

IMPLEMENTACIÓN EN TIEMPO REAL DE UN SISTEMA DE CONTROL NEURO-DIFUSO PARA EL PÉNDULO INVERTIDO

Arturo Rojas-Moreno, César Nuñez-Ocola, Fernando Merchan-Gordillo, Luis E. Córdova-Sosa
arojas@uni.edu.pe cnunez@upc.edu.pe fmg@uni.edu.pe lcordova@fiec.org
Sección de Postgrado - Facultad de Ingeniería Eléctrica y Electrónica

RESUMEN

Este artículo presenta el procedimiento de diseño e implementación de un controlador difuso TKS (Takagi-Kosko-Sugeno) que emplea la técnica de sintonía ANFIS (Adaptive Neuro-Fuzzy Inference System). La metodología desarrollada se aplica a un caso clásico de estudio, pero de características particularmente atractivas para poner a prueba cualquier tipo de controlador: el Sistema Péndulo Invertido (SPI). El SPI comprende una varilla unida por uno de sus extremos a un pivote montado sobre un carro que se desplaza en línea recta sobre rieles. El objetivo de control es mantener la varilla en posición vertical mediante una fuerza aplicada al carro. El procedimiento de diseño requiere de: el modelado del SPI, la selección de una estrategia de control según las variables de entrada/salida del modelo, la selección de la estrategia de control conforme a los tipos de lazos de control sobre las variables, y, el diseño del controlador difuso usando ANFIS para optimizar su rendimiento. Luego de simulado el sistema, la implementación en tiempo real se realiza usando una interfaz HW/SW (Hardware/Software). El software de control está escrito en C++. Los resultados experimentales obtenidos validan el procedimiento de diseño desarrollado.

Términos Claves: Inteligencia Artificial, Diseño + Controladores + Lógica Difusa, Takagi-Kosko-Sugeno, Neuro-Difuso, ANFIS, Péndulo Invertido + Equilibrio de una Varilla + Carro & Péndulo.

ABSTRACT

This paper presents the design and implementation procedure of a Fuzzy Logic Controller (FLC) of the type Takagi-Sugeno-Kahn (TSK) using ANFIS (Adaptive Neuro-Fuzzy Inference System) tuning. The developed control methodology is applied to a classic benchmark, but with attractive particular characteristics to proof the controller: The Inverted Pendulum System (IPS), which consists of a pole mounted on pivot point of a movable cart. The motion of the cart on a pair of rails is intended to hold the pole in a vertical position. The design procedure starts with the modeling of the ISP, followed by setting up the control strategy that takes into account the I/O variables of the dynamic model. After identifying the type of control loops and variables, we shall perform the design of the Fuzzy Logic Controller (FLC) using ANFIS tuning to optimize the controller performance. Finally, already verified through simulation, the IPS is implemented using a HW/SW (Hardware/Software) controller interface, employing programs written in C/C++ code. Experimental results will validate the design procedure depicted above.

Keywords —Artificial Intelligence, Design + Controllers + Fuzzy Logic, Takagi-Sugeno-Kahn, Neuro-Fuzzy, ANFIS, Inverted Pendulum + Broom Balancing + Cart & Pole.

INTRODUCCIÓN

A pesar que el SPI (fig. 1) es un problema atacado generalmente utilizando teoría de control clásica y moderna, debido a las características particulares de

la planta, es un excelente caso de estudio para implementar y validar nuevas estrategias de control basadas en inteligencia artificial, ya sea por medio de

redes neuronales, algoritmos genéticos, lógica difusa, o esquemas unificados neuro-difuso, genético-difuso, neuro-genético y neuro-difuso-genético. El esquema planteado en este trabajo emplea el control basado en "razonamiento difuso" que cualquier ser humano aplicaría para equilibrar una varilla sobre su dedo índice. Esto es, ver (senzar) la posición de la varilla, enviar esta información al cerebro (el controlador difuso) para allí generar la fuerza que el brazo (el amplificador) debe ejercer para que la varilla permanezca horizontal sobre el dedo (el carro montado sobre las rieles).

