



UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

IDENTIFICACIÓN Y CONTROL
DE SISTEMAS USANDO
REDES NEURODIFUSAS

T E S I S
QUE PARA OBTENER EL TÍTULO DE
INGENIERO ELÉCTRICO ELECTRÓNICO
P R E S E N T A
EDGAR BERROSPE JUÁREZ



DIRECTOR DE TESIS

M. I. MARCOS ÁNGEL GONZÁLEZ OLVERA

MÉXICO, D. F. 2009

Jurado:

Presidente: M. I. Antonio Salvá Calleja

Vocal: M. I. Marcos Ángel González Olvera

Secretario: M. I. Ricardo Garibay Jiménez

1er Suplente: Dr. Gerardo Espinosa Pérez

2do Suplente: Dr. Yu Tang Xu

Esta tesis se realizó en el Laboratorio de Control del Posgrado de la Facultad de Ingeniería; con el apoyo del proyecto DGAPA-PAPIIT IN112908, a cargo del Dr. Gerardo Espinosa Pérez.

Tutor de tesis:

M. I. Marcos Ángel González Olvera

Dedico esta tesis a:

Mi familia, principalmente a mi mamá mi papá y mi hermana, por todas las cosas que es imposible enumerar aquí.

La UNAM, por haberme abierto sus puertas en mi adolescencia y haberme hecho una persona libre de espíritu y ávida de conocimiento.

Lorenzo Miranda, por haberme compartido su amistad y conocimiento con tanta generosidad.

Gabriel Alejandro Jaramillo, por haberme mostrado lo bella que puede ser esta profesión.

Marcos Ángel González, por hacer suyo este proyecto.

Gerardo Espinosa, por su interés en mi educación y por las facilidades otorgadas para la realización de este trabajo.

Diego López, por ocuparse de esa parte tan importante de mi formación, la cual va más allá de la parte musical.

Mis amigos, con los que compartí, entre otras, cosas las aulas y el tiempo en esta universidad.

Alejandra Icxih, por nuestro amor y nuestros proyectos.

Índice

1	Introducción	5
1.1	Motivación y Formulación del Problema	5
1.2	Objetivos	8
1.3	Contribuciones	10
1.4	Organización del Trabajo	10
2	Identificación y Control	13
2.1	Identificación	13
2.1.1	Identificación de un Sistema Lineal de Primer Orden a través de la Respuesta al Escalón	14
2.1.2	Método de los Mínimos Cuadrados	15
2.1.3	Método de los Mínimos Cuadrados Recursivo	18
2.1.4	Filtro de Kalman	20
2.1.5	Métodos Basados en el Gradiente	22
2.2	Control	23
2.2.1	Controladores PID	24
2.2.2	Control por Asignación de Polos	26
2.2.3	Control de Equivalencia Cierta	28

3	Redes Neuronales y Sistemas Difusos	33
3.1	Redes Neuronales	34
3.1.1	Neuronas	34
3.1.2	Perceptrón Multicapa	35
3.1.3	Entrenamiento	38
3.2	Sistemas Difusos	39
3.2.1	Estructura de un Sistema Difuso	39
3.2.2	Conjuntos Difusos	41
3.2.3	Operaciones con Conjuntos Difusos	43
3.2.4	Sistemas Difusos Tipo Mamdani	44
3.2.5	Sistemas Difusos Tipo Takagi-Sugeno	46
3.2.6	Desventajas del Control Difuso	48
3.3	Redes Neurodifusas	49
3.4	Redes Neurodifusas Recurrentes	51
3.5	Aproximadores Universales	52
4	Red Neurodifusa Recurrente Entrada-Salida	55
4.1	Estructura de la Red	55
4.2	Linealización de los parámetros antecedentes	59
4.3	Entrenamiento de la Red Usando el Filtro de Kalman	61
4.4	Entrenamiento de la Red Usando el Gradiente	62
4.5	Restricciones para los Parámetros	62
4.6	Algoritmo de Inicialización de Parámetros	65
4.7	Control de Equivalencia Cierta Usando la Red	66
5	Implementación	69
5.1	Equipo	69

<i>ÍNDICE</i>	3
5.1.1 Motor y Módulos	70
5.1.2 Tarjeta de Adquisición de Datos	73
5.1.3 Conexión del Motor con la Computadora	73
5.2 Identificación del Sistema Electromecánico	75
5.2.1 Identificación del Sistema Electromecánico Usando la Respuesta al Escalón	76
5.2.2 Identificación del Sistema Electromecánico Usando Mínimos Cuadra- dos	78
5.2.3 Identificación del Sistema Electromecánico Usando la Red Neu- rodifusa	80
5.2.4 Comparación entre Modelos	85
5.3 Control del Sistema Electromecánico	86
5.3.1 Control PI	89
5.3.2 Control Difuso Tipo Mamdani	89
5.3.3 Control Difuso Tipo Singleton	94
5.3.4 Control Difuso Tipo Takagi-Sugeno	96
5.3.5 Control por Asignación de Polos	98
5.3.6 Control de Equivalencia Cierta Usando la Red Neurodifusa . . .	101
5.3.7 Control de Equivalencia Cierta Usando la Red Neurodifusa Es- timando Parámetros en Línea	106
5.3.8 Comparación entre Esquemas de Control	112
6 Conclusiones	117
A Programas	121

Capítulo 1

Introducción

1.1 Motivación y Formulación del Problema

Son diversas las ramas del conocimiento donde se hace patente la necesidad de contar con modelos matemáticos que describan el comportamiento de los sistemas analizados. Este es el caso de la ingeniería de control, en donde los modelos matemáticos son de vital importancia en la descripción y análisis de los sistemas; facilitando el diseño de esquemas de control .

El análisis y diseño de esquemas de control para sistemas físicos puede llevarse a cabo usando métodos empíricos, del tipo ensayo y error. Para controlar a un sistema, es posible aplicar diversas señales a las entradas de los sistemas y medir sus respuestas. Si el desempeño no es adecuado, algunos de los parámetros pueden ser ajustados o algún controlador puede ser conectado al sistema. Este tipo de métodos se vuelven inoperantes si los sistemas son muy complejos, peligrosos, costosos o simplemente si no es posible o adecuado experimentar con ellos de esta manera. Es en este momento cuando se vuelve necesario contar con modelos matemáticos y métodos analíticos,

que nos permitan obtener un entendimiento más profundo de los sistemas y lograr un diseño más exitoso.

Los modelos hacen posible predecir y simular sistemas, además de ser útiles para abordar problemas tales como el diseño de esquemas de control y la detección de fallas. La capacidad del modelo para describir al sistema determina la calidad de la solución del problema. Aquí es donde surge la necesidad de contar con metodologías de modelado e identificación de sistemas.

Un mismo sistema puede tener varios modelos. Un modelo puede ser más conveniente que otro dependiendo del problema específico que se quiera atacar. Por ejemplo, un transistor se modela en forma diferente cuando opera a alta o baja frecuencia. Un transbordador espacial puede ser modelado como una partícula cuando se analiza su trayectoria o como un cuerpo rígido si se analizan sus maniobras. Por otra parte, las representaciones en el espacio de estados son más convenientes que las representaciones mediante la función de transferencia, cuando se quiere analizar la observabilidad y la controlabilidad.

Los modelos se pueden dividir en dos tipos: *lineales y no lineales*. Los modelos lineales son aquellos que cumplen con el principio de superposición (homogeneidad y aditividad). El principio de superposición establece que la respuesta de un sistema a la suma de señales será igual a la suma de las respuestas de cada una de las señales.

Al obtener un modelo se debe establecer un compromiso entre simplicidad y precisión. Como primer acercamiento al estudio de un sistema es conveniente considerar un modelo lineal. Si el efecto que las propiedades físicas ignoradas tienen sobre la respuesta es pequeño, se tendrá un buen acuerdo entre los resultados del análisis del modelo matemático y los resultados del estudio experimental del sistema físico.

Si los resultados arrojados por el modelo lineal no son satisfactorios, la siguiente

decisión será considerar un modelo no lineal. Al migrar al modelo de un modelo lineal a uno no lineal, puede ocurrir que el modelo no lineal, si no es adecuadamente elegido, se desempeñe peor que el modelo lineal. Una buena estrategia para evitar este efecto indeseado es utilizar un modelo no lineal que contenga un modelo lineal como caso especial (ver [7]).

Dado que no existe una única estructura para representar la totalidad de los modelos no lineales, surge la necesidad de plantear estructuras y modelos que puedan ser utilizados para estudiar un número más grande de sistemas no lineales. Entre las diversas estructuras existentes para la identificación de sistemas no lineales se encuentran los modelos polinomiales, las redes neuronales, los sistemas difusos y las redes neurodifusas.

De la misma forma en que los modelos pueden tener varias estructuras, también existen varios métodos para estimar los parámetros de dichos modelos. El método de estimación que ha de usarse dependerá en cierta medida de la estructura del modelo. Existen relaciones cercanas (algunas de carácter histórico) entre ciertas estructuras de modelos y ciertos métodos de estimación, este es el caso de las redes neuronales y el algoritmo de retropropagación.

Los métodos de estimación existentes pueden clasificarse en lineales y no lineales. Los métodos lineales sirven para estimar los parámetros de modelos que se pueden representar de manera *lineal con respecto a sus parámetros*, los métodos no lineales no requieren que el modelo sea lineal con respecto a sus parámetros. El método de los mínimos cuadrados y el método de los mínimos cuadrados recursivo son ejemplos de métodos de optimización lineales. Por otro lado el algoritmo de retropropagación y el método de Newton son ejemplos de métodos de optimización no lineales.

Debido a que los métodos lineales han sido más ampliamente estudiados y son

más fáciles de aplicar que los métodos no lineales, es recomendable considerar un modelo que sea lineal con respecto a sus parámetros antes que un modelo que no lo sea.

Las redes neuronales y los sistemas difusos son, tal y como lo demuestra Cybenko [24], aproximadores universales. Esto significa que son capaces de aproximar a cualquier función suave y continua con un grado de precisión arbitrario, dentro de un intervalo cerrado. Por esta razón han sido usados para modelar y diseñar sistemas de control.

Las redes neurodifusas nacieron de la combinación de los sistemas difusos y las redes neuronales. Este tipo de estructuras toman ventaja de la capacidad de los sistemas difusos de emular la manera *difusa* en que los humanos pensamos y de capacidad de aprendizaje que poseen las redes neuronales. Las redes neurodifusas recurrentes son redes neuronales que cuentan con lazos de realimentación en sus capas. Este tipo de redes son útiles cuando se trata de modelar e identificar sistemas dinámicos [7].

La red neurodifusa recurrente propuesta por González-Olvera y Tang en [2] y [3] ha probado ser útil identificando al sistema no lineal de referencia propuesto por Narendra y Parthasarathy. Esta red contiene a varios sistemas lineales que son combinados con ayuda de la lógica difusa. La estructura de la red es tal que construye un modelo no lineal de un sistema a partir de distintos modelos lineales del mismo sistema.

