

Los sistemas de freno de emergencia autónoma son sistemas electrónicos que asisten al conductor para brindarle mayor seguridad. Si bien a futuro se piensa que los vehículos tendrán una conducción autónoma, este es uno de los primeros pasos para llevarlo a cabo.

Estos sistemas suelen ser eficaces y útiles para evitar accidentes. Si el conductor se encuentra distraído, o sucede algún acontecimiento imprevisto, es de gran utilidad contar con un dispositivo de frenos automáticos que pueda prevenir un incidente.

Se han desarrollado diversos sistemas que, dependiendo de su costo y complejidad, pueden funcionar solo en ciudades a bajas velocidades, mientras que otros, aunque no garantizan detener el auto antes de la colisión, sí son capaces de reducir drásticamente la velocidad en vehículos que viajen hasta a 210km/h. La detección de dichos eventos se

realiza con diferentes tipos de sensores y son llevados a cabo mediante distintos sistemas de control [1] [2].

En el presente trabajo se describe el desarrollo e implementación de un sistema de lógica difusa, aplicada en forma experimental para el control de velocidad de un carrito móvil con tracción en dos ruedas. La detección de objetos se realiza a través de un sensor ultrasonido HC-SR04 [3] y el algoritmo de control difuso en una placa "Arduino UNO" [4].

Diferentes técnicas de control se utilizan para regular el comportamiento de un sistema y se diferencian en dos grupos. Si la salida del sistema sólo depende únicamente de la entrada es un control a lazo abierto, y si se realimentan parte de las señales del circuito, es un control a lazo cerrado.

Los sistemas de control más utilizados son del tipo PID<sup>1</sup> y su diseño e implementación puede ser analógica o digital. Sin embargo, existen determinados casos en los que no es

<sup>1</sup> Proporcional Integral y Derivativo

posible o es muy complejo aplicar un control tradicional, y se recurre a técnicas más innovadoras como tablas de búsqueda o control difuso [5].

#### A. Control PID

Las técnicas de control clásico suelen ser una buena alternativa para controlar sistemas dinámicos. El control PID es uno de los métodos más utilizados en la industria. Sin embargo, a medida que la complejidad o la precisión de los ajustes del sistema aumentan, el diseño se torna más complejo y, por ende, más costoso.

Cuando la dinámica del proceso a controlar no es lineal, el controlador debe tener la capacidad para compensarla. En los casos que el sistema tenga soluciones lineales por tramos, se diseña el control PID para que ajuste los parámetros en función de la zona en la que se trabaja. En otros, de ser posible, se utiliza una función no lineal en lugar de una transferencia constante.

#### B. Lookup Table

Otro método utilizado frecuentemente son las tablas de búsquedas, más conocidas como Lookup Table.

La técnica se basa en sustituir una rutina de computación por la indización de una matriz. El uso de tablas de consulta suele mejorar notablemente el tiempo de respuesta del sistema. Esto se debe a que el tiempo de búsqueda de un valor en memoria siempre es menor que aquel requerido para realizar un cálculo. El ahorro de tiempo es fundamental cuando se necesita una respuesta inmediata.

Esta técnica es ampliamente utilizada para obtener el resultado de funciones. Así, cuanto más complejo sea el cálculo, mayor es el beneficio temporal obtenido. Sin embargo, cuando se quiere implementar un control en un sistema embebido, existe una limitación que suele ser fundamental: la memoria. Por ejemplo, para almacenar los diferentes valores que puede tomar la salida de una función, considerando como entrada una señal analógica, digitalizada con un conversor A/D<sup>2</sup> de 10 bits de resolución, se necesitan 2048 bytes de memoria, suponiendo que se trabaja con datos de 10 bits. De esta manera, la memoria necesaria es proporcional a la cantidad de entradas que se utilicen, la

resolución del conversor y el tipo de dato que se almacena.

#### C. Control Difuso

Considerada como una técnica para tomar decisiones, la lógica difusa se puede implementar en distintos sistemas de control.

