

Software para el diseño e implementación de controladores difusos en microcontroladores¹

RESUMEN

Se muestra el trabajo y los resultados obtenidos en el desarrollo de una herramienta para el análisis, diseño e implementación de sistemas de lógica difusa en sistemas microcontrolados, microprocesados u otro procesador con compilador compatible con ANSI C. El software permite el diseño de controladores o la utilización de sistemas de inferencia difusa en aplicaciones fuera del control, lo que facilita la implementación sin necesidad de tener conocimiento de programación de dispositivos.

PALABRAS CLAVE

Lógica difusa, microcontroladores, implementación, controlador, inferencia.

1. Introducción

En la década de los años noventa, el control difuso ganó gran aceptación en la industria y en la academia. El éxito mundial de innumerables productos comerciales y aplicaciones ha probado que la tecnología difusa no solo es práctica y poderosa sino también efectiva en términos económicos. Los sistemas del mundo real son no lineales; la modelación exacta es difícil, costosa e incluso imposible en muchos casos. El control difuso tiene la habilidad única para realizar exitosamente tareas de control sin conocer el modelo del sistema, aun si este es no lineal y complejo [1].

En la academia, la verificación de los controladores difusos se realiza por medio de software de simulación, que muestra resultados ideales y no permite ver de manera real el comportamiento de los diseños. En el estudio de control difuso, no hay versatilidad para emular diferentes modelos de sistemas ni tampoco para probar los distintos controladores.

Autores

Johan Calderón Acero²

Ivonne Vanesa Parra Garzón³

Director

Henry Montaña Quintero⁴

1. Proyecto curricular de Tecnología en Electrónica.

2. Tecnólogo en electrónica. Correo electrónico: nahhoj@hotmail.com

3. Tecnóloga en electrónica. Correo electrónico: vanjoh_18@hotmail.com

4. Ingeniero electrónico, docente de la Facultad Tecnológica, Universidad Distrital Francisco José de Caldas. Correo electrónico: hmontana@gmail.com

Considerando lo anterior, se tomó la decisión de desarrollar un software para la implementación de controladores difusos, con el fin de facilitar al educando los procesos de ejemplificación en la práctica de los conceptos teóricos; que agilice el aprendizaje y que permita la apropiación y transferencia de esta tecnología sin necesidad de tener conocimientos muy avanzados sobre el tema.

2. Diseño e implementación del software

La principal finalidad del software es facilitar el desarrollo de sistemas y controladores difusos y la generación de sus códigos para microcontroladores o sistemas microprocesados. Para la implementación de estos sistemas se requiere desarrollar interfaces mínimas como: configuración de la resolución a trabajar; manejo de conjuntos para cada universo, tanto de entradas como de salidas; interfaz de base de reglas; interfaz de inferencia y generación del código para microcontroladores, entre otras.

El sistema permite diseñar cada uno de los conjuntos difusos de acuerdo con la configuración de los universos de entrada. Puede operar con cada uno de ellos definiendo una base de reglas fundamentadas en el juicio del experto. También se muestra el proceso de inferencia difusa y, por último, su valor de defusificación, todo esto según la configuración de características del controlador que se habrá realizado previamente. Se trata de un sistema abierto y escalable, lo que permite la incorporación de nuevas entradas, conjuntos y reglas conforme a las limitaciones contempladas en el diseño.

Se trabajó con un lenguaje de programación orientado a objetos, VB.NET, debido a que el entorno es netamente gráfico. Además, las herramientas de desarrollo incluidas por Microsoft en su Visual Studio.Net son mucho más simples, intuitivas y sencillas de manejar que sus equivalentes en JAVA y en otros lenguajes de programación.

En la siguiente figura se muestra el esquema general y los diagramas de cada uno de sus componentes.

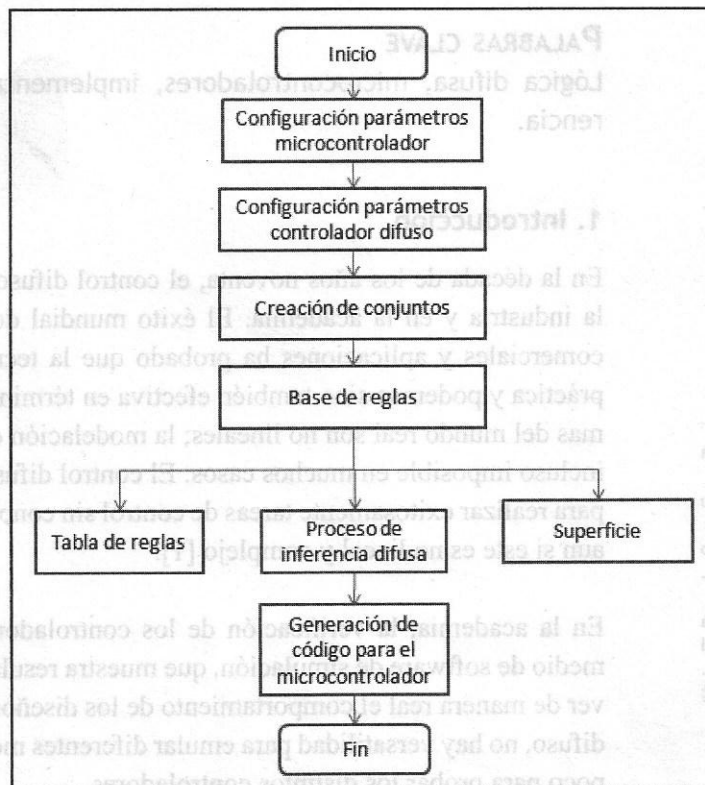


Figura 1. Diagrama general del software Fuzzy Udtecno

Las principales características del software Fuzzy Udtecto son las siguientes:

- Permite configurar los parámetros del controlador difuso.
- Permite el diseño gráfico de los universos de entrada y salida.
- Permite el diseño de bases de reglas completas o incompletas
- Provee herramientas de diseño rápido para los universos de entrada y salida, las variables lingüísticas y la base de reglas.
- Permite analizar el comportamiento global del sistema mediante tablas de entrada-salida y la visualización de la superficie difusa.
- Permite analizar el comportamiento del sistema en entradas particulares mediante la ventana de simulación.
- Provee el código fuente ANSI C, C18, C30 y .fis del sistema diseñado.

La configuración de parámetros del microcontrolador hace posible seleccionar el número de entradas y la resolución a trabajar, mientras que la configuración del controlador difuso permite seleccionar el tipo de im-

plicación (mínimo o producto), agregación (máximo o suma) y defusificación (centro de gravedad, promedio de máximos, primer o último máximo) para el controlador tipo Mamdani.

Para la creación de conjuntos, se definieron cuatro funciones de pertenencia: triangular, trapezoidal, sigmoidea y gaussiana. Insertando y seleccionando cada una de ellas, se puede modificar sus parámetros. Cuantos más conjuntos se elijan, se tiene mayor poder descriptivo y flexibilidad, aunque también un mayor costo de procesamiento en el microcontrolador. Por ello el software permite crear hasta cinco conjuntos difusos para cada entrada y salida. En la Figura 2 se muestra el diagrama de actividades para esta sección del programa.

Se espera que el controlador provea una salida para todos los estados posibles de la planta a controlar. Para ello, todas las variables de entrada y salida del controlador difuso deben poder vincularse a un conjunto difuso; a su vez, cada conjunto difuso debe participar del antecedente o consecuente de una regla. A continuación se detallará cómo se define la base de reglas (ver Figura 2).

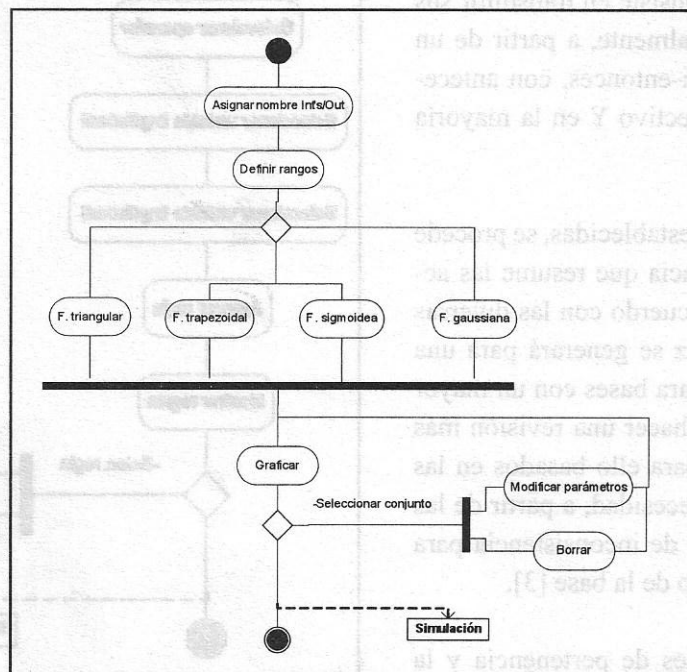


Figura 2a. Diagrama de actividades uml, creación y manipulación de conjuntos difusos