1.2 Objetivos

El objetivo principal del actual trabajo es la identificación de un sistema físico usando la red neurodifusa recurrente entrada-salida propuesta González-Olvera y Tang en [2] y [3], además de la implementación de un esquema de control de equivalencia

cierta basado en el modelo obtenido por medio de la identificación.

También se plantean dos objetivos secundarios. El primero es la obtención de otros modelos para el sistema. Los otros modelos que fueron obtenidos son dos modelos lineales uno de primer orden en tiempo continuo y otro de tercer orden en tiempo discreto. El modelo lineal de primer orden en tiempo continuo es obtenido para evidenciar la necesidad de contar con un modelo que identifique mejor al sistema. El modelo lineal de tercer orden en tiempo discreto es obtenido a manera de preludio, para que la identificación por medio red neurodifusa resulte más sencilla en su interpretación e implementación.

El segundo objetivo secundario es la implementación de otros esquemas de control. En un primer momento la implementación de un control PID surgió de la necesidad de tener un punto de referencia desde el cual evaluar el desempeño del control de equivalencia cierta basado en la red neurodifusa. Una vez implementados los dos esquemas de control anteriores se observó que los conceptos e instrumentos que ya se tenían disponibles podían ser aprovechados para implementar otros esquemas de control. Los otros esquemas de control que fueron implementados son un control PI, un control difuso tipo Mamdani, un control difuso tipo singleton, un control difuso tipo singleton, un control difuso tipo Takagi-Sugeno y un control por asignación de polos.

El sistema físico que fue escogido para la realización de los experimentos es el motor de corriente directa MT150F (desarrollado por Feedback con fines didácticos). Para la implementación del los esquemas de control y del sistema de medición fue necesario el uso de otros módulos del sistema MS150, una tarjeta de adquisición de datos NI USB-6009, una computadora personal y los paquetes de computo Visual Basic y Matlab.

1.3 Contribuciones

Las contribución inicial de este trabajo es la validación experimental la red neurodifusa propuesta en [2] y [3]. Durante el desarrollo de este trabajo surgieron otros problemas, cuya solución condujo a extender lo planteado en [2] y [3].

Se propone un nuevo algoritmo para inicializar los parámetros de la red neurodifusa. Este algoritmo hace uso de la interpretación de la red como la combinación de varios modelos lineales. El algoritmo indica que primero han de encontrarse los modelos lineales que serán ponderados por la red neurodifusa.

Durante el entrenamiento de la red se observó la necesidad de implementar restricciones a algunos de los parámetros, llamados antecedentes, de la red, idea antes planteada en [3]. Por otra parte, cuando se implementó un control de equivalencia cierta basado en la red neurodifusa, se hizo patente la necesidad de implementar restricciones a otros de los parámetros, llamados consecuentes. Por tal motivo, la estructura de las restricciones planteadas en [3] se modificó para poder ser aplicadas a los parámetros consecuentes.

En este trabajo, también se propone e implementa un método para estimar los parámetros de la red en línea. Este método es de gran utilidad cuando se trata con sistemas variantes en el tiempo.

1.4 Organización del Trabajo

El presente trabajo está organizado en cinco capítulos. En el segundo capítulo se tratan los temas relacionados con la identificación y con el control de sistemas. En el tercer capítulo se realiza una exposición de los conceptos básicos de las redes neuronales, los sistemas difusos y las redes neurodifusas. Los conceptos expuestos en

el tercer capítulo explican la manera en que operan los controles difusos, además son necesarios para entender la estructura y el funcionamiento de la red neurodifusa propuesta por González-Olvera y Yu Tang en [2] y [3].

En el cuarto capítulo presenta la estructura de la red neurodifusa recurrente entrada-salida que modelará al sistema electromecánico, del entrenamiento de la red y del esquema de control basado en la misma. La identificación y control del sistema electromecánico y los resultados experimentales se reportan en el quinto capítulo, junto con una explicación y justificación del software y el hardware utilizado. Finalmente el sexto capítulo se presentan las conclusiones finales del trabajo y se esboza el camino a seguir en futuros trabajos acerca del mismo tema.

Capítulo 2

Identificación y Control

Este capítulo está dividido en dos secciones. En la primera sección se plantea el problema de la identificación de sistemas, se plantea un método para identificar sistemas lineales usando su respuesta al escalón, además se estudian los métodos para estimar parámetros de los mínimos cuadrados, de los mínimos cuadrados recursivo, el filtro de Kalman y los métodos basados en el gradiente. La segunda sección esta dedicada al control de sistemas, este tema no se aborda de manera profunda y sólo se explica la estructura de los controladores PID, por asignación de polos y de equivalencia cierta.

2.1 Identificación

La identificación es el proceso de construir un modelo matemático de un sistema g definido por un mapeo

$$y = g(\mathbf{x})$$

donde $\mathbf{x} = [x_1, x_2, \dots, x_m]$ es el vector de entradas y y es la salida del sistema; mediante el empleo de datos del comportamiento del sistema obtenidos de forma experimental.

El objetivo es encontrar f tal que

$$g(\mathbf{x}) = f(\mathbf{x}, \theta) + e(\mathbf{x}) \approx f(\mathbf{x}, \theta), \quad (2.1)$$

donde $\mathbf{x} = [x_1, x_2, \dots, x_m]$ es el vector de entradas, $\theta = [\check{\theta}_1, \check{\theta}_2, \dots, \check{\theta}_n]^T$ es un vector de parámetros que puede ser ajustado para que el error de aproximación $e(\mathbf{x})$ sea lo mas pequeño posible, de acuerdo con los requerimientos del problema.

El hecho de que el modelo f sea un modelo MISO (por las siglas en inglés de *multiple-entrada única-salida*) no representa pérdida de generalidad, ya que un modelo MIMO (por las siglas en inglés de *multiple-entrada multiple-salida*) puede ser descompuesto en varios modelos MISO.

El primer paso en el proceso de identificación es elegir la estructura del modelo. Como ya se menciona en la sección 1.1, existe una amplia gama de estructuras que podemos escoger dependiendo de nuestros objetivos y del conocimiento que tengamos del sistema. Cuando se ha elegido la estructura del modelo, el siguiente paso es elegir el método con el cual se ajustarán sus parámetros.

2.1.1 Identificación de un Sistema Lineal de Primer Orden a través de la Respuesta al Escalón

Cuando es posible identificar a un sistema con un modelo lineal de primer orden, podemos obtener los parámetros del modelo midiendo y analizando la respuesta del sistema cuando es excitando con una entrada escalón.

La función de transferencia de un sistema de primer orden en el dominio s es

$$G(s) = \frac{k}{\tau s + 1} \quad . \quad (2.2)$$

La respuesta en el tiempo al escalón unitario del sistema anterior es

$$y(t) = k(1 - e^{-t/\tau}), \quad \text{para } t \geq 0. \quad (2.3)$$

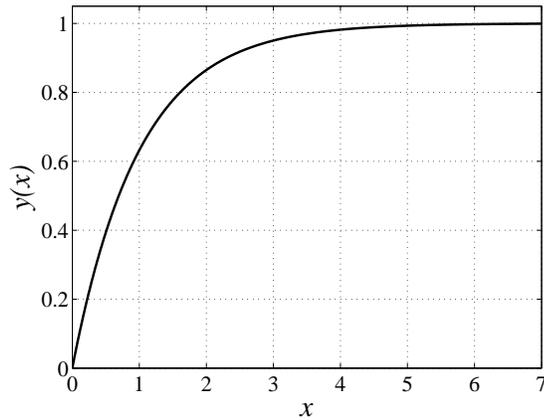


Figura 2.1: Respuesta al escalón unitario de un sistema de primer orden con $k = 1$ y $\tau = 1$

cuya representación gráfica se muestra en la figura 2.1.

El modelo anterior cuenta con dos parámetros que pueden ser ajustados, la ganancia k y la constante de tiempo τ . Cuando la entrada es un escalón unitario la salida en estado estable es $y(\infty) = k$ y τ es el tiempo para el cual la salida ha alcanzado el 63.2% de su valor final, $y(\tau) = 0.632k$.

Un método similar para la obtención de modelos de mayor orden es presentado en [23].

2.1.2 Método de los Mínimos Cuadrados

Con el método anterior sólo es posible determinar dos parámetros. El método de los mínimos cuadrados es un método de estimación lineal, útil para tratar con modelos que contienen una mayor cantidad de parámetros. Para poder aplicar este algún método de estimación lineal es necesario que el modelo f , escogido para identificar

al sistema g , sea lineal con respecto a sus parámetros, es decir

$$\hat{y} = f(\mathbf{x}, \theta) = \phi(\mathbf{x})\theta, \quad (2.4)$$

donde $\phi(x) = [\check{\phi}_1(x), \check{\phi}_2(x), \dots, \check{\phi}_n(x)]$ y $\theta = [\check{\theta}_1, \check{\theta}_2, \dots, \check{\theta}_n]^T$.

El vector ϕ se llama *regresor* y θ_i es un vector de parámetros. Es importante notar que las funciones $\check{\phi}_i(\mathbf{x})$ son funciones conocidas de las entradas x_i y pueden ser funciones no lineales.

Existen varios modelos que se ajustan a los requerimientos de la ecuación 2.35, por ejemplo, los modelos polinomiales

$$\hat{y} = c_0 + c_1x + c_2x^2 + \dots + c_nx^n, \quad (2.5)$$

con $\phi = [1, x, x^2, \dots, x^n]$ y $\theta = [c_0, c_1, c_2, \dots, c_n]^T$

y las ecuaciones en diferencias lineales

$$y_k = b_1x_{k-1} + \dots + b_px_{k-p} + a_1y_{k-1} + \dots + a_qy_{k-q}, \quad (2.6)$$

con $\phi = [x_{k-1}, \dots, x_{k-p}, y_{k-1}, \dots, y_{k-q}]$ y $\theta = [b_1, \dots, b_p, a_1, \dots, a_q]^T$.

Asumimos que hemos obtenido N pares de datos entrada-salida realizando mediciones en el sistema g , donde el i ésimo par de datos es $\{\mathbf{x}^i, y^i\}$, donde $y^i = g(\mathbf{x}^i)$.

Para continuar con la explicación se define el error de identificación como

$$e_i = y^i - \phi(\mathbf{x}^i)\theta \quad (2.7)$$

también se define

$$Y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}, \quad \Phi = \begin{bmatrix} \phi(x^1) \\ \phi(x^2) \\ \vdots \\ \phi(x^N) \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_N \end{bmatrix}, \quad E = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}. \quad (2.8)$$

Por lo tanto el vector de errores de aproximación E se puede escribir como

$$E = Y - \Phi\theta. \quad (2.9)$$

Se escoge una función, a la que llamaremos *función de costo*

$$V(\theta) = \frac{1}{2}E^T E, \quad (2.10)$$

para ser una medida de la calidad de la identificación. Necesitamos elegir una θ que minimice a $V(\theta)$. Dado que $V(\theta)$ es convexa con respecto a θ , un mínimo local es un mínimo global.