Es una metodología donde el comportamiento del sistema se encuentra gobernado por decisiones claves, que se determinan con la experiencia o el conocimiento de los expertos que trabajan en el tema. Dichas decisiones se basan en reglas heurísticas de la forma IF (antecedente) THEN (consecuente), donde el antecedente y el consecuente son también conjuntos difusos, ya sea puros o resultado de operar con ellos [6] [7].

Una de las ventajas de este método, es que el lenguaje natural utilizado en la definición de las reglas suele simplificar la comprensión del problema. Sin embargo, cuando no se tiene suficiente entendimiento del sistema a controlar, optimizar el algoritmo resulta muy difícil.

Dentro de las características más relevantes del control difuso se puede mencionar la adaptabilidad, su tolerancia a la imprecisión, la capacidad para trabajar con problemas no lineales, y su fundamento en el lenguaje de sentido común. Además, dada la simplicidad del sistema se puede implementar rápidamente en un microcontrolador de bajo costo, como se demuestra en el presente trabajo.

### Descripción de la innovación

El sistema desarrollado posee características innovadoras tanto desde el punto de vista de la técnica elegida para el control del sistema, como también la plataforma en la que se implementa.

Por una parte, la técnica de control difuso permite trabajar con funciones de transferencias no lineales, sin necesidad de recurrir a la complejidad de un control PID. Esto posibilita que el sistema desarrollado tenga la función característica deseada siendo una solución de simple aplicación y bajo costo.

Además, en este trabajo se demuestra que es posible aplicar técnicas de lógica difusa, usualmente implementadas en sistemas de alta capacidad de procesamiento, en un microcontrolador con recursos limitados.

Finalmente, se debe mencionar que la innovación no solo es en la implementación del sistema, sino también en la manera que se

---

<sup>2</sup> Analógico Digital

formula el problema a controlar. Explicar a través del lenguaje natural hace que la resolución del problema sea más simple, tanto para su implementación como la comprensión de su funcionamiento.

### Proceso de implementación de la innovación

El diseño del sistema de control difuso propuesto tiene como objetivo regular la velocidad de un carrito móvil de acuerdo a la distancia a los diferentes objetos en su camino. Así, en el caso de interponerse algún elemento frente a su movimiento, el sistema ejecuta las acciones necesarias para detenerlo.

#### a) Hardware

Para medir la distancia del carrito a los objetos se utiliza un sensor ultrasonido HCSR04. El mismo posee características físicas que permiten medir la presencia de elementos desde 2 a 450cm.

Para generar el movimiento del carrito se utilizan dos motores de continua, llevando a cabo el control de su velocidad a través del uso de una señal PWM. Su ciclo de trabajo se regula a través de un parámetro de 8 bits, correspondiendo el 255 a un ciclo de trabajo del 100%, mientras que el 0 representa a 0%, es decir, ausencia de señal. Dicha señal de salida se calcula a través del sistema difuso diseñado.

Es necesario remarcar que la conexión entre la etapa de control y la de potencia requiere una interfaz. Para ello se utiliza un puente H, más precisamente el integrado L298 [8]. Por cada uno de los motores existen tres líneas de control conectadas al circuito, dos utilizadas para describir el sentido de giro, y la señal de PWM que indica la velocidad.

En la placa Arduino UNO se graba el firmware con el algoritmo necesario para implementar el sistema. Tanto la adquisición de datos, como la aplicación de los mismos en el sistema difuso y la ejecución de las acciones

correctivas son tareas del microcontrolador ATmega328P, de la compañía ATMEL [9]. Dentro de sus características más importantes se encuentran su memoria FLASH de 32KB, 2 KB de SRAM, 1KB de EEPROM, 16MHz de Clock de velocidad y 14 pines de entrada salida, de los cuales 6 pueden ser utilizados como salida PWM.

En la Figura 1 se observa la disposición de los componentes mencionados para conformar al carrito móvil.

