

# Sistema de control difuso usando IVs

<sup>1</sup>Romeo Urbieta Parrazales, <sup>2</sup>Luis Pastor Sánchez Fernández, <sup>3</sup>Diego Ramírez Martínez

<sup>1,2,3</sup> Centro de Investigación en Computación  
Instituto Politécnico Nacional

Av. Juan de Dios Bátiz s/n C. P. 07738 México D. F., Tel. 57296000 Ext. 56579.

<sup>1</sup>urbieta@cic.ipn.mx, <sup>2</sup>lsanchez@cic.ipn.mx, <sup>3</sup>diegoramirez46@hotmail.com

**Resumen.** Este trabajo presenta la implementación de un sistema de control difuso aplicado a la variable de velocidad de un motor de corriente continua usando *Instrumentos Virtuales* para controlar y monitorear dicha variable. Los parámetros del controlador difuso están basados en pruebas realizadas. El hardware del sistema consiste en: una Computadora Personal Pentium tres, una interfaz de conversión analógica a digital y digital analógica de alta resolución instalada en un bus PCI, y el proceso en si. El software fue de dos clases: básico y de aplicación. El software básico fue Windows noventa y ocho como plataforma de trabajo y como software de aplicación, *Instrumentos Virtuales*. Se presentan diferentes esquemas de desarrollo del control difuso así como resultados del mismo en instrumento virtual.

**Palabra Clave:** Instrumentos Virtuales, Control Difuso.

## 1. Introducción

Muchos sistemas reales son difíciles o imposible de modelar matemáticamente, lo cual es requerido para el diseño de muchos algoritmos de control tradicional. Otros procesos que pueden o no pueden ser modelados matemáticamente son demasiado complejos o no lineales para ser controlados con estrategias tradicionales. Para esto, un controlador difuso puede ser implementado rápida y fácilmente. La lógica difusa es un método de decisión basado en reglas y algunas de sus aplicaciones es en sistemas expertos y en control de procesos. Todo proceso que una persona puede controlar manualmente de acuerdo a su experiencia, es susceptible de ser controlado con control difuso ya que las reglas lingüísticas que una persona experta puede describir de una forma intuitiva y general puede ser directamente trasladada a una base de reglas[1]. La base de la lógica difusa es la teoría de los conjuntos difusos que fue desarrollada por Lotfi Zadeh en la década de los 1960s.[2]

El módulo de construcción primaria de los sistemas de lógica difusa es la *Variable Lingüística*. Cuando se crea una variable lingüística, primero se debe determinar cuantos términos definen a la variable lingüística. En muchas aplicaciones, entre tres y siete términos son suficientes para definir una variable lingüística. No tiene sentido usar menos de tres términos, porque muchos conceptos lingüísticos tienen por lo menos dos términos extremos y un término medio entre ellos. Por otro lado, los sistemas lingüísticos que usan más de siete términos son difíciles de manejar porque los humanos usan su memoria de corto término para interpretar cantidades técnicas y

nuestra memoria de corto término solo puede procesar hasta siete símbolos simultáneamente.

Las variables lingüísticas usualmente tienen un número impar de términos porque ellos son definidos simétricamente e incluyen un término medio entre los extremos. Como un punto de arranque, usar por lo menos tres o cinco términos en las variables de entrada y cinco o siete términos en las variables de salida.

Si no hay información disponible a priori, empezar con términos igualmente espaciados dentro del rango de la variable asociada, con cada término completamente traslapado con sus términos adyacentes. Se debe cubrir la región estable deseada del sistema con más términos lingüísticos que tengan un pequeño intervalo de influencia en lugar de tratar de cubrir las regiones de los bordes que tienen solo unos pocos términos lingüísticos con un gran intervalo de influencia. Una distribución de términos hecha de esta forma, hace al controlador más sensible dentro de la región de estado estable del sistema.

Los efectos de disturbios en los valores de entrada tales como ruidos, deben ser tomados en cuenta. No crear funciones de pertenencia con un intervalo de influencia que sea más pequeño que la amplitud de la señal de ruido.

Para construir una base de reglas, definir una regla para cada combinación de términos de las variables de entrada usadas en la parte IF de la regla. Entonces seleccionar el término más conveniente de la variable de salida para especificar la parte THEN de cada regla. Para un sistema difuso, el número total de reglas posibles es  $N = p^m$ , donde:  $p$  = número de términos para cada variable de entrada,  $m$  = número de variables de entrada y  $N$  = número de reglas. Se puede cambiar el comportamiento de un controlador para una cierta situación de entrada, modificando una regla particular. Modificar una regla tiene mayor influencia en la característica del controlador que las modificaciones a las funciones de pertenencia. Un controlador difuso tiene una estructura estática y determinística.

## **2. Instrumentos Virtuales (IVs).**

El lenguaje de programación gráfica LabVIEW es conocido como una buena herramienta para la construcción de sistemas adquisición de datos, de instrumentación y de control. Después de la primera versión, aparecida para Macintosh en 1986 y para Windows y Unix en 1992, en las sucesivas ha ido ofreciendo a los usuarios una mayor facilidad y rapidez en la programación, disminuyendo radicalmente los tiempos para la elaboración de soluciones.

Los sistemas operativos de tiempo real son muy caros y exigen especialistas para trabajar con los mismos. De aquí el interés de la solución (software y hardware) para control en tiempo real que ofrece ahora National Instruments.

Con una relación precio/prestaciones realmente bueno, permite ampliar un sistema basado en PC Windows con un sistema de control en tiempo real, que se integra con

la instrumentación, la adquisición de datos, accionamientos, etc. creando una solución completa [3].