Si se obtiene la derivada de $V(\theta)$ con respecto de θ y se iguala a cero, se obtiene una ecuación que permite encontrar a $\hat{\theta}$, el valor de θ que minimiza a $V(\theta)$.

Otra forma de encontrar $\hat{\theta}$ es notar que

$$2V = E^T E = Y^T Y - Y^T \Phi \theta - \theta^T \Phi^T Y + \theta^T \Phi^T \Phi \theta. \quad (2.11)$$

Si $\Phi^T \Phi$ es invertible, se puede sumar y restar $Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y$ a la ecuación (2.11) para obtener

$$\begin{aligned} 2V &= Y^T Y - Y^T \Phi \theta - \theta^T \Phi^T Y + \theta^T \Phi^T \Phi \theta \\ &+ Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y - Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y, \end{aligned} \quad (2.12)$$

entonces

$$\begin{aligned} 2V &= Y^T (I - \Phi (\Phi^T \Phi)^{-1} \Phi^T) Y \\ &+ (\theta - (\Phi^T \Phi)^{-1} \Phi^T Y)^T \Phi^T \Phi (\theta - (\Phi^T \Phi)^{-1} \Phi^T Y). \end{aligned} \quad (2.13)$$

Es importante notar que el primer término de la ecuación (2.13) no depende de θ y que el menor valor que puede tomar el segundo término es cero. El valor de θ que logra que el segundo término sea cero y que por lo tanto minimiza a V es

$$\hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T Y. \quad (2.14)$$

2.1.3 Método de los Mínimos Cuadrados Recursivo

El método de los mínimos cuadrados ha probado ser exitoso en varias aplicaciones, pero se vuelve inoperante cuando el número N de pares de datos entrada-salida $\{\mathbf{x}^i, y^i\}$ es grande, ya que las dimensiones de Φ y de Y dependen de N y hacen que el cálculo de $(\Phi^T \Phi)^{-1}$ sea muy pesado computacionalmente. El método de los mínimos cuadrados recursivo nos permite actualizar $\hat{\theta}$ cada vez que tenemos un nuevo par de datos entrada-salida sin la necesidad de usar todos los datos anteriores y sin calcular $(\Phi^T \Phi)^{-1}$ de manera explícita.

Consideremos la matriz de $n \times n$

$$P_k = (\Phi^T \Phi)^{-1} = \left(\sum_{i=1}^k \phi_i^T \phi_i \right)^{-1}, \quad (2.15)$$

donde $0 \leq i \leq k$ y k es un índice que aumenta cada vez que tenemos un nuevo par de datos.

De acuerdo con la ecuación 2.15 la inversa de P_k es

$$P_k^{-1} = \sum_{i=1}^{k-1} \phi_i^T \phi_i + \phi_k^T \phi_k, \quad (2.16)$$

entonces

$$P_k^{-1} = P_{k-1}^{-1} + \phi_k^T \phi_k. \quad (2.17)$$

Combinando las ecuaciones (2.14) y (2.15) obtenemos

$$\begin{aligned} \hat{\theta}_k &= (\Phi^T \Phi)^{-1} \Phi^T Y, \\ &= P_k \left(\sum_{i=1}^k \phi_i^T y^i \right), \\ &= P_k \left(\sum_{i=1}^{k-1} \phi_i^T y^i + \phi_k^T y^k \right). \end{aligned} \quad (2.18)$$

Por lo tanto

$$\hat{\theta}_{k-1} = P_{k-1} \sum_{i=1}^{k-1} \phi_i^T y^i, \quad (2.19)$$

$$P_{k-1}^{-1} \hat{\theta}_{k-1} = \sum_{i=1}^{k-1} \phi_i^T y^i. \quad (2.20)$$

Despejando P_{k-1}^{-1} en la ecuación (2.17), y sustituyéndolo en (2.20) obtenemos

$$\sum_{i=1}^{k-1} \phi_i^T y^i = (P_k^{-1} - \phi_k^T \phi_k) \hat{\theta}_{k-1}. \quad (2.21)$$

Sustituyendo el resultado anterior en la ecuación (2.18)

$$\begin{aligned} \hat{\theta}_k &= P_k (P_k^{-1} - \phi_k^T \phi_k) \hat{\theta}_{k-1} + P_k \phi_k^T y^k, \\ &= \hat{\theta}_{k-1} - P_k \phi_k^T \phi_k \hat{\theta}_{k-1} + P_k \phi_k^T y^k, \\ &= \hat{\theta}_{k-1} + P_k \phi_k^T (y^k - \phi_k \hat{\theta}_{k-1}). \end{aligned} \quad (2.22)$$

Esta ecuación nos permite calcular $\hat{\theta}_k$ a partir de P_k , $\hat{\theta}_{k-1}$ y el último par de datos entrada-salida $\{\mathbf{x}^k, y^k\}$. Es importante notar que $y^k - \phi_k \hat{\theta}_{k-1}$ es el error que se tiene al calcular y^k usando $\hat{\theta}_{k-1}$. La ecuación (2.17) nos provee de un método para calcular P_k pero requiere del cálculo de la inversa de una matriz cada vez que se obtiene un nuevo par de datos entrada-salida. Para evitar esto se emplea la relación

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}. \quad (2.23)$$

De la ecuación (2.17) sabemos que

$$P_k = (P_{k-1}^{-1} + \phi_k^T \phi_k)^{-1}, \quad (2.24)$$

por lo tanto, si usamos la ecuación (2.23) considerando que $A = P_{k-1}^{-1}$, $B = \phi_k^T$, $C = I$ y $D = \phi_k$ obtenemos

$$P_k = P_{k-1} - P_{k-1} \phi_k^T (I + \phi_k P_{k-1} \phi_k^T)^{-1} \phi_k P_{k-1}, \quad (2.25)$$

es importante notar que el término $I + \phi_k P_{k-1} \phi_k^T$ es un escalar.

Las ecuaciones (2.25) y (2.22) constituyen el método de los mínimos cuadrados recursivo

$$\hat{\theta}_k = \hat{\theta}_{k-1} + P_k \phi_k^T (y^k - \phi_k \hat{\theta}_{k-1}), \quad (2.26)$$

$$P_k = P_{k-1} - \frac{P_{k-1}\phi_k^T\phi_k P_{k-1}}{1 + \phi_k P_{k-1}\phi_k^T}. \quad (2.27)$$

Este método da el mismo peso a los datos viejos que a los datos nuevos. En aplicaciones con sistemas variantes en el tiempo en las que suponemos que los parámetros $\hat{\theta}$ varían en el tiempo es conveniente dar mayor peso a los datos más recientes. Este objetivo se puede alcanzar al elegir a la función de costo de la siguiente manera

$$V(\theta) = \frac{1}{2} \sum_{i=1}^k \lambda^{k-i} (y^i - \phi(x^i)\theta), \quad 0 < \lambda \leq 1, \quad (2.28)$$

donde λ se llama *factor de olvido*.

Usando un proceso similar al descrito anteriormente llegamos a que el algoritmo de estimación se estará dado por

$$\hat{\theta}_k = \hat{\theta}_{k-1} + P_k \phi_k^T (y^k - \phi_k \hat{\theta}_{k-1}), \quad (2.29)$$

$$P_k = \frac{1}{\lambda} \left(P_{k-1} - \frac{P_{k-1}\phi_k^T\phi_k P_{k-1}}{\lambda + \phi_k P_{k-1}\phi_k^T} \right). \quad (2.30)$$

Un valor pequeño de λ nos permite estimar rápidamente parámetros que varían con el tiempo, pero hace que el proceso sea más sensible a las perturbaciones.

2.1.4 Filtro de Kalman

El filtro de Kalman es un estimador de estados que es útil cuando de cuenta con mediciones que contienen ruido blanco gaussiano. El filtro de Kalman también puede ser usado para estimar los parámetros de un modelo como el de la ecuación (2.35), tal y como se verá a continuación.

Consideremos el siguiente modelo en espacio de estados en tiempo discreto.

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{G}\mathbf{x}_k + \mathbf{H}\mathbf{u}_k + \omega_k, \\ y_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \gamma_k, \end{aligned} \quad (2.31)$$

donde ω_k y γ_k son ruido blanco gaussiano y además

$$\begin{aligned} E[\omega_k] &= 0, & E[\omega\omega^T] &= R_1, \\ E[\gamma_k] &= 0, & E[\gamma\gamma^T] &= R_2, \\ E[x_0x_0^T] &= P_0, \end{aligned} \quad (2.32)$$

el operador $E[\cdot]$ es la esperanza matemática y $E[zz^T]$ es la varianza de z .

Asumimos que los datos y_k y u_k contienen ruido blanco gaussiano. El problema de la estimación óptima del estado \mathbf{x}_k basada en los datos y_k y u_k y el conocimiento del modelo de la ecuación (2.31) puede ser resuelto minimizando

$$J(\hat{\mathbf{x}}_k) = E[(\hat{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1})^2]. \quad (2.33)$$

El filtro de Kalman que predice el estado $\hat{\mathbf{x}}_k$ a partir de los datos en tiempo k es

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} &= \mathbf{G}\hat{\mathbf{x}}_k + \mathbf{H}u_k + K_k(y_k - \mathbf{C}\hat{\mathbf{x}}_k), \\ K_k &= \mathbf{G}P_k\mathbf{C}^T(R_2 + \mathbf{C}P_k\mathbf{C}^T)^{-1}, \\ P_{k+1} &= \mathbf{G}P_k\mathbf{G}^T + R_1 - \mathbf{G}P_k\mathbf{C}^T(R_2 + \mathbf{C}P_k\mathbf{C}^T)^{-1}\mathbf{C}P_k\mathbf{G}^T, \end{aligned} \quad (2.34)$$

la cual es una expresión recursiva donde el estado estimado es ajustado cada vez que un nuevo par de datos entrada salida está disponible. El filtro de Kalman minimiza el criterio de la ecuación (2.33).

Consideremos el siguiente modelo

$$\begin{aligned} \theta_{k+1} &= \theta_k + \omega_k, \\ \hat{y}_k &= \phi_k\theta_k + \gamma_k, \end{aligned} \quad (2.35)$$

que es similar al de la ecuación (2.35), solo que en este se toman en cuenta el efecto de los ruidos γ_k y ω_k , en el caso ideal ambos ruidos son cero.

Entonces el problema de la estimar los parámetros θ se puede resolver usando el

filtro de Kalman de la siguiente manera

$$\begin{aligned}\hat{\theta}_{k+1} &= \hat{\theta}_k + K_k(y_k - \phi_k \hat{\theta}_k), \\ K_k &= \frac{P_k \phi_k^T}{R_2 + \phi_k P_k \phi_k^T}, \\ P_{k+1} &= P_k + R_1 - \frac{P_k \phi_k^T \phi_k P_k}{R_2 + \phi_k P_k \phi_k^T}.\end{aligned}\tag{2.36}$$

Es importante notar que si $R_1 = 0$ y $R_2 = \lambda$, entonces se recuperan las ecuaciones (4.29) y (4.27) del método de los mínimos cuadrados, las cuales fueron deducidas al final de la sección anterior.