#### b) Software

El controlador tipo Mamdani implementado se basa en la toma de decisiones utilizando lógica difusa. Un esquema representativo del mismo se observa en la Figura 2, y puede ser resumido en los siguientes pasos:

- La información extraída del sensor ultrasonido es acondicionada para que sea procesado en el sistema difuso.

- El fuzzificador toma los datos y los convierte en valores difusos para que los pueda procesar el mecanismo de inferencia. Los mismos corresponden a los niveles de pertenencia para cada entrada en los diferentes conjuntos difusos en los cuales se ha dividido el sistema.

- Se determinan las reglas en las que se basa el control difuso. Las mismas se encuentran conformadas a través de las proposiciones del tipo “If ... Then ...”, donde el antecedente se encuentra conformado por la variable lingüística distancia, correspondiente a la entrada, mientras que la salida o consecuente se representa con la variable velocidad. Así, a través del uso de las proposiciones se expresa la relación que une a ambos conjuntos difusos.

- La tarea del mecanismo de inferencia difusa consiste en tomar los valores de pertenencia, y apoyado en la base de reglas generar una salida difusa.

- Por último, debido a que el mecanismo de inferencia genera una salida difusa existe una etapa que toma ese valor y lo convierte en un número real. Dicha etapa se denomina defuzzificación, y se lleva a cabo mediante el método del centroide.

### Resultados

El sistema implementado en la placa “Arduino UNO” se simula en MatLab [10] utilizando “fuzzy logic toolbox” para comparar los resultados obtenidos.

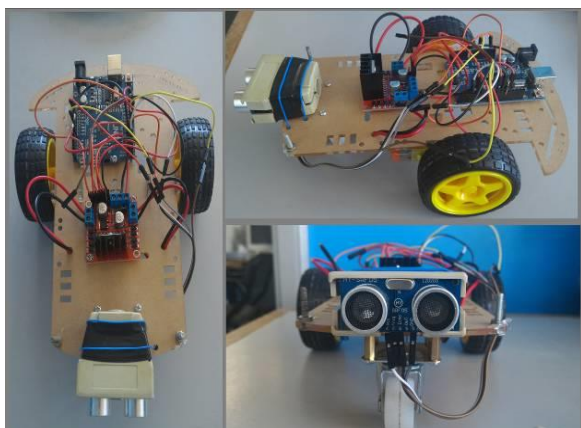
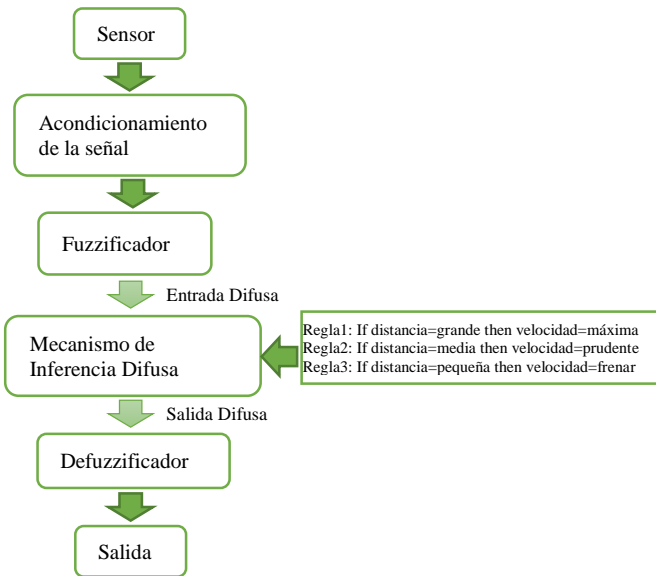