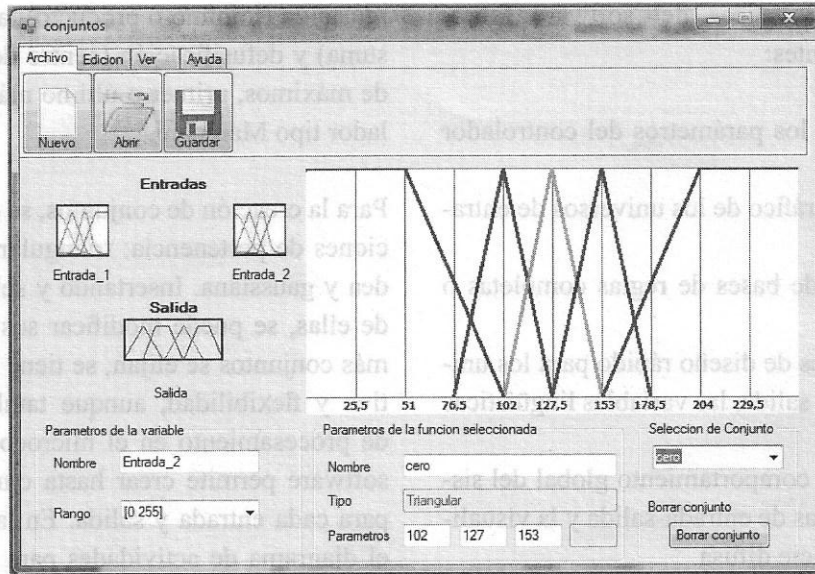


Figura 2b. Formulario de conjuntos Fuzzy Udtecno

Un problema genérico que puede ser resuelto en términos de razonamiento aproximado por un ser humano experto en la materia, generalmente siguiendo el esquema de modus ponens generalizado; el experto es capaz, mediante un conjunto de reglas que constituyen su experiencia sobre el tema, de elaborar unas conclusiones o consecuentes a partir de unos hechos observados o de unos antecedentes. Consiste en transmitir sus conocimientos, al menos parcialmente, a partir de un conjunto de n reglas del tipo si-entonces, con antecedentes relacionados por el conectivo Y en la mayoría de los casos [2].

En concordancia con las reglas establecidas, se procede a diseñar una matriz de inferencia que resume las acciones de control a tomar, de acuerdo con las diversas condiciones posibles. La matriz se generará para una o dos entradas. Sin embargo, para bases con un mayor número de reglas es necesario hacer una revisión más sistemática. Existen métodos para ello basados en las definiciones de posibilidad y necesidad, a partir de las que puede obtenerse un índice de inconsistencia para cada regla respecto del conjunto de la base [3].

Una vez definidas las funciones de pertenencia y la base de reglas, se puede observar en la ventana de si-

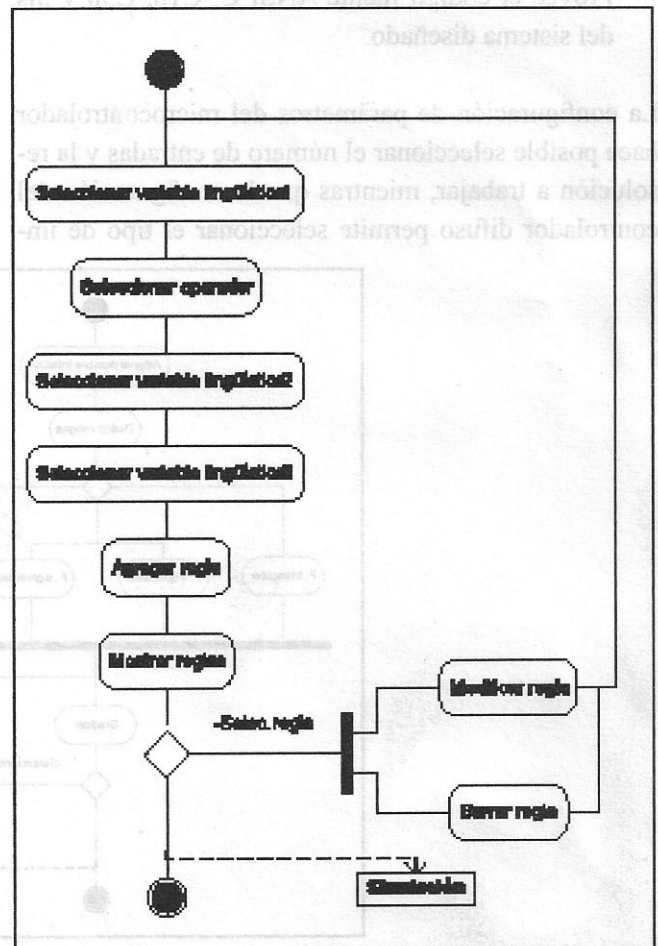


Figura 3a. Diagrama de actividades uml

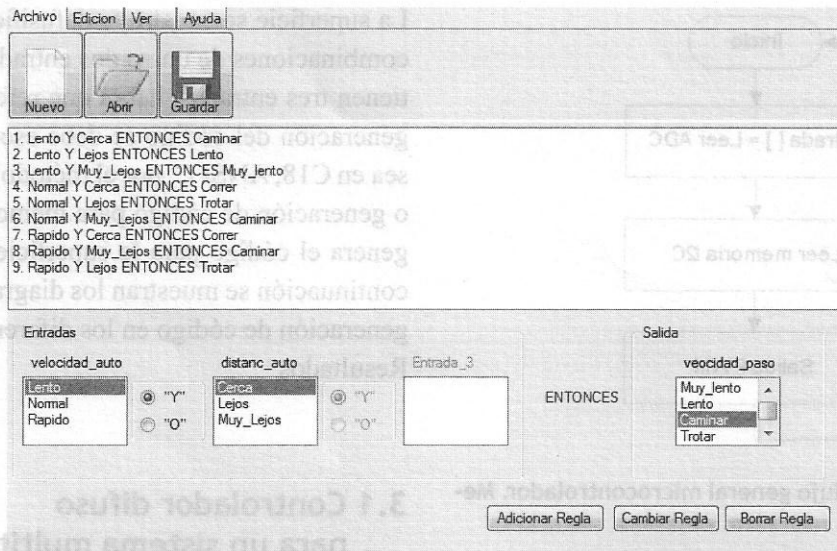


Figura 3b. Ventana de base de reglas Fuzzy Udtecto

mulación el proceso, donde el usuario ingresa valores para cada una de las entradas definidas. Se determina también en qué grado las variables de entrada pertenecen a los conjuntos difusos a través de su función de pertenencia y se valida qué reglas se activan de acuerdo