2.1.5 Métodos Basados en el Gradiente

Los métodos de estimación basados en el gradiente son los más comunes dentro de los métodos de estimación no lineales (no requieren que el modelo sea lineal con respecto a sus parámetros). En los métodos basados en el gradiente los parámetros son actualizados con base en a la información de las derivadas parciales de la función de costo con respecto a los parámetros (gradiente de la función de costo), con el objetivo de llevar a la función de costo a un mínimo local.

Una función de costo puede ser

$$V_k(\theta) = \frac{1}{2}(\hat{y}_k - y_k)^2,\tag{2.37}$$

y el gradiente es

$$g = \frac{\partial V(\theta)}{\partial \theta}.\tag{2.38}$$

El gradiente siempre apunta en la dirección de mayor aumento de V . Por lo tanto θ evoluciona en la dirección en la que V disminuye, tal y como se muestra en la figura 2.2. En este caso el paraboloide mostrado es la gráfica de la función escalar $z = x^2 + y^2$, por lo tanto el gradiente de z es un vector de dos elementos ($\nabla z \in \mathbb{R}^2$), si avanzamos

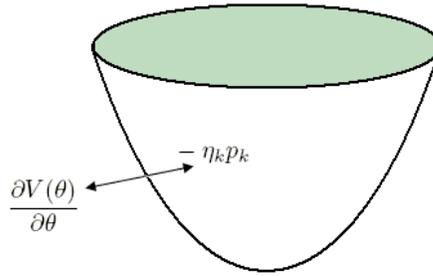


Figura 2.2: Paraboloide. Función escalar de x y y

en la dirección contraria a ∇z , encontraremos dos nuevos valores de x y y para los cuales z tendrá un menor valor.

La forma de actualizar los parámetros es la siguiente

$$\theta_{k+1} = \theta_k - \eta_k p_k, \quad p_k = R_k g_k, \quad (2.39)$$

donde R_k es una matriz que rota y escala al gradiente.

Cuando el sistema es no lineal pueden existir varios mínimos locales para la función de costo. Si inicialmente θ está situado cerca de algún mínimo local, será difícil que θ alcance otro valor que haga que el modelo tenga mejor desempeño.

Existen diversos métodos basados en el gradiente los cuales se diferencian en la elección de η y R . En el método del *descenso en la mayor inclinación* η es una constante y $R = I$.

2.2 Control

El objetivo de la teoría de control es forzar a los sistemas a comportarse de la manera deseada. La forma de lograr que la salida de un sistema, al que llamaremos *planta*, se comporte de la manera deseada es manipular su señal entrada. El conocimiento de un modelo matemático del sistema y el uso de la realimentación son dos elementos

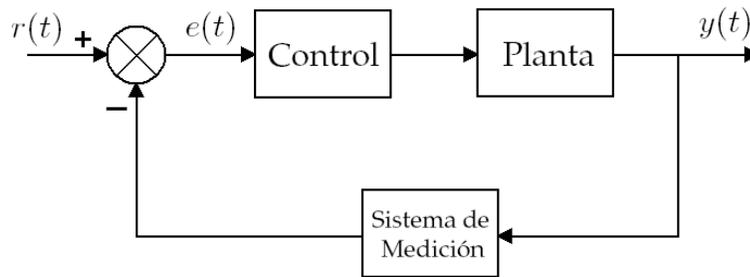


Figura 2.3: Diagrama de bloques de un sistema con control en lazo cerrado

que pueden ser de gran utilidad en el diseño de esquemas de control.

Los sistemas de control en lazo cerrado son sistemas donde la señal de entrada a la planta es función de la señal de salida y una señal de referencia. La figura 2.3 muestra un sistema de lazo cerrado donde la salida $y(t)$ es obtenida, por medio de un sistema de medición, y comparada con una referencia $r(t)$ para obtener una señal de error $e(t)$. La señal de error se introduce al bloque *control*, el cual conectado en cascada con la planta. El objetivo de este sistema es que la salida $y(t)$ tenga un comportamiento lo más cercano posible a la señal de referencia.

El bloque *control* de la figura 2.3 puede ser ocupado por diversos tipos de sistemas como, por ejemplo, los controladores PID o los sistemas difusos descritos en las secciones 3.2.4 y 3.2.5.

2.2.1 Controladores PID

Los controladores Proporcionales-Integrales-Derivativos o PID son ampliamente usados y han demostrado ser de gran utilidad en el diseño de sistemas de control. El éxito de este tipo de controles estriba en que se pueden utilizar de forma casi general en un gran número de problemas de control.

Los controladores PID se conectan como se muestra en la figura 2.3, por lo que su entrada es la señal de error; la salida del controlador se llama comúnmente señal de control. Los controladores PID combinan una acción proporcional, una acción integral y una acción derivativa. La acción proporcional produce señales de control proporcionales a la señal de error. Si la salida del sistema se encuentra lejos de la referencia, la señal de error será grande y la acción proporcional producirá un señal de control grande, la cual producirá una disminución en la señal de error. La acción integral produce señales de control proporcionales a la integral de la señal de error, este tipo de señales de control son de utilidad para la eliminación de errores en estado estable. Finalmente la acción derivativa produce señales de control proporcionales a la derivada de la señal de error, este tipo de señales de control son de utilidad en la eliminación del sobrepaso y las oscilaciones.

La función de transferencia de un controlador PID es la siguiente

$$\begin{aligned}
 G_c(s) &= K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right), \\
 &= K_p + \frac{K_i}{s} + K_d s, \\
 &= \frac{K_d s^2 + K_p s + K_i}{s},
 \end{aligned} \tag{2.40}$$

donde K_p se llama constante proporcional, K_i se llama constante integral, K_d se llama constante derivativa, τ_i se llama tiempo integral y τ_d se llama tiempo derivativo.

El tiempo integral es el tiempo para el cual la parte de la señal de control debida a la acción proporcional es igual a la parte de la señal de control debida a la acción integral, cuando la señal de entrada al controlador es constante. El tiempo derivativo es el tiempo para el cual la parte de la señal de control debida a la acción proporcional es igual a la parte de la señal de control debida a la acción derivativa, cuando la señal de entrada al controlador tiene una derivada constante.

El diseño de un controlador PID para un determinado sistema puede seguir los

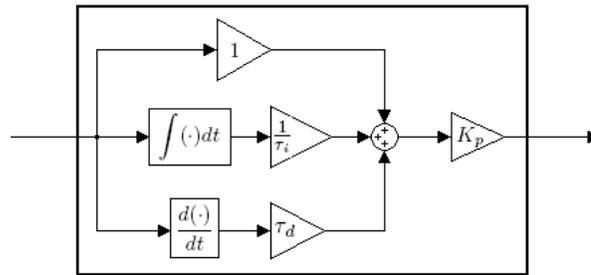


Figura 2.4: Diagrama de bloques de un controlador PID

siguientes pasos:

1. Obtener la respuesta en lazo cerrado del sistema y determinar que necesita ser mejorado.
2. Hacer $K_i = K_d = 0$ y aumentar K_p para mejorar el tiempo de subida.
3. Aumentar K_d para reducir el sobrepaso.
4. Aumentar K_i para eliminar el error en estado estable.

Existen otros métodos para sintonizar controladores PID, como el método de Ziegler-Nichols. Por supuesto también son de utilidad las técnicas del lugar geométrico de las raíces y del análisis de la respuesta en frecuencia.

2.2.2 Control por Asignación de Polos

Si el sistema que queremos controlar es lineal, controlable y todas las variables de estado son medibles o pueden ser estimadas, los polos de lazo cerrado del sistema se pueden colocar en cualquier posición deseada mediante una adecuada realimentación del estado. A esta técnica de control se le conoce como *asignación de polos*.

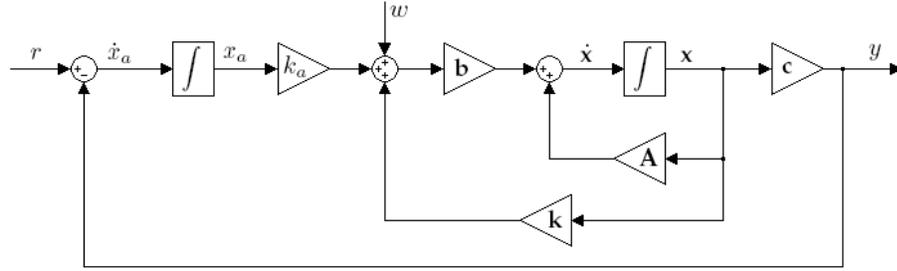


Figura 2.5: Diagrama de bloques de un controlador por asignación de polos

Consideremos es siguiente sistema SISO lineal

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{b}u, \\ y &= \mathbf{cx}.\end{aligned}\tag{2.41}$$

si se considera que existen algunas perturbaciones acopladas (w) a la ecuación de estados, las ecuaciones anteriores se expresan mediante

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{b}u + \mathbf{b}w \\ y &= \mathbf{cx}\end{aligned}\tag{2.42}$$

El objetivo es diseñar un sistema en lazo cerrado tal que la salida $y(t)$ sea capaz de seguir cualquier entrada escalón a pesar de la presencia de las perturbaciones w . Para logra este objetivo, además de la realimentación de estados, se introduce un integrador y una realimentación unitaria, tal y como se muestra en la figura 2.5. La salida del integrador es x_a y se considera como una nueva variable de estado. De la figura 2.5 tenemos que

$$\dot{x}_a = r - y = r - \mathbf{cx}\tag{2.43}$$

$$u = [\mathbf{k} \quad k_a] \begin{bmatrix} \mathbf{x} \\ x_a \end{bmatrix}\tag{2.44}$$

Sustituyendo las ecuaciones 2.43 y 2.44 en 2.42 obtenemos al modelo que describe

al sistema en lazo cerrado de la figura 2.5.

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{x}_a \end{bmatrix} &= \begin{bmatrix} \mathbf{A} + \mathbf{b}\mathbf{k} & \mathbf{b}k_a \\ -\mathbf{c} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ x_a \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} r + \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix} w \\ y &= [\mathbf{c} \ 0] \begin{bmatrix} \mathbf{x} \\ x_a \end{bmatrix} \end{aligned} \quad (2.45)$$

Si el par (\mathbf{A}, \mathbf{b}) es controlable y la función de transferencia $\mathbf{c}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}$ no tiene algún cero en el origen, entonces todos los valores característicos de la matriz \mathbf{A} de la ecuación (2.45) pueden ser ubicados arbitrariamente seleccionando $[\mathbf{k} \ k_a]$ adecuadamente. La demostración de este hecho y de cómo el sistema en lazo cerrado es capaz de hacer un seguimiento de la referencia se puede encontrar en [18].

