

IDENTIFICACIÓN DE SISTEMAS NO LINEALES POR MEDIO DE LÓGICA DIFUSA

Miguel Ángel Hernández López, Yu Tang.
 División de Estudio de Posgrado, Facultad de Ingeniería. U. N. A. M.
angel_universidad@hotmail.com, tang@servidor.unam.mx

Resumen: Presentamos un esquema de identificación de sistemas no lineales por medio de modelos basados en Lógica Difusa (LD). La identificación la realizamos a partir de datos experimentales que se tiene de la planta de interés, es por ello para definir los parámetros del Sistema de Lógica Difusa (SLD) hacemos uso de dos métodos de búsqueda. El primero, es el de Algoritmos Genéticos (AG's), con el cual llevamos a cabo una búsqueda global de los parámetros y definimos la estructura del modelo. El segundo, es el Método Gradiente (MG), el cual lo utilizamos para llevar a cabo un exploración local en el espacio de solución.

Palabras clave: Identificación de sistemas, aproximador universal, algoritmos genéticos, método gradiente, lógica difusa.

I INTRODUCCIÓN:

Desde hace varios años, se ha buscado incorporar nuevas técnicas al área de control automático con el propósito de facilitar el trabajo con sistemas complejos o de aquellos cuya información es incompleta. Una de las áreas que más penetración ha tenido es la lógica difusa [2], [3], [7], y los métodos de búsqueda basados en procesos biológicos (como son los algoritmos genéticos [4], [6] y la programación genética [1]). El objetivo que se persiguió en este trabajo fue el de diseñar un SLD con el cual a partir de datos experimentales (como son las medidas de las entradas y estados), se pueda obtener un modelo matemático, que permita identificar el comportamiento dinámico de la planta de interés. En especial, de aquellos sistemas no lineales que tengan la siguiente estructura:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x}) \mathbf{u} \quad (1)$$

Donde $\mathbf{x} \in \mathbb{R}^n$ es el vector de estados y $\mathbf{u} \in \mathbb{R}^m$ es el vector de entradas respectivamente, y se encuentran disponibles para su medición, con $\mathbf{f}(\mathbf{x})$ y $\mathbf{g}(\mathbf{x})$ funciones continuas de \mathbf{x} .

La forma en la cual diseñamos el SLD es la siguiente. En primer lugar, realizamos la obtención de datos experimentales de la planta (lecturas de entradas aplicadas y del comportamiento de los estados del sistema), posteriormente aplicamos los AG's y el MG, con el propósito de definir la estructura del SLD y sus parámetros.

En la siguiente sección haremos una descripción de los SLD que utilizamos. En la sección III, presentamos la forma en la cual modelamos los sistemas físicos por medio de SLD. Los AG's y el MG los explicamos en las secciones IV y V. En la

sección VI, presentamos los resultados obtenidos, y por ultimo en la sección VII, exponemos las conclusiones obtenidas.

II Lógica Difusa

Los SLD los utilizamos para realizar la identificación de sistemas no lineales, debido a que pueden aproximar una función dada dentro de un conjunto compacto con una precisión arbitraria alta. De los dos esquemas más comunes que se enejan de SLD, elegimos utilizar los modelos tipo Sugeno [3]. Su estructura está dividida en dos bloques, uno formado por Conjuntos Difusos (CD), y otro conformado por un conjunto de reglas *If - Then*.

II-1 Funciones membresía y conjuntos difusos

Una Función Membresía (FM), es una función que determina el grado de pertenencia de los elementos de un espacio a un determinado conjunto (existen diferentes tipos de funciones, como son triangulares, trapezoidales, gaussiana, etc. Optamos por solo utilizar funciones triangulares). Es así que podemos definir a los CD en términos de FM's como aquellos cuyo universo de discusión se encuentra dividido en subconjuntos, a los cuales se les asocia una FM mediante la cual se determina el grado de pertenencia de cada uno de los elementos del universo.

II-2 Reglas *If - Then*

El segundo bloque de los SLD, lo compone un conjunto de reglas *IF-THEN*, que podemos interpretar como, si (*IF*) las condiciones "*C*" se cumplen, entonces (*THEN*) la salida es "*Y*". Las reglas a su vez están divididas en dos partes, una denominada antecedente y otra consecuente. La parte antecedente, obtiene el grado de pertenencia de las variables de entrada a un determinado conjunto difuso. Mientras que la parte consecuente (que puede estar constituida de dos formas, una en la cual la salida es igual a una constante denominada singleton y otra en la que es igual a un polinomio), determina el valor que tendrá la salida, si las variables de entrada pertenecen a la parte antecedente correspondiente. De esta forma, la estructura de las reglas puede ser alguna de las siguientes:

$\text{IF } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\ \text{Then } y_r = b_0_r + b_1_r * x_1 + \dots + b_{n_r} * x_n$	$\text{IF } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\ \text{Then } y_r = b_0_r$
--	--

Donde x_i es la i -ésima variables de entrada, y A_i su conjunto difuso asociado, y_r la salida de la r -ésima regla y $b0_r, b1_r, \dots, bn_r$ son constantes.

