

PID neuro-difuso para motores de C.A.

C. A. Carrillo Santos, L. E. Ramos Velasco
Universidad Autónoma del Estado de Hidalgo, UAEH.
Centro de Investigación en Tecnologías de Información y Sistemas, CITIS
Pachuca, Hidalgo, México +52 (771) 717-2000, Ext.6734
ing_carlos_carrillo@yahoo.com, lramos@uaeh.edu.mx
J. C. Ramos Fernández
Universidad Politécnica de Pachuca
Zempoala, Hidalgo, México.
jramos@upp.edu.mx

Resumen—En este artículo se presenta el diseño e implementación de un controlador Proporcional, Integral y Derivativo (PID) neuro-difuso para su aplicación a sistemas no lineales SISO, de modelo matemático desconocido, usando para ello un motor de inducción trifásico de CA, empleando una red neuronal de base radial con funciones de activación wavelet hijas (*wavenet*) y en cascada un filtro de respuesta infinita al impulso (IIR), adicionalmente se utilizan dos esquemas de control difuso tipo Mamdani, el primero de ellos calcula en línea los valores de las tasas de aprendizaje de la red neuronal artificial y el segundo calcula las ganancias ponderadas de un controlador PID discreto. Se presentan los resultados obtenidos experimentalmente utilizando algoritmos desarrollados en la plataforma de LabVIEW 2011.

Palabras clave: Función de aproximación, redes neuronales, control de motores, controlador PID.

I. INTRODUCCIÓN

El motor de inducción con rotor jaula de ardilla, es sin duda el más popular en la industria, debido a su robustez y bajo costo, a una velocidad constante; en años recientes se ha trabajado con distintas técnicas para controlar su velocidad, y así poder aumentar la gama de aplicaciones que se le pueden dar. En este trabajo se propone un controlador PID que como es bien sabido es también el más utilizado en la industria, con modificaciones en sintonización de las ganancias, empleando para esto redes neuronales, teoría wavelet y lógica difusa logrando un controlador PID neuro-difuso.

El funcionamiento del controlador PID en general se basa en actuar en forma proporcional, integral y derivativa sobre la señal de error $e(t)$, definida como la diferencia entre la señal de referencia $y_{ref}(t)$ y la señal de salida del proceso $y(t)$, con la finalidad de generar la señal de control $u(t)$ que manipule en forma deseada la salida del proceso.

Donde k_p , k_i y k_d son las ganancias del controlador a ser sintonizadas. Existen distintas técnicas analíticas y experimentales con el fin de sintonizar dichas ganancias. (Amstrom and Hagglund, 2006). Una alternativa es auto-sintonizar en línea dichas ganancias como en (Cruz, Ramos y Espejel, 2009), (Cruz, Ramos y Espejel, 2010) y (Sedighzadeh and Rezazadeh, 2008) donde utilizan redes

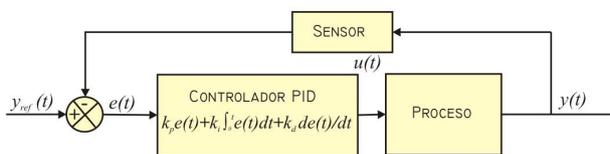


Figura 1. Esquema del controlador PID.

neuronales artificiales *wavenet* para identificar la planta y calcular esas ganancias, esta alternativa es utilizada en el presente trabajo usando distintas *wavelets*, con el fin de comparar cuál de ellas ofrece mejores resultados en la tarea de controlar la velocidad de un motor de inducción de CA trifásico. Dichos motores son mejor conocidos como los caballos de batalla de la industria (Bocker and Mathapati, 2007), son muy utilizados debido a sus ventajas como la fiabilidad, eficiencia y bajo costo en comparación a motores utilizados en aplicaciones similares. Existen algunas técnicas para manipular la velocidad de estas máquinas eléctricas como el control de campo-orientado o también llamado control-vectorial, el control sin sensores y algunos basados en redes neuronales, entre otros (Bocker and Mathapati, 2007), (Holtz, 2002) y (Merabet, Ouhrouche and Bui, 2006).

La propuesta de control descrita en este trabajo pretende ser una técnica en la cual, solo se requiere tener la señal de entrada $u(k)$ y señal de salida del motor de inducción $y(k)$ para llevar a cabo la tarea de control, sin hacer uso de su modelo matemático ni de los parámetros del mismo, a diferencia de (Islas, Ramos y García, 2010), donde presenta la identificación y control *wavenet* de un motor de corriente directa teniendo buen desempeño en lazo cerrado.

