

01170



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

PROGRAMA DE MAESTRIA Y DOCTORADO EN INGENIERIA

IDENTIFICACION DE SISTEMAS NO LINEALES POR MEDIO
DE LOGICA DIFUSA.

T E S I S
QUE PARA OBTENER EL GRADO DE
MAESTRO EN INGENIERIA
(ELECTRICA)
P R E S E N T A
HERNANDEZ LOPEZ MIGUEL ANGEL

DIRECTOR DE TESIS: DR. YU TANG XU

MEXICO, D. F.

2004



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ESTA TESIS NO SALE
DE LA BIBLIOTECA

Agradecimientos

A la Universidad Nacional Autónoma de México, por brindarme la oportunidad de poder continuar con mi desarrollo profesional.

A los profesores del área de control automático, por los conocimientos y consejos que me dieron durante mis estudios, pero también, por la amistad que me brindan. En especial a los doctores Leonid Fridman, Héctor Benítez, Jaime Moreno, Gerardo Espinosa, Marco Arteaga.

Al Dr. Yu Tang, por los conocimientos y asesoría que me dió, así como por sus consejos, apoyo y sobre todo por la amistad que me otorga y que espero conservar.

Al Dr. Jesús Savage Carmona, quien ha sido una persona clave en mi desarrollo profesional y académico.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el apoyo otorgado para la realización de mis estudios de maestría.

Dedicatoria

A mis padres Jesús Hernández Villanueva y María Guadalupe López y mi hermana Beatriz Hernández López, por el apoyo incondicional que siempre me dieron, pero sobre todo por su cariño y comprensión.

Dedicatoria

A Natalia Canul Meléndez y Gonzalo N. Hernández Canul, personas fundamentales en mi vida a quienes agradezco el cariño y amor que me brindan, la comprensión y la paciencia que me tienen, así como la energía e ilusión que con ello me han entregado.

ÍNDICE

CAPITULO I	
INTRODUCCIÓN	1
1.1 Descripción de la tesis	1
1.2 Trabajos previos	3
1.3 Desarrollo	4
1.4 Organización	7
CAPITULO II	
SISTEMAS DE LÓGICA DIFUSA	8
2.1 Conceptos básicos	8
2.1.1 Sistemas de lógica difusa	8
2.1.2 Funciones membresía	9
2.1.3 Conjuntos difusos	11
2.1.4 Reglas <i>If-Then</i>	14
2.2 SLD como un método de aproximación universal de funciones	15
2.3 Ejemplo	20
CAPITULO III	
ALGORITMOS DE BUSQUEDA Y OPTIMIZACIÓN	24
3.1 Algoritmos genéticos	24
3.1.1 Problema de optimización	26
3.1.2 Representación	27
3.1.3 Decodificación del cromosoma	28
3.1.4 Inicialización de la Población	29

3.1.5 Estructura del AG	29
3.1.6 Criterios de paro	32
3.2 Método gradiente	33
CAPITULO IV	
MODELADO DE SISTEMAS NO LIENEALES POR SLD's	40
4.1 Clase de sistemas no lineales	40
4.2 Aproximación de las dinámicas no lineales por SLD's	43
4.3 Búsqueda de SLD's por medio de AG's	48
4.3.1 Algoritmo genético	51
4.3.2 Operadores genéticos	56
4.4 Búsqueda de parámetros óptimos por algoritmo gradiente	65
CAPITULO V	69
RESULTADOS	
5.1 Sistema de dos tanques (datos simulados)	70
5.2 Sistema de dos tanques (datos experimentales)	74
5.3 Discusión de resultados	80
CAPITULO VI	
CONCLUSIONES	84
Anexo A	
Manual del programa	87
A.1 Codificación	87
A.2 Uso del programa	90

Anexo B	
Notación y abreviaciones	93
Bibliografía	94

CAPITULO I

INTRODUCCIÓN

1.1 Descripción de la tesis

Dentro del análisis de sistemas no lineales, un aspecto fundamental y clave, es el de poder contar con un modelo matemático de la planta, que describa su dinámica, con el propósito de poder realizar el diseño de controladores y observadores, realizar la detección de fallas, y así como poder llevar a cabo simulación de los esquemas diseñados. Lo anterior es principalmente por cuestiones de seguridad, debido a que antes de poder implementar un controlador, éste debe ser probado, con el propósito de que el sistema físico no sufra daños.

Para poder llevar a cabo la obtención del modelo matemático, existen varias formas, uno es por medio de la aplicación de leyes físicas, y dentro de los cuales se encuentran los métodos de redes, ecuaciones de Euler - Lagrange, etc. Una segunda forma de obtener el modelo matemático, es por medios estadísticos, los cuales a partir de la obtención de datos experimentales, se busca conocer el comportamiento dinámico del sistema a analizar.

Sin embargo, y pese a que los métodos anteriores tienen ya muchos años de existencia y cuentan con una base matemática fuerte, no siempre es posible obtener el modelo deseado. Esta situación se presenta debido a diferentes causas, entre las cuales se encuentran:

- El desconocimiento de la estructura exacta del sistema, ya sea por su complejidad, o bien por la falta de información del mismo.
- La existencia de fenómenos no lineales (fricción, zonas muertas, saturación, etc.).
- El desconocimiento de los valores exactos de los parámetros que componen el modelo.

Ante esta situación, se hace necesario recurrir a las alternativas con que se cuenta, como es la de hacer uso de modelos de aproximación universal (como pueden ser los modelos basados en Redes Neuronales, Sistemas de Lógica Difusa (SLD), series de Taylor, etc.), los parámetros de los aproximadores presentan la característica de que a partir del conocimiento de datos experimentales de la planta, se pueden ser diseñados y sintonizados. Estos tienen la ventaja de convertir la incertidumbre, tanto paramétrica como no paramétrica en el sistema no lineal en solo incertidumbre paramétrica, mientras las funciones base del aproximador universal no cambien. A una función se le llama aproximador universal si ésta puede aproximar a una función dada dentro de un conjunto compacto con una precisión arbitrariamente alta [5], [7].

Dentro de este contexto se encuentra el trabajo realizado, y que tuvo como objetivo obtener un esquema por medio del cual se puedan encontrar los de un SLD que modele la dinámica de un sistema no lineal. Lo anterior se realiza 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:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x}) \mathbf{u} \quad (1.1)$$

Donde $\mathbf{x} \in \mathbb{R}^n$ es el vector de estados, $\mathbf{u} \in \mathbb{R}^m$ es el vector de entradas del sistema, con $\mathbf{f}(\mathbf{x})$ y $\mathbf{g}(\mathbf{x})$ funciones continuas de \mathbf{x} . Suponemos además que el vector de estados, se encuentra disponible para su medición. Para llevar a cabo la identificación de los sistemas no lineales, hacemos uso de SLD's.

Con base en lo anterior, desarrollamos un esquema para llevar a cabo la identificación de un sistema no lineal, por medio de un modelo basado en SLD. Los parámetros del SLD son obtenidos por medio del uso de dos métodos de búsqueda. El primero, el de Algoritmos Genéticos (AG's), que permite realizar una exploración global, y el segundo, el Método Gradiente (MG), con el cual se realiza una exploración local.

1.2 Trabajos previos

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. Una de las áreas que más penetración han tenido, es la asociada a la inteligencia artificial (como son la lógica difusa, los algoritmos genéticos o las redes neuronales). Estas técnicas se han incorporado al diseño de controladores, estimadores de estados, identificación de sistemas, detección de fallas, etc. Es así que en la actualidad podemos encontrar diferentes trabajos, como son los realizados por Takagi y Sugeno (1995), sobre el diseño de leyes control y el modelado e identificación de sistemas dinámicos por medio

de sistemas de lógica difusa, o los llevados a cabo por Wang (1994), quien presenta diferentes técnicas por medio de las cuales se pueden determinar los parámetros de un modelo difuso tanto para realizar leyes de control como para el modelado de sistemas físicos, principalmente por medio de leyes es de adaptación. En el campo de la programación genética, se encuentran los trabajos de Gary, Murria y Smith (1998), o los de Setnes y Roubos (2000), quienes presentan diferentes esquemas por medio de los cuales se busca modelar sistemas físicos, haciendo uso de algoritmos evolutivos. En el área de sistemas evolutivos se encuentran los trabajos llevados a cabo por Goldberg (1989), quien hace una descripción de los algoritmos genéticos, sus bases y diferentes formas de aplicación de los mismos.

1.3 Desarrollo

Como acabamos de mencionar, buscamos obtener un esquema por medio del cual podamos llevar a cabo el modelado del comportamiento dinámico de un sistema físico no lineal. El tipo de sistemas que buscamos modelar, es aquel en el cual existe una carencia de información sobre los elementos que lo componen, convirtiéndose en una caja negra (figura 1.1), de la cual lo único que se sabe es que al aplicar ciertas entradas, se presentan ciertas salidas, además de lo anterior, se supone o se sabe que el modelo matemático real, presenta una estructura similar a la ecuación (1). En nuestro caso las salidas de la caja negra son los estados del sistema, los cuales se supone son pueden ser medidos. Debido a esta carencia de información, es imposible poder llevar a cabo un modelado clásico, es por ello que para lleva a cabo el modelado del sistema,

recurrimos a medios alternativos, más específicamente a Sistemas de Lógica Difusa (SLD).

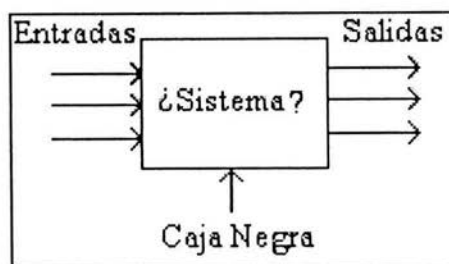


Fig. 1.1 sistemas cuya estructura es desconocida

Con base en lo anterior, convertimos un problema con incertidumbre paramétrica y no paramétrica, en uno con únicamente incertidumbre paramétrica. Lo anterior se debe a que ahora solo es necesario determinar los parámetros que componen al SLD por medio del cual llevaremos a cabo la identificación del sistema dinámico.

Para llevar a cabo la identificación de los parámetros del modelo matemático difuso, hacemos uso de dos métodos de búsqueda y optimización. El primer método con el cual comenzamos a realizar la búsqueda es el de Algoritmos Genéticos (AG's), el cual nos permite realizar una búsqueda global dentro del espacio de las diferentes soluciones, de tal forma que al concluir la ejecución del AG, contemos con un SLD que brinde una aproximación inicial de la dinámica del sistema físico a identificar. Una vez que el AG ha finaliza o su ejecución, aplicamos el segundo método, el cual esta basado en la información obtenida a partir del gradiente de una función a optimizar, por lo cual lo designamos como Método Gradiente (MG). El MG, nos permite llevar a cabo

la búsqueda de la solución óptima, que es el centro de una bola, dentro de la cual se llevara a cabo la búsqueda.

El proceso anterior lo podemos resumir en la figura 1.2. En ella se presenta al conjunto A, como el conjunto de todos los SLD, de los cuales algunos pueden llevar a cabo la representación de la dinámica de un sistema físico F, en especial los que se encuentran dentro de la bola B, cuyo centro es el modelo M_0 . Donde M_0 es el SLD óptimo por medio del cual podemos modelar al sistema F. De esta forma, ahora el problema es encontrar los parámetros de M_0 . Para lograr obtener los parámetros de M_0 , empleamos dos métodos. En primer lugar el de AG's, el cual a partir de un conjunto inicial C de SLD, va explorando a través de A. Una vez que el AG ha terminado su ejecución, definimos a partir del error de identificación del SLD obtenido, el radio α , con el cual definiremos el conjunto B. Una vez definido B, por medio de la aplicación MG, buscaremos llegar a M_0 , sintonizando los parámetros del SLD y obtener un modelo que represente de forma más precisa la dinámica del sistema F.

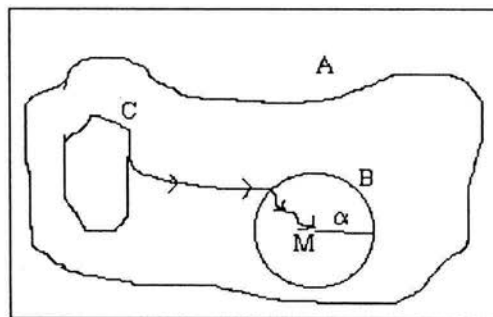


Fig. 1.2 Representación del proceso de identificación de un sistema físico por medio de lógica difusa

De esta forma, los que buscamos en primer lugar, mostrar como los AG's pueden ser utilizados para llevar a cabo el modelado de sistemas físicos no lineales. En segundo lugar es mostrar una forma en la cual los AG's y los métodos basados en el gradiente de una función a optimizar, pueden ser perfectamente combinados para llevar a cabo la identificación de parámetros, si se les ve como métodos complementarios.

1.4 Organización

En los siguientes capítulos, haremos una breve descripción de lo que son los SLD's (capítulo 2), los AG's y MG, (capítulo 3). En el capítulo 4, describimos el sistema diseñado, de la forma que hacemos la identificación de los sistemas no-lineales por medio SLD's, y la descripción de los métodos de búsqueda usados. En el capítulo 5, mostramos los resultados obtenidos tras llevar a cabo la aplicación del esquema diseñado, en la identificación de un sistema hidráulico. En el capítulo 6, presentamos las conclusiones obtenidas a lo largo del trabajo realizado. Por último agregamos dos apéndices, en el primero, explicamos el programa realizado, así como la forma de utilizarlo. En el segundo apéndice, explicamos la notación y las abreviaturas utilizadas.

CAPITULO II

SISTEMAS DE LÓGICA DIFUSA

Como se sabe, existen diferentes métodos para aproximar una función, como son las redes neuronales, polinomios y sistemas de lógica difusa. En este trabajo se hace uso de los Sistemas de Lógica Difusa (SLD's) para poder llevar a cabo esta tarea. Más específicamente, utilizamos los sistemas de tipo Sugeno. Es por ello que en este capítulo, se hace una descripción breve de los SLD's.

2.1 Conceptos básicos

En las siguientes subsecciones, daremos una descripción de lo que es la lógica difusa y de los elementos que la componen.

2.1.1 Sistemas de lógica difusa

En contraste con la lógica convencional, que utiliza conceptos absolutos para referirse a la realidad, la Lógica Difusa (LD) se define en grados de pertenencia entre 0 y 1, siguiendo patrones de razonamiento similares a los del pensamiento humano. Así por ejemplo, mientras dentro del marco rígido de la lógica convencional un recinto está solamente "oscuro" (0) o "claro" (1), para la lógica difusa son posibles también todas las condiciones relativas intermedias percibidas por la experiencia humana como "muy claro", "algo oscuro", "ligeramente claro", "extremadamente oscuro", etc. La lógica difusa por lo tanto,

admite la pertenencia parcial de un elemento a un determinado conjunto y permite las transiciones graduales entre ser plenamente miembro de un conjunto o no serlo en absoluto. Si un elemento es parcialmente miembro de un conjunto, al mismo tiempo, también es parcialmente no miembro del mismo conjunto. Dado que la lógica difusa permite la pertenencia y la no pertenencia de un elemento a un conjunto (aunque no simultáneamente), puede considerarse como una generalización de la teoría clásica de los conjuntos.

2.1.2 Funciones membresía

Análogo al proceso de razonamiento del ser humano, y para poder trabajar de una manera más sencilla con sistemas complejos, la LD utiliza variables lingüísticas. Una descripción de este tipo, requiere la utilización de términos imprecisos, tales como “rápido” o “lento”. La variable lingüística “velocidad” podría tener valores como “muy lenta”, “lenta”, “rápida”, etc. Una función membresía representa a cada uno de los conjuntos de LD y actúa como elemento de conversión entre el mundo real, visto según la lógica convencional, y el visto según la LD. Por ejemplo:

- MUY LENTA (110 km/h)= 0
- LENTA (110 km/h)= 0
- SEMI-RAPIDA (110 km/h)=0.3
- RAPIDA (110 km/h)=0.8
- MUY RAPIDA (110 km/h)=0.4

En términos generales, una Función Membresía (FM) (*Membership Function*), es una función que determina el grado de pertenencia de los elementos de un conjunto. Se

denota generalmente por μ y puede adoptar valores entre 0 y 1. Las funciones membresía que se suelen utilizar son las que se muestran en la figura 2.1

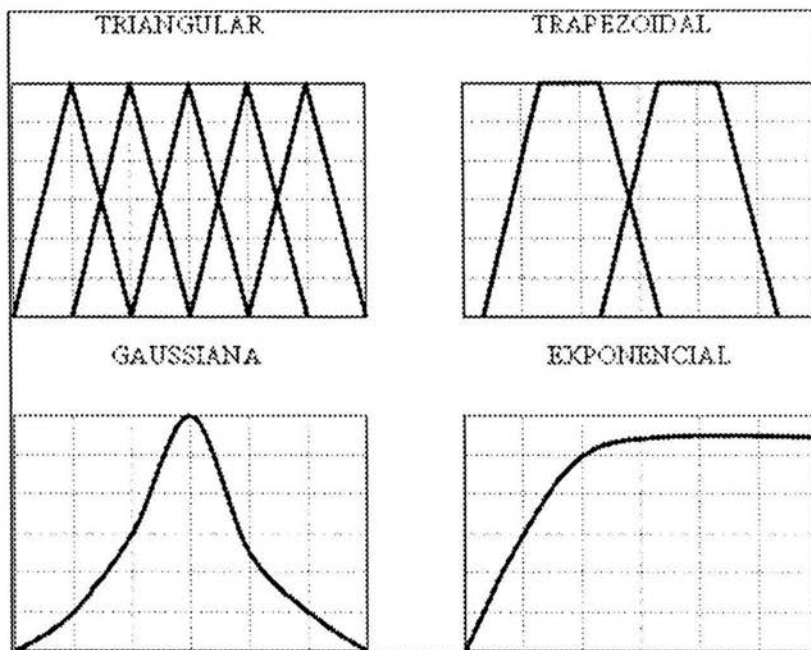


FIG. 2.1 Funciones membresía más comunes

Se puede observar que las FM's de la figura 2.1 se superponen, por lo que un individuo x podría tener distintos grados de pertenencia en dos conjuntos al mismo tiempo. Esto indica que posee cualidades asociadas con ambos conjuntos.