La forma en la cual trabajan los SLD es la siguiente: Primero se evalúa la parte antecedente de cada regla, esto es, se verifica el grado de pertenencia de cada una de las variables de entrada a un determinado subconjunto de su espacio de discusión. Una vez que se tiene el grado de pertenencia de cada variable, se determina el nivel de activación (w_i) a utilizar, para lo cual existen varios métodos, uno de ellos y el que decidimos utilizar, es el producto de todos los niveles de activación que se tienen en cada regla. El segundo paso, es el de evaluar la parte consecuente de las reglas *If-Then*, y que consiste en sustituir los valores de las variables de entrada en el polinomio correspondiente. Lo anterior se hace con cada una de las reglas del modelo difuso. De esta forma, la salida del SLD estará definida por la siguiente ecuación:

$$\hat{x} = \frac{\sum_{k=1}^r \hat{x}_k w_k}{\sum_{k=1}^r w_k} \quad (2)$$

Donde r es el número de reglas que tienen el sistema.

III Modelado de sistemas dinámicos usando SLD's

El tipo de sistemas que buscamos identificar son aquellos cuyo modelo tiene la forma dada por la ecuación (1). Es así que para un sistema de "n" estados con "m" entradas, proponemos la siguiente estructura para las reglas *if-then*.

If x_1 is A_1 and ... and x_n is A_n
Then $\dot{x}_1 = a0_1^1 + a1_1^1 x_1 + \dots + an_1^1 x_n + b0_1^1 u_1 + \dots + bm_1^1 u_m$
 \vdots
And $\dot{x}_n = a0_n^1 + a1_n^1 x_1 + \dots + an_n^1 x_n + b0_n^1 u_1 + \dots + bm_n^1 u_m$

De tal forma que el valor de cada una de las funciones $f_i(x)$ y $g_{ij}(x)$ estará dado por la ecuación (2). Tomando en cuenta lo anterior, el sistema no lineal (1) puede ser representado en términos de un SLD como:

$$\begin{aligned} f(x) &= \hat{\theta}_f^T \varepsilon_f(x) + \Delta f(x) \\ g(x) &= \hat{\theta}_g^T \varepsilon_g(x) + \Delta g(x) \end{aligned} \quad (3)$$

De esta manera una aproximación de las dinámicas dadas en (1) es

$$\dot{\hat{x}} = -\alpha e + \hat{f}(x) + \hat{g}(x) u \quad (4)$$

Donde $e = x - \hat{x}$ (5)

Con $\alpha > 0$, y

$$\begin{aligned} \hat{f}(x) &= \hat{\theta}_f^T \varepsilon_f(x) \\ \hat{g}(x) &= \hat{\theta}_g^T \varepsilon_g(x) \end{aligned} \quad (5)$$

De acuerdo a la forma que tienen las reglas difusas y a (5), podemos definir las matrices $\theta_f, \theta_g, \varepsilon_f$ y ε_g de la siguiente forma

$$\begin{aligned} \hat{\theta}_g &= \begin{bmatrix} a0_r^1 & a1_r^1 & \dots & an_r^1 \\ \vdots & \vdots & \ddots & \vdots \\ a0_r^n & a1_r^n & \dots & an_r^n \end{bmatrix}; \hat{\theta}_f = \begin{bmatrix} b0_r^1 & \dots & bm_r^1 \\ \vdots & \ddots & \vdots \\ b0_r^n & \dots & bm_r^n \end{bmatrix} \quad (6) \\ \varepsilon_f^T &= \begin{bmatrix} w_1 & w_1 & x_1 & \dots & w_1 & x_n \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ w_r & w_r & x_1 & \dots & w_r & x_n \end{bmatrix}; \varepsilon_g^T = \begin{bmatrix} w_1 & u_1 & \dots & w_1 & u_m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ w_r & u_1 & \dots & w_r & u_m \end{bmatrix} \quad (7) \end{aligned}$$