MODELO DEL SISTEMA PÉNDULO INVERTIDO

La fig. 1 muestra el SPI donde θ es la posición angular de la varilla, x es la posición del carro, F es la fuerza aplicada al carro, u es el voltaje de control que ingresa al amplificador, g es la aceleración de la gravedad, lv es la longitud de la varilla, le es la distancia de pivote al centro de masa de la esfera, me es la masa de la esfera, mv es la masa de la varilla, mc es la masa del carro, Ie es la inercia de la esfera, Iv es la inercia de la varilla, Ka es la ganancia del amplificador, Ra es la resistencia de armadura del motor, La es la inductancia de armadura del motor, Vb es el voltaje contraelectromotriz, Km es la constante del motor, Jn es la inercia del eje primario del motor, Bn es la fricción viscosa del eje primario de motor. Jo es la inercia del eje secundario del motor, Bo es la fricción viscosa del eje secundario del motor, θn es el ángulo de giro del eje primario del motor, n es el factor de reducción de velocidad, Jp es la inercia de la polea, y rp es el radio de la polea. Los parámetros del sistema SPI son:

$Ka = 14.9$	$g = 9.8$	$mv = 0.0638$
$me = 0$	$mc = 0.2000$	$lv = 0.7850$
$le = 0.7850$	$Ix = 0.0131$	$Kx = 1.2826$
$Jx = 6.2819$	$Bx = 12.2968$	(Sistema Internacional)

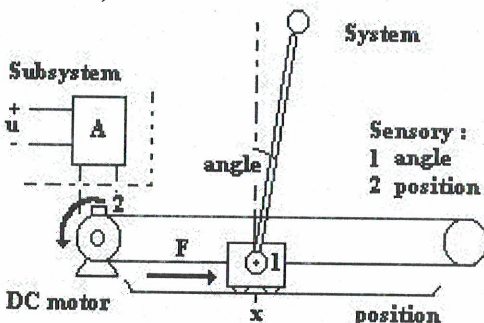


Fig. 1 Configuración del Sistema Péndulo Invertido (SPI).

Las fórmulas que gobiernan el SPI son [10], [9]:

$$\ddot{\theta} = \frac{M_1 \cdot g \cdot \text{sen} \theta - F \cdot \cos \theta - M_2 \cdot \dot{\theta}^2 \cdot \cos \theta \cdot \text{sen} \theta}{(M_1/M_2)I - M_2 \cdot \cos \theta} \quad (1)$$

$$\ddot{x} = \frac{F + M_2 \left(\dot{\theta}^2 \cdot \text{sen} \theta - \ddot{\theta} \cdot \cos \theta \right)}{M_1} \quad (2)$$

$$F = Kx \cdot Ka \cdot u - Jx \cdot \ddot{\chi} - Bx \cdot \dot{\chi} \quad (3)$$

donde:

$$M_1 = mc + me + mv;$$

$$M_2 = me \cdot le + mv \cdot lv/2;$$

$$Ix = Ie + Iv;$$

$$Kx = \frac{Km}{Ra \cdot n \cdot rp};$$

$$Jx = \frac{Jeq}{n^2 \cdot rp^2};$$

$$Bx = \frac{Beq}{n^2 \cdot rp^2} + \frac{Kb \cdot Km}{n^2 \cdot rp^2 \cdot Ra};$$

$$Jeq = Jn + n^2 \cdot (Jo + Jp);$$

$$Beq = Bm + n^2 \cdot Bo;$$

$$Vb = Kb \cdot \dot{\theta} n;$$

ESTRATEGIA DE CONTROL

La estrategia de control a emplear requiere seleccionar adecuadamente las variables de entrada y salida del IPS.