II. CONTROLADOR PID WAVENET

El esquema del controlador PID *wavenet* se muestra en la Figura 2. Donde se pueden identificar en su arquitectura que dicho controlador consta de tres etapas para poder manipular la salida de un sistema SISO no lineal en una

forma deseada. A.-Identificación del sistema, B.- Controlador PID discreto y C.- Auto-sintonización de ganancias PID. A continuación se describen cada una de las etapas.

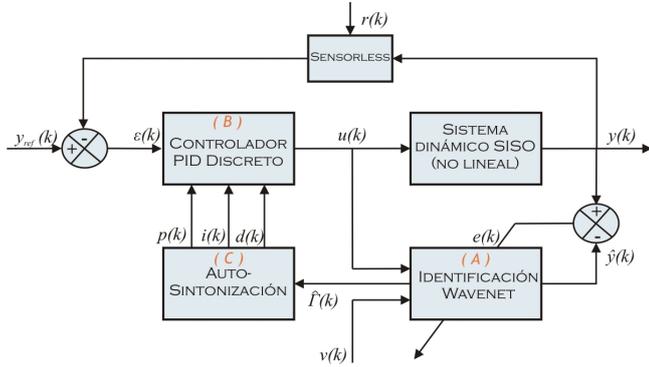


Figura 2. Esquema de control *wavenet* para un sistema dinámico SISO.

II-A. IDENTIFICACIÓN DEL SISTEMA

El proceso de identificación se hace mediante una red neuronal de base radial en la que las funciones de activación son funciones *wavelet* hijas $\psi_j(\tau)$ del tipo RASP2, dicha selección de esta *wavelet* se debe a que presenta buenos resultados y menor costo computacional de un grupo de diez *wavelets* madre tanto en simulación numérica como experimentalmente, se le añadió a la salida un filtro IIR en cascada que tiene como función filtrar (podar) las neuronas que tienen “poca” contribución en el proceso de identificación, permitiendo con esto reducir el número de iteraciones en el proceso de aprendizaje (Haykin, 2001). Dichos elementos se observan en las Figuras 3 y 4 respectivamente.

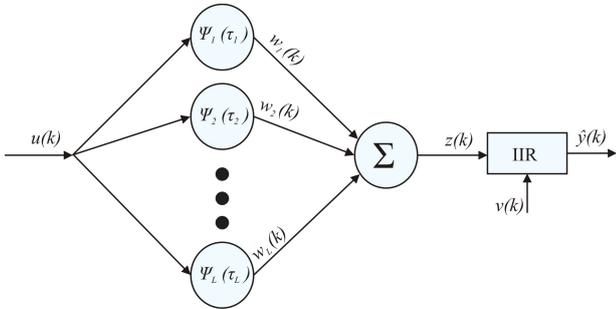


Figura 3. Diagrama *wavenet*, donde $\psi_j(\tau) = \frac{k-b_j}{a_j}$, $1 \leq j \leq L$.

La función *wavelet* $\psi(k)$ es llamada *wavelet* madre porque las diferentes *wavelets* generadas a partir de ella, por su dilatación o contracción y traslación, se les llama *wavelets* hijas $\psi_{a,b}(\tau)$, representadas matemáticamente como (Daubechies, 1992):

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (1)$$

con $a \neq 0$; $a, b \in \mathbb{R}$ y

$$\tau = \frac{k-b}{a} \quad (2)$$

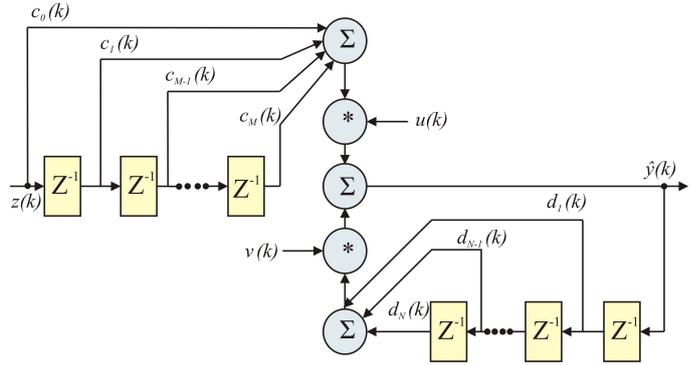


Figura 4. Estructura del filtro IIR.

TABLA I
REPRESENTACIÓN MATEMÁTICA DE DIFERENTES WAVELETS MADRE.