2.2.3 Control de Equivalencia Cierta

Consideremos un sistema no lineal que tiene la siguiente forma

$$x^{(n)} = f(x, \dot{x}, \dots, x^{(n-1)}) + g(x, \dot{x}, \dots, x^{(n-1)})u, \quad y = x^{(n)}, \quad (2.46)$$

donde u es la entrada, y es la salida y $\bar{x} = [x, \dot{x}, \dots, x^{(n-1)}]$ es el estado del sistema.

Para que el sistema sea controlable se requiere que $g(\bar{x}) \neq 0$.

Si las funciones f y g son conocidas, la siguiente ley de control

$$u = \frac{1}{g(\bar{x})}(-f(\bar{x}) + \nu), \quad (2.47)$$

donde $g(\bar{x}) \neq 0$, puede ser aplicada al sistema de la ecuación (2.46) para hacer que el sistema en lazo cerrado sea

$$x^{(n)} = \nu. \quad (2.48)$$

Si elegimos a ν como

$$\nu = x_d^{(n)} - k_1 e^{(n-1)} - k_2 e^{(n-2)} - \dots - k_n e, \quad (2.49)$$

donde $x_d(t)$ es la trayectoria deseada y $e(t) = x(t) - x_d(t)$ es el error de seguimiento.

El sistema en lazo cerrado de la ecuación (2.48) será

$$x^{(n)} = x_d^{(n)} - k_1 e^{(n-1)} - k_2 e^{(n-2)} - \dots - k_n e, \quad (2.50)$$

$$e^{(n)} + k_1 e^{(n-1)} + k_2 e^{(n-2)} + \dots + k_n e = 0. \quad (2.51)$$

Por lo tanto si el polinomio $s^n + k_1 s^{n-1} + \dots + k_n$ tiene sus raíces en el semiplano izquierdo del plano complejo, entonces

$$\lim_{t \rightarrow \infty} e(t) = 0. \quad (2.52)$$

La ley de control de la ecuación (2.47) usa la linealización por realimentación para cancelar la dinámica no lineal de la planta por la dinámica lineal y estable de ν .

La ley de control de la ecuación (2.47) asume que tenemos perfecto conocimiento de f y g . Si no tenemos perfecto conocimiento de f y g pero asumimos que hemos identificado al sistema por medio de

$$\hat{y} = \hat{f}(\bar{x}) + \hat{g}(\bar{x})u, \quad (2.53)$$

podemos usar la ley de control

$$u_c = \frac{1}{\hat{g}(\bar{x})}(-\hat{f}(\bar{x}) + \nu). \quad (2.54)$$

siempre y cuando $\hat{g}(\bar{x}) \neq 0$.

Escogiendo ν de la misma forma que se hizo anteriormente, encontraremos que la siguiente dinámica para el error

$$e^{(n)} = -k_1 e^{(n-1)} - \dots - k_n e + (\hat{f}(\bar{x}) - f(\bar{x})) + (\hat{g}(\bar{x}) - g(\bar{x}))u_c. \quad (2.55)$$

Para analizar el comportamiento del error podemos reescribir la ecuación (2.53) como un sistema en variables de estado de la siguiente manera.

$$\dot{\mathbf{e}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_n & -k_{n-1} & -k_{n-2} & \cdots & -k_1 \end{bmatrix} \mathbf{e} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \left[(\hat{f}(\bar{x}) - f(\bar{x})) + (\hat{g}(\bar{x}) - g(\bar{x}))u_c \right],$$

$$\dot{\mathbf{e}} = \mathbf{A}_e \mathbf{e} + \mathbf{b}_e u_b, \quad (2.56)$$

$$e = [1 \ 0 \ \cdots \ 0 \ 0] \mathbf{e}, \quad (2.57)$$

donde $\mathbf{e}^T = [e, \dot{e}, \ddot{e}, \dots, e^{(n)}]$.

La solución de la ecuación (2.56) es la siguiente

$$\mathbf{e}(t) = e^{\mathbf{A}_e t} \mathbf{e}(0) + \int_0^t e^{(t-\tau)\mathbf{A}_e} \mathbf{b}_e u_b(\tau) d\tau. \quad (2.58)$$

Para estimar la solución podemos usar el siguiente hecho

$$\|e^{\mathbf{A}_e t}\| \leq k e^{-\lambda t}, \quad (2.59)$$

por lo tanto

$$\begin{aligned} \|\mathbf{e}(t)\| &\leq k e^{-\lambda t} \|\mathbf{e}(0)\| + \int_0^t k e^{-\lambda(t-\tau)} \|\mathbf{b}_e\| \|u_b(\tau)\| d\tau \\ &\leq k e^{-\lambda t} \|\mathbf{e}(0)\| + \frac{k \|\mathbf{b}_e\|}{\lambda} \sup_{\tau} \|u_b(\tau)\|. \end{aligned} \quad (2.60)$$

En la ecuación anterior podemos ver que la respuesta a entrada cero decae exponencialmente a cero mientras que la respuesta a estado cero es acotada para toda entrada acotada. Por lo tanto si $u_b = (\hat{f}(\bar{x}) - f(\bar{x})) + (\hat{g}(\bar{x}) - g(\bar{x}))u_c$ es acotado, el

error de seguimiento e será acotado. Además entre menor sea el valor de u_b , menor será el error de seguimiento.

Capítulo 3

Redes Neuronales y Sistemas

Difusos

El diccionario Merriam-Webster define inteligencia como *“la habilidad de pensar, entender o lidiar con situaciones difíciles; la habilidad de aplicar el conocimiento para manipular el ambiente o pensar abstractamente”*. La inteligencia artificial es la rama de la ciencia informática dedicada al desarrollo de agentes racionales no vivos. Los sistemas de control que incorporan en su diseño conocimientos de la inteligencia artificial tales como la lógica difusa y las redes neuronales, además de los conocimientos propios de la ingeniería de control son denominados *sistemas de control inteligentes*. Este capítulo está dividido en tres secciones dedicadas al estudio de las redes neuronales, los sistemas difusos y las redes neurodifusas, respectivamente.

3.1 Redes Neuronales

Los sistemas nerviosos de los animales y los humanos son de tal naturaleza que nos permiten, no solo procesar información, sino también realizar procesos de aprendizaje. Cuando la neurociencia pudo explicar de forma relativamente convincente el funcionamiento de la neurona, unidad principal de procesamiento de información que posee el cerebro, surgió casi de manera automática la idea de emular dicho funcionamiento con un elemento artificial llamado *neurona artificial*.

Las redes neuronales artificiales son modelos matemáticos inspirados en nuestro entendimiento de las estructuras biológicas en los cerebros de los humanos y animales. Una red neuronal consiste de una colección de neuronas artificiales interconectadas, las cuales colaboran procesando la entrada para producir una salida.

3.1.1 Neuronas

Para nuestros fines definiremos *neurona* como la unidad elemental de procesamiento de información. La información dentro de una red neuronal se transmite entre neuronas por medio de conexiones, donde cada conexión tiene asociado un *peso* por el cual se pondera la señal transmitida. Cada neurona tiene asociada una función $f(\cdot)$ llamada *función de activación* y un parámetro b llamado *umbral*.

Las neuronas operan de la siguiente manera: las entradas x_1, x_2, \dots, x_n llegan a la neurona a través de las conexiones con pesos asociados w_1, w_2, \dots, w_n , entonces la entrada a la función de activación es la suma ponderada $\sum_{i=1}^n w_i x_i$. La salida y de la neurona queda entonces definida por

$$y = f \left(\sum_{i=1}^n w_i x_i - b \right). \quad (3.1)$$

Si se considera otra entrada x_0 definida como $x_0 = 1$, el umbral b puede ser tratado

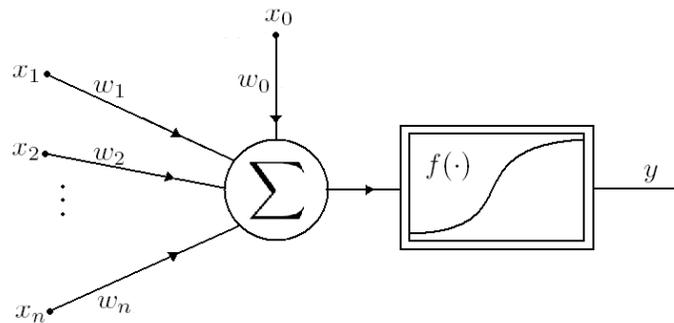


Figura 3.1: Perceptrón: iésima neurona oculta de un perceptrón multicapa

como otro peso $w_0 = -b$. Gracias a esta representación el proceso de ajuste de los pesos y los umbrales puede ser realizado de la misma manera, y la ecuación (3.1) se transforma en

$$y = f \left(\sum_{i=0}^n w_i x_i \right). \quad (3.2)$$

En la figura 3.1 se muestra la representación gráfica de una neurona. Esta neurona también es llamada *perceptrón*.

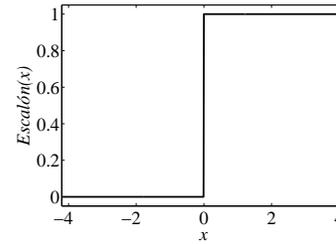
Algunos ejemplos de las funciones de activación más comúnmente empleadas se muestran en la figura 3.2, con sus respectivas gráficas. Las dos últimas funciones tienen la ventaja de ser derivables, lo cual es una característica importante durante el proceso de aprendizaje.

3.1.2 Perceptrón Multicapa

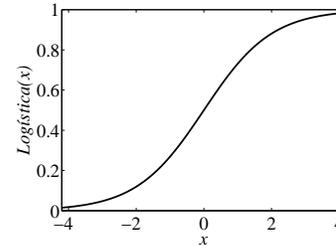
Cuando varios perceptrones son usados en paralelo y conectados a una neurona de salida, tal y como se muestra en la figura 3.3, se obtiene una red neuronal llamada *perceptrón multicapa con una capa oculta*. El perceptrón multicapa es el tipo de red neuronal más ampliamente usado y conocido.

En la terminología de las redes neuronales, la red de la figura 3.3 se describe de la

$$\text{Escalón}(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$$



$$\text{Logística}(x) = \frac{1}{1 + e^{-x}}$$



$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

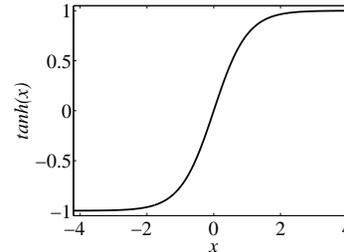


Figura 3.2: Funciones de activación

siguiente manera. El nodo cuya salida es la salida de la red es llamado *neurona de salida* y el conjunto de las neuronas de salida es llamado *capa de salida* (en este caso solo hay una neurona de salida, la cual realiza una combinación lineal de sus entradas). Los perceptrones del centro se llaman *neuronas ocultas* y el conjunto de neuronas ocultas se llama *capa oculta*. Los nodos de entrada son comúnmente llamados *neuronas de entrada* y juntas estas neuronas se llaman *capa de entrada*. Debido a que la literatura no está estandarizada en cuanto a si las entradas deben ser contadas como una capa o no, la red de la figura 3.3 puede ser llamada red neuronal de dos o de tres capas. Para evitar confusiones es conveniente llamar a está red *red neuronal con una capa oculta*.