A. Teniendo en Cuenta las Variables de Entrada/Salida

Del modelo del SPI podemos seleccionar (ver fig.2):

- Variables de estado (salida del modelo).

$$X = (x_1, x_2, x_3, x_4)^T = (\theta, \dot{\theta}, \chi, \dot{\chi})^T \quad (4)$$

- Variable de control (entrada del modelo).

$$U = u \tag{5}$$

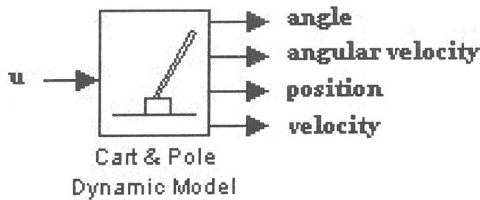


Fig. 2 Variables de estado y de control para el péndulo.

El diagrama de bloques para simulación del modelo del SPI se muestra en la fig. 3.

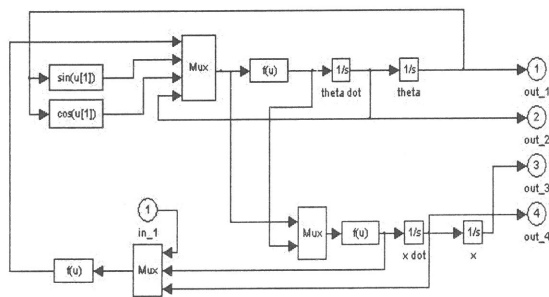


Fig.3 Diagrama de bloques del modelo del SPI.

B. Teniendo en Cuenta los Lazos de Control

Los controladores basados en Lógica Difusa (LD) permiten construir lazos de control con razonamiento difuso para cada variable que controlan, con respecto a una referencia (set point) Difusa. Para el SPI podemos establecer dos opciones de control sobre las variables del sistema:

- Control en lazo cerrado: χ (posición del carro) con set point constante o variante en el tiempo.
- Control en lazo difuso: θ (posición angular de la varilla), θ' (velocidad angular) y χ' (velocidad).

Dichas opciones de control se aprecian mejor en el esquema de la fig. 4, en donde se ha incluido el modelo del SPI dentro del diagrama de bloques con fines de simulación.

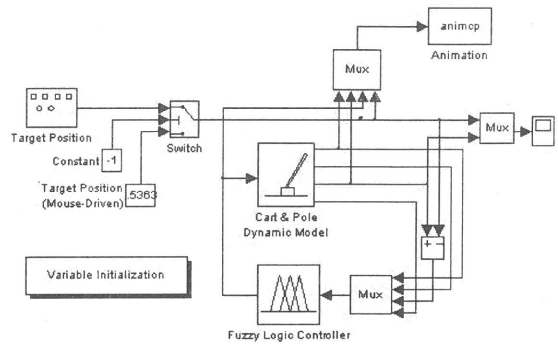


Fig. 4 Opciones de control de acuerdo a los lazos de control.

DISEÑO DEL CONTROLADOR DIFUSO

Los requerimientos que debe satisfacer el controlador basado en LD son:

- *Superar la gran inestabilidad del SPI.*- El SPI es ciertamente inestable en presencia de perturbaciones moderadas, lo cual lo hace difícil su control.
- *Trabajar con la característica no lineal del SPI.*- El modelo dinámico de la planta es no lineal. Sin embargo, la LD no requiere linealizar dicho modelo.
- *Absorber las imperfecciones del modelado.*- Debido a las fricciones no lineales, al fenómeno de saturación, a las perturbaciones externas, al ruido de medición, etc.
- *Adaptarse a las condiciones de operación no modeladas.*- Se refiere a las variables y ruido no modelados, a la variación de los parámetros con el tiempo, a las imperfecciones de toda índole, etc.

Los componentes de un algoritmo de control basado en LD se ilustran en la fig.5 en donde podemos observar que los datos pasan a través de los bloques pre-procesamiento, controlador, y post-procesamiento [1]-[6].

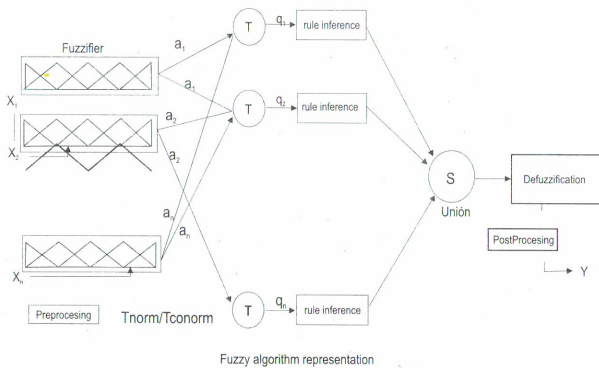


Fig. 5 Algoritmo de control basado en LD (Lógica Difusa).