Wavelet madre	$\psi(\tau)$
Morlet	$\cos(\omega_0\tau)e^{-0,5\tau^2}$
RASP1	$\frac{\tau}{(\tau^2+1)^2}$
RASP2	$\frac{\tau \cos(\tau)}{\tau^2+1}$
RASP3	$\frac{\sin(\pi\tau)}{\tau^2-\frac{1}{2}}$
POLYWOG1	$\tau e^{-\frac{\tau^2}{2}}$
POLYWOG2	$(\tau^3 - 3\tau)e^{-\frac{\tau^2}{2}}$
POLYWOG3	$(\tau^4 - 6\tau^2 + 3)e^{-\frac{\tau^2}{2}}$
POLYWOG4	$(1 - \tau^2)e^{-\frac{\tau^2}{2}}$
POLYWOG5	$(3\tau^2 - \tau^4)e^{-\frac{\tau^2}{2}}$
Shannon	$\frac{\sin(2\pi\tau) - \sin(\pi\tau)}{\pi\tau}$

donde a es la variable de escala, que permite hacer dilataciones y contracciones; y b es la variable de traslación, que permite el desplazamiento en el instante k y \mathbb{R} el conjunto de los números reales. La representación matemática y la derivada parcial con respecto de b de algunas *wavelets* madre se muestran en las Tablas I y II, respectivamente:

La señal de aproximación de la *wavenet* con filtro IIR $\hat{y}(k)$ puede ser calculada como:

$$\hat{y}(k) = \sum_{i=0}^M c_i z(k-i)u(k) + \sum_{j=1}^N d_j \hat{y}(k-j)v(k) \quad (3)$$

donde

$$z(k) = \sum_{l=1}^L w_l \psi_l(k) \quad (4)$$

L es el número de *wavelets* hijas, w_l son los pesos de cada neurona en la *wavenet*, c_i y d_j son los coeficientes de adelanto y atraso del filtro IIR, respectivamente. M y N representan el número de los coeficientes de adelanto y atraso del mismo filtro, respectivamente. Los parámetros de

TABLA II
DERIVADA PARCIAL CON RESPECTO A b .

Wavelet madre	$\frac{\partial \psi(\tau)}{\partial b}$
Morlet	$\frac{1}{a}[\omega_0 \sin(\omega_0 \tau) e^{-0.5\tau^2} + \tau h(\tau)]$
RASP1	$\frac{1}{a} \frac{3\tau^2 - 1}{(\tau^2 + 1)^3}$
RASP2	$\frac{\tau \frac{\tau^2 + 1}{a} \sin(\tau) + \frac{\tau^2 - 1}{a} \cos(\tau)}{(\tau^2 + 1)^2}$
RASP3	$\frac{\frac{2\tau}{a} \sin(\pi\tau) - \pi \frac{\tau^2 - 1}{a} \cos(\pi\tau)}{(\tau^2 - 1)^2}$
POLYWOG1	$\frac{1}{a}(\tau^2 - 1)e^{-\frac{\tau^2}{2}}$
POLYWOG2	$\frac{1}{a}(\tau^4 - 6\tau^2 + 3)e^{-\frac{\tau^2}{2}}$
POLYWOG3	$\frac{1}{a}(\tau^5 - 10\tau^3 + 15\tau)e^{-\frac{\tau^2}{2}}$
POLYWOG4	$\frac{1}{a}(3\tau - \tau^3)e^{-\frac{\tau^2}{2}}$
POLYWOG5	$\frac{1}{a}(-\tau^5 + 7\tau^3 - 6\tau)e^{-\frac{\tau^2}{2}}$
Shannon	$\frac{\pi}{a} \frac{-2\pi\tau \cos(2\pi\tau) + \pi\tau \cos(\pi\tau) + \sin(2\pi\tau) - \sin(\pi\tau)}{(\pi\tau)^2}$

la *wavenet* en forma matricial están dados por:

$$\mathbf{A}(k) \triangleq [a_1(k), a_2(k), \dots, a_L(k)]^T \quad (5)$$

$$\mathbf{B}(k) \triangleq [b_1(k), b_2(k), \dots, b_L(k)]^T \quad (6)$$

$$\mathbf{W}(k) \triangleq [w_1(k), w_2(k), \dots, w_L(k)]^T \quad (7)$$

y los parámetros del filtro IIR, representados en la misma forma:

$$\mathbf{C}(k) \triangleq [c_0(k), c_1(k), \dots, c_M(k)]^T \quad (8)$$

$$\mathbf{D}(k) \triangleq [d_1(k), d_2(k), \dots, d_N(k)]^T \quad (9)$$

son optimizados por medio de un algoritmo de aprendizaje basado en mínimos cuadrados medios (LMS), tras minimizar una función de costo \mathbf{E} . Para lograr esto se define el error de estimación $e(k)$ como la diferencia entre la salida real de la planta $y(k)$ y la salida de la red neuronal y $\hat{y}(k)$, es decir