con los valores de pertenencia (ver Figura 4). Con las reglas activas se realiza la implicación entre antecedente y consecuente. Para que se cumpla la equivalencia entre lógica proposicional o clásica y lógica difusa [2], se debe traducir por una t-norma, cuyos representantes

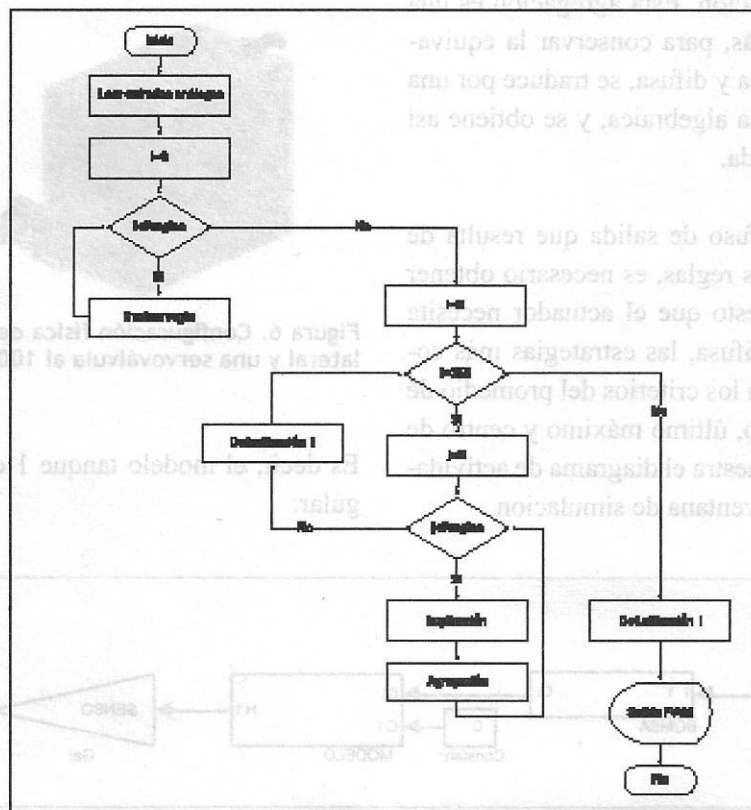


Figura 4. Diagrama de flujo general microcontrolador. Método vector

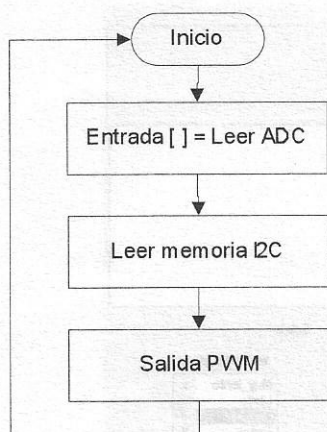


Figura 5. Diagrama de flujo general microcontrolador. Memorias I2C

más utilizados en el ámbito de la ingeniería son el mínimo y el producto.

Una vez evaluadas todas las reglas y obtenidos los conjuntos difusos de salida modificados, hay que realizar la agregación de todas las reglas para obtener un resultado único de su actuación. Esta agregación es una unión lógica; una vez más, para conservar la equivalencia entre lógicas clásica y difusa, se traduce por una t-norma, máximo o suma algebraica, y se obtiene así el conjunto difuso de salida.

A partir del conjunto difuso de salida que resulta de la agregación de todas las reglas, es necesario obtener un resultado escalar. Puesto que el actuador necesita tener una magnitud no difusa, las estrategias más comúnmente empleadas son los criterios del promedio de máximos, primer máximo, último máximo y centro de área. En la Figura 5 se muestra el diagrama de actividades correspondiente a la ventana de simulación.

La superficie se construye defusificando para todas las combinaciones de una o dos entradas del sistema. Si se tienen tres entradas, habrá que referenciar una. Para la generación del código se debe escoger el lenguaje, ya sea en C18, ANSI C, por el método vector, .fis (Matlab) o generación de código para memorias I2C (cuando se genera el código para la superficie) en CCS o C18. A continuación se muestran los diagramas de flujo para la generación de código en los diferentes lenguajes. 3. Resultados

3.1 Controlador difuso para un sistema multitanque a escala

Se implementó un controlador difuso para la planta Sistema Multitanque a Escala, diseñado e implementado por estudiantes de Ingeniería en Control de la Universidad Distrital Francisco José de Caldas. Se tomó la siguiente configuración [4]:

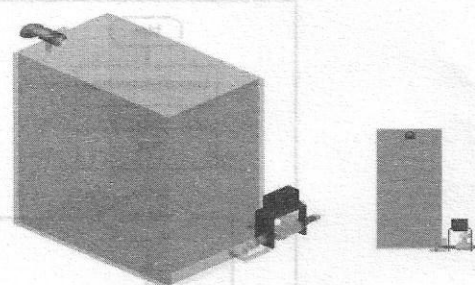


Figura 6. Configuración física de un tanque con desagüe lateral y una servoválvula al 100% de apertura

Es decir, el modelo tanque 1 con capacitancia rectangular:

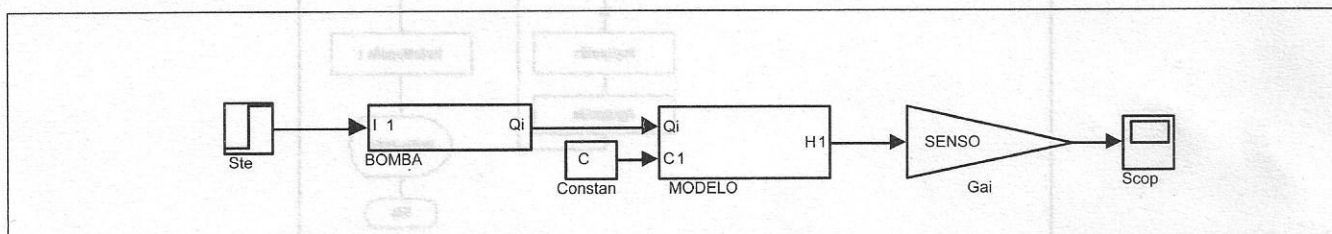


Figura 7. Controlador difuso y planta

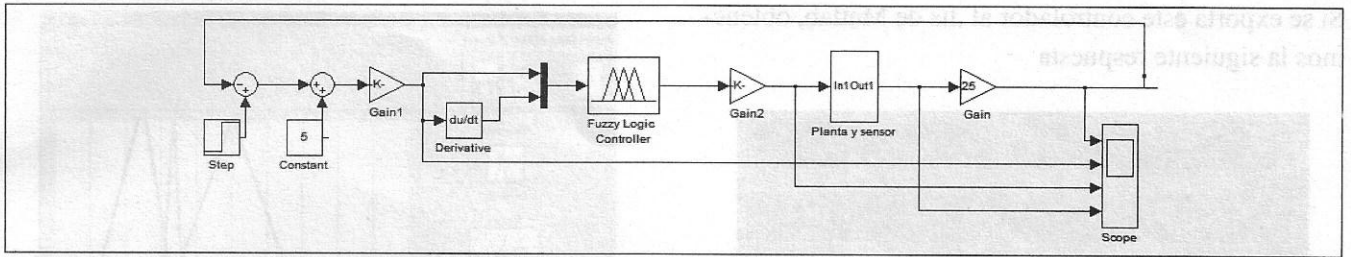


Figura 8. Señales de entrada y salida

Se adecuan las señales de entrada y salida del controlador difuso de la siguiente forma:

Se diseñó un controlador difuso tomando como universos de discurso el error y la derivada del error para las entradas y el actuador para la salida. Se tiene una implicación de mínimo, agregación de máximo y defusificación centro de gravedad. Se muestran las funciones de membresía creadas para cada entrada y salida y la base de reglas.