El pre-procesamiento puede ser un escalamiento lineal o no lineal, así como una cuantificación en el caso que las funciones de pertenencia (o de membresía) sean discretizadas (vectores). Si no fuera así, entonces las funciones de pertenencia deben ser implementadas mediante una función sencilla. Cuando se diseña la base de reglas, el diseñador necesita considerar el número de conjuntos difusos de las entradas, así como su forma, y su traslape. Las reglas deben ser implementadas por el diseñador, a menos que se empleen medios más avanzados tales como auto-organización o redes neuronales. También debemos tomar una decisión sobre el método de activación (norma T, fig.5), que puede ser el método Producto o el método Mínimo. Luego debemos seleccionar el método de “defuzzyficación” (entendiéndose este término como la transición de la respuesta difusa a la respuesta real), y también el método de agregación de las inferencias (norma S, fig.7).

El post-procesamiento consiste en escalar la salida. En caso que el controlador sea del tipo incremental, el post-procesamiento debe incluir un integrador. La siguiente es una lista de decisiones de diseño:

- *Decisiones relacionadas con la base de reglas:* número de entradas/salidas, reglas de inferencia, universos del discurso continuo/discreto, número de funciones de pertenencia, ancho de base y traslape de las funciones de pertenencia, salida “Singleton” (basada en funciones de pertenencia tipo impulso unitario).
- *Decisiones relacionadas con el motor de inferencia:* conectivos, modificadores, operación de activación, operación de agregación y operación

de acumulación.

- *Método de Defuzzyficación:* COG (Center of Gravity), COGS (COG con Singleton), MOM (Mean of Maximums), Yager, etc.
- *Pre- y Post-procesamiento:* Escalamiento, factores de ganancia, cuantificación y tiempo de muestreo.

Algunos de las decisiones mencionadas anteriormente deben ser siempre consideradas. Otros, según el caso.

CONTROLADOR DIFUSO TIPO TKS

La fig. 6 muestra el esquema general del controlador tipo Sugeno, mientras que la fig. 7 ilustra el tipo de función de pertenencia para todas las entradas del controlador.

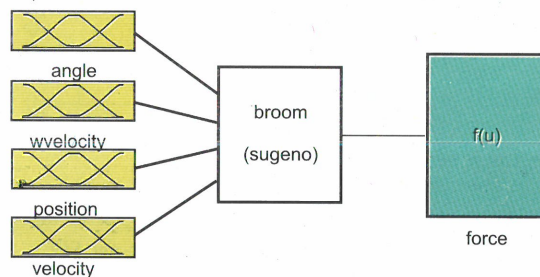


Fig. 6 Esquema general del controlador tipo Sugeno.

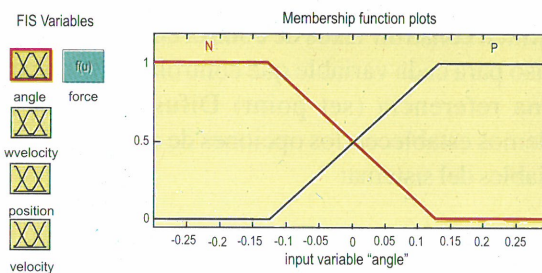


Fig. 7 Funciones de pertenencia para las entradas x1 (posición angular), x2 (velocidad angular), x3 (posición) y x4 (velocidad lineal del carro).