$$e(k) = y(k) - \hat{y}(k) \quad (10)$$

La función de energía del error de estimación se define como:

$$\mathbf{E} = \frac{1}{2} \sum_{k=1}^T e^2(k) \quad (11)$$

Para minimizar \mathbf{E} se aplica el método del gradiente de pasos descendentes, que utiliza las siguientes derivadas:

$$\frac{\partial \mathbf{E}}{\partial w_l(k)} = -e(k) \mathbf{C}(k)^T \Psi_l(\tau) u(k) \quad (12)$$

$$\frac{\partial \mathbf{E}}{\partial b_l(k)} = -e(k) \mathbf{C}(k)^T \Psi_{b_l}(\tau) w_l(k) u(k) \quad (13)$$

$$\frac{\partial \mathbf{E}}{\partial a_l(k)} = \tau_l \frac{\partial \mathbf{E}}{\partial b_l(k)} \quad (14)$$

$$\frac{\partial \mathbf{E}}{\partial c_m(k)} = -e(k) z(k - M) u(k) \quad (15)$$

$$\frac{\partial \mathbf{E}}{\partial d_n(k)} = -e(k) \hat{y}(k - N) u(k) \quad (16)$$

donde

$$\Psi_l(\tau) = [\psi_l(\tau), \psi_l(\tau - 1), \dots, \psi_l(\tau - M)]^T \quad (17)$$

$$\Psi_{b_l}(\tau) = \left[\frac{\partial \psi_l(\tau)}{\partial b_l(k)}, \frac{\partial \psi_l(\tau - 1)}{\partial b_l(k)}, \dots, \frac{\partial \psi_l(\tau - M)}{\partial b_l(k)} \right]^T \quad (18)$$

La actualización de los parámetros cumple con la siguiente regla (Cruz, Ramos y Espejel, 2009), (Cruz, Ramos y Espejel, 2010) y (Sedighizadeh and Rezazadeh, 2008)

$$\Delta \theta(k) = -\frac{\partial \mathbf{E}}{\partial \theta(k)} \quad (19)$$

$$\theta(k + 1) = \theta(k) + \mu_\theta \Delta \theta(k) \quad (20)$$

donde θ puede ser cualquiera de los parámetros a ser ajustados: $\mathbf{W}(k)$, $\mathbf{A}(k)$, $\mathbf{B}(k)$, $\mathbf{C}(k)$ o $\mathbf{D}(k)$. El valor de $\mu_\theta \in \mathbb{R}$ representa el coeficiente de velocidad de aprendizaje para cada uno de los parámetros.

II-B. Controlador PID discreto

Considerando un sistema dinámico SISO no lineal, pueden ser representado por la ecuación de estado discreto siguiente (Levin and Narendra, 1993), (Levin and Narendra, 1996)

$$x(k + 1) = f[x(k), u(k), k] \quad (21)$$

$$y(k) = g[x(k), k] \quad (22)$$

donde $x(k) \in \mathbb{R}^n$, $u(k), y(k) \in \mathbb{R}$ y $f, g \in \mathbb{C}$ son funciones desconocidas, siendo \mathbb{C} el conjunto de funciones suaves. La entrada $u(k)$ y la salida del sistema $y(k)$ son los únicos datos accesibles que se tienen del sistema a ser controlado. Si el sistema linealizado alrededor del punto de equilibrio es observable (Levin and Narendra, 1993), (Levin and Narendra, 1996), existe una representación entrada-salida dada por

$$y(k + 1) = \beta[\mathbf{Y}(k), \mathbf{U}(k)] \quad (23)$$

donde

$$\mathbf{Y}(k) = [y(k) \ y(k - 1), \dots, y(k - n + 1)] \quad (24)$$

$$\mathbf{U}(k) = [u(k) \ u(k - 1), \dots, u(k - n + 1)] \quad (25)$$

es decir, sí existe una función β que mapea la salida $y(k)$ y la entrada $u(k)$ y sus $n - 1$ valores pasados, en $y(k + 1)$. Un modelo alternativo de una planta desconocida que puede simplificar el algoritmo de la señal de control es el siguiente:

$$y(k + 1) = \Phi[\mathbf{Y}(k), \mathbf{U}(k)] + \Gamma[\mathbf{Y}(k), \mathbf{U}(k)] \cdot u(k) \quad (26)$$

sí los términos Φ y Γ son exactamente conocidos, el control $u(k)$ que obtiene una salida deseada $y_{ref}(k + 1)$ es

$$u(k) = \frac{y_{ref}(k + 1) - \Phi[\mathbf{Y}(k), \mathbf{U}(k)]}{\Gamma[\mathbf{Y}(k), \mathbf{U}(k)]} \quad (27)$$