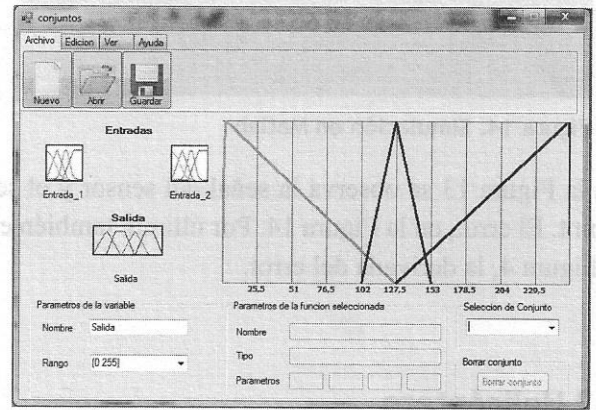


Figura 11. Universo de salida: actuador bomba Fuzzy Udtectno

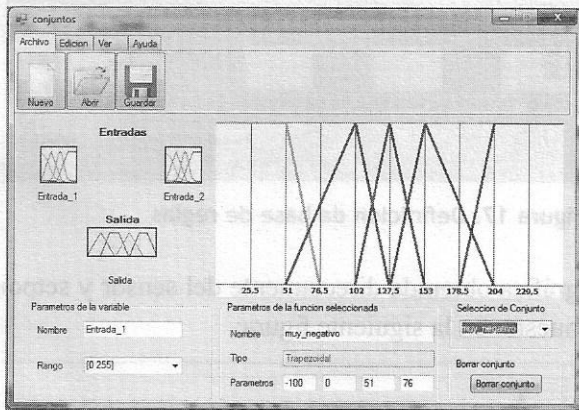


Figura 9. Universo de entrada: error Fuzzy Udtectno

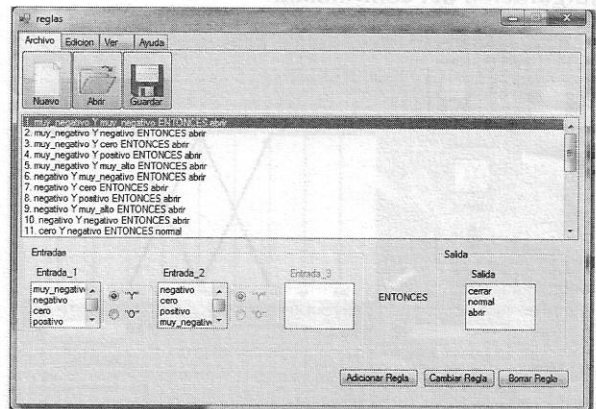


Figura 12. Base de reglas Fuzzy Udtectno

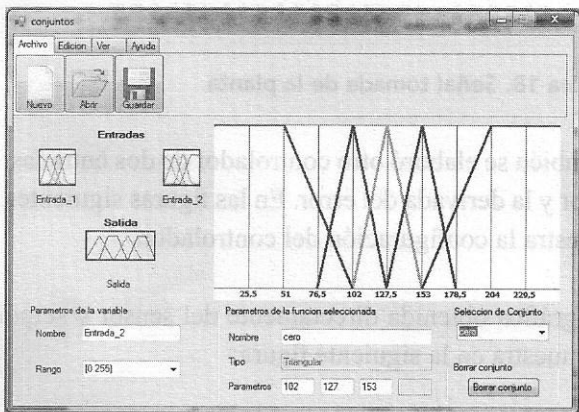


Figura 10. Universo de entrada: derivada del error Fuzzy Udtectno

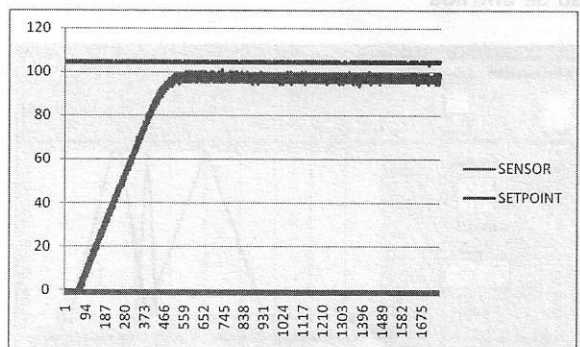


Figura 13. Valores tomados de una prueba real

Si se exporta este controlador al .fis de Matlab, obtenemos la siguiente respuesta

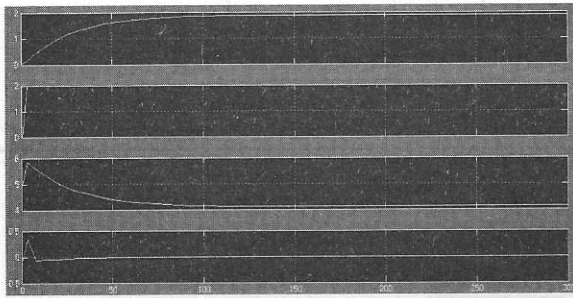


Figura 14. Simulación en Matlab

En la Figura 13 se observa la señal del sensor y el setpoint. El error, en la Figura 14. Por último, también en la Figura 4, la derivada del error.

3.2 Helicóptero

Se realizó un control para manejar el ángulo de elevación de un helicóptero. A continuación se mostrará la configuración del controlador.