Una de los métodos para realizar un Sistema de Inferencia Difusa (Fuzzy Inference System: FIS) es el establecida por Takagi, Kosko y Sugeno (TKS) [7], [8] (más conocido como FIS tipo Sugeno) que consiste de un modelo sobre el cual las variables de control son caracterizadas en función de las variables de estado

del proceso. Esto se puede observar claramente en el conjunto de ecuaciones lingüísticas descritas abajo. La idea básica del FIS-Sugeno también se ilustra operacionalmente en la fig. 8, donde se muestra el Motor de Inferencia. Los conjuntos difusos de salida pueden ser Singletons, así como también combinaciones lineales de las entradas, o una función $f(u)$ de las entradas (Singletons Móviles [7]). Así tenemos

- μ_{Ri} : grado de pertenencia inferido para la i -ésima regla.
- y_{Ri} : resultado real parcial de la i -ésima regla o la posición del Singleton Móvil parcial para la i -ésima regla.
- x_1, x_2, x_3, x_4 : entradas del controlador.
- p_j, q_j, r_j, s_j, t_j : parámetros de la función $f(u)$
- de salida del controlador para la j -ésima regla.

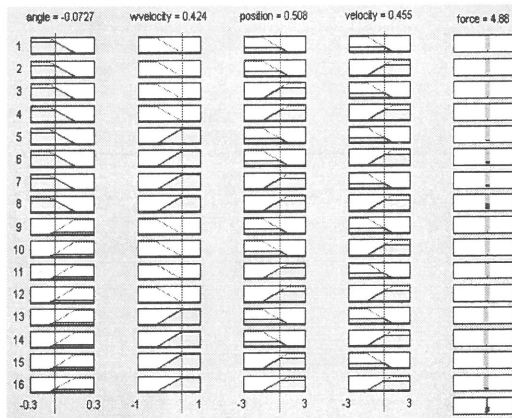


Fig. 8 Esquema operacional del Motor de Inferencia.

La función $f(u)$ posee la forma:

$$\begin{aligned}
 y_{R1} &= p_1 \cdot x_1 + q_1 \cdot x_2 + r_1 \cdot x_3 + s_1 \cdot x_4 + t_1 ; \\
 y_{R2} &= p_2 \cdot x_1 + q_2 \cdot x_2 + r_2 \cdot x_3 + s_2 \cdot x_4 + t_2 ; \\
 y_{R3} &= p_3 \cdot x_1 + q_3 \cdot x_2 + r_3 \cdot x_3 + s_3 \cdot x_4 + t_3 ; \\
 y_{R4} &= p_4 \cdot x_1 + q_4 \cdot x_2 + r_4 \cdot x_3 + s_4 \cdot x_4 + t_4 ; \\
 &\vdots \\
 y_{R16} &= p_{16} \cdot x_1 + q_{16} \cdot x_2 + r_{16} \cdot x_3 + s_{16} \cdot x_4 + t_{16} ;
 \end{aligned} \quad (6)$$

Los grados de pertenencia para cada regla son:

$$\begin{aligned}
 \mu_{R1} &\Rightarrow y_{R1} ; \\
 \mu_{R2} &\Rightarrow y_{R2} ; \\
 \mu_{R3} &\Rightarrow y_{R3} ; \\
 &\vdots \\
 \mu_{R16} &\Rightarrow y_{R16} ;
 \end{aligned} \quad (7)$$

La función de defuzzyficación se formula como:

$$u = \sum_{i=1}^{16} \mu_{Ri} \cdot y_{Ri} = \mu_{R1} \cdot y_{R1} + \dots + \mu_{R16} \cdot y_{R16} \quad (8)$$

como :

$$C = [c_{ij}] = \begin{pmatrix} p_1 & q_1 & r_1 & s_1 & t_1 \\ p_2 & q_2 & r_2 & s_2 & t_2 \\ p_3 & q_3 & r_3 & s_3 & t_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{16} & q_{16} & r_{16} & s_{16} & t_{16} \end{pmatrix} \quad (9)$$

$$\begin{aligned}
 p_i, q_i, r_i, s_i, t_i &= c_{i1}, c_{i2}, c_{i3}, c_{i4}, c_{i5} \\
 c_{ij} ; j &= 1, 5, x_5 = 1 ;
 \end{aligned} \quad (10)$$