### 3. Implementación del Controlador con Instrumentos Virtuales.

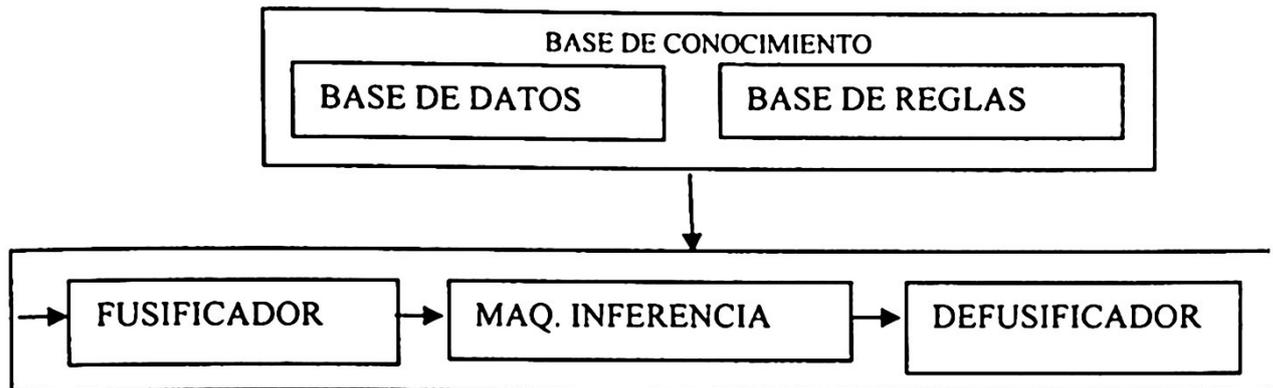


Fig. 1. Diagrama de bloques de un CLD.

El controlador difuso es un sistema que consta de cuatro componentes básicos: fusificador, máquina de inferencia, defusificador y la base de conocimiento. Se observa que el primer paso es convertir las señales de los sensores a variables lingüísticas (fusificación). En seguida se ejecuta el paso de Inferencia difusa para derivar conclusiones de la base de reglas como valores lingüísticos y por último este valor lingüístico se convierte a un valor real en el paso de defusificación que es la salida de control analógica.

La inferencia difusa consiste de dos componentes:

Agregación – es la evaluación de la parte IF de cada regla.

Composición – es la evaluación de la parte THEN de cada regla.

Para implementar el Controlador Difuso con instrumentos virtuales se usa el kit de lógica difusa que tiene National Instruments. Este kit contiene un editor de conjuntos difusos y un editor de base de reglas. Con el editor de conjuntos difusos se crea las variables lingüísticas, los términos lingüísticos y se establecen los rangos, con el editor de la base de reglas se crean las reglas, se asignan los factores de peso, se selecciona el método de defusificación y el valor tomado por la variable de salida cuando ninguna regla está activa.

Estos parámetros se almacenan en un archivo con extensión xx.fc. En seguida se crea el programa gráfico orientado a flujo de datos para realizar los cálculos necesarios en cada paso y para recuperar los parámetros contenidos en el archivo .fc. [6],[7]