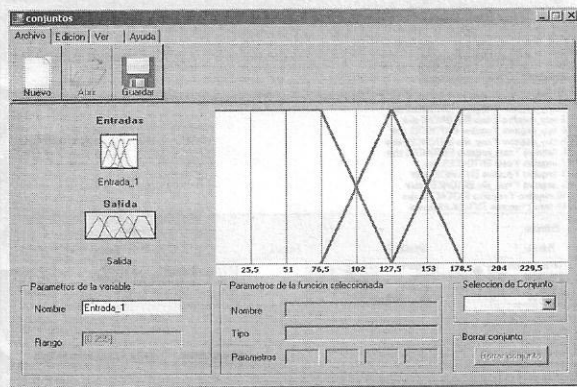


Figura 15. Funciones de membrecía para el universo de entrada

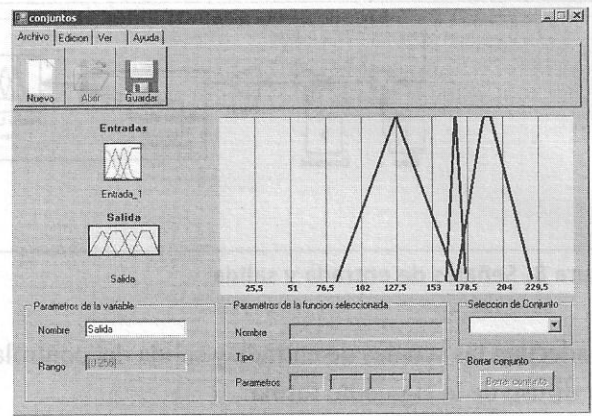
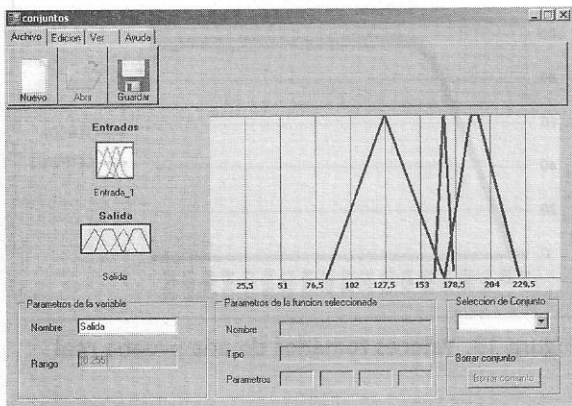


Figura 16. Funciones de membrecía para el universo de salida

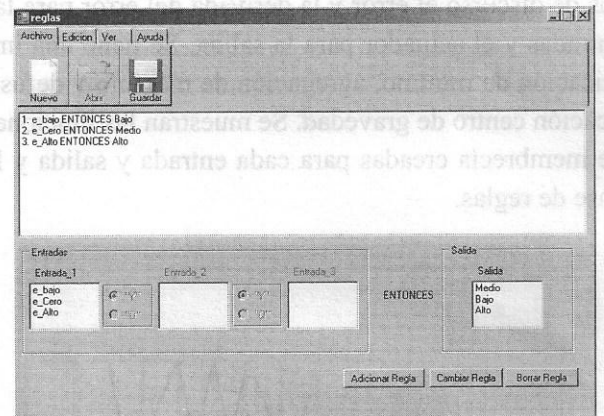


Figura 17. Definición de base de reglas

La gráfica obtenida directamente del sensor y setpoint se muestra en la siguiente figura:

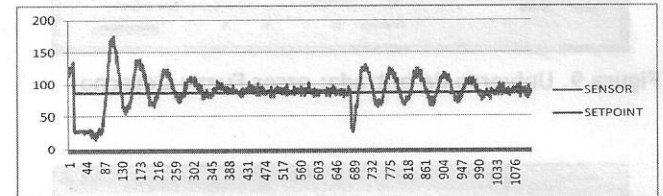


Figura 18. Señal tomada de la planta

También se elaboró otro controlador de dos entradas: el error y la derivada del error. En las figuras siguientes se muestra la configuración del controlador.

La gráfica obtenida directamente del sensor y setpoint se muestra en la siguiente figura:

Como se puede observar en las gráficas, el control de dos entradas tiene un mejor comportamiento, porque

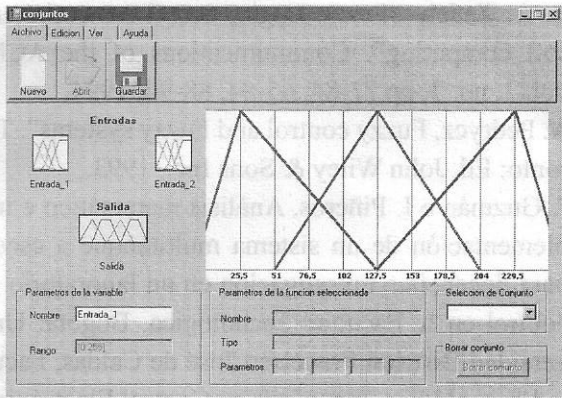


Figura 19. Conjuntos universo de entrada 1 (error)

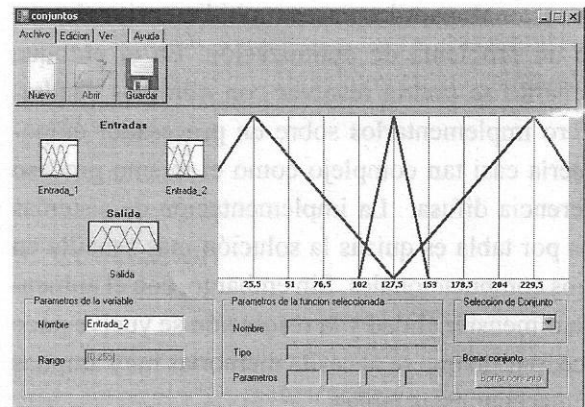


Figura 20. Conjuntos universo de entrada 2 (derivada del error)

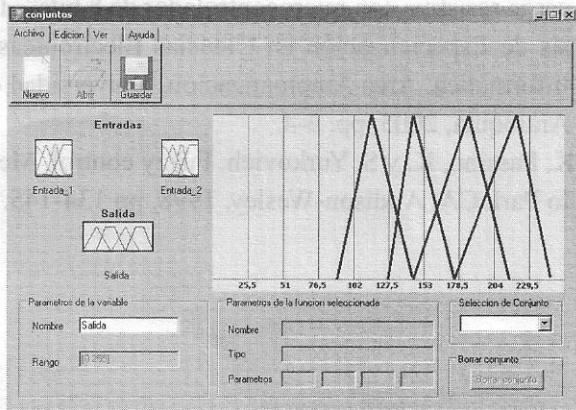


Figura 21. Conjuntos universo de salida (pwm)