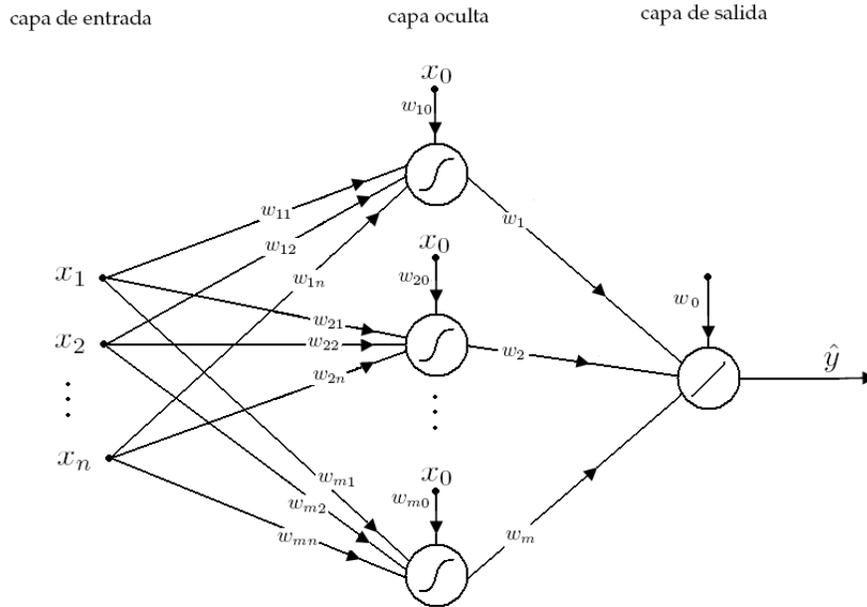


Figura 3.3: Perceptrón multicapa

El modelo matemático del perceptrón multicapa de la figura 3.3 es el siguiente

$$\hat{y} = \sum_{i=0}^m w_i f_i \left(\sum_{j=0}^n w_{ij} x_j \right), \quad \text{con } f_0(\cdot) = 1 \text{ y } x_0 = 1. \quad (3.3)$$

donde w_i son los pesos de la capa de salida, w_{ij} son los pesos de la capa oculta y x_j son las señales de entrada.

Existen otros tipos de perceptrones multicapa y de redes neuronales. Las redes pueden usar distintos métodos para obtener el escalar que es procesado por la función de activación. Al proceso de encontrar este escalar se le llama *mecanismo de construcción*. En el perceptrón multicapa, definido por la ecuación (3.3), el mecanismo de construcción consiste en realizar el producto escalar del vector de pesos $w = [w_0, \dots, w_m]$ y el vector de entradas $x = [x_0, \dots, x_n]$. Otro mecanismo de construcción consiste en encontrar la distancia bajo alguna norma entre el vector w y el

vector x . En [20] se pueden consultar otros mecanismos de construcción.

También existen redes donde el procesamiento realizado en la neurona de salida no es una combinación lineal de sus entradas. Por ejemplo, puede ser que la neurona de salida sea un perceptrón como los de las capas ocultas. Este tipo de redes son descritas en [20] y [7]. Otra diferencia puede consistir en el uso de más de una capa oculta, esto mejora la capacidad de la red para aproximar funciones, pero hace que la red sea más compleja.

3.1.3 Entrenamiento

De acuerdo con Cybenko en [24], una red neuronal con funciones de activación suaves es capaz de aproximar cualquier función continua con un grado de precisión arbitrario, dentro de un intervalo cerrado. Sin embargo, los pesos w de las conexiones entre neuronas deben ser escogidos adecuadamente para que la salida de la neurona sea la deseada.

Al proceso de ajustar los pesos w para encontrar la red neuronal adecuada se le llama *entrenamiento*. Para poder entrenar a una red es necesario contar con información de la relación entrada-salida que queremos lograr. Por lo general la información se encuentra en forma de pares de datos entrada-salida $\{x^i, y^i\}$, y el ajuste de los pesos se realiza en base a la comparación de la salida actual de la red con la salida deseada.

Existen varios algoritmos de entrenamiento, los cuales pueden consultarse en [20], [21] y [7]. El algoritmo más conocido es el algoritmo de retropropagación [8]. Hablando estrictamente, el algoritmo de retropropagación es un método para calcular el gradiente de la salida de un perceptrón multicapa con respecto a sus pesos w . El algoritmo de retropropagación está basado en la aplicación de la regla de la ca-

dena para el cálculo de las derivadas parciales del error entre la salida obtenida y la deseada.

3.2 Sistemas Difusos

Los humanos somos capaces de realizar cierto tipo de actividades como andar en bicicleta, manejar un auto o lanzar una pelota de béisbol; y podemos llegar a ser muy hábiles desempeñando este tipo de actividades a pesar de que no contamos con sensores precisos o con sistemas de ecuaciones que describan el comportamiento de los sistemas con los que lidiamos.

La lógica difusa, en la que se basan los sistemas difusos, fue creada por Lotfi Zadeh [9] como una extensión de la lógica booleana. Mientras la lógica booleana asigna a una variable los valores 0 o 1 (verdadero o falso), la lógica difusa nos permite asignar a una variable cualquier valor en el intervalo $[0, 1]$, lo cual permite que un elemento de un conjunto pertenezca de forma parcial a éste. Esta extensión es motivada por la observación de la forma vaga e imprecisa en que los humanos pensamos y nos comunicamos.

3.2.1 Estructura de un Sistema Difuso

En los sistemas difusos las decisiones se toman con base en reglas del tipo *Si... entonces...*, la parte inmediatamente después del *si* es llamada *parte antecedente* y la parte inmediatamente después del *entonces* es llamada *parte consecuente*. Un buen ejemplo de este tipo de reglas se puede encontrar cuando un entrenador de béisbol trata de enseñar a un grupo de jóvenes peloteros como batear bolas rápidas y les dice

"Si la bola viene rápido, entonces hagan el swing antes."

En el ejemplo anterior y en general en la terminología de los sistemas difusos las

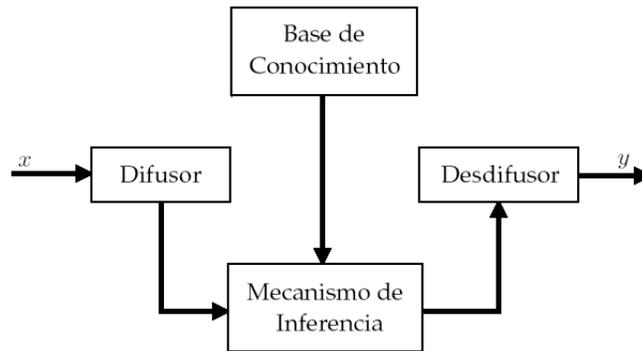
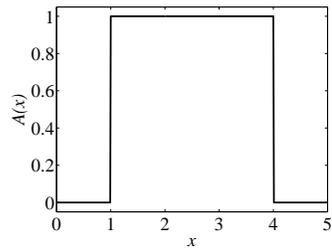


Figura 3.4: Estructura de un sistema difuso

variables como la velocidad de la bola y tiempo en que se debe hacer el swing se llaman *variables lingüísticas*, y los conjuntos a los que pueden pertenecer como rápido y antes se llaman *conjuntos difusos*. Las reglas del tipo *Si... entonces* se llaman *reglas difusas*. El peso que tendrá en la decisión final la parte consecuente de la regla (la parte del entonces) dependerá de que tan verdadera sea la parte antecedente (la parte del Si)

La estructura básica de sistema difuso es mostrada en la figura 3.4. Un sistema difuso está compuesto por las siguientes partes:

- *Difusor*. Convierte las entradas del sistema en información que el mecanismo de inferencia puede usar.
- *Base de conocimiento*. Es el conjunto de reglas difusas. Se construye usando el conocimiento previo que se tiene del sistema. Este conocimiento puede ser extraído de fuentes tales como las leyes físicas que gobiernan al sistema y el conocimiento de un experto.
- *Mecanismo de inferencia*. Emula la forma de tomar decisiones del experto.
- *Desdifusor*. Convierte la conclusión del mecanismo de inferencia en las señales

Figura 3.5: $A = (1, 4)$

de salida del sistema.

3.2.2 Conjuntos Difusos

Antes de definir matemáticamente a los conjuntos difusos, analizaremos a los conjuntos booleanos de un modo tal que nos permita extender la noción de conjunto a la de conjunto difuso. Un subconjunto booleano A de un conjunto X puede ser definido por medio de una *función de pertenencia* $A : X \rightarrow \{0, 1\}$

$$A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases} . \quad (3.4)$$

Los conjuntos booleanos son adecuados para definir a conjuntos tales como las personas mayores de edad o los números reales mayores que 1 y menores que 4, conjunto cuya función de pertenencia se muestra en la figura 3.5. Recordemos el caso del béisbol, mencionado en la sección anterior, podemos darnos cuenta que los conjuntos booleanos no son adecuados para definir a conjuntos tales como las bolas rápidas, las cervezas frías o las porristas guapas. A diferencia de los elementos de los conjuntos booleanos, los elementos de los conjuntos difusos pueden tener *grados de pertenencia* en el intervalo $[0, 1]$.

El subconjunto difuso A de un conjunto X se define por medio de una función

$A : X \rightarrow [0, 1]$. El conjunto X se llama *universo de discurso*. La función A se llama *función de pertenencia* de A . Comúnmente se denota a la función de pertenencia del conjunto A como μ_A , pero en este texto no se hace tal distinción por considerarse que la notación μ_A complica la escritura. Las funciones de pertenencia más comunes son las funciones triangulares, trapezoidales, gaussianas y sigmoides, como las de la figura 3.2. Ejemplos de las funciones anteriores, excepto de las sigmoides, con sus respectivas gráficas se pueden apreciar en la figura 3.6.