En el presente trabajo, optamos por únicamente utilizar funciones membresía de tipo triangular, como la que se muestra en la figura 2.2, en donde podemos ver los tres parámetros que componen las FM's, de tal forma que se tiene tres puntos ($[X_i, 1]$, $[X_c, 1]$, $[X_d, 1]$), a partir de los

cuales se definen las dos rectas que conforman la función triangular, cuyas ecuaciones están definidas por la ecuación (2.1). Debido a esta composición de dos rectas de las FM's, es que las elegimos, debido a que al no tener que ser simétricas como sucede en otros casos, permite un mayor manejo del espacio de discusión de las variables de entrada sin necesidad de utilizar un gran número de funciones.

$$\mu_B(x) = \begin{cases} \frac{x-x_C}{x_D-x_C} & \text{si } x_I \leq x < x_C \\ \frac{x_D-x}{x_D-x_C} & \text{si } x_C \leq x < x_D \\ 0 & \text{Cualquier otro punto} \end{cases} \quad (2.1)$$

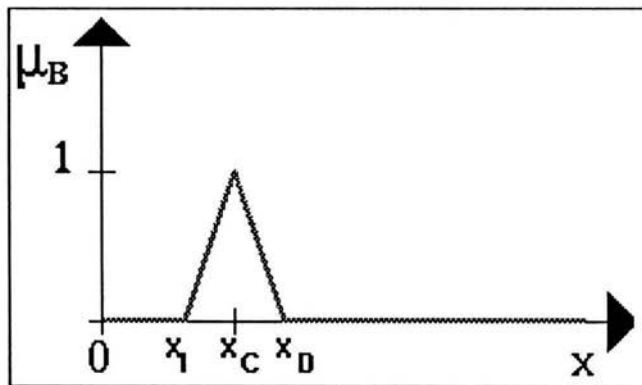


FIG. 2.2 Representación gráfica de la FM utilizadas

2.1.3 Conjuntos difusos

Un conjunto difuso es aquel que puede contener elementos con grados de pertenencia parciales, a diferencia de los conjuntos clásicos, en los cuales los elementos pueden

solamente "Pertener" ó "No pertenecer" a dichos conjuntos.

Por ejemplo, sea el conjunto B, definido como:

$$B = \{\text{conjunto de edades de la gente joven}\}.$$

Un intento para construir este conjunto de forma clásica es definir un intervalo en años de la siguiente manera:

$$B = \{(x, \mu_B(x))\}$$
$$\mu_B(x) = \begin{cases} 1 & \text{Si } 0 \leq x \leq 20 \\ 0 & \text{Si } x > 20 \end{cases}$$

Lo anterior implicaría que una persona sería joven hasta el día de su cumpleaños número 20, pero al siguiente día ya no lo sería. Ahora, si cambiamos el límite superior del intervalo, el problema persistiría. Una forma más natural de construir el conjunto B, es eliminando esa estricta separación entre ser joven y no serlo, admitiendo grados de pertenencia intermedios entre 0 y 1. Por lo tanto, el conjunto B será un conjunto difuso.

La función de pertenencia que podría describir el conjunto B, sería la que se muestra en la figura 2.3 y cuya ecuación es la (2.2).

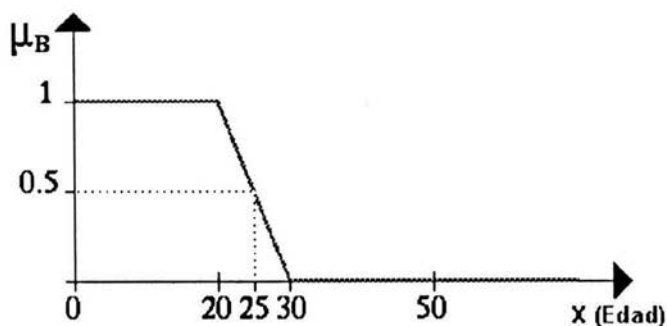


FIG. 2.3 representación difusa para la determinación de si una persona es joven

$$B = \left((x, \mu_B(x)), \begin{cases} \mu_B(x) = 1 & 0 \leq x \leq 20 \\ \mu_B(x) = -0.1x + 3 & 20 < x < 30 \\ \mu_B(x) = 0 & x \geq 30 \end{cases} \right) \quad (2.2)$$

De esta manera una persona de 25 años es todavía joven pero con un grado del 50%. El grado de pertenencia de x en B , se representa por $\mu_B(x)$. De tal forma que haciendo uso de FM y notación matemática, un conjunto difuso B lo podemos definir como:

$$B = \{(x, \mu_B(x)) \mid x \in X\}$$

$$\mu_B: X \rightarrow [0,1]$$

El conjunto difuso B es la unión de los grados de pertenencia para todos los puntos en el universo de discusión X .

Podemos definir un conjunto difuso, como aquel 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.

2.1.4 Reglas *If – Then*

La estructura más utilizada para los SLD's, está constituida básicamente por dos bloques. El primero, es el de conjuntos difusos, en donde se realiza la transformación de los valores numéricos en valores lingüísticos. El segundo bloque lo compone un conjunto de reglas *IF-THEN*, y 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 es por medio del uso de conjuntos difusos, y la segunda por el uso de polinomios), determina el valor que tendrá la salida, si las variables de entrada pertenecen a la parte antecedente correspondiente. Un ejemplo de esta estructura es el que mostramos a continuación:

Antecedente	Consecuente
<i>IF</i> servicio <i>IS</i> Excelente <i>AND</i> comida <i>IS</i> Excelente	<i>THEN</i> propina es buena
<i>IF</i> servicio <i>IS</i> Excelente <i>AND</i> comida <i>IS</i> Buena	<i>THEN</i> propina es regular
<i>IF</i> servicio <i>IS</i> Bueno <i>AND</i> comida <i>IS</i> Excelente	<i>THEN</i> propina es regular
<i>IF</i> servicio <i>IS</i> Bueno <i>AND</i> comida <i>IS</i> buena	<i>THEN</i> propina es mala

En el ejemplo anterior, la parte consecuente está constituida por conjuntos difusos. Si utilizamos polinomios, en primer lugar, es necesario que las variables de entrada sean numéricas, como podrían ser la temperatura, la velocidad o la presión, para así poderlas evaluar en el polinomio. Por ejemplo, en la siguiente tabla, presentamos un caso, en donde las variables de entrada, son la temperatura (T) y la humedad (H) de una habitación y como salida la velocidad de giro de las aspas de un ventilador con el cual se controla la temperatura.

Antecedente	Consecuente
<i>IF</i> Temperatura <i>IS</i> Alta <i>AND</i> Humedad <i>IS</i> Alta	<i>THEN</i> $V = 5 + 6T + 7H$
<i>IF</i> Temperatura <i>IS</i> Alta <i>AND</i> Humedad <i>IS</i> Baja	<i>THEN</i> $V = 3 + 5T + 5H$
<i>IF</i> Temperatura <i>IS</i> Baja <i>AND</i> Humedad <i>IS</i> Alta	<i>THEN</i> $V = 2 + 4T + 4H$
<i>IF</i> Temperatura <i>IS</i> Baja <i>AND</i> Humedad <i>IS</i> Baja	<i>THEN</i> $V = 1 + 2T + 1H$

2.2 SLD como un método de aproximación universal de funciones

Como ya se mencionó, los SLD son considerados aproximadores universales, debido a que pueden aproximar una función dada, en un conjunto compacto (cerrado y acotado) con una precisión arbitrariamente alta. En la actualidad existen dos tipos de SLD, los tipo Mamdani y tipo Sugeno. En nuestro caso, optamos por utilizar únicamente modelos del tipo Sugeno, debido a que consideramos que es un método más representativo para llevar a cabo la aproximación, puesto que intervienen de

forma directa las variables de de entrada del SLD en la salida del mismo. Aunque en ambos casos la estructura general es la misma, siendo la diferencia entre uno y otro, la forma en la cual manejan la parte consecuente. Los modelos tipo Sugeno se caracterizan por aproximar los sistemas haciendo uso de valores constantes (conocidos como singletons), o bien por polinomios de primer grado. Así la forma que tendrá cada una de las partes consecuentes de las reglas difusas es:

$$\begin{aligned} & \text{If } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\ & \text{Then } xp_{ir} = b0_{ir} + b1_{ir} * x_1 + \dots + bn_{ir} * x_n \end{aligned}$$

O bien

$$\begin{aligned} & \text{If } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\ & \text{Then } xp_r = b0_{ir} \end{aligned}$$

Donde x_i es la i -ésima variables de entrada, A_i es el conjunto difuso asociado a la i -ésima variable de entrada, xp_{ir} es la i -ésima salida de la regla número r , y $b0_{ir}$, $b1_{ir}, \dots$, bn_{ir} son constantes reales.

La parte antecedente de la regla (IF x_1 is A_1 and... and x_n is A_n), utiliza los conjuntos difusos definidos para cada una de las variables de entrada, y lo que hace es obtener el grado de pertenencia (denotado por w) de las variables de entrada a una región del espacio de discusión, definido por éstas y por las diferentes FM. Una vez que se tiene el grado de pertenencia de cada una de las variables de entrada, se requiere determinar el nivel de activación de la regla, para ello se puede hacer uso de la norma T o co-norma T [8](Las norma y co-norma T son formas de la intersección unión y

negación de conjuntos difusos. En nuestro caso y debido a la estructura de las reglas difusas utilizadas, es necesario recurrir a la norma T producto, en donde el nivel de activación que tendrá la regla, es el producto de todos los grados de pertenencia de cada una de las variables de entrada).

Un ejemplo de lo anterior se “x” (comprendido entre x_{\min} y x_{\max}), se dividió en dos regiones A_1 y A_2 , por lo cual se tendrán solo dos reglas

$$\text{IF } x_1 \text{ is } A_1 \text{ Then } xp_1 = b0_1 + b1_1 * x_1$$

$$\text{IF } x_1 \text{ is } A_2 \text{ Then } xp_2 = b0_2 + b1_2 * x_1$$

Para el valor x_0 se producirán dos niveles de activación (w), para A_1 se tiene $w_1 = 0.25$ y para A_2 $w_2 = 0.8$.

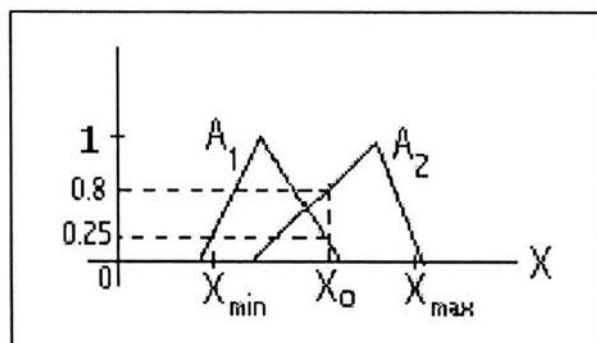


Fig. 2. 4 Funciones membresía de la variable X.

En la figura 2.4, se puede observar también que los conjuntos difusos se traslapan. El efecto de lo anterior es el de que al pasar x de una a otra región, la salida no sea discontinua como podemos ver en figura 2.5-b, sino que este cambio se dé de manera suave (figura 2.5-a).

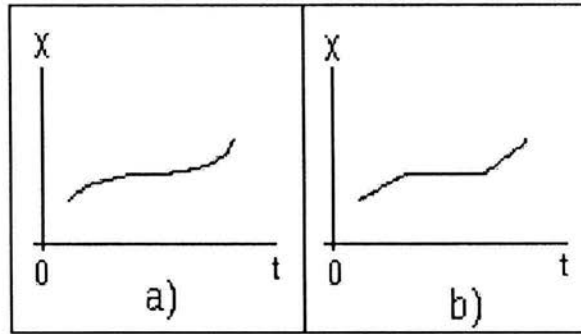


Fig. 2.5 Diferencia en la salida cuando los conjuntos difusos tienen zonas que se interceptan (a) y cuando están disjuntas (b)

En el caso que se tengan dos variables de entrada x_1 y x_2 , cada una tendrá una división de sus espacios, similar a lo mostrado en el ejemplo anterior, pero ahora los conjuntos difusos formarán un plano dividido en regiones (ver figura 2.6), de tal manera que la parte antecedente de cada regla, verifica si las variables pertenecen o no a cada uno de los subespacios y de acuerdo a esto obtener un grado de pertenencia diferente.

De esta misma forma, cuando tenemos n variables de entrada, cada una de las reglas tendrá a su vez n niveles de activación. De tal manera que será necesario determinar el nivel a utilizar en cada regla para lo cual existen varios métodos, como son tomar valor el mínimo o bien el productos de todo ellos. En nuestro caso, optamos porque el nivel de activación fuera el producto de todos los niveles que se tienen en cada regla, con el propósito de tomar en cuenta el grado de pertenencia de cada variable de entrada a una región respectiva.

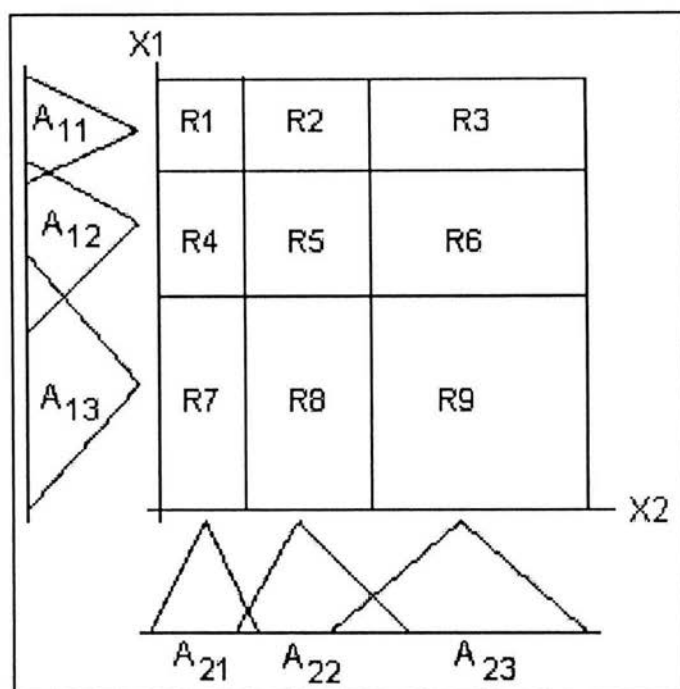


Fig. 2.6 División en subregiones el plano X_1 y X_2

Una vez que se tiene el nivel de activación de la regla, el paso siguiente es evaluar la parte consecuente. A este proceso se denomina defusificación y su función es la de convertir los valores difusos de las reglas en valores físicos. Los métodos de defusificación mas frecuentemente usados son:

1. Calculo del centro de gravedad de un área compuesta
2. Suma de los centros de gravedad de las áreas que componen la salida
3. Cociente entre las sumas de las salidas y los niveles de activación de cada una de las reglas del SLD.

4. Máximo o mínimo valor de la salida del conjunto de reglas
5. Valor medio entre el máximo y mínimo valor que se tiene a la salida de cada una de las reglas del SLD.

Los dos primeros métodos se ajustan más a los SLD de tipo Mamdani. De acuerdo a la estructura de los SLD tipo Sugeno, los métodos de defusificación 3 a 5 son los más apropiados, por lo cual optamos por elegir el método 3 por considerar que es el método más representativo al tomar en cuenta la salida propuesta de todas las reglas. De esta forma lo único que se hace es sustituir los valores de x_i en el polinomio correspondiente (Then $x_{pr} = b_{1r}$ ó Then $x_{pr} = b_{0r} + b_{1r} * x_1 + \dots + b_{nr} * x_n$), lo cual dará un valor x_{pr} por cada regla. Con lo que si se tiene r reglas, se tendrán r valores de x_p . De esta forma la salida del sistema difuso estará dada por la ecuación (2.3)

$$x_p = \frac{\sum_{k=1}^r x_{p_k} \cdot w_k}{\sum_{k=1}^r w_k} \quad (2.3)$$

Donde r es el número de reglas que tienen el sistema difuso, w es el grado de pertenencia obtenido en cada una de las reglas.

2.3 Ejemplo

Los SLD realizan la aproximación de funciones, dividiendo en primer lugar el espacio de discusión de la variable de salida en subconjuntos, y es cada subconjunto busca realizar una aproximación por medio de rectas, planos o hiperplanos en el caso de sistemas multidimensionales. Por en el caso de sistemas de una entrada una salida, buscaremos

aproximar la salida por medio de diferentes rectas, dependiendo de la región en la cual se encuentre la variable de entrada. Un ejemplo de lo anterior es el que se muestra en la figura 2.7, donde la función $f(x)$ es aproximada por cuatro rectas, dependiendo del valor que tenga la entrada y por lo tanto de la región en la que se encuentre.

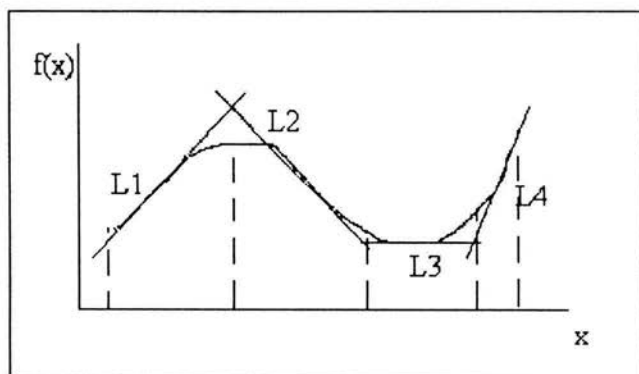


Fig. 2.7 Aproximación de una función por medio de rectas

A continuación presentaremos un ejemplo, de la forma en la cual se evalúa cada una de las bases de reglas difusas que se tienen. Para ello haremos uso de los siguientes conjuntos difusos: para x_1 el que se muestra en la figura 2.8, en donde como se muestra los niveles de activación para $x_1 = 35$, son $A_{11}(x_1) = 0.0$ y $A_{12}(x_1) = 0.6$.