Sin embargo los términos Φ y Γ son desconocidos. Por lo tanto se utiliza una red neuronal *wavenet* para aproximar las dinámicas del sistema como:

$$\hat{y}(k + 1) = \hat{\Phi}[\mathbf{y}(k), \Theta_\Phi] + \hat{\Gamma}[\mathbf{y}(k), \Theta_\Gamma] \cdot u(k) \quad (28)$$

comparando la expresión anterior (28) con la salida de la *wavenet* con filtro IIR (3), se concluye que

$$\hat{\Phi}[y(k), \Theta_{\Phi}] = \sum_{j=1}^N d_j \hat{y}(k-j)v(k) \quad (29)$$

$$\hat{\Gamma}[y(k), \Theta_{\Gamma}] = \sum_{i=0}^M c_i z(k-i) \quad (30)$$

$$z(k) = \sum_{l=1}^L w_l \psi_l(k) \quad (31)$$

Por lo tanto, si las dos no linealidades Φ y Γ son estimadas por las dos funciones de la *wavenet* $\hat{\Phi}$ y $\hat{\Gamma}$ con parámetros ajustables Θ_{Φ} y Θ_{Γ} respectivamente, la señal de control del PID que sigue la referencia deseada $y_{ref}(k)$ puede ser calculada como (Astrom, 1997), (Kuo, 1997), (Ogata, 1996):

$$u(k+1) = u(k) + p(k)[\varepsilon(k) - \varepsilon(k-1)] + i(k)\varepsilon(k) + d(k)[\varepsilon(k) - 2\varepsilon(k-1) + \varepsilon(k-2)] \quad (32)$$

donde $p(k)$, $i(k)$ y $d(k)$ son las ganancias proporcional, integral y derivativas del controlador PID, $u(k)$ es la entrada de la planta al instante k y el error de seguimiento $\varepsilon(k)$ se define como la diferencia entre la salida de la planta $y(k)$ y la señal de referencia $y_{ref}(k)$, es decir

$$\varepsilon(k) = y(k) - y_{ref}(k) \quad (33)$$

II-C. Auto-sintonización de ganancias PID

Debido a que las ganancias $p(k)$, $i(k)$ y $d(k)$ fueron consideradas dentro de la función de costo \mathbf{E} (11) pueden ser actualizadas de la siguiente manera:

$$p(k) = p(k-1) + \mu_p e(k) \hat{\Gamma}(k) [\varepsilon(k) - \varepsilon(k-1)] \quad (34)$$

$$i(k) = i(k-1) + \mu_i e(k) \hat{\Gamma}(k) \varepsilon(k) \quad (35)$$

$$d(k) = d(k-1) + \mu_d e(k) \hat{\Gamma}(k) [\varepsilon(k) - 2\varepsilon(k-1) + \varepsilon(k-2)] \quad (36)$$

donde $\hat{\Gamma}(k)$ es parte de la identificación del sistema descrita por (30). Las variables μ son las tasas de aprendizaje de las ganancias del controlador PID discreto.

III. SISTEMA DIFUSO PARA LA OBTENCIÓN DE LAS TASAS DE APRENDIZAJE μ DEL CONTROLADOR PID DISCRETO

Considere los dos niveles de control presentados en la Figura 5, donde la entrada al proceso representa una señal externa, el controlador es un PID discreto (32) y el sistema difuso calcula las tasas de aprendizaje en tiempo real. Suponga que se pueden determinar los rangos de las tasas de aprendizaje $[\mu_{p_{min}}; \mu_{p_{max}}]$, $[\mu_{i_{min}}; \mu_{i_{max}}]$ y $[\mu_{d_{min}}; \mu_{d_{max}}]$, que se encuentran en las reglas de sintonización de las ganancias del controlador PID que estabilizan en lazo cerrado un proceso que puede ser identificado por una red neuronal *wavelet*, por lo tanto la tasa de aprendizaje proporcional $\mu_p \in [\mu_{p_{min}}; \mu_{p_{max}}]$, tasa de aprendizaje integral

$\mu_i \in [\mu_{i_{min}}; \mu_{i_{max}}]$ y la tasa de aprendizaje derivativa $\mu_d \in [\mu_{d_{min}}; \mu_{d_{max}}]$. Por conveniencia las μ_p , μ_i y μ_d son normalizadas para que tengan un rango de cero a uno con la siguientes transformaciones lineales:

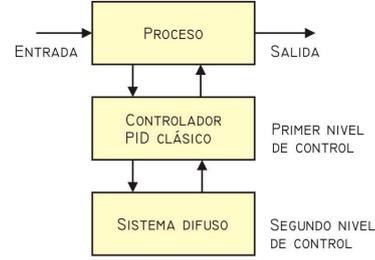


Figura 5. Sistema de control difuso con dos niveles de control.