De esta forma la dinámica del error de estimación estará definida como

$$\dot{e} = -\alpha e + [f(x) - \hat{f}(x)] + [g(x) - \hat{g}(x)] u \quad (8)$$

Sustituyendo las ecuaciones (3) y (5) en (8), y realizando las siguientes agrupaciones de términos:

$$\begin{aligned} \omega &= \Delta f(x) + \Delta g(x) \\ \tilde{\theta}_f &= \hat{\theta}_f^* + \hat{\theta}_f; \tilde{\theta}_g = \hat{\theta}_g^* + \hat{\theta}_g \end{aligned} \quad (9)$$

Tendremos que

$$\dot{e} = -\alpha e + \tilde{\theta}_f^T \varepsilon_f(x) + \tilde{\theta}_g^T \varepsilon_g(x) + \omega \quad (10)$$

Para demostrar como se pueden utilizar los SLD como medio de estimación, haremos uso de la teoría Lyapunov, utilizando la siguiente función cuadrada de Lyapunov.

$$V = \frac{1}{2} e e^T + \frac{1}{2\gamma} \text{Tr}(\tilde{\theta}_f \tilde{\theta}_f^T) + \frac{1}{2\gamma} \text{Tr}(\tilde{\theta}_g \tilde{\theta}_g^T) \quad (11)$$

Derivando V , y después de algunas manipulaciones algebraicas, tendremos que

$$\dot{V} = -\alpha e^T e + e^T \omega \quad (12)$$

Con lo cual

$$\dot{V} \leq -\alpha \|e\|^2 + \|e\| \|\omega\| \quad (13)$$

Definiendo ω_0 como

$$\omega_0 = \sup_{t \rightarrow \infty} \|\omega(t)\| \quad (14)$$

El cual asumimos es finito, por lo tanto

$$\dot{V} \leq -\alpha \|e\| + \omega_0 \|e\| \quad (14)$$

De esta forma para

$$\|e\| > \frac{\omega_0}{\alpha} \quad (15)$$

Tendremos que

$$\dot{V} < 0 \quad (16)$$

Mientras la condición (15) se cumpla, la señal de error “e” disminuye, por lo que el error e(t) tiende a partir de cualquier condición inicial, a un círculo de radio r, donde

$$r = \frac{\omega_o}{\alpha} \quad (17)$$

Por lo cual, podemos también concluir que la salida estimada es acotada.

IV Algoritmos genéticos

Los AG’s son un método de búsqueda y optimización, los cuales a partir una población inicial, de su evaluación, clasificación y a la aplicación de operadores genéticos (denominados cruza y mutación), se puede obtener un individuo capaz de llevar a cabo las funciones deseadas. De esta forma, decidimos utilizar los AG’s para determinar la estructura de un SLD y sus parámetros, mediante el cual realicemos la identificación de un sistema no lineal. Para ello se propuso seguir el siguiente AG

- Generar la población inicial
- Mientras i < gen_max
 - Evaluar población
 - Clasificar la población
 - Elige padre(s)
 - Aplicar operador genético
- Presentar resultados

Generar la población inicial: Una vez que se tiene los campos de las estructura dinámicas, se generan de forma aleatoria los valores reales o enteros correspondientes a cada campo.

Evaluar población: Lo que hacemos en este punto es simular todos los SLD de la población. Para ello se utilizan las mismas condiciones iniciales y las mismas entradas que se aplicaron al sistema real, una vez que el sistema es evaluado, procedemos a obtener el error de estimación cuadrático, determinado por (18) y que a su vez se convierte en nuestra función objetivo, con la cual buscamos reducir el error de estimación.

$$E = \sum_{i=1}^n \sum_{j=1}^k (x_{ij} - \hat{x}_{ij})^2 \quad (18)$$

Donde n es el número de estados y k es la cantidad de muestras que se tienen.

Clasificar la población: Una vez que cada uno de los SLD fue evaluado y obtuvimos su error de estimación, clasificamos la población de menor a mayor de acuerdo a su error. Ya clasificada la población, el mejor elemento conserva todas sus características y los demás son utilizados para generar la nueva población y después son eliminados.

Elección de padre(s): Aquí lo que hacemos es determinar uno o dos elementos de la población anterior, de tal forma que por medio de la aplicación de operadores genéticos generamos uno o dos nuevos elementos. El método de selección que

utilizamos fue el de la ruleta, en el cual los mejores elementos tienen una mayor probabilidad de ser seleccionados.