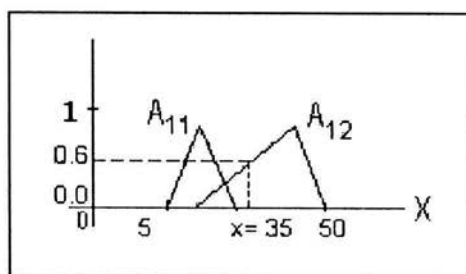


Fig. 2.8 Conjunto difuso para el estado 1

Para el segundo estado, se tiene el conjunto mostrado en la figura 2.9, en donde tenemos que los niveles de activación para $x_2 = 1.03$ son $A_{21}(x_2) = 0.93$ y $A_{22}(x_2) = 0.45$.

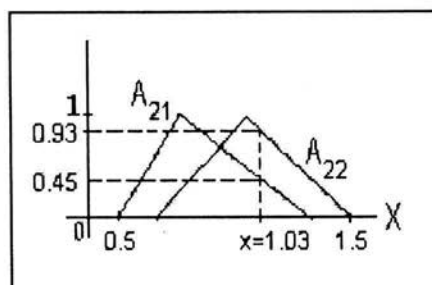


Fig. 2.9 Conjunto difuso para el estado 2

Si definimos la siguiente base de reglas difusas

1) IF x_1 is A_{11} and x_2 is A_{21}

$$\text{Then } xp_{11} = 0.25 + 0.03 * x_1 + 0.65 * x_2 + 0.9 * u_1$$

2) IF x_1 is A_{11} and x_2 is A_{22}

$$\text{Then } xp_{12} = 0.32 + 0.01 * x_1 + 0.36 * x_2 + 0.4 * u_1$$

3) IF x_1 is A_{12} and x_2 is A_{21}

$$\text{Then } xp_{13} = 0.53 + 0.91 * x_1 + 0.1 * x_2 + 0.68 * u_1$$

4) IF x_1 is A_{12} and x_2 is A_{22}

$$\text{Then } xp_{14} = 0.7 + 0.08 * x_1 + 0.82 * x_2 + 0.53 * u_1$$

y tenemos que $u = 3.7$, el resultado de la evaluación es:

Regla	$A(x_1)$	$A(x_2)$	Tv	Consecuente	xp
1	0.0	0.93	0.0	5.2995	0.0
2	0.0	0.45	0.0	2.5948	0.0
3	0.6	0.93	0.558	35.0299	19.5466
4	0.6	0.45	0.27	6.3156	1.7052
		Suma	0.828	Suma	21.2518

La salida del sistema, dada por la ecuación 2.3, es

$$xp_1 = 21.2518 / 0.828$$

$$xp_1 = 25.6664$$

CAPITULO III

ALGORITMOS DE BUSQUEDA Y OPTIMIZACIÓN

En este capítulo, haremos una descripción de los métodos utilizados para realizar la búsqueda de los parámetros del SLD, con el cual se busca realizar la identificación de la dinámica de un sistema no lineal. Estos métodos son el de Algoritmos Genéticos (AG's) [6] y el Método Gradiente (MG) [5].

3.1 Algoritmos genéticos

Los AG's son métodos de optimización, que surgen como una herramienta para la solución de problemas complejos en los cuales se involucran búsqueda y optimización. Están inspirados en el principio Darwiniano de selección natural y reproducción genética. Los principios de la naturaleza en los cuales están inspirados los AG's son muy simples. De acuerdo con la teoría de Darwin, el principio de selección, privilegia a los individuos más aptos, y los dota con una mayor longevidad, y con una mayor probabilidad de reproducción. Los individuos con más descendientes tienen una mayor oportunidad de transmitir su información genética a las próximas generaciones. Tal información genética constituye la identidad de cada individuo y están representados en los cromosomas. Estos principios son imitados en la construcción de algoritmos computacionales, que buscan la mejor solución para un determinado

problema, por medio de la evolución de una población de soluciones, codificadas a través de cromosomas artificiales. En los AG's, cada cromosoma es una estructura de datos que representa un elemento del espacio de soluciones del problema. Los Cromosomas son sometidos a un proceso de evolución que involucra evaluación, selección, recombinación (cruza) y mutación. Después de varios ciclos de evolución la población deberá contener individuos más aptos.

La analogía entre los AG's y los sistemas naturales se muestra a través de la siguiente tabla:

Naturaleza	Algoritmos Genéticos
Cromosoma	Palabra binaria, vector, etc.
Gen	Característica del problema
Genotipo	Estructura
Fenotipo	Estructura sometida al problema
Individuo	Solución
Generación	Ciclo

Podemos caracterizar los AG's a través de los siguientes componentes:

1. Problema de optimización
2. Representación de soluciones del problema
3. Decodificación del Cromosoma
4. Inicialización de la Población
5. Función objetivo
6. Estructura del AG
 - a. Evaluación
 - b. Criterio de selección

c. Operadores genéticos

7. Criterio de paro

A continuación describimos cada uno de los componentes anteriormente expuestos, incluyéndose ejemplos en algunos casos.

3.1.1 Problema de optimización

Los AG's son particularmente aplicados en problemas complejos de optimización: problemas con diversos parámetros o características que precisan ser combinadas en busca de la mejor solución; problemas con muchas restricciones o condiciones que no pueden ser representadas matemáticamente y problemas con grandes espacios de búsqueda. Los AG's han sido aplicados en diversos problemas de optimización, tales como:

Optimización de Funciones Matemáticas,
Optimización Combinatoria,
Optimización de Planeamiento,
Problema de Optimización de Rutas de Vehículos,
Optimización de Layout de Circuitos,
Optimización de Distribución,
Optimización en Negocios,
Síntesis de Circuitos Electrónicos, Etc.

En nuestro caso, lo que buscamos es identificar el comportamiento dinámico de un sistema físico no lineal, por medio del uso de SLD's.

3.1.2 Representación

Definir una representación apropiada, es parte del arte de usar AG's. Se debe usar una representación mínima, que sea completamente expresiva, de tal forma que sea capaz de representar cualquier solución al problema, que no admita soluciones no factibles. En caso de que el diseño de los cromosomas permita representar soluciones no factibles, entonces corresponde diseñar la función objetivo, de tal forma que ésta solo dé crédito parcial a las mismas. La representación no debe contener información innecesaria para dar la solución al problema. Aunque, puede haber mérito usando una representación que contiene material genético extra. Esto tiende a aumentar el tamaño del espacio de la búsqueda y a disminuir así la eficiencia del AG durante la exploración. El número de posibles representaciones es interminable. La representación de las posibles soluciones, dentro del espacio de búsqueda de un problema, define la estructura del cromosoma que va a ser manipulado por el algoritmo. Por ser más simple y fácil de manipular a través de los operadores genéticos, la representación binaria es la más empleada. Por ejemplo, sea el problema de encontrar el valor máximo de la función $f(x) = x^2$, siendo x entero $[0,63]$. Se pueden representar las soluciones del problema a través de un cromosoma de 6 bits.

C1: 0 0 1 0 0 1 representa $x = 9$

C2: 0 0 0 1 0 0 representa $x = 4$

En casos más complejos, podríamos incluso utilizar una representación real o compleja. Por ejemplo, si ahora lo que buscáramos es obtener la pareja de números (x, y) , que minimicen la función $f(x) = y^2 + 5x$, se podría no solo

utilizar una representación binaria, sino una representación real, en donde la solución del problema sería un cromosoma definido por parejas de números reales:

C1: (1.8, 0.07)

C2: (3.6, 1.7)

3.1.3 Decodificación del cromosoma

La decodificación del cromosoma, consiste básicamente en la construcción de la solución real del problema a partir de un cromosoma. El proceso de decodificación, construye la solución para que ésta sea evaluada por el problema. Es por ello que se debe utilizar la estructura de datos más apropiada para el problema en cuestión. El problema puede depender de una secuencia de sucesos. En cuyo caso una representación basada en el orden (lista o serie) podría ser apropiada. En muchos casos, se deben escoger a operadores que mantengan la integridad de la secuencia; los cruzamientos podrían generar reordenamientos de la secuencia, teniendo en cuenta que no haya elementos repetidos. En otros problemas, se podría desear representar las soluciones explícitamente, como árboles y aplicar directamente sobre estos los operadores. Alternativamente, muchas personas organizan árboles sobre arreglos o cadenas parciales y aplican los operadores a estas cadenas en lugar de a los árboles [6].

Algunos problemas incluyen una mezcla de elementos continuos y discretos, caso en que se puede necesitar crear una nueva estructura, para representar la mezcla de información. En estos casos, se deben definir operadores genéticos que respeten la estructura de la solución. Por ejemplo, en la búsqueda de una solución en la que

intervengan reales y enteros podrían usarse operadores de cruzamiento, que crucen partes enteras con partes enteras y partes flotantes con partes flotantes, pero que nunca mezclen partes enteras con flotantes.

3.1.4 Inicialización de la población

La inicialización de la población determina el proceso de creación de los individuos para el primer ciclo del algoritmo. Normalmente, la población inicial se forma a partir de individuos creados de forma aleatoria.

Las poblaciones iniciales creadas de forma aleatoria pueden ser sembradas con cromosomas buenos para conseguir una evolución más rápida, si se conocen a priori, el valor de estas “semillas” buenas, en caso contrario todos los elementos son determinados de manera aleatoria.

3.1.5 Estructura del AG

Una vez que el AG comienza a trabajar, está básicamente compuesto por los siguientes elementos, método de evaluación o función objetivo, un criterio de selección de individuos y de los operadores genéticos, los cuales a continuación describimos

Función objetivo

La evaluación es la unión entre el AG y el mundo externo. La evaluación se realiza a través de una función objetivo, que representa de forma adecuada el problema y tiene como objetivo suministrar una medida de aptitud, de cada individuo en la población actual. La función objetivo juega

un papel importante en la clasificación potencial de las soluciones en términos de sus características; es el criterio de optimización y evaluación de la calidad de los individuos. Se desarrolla a partir de los valores del fenotipo (donde designamos como fenotipo al conjunto de genes o elementos que componen al cromosoma). Por ejemplo supongamos que tenemos los cromosomas v_1 y v_2 :

v_1 (1000101110110101000111),
 v_2 (0000001110000000010000)

Y que estos representan a los valores $x_1 = 0.637179$ y $x_2 = -0.958973$ respectivamente. Si evaluamos a los cromosomas según determinada función $f(v)$ y obtenemos como resultado: $f(v_1) = 1.586345$ y $f(v_2) = 0.078878$ entonces, si estamos maximizando, podemos concluir que el cromosoma v_1 es el mejor de los dos.

Criterio de selección

La selección, es un elemento que aporta mayor potencia y robustez a la técnica de búsqueda, este operador cumple la función de hacer una selección de los mejores individuos, de manera que estos sean considerados en el proceso de generación de la nueva población. Existen varios mecanismos para su ejecución, como por ejemplo el método de la ruleta, piscina de selección [6], etc. En este punto, también es necesario considerar el mecanismo de reemplazo.

Los mecanismos de reemplazo por su parte, están muy relacionados con el tipo de AG que se implemente, siendo los más populares:

- Simple, descrito por Goldberg (1989), este algoritmo, no realiza reemplazo y tiene elitismo opcional. En cada generación el algoritmo crea una nueva población de individuos a partir de la cual se repite el proceso. La vieja población es desechada en cada generación.
- Con poblaciones de tamaño fijo, la nueva población se forma a partir de la población vieja y sus descendientes. Para conformar la nueva población el usuario, debe especificar el porcentaje de individuos viejos a reemplazar en cada generación, por los nuevos individuos.
- Incremental, en el que en cada generación solo uno o dos descendientes sobreviven.
- Con migraciones, este algoritmo permite la evolución de varias poblaciones de forma paralela, donde cada una de ellas evoluciona siguiendo un modelo de AG con tamaño fijo. En cada generación, el algoritmo establece una política de emigración entre las diferentes poblaciones.

Operadores genéticos

Entre los operadores genéticos más utilizados en la resolución de problemas utilizando AG se señalan:

El operador de cruza, permite el cambio de código genético entre los individuos, introduciendo así, instancias de nuevas combinaciones de genes. Éste facilita al AG una

búsqueda con eficiencia en espacios de varias dimensiones y su probabilidad de ocurrencia debe ser alta, para lograr una elevada introducción de nuevas soluciones del problema en cuestión.

El operador de mutación, provoca la posible aparición de nuevas características en el individuo, al mutar uno de sus genes, posibilitando así, la búsqueda en áreas del espacio solución del problema.

3.1.6 Criterios de paro

Los criterios de paro del algoritmo genético, están muy relacionados con la naturaleza del problema en cuestión. No obstante, podemos citar algunos de los criterios de paro más generales:

- Número de generaciones determinado.
- El mejor individuo de una población alcance un valor predeterminado de la función objetivo.
- Una determinada cantidad de individuos de una población sobrepase un valor de umbral predeterminado.
- La mayoría de los individuos de una población se encuentran en la vecindad de un punto. O sea que la mayoría de los individuos haya convergido a un mismo punto.

El criterio que optamos por utilizar para detener la ejecución del AG, fue el de producir un número máximo de generaciones.

3.2 Método Gradiente

Uno de los problemas que se presentan en el uso de AG's, es que en el momento en el que la solución se encuentra cercana a la ideal, la convergencia se vuelve lenta, es por ello que se hace necesario utilizar un método que nos facilite la obtención de la solución óptima cuando nos encontramos en esta situación. El MG, es una técnica por medio de la cual se pueden optimizar los parámetros de un SLD. Se basa en la información que se obtiene a partir del gradiente de una función de desempeño, la cual se busca optimizar, lo cual nos permite elegir los parámetros de un SLD, con una mayor velocidad de convergencia, cuando la solución ya se encuentra cerca de la ideal.

Para poder explicar mejor la forma de trabajar del método gradiente, haremos uso de una explicación gráfica. Ubiquémonos en la figura 3.1, donde el plano P representa el espacio de parámetros del SLD que soluciona el problema de identificación, y donde F_o representa el conjunto de parámetros óptimos y que nos brinda la mejor aproximación, con M_F como el radio del círculo dentro del cual se encuentran los mejores parámetros del SLD y al cual pertenece el F_{AG} que se obtiene como resultado del AG. Lo que hace el MG, es ajustar todos los parámetros de la parte consecuente de tal forma que conforme se va aplicando el algoritmo al SLD obtenido a partir del AG, converja de manera más rápida a la solución óptima.

El método consiste en la modificación de los parámetros de la parte consecuente de las reglas, de un SLD, de acuerdo al error que se va teniendo en cada una de las muestras. Para explicar el método, nos basaremos en reglas de la forma.

IF x_1 is A_1 and ... and x_n is A_n
 Then $x_{p_{ir}} = a_{0_{ir}} + a_{1_{ir}} * x_1 + \dots + a_{n_{ir}} * x_n$

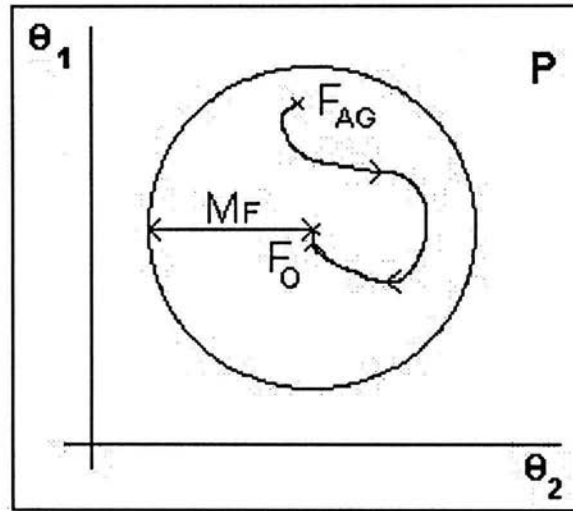


FIG. 3.1 Representación gráfica del método gradiente

A partir de esta estructura de reglas y considerando los coeficientes de los polinomios de la parte consecuente de las reglas, se define el siguiente vector de parámetros para un SLD con n entradas:

$$\theta = \begin{bmatrix} a_{0_{11}} & a_{1_{11}} \dots a_{n_{11}} & a_{0_{12}} & a_{1_{12}} \dots a_{n_{12}} \dots a_{0_{1r}} & a_{1_{1r}} \dots a_{n_{1r}} \\ a_{0_{21}} & a_{1_{21}} \dots a_{n_{21}} & a_{0_{22}} & a_{1_{22}} \dots a_{n_{22}} \dots a_{0_{2r}} & a_{1_{2r}} \dots a_{n_{2r}} \\ \vdots & & & & \vdots \\ a_{0_{n1}} & a_{1_{n1}} \dots a_{n_{n1}} & a_{0_{n2}} & a_{1_{n2}} \dots a_{n_{n2}} \dots a_{0_{nr}} & a_{1_{nr}} \dots a_{n_{nr}} \end{bmatrix}$$

(3.1)

De igual forma y considerando el nivel de activación de cada regla w_r , definimos también el vector ξ como:

$$\xi = [w_1 \ w_1 * x_1 \dots \ w_1 * x_n \dots \ w_r \ w_r * x_1 \dots \ w_r * x_n]^T \quad (3.2)$$

Si el error de estimación e , es definido como:

$$e = x - x_e \quad (3.3)$$

Donde x , es el valor real y $x_e = \theta \xi$, es el valor estimado.

Con el propósito de simplificar la explicación del modo de trabajar de los SLD, podemos Agrupar la parte consecuente de cada regla como

$$g_x(\theta, k) = a_{0r} + a_{1r} * x_1 + \dots + a_{nr} * x_n \quad (3.4)$$

Y haciendo $x_{pi} = f_i(x | \theta)$, tendremos que la salida del SLD estará dada por

$$f_i(x | k) = \frac{\sum_{j=1}^r g_j(x, k) \mu_j(x, k)}{\sum_{j=1}^r \mu_j(x, k)} \quad (3.5)$$

Con $\mu_j(x, k)$ es el nivel de activación de la j -ésima regla, está definida como

$$\mu_j(x, k) = \min [A_1(x_1) \dots A_n(x_n)] \quad (3.6)$$

de tal forma que $f_i(x|k)$ es la suma ponderada de los valores de salida de cada regla difusa.