Figura 1: Perspectivas del carrito móvil



**Figura 2: Diagrama de flujo del sistema**

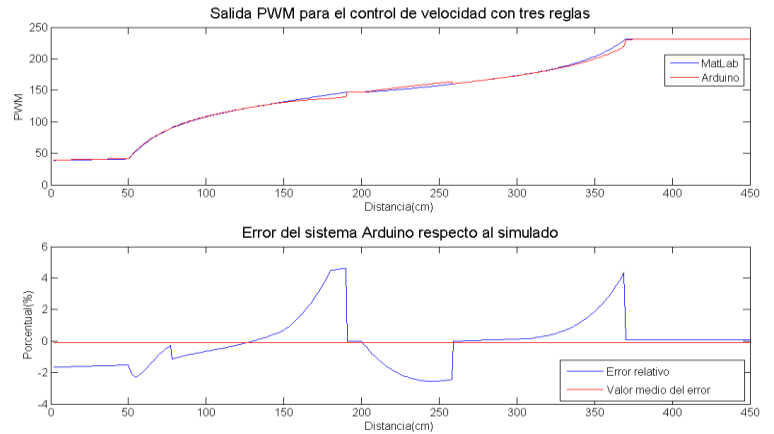
En las Figuras 3 y 4 se presentan las salidas de dos sistemas difusos con tres y cinco reglas, respectivamente. En cada gráfica se observa el resultado deseado y el obtenido en la práctica; mientras que en la parte inferior de cada una se representan los errores de los valores calculados en la práctica con respecto a aquellos obtenidos en la simulación.

Antes de comparar los resultados, vale aclarar que existe una consideración práctica para que las ruedas del móvil puedan vencer a la fuerza de rozamiento del piso, y es por ello que del rango de 0 a 255 valores se restringe de 10 a 255. Así, para distancias menores a 40 cm el sistema difuso determina que el PWM se carga con valores cercanos a 40, siendo este el mínimo capaz de vencer a la fricción de las ruedas utilizadas en el campo de acción de su movimiento.

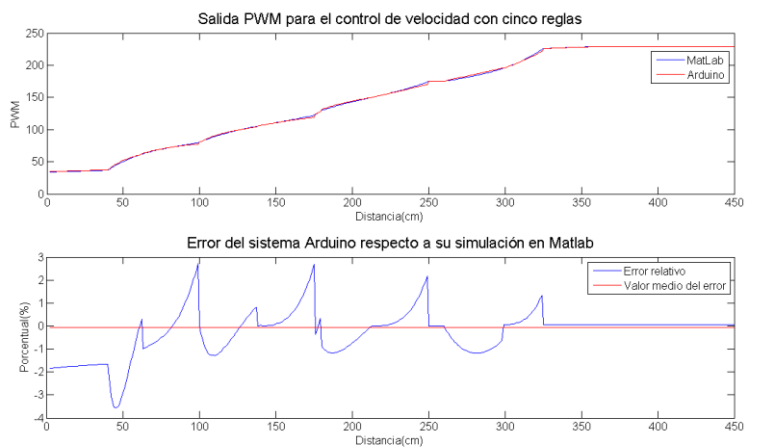
Como se puede verificar, el sistema real se aproxima al teórico para ambos casos. Observando las gráficas de salida se diferencian tres regiones:

- A. Los valores de PWM crecen linealmente con una pendiente muy baja.
- B. Para distancias mayores a 40cm la señal de salida comienza a crecer con mayor intensidad.
- C. La velocidad comienza a ser máxima y constante para las distancias superiores.

Analizando el error relativo para cada uno de los casos, se observa que el sistema con cinco reglas tiene un seguimiento levemente mejor a la señal deseada, con



**Figura 3: Sistema difuso con 3 reglas**



**Figura 4: Sistema difuso con 5 reglas**

errores picos que van desde -3.5% a 2.8%; mientras que para el sistema con 3 reglas van de -2.7% a 4.7%.

Sin embargo, la zona más crítica del sistema es la que corresponde a distancias menores a 40cm (región A). Allí el sistema debe actuar con mayor similitud al planteo teórico con el fin de evitar colisiones. Comparando ambos resultados en la zona de interés, el sistema difuso de tres reglas tiene como máximo un error aproximado al 2%, mientras que el de cinco reglas supera al 3.5%. En la Figura 5 se observan las funciones de pertenencia que dieron origen a la salida del sistema difuso con tres reglas.