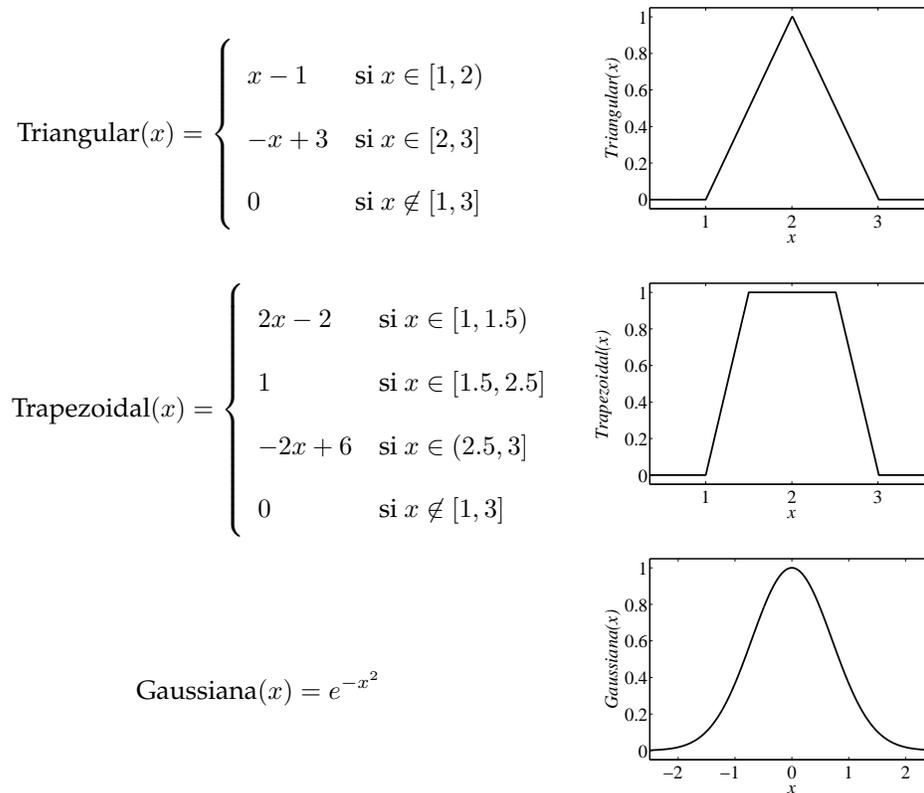


Figura 3.6: Funciones de pertenencia.

Un caso especial de conjunto difuso es el llamado *singleton*. Este caso ocurre cuando la variable lingüística es completamente cierta para un solo valor y es completamente falsa para todos los demás. Un ejemplo de este tipo de conjuntos se muestra en la figura 3.7. Los singletons son adecuados para definir a conjuntos tales como la

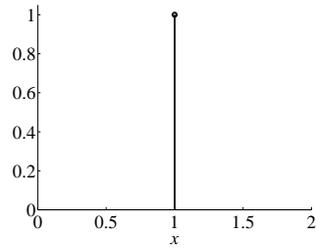


Figura 3.7: Singleton.

temperatura a la que hierve el agua a una atmósfera de presión, la cual es única y no da oportunidad de ambigüedad.

3.2.3 Operaciones con Conjuntos Difusos

Las operaciones lógicas en los conjuntos difusos son extensiones de las operaciones lógicas booleanas. Esto implica que estos dos tipos de operaciones son iguales para los casos especiales en que los grados de pertenencia solo son 0 ó 1.

Negación (not). En los conjuntos booleanos la negación del conjunto A es un conjunto $\neg A$ que contiene a los elementos que no pertenecen al conjunto A . La negación de un conjunto difuso A con una función de pertenencia $A(x)$ se define como

$$\neg A(x) = 1 - A(x). \quad (3.5)$$

Intersección (and). En los conjuntos booleanos la intersección del conjunto A y el conjunto B es un conjunto $A \cap B$ que contiene a los elementos que pertenecen tanto al conjunto A como al conjunto B . Para obtener la intersección de dos conjuntos difusos A y B existen varios operadores llamados *t-normas*. Las t-normas más comunes son:

$$\text{Mínimo : } A \cap B(x) = A(x) \wedge B(x) = \min\{A(x), B(x)\}, \quad (3.6)$$

$$\text{Producto algebraico : } A \cap B(x) = A(x) \cdot B(x) = A(x)B(x), \quad (3.7)$$

$$\text{Diferencia acotada : } A \cap B(x) = \max\{0, A(x) + B(x) - 1\}. \quad (3.8)$$

Unión (or). En los conjuntos booleanos la unión del conjunto A y el conjunto B es un conjunto $A \cup B$ que contiene a los elementos que pertenecen al conjunto A y a los elementos que pertenecen al conjunto B . Para obtener la unión de dos conjuntos difusos A y B existen varios operadores llamados *t-conormas*. Las t-conormas más comunes son:

$$\text{Máximo : } A \cup B(x) = A(x) \vee B(x) = \max\{A(x), B(x)\}, \quad (3.9)$$

$$\text{Suma algebraica : } A \cup B(x) = A(x) + B(x) - A(x)B(x), \quad (3.10)$$

$$\text{Suma acotada : } A \cup B(x) = \min\{1, A(x) + B(x)\}. \quad (3.11)$$

En [1] se realiza un análisis detallado de las propiedades de este tipo de operaciones.

3.2.4 Sistemas Difusos Tipo Mamdani

En los sistemas difusos tipo Mamdani la parte consecuente de las reglas de la *base de conocimiento* implica un conjunto difuso. Por lo tanto las reglas en este tipo de sistemas tienen la siguiente forma

$$R_i: \quad \text{Si } x_1 \text{ es } A_{i1} \text{ y } x_2 \text{ es } A_{i2} \text{ y } \dots \text{ y } x_n \text{ es } A_{in} \text{ entonces } y \text{ es } B_i,$$

donde x_1, x_2, \dots, x_n son las entradas del sistema, y es la salida del sistema, A_{ij} es el conjunto al que debe pertenecer la entrada x_j para que la salida y pertenezca al conjunto B_i según la regla i . Comúnmente varios de los conjuntos A_{ij} son idénticos para diferentes reglas, lo mismo es válido para los conjuntos B_i .

El primer paso para encontrar la salida del sistema difuso es encontrar el grado de pertenencia de las entradas a los conjuntos A_{ij} ; a este proceso se le conoce como *difusión*.

El siguiente paso, llamado *agregación*, consiste en combinar los grados de pertenencia de las entradas usando una t-norma, que comúnmente es el mínimo o el pro-

ducto algebraico

$$A_i(\mathbf{x}) = \bigwedge_{j=1}^n A_{ij}(x_j), \quad (3.12)$$

$$A_i(\mathbf{x}) = \prod_{j=1}^n A_{ij}(x_j), \quad (3.13)$$

donde $\mathbf{x} = [x_1, x_2, \dots, x_n]$.

El siguiente paso, llamado *activación*, consiste en encontrar el grado de verdad de la parte antecedente de cada regla R_i . Esto se logra una vez más usando una t-norma, que comúnmente es el mínimo o el producto algebraico

$$C_i(y) = A_i(\mathbf{x}) \bigwedge B_i(y), \quad (3.14)$$

$$C_i(y) = A_i(\mathbf{x}) \cdot B_i(y). \quad (3.15)$$

Una vez encontradas las salidas de cada regla R_i es necesario combinarlas, este proceso se llama *acumulación*. La combinación de los conjuntos C_i se logra aplicando la t-conorma máximo

$$C(y) = \bigvee_{i=1}^m C_i(y), \quad (3.16)$$

donde m es el número de reglas.

El conjunto C nos dice qué tan cierta es cada salida y . Para calcular la salida final del sistema Mamdani debemos hacer la *desdifusión* del conjunto C . Existen varios métodos para realizar la desdifusión, varios de ellos se analizan en [1]. El método más ampliamente usado es el método del *centroide*. Si se contara con un cuerpo de espesor y densidad uniforme, cuyas caras anterior y posterior tuvieran la forma del $C(y)$, el punto $\hat{y}(\mathbf{x})$ calculado en la ecuación anterior correspondería con la coordenada en el eje y del centro de gravedad (centroide) del cuerpo. La forma de calcular la salida final \hat{y} del sistema difuso usando el método del centroide es la siguiente

$$\hat{y}(\mathbf{x}) = \frac{\int yC(y) dy}{\int C(y) dy} \quad (3.17)$$

Una forma alternativa de calcular el centroide $\hat{y}(\mathbf{x})$ es la siguiente

$$\hat{y}(\mathbf{x}) = \frac{\sum_{k=1}^p y_k C(y_k)}{\sum_{k=1}^p C(y_k)} \quad . \quad (3.18)$$

donde y_k es la variable y muestreada.

En la figura 3.8 se esquematiza el funcionamiento de un sistema difuso tipo Mamdani cuya base de conocimiento contiene dos reglas, donde la agregación y la activación se realizan utilizando la t-norma mínimo, la acumulación se realiza utilizando la t-conorma máximo y la desdifusión se realiza utilizando el método del centroide.

Cuando los conjuntos B_i de la parte consecuente de las reglas difusas son singletons, el sistema se nombra *sistema difuso tipo singleton*. En este caso no importa si usa el mínimo o el producto algebraico para realizar el proceso de activación. Además el método de desdifusión del centroide se convierte en un promedio ponderado de la forma

$$\hat{y}(\mathbf{x}) = \frac{\sum_{i=1}^m y_i A_i(\mathbf{x})}{\sum_{i=1}^m A_i(\mathbf{x})} \quad , \quad (3.19)$$

donde m es el número de reglas e y_i es el valor de y para el cual $B_i(y) = 1$.

3.2.5 Sistemas Difusos Tipo Takagi-Sugeno

En los sistemas difusos tipo Takagi-Sugeno la parte consecuente de las reglas de la base de conocimiento implica una función de las variables de entrada. Por lo tanto las reglas en este tipo de sistemas toman la siguiente forma

$$R_i: \quad \text{Si } x_1 \text{ es } A_{i1} \text{ y } x_2 \text{ es } A_{i2} \text{ y } \dots \text{ y } x_n \text{ es } A_{in} \text{ entonces } y = f_i(x_1, x_2, \dots, x_n).$$

En este tipo de sistemas la difusión y la agregación se llevan a cabo de la misma