Donde X es el universo de búsqueda, A_i es el conjunto difuso del i -ésimo estado (cabe mencionar que las entradas al SLD son los estados del sistema físico a identificar).

La ley de adaptación de los parámetros θ_{ij} estará definida de acuerdo al valor que tenga la norma 2 de la matriz θ . Se eligió emplear la norma dos, al considerar que el espacio de parámetros con un conjunto multidimensional.

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

$$\theta_{ij}(k+1) = \theta_{ij}(k) - \lambda \frac{\partial e_i^2}{\partial \theta_{ij}} \quad (3.7)$$

Cuando $\lambda > 0$ y donde

$$e_i = f_i(x | \theta) - X_i \quad (3.8)$$

Por cual, a partir de 3.8 podemos definir el error cuadrático como:

$$e_i^2 = 0.5 * (f_i(x | \theta) - X_i)^2 \quad (3.9)$$

De tal forma que

$$\frac{\partial e_i^2}{\partial \theta_{ij}} = e_i \frac{\partial f_i(x | \theta(k))}{\partial g_1(x, k)} \frac{\partial g_1(x, k)}{\partial \theta_{ij}} \quad (3.10)$$

Y

$$\frac{\partial f_i(x|\theta(k))}{\partial g_i(x|k)} = \frac{\mu_i(x|k)}{\sum_{j=1}^r \mu_j(x|k)} \quad (3.11)$$

$$\frac{\partial g_i(x|\theta(k))}{\partial a_{i0}(k)} = 1 \quad (3.12)$$

$$\frac{\partial g_i(x|\theta(k))}{\partial a_{ij}(k)} = x_j \quad (3.13)$$

El caso en el cual $\|\theta\|_2 > M_r$, se representa en la figura 3.2, en donde como podemos ver los parámetros del vector θ , están fuera del espacio óptimo, debido a esto, es necesario obtener la proyección del vector θ , por lo que la ley de adaptación estará dada por la ecuación:

$$\theta_i(k+1) = \theta_i(k) - \lambda \frac{\partial e^2}{\partial \theta_{ij}} + p(e, \theta, \xi) \frac{\partial g_i(x|\theta(k))}{\partial \theta_j(k)} \theta_i(k) \quad (3.14)$$

Con

$$p(e, \theta, \xi) = \frac{e_i \theta_r \xi(x)}{\langle \|\theta_r\|_2 } \quad (3.15)$$

El desarrollo de la ley de adaptación para SLD con una sola salida y múltiples entradas se encuentra en [5].

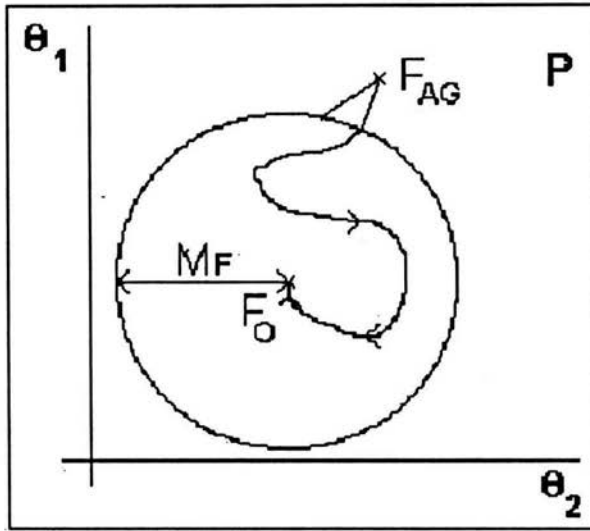


Fig. 3.2 Caso en el cual es necesario realiza la proyección del vector θ , debido a que se encuentra fuera del espacio óptimo

Los AG's y el MG son combinados con la intención de obtener un SLD por medio del cual se identifique la identificación de la dinámica de un sistema no lineal. Lo anterior se facilita por la manera en que ambos métodos se aplican, ya que mientras con los AG's podemos discriminar las diferentes posibles soluciones a nuestro problema dentro de un conjunto amplio, en tanto que el MG nos brinda la posibilidad de hacer una búsqueda local, al solo buscar dentro de un conjunto reducido, encerrado dentro de una bola de radio α , dentro del cual se encuentra la solución óptima, ahora en caso de que el MG no pueda mejorar la aproximación, el esquema entregara como resultado al sistema que se obtiene del AG.

Ahora, uno de los problemas que se tienen dentro de los AG's es el problema de la convergencia, lo anterior se debe a que no es posible asegura que existe, a menos que esta sea

como en nuestro caso por construcción. Lo anterior lo logramos debido a que a la siguiente generación y de acuerdo a la función objetivo que tenemos aseguramos que el error de estimación se mantenga o bien disminuya.

CAPITULO IV

MODELADO DE SISTEMAS NO LINEALES POR SLD's

4.1 Clase de sistemas no lineales

El objetivo que se persigue es el poder diseñar un sistema de lógica difusa (SLD), que sea capaz de reproducir la dinámica de un sistema físico no lineal. Los sistemas que buscamos identificar se caracterizan por tener disponible el vector de estados para su medición y al igual que de las diferentes entradas aplicadas, además suponemos que tienen la siguiente estructura:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x}) \mathbf{u} \quad (4.1-a)$$

Donde $\mathbf{x} \in \mathbb{R}^n$ es el vector de estados, $\mathbf{u} \in \mathbb{R}^m$ es el vector de entradas del sistema, con $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^{n \times 1}$ y $\mathbf{g}(\mathbf{x}) \in \mathbb{R}^{n \times m}$ funciones continuas de \mathbf{x} . De tal forma que la ecuación en variables de estado tiene la forma:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} g_{11} & g_{12} & \dots & g_{1m} \\ g_{21} & g_{22} & \dots & g_{2m} \\ \vdots & \vdots & & \vdots \\ g_{n1} & g_{n2} & \dots & g_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \quad (4.1-b)$$

Para poder diseñar el modelo difuso que cumpla con el objetivo planteado, se hizo uso del método de Algoritmos

Genéticos (AG's) y Método Gradiente (MG). Los AG's se utilizan para realizar una búsqueda global y definir la estructura del SLD (los conjuntos difusos y la base de reglas). El MG lo aplicamos únicamente para ajustar los coeficientes de la parte consecuente de la base de reglas.

Un ejemplo representativo del tipo de sistemas que buscamos identificar, es el sistema hidráulico de dos tanques que mostramos en la figura 4.1, es un ejemplo del tipo de sistemas que deseamos identificar.

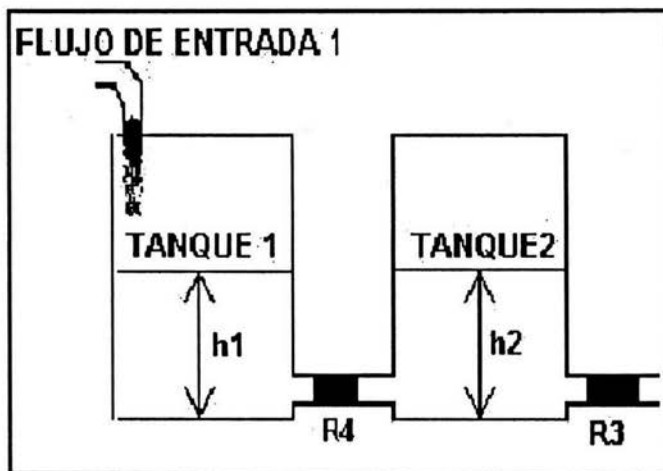


Fig. 4.1 sistema no lineal de dos tanques

El modelo matemático de este sistema es sencillo de obtener por medio de leyes físicas. Si elegimos como estados del sistema, el nivel de agua en cada uno de los tanques. De tal forma que el volumen del líquido en el tanque es

$$\boxed{V(h) = A h} \quad (4.2)$$

Donde A es el área transversal del tanque y es constante y h es el nivel de agua dentro. Ahora si definimos la variación del volumen de líquido dentro del tanque como:

$$\boxed{\frac{dV}{dt} = Q_{\text{ent}} - Q_{\text{sal}}} \quad (4.3)$$

La variación de los estados, estará dada por la diferencia entre el agua que entra menos la que sale, como se muestra en la siguiente ecuación:

$$\boxed{\dot{h} = \frac{1}{A} Q_{\text{ent}} - \frac{1}{A} Q_{\text{sal}}} \quad (4.4)$$

De la ecuación de Bernoulli, el flujo de salida los definimos de la siguiente forma:

$$\boxed{Q_{12} = s\sqrt{|h_1 - h_2|}} \quad (4.5)$$

Donde s es una constante que depende de los parámetros del sistema. Por lo que el modelo matemático del sistema, estará definido por la ecuación (4.6), que tiene la estructura presentada por las ecuaciones (4.1-a) y (4.1-b).

$$\boxed{\begin{bmatrix} \dot{h}_1 \\ \dot{h}_2 \end{bmatrix} = \begin{bmatrix} -\frac{s}{A}\sqrt{|h_1 - h_2|} \\ \frac{s}{A}\sqrt{|h_1 - h_2|} - \frac{s}{A}\sqrt{h_2} \end{bmatrix} + \begin{bmatrix} \frac{1}{A} \\ 0 \end{bmatrix} \mathbf{u}} \quad (4.6)$$

4.2 Aproximación de las dinámicas no lineales por SLD's

Considérese el sistema no lineal, con n estados y m entradas, cuya estructura se muestra en la ecuación (4.1).

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x}) \mathbf{u} \quad (4.7)$$

la ecuación (4.7) la podemos escribir en términos de un SLD como:

$$\hat{\mathbf{f}}(\mathbf{x}) = \theta_{\mathbf{f}}^* \mathbf{c}(\mathbf{x}) + \Delta \mathbf{f}(\mathbf{x}) \quad (4.8)$$

y

$$\hat{\mathbf{g}}(\mathbf{x}) = \theta_{\mathbf{g}}^* \mathbf{c}(\mathbf{x}) + \Delta \mathbf{g}(\mathbf{x}) \quad (4.9)$$

Donde θ^* es el vector de parámetros de los polinomios de la parte consecuente de las reglas y que se define de la misma forma que en (3.1), $\Delta \mathbf{f}(\mathbf{x})$ y $\Delta \mathbf{g}(\mathbf{x})$ son los errores de aproximación de las funciones y $\mathbf{c}(\mathbf{x})$ es un vector que tiene como renglones los niveles de activación de cada una de las reglas del SLD, y que es definido de igual forma que en la ecuación (3.2).

Así que un aproximación de las dinámicas dadas en (4.7) es

$$\dot{\hat{\mathbf{x}}} = -\alpha \mathbf{e} + \hat{\mathbf{f}}(\mathbf{x}) + \hat{\mathbf{g}}(\mathbf{x}) \mathbf{u} \quad (4.10)$$

Donde

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} \quad (4.11)$$

Con $\alpha > 0$, y

$$\hat{f}(x) = \hat{\theta}_f^T c(x) \quad (4.12)$$

$$\hat{g}(x) = \hat{\theta}_g^T c(x) \quad (4.13)$$

Donde θ_f y θ_g , están definidas por la ecuación 3.1, mientras que ε , se define como

$$\mathcal{E} = [w_1 \ w_1^* \ x_1 \ \dots \ w_1 \ x_n \ \dots \ w_r \ w_r^* \ x_1 \ \dots \ w_r \ x_n]^T \quad (4.14)$$

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

$$\dot{\tilde{e}} = -\alpha e + [f(x) - \hat{f}(x)] + [g(x) - \hat{g}(x)]u \quad (4.15)$$

Sustituyendo las ecuaciones (4.8), (4.9), (4.12) y (4.13) en (4.15), tenemos

$$\dot{\tilde{e}} = -\alpha e + [\theta_f^* - \hat{\theta}_f] c(x) + [\theta_g^* - \hat{\theta}_g] c(x)u + \Delta f(x) + \Delta g(x)u \quad (4.16)$$

Si agrupamos los siguientes términos

$$w = \Delta f(x) + \Delta g(x)u \quad (4.17)$$

$$\tilde{\theta}_f = \theta_f^* - \hat{\theta}_f \quad (4.18)$$

$$\tilde{\theta}_g = \theta_g^* - \hat{\theta}_g \quad (4.19)$$

Tendremos que

$$\dot{\tilde{\mathbf{e}}} = -\alpha \mathbf{e} + \tilde{\boldsymbol{\theta}}_f \boldsymbol{\varepsilon}(\mathbf{x}) + \tilde{\boldsymbol{\theta}}_g \boldsymbol{\varepsilon}(\mathbf{x}) \mathbf{u} + \boldsymbol{\omega} \quad (4.20)$$

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

$$V = \frac{1}{2} \mathbf{e} \mathbf{e}^T + \frac{1}{2\gamma} \text{Tr} (\tilde{\boldsymbol{\theta}}_f \tilde{\boldsymbol{\theta}}_f^T) + \frac{1}{2\gamma} \text{Tr} (\tilde{\boldsymbol{\theta}}_g \tilde{\boldsymbol{\theta}}_g^T) \quad (4.21)$$

Derivando V tendremos que

$$\dot{V} = \mathbf{e}^T \dot{\tilde{\mathbf{e}}} + \frac{1}{\gamma} \text{Tr} \left(\dot{\tilde{\boldsymbol{\theta}}}_f \tilde{\boldsymbol{\theta}}_f^T \right) + \frac{1}{\gamma} \text{Tr} \left(\dot{\tilde{\boldsymbol{\theta}}}_g \tilde{\boldsymbol{\theta}}_g^T \right) \quad (4.22)$$

Sustituyendo e en (4.15) y (4.20)

$$\dot{V} = \mathbf{e}^T (-\alpha \mathbf{e} + \tilde{\boldsymbol{\theta}}_f \boldsymbol{\varepsilon}(\mathbf{x}) + \tilde{\boldsymbol{\theta}}_g \boldsymbol{\varepsilon}(\mathbf{x}) \mathbf{u} + \boldsymbol{\omega}) + \frac{1}{\gamma} \text{Tr} \left(\dot{\tilde{\boldsymbol{\theta}}}_f \tilde{\boldsymbol{\theta}}_f^T \right) + \frac{1}{\gamma} \text{Tr} \left(\dot{\tilde{\boldsymbol{\theta}}}_g \tilde{\boldsymbol{\theta}}_g^T \right) \quad (4.23)$$

De la ecuación anterior se tiene que

$$\dot{V} = -\alpha \mathbf{e}^T \mathbf{e} + \frac{1}{\gamma} \text{Tr} \left[\tilde{\boldsymbol{\theta}}_f \left(\dot{\tilde{\boldsymbol{\theta}}}_f^L + \gamma \boldsymbol{\varepsilon}(\mathbf{x}) \mathbf{e}^T \right) \right] + \frac{1}{\gamma} \text{Tr} \left[\tilde{\boldsymbol{\theta}}_g \left(\dot{\tilde{\boldsymbol{\theta}}}_g^L + \gamma \boldsymbol{\varepsilon}(\mathbf{x}) \mathbf{u} \mathbf{e}^T \right) \right] + \mathbf{e} \boldsymbol{\omega} \quad (4.24)$$

El termino $\alpha \mathbf{e}$ de la ecuación (4.20) se agrego restando ya que como se puede apreciar en (4.24) se tendrá siempre un término negativo en la derivada de la función positiva

definida 4.22, lo cual y de acuerdo con la teoría de Lyapunov ayudara a la estabilidad del sistemas.

Si elegimos

$$\begin{cases} \dot{\tilde{\theta}}_f^* = \gamma e(x) e^T \\ \dot{\tilde{\theta}}_g^* = \gamma e(x) u e^T \end{cases} \quad (4.25)$$

Entonces

$$\dot{V} = -\alpha e^T e + e^T \omega \quad (4.26)$$

De esta forma si consideramos la norma 2 y la equivalencia de normas, tendremos que

$$\dot{V} \leq -\alpha \|e\|^2 + \|e\| \|\omega\| \quad (4.27)$$

Definiendo ω_0 como

$$\omega_0 = \sup_{t \rightarrow \infty} \|\omega(t)\| \quad (4.28)$$

El cual asumimos es finito, por lo cual

$$\dot{V} \leq -\alpha \|e\| + \omega_0 \|e\| \quad (4.29)$$

De esta forma para

$$\|e\| > \frac{\omega_0}{\alpha} \quad (4.30)$$

Tendremos que

$$\dot{V} < 0 \quad (4.31)$$

Mientras la condición (4.30) 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

$$r = \frac{\omega_0}{\alpha} \quad (4.32)$$

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

Lo antes expuesto no sirve para demostrar por medio de la teoría de estabilidad de Lyapunov, como un sistema puede ser aproximado por medio de una SLD dentro de un conjunto compacto con una precisión arbitrariamente pequeña, las cual estará en función del valor que tenga α .

La teoría de Lyapunov, también puede ser utilizada en la elección de los parámetros de las reglas y en la ubicación de las funciones membresía, con el propósito de establecer como resultado un SLD estable.

Para poder determinar los parámetros de los SLD's, utilizamos en primer lugar el método de AG's, con el propósito de realizar una búsqueda global, y posteriormente el MG, para realizar una búsqueda local.