$$\mu'_p = \frac{\mu_p - \mu_{p_{min}}}{\mu_{p_{max}} - \mu_{p_{min}}} \quad (37)$$

$$\mu'_i = \frac{\mu_i - \mu_{i_{min}}}{\mu_{i_{max}} - \mu_{i_{min}}} \quad (38)$$

$$\mu'_d = \frac{\mu_d - \mu_{d_{min}}}{\mu_{d_{max}} - \mu_{d_{min}}} \quad (39)$$

De lo anterior los parámetros sintonizados por el sistema difuso son μ'_p , μ'_i y μ'_d . Si podemos determinar estos parámetros, entonces las ganancias del PID pueden ser obtenidas con las ecuaciones (34), (35) y (36). Asumiendo que las entradas al sistema difuso son el error de identificación $e(k)$, la aproximación a su derivada $e_1(k) = [e(k) - e(k-1)]$, el error de seguimiento $\varepsilon(k)$ y la aproximación a su derivada $\varepsilon_1(k) = [\varepsilon(k) - \varepsilon(k-1)]$ el sistema difuso de sintonización consiste de dos módulos difusos como se muestra en la Figura 6.

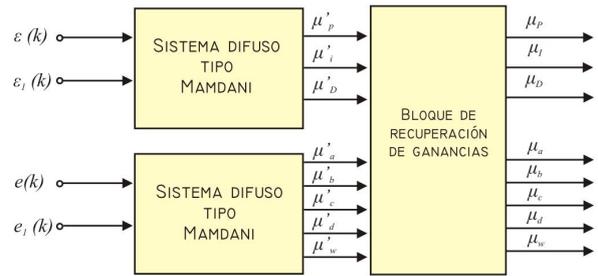


Figura 6. Sistema difuso para el PID wavenet.

Ahora las reglas difusas tienen la estructura del tipo SI-ENTONCES, que a continuación se muestran:

$$\mathcal{R}_i : \mathbf{SI} \ \varepsilon(k) \ \text{es} \ A^i \ \mathbf{Y} \ \varepsilon_1(k) \ \text{es} \ B^i \quad (40)$$

ENTONCES $\mu'_p = C^i, \mu'_i = D^i, \mu'_d = E^i$,

donde A^i , B^i , C^i , D^i y E^i son los conjuntos difusos de la i -ésima regla con $i = 1, 2, \dots, M$.

Para el error de identificación $e(k)$ tenemos la siguiente estructura de reglas difusas:

$$\mathcal{R}_i : \text{SI } e(k) \text{ es } F^i \text{ y } e_1(k) \text{ es } G^i \\ \text{ENTONCES } \mu'_A = H^i, \mu'_B = I^i, \quad (41) \\ \mu'_C = J^i, \mu'_D = K^i, \mu'_W = L^i$$

donde F^i , G^i , H^i , I^i , J^i , K^i y L^i son los conjuntos difusos de la i -ésima regla, con $i = 1, 2, \dots, M$.

La obtención de las reglas se hace de manera experimental basados en la respuesta típica de el proceso para la sintonización del PID. En la Figura 7 se muestra un ejemplo típico de la respuesta de un proceso, en donde alrededor del punto a_1 , se requiere una señal de control grande y una respuesta rápida, para producir una señal de control grande, la ganancia proporcional μ'_P tiene que ser grande, la ganancia derivativa μ'_D pequeña y la ganancia integral μ'_I grande. Para una μ'_P y μ'_D dadas, la ganancia integral es inversamente proporcional a μ'_I , por consecuencia para una ganancia integral μ'_I grande α tiene que ser pequeña, entonces la regla alrededor del punto a_1 se puede escribir como:

$$\mathcal{R}_1 : \text{SI } \varepsilon(k) \text{ es } EN \text{ Y } \varepsilon_1(k) \text{ es } E1N \\ \text{ENTONCES } \mu'_p = Pg, \mu'_i = Ig, \mu'_d = Dp \quad (42)$$

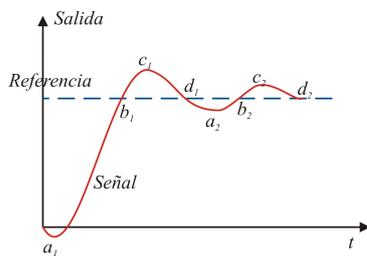


Figura 7. Respuesta típica de un proceso.