Las características del controlador basado en

$$u = \sum_{i=1}^{16} \mu_{Ri} (p_i \cdot x_1 + q_i \cdot x_2 + r_i \cdot x_3 + s_i \cdot x_4 + t_i) \quad (11)$$

Las características del controlador basado en LD hasta ahora discutidas, pueden describirse como:

- Planta: SPI.
- Tipo de FIS: Sugeno.
- Número de Entradas/Salidas: 4/1.
- Funciones de pertenencia: 2 por entrada
- Tipo de función de salida $f(u)$: Singleton
- Función $f(u)$: Lineal de primer orden
- Número de Parámetros $f(u)$: 5
- Método AND: Producto.(norma T)
- Método OR: Máximo.(norma S)
- Número de Reglas: 16.(funciones de salida)
- Método de Implicación: Producto.
- Método de Agregación: Máximo.(norma S)
- Método de Defuzzyfication: Promedio Ponderado.

SINTONÍA ANFIS

ANFIS (Adaptive Neuro Fuzzy Inference System) es una arquitectura que es funcionalmente equivalente a una base de reglas difusas del tipo Sugeno [7]. ANFIS es un método para sintonizar una base de reglas existentes mediante un algoritmo de aprendizaje basado en la recolección de datos de entrenamiento, lo que permite a la base de reglas adaptarse a su entorno (fig. 9).

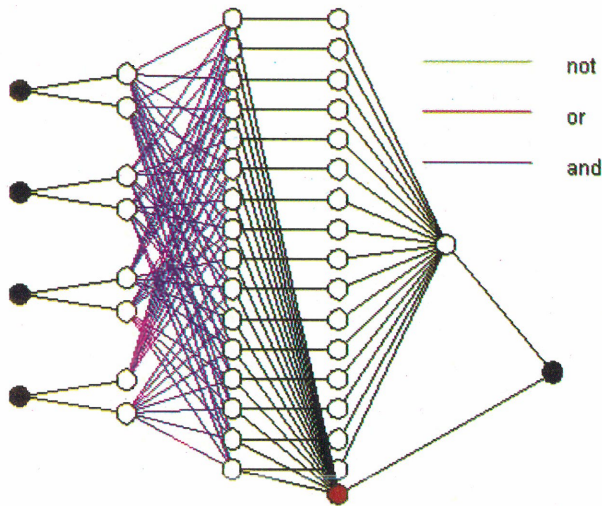


Fig. 9 Red ANFIS.

En el diseño del controlador se utilizó sintonía ANFIS para definir los parámetros de la función $f(u)$, i.e., la matriz:

$$u = \sum_{i=1}^{16} \begin{bmatrix} p_i \cdot (\mu_{Ri} \cdot x_1) \\ + q_i \cdot (\mu_{Ri} \cdot x_2) \\ + r_i \cdot (\mu_{Ri} \cdot x_3) \\ + s_i \cdot (\mu_{Ri} \cdot x_4) \\ + t_i \cdot (\mu_{Ri} \cdot x_5) \end{bmatrix} \quad (12)$$

16 5

$$u = \sum_{i=1}^{16} \sum_{j=1}^5 c_{ij} \cdot (\mu_{Ri} \cdot x_j) \quad (13)$$

Dadas las ecuaciones (9) al (13), y aplicando el método de los mínimos cuadrados sobre los datos muestreados de entrada (previamente definidos en el entrenamiento) se pueden obtener los nuevos valores (parámetros consecuentes) para la matriz C (ecuación (9)).

SOFTWARE Y SIMULACIÓN DEL SPI

El controlador diseñado se implementó en código C++. La síntesis del controlador a código C++ se obtuvo empleando dos funciones; la primera, para la fuzzyficación de funciones de pertenencia trapezoidal, y la segunda, para implementar la función de inferencia $f(u)$ de primer orden, que es una función lineal de las entradas. Además se usaron rutinas para la defuzzyficación (método del promedio ponderado), para el pre-procesamiento (escalamiento de las entradas) y para obtener las derivadas de la posición y del ángulo. Finalmente se emplearon rutinas de post-procesamiento (escalamiento de la salida). En la fig.10 se muestra el esquema de animación de las simulaciones. En las figuras 11 y 12 se muestran los resultados de la simulación del SPI para distintos set points.