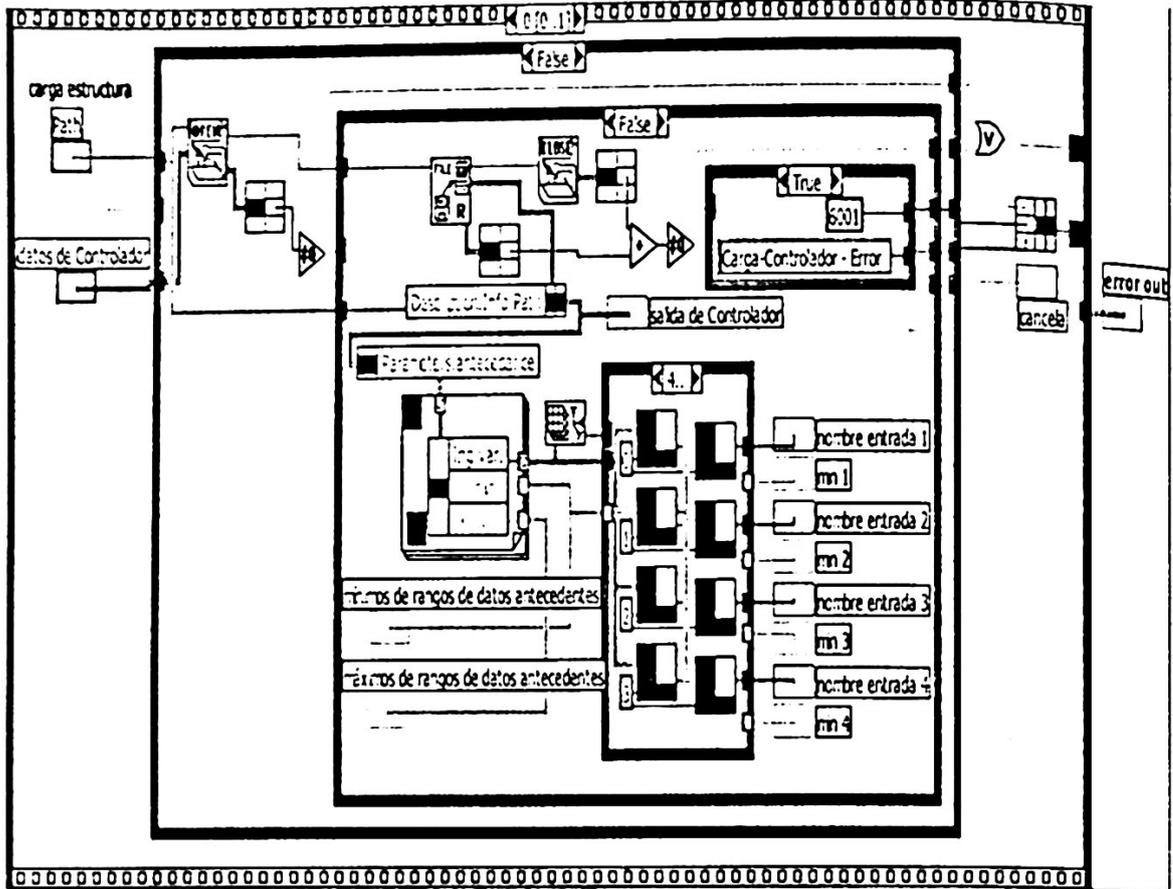


Fig. 2. Carga de parámetros del controlador

Para este sistema se crearon dos variables lingüísticas antecedentes (error y derivada del error) con tres funciones de pertenencia triangulares cada una y una variable de salida también con tres términos lingüísticos.

Se creó una base de 9 reglas con un factor de peso de 1.0 para todas. Se usó el método de inferencia max-min de Mamdani y el centro de máximos como defusificador. Estos parámetros se cargan al controlador usando el IV mostrado en la figura 2. La adquisición de datos se hizo con el IV de la figura 3.

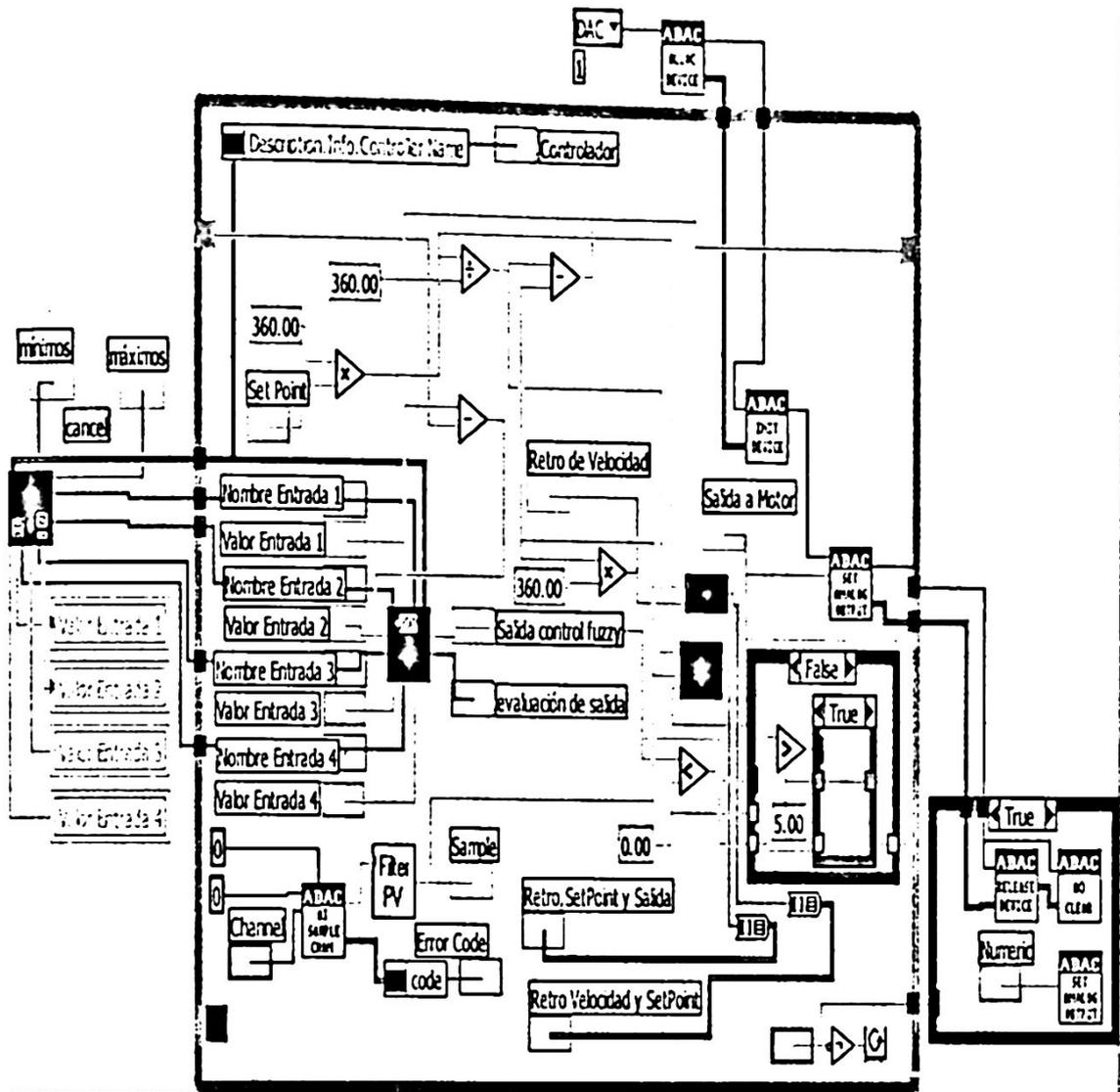


Fig. 3. Adquisición de datos

Por lo que respecta a la función Control Difuso, que es propiamente el algoritmo de control difuso, tenemos los tres pasos del controlador; la fusificación, la inferencia difusa y el defusificador.