IV. RESULTADOS EXPERIMENTALES

La auto-sintonización del controlador PID neuro-difuso con la wavelet RASP2 se aplicó experimentalmente al seguimiento de una referencia de voltaje como se muestra en la Figura 8. El algoritmo es implementado en la plataforma de LabVIEW 2011 y usando una tarjeta de adquisición de datos NI-USB 6008, ambos de National Instruments.

El esquema de conexión entre la computadora, la tarjeta de adquisición de datos, la etapa de potencia y el motor de inducción trifásico jaula de ardilla se muestra en la Figura 9.

La interfaz desarrollada para la configuración del algoritmo y sus parámetros se muestra en la Figura 10.

Los parámetros de la red neuronal *wavenet* se pueden observar en la Tabla III. los valores iniciales son mostrados en la Tabla IV.

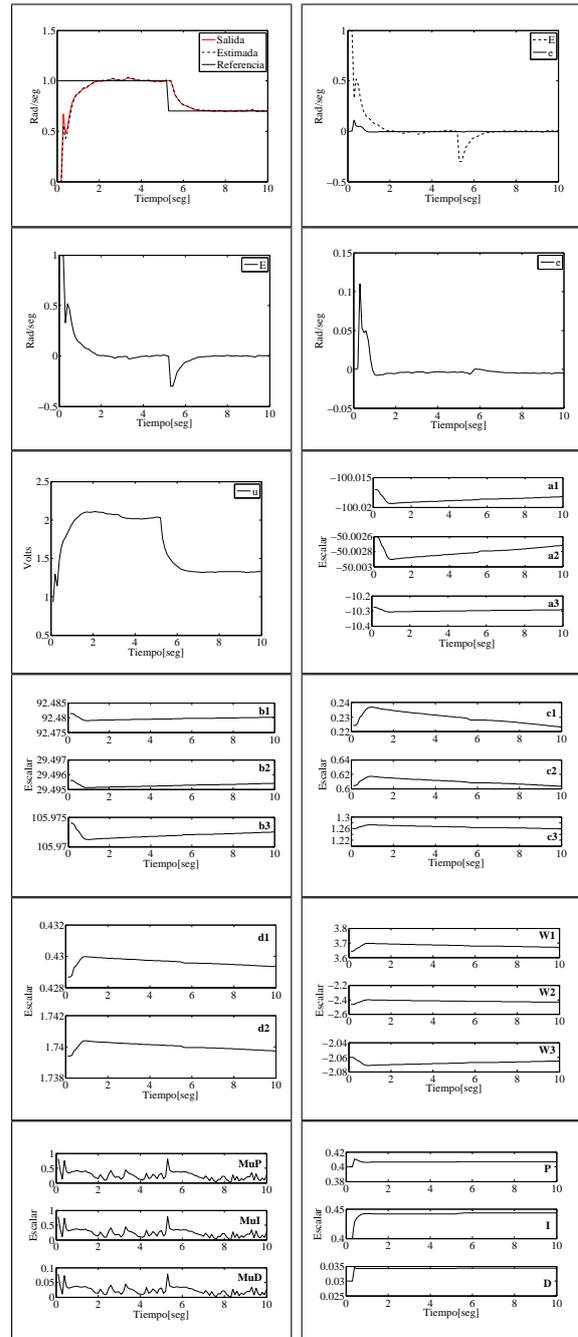


Figura 8. Valores iniciales y comportamiento de variables.

TABLA III
PARÁMETROS DE *wavenet*, FILTRO IIR Y EXPERIMENTO.

Parámetro	Valor
Neuronas	3
Coefficientes del filtro IIR C	3
Coefficientes del filtro IIR D	2
Épocas	4
Periodo de muestreo k	100 ms
Señal de persistencia v	0.13

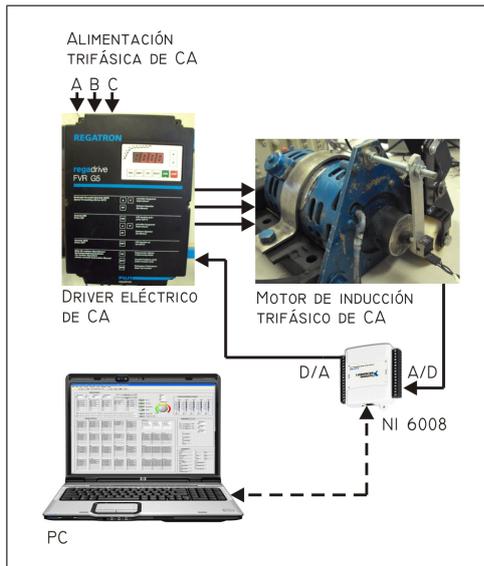


Figura 9. Esquema de la plataforma experimental.