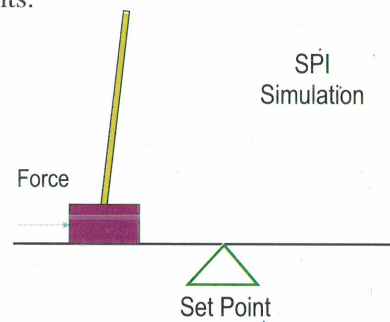


Fig.10 Esquema de Simulación SPI.

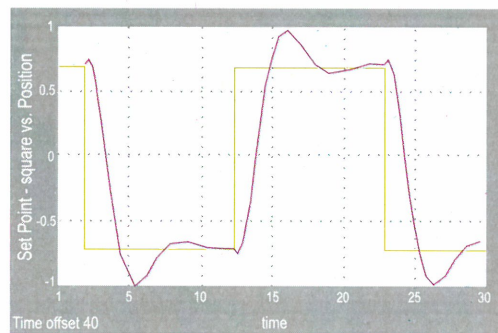


Fig.11 Simulación para el set point definido como una función de tren de pulsos.

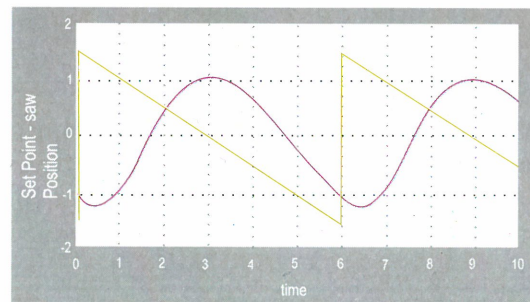


Fig.12 Simulación para el Set Point definido como una onda diente de sierra.

VALIDACIÓN EXPERIMENTAL DEL SPI

El procesamiento de datos desde y hacia la planta la realiza la interfaz HW/SW, la cual procesa y ejecuta el algoritmo de control, tal como se muestra en la fig. 13.

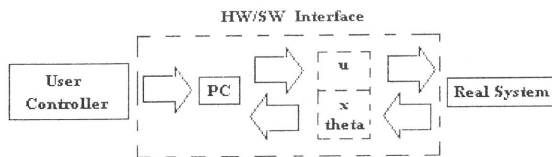


Fig. 13 Procesamiento de datos de entrada/salida.

Con la simulación se determinaron los parámetros nominales del controlador para ser probados en la planta. Los resultados experimentales se obtuvieron luego de posicionar la varilla verticalmente para hacer reaccionar el sistema según el razonamiento difuso planteado al inicio. De la experiencia adquirida, cabe anotar que el éxito en la implementación de controladores difusos se basa en gran medida en su capacidad para sintonizarse con su entorno. En nuestro caso, luego de las primeras pruebas, se afinó el modelo variando ganancias (sintonía de ganancias) tanto en las entradas de la planta como en la salida del controlador. Los resultados experimentales mostrados en las figuras 14, 15, y 16 muestran el desempeño satisfactorio del controlador difuso.

En la fig.14 vemos el efecto del limitador sobre el voltaje de control, lo que no permite generar mayores voltajes de control requeridos para el control de posición. Asimismo observamos una gran oscilación de voltaje, lo que nos demuestra que estamos frente a un sistema bastante inestable.

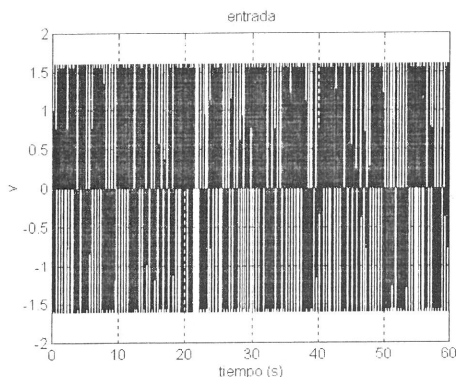


Fig. 14 Voltaje de control vs. tiempo.