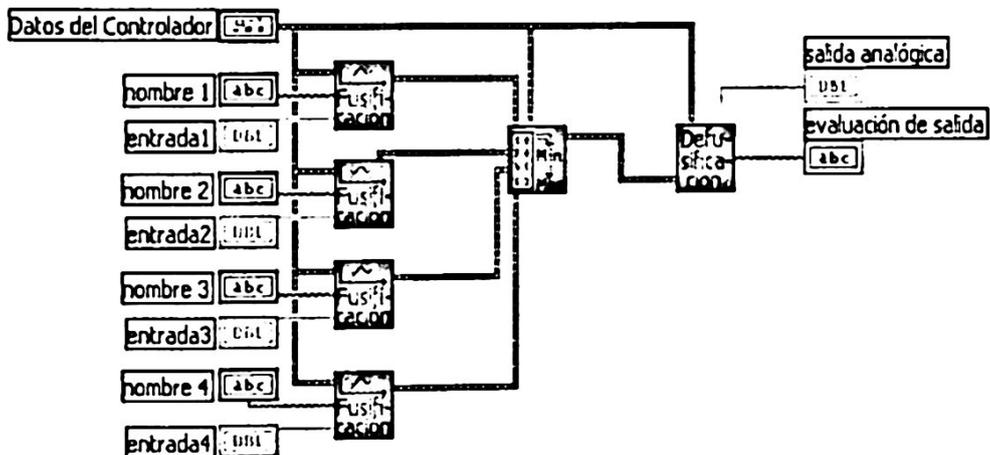


Fig. 4. Arquitectura del instrumento virtual del Controlador Difuso.

Como entrada 1, se tiene el valor del error obtenido con la diferencia entre el "Set point" y el valor de 0 a 5 voltios del sensor de velocidad, en la entrada analógica 1 de la tarjeta de adquisición de datos y la derivada del error en la entrada 2. Los nombres de cada entrada son las variables lingüísticas que vienen del archivo xx.fc (en nuestro caso es el Controlmotor1.fc) Los restantes datos del controlador como nombres de términos, rangos, tipos, valores de los puntos a,b,c,d normalizados de cada término de entrada y de salida, los datos de la base de reglas, del método de defusificación y de inferencia difusa usada, igualmente vienen del archivo .fc [6],[7].

El IV que realiza la fusificación está en la figura 3, y se usa una subIV defusificadora para cada entrada.