4.3 Búsqueda de SLD's por medio de AG's

Como ya mencionamos, el tipo de sistemas que deseamos identificar son aquellos cuya estructura tiene la forma dada por la ecuación (4.1). De tal forma que lo que buscamos, es la identificación del sistema por medio de modelos basados en lógica difusa, para lo cual se propone la estructura mostrada en la ecuación (4.33).

$$\boxed{\hat{x} = -\hat{a}e + \hat{f}(\mathbf{x}) + \hat{g}(\mathbf{x})\mathbf{u}} \quad (4.33)$$

Donde $\mathbf{x} \in \mathbb{R}^n$ es el vector de estados, $\mathbf{u} \in \mathbb{R}^m$ es el vector de entradas del sistema, y con $\hat{f}(\mathbf{x})$ y $\hat{g}(\mathbf{x})$ son SLD's. En los cuales la parte antecedente es la misma para todos, cambiando solo la parte consecuente. Para propósitos de nuestro trabajo, se optó porque la parte consecuente de \hat{f} , fuera representada por polinomios de primer grado ($\hat{f}(\mathbf{x})_{1r} = a_{01r} + a_{11r} * x_1 + \dots + a_{n1r} * x_n$), mientras que para las funciones \hat{g} , solo utilizamos singletons ($\hat{g}_{ij} = b_{0ij}$). Es así que para \hat{f} se tendrán n funciones, mientras que para \hat{g} habrá $m * n$, de tal forma que la r - esima regla tendrán la siguiente estructura:

If x_1 is A_1 and ... and x_n is A_n
 Then $x_{p1r} = a_{01r} + a_{11r} * x_1 + \dots + a_{n1r} * x_n$
 :
 :
 And $x_{pnr} = a_{0nr} + a_{1nr} * x_1 + \dots + a_{nnr} * x_n$
 And $g_{11r} = b_{011r}$
 :
 :
 And $g_{1mr} = b_{0nmr}$

Tomando en cuenta lo anterior, el n-ésimo estado, estará dado por:

$$\hat{x}_n^* = -\alpha e_n + \hat{f}_n(x) + \hat{g}_{n1}(x) * u_1 + \dots + \hat{g}_{nm}(x) * u_m \quad (4.34)$$

O bien:

$$\hat{x}_i^* = -\alpha e_i + \frac{\sum_{r=1}^k \hat{f}_{ir} * w_r}{\sum_{r=1}^k w_r} + \frac{\sum_{r=1}^k \hat{g}_{ir} * w_r}{\sum_{r=1}^k w_r} u_1 + \dots + \frac{\sum_{r=1}^k \hat{g}_{imr} * w_r}{\sum_{r=1}^k w_r} u_m \quad (4.35)$$

La ecuación (4.35) la podemos agrupar de tal forma que solo se tenga una sumatoria, como muestra la ecuación (4.36).

$$\hat{x}_i^* = -\alpha e_i + \frac{\sum_{r=1}^k (\hat{f}_{ir} + \hat{g}_{ir} * u_1 + \dots + \hat{g}_{imr} * u_m) * w_r}{\sum_{r=1}^k w_r} \quad (4.36)$$

Considerando la teoría estimación y por cuestiones de estabilidad del sistema, a la ecuación (4.36), es necesario agregarle un término adicional, y con el cual tomaremos en cuenta el error de estimación que se tiene, denotado por la ecuación (4.37) como:

$$e_i = x_i - \hat{x}_i \quad (4.37)$$

Es así que el modelo de estimación que se tiene es:

$$\hat{x}_i^* = -\alpha e_i + \frac{\sum_{r=1}^k (\hat{E}_{ir} + \hat{G}_{i1r} * u_1 + \dots + \hat{G}_{imr} * u_m) * w_r}{\sum_{r=1}^k w_r}$$

(4.38)

Donde e es el error de estimación y α es una constante positiva.

Considerando la ecuación (4.36) y con el propósito de simplificar la programación, se optó porque cada una de las reglas tenga la siguiente estructura

If x_1 is A_1 and ... and x_n is A_n
Then $x_{p_{jr}} = a_{0_{jr}} + a_{1_{jr}} * x_1 + \dots + a_{n_{jr}} * x_n +$
 $b_{0_{jr}} * u_{1r} + \dots + b_{m_{jr}} * u_m$

De tal forma que si se tiene un sistema de dos estados, una entrada y 4 MF, la base de reglas es:

Para el primer estado:

If x_1 is A_{11} and x_2 is A_{21}
Then $x_{p_{11}} = a_{0_{11}} + a_{1_{11}} * x_1 + a_{2_{11}} * x_2 + b_{0_{11}} * u_1$
If x_1 is A_{11} and x_2 is A_{22}
Then $x_{p_{11}} = a_{0_{12}} + a_{1_{12}} * x_1 + a_{2_{12}} * x_2 + b_{0_{12}} * u_1$
If x_1 is A_{12} and x_2 is A_{21}
Then $x_{p_{11}} = a_{0_{13}} + a_{1_{13}} * x_1 + a_{2_{13}} * x_2 + b_{0_{13}} * u_1$
If x_1 is A_{12} and x_2 is A_{22}
Then $x_{p_{11}} = a_{0_{14}} + a_{1_{14}} * x_1 + a_{2_{14}} * x_2 + b_{0_{14}} * u_1$

Para el segundo:

If x_1 is A_{11} and x_2 is A_{21}

*Then $x_{p21} = a_{021} + a_{121} * x_1 + a_{221} * x_2 + b_{021} * u_1$*

If x_1 is A_{11} and x_2 is A_{22}

*Then $x_{p21} = a_{022} + a_{122} * x_1 + a_{222} * x_2 + b_{022} * u_1$*

If x_1 is A_{12} and x_2 is A_{21}

*Then $x_{p21} = a_{023} + a_{123} * x_1 + a_{223} * x_2 + b_{023} * u_1$*

If x_1 is A_{12} and x_2 is A_{22}

*Then $x_{p21} = a_{024} + a_{124} * x_1 + a_{224} * x_2 + b_{024} * u_1$*

A continuación haremos una descripción del AG y los Operadores Genéticos (OG) utilizados en el trabajo.

4.3.1 Algoritmo genético

Como ya mencionamos, 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 cruce 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 $i < \text{gen_max}$
 - Evaluar población
 - Clasificar la población
 - Elige padre(s)
 - Aplica operador genético

Cruza y/o mutación o agrega o quita reglas

- Presentar resultados

A continuación haremos una descripción detallada de cada uno de los puntos mencionados, aunque previamente explicaremos la estructura del cromosoma empleado.

Cromosoma de un SLD: La representación del SLD por medio del cual buscamos la identificación de la dinámica de un sistema físico no lineal de n estado y m entradas, es por medio de dos matrices. La primer matriz, contiene los parámetros de las funciones membresía que componen cada uno de los conjuntos difusos que se tienen. La segunda matriz, contiene los coeficientes de los polinomios que componen la parte consecuente de cada una de las reglas del SLD. De esta forma, un sistema que tiene r reglas y s funciones membresía por cada conjunto difuso asociado a cada estado del sistema, donde el k -ésimo conjunto difuso y la j -ésima regla difusa tienen la estructura mostrada a continuación:

$$A_k = \{ [x_{i_{k1}}, x_{c_{k1}}, x_{d_{k1}}], \dots, [x_{i_{ks}}, x_{c_{ks}}, x_{d_{ks}}] \} \quad k=1, \dots, n$$

If x_1 is A_1 and ... and x_n is A_n

$$\text{Then } xp_{1j} = P_{1j}(x) + P_{1j}(u)$$

:

$$\text{And } xp_{nj} = P_{nj}(x) + P_{nj}(u) \quad j=1, \dots, r$$

$$P_{ij}(x) = a_{0ij} + a_{1ij} * x_1 + \dots + a_{nij} * x_n$$

$$P_{ij}(u) = b_{1ij} * u_1 + \dots + b_{mij} * u_m \quad i = 1, \dots, n$$

se tendrá el siguiente par de matrices

$$B = \begin{bmatrix} x_{i_{11}} & xc_{11} & xd_{11} & \dots & x_{i_{1s}} & xc_{1s} & xd_{1s} \\ x_{i_{21}} & xc_{21} & xd_{21} & \dots & x_{i_{2s}} & xc_{2s} & xd_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_{i_{n1}} & xc_{n1} & xd_{n1} & \dots & x_{i_{ns}} & xc_{ns} & xd_{ns} \end{bmatrix}$$

$$\Theta = \begin{bmatrix} a_{0_{11}} a_{1_{11}} \dots a_{n_{11}} b_{1_{11}} \dots b_{m_{11}} & a_{0_{12}} a_{1_{12}} \dots a_{n_{12}} b_{1_{12}} \dots b_{m_{12}} & \dots & a_{0_{1r}} a_{1_{1r}} \dots a_{n_{1r}} b_{1_{1r}} \dots b_{m_{1r}} \\ a_{0_{21}} a_{1_{21}} \dots a_{n_{21}} b_{1_{21}} \dots b_{m_{21}} & a_{0_{22}} a_{1_{22}} \dots a_{n_{22}} b_{1_{22}} \dots b_{m_{22}} & \dots & a_{0_{2r}} a_{1_{2r}} \dots a_{n_{2r}} b_{1_{2r}} \dots b_{m_{2r}} \\ \vdots & \vdots & \ddots & \vdots \\ a_{0_{n1}} a_{1_{n1}} \dots a_{n_{n1}} b_{1_{n1}} \dots b_{m_{n1}} & a_{0_{n2}} a_{1_{n2}} \dots a_{n_{n2}} b_{1_{n2}} \dots b_{m_{n2}} & \dots & a_{0_{nr}} a_{1_{nr}} \dots a_{n_{nr}} b_{1_{nr}} \dots b_{m_{nr}} \end{bmatrix}$$

donde B es la matriz de parámetros de las funciones membresía de los conjuntos difusos y, Θ es la matriz de coeficientes de los polinomios de la parte consecuente de las reglas difusas.

Generar la población inicial: Una vez que se tiene los campos de la estructura dinámica, se generan de forma aleatoria los valores reales o enteros correspondientes. Aunque cabe hacer una distinción entre la forma de generar los parámetros de la parte consecuente de las reglas y los de las FM. Los coeficientes de los polinomios que componen la parte consecuente de las reglas, se eligen de manera aleatoria por medio de una distribución uniforme. Los parámetros de las FM's, se determinan de tal forma que éstas se encuentren dentro del rango en el cual el estado asociado se desenvuelve. Por ejemplo, si el rango de valores de una de las variables está comprendido entre $X_{\min.}$ y $X_{\max.}$, el centro de la función triangular pertenecerá a $[X_{\min.}, X_{\max.}]$, mientras que para definir las cotas derecha e izquierda se abre el rango, al intervalo $[X_{\min.} - 0.25R, X_{\max.} + 0.25R]$, donde:

$$R = |x_{\max.} - x_{\min.}| \quad (4.39)$$

Un ejemplo de lo anterior se muestra en la figura 4.2. En donde el centro de todas la FM se encuentra dentro del intervalo $[X_{\min.}, X_{\max.}]$, mientras que las cotas izquierda y derecha de las FM's A_1 y A_3 respectivamente, están fuera de dicho rango.

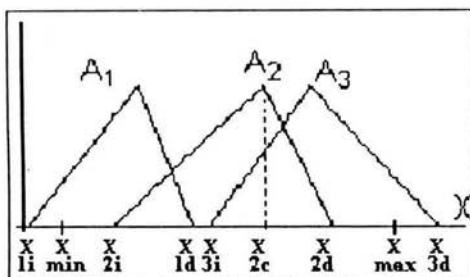


Fig. 4.2 Distribución de las funciones membresía para un estado

Evaluar población: En este punto se simula cada uno de los sistemas difusos 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, se procede a obtener el error de estimación cuadrático, el cual a su vez sirve como criterio de clasificación, de manera que la ecuación de error (4.40) se convierte en nuestra función objetivo, con la cual buscamos reducir el error de estimación.

$$E = \sum_{i=1}^n \sum_{j=1}^k (x_{ji} - \check{x}_{ji})^2 \quad (4.40)$$

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 sistemas difusos fue evaluado obteniendo su error de

estimación, se acomodan los elementos de la población, de menor a mayor de acuerdo a su error. Habiendo clasificado la población, el mejor elemento conserva todas sus características, mientras que los demás, serán utilizados para generar la nueva población y después son eliminados.

Elección de padre(s): Aquí se determina 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 usado es la ruleta, en el cual los mejores elementos tienen una mayor probabilidad de ser seleccionados.

Operadores Genéticos (OG): Son una modificación sistemática que sufren los parámetros de cada uno de los cromosomas, estos operadores son:

- a) Cruza
- b) Mutación

Además de la aplicación de los OG's, se aplicaron también dos algoritmos adicionales, que tiene como propósito agregar o quitar reglas a cada SLD

Criterio de paro: El criterio de paro que utilizamos, es el número de generaciones a producir, aunque el programa se diseñó para que si el usuario desea finalizar el AG antes, lo pueda hacer y le presente los resultados obtenidos, o bien, si desea tener un mayor número de generaciones, también lo pueda llevar a cabo.

4.3.2 Operadores genéticos

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 2 funciones membresía por estado, por lo que el número de reglas posible es 4. Las funciones membresía se denotarán por A_{ij} donde i representara el estado a la cual pertenece la FM y j el número de función. Los parámetros de las FM's se representarán como un vector de tres componentes, acomodados como sigue $[e_{ij}, ec_{ij}, ed_{ij}]$ y $[f_{ij}, fc_{ij}, fd_{ij}]$, donde e_i y f_i representan la cota izquierda, ec y fc el centro y ed y fd la cota derecha de las FM's. Mientras que los coeficientes de la parte consecuente de las reglas serán representados por las letras ak_{ir} , bk_{ir} y ck_{ir} , dk_{ir} , donde k va de 0 a n (con n como el numero de variables de entrada del SLD), i el número de estado y r el número de regla. De esta forma los sistemas son:

Sistema 1

$$A_{11} = [e_{11}, ec_{11}, ed_{11}].$$

$$A_{12} = [e_{12}, ec_{12}, ed_{12}].$$

$$A_{21} = [e_{21}, ec_{21}, ed_{21}].$$

$$A_{22} = [e_{22}, ec_{22}, ed_{22}].$$

IF x_1 is A_{11} and x_2 is A_{21}

THEN $x_{p11} = a_{011} + a_{111} * x_1 + a_{211} * x_2 + b_{011} * u$

IF x_1 is A_{11} and x_2 is A_{22}

THEN $x_{p11} = a_{012} + a_{112} * x_1 + a_{212} * x_2 + b_{012} * u$

IF x_1 is A_{12} and x_2 is A_{21}

THEN $x_{p11} = a_{013} + a_{113} * x_1 + a_{213} * x_2 + b_{013} * u$

IF x_1 is A_{12} and x_2 is A_{22}

THEN $x_{p11} = a_{014} + a_{114} * x_1 + a_{214} * x_2 + b_{014} * u$

Sistema 2

$$A_{11} = [f_{i11}, f_{c11}, f_{d11}].$$

$$A_{12} = [f_{i12}, f_{c12}, f_{d12}].$$

$$A_{21} = [f_{i21}, f_{c21}, f_{d21}].$$

$$A_{22} = [f_{i22}, f_{c22}, f_{d22}].$$

If x_1 is A_{11} and x_2 is A_{21}

$$\text{THEN } xp_{11} = c0_{11} + c1_{11} * x_1 + c2_{11} * x_2 + d0_{11} * u$$

If x_1 is A_{11} and x_2 is A_{22}

$$\text{THEN } xp_{11} = c0_{12} + c1_{12} * x_1 + c2_{12} * x_2 + d0_{12} * u$$

If x_1 is A_{12} and x_2 is A_{21}

$$\text{THEN } xp_{11} = c0_{13} + c1_{13} * x_1 + c2_{13} * x_2 + d0_{13} * u$$

If x_1 is A_{12} and x_2 is A_{22}

$$\text{THEN } xp_{11} = c0_{14} + c1_{14} * x_1 + c2_{14} * x_2 + d0_{14} * u$$

Algoritmos de cruce

Los algoritmos de cruce, son un medio por el cual podemos generar nuevas cromosomas, a partir de la combinación de los parámetros de dos individuos de la generación anterior. En este trabajo, se opta por incluir tres algoritmos de cruce, que son:

- Cruce aritmética simple
- Cruce aritmética entera
- Cruce Heurística

dichos algoritmos fueron utilizados en [1], [4] y [7], presentando buenos resultados, los cuales a continuación se aplica

Cruza aritmética simple: En este caso lo único que se hace es intercambiar parámetros a partir de un punto de cruce. Este punto es determinado de manera aleatoria. En el caso de los conjuntos difusos, lo que hacemos es generar un número aleatorio dentro del rango $[1, \beta)$ (donde β es el número de FM's que tenemos), con lo cual aseguramos intercambiar al menos una FM y cuando mas $\beta-1$. Para la parte consecuente de las reglas, hacemos lo mismo, generamos un número en el rango $[1, \tau)$ (donde τ es el número de reglas), con lo que aseguramos intercambiar al menos una regla y cuando mas $\tau -1$. Esto dos aspectos se pueden ver de forma mas clara en el siguiente ejemplo. A partir de los sistemas 1 y 2 mostrados al principio de la sección, les aplicamos el algoritmo con los puntos de cruce 1, para las FM's y 4 para la base de reglas, con lo que tendremos los siguientes nuevos elementos (en donde las partes intercambiadas se muestran resaltadas).

Hijo 1

$$A_{11} = [c_{i11}, e_{c11}, e_{d11}].$$

$$A_{12} = [f_{i12}, f_{c12}, f_{d12}].$$

$$A_{21} = [c_{i21}, e_{c21}, e_{d21}].$$

$$A_{22} = [f_{i22}, f_{c22}, f_{d22}].$$

IF x_1 is A_{11} and x_2 is A_{21}

$$\text{THEN } xp_{11} = a_{011} + a_{111} * x_1 + a_{211} * x_2 + b_{011} * u$$

IF x_1 is A_{11} and x_2 is A_{22}

$$\text{THEN } xp_{11} = a_{012} + a_{112} * x_1 + a_{212} * x_2 + b_{012} * u$$

IF x_1 is A_{12} and x_2 is A_{21}

$$\text{THEN } xp_{11} = a_{013} + a_{113} * x_1 + a_{213} * x_2 + b_{013} * u$$

IF x_1 is A_{12} and x_2 is A_{22}

$$\text{THEN } xp_{11} = c_{014} + c_{114} * x_1 + c_{214} * x_2 + d_{014} * u$$

Hijo 2

$$A_{11} = [f_{i11}, fc_{11}, fd_{11}].$$

$$A_{12} = [e_{i12}, ec_{12}, ed_{12}].$$

$$A_{21} = [f_{i21}, fc_{21}, fd_{21}].$$

$$A_{22} = [e_{i22}, ec_{22}, ed_{22}].$$

IF x_1 is A_{11} and x_2 is A_{21}

$$\text{THEN } xp_{11} = c0_{11} + c1_{11} * x_1 + c2_{11} * x_2 + d0_{11} * u$$

IF x_1 is A_{11} and x_2 is A_{22}

$$\text{THEN } xp_{11} = c0_{12} + c1_{12} * x_1 + c2_{12} * x_2 + d0_{12} * u$$

IF x_1 is A_{12} and x_2 is A_{21}

$$\text{THEN } xp_{11} = c0_{13} + c1_{13} * x_1 + c2_{13} * x_2 + d0_{13} * u$$

IF x_1 is A_{12} and x_2 is A_{22}

$$\text{THEN } xp_{11} = a0_{14} + a1_{14} * x_1 + a2_{14} * x_2 + b0_{14} * u$$

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 del hijo 1 los obtenemos mediante la siguiente ecuación:

$$h1 = v * p1 + (1-v) * p2 \quad (4.41)$$

mientras que los del hijo dos:

$$h2 = v * p2 + (1-v) * p1 \quad (442)$$

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 la diferencia de los parámetros de los padres, de tal forma que los parámetros del hijo 1 los obtenemos mediante la ecuación:

$$h1 = p1 - v * (p2 - p1) \quad (4.43)$$

y los del hijo dos por:

$$h2 = p2 - v * (p1 - p2) \quad (4.44)$$

Donde v es un número decimal aleatorio y que de igual forma el operador se aplica a los parámetros de la parte consecuente de las reglas, y a los correspondientes a los conjuntos difusos.