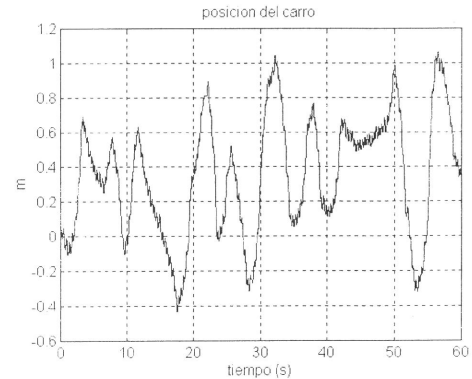


Fig.15 Posición del carro vs. tiempo.

La fig. 15 (posición del carro) se obtuvieron con un Set Point constante (posición inicial del carro).

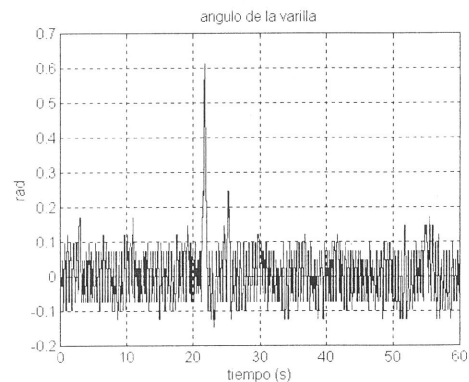


Fig.16 Ángulo de la varilla vs. tiempo.

Con respecto al ángulo de la varilla (fig.19), el error de posición fue de ± 0.1 radianes (± 6 grados sexagesimales), En dicha figura se observa un pico en donde el controlador difuso puso a prueba su capacidad de estabilización, impidiendo la eminente caída de la varilla.

CONCLUSIONES

Este artículo desarrolla la metodología de diseño e implementación del sistema de control de posición del péndulo invertido (un caso atractivo de estudio) empleando un controlador basado en inteligencia artificial.

El controlador usado es del tipo difuso TKS (Takagi-Kosko-Sugeno) que emplea la técnica de sintonía ANFIS (Adaptive Neuro-Fuzzy Inference System) para optimizar su rendimiento.

Los resultados experimentales ilustrados en las figuras 14, 15 y 16 demuestran que el objetivo de control

(mantener la varilla en posición vertical mediante una fuerza aplicada al carro) se ha cumplido satisfactoriamente.

La implementación en tiempo real del sistema diseñado demuestra que es factible cerrar la brecha que existe entre la teoría de nuevas técnicas basadas con inteligencia artificial y su aplicación práctica.

REFERENCIAS

1. César Nuñez Ocola, "Controlador de Lógica Difusa Implementado en PLC's...", Tesis de Grado, UNSA, 1996.
2. M. J. Patyra D.M. Mlynek "Fuzzy Logic: Implementation and Applications"
3. Enzo Gandolfi. Alessandro Gabrielli. Massimo Masetti. "Introduction to the Fuzzy Logic" Apr95 AIHNP Conference, Pisa.
4. Kosko, Bart, "Neural Networks and Fuzzy Systems", Englewood Cliffs, N.J.: Prentice Hall. Inc. 1992.5.
5. Jan Jantzen, "Design Of Fuzzy Controllers", Technical University of Denmark, Department of Automation, August 1998.
6. Jan Jantzen, "Neurofuzzy Modelling", Technical University of Denmark, Department of Automation, October 1998.
7. Takagi, T. and Sugeno, M. (1985). "Fuzzy identification of systems and its applications to modeling and control", 116–132. Tagkagi & Sugeno 1986.
8. Jang, Sun & Mizutani, *Neuro-fuzzy modeling and control*, 1997.
9. Jan Jantzen, "Analysis Of A Pendulum Problem", Technical University of Denmark, Department of Automation, Tech. report no 98-E 863 (cartball), 19 Aug 1998.
10. Arturo Rojas Moreno, Control Avanzado—Diseño y Aplicaciones en Tiempo Real, Publicación Independiente, Lima, 2001.