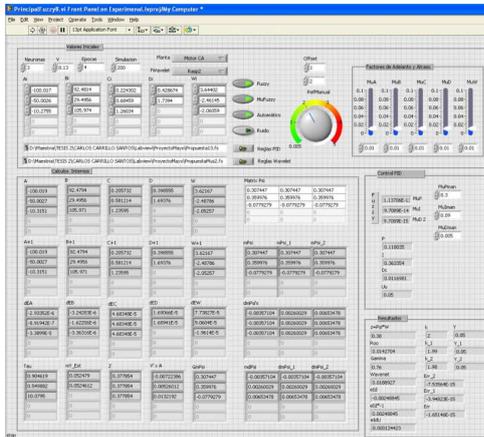


Figura 10. Interfaz gráfica de Usuario.

TABLA IV

VALORES INICIALES Y TASA DE APRENDIZAJE DEL CONTROLADOR PID wavenet.

Parámetro	Valor inicial
W	[3.64402, -2.46145, -2.659]
A	[-100.017, -50.0026, -10.2755]
B	[92.4814, 29.4956, 105.974]
C	[0.224302, 0.60459, 1.26034]
D	[0.428674, 1.7394]
<i>p</i>	0.4
<i>i</i>	0.4
<i>d</i>	0.03
Parámetro	Valor
μ_w	0.01
μ_a	0.01
μ_b	0.01
μ_c	0.01
μ_d	0.01
μ_p	0.3
μ_i	0.09
μ_D	0.005

V. CONCLUSIONES

A partir de las simulaciones y experimentos realizados, se pudo comprobar que la combinación de ventajas de la lógica difusa con las redes neuronales y el controlador PID ofrecen una alternativa viable para controlar sistemas no lineales SISO. Con lo se concluye que el PID neuro-difuso presenta resultados satisfactorios para plantas de naturaleza estable.

VI. AGRADECIMIENTOS

CONACYT por la beca de Maestría otorgada durante el periodo agosto 2009 - junio 2011, con numero de registro 230628

REFERENCIAS

- Astrom, K. (1997). Computer-Controlled Systems, theory and design. Prentice-Hall.
- Astrom, K. and T. Hagglund (2006). Advanced PID Control. ISA.
- Bocker, J. and Mathapati S. (2007). State of the Art of Induction Motor Control. in Electric Machines and Drives Conference, 2007. IEDMC'07. IEEE International. Paderborn, Germany, 1459-1464.
- Cruz, J., L. Ramos y M. Espejel (2009). PID Wavelet Auto-Sintonizado con una Red Neuronal Wavenet. V Semana Nacional de Ingeniería Electrónica SENIE09, Ocotlán, Jalisco, México, 7-9 Octubre. ISBN 978-607-477-073-5, pp 23-31.
- Cruz, J., L. Ramos y M. Espejel (2010). A Self-Tuning Wavelet PID Controller Using Wavelet Networks. 20th International Conference on Electronics Communications and Computers (CONIELE-COMP 2010). Puebla, México.
- Daubechies I. (1992). Ten Lectures on Wavelets. SIAM
- Haykin S. (2001). Kalman Filtering and Neural Networks. Wiley
- Holtz J. (2002). Sensorless Control of Induction Motor Drives. in Proceedings of the IEEE. Wuppertal, Germany. Vol. 90, No. 8, pp. 1359-1394.
- Islas O., Ramos L. E. y García J.(2010). Identificación y Control Wavenet de un Motor de CD Congreso Anual de la Asociación de México de Control Automático(AMCA), Puerto Vallarta, Jalisco, Mexico, 6-8 Octubre, 2010.
- Kuo B. (1997). Sistemas de Control Digital. C.E.C.S.A.
- Levin, A. and K. Narendra (1993). Control of Nonlinear Dynamical Systems Using Neural Networks: Controllability and Stabilization. IEEE Transactions on Neural Networks, Vol. 4, pp. 192-206.
- Levin, A. and K. Narendra (1996). Control of Nonlinear Dynamical Systems Using Neural Networks - Part II: Observability, Identification, and Control. IEEE Transactions on Neural Networks, Vol. 7, pp. 30-42.
- Merabet A., Ouhrouche M. and Bui R.(2006). Neural Generalized Predictive Controller for Induction Motor. International Journal of Theoretical and Applied Computer Sciences. Quebec, Canada. pp. 83-100.
- Ogata K.(1996). Sistemas de Control en Tiempo Discreto. Prentice Hall.
- Sedighzadeh M. and A Rezazadeh(2008). Adaptive PID Control of Wind Energy Conversion Systems. Using RASPI Mother Wavelet Basis Function Network. Proceeding of World Academy of Science, Engineering and Technology, pp. 269-273.