Algoritmos de mutación

Los algoritmos de mutación, son un método a partir del cual se generan nuevos individuos, como resultado de modificaciones que sufren los parámetros de los cromosomas de la generación anterior. Los algoritmos de mutación utilizados son:

- Mutación uniforme
- Mutación uniforme simple
- Mutación escalar
- Mutación total

En el primer algoritmo fue utilizado en [1] y [7] presentando buenos resultados, mientras que el segundo, esta baso en el primero, solo que aquí los datos se ven de una forma diferente, y el tercer algoritmo, se propuso con la intención de probar su funcionamiento.

A continuación explicaremos los operadores de mutación realizados, donde la parte que fue mutada aparecerá resaltada en los ejemplos.

Mutación uniforme: En este operador, cada función membresía de los conjuntos difusos se ve como un solo ente, de tal forma que si sufre una mutación, cambiaremos los tres parámetros que la componen y no solo uno. En el caso de la base de reglas difusas, sucede una situación similar, en donde cada regla se ve como un ente del conjunto, de tal manera que si se decide mutarla, modificaremos todos sus coeficientes. La probabilidad de que cada FM o regla sea mutada, varia entre 0.05 y 0.1. Lo anterior lo mostramos en el siguiente ejemplo, en donde como podemos ver la función membresía A_{21} fue mutada, y por lo tanto cambiaron sus tres parámetros, mientras que en la base de reglas se decidió mutar solo la número 2 (de acuerdo a los sistemas mostrados en las paginas 47 y 48).

Hijo

$$A_{11} = [e_{i11}, e_{c11}, e_{d11}].$$

$$A_{12} = [e_{i12}, e_{c12}, e_{d12}].$$

$$A_{21} = [E_{i21}, E_{c21}, E_{d21}].$$

$$A_{22} = [e_{i22}, e_{c22}, e_{d22}].$$

If x_1 is A_{11} and x_2 is A_{21}

THEN $x_{p11} = a_{011} + a_{111} * x_1 + a_{211} * x_2 + b_{011} * u$

IF x_1 is A_{11} *and* x_2 is A_{22}

THEN $x_{p11} = c_{012} + c_{112} * x_1 + c_{212} * x_2 + d_{012} * u$

IF x_1 is A_{12} *and* x_2 is A_{21}

THEN $x_{p11} = a_{013} + a_{113} * x_1 + a_{213} * x_2 + b_{013} * u$

IF x_1 is A_{12} *and* x_2 is A_{22}

THEN $x_{p11} = a_{014} + a_{114} * x_1 + a_{214} * x_2 + b_{014} * u$

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. La probabilidad de que cada parámetro sea mutado varía entre 0.05 y 0.1. Para ejemplificar lo anterior explicaremos el siguiente caso. En la parte de los conjuntos difusos solo se modificó el parámetro ec_{21} por Ec_{21} de la función A_{21} , el nuevo elemento es determinado de forma aleatoria, requiriendo únicamente que se encuentre dentro del rango de operación de la variable de entrada. La elección de los nuevos coeficientes de los polinomios de la parte consecuente de las reglas se eligen de manera aleatoria. En el caso de las reglas presentamos un mayor número de cambios, más precisamente en las reglas 1, 3 y 4 (conforme a los ejemplos de las páginas 47 y 48).

Sistema 1

$$A_{11} = [ei_{11}, ec_{11}, ed_{11}].$$

$$A_{12} = [ei_{12}, ec_{12}, ed_{12}].$$

$$A_{21} = [ei_{21}, **Ec**_{21}, ed_{21}].$$

$$A_{22} = [ei_{22}, ec_{22}, ed_{22}].$$

If x_1 is A_{11} and x_2 is A_{21}

THEN $x_{p11} = \mathbf{A0}_{11} + a_{111} * x_1 + a_{211} * x_2 + b_{011} * u$

If x_1 is A_{11} and x_2 is A_{22}

THEN $x_{p11} = a_{012} + a_{112} * x_1 + a_{212} * x_2 + b_{012} * u$

If x_1 is A_{12} and x_2 is A_{21}

THEN $x_{p11} = a_{013} + \mathbf{A1}_{13} * x_1 + a_{213} * x_2 + b_{013} * u$

If x_1 is A_{12} and x_2 is A_{22}

THEN $x_{p11} = a_{014} + a_{114} * x_1 + a_{214} * x_2 + \mathbf{B0}_{14} * u$

Mutación escalar: Este operador, consiste en modificar los parámetros de la parte consecuente de las reglas y/o los de las FM's que componen el SLD. En el caso de las FM's, lo que hacemos es sumar o restar un factor de corrimiento a cada parámetro, el cual se genera de forma aleatoria. En la parte consecuente de las reglas, lo que se hacemos es multiplicar todos los coeficientes de los polinomios por un mismo factor, que de igual forma es generado de manera aleatoria. Esto se muestra en el siguiente ejemplo, en donde como podemos ver, a todas las FM's se le hizo un corrimiento definido por "q", que puede ser un número positivo o negativo, mientras que en la parte consecuente, todos los coeficientes de las reglas fueron multiplicados por "p".

Sistema 1

$$A_{11} = [ei_{11}+q, ec_{11}+q, ed_{11}+q].$$

$$A_{12} = [ei_{12}+q, ec_{12}+q, ed_{12}+q].$$

$$A_{21} = [ei_{21}+q, ec_{21}+q, ed_{21}+q].$$

$$A_{22} = [ei_{22}+q, ec_{22}+q, ed_{22}+q].$$

If x_1 is A_{11} and x_2 is A_{21}

THEN $x_{p11} = p * a_{011} + p * a_{111} * x_1 + p * a_{211} * x_2 + p * b_{011} * u$

If x_1 is A_{11} and x_2 is A_{22}

THEN $x_{p11} = p \cdot a_{012} + p \cdot a_{112} \cdot x_1 + p \cdot a_{212} \cdot x_2 + p \cdot b_{012} \cdot u$

If x_1 is A_{12} and x_2 is A_{21}

THEN $x_{p11} = p \cdot a_{013} + p \cdot a_{113} \cdot x_1 + p \cdot a_{213} \cdot x_2 + p \cdot b_{013} \cdot u$

If x_1 is A_{12} and x_2 is A_{22}

THEN $x_{p11} = p \cdot a_{014} + p \cdot a_{114} \cdot x_1 + p \cdot a_{214} \cdot x_2 + p \cdot b_{014} \cdot u$

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

El propósito del trabajo como ya se menciona, es generar un SLD que identifique el comportamiento de un sistema no lineal, pero además, que sea con el menor número de reglas posible. Es por ello que a cada una de las reglas que se tiene, se le agrego 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, entonces

$$x_{p11} = a_{012} + a_{112} \cdot x_1 + a_{212} \cdot x_2 + b_{012} \cdot u$$

Si la bandera es 0, entonces

$$x_{p11} = 0$$

De esta forma para agregar o disminuir reglas, lo único que se hace es cambiar la bandera de uno a cero y de cero a uno. Una vez que se hicieron los cambios se evalúa el sistema, si

el error disminuyó se mantienen los cambios, de lo contrario se deja el sistema como estaba antes de realizar las modificaciones, Este algoritmo se incluyó debido a que al generarse las reglas difusas, se toman en cuenta regiones o situaciones en la cuales el sistema físico no se puede encontrar.

Los algoritmos antes presentados (cruza y mutación), son aplicados dentro del esquema diseñado para llevar a cabo la identificación de sistemas no lineales por medio SLD. Presentando buenos resultados al realizar la combinación de ambos algoritmos.

4.4 Búsqueda de parámetros óptimos por método gradiente

Lo que buscamos hacer con el MG, es ajustar los parámetros de la parte consecuente de las reglas del SLD que se obtiene a partir del AG, con el propósito de mejorar su desempeño. Como entrada al MG tendremos un conjunto de reglas de la siguiente forma.

If x_1 is A_1 *and* ... *and* x_n is A_n

Then $x_{p_{ir}} = a_{0_{ir}} + a_{1_{ir}} * x_1 + \dots + a_{n_{ir}} * x_n + b_{1_{ir}} * u_1 + \dots + b_{m_{ir}} * u_m$

Donde r representa el número de regla, i representa el estado al cual pertenece el SLD (recordemos que por cada estado, existe una parte consecuente independiente en la base de reglas).

A partir de las reglas, formamos el vector de parámetros

$$\varepsilon = \begin{bmatrix} a_{011} a_{11} \dots a_{n11} b_{111} \dots b_{m11} a_{012} a_{112} \dots a_{n12} b_{112} \dots b_{m12} \dots a_{01r} a_{11r} \dots a_{n1r} b_{11r} \dots b_{m1r} \\ a_{021} a_{121} \dots a_{n21} b_{121} \dots b_{m21} a_{022} a_{122} \dots a_{n22} b_{122} \dots b_{m22} \dots a_{02r} a_{12r} \dots a_{n2r} b_{12r} \dots b_{m2r} \\ \dots \\ a_{0r1} a_{1r1} \dots a_{nr1} b_{1r1} \dots b_{mr1} a_{0r2} a_{1r2} \dots a_{nr2} b_{1r2} \dots b_{mr2} \dots a_{0rr} a_{1rr} \dots a_{nr} b_{1rr} \dots b_{mrr} \end{bmatrix} \quad (4.45)$$

y el vector ε como

$$\varepsilon = \begin{bmatrix} w_{11} w_{11} x_1 \dots w_{11} x_n w_{11} u_1 \dots w_{11} u_m \dots w_{1r} w_{1r} x_1 \dots w_{1r} x_n w_{1r} u_1 \dots w_{1r} u_m \\ w_{21} w_{21} x_1 \dots w_{21} x_n w_{21} u_1 \dots w_{21} u_m \dots w_{2r} w_{2r} x_1 \dots w_{2r} x_n w_{2r} u_1 \dots w_{2r} u_m \\ \dots \\ w_{nr1} w_{nr1} x_1 \dots w_{nr1} x_n w_{nr1} u_1 \dots w_{nr1} u_m \dots w_{nr} w_{nr} x_1 \dots w_{nr} x_n w_{nr} u_1 \dots w_{nr} u_m \end{bmatrix} \quad (4.46)$$

Si definimos $g_r(x, k)$ como

$$g_r(x, k) = a_{0ri} + a_{1ri} * x_1 + \dots + a_{nri} * x_n + b_{1ri} * u_1 + \dots + b_{mri} * u_m \quad (4.47)$$

La salida del SLD estará dada por

$$xp_i = \frac{\sum_{j=1}^r g_j(x, k) \mu_j(x, k)}{\sum_{j=1}^r \mu_j(x, k)} \quad (4.48)$$

Donde $\mu_j(x, k)$ esta definida como

$$\mu_j(x, k) = \operatorname{argmin} [A_1(x_1) \dots A_n(x_n)] \quad (4.49)$$

De tal forma que la ley de adaptación de los parámetros θ_{ij} es la siguiente.

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

$$\theta_{ij}(k+1) = \theta_{ij}(k) - \lambda \frac{\partial e_i}{\partial \theta_{ij}} \quad (4.50)$$

Cuando $\lambda > 0$ y

$$e_i = 0.5 * (f_i(x | \theta) - X_i)^2 \quad (4.51)$$

Donde $x p_i = f_i(x | \theta)$, de tal forma que de (4.48) tenemos

$$\frac{\partial e_i}{\partial \theta_{ij}} = e_i \frac{\partial f_i(x | \theta(k))}{\partial g_i(x, k)} \frac{\partial g_i(x, k)}{\partial \theta_{ij}} \quad (4.52)$$

Donde

$$e_i = x p_i - x_i \quad (4.53)$$

Y

$$\frac{\partial x p_i}{\partial g_i(x | k)} = \frac{\mu_i(x | k)}{\sum_{j=1}^r \mu_j(x | k)} \quad (4.54)$$

$$\frac{\partial g_i(x | \theta(k))}{\partial a_{i0}(k)} = 1 \quad (4.55)$$

$$\frac{\partial g_i(x | \theta(k))}{\partial a_{ij}(k)} = X_j \quad (4.56)$$

$$\frac{\partial g_i(x | \theta(k))}{\partial b_{ij}(k)} = U_j \quad (4.57)$$

Si $\|\theta\|_2 > M_r$ es necesario obtener la proyección de los parámetros, por lo que la ley de adaptación estará dada por la ecuación:

$$\theta_i(k+1) = \theta_i(k) - \lambda \frac{\partial e}{\partial \theta_{ij}} + p(e, \theta, \epsilon) \frac{\partial g_i(x|\theta(k))}{\partial \theta_j(k)} \theta_i(k) \quad (4.58)$$

Con

$$p(e, \theta, \epsilon) = \frac{e_i \theta_r \epsilon(x)}{(\|\theta_r\|_2)^2} \quad (4.59)$$

En el siguiente capítulo, mostraremos los resultados obtenidos de la aplicación del esquema diseñado a la identificación de sistemas no lineales denotados por la ecuación 4.1-a y 4.1-b. La identificación se lleva a cabo por medio de un SLD, al ser estos una herramienta mediante la cual se puede llevar a cabo la aproximación de las dinámicas de sistemas físicos, como se demostró en el punto 4.2. Para llevar a cabo el diseño del SLD, se aplicaron en primero AG's (utilizando diferente algoritmos de cruce y mutación), con el propósito de establecer la estructura y los parámetros iniciales del SLD a emplear para realizar la identificación. Posteriormente, al SLD obtenido a partir del AG, es empleado el MG gradiente con el propósito de sintonizar de una manera más precisa los coeficientes de la parte consecuente de las reglas del SLD.

RESULTADOS

El esquema de identificación por medio de SLD desarrollado, fue aplicado al sistema de dos tanques mostrado en la figura 5.1. El sistema a identificar consta de dos tanques conectados por una válvula, al primer tanque se encuentra conectada una bomba, por medio de la cual se le suministra agua al sistema, mientras que el segundo tiene una segunda válvula que permite la salida de agua. En este caso los estados son los niveles de agua de cada uno de los tanques.

Los resultados que presentamos son de dos tipos, en primer lugar, mostramos los correspondientes a datos obtenidos por medio de simulación, y posteriormente de datos experimentales.

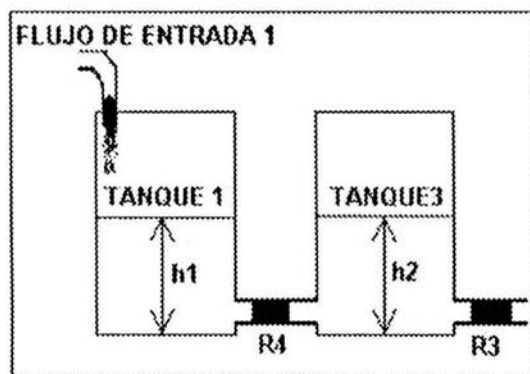


Fig. 5.1 Sistema de dos tanques, utilizados en el segundo experimento

5.1 Sistema de dos tanques (datos simulados)

En este caso, los datos con los cuales se entrenó al SLD, con el cual se busca realizar la identificación, fueron obtenidos por medio de una simulación del sistema en MATLAB. La señal de entrada que se aplicó fue una Señal Binaria Pseudo Aleatoria (SBPA) mostrada en la figura 5.2.

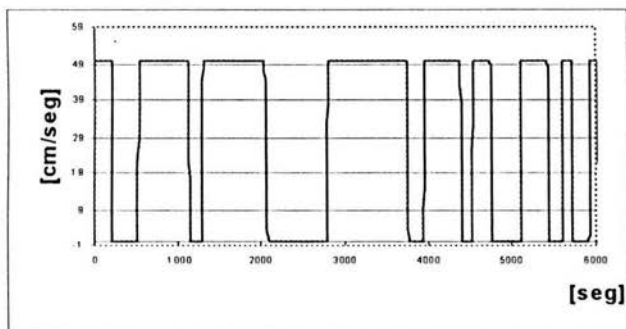


Fig. 5.2 Señal de entrada aplicada al segundo experimento

En la figura 5.3, se muestra el comportamiento de los estados x_1 y x_2 tras ser aplicada la SBPA.

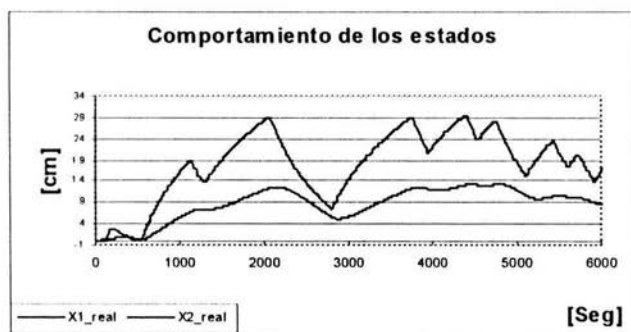


Fig. 5.3 Comportamiento de los estados del experimento 2, Después de aplicar una señal cuadrada

En la figura 5.4, presentamos el comportamiento del error de estimación durante la aplicación del AG. En esta gráfica, podemos ver como durante las primeras 500 generaciones, se da una velocidad de convergencia rápida, y en cambio en las restantes 4500 la velocidad es ya muy lenta.