Operadores Genéticos: Son una modificación sistemática que sufren los parámetros de cada uno de los cromosomas, estos operadores son Cruza y Mutación. Además de la aplicación de los operadores, aplicamos también dos algoritmos adicionales, que tiene como propósito agregar o quitar reglas a cada SLD

A continuación explicamos cada uno de los operadores genéticos que utilizamos, para ello haremos uso de dos sistemas difusos como ejemplo. Estos sistemas tiene dos estados, una entrada, y con solo 2 funciones membresía por estado, por lo que el número de reglas posible es 4

Sistema 1

$$A_{11} = [ei_{11}, ec_{11}, ed_{11}]. \quad A_{12} = [ei_{12}, ec_{12}, ed_{12}]. \\ A_{21} = [ei_{21}, ec_{21}, ed_{21}]. \quad A_{22} = [ei_{22}, ec_{22}, ed_{22}].$$

IF x₁ is A₁₁ and x₂ is A₂₁ *THEN* xp₁₁ = a₀₁₁ + b₀₁₁*u
IF x₁ is A₁₁ and x₂ is A₂₂ *THEN* xp₁₁ = a₀₁₂ + b₀₁₂*u
IF x₁ is A₁₂ and x₂ is A₂₁ *THEN* xp₁₁ = a₀₁₃ + b₀₁₃*u
IF x₁ is A₁₂ and x₂ is A₂₂ *THEN* xp₁₁ = a₀₁₄ + b₀₁₄*u

Sistema 2

$$A_{11} = [fi_{11}, fc_{11}, fd_{11}]. \quad A_{12} = [fi_{12}, fc_{12}, fd_{12}]. \\ A_{21} = [fi_{21}, fc_{21}, fd_{21}]. \quad A_{22} = [fi_{22}, fc_{22}, fd_{22}].$$

IF x₁ is A₁₁ and x₂ is A₂₁ *THEN* xp₁₁ = c₀₁₁ + d₀₁₁*u
IF x₁ is A₁₁ and x₂ is A₂₂ *THEN* xp₁₁ = c₀₁₂ + d₀₁₂*u
IF x₁ is A₁₂ and x₂ is A₂₁ *THEN* xp₁₁ = c₀₁₃ + d₀₁₃*u
IF x₁ is A₁₂ and x₂ is A₂₂ *THEN* xp₁₁ = c₀₁₄ + d₀₁₄*u

Cruza aritmética simple: En este caso lo único que se hace es intercambiar parámetros a partir de un punto de cruza. Este punto es determinado de manera aleatoria. En el caso de los conjuntos difusos y las reglas, lo que hacemos es generar un número aleatorio dentro del rango [1, β] o [1, τ] (donde β y τ son el número de FM’s y reglas que tenemos respectivamente), con lo cual aseguramos intercambiar al menos una FM o una regla y cuando mas β-1 o τ-1. Esto dos aspectos se pueden ver en el siguiente ejemplo. A partir de los sistemas 1 y 2 mostrados, se tomaron los puntos de cruza 1, para las FM’s y 4 para la base de reglas, con lo que tendremos los siguientes nuevos elementos

Hijo 1

$$A_{11} = [ei_{11}, ec_{11}, ed_{11}]. \quad A_{12} = [fi_{12}, fc_{12}, fd_{12}]. \\ A_{21} = [ei_{21}, ec_{21}, ed_{21}]. \quad A_{22} = [fi_{22}, fc_{22}, fd_{22}].$$

IF x₁ is A₁₁ and x₂ is A₂₁ *THEN* xp₁₁ = a₀₁₁ + b₀₁₁*u
IF x₁ is A₁₁ and x₂ is A₂₂ *THEN* xp₁₁ = a₀₁₂ + b₀₁₂*u
IF x₁ is A₁₂ and x₂ is A₂₁ *THEN* xp₁₁ = a₀₁₃ + b₀₁₃*u
***IF* x₁ is A₁₂ and x₂ is A₂₂ *THEN* xp₁₁ = c₀₁₄ + d₀₁₄*u**

Hijo 2

$$A_{11} = [fi_{11}, fc_{11}, fd_{11}]. \quad A_{12} = [ei_{12}, ec_{12}, ed_{12}]. \\ A_{21} = [fi_{21}, fc_{21}, fd_{21}]. \quad A_{22} = [ei_{22}, ec_{22}, ed_{22}].$$

IF x₁ is A₁₁ and x₂ is A₂₁ *THEN* xp₁₁ = c₀₁₁ + d₀₁₁*u
IF x₁ is A₁₁ and x₂ is A₂₂ *THEN* xp₁₁ = c₀₁₂ + d₀₁₂*u
IF x₁ is A₁₂ and x₂ is A₂₁ *THEN* xp₁₁ = c₀₁₃ + d₀₁₃*u
***IF* x₁ is A₁₂ and x₂ is A₂₂ *THEN* xp₁₁ = a₀₁₄ + b₀₁₄*u**