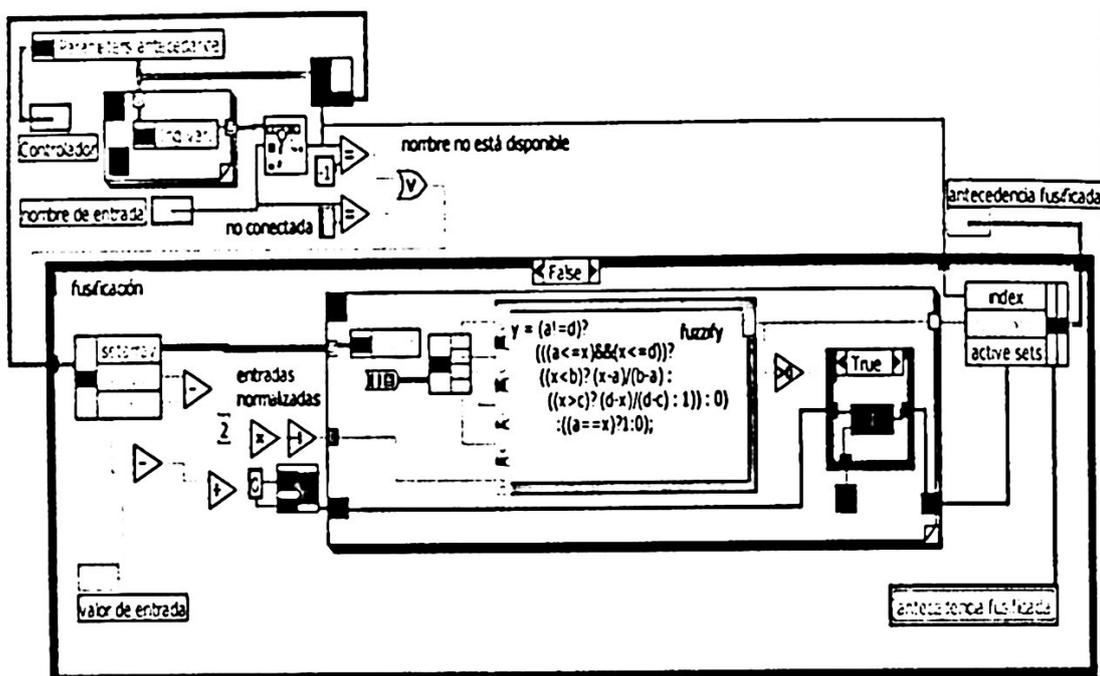


Fig. 5. Fusificación

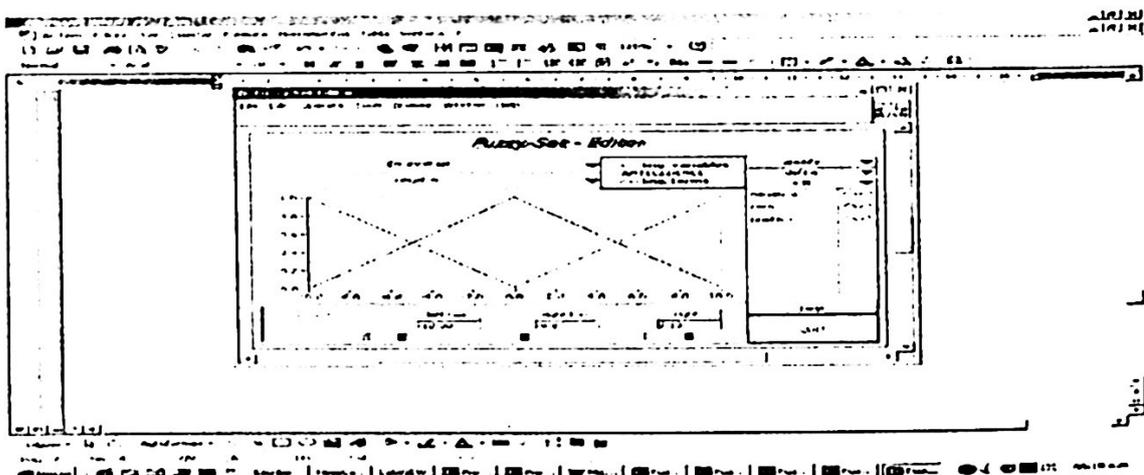


Fig. 6. Editor de conjuntos difusos

En la figura 6 tenemos las funciones de pertenencia de la variable lingüística Error y en la figura 7 se presenta un ejemplo de los puntos de las funciones de pertenencia que toma el editor difuso y los pasa al controlador.

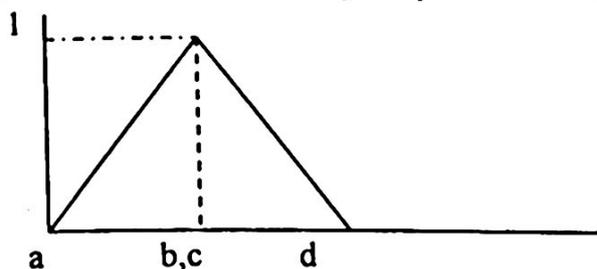


Fig. 7. Puntos de la función de pertenencia.

En la figura 5 se puede observar que el VI de fusificación toma los puntos a, b, c, d normalizados de cada uno de los términos lingüísticos y los pasa en un arreglo hasta un lazo for para hacer su auto indexación. Dentro de la estructura case general se lee el valor de entrada real y se normaliza. Un nodo de fórmulas toma los puntos a, b, c, d y hace los cálculos para obtener los grados de pertenencia en cada función. Las fórmulas de las rectas que forman las funciones de pertenencia están en el siguiente algoritmo:

El grado de pertenencia "y" =  $(x-a)/(b-a)$  si "a" es diferente de "d" y "a" es menor o igual a "x" y "x" es menor o igual a "d" y "x" es menor a "b", de lo contrario si "x" es mayor a "c"  $y=(d-x)/(d-c)$  si no  $y=1$ .

Si "x" es menor que "a" ó mayor que "d"  $y=0$ .

Si "a" es igual a "d" y "a" es igual a "x"  $y=1$ , si no  $y=0$

$y = (a \neq d)?$

$((a \leq x) \& \& (x \leq d))?$

$((x < b)? (x-a)/(b-a) :$

$((x > c)? (d-x)/(d-c) : 1) : 0)$

$:(a = x)? 1 : 0);$

(1)

La salida de la fusificación da los índices de las variables lingüísticas, los grados de pertenencia para las funciones activas y los conjuntos activos y estas pasan al VI de la Inferencia difusa que se muestra en seguida:

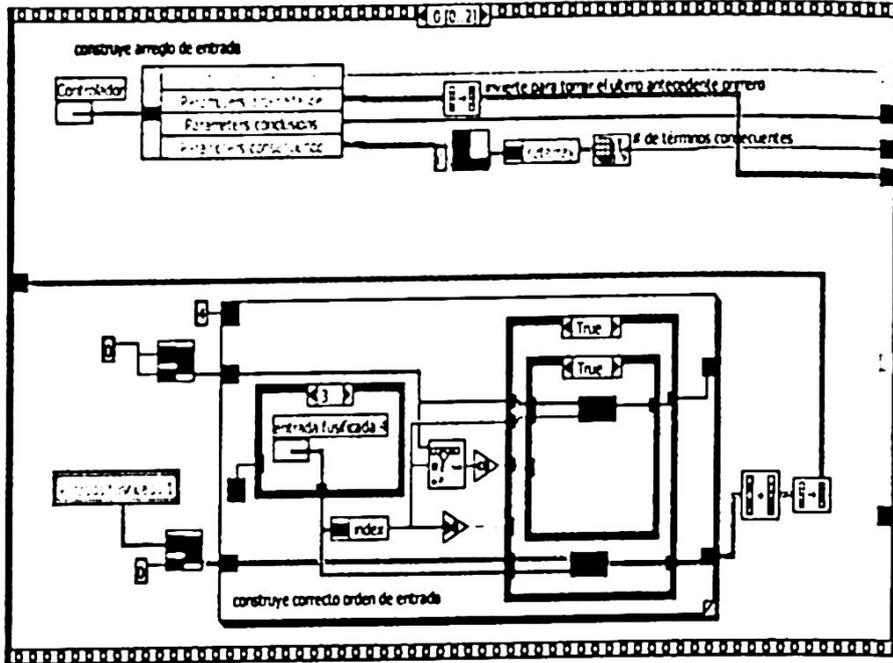


Fig. 8. IV de Inferencia difusa-marco 0

En el IV de la figura 8 se muestra el primer armazón de la secuencia en la que se toman los grados de pertenencia de los conjuntos activos y se colocan en un arreglo de una dimensión. Se recupera del archivo ControlMotor1.fc, el # de términos consecuentes, los factores de peso y los antecedentes. Todo ello se pasa al segundo marco de la secuencia mostrada en la figura 9. En el segundo marco se lee el arreglo de los grados de pertenencia y los conjuntos fuzzy activos y se efectúa con ellos la implicación de Mamdani que dice que:

$$\mu_{R_k}(x,y) = \min (\mu_A(x), (\mu_B(y)) \quad (2)$$

Obteniéndose los mínimos para las reglas encendidas y las reglas activas.

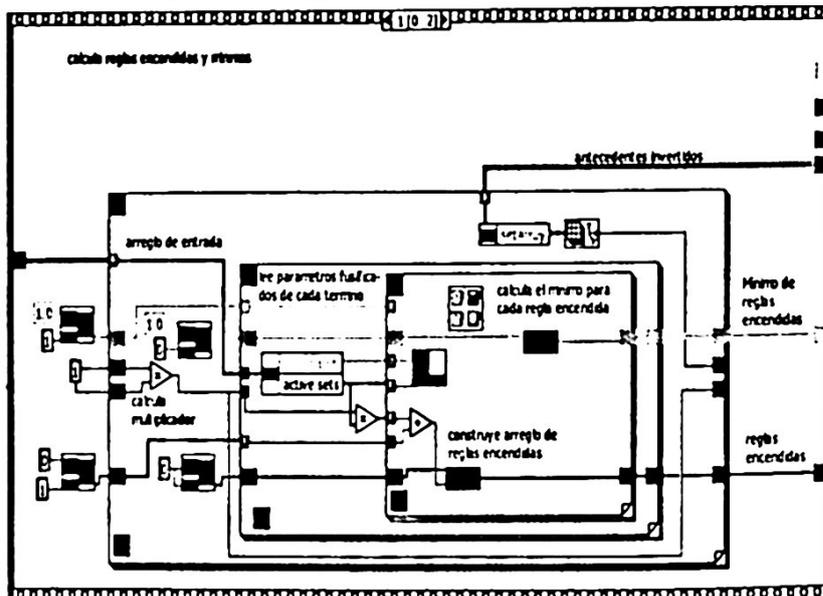


Fig. 9. VI de Inferencia difusa-marco 1.

Tabla 1. Base de reglas del controlador

	Error	Derivada error	Salida	DoS
1	Negativo	Negativo	Negativo	1.0
2	Negativo	Cero	Negativo	1.0
3	Negativo	Positivo	Cero	1.0
4	Cero	Negativo	Negativo	1.0
5	Cero	Cero	Cero	1.0
6	Cero	Positivo	Positivo	1.0
7	Positivo	Negativo	Cero	1.0
8	Positivo	Cero	Positivo	1.0
9	Positivo	Positivo	Positivo	1.0

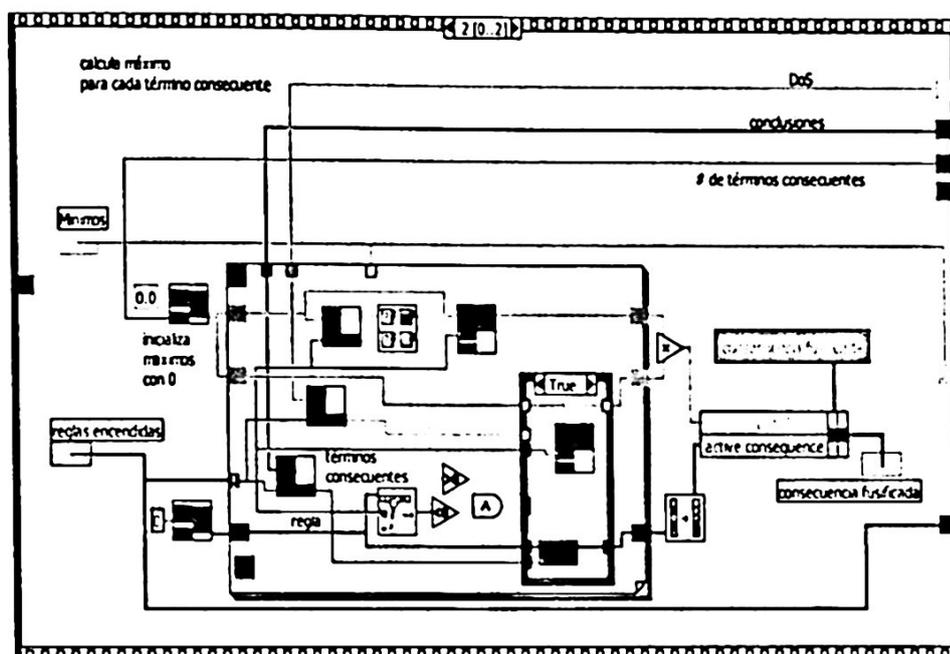


Fig. 10. IV de Inferencia difusa-marco 2.