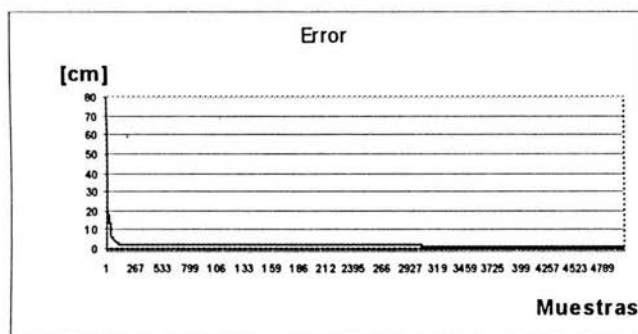


Fig. 5.4 Error de estimación a una señal de entrada cuadrada

En la figura 5.5, presentamos la comparación entre los datos de entrenamiento y los obtenidos SLD resultado del AG. Como podemos ver la dinámica del sistema ya se sigue, presentándose únicamente un error en la magnitud de la estimación.

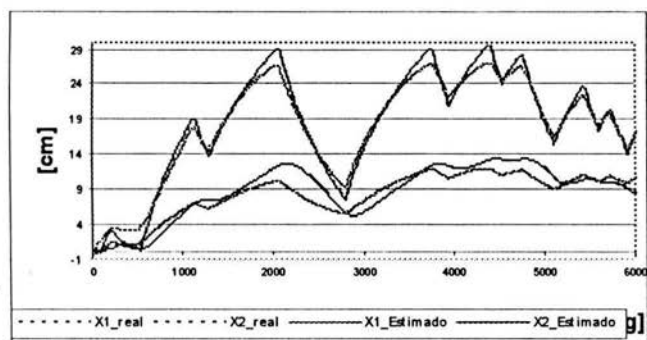


Fig. 5.5 Salida del SLD obtenido del AG

En la figura 5.6, presentamos el comportamiento que tuvo la norma del vector de parámetros del SLD obtenido del AG, conforme fue aplicado el MG.

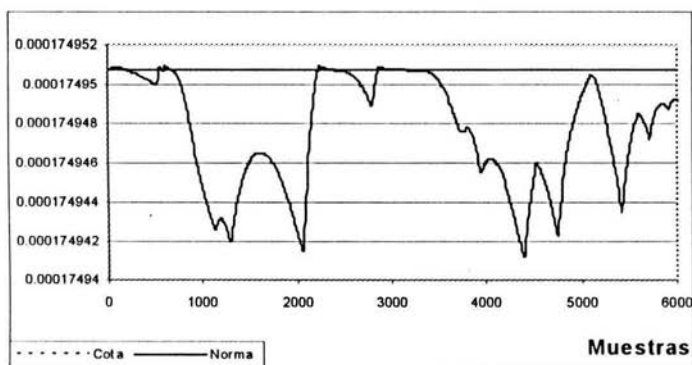


Fig. 5.6 Comportamiento de la norma 2 del vector de parámetros Θ

En la gráfica de la figura 5.7, mostramos la salida del SLD después de la aplicación del MG, sobre los parámetros de la parte consecuente de las reglas.

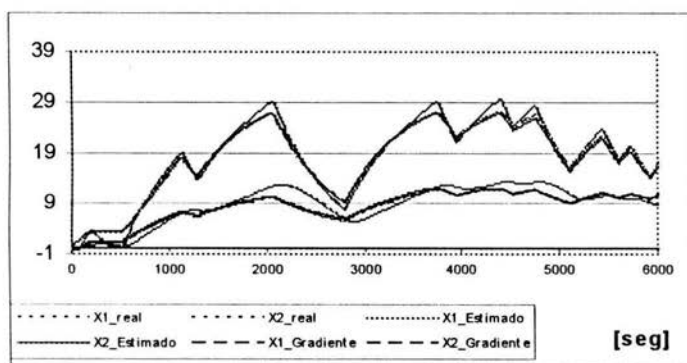


Fig. 5.7 Comparación entre las salidas de los SLD obtenidos del AG y MG

A continuación presentamos la información sobre los conjuntos difusos y las reglas que componen el SLD con el cual se lleva a cabo la identificación del sistema no lineal (ésta es la salida que se obtiene en el archivo SLD.txt, generado por el programa). En la figura 5.8 y 5.9, mostramos de forma gráfica las FM's que forman el conjunto difuso asociado a cada uno de los estados del sistema.

La base de reglas que obtuvo como resultado es

Para el estado 1 se tienen las siguientes funciones membresía:

Λ_1 .- Tipo: Triangular
Parámetros [-9.084384, 20.960299, 36.055237]
 Λ_2 .- Tipo: Triangular
Parámetros [-0.832439, 11.370654, 21.770197]
 Λ_3 .- Tipo: Triangular
Parámetros [-12.669076, 3.348529, 29.673872]

Para el estado 2 se tienen las siguientes funciones membresía:

Λ_1 .- Tipo: Triangular
Parámetros [-2.373866, 8.444750, 15.452755]
 Λ_2 .- Tipo: Triangular
Parámetros [-4.454601, 0.459088, 7.699459]
 Λ_3 .- Tipo: Triangular
Parámetros [9.493004, 11.977927, 14.211250]

La base de reglas para el F1 es:

if X1 is Λ_1 & X2 is Λ_1
Then Y = 0.006001 + -0.006893 * X1 + -0.001331 * X2 + 0.005065 * U1
if X1 is Λ_3 & X2 is Λ_2
Then Y = 0.001486 + -0.000634 * X1 + 0.004849 * X2 + 0.000222 * U1

La base de reglas para el F2 es:

if X1 is Λ_1 & X2 is Λ_1
Then Y = -0.004179 + 0.005444 * X1 + -0.005975 * X2 + -0.000211 * U1
if X1 is Λ_2 & X2 is Λ_1
Then Y = -0.008787 + -0.002233 * X1 + -0.007515 * X2 + -0.000589 * U1
if X1 is Λ_1 & X2 is Λ_2
Then Y = -0.002144 + 0.008567 * X1 + 0.004065 * X2 + 0.001325 * U1
if X1 is Λ_3 & X2 is Λ_2
Then Y = 0.001545 + 0.008478 * X1 + 0.000103 * X2 + 0.000259 * U1
if X1 is Λ_1 & X2 is Λ_3

Then $Y = 0.001416 + 0.000398 * X1 + -0.002411 * X2 + 0.000509 * U1$
 if $X1$ is $A2$ & $X2$ is $A3$

Then $Y = -0.006546 + 0.002540 * X1 + -0.006077 * X2 + -0.000014 * U1$

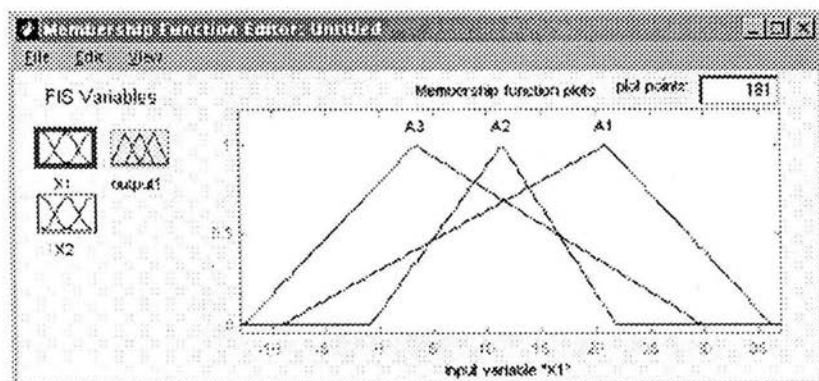


Fig. 5.8 Conjunto difuso del estado 1

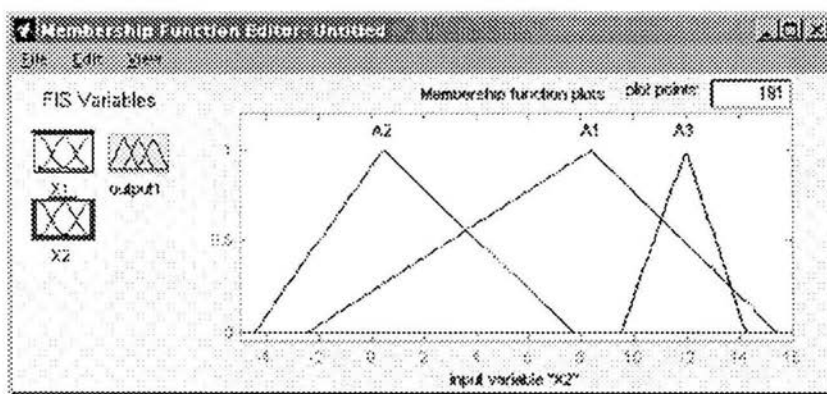


Fig. 5.9 Conjunto difuso del estado 2

5.2 Sistema de dos tanques (datos experimentales)

Presentamos los resultados obtenidos tras la aplicación del esquema diseñado al sistema de dos tanques, en donde los datos de entrenamiento se obtuvieron ahora de forma experimental. El experimento consistió en la aplicación de

una SBPA, que mostramos en la figura 5.10. El comportamiento de los estados lo podemos ver en la figura 5.11.

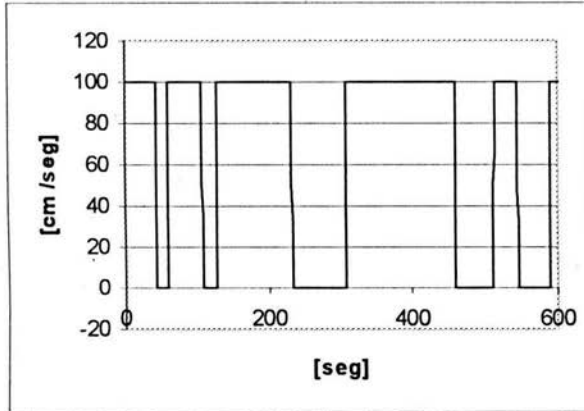


Fig. 5.10 Señal aplicada al sistema real.

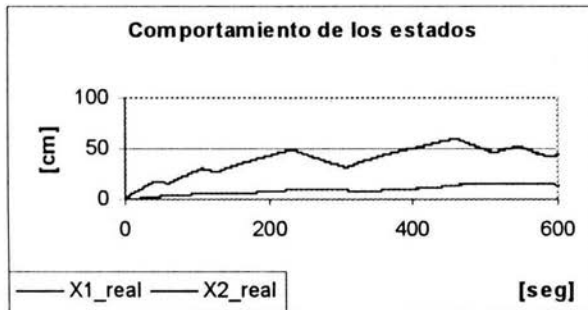
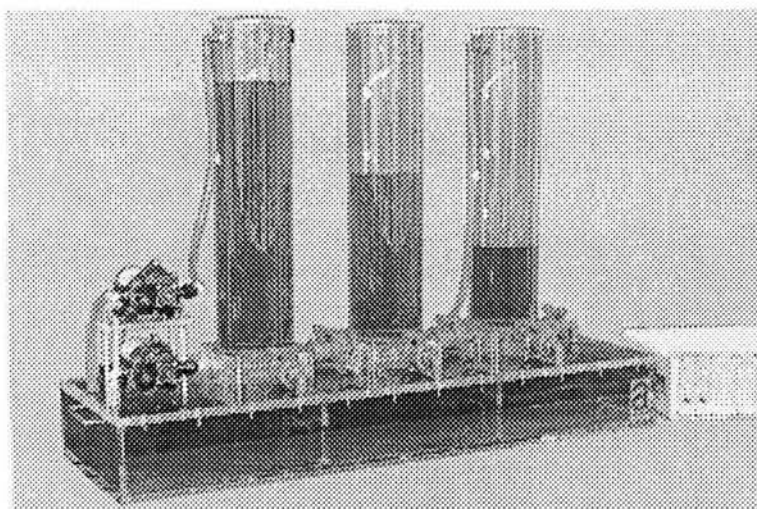


Fig. 5.11 Respuesta de los del sistema tras aplicar la señal de la fig. 5.10.

Para llevar a cabo la captura de los datos, hicimos uso de una interfaz diseñada en LABVIEW, y además del sistema DTS200 desarrollado por AMIRA y que mostramos en la figura 5.12.



Sistema DTS200 desarrollado por AMIRA

Los datos del experimento se utilizaron como entrada al sistema desarrollado, para entrenar un SLD por medio del cual reproducimos la dinámica del sistema real. En la figura 5.13, vemos la comparación entre los datos del comportamiento de los estados del sistema físico, y los del SLD obtenido a partir de la aplicación del AG.

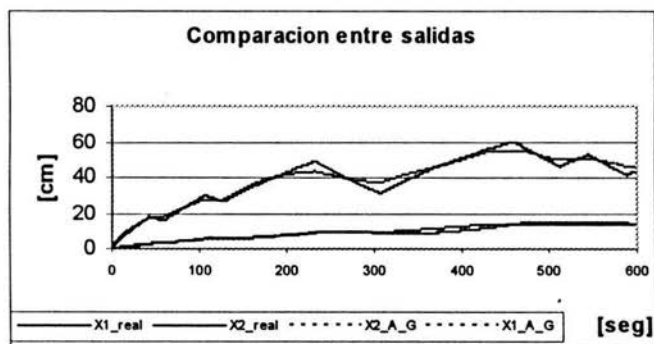


Fig. 5.13 Comparación entre los estados reales y los estados estimados, después de aplicar el AG.

En la Figura 5.14, mostramos el comportamiento del error de estimación del mejor individuo de cada una de las generaciones del AG.

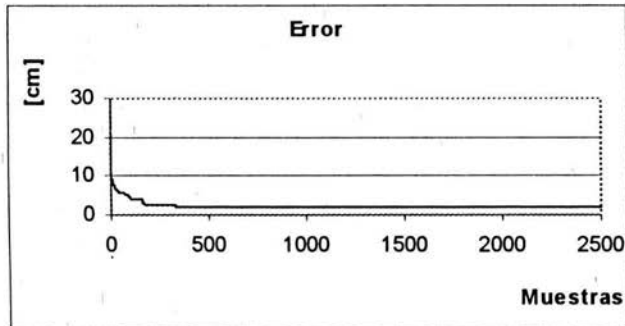


Fig. 5.14 Comportamiento del error de estimación durante la aplicación del AG.

Al SLD resultado del AG, le aplicamos el MG con el propósito de mejorar su desempeño, obteniendo el comportamiento mostrado en la figura 5.15.

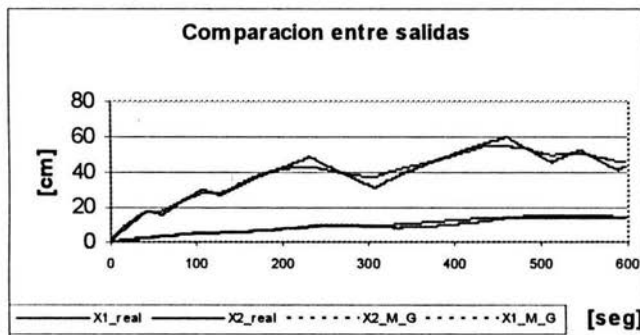


Fig. 5.15 Salida del SLD después de aplicar el MG al sistema obtenido por el AG.

En la figura 5.16, 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 regresamos de nueva cuenta los parámetros del SLD al espacio óptimo.

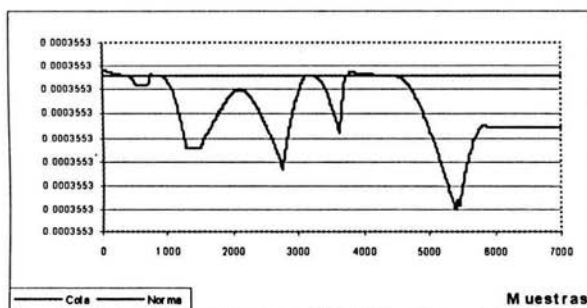


Fig. 5.16 Aplicación de la ley de adaptación del MG al vector Θ

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 = 0.008715 + 0.000508 * X1 + -0.000275 * X2 + 0.000343 * U1

if X1 is A2 & X2 is A1

Then Y = 0.001326 + 0.000158 * X1 + 0.000430 * X2 + 0.002193 * U1

if X1 is A3 & X2 is A1

Then Y = 0.003203 + 0.006248 * X1 + 0.002705 * X2 + -0.004459 * U1

if X1 is A2 & X2 is A2

Then Y = -0.000362 + 0.000996 * X1 + -0.007147 * X2 + 0.008432 * U1

if X1 is A3 & X2 is A2

Then Y = -0.000060 + -0.000050 * X1 + -0.000514 * X2 + -0.001450 * U1

if X1 is A1 & X2 is A3

Then Y = 0.000323 + 0.000062 * X1 + 0.000079 * X2 + 0.000320 * U1

if X1 is A2 & X2 is A3

Then Y = -0.005598 + 0.000010 * X1 + -0.000005 * X2 + 0.000001 * U1

if X1 is A3 & X2 is A3

Then Y = -0.000064 + -0.006116 * X1 + 0.000990 * X2 + 0.000187 * U1

La base de reglas para el F2 es:

if X1 is A1 & X2 is A1

Then Y = 0.006129 + 0.000086 * X1 + 0.000369 * X2 + 0.000887 * U1

if X1 is A2 & X2 is A2

Then Y = 0.006271 + 0.000093 * X1 + -0.000093 * X2 + 0.000001 * U1

if X1 is A2 & X2 is A3

Y = -0.006122 + 0.000345 * X1 + -0.000508 * X2 + 0.000116 * U1

if X1 is A3 & X2 is A3

Then Y = 0.003950 + -0.000024 * X1 + -0.000002 * X2 + 0.000492 * U1

La representación gráfica de la funciones membresía de cada uno de los conjuntos difusos asociados a cada estado, se muestran en las figuras 5.17 y 5.18.