Cruza aritmética entera: En este operador, los parámetros de los nuevos individuos son obtenidos a partir de una combinación lineal. La forma en el cual se lleva a cabo la generación de los parámetros de los nuevos individuos es la siguiente, sean p_1 y p_2 , parámetros de los padres 1 y 2 respectivamente y v un número decimal aleatorio, de tal forma que los parámetros de los hijos 1 y 2 los obtenemos de siguiente manera:

$$h_1 = v p_1 + (1-v) p_2; \quad h_2 = v p_2 + (1-v) p_1 \quad (19)$$

Esta combinación lineal, se aplica tanto a los parámetros de los conjuntos difusos, como a los coeficientes de la parte consecuente de cada una de las reglas difusas.

Cruza Heurística: El algoritmo del operador es similar al del anterior, solo que en este caso para obtener los genes de los hijos, utilizamos las siguientes ecuaciones:

$$h_1 = p_1 - v (p_2 - p_1); \quad h_2 = p_2 - v (p_1 - p_2) \quad (20)$$

Mutación uniforme: En este operador, cada FM y cada regla se ve como un solo ente, de tal forma que si sufre una mutación, cambiaremos todos los parámetros que la componen y no solo uno. Lo anterior lo mostramos en el siguiente ejemplo, a partir del sistema 1, la regla 2 y la FM A_{21} fueron mutadas, por lo tanto cambiaron todos los parámetros que la forman.

$$A_{11} = [e_{i11}, e_{c11}, e_{d11}]. \quad A_{12} = [e_{i12}, e_{c12}, e_{d12}]. \\ A_{21} = [E_{i21}, E_{c21}, E_{d21}]. \quad A_{22} = [e_{i22}, e_{c22}, e_{d22}].$$

IF x_1 is A_{11} and x_2 is A_{21} THEN $x_{p11} = a_{011} + b_{011} * u$
 IF x_1 is A_{11} and x_2 is A_{22} THEN $x_{p11} = A_{012} + B_{012} * u$
 IF x_1 is A_{12} and x_2 is A_{21} THEN $x_{p11} = a_{013} + b_{013} * u$
 IF x_1 is A_{12} and x_2 is A_{22} THEN $x_{p11} = a_{014} + b_{014} * u$

Mutación uniforme simple: Este operador es similar al anterior, solo que en este caso cada uno de los parámetros de los SLD son entes individuales, de tal manera que se va determinando uno a uno si será mutado. Para ejemplificar lo anterior explicaremos el siguiente caso. En la parte de los conjuntos difusos solo se modificó el parámetro ec_{22} por Ec_{22} de la función A_{22} . En el caso de las reglas, solo la 1 presento mutaciones (lo anterior de acuerdo al sistema 1 mostrado).

$$A_{22} = [e_{i22}, E_{c22}, e_{d22}]. \\ IF x_1 is A_{11} and x_2 is A_{21} THEN x_{p11} = A_{011} + b_{011} * u$$

Mutación escalar: Este operador, consiste en modificar todos los parámetros del SLD. En el caso de los CD's, se realiza un corrimiento de todas las de FM's. En la parte consecuente de las reglas, multiplicamos todos los coeficientes de los polinomios por un mismo factor. Para mostrar lo anterior, hacemos uso de la FM A_{11} y la regla 3 del sistema 1, y donde los parámetros "q" y "p", son determinados de forma aleatoria. Si bien en el ejemplo solo presentamos una FM y una regla, cuando se aplica el algoritmo a un SLD, se hace lo mismo a todas las FM's y a todas las reglas que lo componen.

$$A_{11} = [e_{i11}+q, e_{c11}+q, e_{d11}+q]. \\ IF x_1 is A_{12} and x_2 is A_{21} THEN x_{p11} = p * a_{013} + p * b_{013} * u$$

Mutación total: Éste, mas que un algoritmo, lo que se hace es generar un nuevo SLD cambiando todos los parámetros, tanto de las FM de los conjuntos difusos, como los coeficientes de la parte consecuente de las reglas.

Aumento y disminución de reglas

Cada una de las reglas que se tiene, cuenta con un parámetro adicional denominado bandera, y cuyo valor es 1 o 0. La forma de manejar este parámetro es la siguiente:

Si la bandera es 1: $x_{p1}^2 = a_{01}^2 + a_{11}^2 * x_1 + a_{21}^2 * x_2 + b_{01}^2 * u$

Si la bandera es 0: $x_{p1}^2 = 0$

Para agregar o disminuir reglas, lo único que se hace es cambiar la bandera de uno a cero y de cero a uno.