En el tercer marco de la secuencia del IV de inferencia, figura 10, se aplica la relación:

$$R = \cup R_k \text{ y } \mu_R(x,y) = \bigvee_{k=1}^n \mu_{R_k}(x, y) \quad (3)$$

a través del operador de agregación max a la implicación de Mamdani se tiene:

$$\begin{aligned} \mu_R(x,y) &= \max \mu_{R_k}(x, y) \\ \mu_R(x,y) &= \max (\min (\mu_A(x), \mu_B(y))) \end{aligned} \quad (4)$$

Corrigiendo la salida con el factor de peso de cada regla, tenemos los grados de pertenencia de los conjuntos difusos de salida y las consecuencias activas.

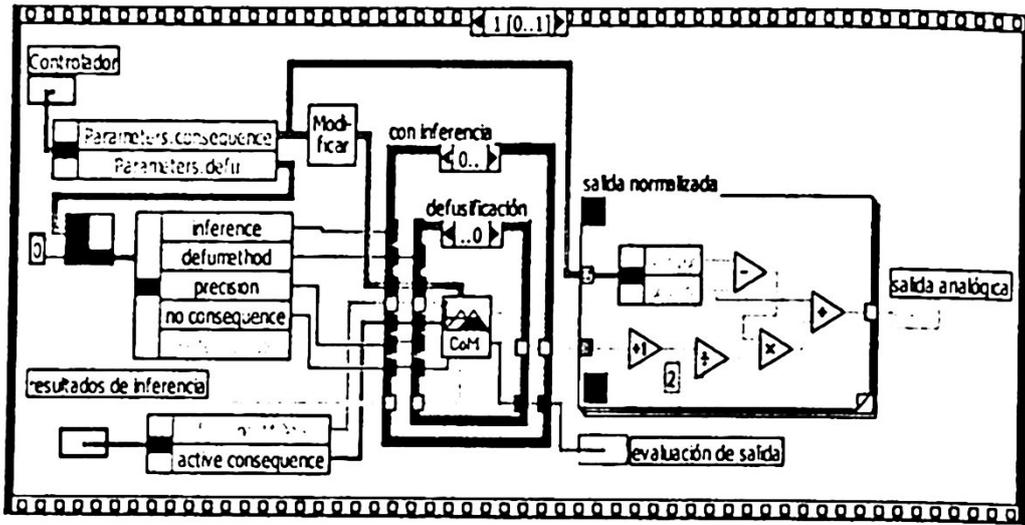


Fig. 11. IV de defusificación

Antes de realizar la defusificación se debe modificar los parámetros consecuentes configurados en la base de datos. Esto se hace con el IV de la figura 12, [6].