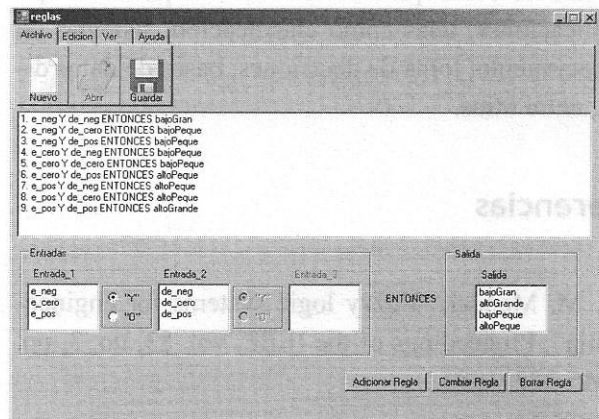


Figura 22. Definición de base de reglas

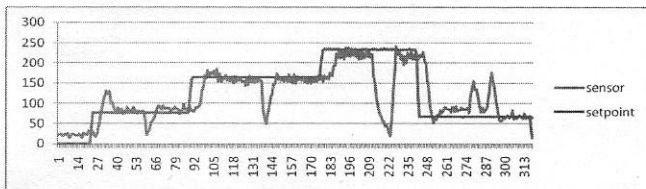


Figura 23. Señal tomada de la planta

este trabaja con la derivada del error que ayuda a predecir el comportamiento del controlador.

4. Futuros desarrollos

Después del trabajo realizado, quedan expectativas por cubrir. Entre los futuros trabajos a realizar se podrían nombrar los siguientes:

- Hacer extensiones para otras familias de microcontroladores.
- Implementar otro tipo de controladores basados en lógica difusa, tales como Takagi Sugeno, o lógica difusa tipo II.
- Implementar comunicación RS232, e incluso Ethernet.
- Implementar algoritmos de optimización para reducir el tamaño de los datos en el caso de superficies.

5. Conclusiones

Almacenar unos cuantos puntos del mapa no lineal para luego interpolarlos, es una solución sencilla en el caso

unidimensional; para los casos multidimensionales, se vuelve un problema de optimización. En el enfoque más sencillo se podría resolver con mínimos cuadrados; pero implementarlos sobre un procesador empujado sería casi tan complejo como el mismo proceso de inferencia difusa. La implementación de sistemas difusos por tabla es quizás la solución más sencilla en términos computacionales. Sin embargo, con el aumento de la dimensionalidad y la resolución se vuelve poco práctico, puesto que se necesita memorias muy amplias para almacenar estas tablas.

La herramienta desarrollada no solo tiene utilidad en controladores, sino que también funciona para otro tipo de aplicaciones, tales como clasificación, sistemas de reconocimiento, toma de decisiones, bases de datos difusas, entre otras.

Referencias

[1] J. M. Mendel, "Fuzzy logic systems for engineering". Proceedings of the IEEE, vol. 83, no. 3, pp. 345-377, 1995.

[2] L. A. Zadeh, "Fuzzy Logic, neural networks and soft computing". Communications of the ACM, vol. 3, no. 3, pp.77-84, 65-84, 88-94, 1994.

[3] W. Pedrycz, Fuzzy control and Fuzzy systems". Toronto: Ed. John Wiley & Sons Inc., 1993.

[4] F. Guzmán e I. Piñeros. Análisis matemático e implementación de un sistema multitanque a escala para docencia e investigación en un laboratorio de control en la Facultad Tecnológica. Bogotá: Universidad Distrital Francisco José de Caldas, Facultad Tecnológica, Ingeniería en Control Electrónico, 2009, pp. 40-107.

[5] G. Meneses, "Control difuso de temperatura para carga resistiva con microcontrolador de 8 bits". Tesis de Especialización en Ciencias Electrónicas e Informática: Área Automatización, Universidad de Antioquia, 2005, pp. 3-4.

[6] K. Passino, K. y S. Yurkovich. Fuzzy control. Menlo Park CA: Addison-Wesley, 1998, pp 134-145.

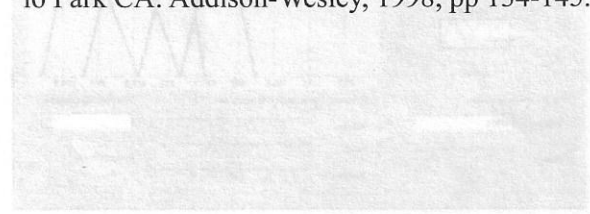


Figura 21. Señal tomada de la planta

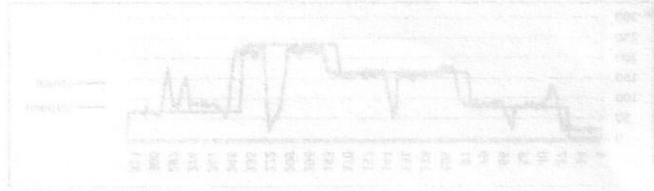


Figura 23. Señal tomada de la planta