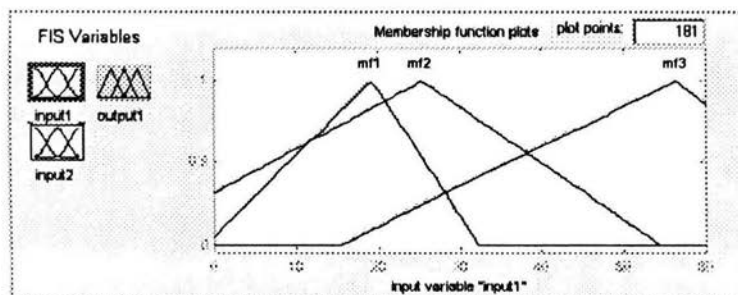


Fig. 5.17 Funciones membresía del conjunto difuso del primer estado.

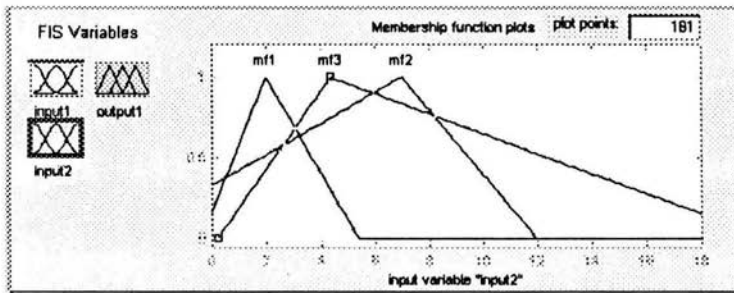


Fig. 5.17 Funciones membresía del conjunto difuso del segundo estado.

Como podemos apreciar, en los dos casos el AG y MG presentan una buena aproximación al lograr reproducir la dinámica del sistema, presentado solo un error en la magnitud de las señales.

5.3 Discusión de resultados

Las observaciones obtenidas a partir de los resultados presentados se pueden resumir los siguientes puntos:

1. El primer punto a destacar, es la necesidad de contar con un número variado de operadores genéticos, lo anterior, debido a que esto le dará al proceso un mayor espacio de búsqueda y la velocidad de convergencia será mayor.
2. Un aspecto importante en este caso, fue también el de que las tasas de cruce y mutación fueron diferentes a las usuales. En primer lugar se utilizó una tasa de 65/35% para determinar si el nuevo elemento que se generaría, sería resultado de cruce o mutación respectivamente. Estas tasas se dieron principalmente, una vez que se determinaba que el

nuevo elemento saldría que una mutación, se verificaba de nueva cuenta la probabilidad de que cada parámetro del SLD fuera mutado, dicha probabilidad está dentro del rango [0.05, 0.1].

3. Un factor que ayudó a incrementar la velocidad de convergencia del sistema, fue el no tener todas las posibles reglas del SLD activas, sino cuando mucho la mitad. En este caso, para la primera generación la probabilidad de que una regla estuviera activa, fue de 25%, mientras que durante la aplicación del AG, la posibilidad de que una regla se activara era de 4%, mientras que la de que se desactivara una regla era de 12.5%.
4. En un inicio, se decidió utilizar una parte consecuente para las funciones $f(x)$ y otra para $g(x)$ de forma separada, lo que provocó que la mayoría de las reglas de la parte de $g(x)$ estuvieran inactivas, y el AG intentara hacer la identificación únicamente con las funciones de $f(x)$. Para solucionar esto, se optó porque las reglas tuvieran la siguiente estructura

$$\begin{aligned}
 & \text{IF } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\
 & \text{Then } xp_{ir} = a0_{ir} + a1_{ir} * x_1 + \dots + an_{ir} * x_n + \\
 & \quad b11_{ir} * u_1 + \dots + b1m_{ir} * u_m
 \end{aligned}$$

Y así evitar que el AG deseche las reglas asociadas a la parte de $g(x)$.

5. Los primeros SLD que se obtienen de la aplicación de los AG's, generan una aproximación por medio de

líneas segmentadas, las cuales siguen la forma de la señal, pero no las dinámicas rápidas (Lo anterior lo ilustramos en la figura 6.1, en donde la línea punteada es la salida del SLD. Como podemos ver en la gráfica, la estimación sigue únicamente la forma general de señal original [línea continua], pero no las dinámicas rápidas). Mejorar la situación anterior requiere en un inicio de varias generaciones, para evitarlo dejamos que el AG ubicara libremente las funciones membresía dentro del rango de operación de las variables de entrada.

6. Un aspecto que dió buenos resultados, fue el utilizar tasas de mutación de parámetros variables, esto fue utilizar tasas que estuvieran en función del error de estimación. En este proyecto comenzamos por utilizar tasas de 0.2 y conforme el error disminuía, también lo hacia la probabilidad de mutación, hasta llegar a 0.05.

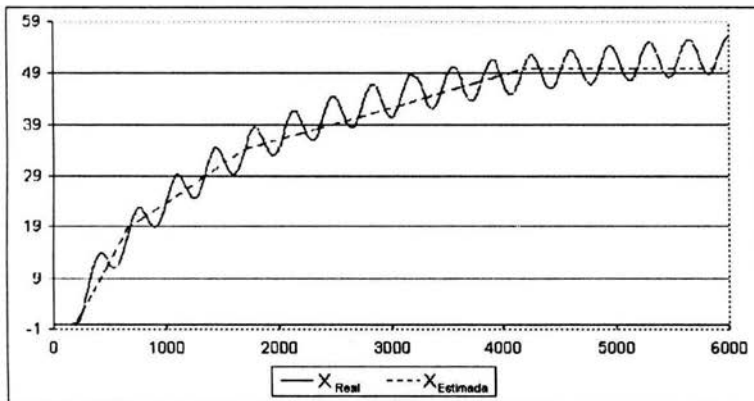


Fig. 6.1 Gráfica de una de las primeras aproximaciones obtenidas por el AG

7. Durante el trabajo se utilizaron dos modelos de aproximación, el primero denotado por la ecuación 6.1

$$\dot{\mathbf{X}} = \hat{\mathbf{f}}(\mathbf{x} | \boldsymbol{\theta}) + \hat{\mathbf{g}}(\mathbf{x} | \boldsymbol{\theta})\mathbf{u} \quad (6.1)$$

El segundo modelo fue

$$\dot{\mathbf{X}} = -\alpha\mathbf{e} + \hat{\mathbf{f}}(\mathbf{x} | \boldsymbol{\theta}) + \hat{\mathbf{g}}(\mathbf{x} | \boldsymbol{\theta})\mathbf{u} \quad (6.2)$$

A final se optó por elegir el modelo denotado por la ecuación (6.2), debido a que además de proporcionar una mayor estabilidad (ver capítulo 4), también, permite una mayor velocidad de convergencia.

CAPITULO VI

CONCLUSIONES

Podemos concluir en primer lugar que para el caso en los cuales obtener un modelo matemático por los métodos tradicionales no es posible, el recurrir a SLD, es una buena opción, al poder reproducir las dinámicas del sistema de interés.

En lo que se refiera a la combinación de los AG's y los MG's, es una muy buena opción, sobre todo cuando no se tiene idea de los rangos en los cuales se encuentra los parámetros del SLD y además por que en el caso del método gradiente cuando la aproximación inicial no es buena, es necesario que los parámetros tengan una correlación entre si, para evitar una divergencia en la aproximación.

En cuanto a los algoritmos propuestos dentro del AG, la forma en la cual podemos realizar su evaluación es por medio de si contribuyeron o no a mejorar la velocidad de convergencia del proceso de identificación. En el caso de los tres algoritmos propuestos (mutación aritmética, mutación escalar y el de agregar y disminuir reglas), mostraron un buen desempeño, sobre todo el algoritmo que agrega y disminuye reglas el cual incrementa la velocidad de convergencia de manera significativa, puesto que no tiene que evaluar situaciones irreales, además de simplificar los modelos obtenidos. Mientras que los algoritmos de

mutación, ambos presentan dos buenas alternativas para llevar a cabo la exploración de datos, debido a que el algoritmo de mutación aritmética permite modificar todo un elemento del SLD (regla o FM), y el de mutación escalar permite hacer desplazamientos tanto de las funciones membresía, como de los hiperplanos con los cuales se lleva a cabo las aproximaciones, si necesidad de modificar su forma.

Como continuación del trabajo realizado, existen tres temas principales que pueden desarrollarse de manera inmediata, estos temas son:

- Incrementar del tipo de sistemas a identificar.
- El uso de parámetros variables de los algoritmos de búsqueda
- El uso de estructuras diferentes en los SLD

A continuación con mayor detalle cada uno de estos tres puntos.

En lo referente a la identificación de sistemas no lineales, se puede extender el tipo de sistemas a identificar a una mayor variedad, como podrían ser los modelos cuya estructura es la mostrada en la ecuación (6.3).

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (6.3)$$

En la actualidad existen dos tipos principales de SLD definidos, que son los modelos de tipo Sugeno y tipo Mamdani, en donde como mencionamos en el capítulo 3, se diferencian por la forma en que manejan la parte

consecuente de la reglas *IF-THEN*. Uno de los puntos en los que se pueden llevar a cabo pruebas, es el de manejar estructuras diferentes, como podría ser el uso de funciones senoidales en lugar de polinomios o conjuntos difusos.

Dentro de los AG's, un aspectos en el cual es necesario realizar un análisis, es el correspondiente al uso de parámetros variables, como pueden ser las tazas de cruza y mutación, el número de individuos que son eliminados en cada generación, etc. Lo anterior lo mencionamos debido a que durante el desarrollo del trabajo, el utilizar tazas de mutación en función del error presento buenos resultados, permitiendo una mayor velocidad de convergencia.

Anexo A

Manual del programa

A continuación presentaremos una explicación de la forma en la cual programamos el sistema desarrollado. Comenzaremos por explicar la codificación, y posteriormente la forma de utilizar el programa.

A.1 Codificación

Con el propósito de crear un programa capaz de identificar sistemas no lineales de cualquier orden y número de entradas, se optó por utilizar programación dinámica para definir cada uno de los elementos que componen el SLD. Para entender de una la forma más clara como se hizo lo anterior, haremos uso de la figura A.1, en donde mostramos la estructura de datos correspondiente a un cromosoma. Para llevar a cabo la codificación se partió del uso de reglas cuya estructura es la siguiente:

$$\begin{aligned} & \text{IF } x_1 \text{ is } A_1 \text{ and } \dots \text{ and } x_n \text{ is } A_n \\ & \text{Then } xp_{ir} = a0_{ir} + a1_{ir} * x_1 + \dots + an_{ir} * x_n + \\ & \quad b1_{ir} * u_1 + \dots + bm_{ir} * u_m \end{aligned}$$

Donde A_i representa el conjunto difuso de la i -ésima variable entrada, y $a0_{ir}, \dots, an_{ir}, b1_{ir}, \dots, bm_{ir}$, son constantes reales asociadas a la r -ésima regla.

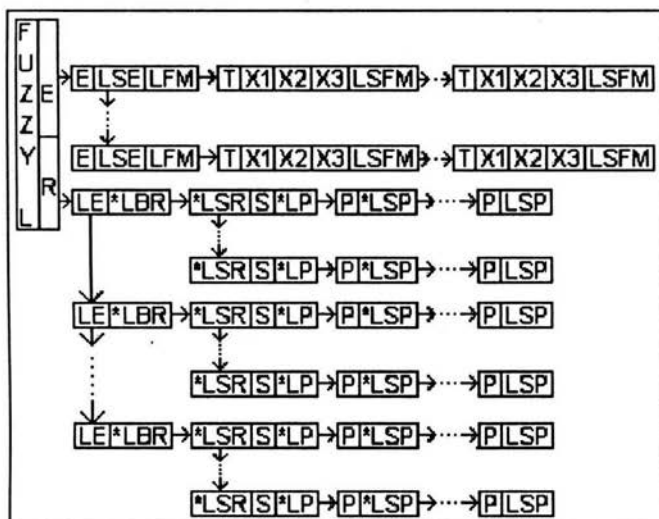


FIG. A.1 Estructura de datos cada uno de los sistemas difusos

A continuación se presenta la descripción de las abreviaturas utilizadas.

- E: Número entero que representa el estado al que pertenecen las funciones membresía.
- LSE: Liga al conjunto de funciones membresías del siguiente estado.
- LFM: Liga al conjunto de funciones membresía por estado
- T: Tipo de función membresía.
- x1, x2, x3: Cota mínima, centro y cota superior de una función membresía triangular.
- LSFM: Liga a la siguiente función membresía.
- LE: Liga a la base de reglas correspondiente al siguiente estado.
- LBR: liga a la base de reglas del estado.

- LSR: liga a la siguiente regla.
- S: singleton de $f(X)$
- LP: liga al siguiente parámetro.
- P: singleton correspondiente a cada una de las entradas.
- LSP: liga al siguiente parámetro.

Para la definición de los conjuntos difusos correspondientes a cada estado, asignamos a cada FM un nodo de la lista ligada. Este nodo como se podrá ver en la figura A.1 tiene cuatro campos. El primero, indica el tipo de FM que se tiene, los siguientes tres campos contiene los parámetros necesarios para definir la función triangular, descrita por la ecuación (A.1) y que se muestra en la figura A.2.

$$\mu_B(x) = \begin{cases} \frac{x-x_C}{x_D-x_C} & \text{si } x_I \leq x < x_C \\ \frac{x_D-x}{x_D-x_C} & \text{si } x_C \leq x < x_D \\ 0 & \text{Cualquier otro punto} \end{cases}$$

(A.1)

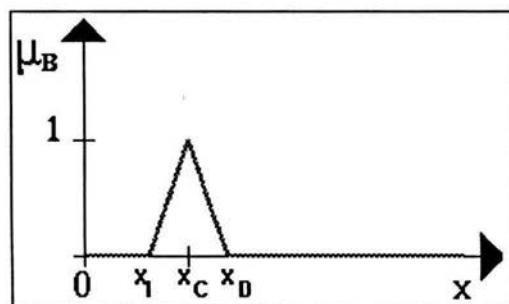


FIG. A.2 Representación gráfica de la FM utilizadas

En lo que se refiere a las reglas, lo único que guardamos en memoria son los parámetros de la parte consecuente, lo

anterior se debe a que las reglas se van revisando en orden, por lo cual se sabe que FM se debe evaluar en la parte antecedente.

A.2 Uso del programa

El uso del programa es sencillo, lo único que se necesita es entregar los datos que se piden al iniciar su ejecución, estos datos son:

- El número de estados
- El número de entradas
- La dirección y nombre del archivo en la cual se encuentran los datos de entrenamiento
- El número de datos de entrenamiento
- El tiempo de muestreo
- El número de funciones membresía que se desea utilizar
- La cantidad de generaciones que desea correr

En cuanto al archivo de entrada, éste debe ser un archivo tipo texto, en el cual los datos se encuentren en forma de columnas, donde las primeras columnas corresponden a la medidas de las entradas, y posteriormente las de los estados (los programas bloc de notas de Windows y edit de MS-DO pueden ser utilizados para realizar el archivo de entrada). En la figura A.3 mostramos del acomodo anterior, en donde vemos el archivo correspondiente a los datos de entrenamiento para el sistema de dos tanques que presentamos en el capítulo 5. En la figura A3, en la primera columna están los datos casociados a la entrada, y las dos siguientes columnas las medidas de los estados.

50.0000	0	0
50.0000	0.0000	0
50.0121	0.0029	0.0000
50.0241	0.0051	0.0000
50.0378	0.0080	0.0001
50.0702	0.0149	0.0002
50.1129	0.0239	0.0004
50.1694	0.0358	0.0008
50.2423	0.0510	0.0013
50.3427	0.0719	0.0021
50.4939	0.1031	0.0036
50.7323	0.1519	0.0063
51.1449	0.2356	0.0120
51.5947	0.3262	0.0193
52.0444	0.4162	0.0274
52.4940	0.5059	0.0362
52.9433	0.5954	0.0457
53.3924	0.6846	0.0558
53.8412	0.7741	0.0664
54.2897	0.8634	0.0774
54.7379	0.9528	0.0890

Fig. A.3 Archivo con datos de entrenamiento

Una vez que el programa ha terminado de correr, genera los siguientes archivos:

- SLD.TXT: Contiene los datos del sistema de lógica difusa con el que realizamos la estimación
- ALG_GEN.TXT: Presenta una simulación del SLD después de aplicado el AG.
- GRADIENTE.TXT: Presenta una simulación del SLD después de la aplicación del MG

- FITNESS.TXT: Contiene el comportamiento del error de estimación que se obtuvo por cada generación

Estos archivos se presentan en formato texto para facilitar su manipulación y se encuentran en la carpeta.

C:\sld

Anexo B

Notación y abreviaciones

A continuación, presentamos la principal notación utilizada en el presente trabajo.

n	Número de estados
m	Número de entradas
X_i	i -ésimo estado
A_{ij}	j -ésima FM del i -ésimo estado
X_{p_i}	Derivada del i -ésimo estado
r	r -ésima regla
θ	Vector de parámetros del SLD
W	Nivel de activación de cada regla

En el trabajo utilizamos varias abreviaturas, las cuales explicamos a continuación.

SLD	Sistema de Lógica Difusa
AG	Algoritmo Genético
MG	Método Gradiente
FM	Función Membresía
OG	Operador Genético
SBPA	Señal Binaria Pseudo Aleatoria

Bibliografia

- [1] J. Gary, J. David, Murray-Smith. "Nonlinear model structure identification using genetic programming", Control Engineering Practice (1998).
- [2] Chuen, C. 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] T. Takagi, M Sugeno. "Fuzzy identification of systems and its applications to modeling and control", IEEE Transactions on Systems, Man and Cybernetics, vol. 15, No 1 January / February 1985.
- [4] M. Setnes, H. Roubos. "GA fuzzy modeling and classification: Complexity and performance", IEEE Transaction on Fuzzy Systems, vol. 8, No 5, October 2000.
- [5] L. Wang. "Adaptive fuzzy systems and control", Prentice Hall, 1994.
- [6] D. Goldberg. "Genetics algorithms in search, optimization, and machine learning", Addison Wesley, 1989.
- [7] K. Passino, S Yurkovich. "Fuzzy control", Addison Wesley, 1998.