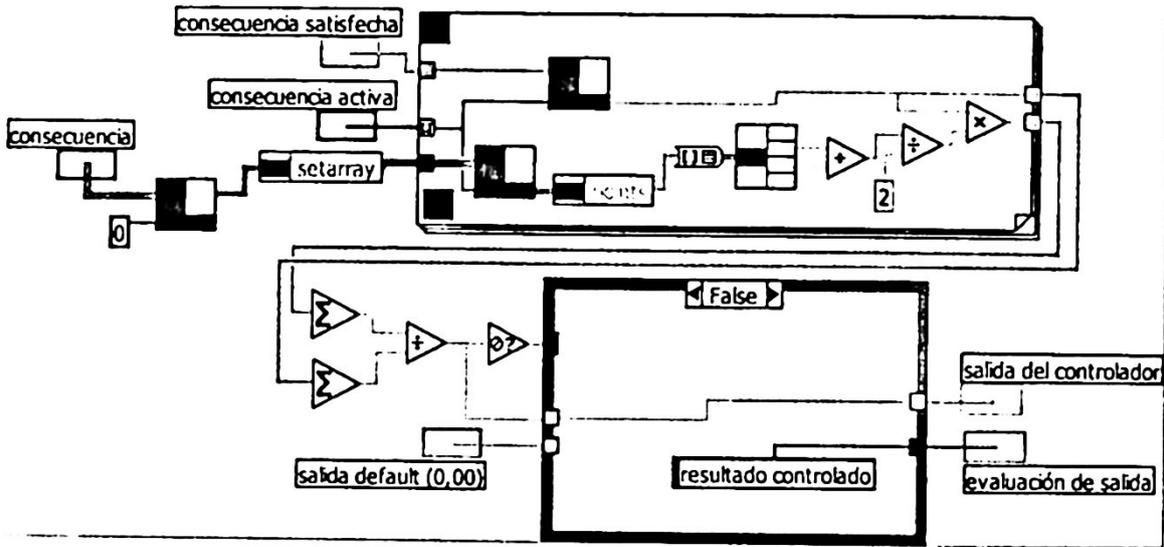


Fig. 12. IV de Modificación

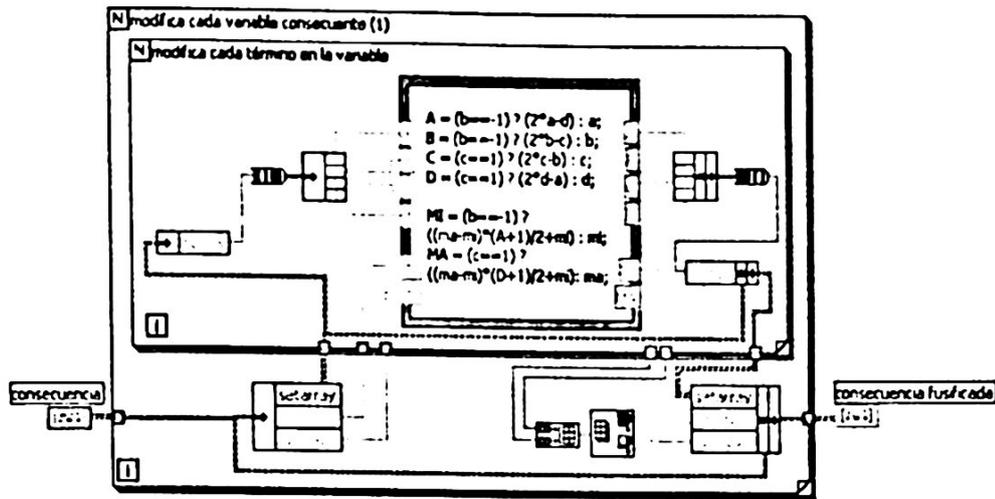


Fig. 13. IV de defusificación

En la figura 13 tenemos el IV de defusificación que recibe los grados de pertenencia de los conjuntos difusos de salida de la inferencia, las consecuencias activas y los puntos a, b, c, d modificados de los términos consecuentes y realiza la siguiente operación para utilizar el método de Centro de Máximos, también llamado Método de Pesos:

$$(5) \quad Z = \frac{\sum_{k=1}^m w_k z_k}{\sum_{k=1}^m w_k}$$

El IV de la figura 13 da una salida abrupta y el IV de la figura 11 termina aplicando los factores de escalamiento para obtener la salida analógica final. Se desarrollaron las pruebas y ajustes de parámetros, obteniéndose los resultados mostrados en las figuras 14 y 15.

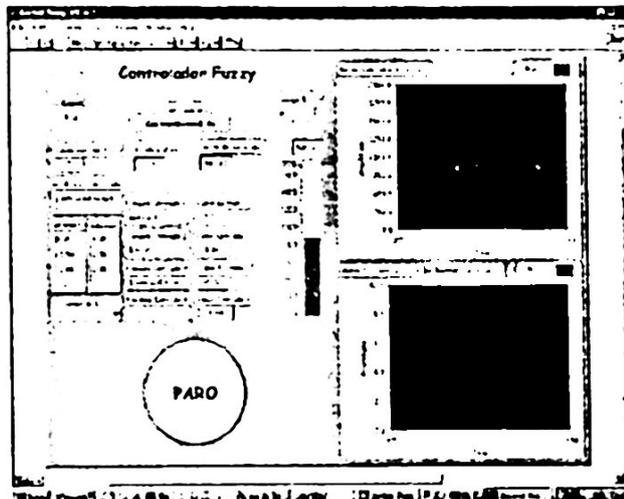


Fig. 14. Monitoreo del Control Difuso

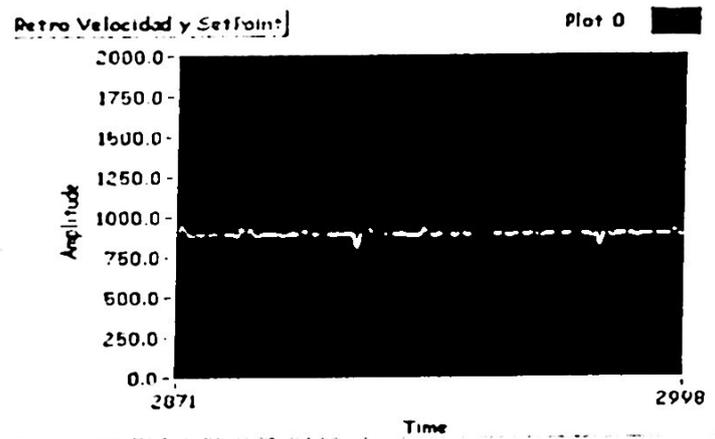


Fig. 15. Monitoreo del Set Point y la velocidad

## Conclusiones

Se desarrollo un control interactivo para regular la velocidad de un motor de CD con lógica difusa, presentando la facilidad de poderse usar para cualquier otro tipo de control, con pequeñas modificaciones, manejando hasta 4 señales de entrada. La utilidad del sistema logrado es evidente.

Por el lado de la lógica difusa aplicada al control de procesos es claro que sin ser la panacea que va a resolver todos los problemas del mundo, si puede ser una herramienta muy útil sola o combinada con redes neuronales y algoritmos genéticos, para resolver problemas importantes donde los controles tradicionales no lo han hecho o que presentan desventajas en su aplicación.

Específicamente para este caso, con el control difuso se ve mejorado el funcionamiento del motor, sobre todo durante disturbios grandes, comparado con un control PID

## Bibliografía

- [1] George and Maria Bojadziev. Fuzzy Set, Fuzzy Logic, Applications. World Scientific.1995.
- [2] W.A.R.P. Application note. Fuzzy Control of a DC Motor. SGS Thompson. Microelectronics. June 1994.
- [3] Bruce Mihura. LabView for Data Acquisition. Virtul Instrument Series. National Instrument. Prentice Hall, 2001.
- [4] Diego J. Ramirez Martinez & Romeo Urbieta P. Regulación de Velocidad de un motor de CD con Control Difuso. Implementación a través de un Algoritmo con LabView. Reporte 2003.
- [5] Jelena Godjevac. Clarifying Ideas on Programing Lenguages and Fuzzy Control Theory by Using Mobile Robots. EPFL Microcomputing Laboratory IN-F Ecublens CH-1015 Lausane.
- [6] Jelena Godjevac. Neuro-Fuzzy Controllers. Design and Application. Collection META Presses Polytechniques et Universitaires Romandes.
- [7] National Instruments. Fuzzy Logic for G Toolkit Reference Manual. March 1997 Edition. Part Number 321511A-01

# **Análisis y Procesamiento de Señales**

---

---