V Método Gradiente

Lo que buscamos hacer con el MG, es ajustar los coeficientes del polinomio de la parte consecuente de las reglas del SLD que se obtiene a partir del AG. De esta forma y tomando en cuenta la estructura que tiene cada una de las reglas, definimos la siguiente función

$$g_r(x, k) = a_0^i + a_1^i x_1 + \dots + a_n^i x_n + b_1^i u_1 + \dots + b_m^i u_m \quad (21)$$

Con lo que la salida del SLD estará dada por

$$\hat{x}_i = \frac{\sum_{j=1}^r g_j(x, k) \mu_j(x, k)}{\sum_{j=1}^r \mu_j(x, k)} \quad (22)$$

Donde $\mu_j(x, k) = \min. [A_1(x_1) \dots A_n(x_n)] \quad (23)$

De tal forma que la ley de adaptación de los parámetros θ_{ij} (donde θ_{ij} representa los parámetros de los vectores θ_f o θ_g) es

Si $\|\theta\|_2 \leq M$, donde M es una constante positiva, entonces

$$\theta_{ij}(k+1) = \theta_{ij}(k) - \lambda \frac{\partial e_i}{\partial \theta_{ij}} \quad (24)$$

Cuando $\lambda > 0$ y $e_i = x_i - \hat{x}_i \quad (25)$

Por lo cual

$$\frac{\partial e_i}{\partial \theta_{ij}} = -e \frac{\partial \hat{x}_i(x|\theta(k))}{\partial g_1(x, k)} \frac{\partial g_1(x, k)}{\partial \theta_{ij}} \quad (26)$$

Con

$$\frac{\partial \hat{x}_i}{\partial g_1(x, k)} = \frac{\mu_i(x, k)}{\sum_{i=1}^r \mu_i(x, k)} \quad (27)$$

$$\frac{\partial g_1(x, \theta(k))}{\partial \theta_{i0}} = 1 \quad (28)$$

$$\frac{\partial g_1(x, \theta(k))}{\partial a_j^i} = x_j \quad (29)$$

$$\frac{\partial g_1(x, \theta(k))}{\partial b_j^i} = u_j \quad (30)$$

Si $\|\theta\|_2 > M$, es necesario obtener la proyección de los parámetros, por lo que la ley de adaptación estará definida como:

$$\theta_{ij}(k+1) = \theta_{ij}(k) - \lambda \frac{\partial e_i}{\partial \theta_{ij}} + p(e, \theta, \varepsilon) \frac{\partial g_1(x, \theta(k))}{\partial \theta_{ij}} \quad (31)$$

Con

$$p(e, \theta, \varepsilon) = \frac{e_i \theta_i \varepsilon(x)}{\|\theta_i\|_2} \quad (32)$$

VI RESULTADOS

El esquema de identificación por medio de SLD desarrollado, fue aplicado a datos experimentales obtenidos del sistema de dos tanques mostrado en la Fig. 1. El experimento consistió en la aplicación de una señal binaria pseudo aleatoria, con la cual los estados tuvieron el comportamiento mostrado en la Fig. 2.

Los datos de la entrada y del comportamiento de los estados, fueron utilizados para realizar el entrenamiento de un SLD, por medio del cual se llevará a cabo la identificación del sistema.

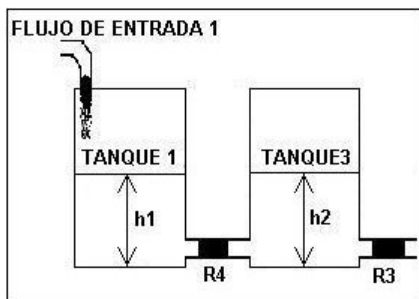


Fig. 1 Sistema de dos tanques, utilizados en el segundo experimento

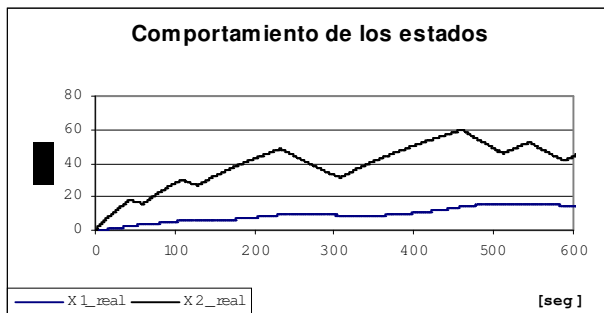


Fig. 2 Respuesta de los estados del sistema.