Analizando los procesos descritos, la similitud de comportamiento que tienen ambos sistemas en general, y el mejor desempeño en la zona de interés, se determina que el sistema difuso de Mamdani con tres reglas de inferencia es más adecuado para la implementación.

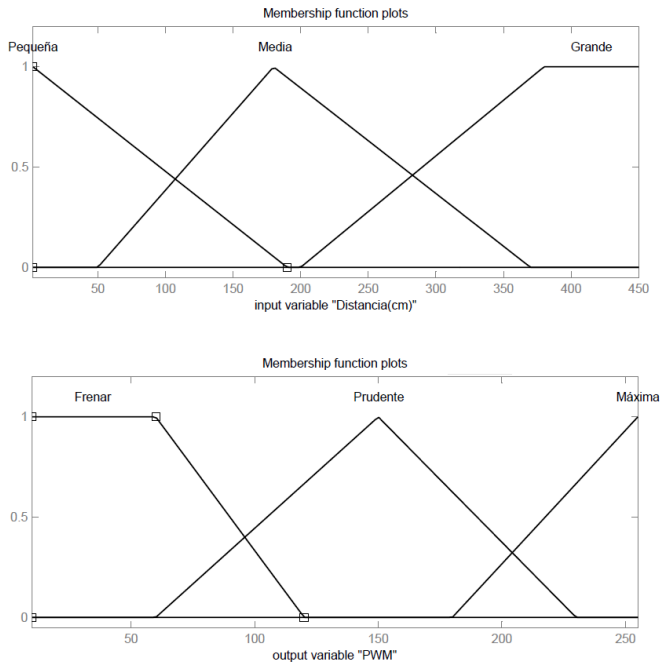


Figura 5: Sistema difuso Mamdani de 3 reglas utilizado

### Conclusiones o discusión abierta

Utilizando la lógica difusa se implementó un sistema de frenos autónomo, práctico, versátil y de bajo costo.

Para el diseño del sistema teórico se utilizó el *toolbox* de lógica difusa de Matlab. Una vez conforme con el rendimiento del control, esta técnica se implementó en una plataforma Arduino UNO.

Durante el desarrollo del proyecto se realizaron diferentes pruebas de control difuso del tipo Mamdani a un caso particular, variando la cantidad de reglas y las funciones de pertenencias implementadas. Así, se comprobó que no siempre se obtienen mejoras significativas cuando se hace más complejo el sistema.

Por otra parte, la simplicidad del algoritmo de control difuso que se utilizó, permitió que se pueda llevar a cabo en un microcontrolador de bajo costo, con poca memoria y velocidad estándar, tal como el utilizado en el proyecto.

### Referencias

[1] <http://www.euroncap.com/es/seguridad-en-los-vehiculos/descripción-de-los-premios/frenado-de-emergencia-autónomo/>

[2] <http://www.circulaseguro.com/los-sistemas-de-frenada-autonoma-reducen-un-38-los-alcances/>

[3] Ultrasonido HCSR04, <http://www.pridopia.co.uk/pi-user-manual/A4-ultra-sonic.pdf>.

[4] Arduino UNO, <https://www.arduino.cc/en/Main/ArduinoBo ardUno>.

[5] Ferreyra, Fuentes, Sacristán, "Control difuso una alternativa para aplicaciones de alta precisión", Universidad Autónoma Metropolitana.

[6] Kecman, "Learning And Soft Computing", Capítulo 6: 'Fuzzy logic System'.

[7] Inteligencia artificial, <http://ccia.ei.uvigo.es/docencia/IA>.

[8] Puente H L298, <http://www.alldatasheet.com/datasheet-pdf/pdf/22437/STMICROELECTRONICS/L298.html>.

[9] Atmel Corporation, <http://www.atmel.com/>.

[10] Matlab, <http://www.mathworks.com/>.