La gráfica del error de estimación del mejor individuo de cada una de las generaciones del AG, es mostrado en la Fig. 3. Mientras que la comparación entre los datos de entrenamiento y los obtenidos del SLD resultado del AG los mostramos en la Fig. 4. En la cual como podemos ver la dinámica del sistema ya se sigue, presentándose únicamente un error en la magnitud de la estimación.

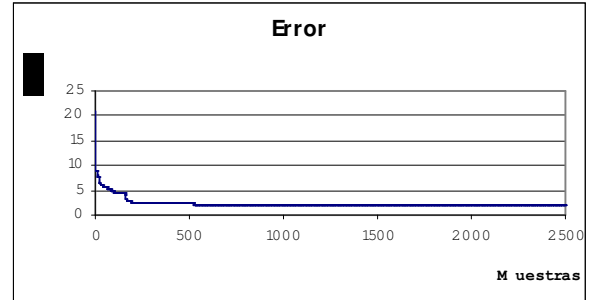


Fig. 3 Comportamiento del error de estimación durante la aplicación del AG.

Al SLD resultado del AG, le aplicamos el MG con el propósito de mejorar su desempeño, obteniendo el comportamiento mostrado en la Fig. 5.

En la Fig. 6, apreciamos el comportamiento de la norma del vector de parámetros del SLD durante la aplicación de la ley de adaptación del MG, en donde como podemos apreciar la adaptación fue constante. En los casos en los cuales la magnitud de la norma del vector de parámetros fue mayor a la de la cota establecida, por medio de la aplicación de la proyección de nueva cuenta regresamos los parámetros del SLD al espacio óptimo.

A continuación, presentamos la salida de archivo que contiene información de la base de reglas y las funciones membresía de los conjuntos difusos del SLD, por medio del cual se lleva a cabo la identificación de la planta.

Para el estado 1 se tienen las siguientes funciones membresía:

- A1.- Tipo: Triangular Parámetros [-0.991253, 19.276857, 32.222168]
- A2.- Tipo: Triangular Parámetros [-11.504105, 25.428959, 54.356892]
- A3.- Tipo: Triangular Parámetros [15.318600, 56.412201, 81.359337]

Para el estado 2 se tienen las siguientes funciones membresía:

- A1.- Tipo: Triangular Parámetros [-0.428058, 2.198618, 5.959681]
- A2.- Tipo: Triangular Parámetros [-3.839014, 7.768995, 13.278111]
- A3.- Tipo: Triangular Parámetros [0.214215, 4.840921, 18.278776]

La base de reglas para el F1 es:

- If X1 is A1 & X2 is A1 Then Y= 0.0087+0.0005*X1-0.0002*X2+0.0003*U1
- If X1 is A2 & X2 is A1 Then Y= 0.0013+0.0001*X1+0.0004*X2+0.0021*U1
- If X1 is A3 & X2 is A1 Then Y= 0.0032+0.0062*X1+0.0027*X2 -0.0044*U1
- If X1 is A2 & X2 is A2 Then Y= -0.0003 +0.0009*X1-0.0071*X2+0.0084*U1
- If X1 is A3 & X2 is A2 Then Y= -0.0001-0.0000*X1-0.0005*X2-0.0014*U1
- If X1 is A1 & X2 is A3 Then Y= 0.0003+0.0000*X1+0.0000*X2+ 0.0003*U1
- If X1 is A2 & X2 is A3 Then Y= -0.0055+0.0000*X1-0.0000*X2+0.0000*U1
- If X1 is A3 & X2 is A3 Then Y= -0.0000-0.0061*X1+0.0009*X2+0.0001*U1

La base de reglas para el F2 es:

- If X1 is A1 & X2 is A1 Then Y= 0.0061+0.0000*X1+0.0003*X2+ 0.0008*U1
- If X1 is A2 & X2 is A2 Then Y= 0.0062+0.0000*X1-0.0000*X2+ 0.0000*U1
- If X1 is A2 & X2 is A3 Then Y= -0.0061+0.0003*X1-0.0005*X2+ 0.0001*U1
- If X1 is A3 & X2 is A3 Then Y= 0.0039-0.0000*X1-0.0000*X2+0.0004*U1

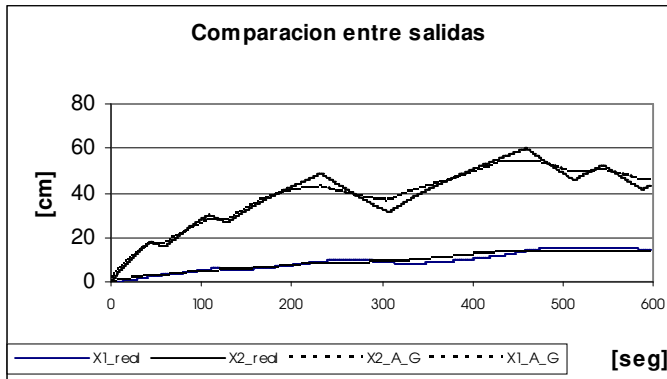


Fig. 4 Comparación entre los estados reales y los estados estimados, después de aplicar el AG.

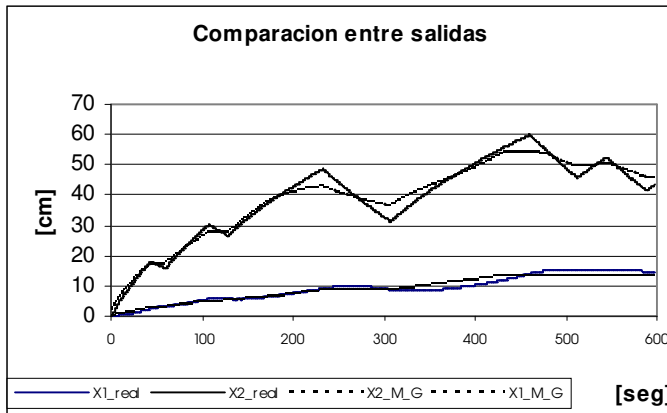


Fig. 5 Salida del SLD después de aplicar el MG al sistema obtenido por el AG.

VII CONCLUSIONES

A lo largo del desarrollo del trabajo se tuvieron diferentes observaciones, las cuales permitieron llegar a las siguientes conclusiones. En primer lugar, al generar la población inicial no es conveniente tener todas las reglas del SLD activas, sino solo activar a lo más un 25% de estas y durante la ejecución del AG agregar o quitar reglas. Un segundo aspecto es el hecho de que las FM's se puedan ubicar libremente en el rango de la variable, lo cual generará una mejor reproducción de la dinámica. Un tercer aspecto que se pudo observar fue el de que conforme se agregó una mayor variedad de posibles modificaciones a los parámetros, también se incrementó la velocidad de convergencia. Una situación que presento buenos resultados, fue el utilizar tasas de cruce y mutación que estuvieran en función del error que se tenía en cada una de las generaciones, en especial de la tasa de mutación la cual se encontró dentro del rango [0.01, 0.1].

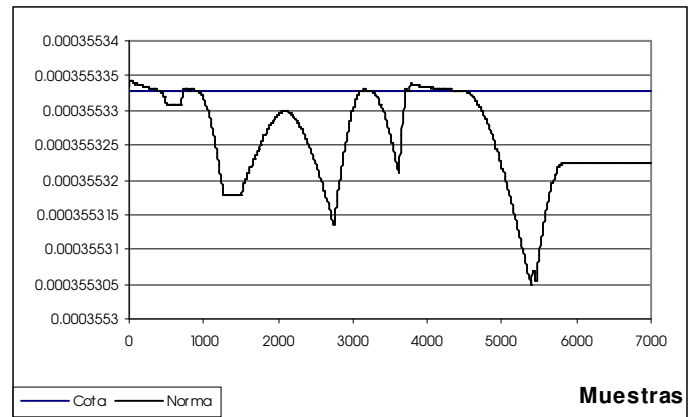


Fig. 5.16 Aplicación de la ley de adaptación del MG al vector Θ

REFERENCIAS

- [1] J. Gary, J. David, Murray-Smith. "Nonlinear model structure identification using genetic programming", Control Engineering Practice (1998).
- [2] Chuen Chien Lee "Fuzzy logic in control systems: Fuzzy logic controller – part I and II", IEEE transactions on systems, man and cybernetics, vol. 20, No 2 March/April 1990.
- [3] Tomohiro Takagi, Michio Sugeno, "Fuzzy identification of systems and its applications to modeling and control", IEEE transactions on systems, man and cybernetics, vol SMC-15, No 1 January / February 1985.
- [4] Magne Setnes and Hans Roubos, "GA fuzzy modeling and classification: Complexity and performance", IEEE transaction on fuzzy systems, vol. 8, No 5, October 2000.
- [5] Li xi Wang, "Adaptive fuzzy systems and control", Prentice Hall, 1994.
- [6] David e. Goldberg "Genetics algorithms in search, optimization, and machine learning", Addison Wesley, 1989.
- [7] Kevin M. Passino, Stephen Yurkovich, "Fuzzy control", Addison Wesley, 1998.