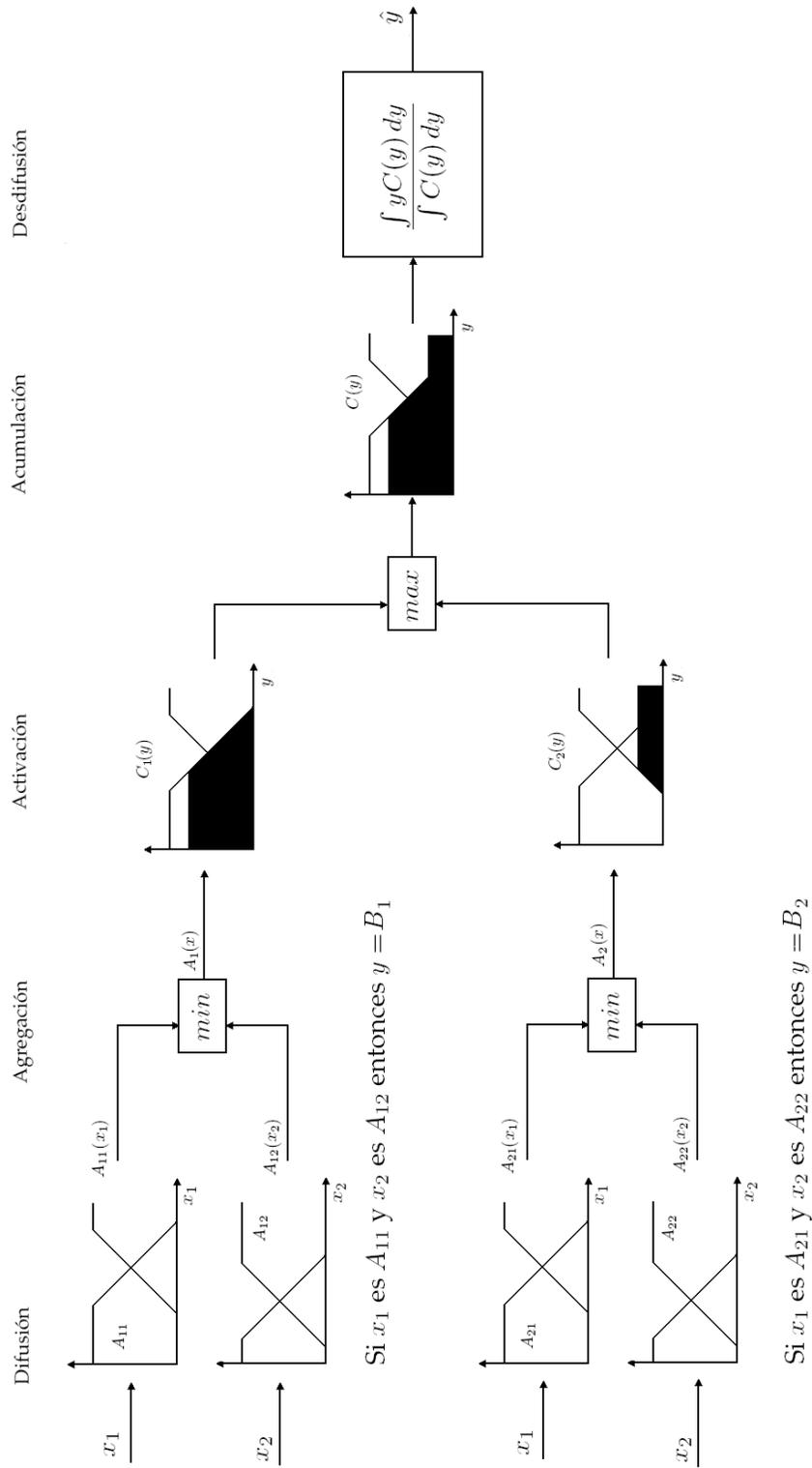


Figura 3.8: Sistema difuso tipo Mamdani

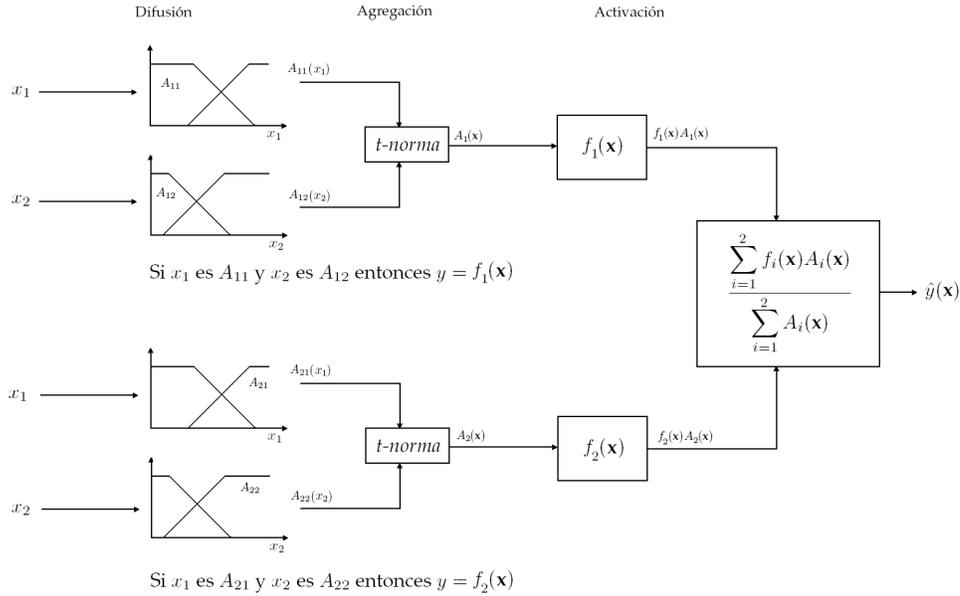


Figura 3.9: Sistema difuso tipo Takagi-Sugeno

forma que en los sistemas tipo Mamdani. Una vez obtenidos los valores $A_i(\mathbf{x})$, la salida de un sistema tipo Takagi-Sugeno con m reglas se calcula de la siguiente manera

$$\hat{y}(\mathbf{x}) = \frac{\sum_{i=1}^m f_i(\mathbf{x})A_i(\mathbf{x})}{\sum_{i=1}^m A_i(\mathbf{x})} \quad (3.20)$$

Si las funciones f_i son trivialmente escogidas como constantes, el sistema se convierte en un sistema difuso tipo singleton.

En la figura 3.9 se muestra la representación gráfica de un sistema difuso tipo Takagi-Sugeno que tiene dos reglas difusas y dos variables de entrada.

3.2.6 Desventajas del Control Difuso

Los sistemas difusos pueden ser usados como controladores, si se colocan en la posición del bloque "control" de la figura 2.3. Cuando los sistemas difusos se usan como con-

troles existen ciertas desventajas asociadas.

Los sistemas difusos están diseñados para emular el conocimiento de un experto. Pero debido a que las normas-t, conormas-t, y los métodos de desdifusión son escogidos por el diseñador; puede ocurrir que el peso que las decisiones del diseñador tienen sobre el funcionamiento del sistema sea mayor que el peso que tiene el conocimiento del experto.

En sistema difuso se puede interpretar como un bloque que realiza un mapeo no lineal $u = f(x_1, x_2, \dots, x_n)$, donde u es la salida del sistema difuso y x_i son las entradas. El mapeo f puede ser implementado usando otro tipo de estructuras, como los polinomios y las "lookup tables". Cuando se usan los polinomios y las "lookup tables", la implementación del mapeo f puede ser más sencillo, al igual que la modificación del mismo.

Cuando se usa un sistema difuso tipo Mamdani, no existe una descripción analítica del mapeo f . Este hecho dificulta el análisis de la estabilidad del sistema, para el cual solo existen métodos basados en simulaciones.

3.3 Redes Neurodifusas

Las redes neurodifusas son redes neuronales cuya estructura ha sido apropiadamente modificada para realizar las funciones propias de un sistema difuso. La razón para representar un sistema difuso en los términos de una red neuronal es aprovechar la capacidad de aprendizaje de las redes neuronales para mejorar el desempeño del sistema difuso.

Como primer paso para establecer la similitud entre las redes neuronales y los sistemas difusos podemos observar el hecho de que las ecuaciones (3.3) y (3.19) tienen

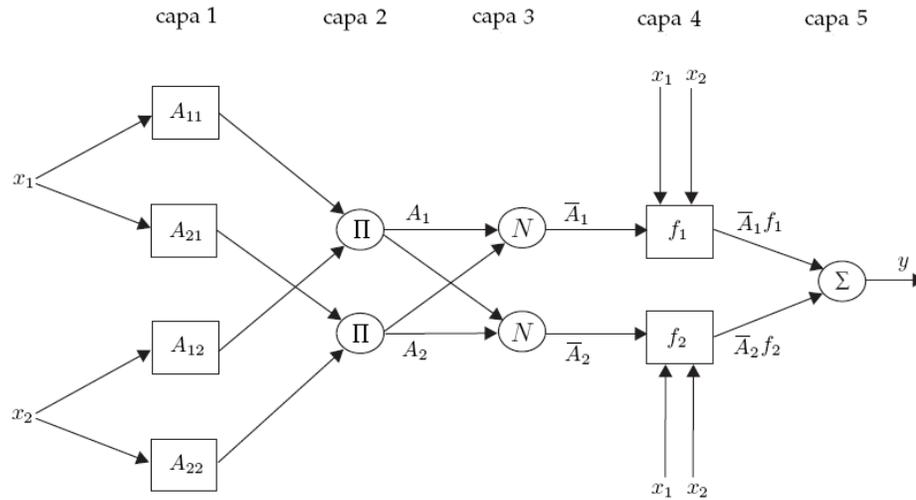


Figura 3.10: ANFIS

la siguiente forma

$$y = \sum_{i=0}^m w_i f_i(\mathbf{x}), \quad (3.21)$$

con w_i constantes.

La red neuronal ANFIS (por las siglas en inglés de *adaptive network based fuzzy inference system*) emplea un sistema difuso tipo Takagi-Sugeno. La arquitectura básica de una red ANFIS se muestra en la figura 3.10. Los nodos cuadrados contienen parámetros adaptables, por otra parte, los nodos circulares son estáticos.

La red de la figura 3.10 considera las dos reglas siguientes

R_1 : Si x_1 es A_{11} y x_2 es A_{12} entonces $y = f_1(x)$.

R_2 : Si x_1 es A_{21} y x_2 es A_{22} entonces $y = f_2(x)$.

La función de la capa 1 realizar la difusión de las entradas. La capa 2 realiza el producto de los grados de pertenencia de las entradas, tal y como se hace en la

ecuación 3.13. La capa 3 es un normalizador, la salidas de estos nodos son

$$\bar{A}_i = \frac{A_i(\mathbf{x})}{\sum_{i=1}^2 A_i(\mathbf{x})}, \quad \text{con } i = 1, 2. \quad (3.22)$$

Finalmente la capa 4 genera la salida

$$y(\mathbf{x}) = \sum_{i=1}^2 \bar{A}_i f_i(\mathbf{x}). \quad (3.23)$$

Es importante notar que combinando las ecuaciones (3.23) y (3.22) se obtiene la ecuación (3.20).

Cuando se realice el entrenamiento de esta red se buscará optimizar los parámetros involucrados tanto en las funciones de pertenencia A_{ij} como en las funciones f_i . La estructura de esta red puede ser extendida para aceptar más entradas y reglas.

Los sistemas difusos pueden ser optimizados sin la necesidad de ser dibujados en forma de redes neuronales. Existen métodos de entrenamiento que no tienen relación con las redes neuronales, a pesar de esto el nombre de *redes neurodifusas* ha sobrevivido para nombrar a los sistemas difusos que son entrenados basándose en datos de cómo debe ser su comportamiento.

3.4 Redes Neurodifusas Recurrentes

El término recurrente se usa para designar a los sistemas en los que la salida actual depende de valores anteriores de la salida. Los modelos recurrentes, en contraste con los modelos estáticos, son capaces de almacenar información acerca del pasado, y son por tal motivo, más apropiados para analizar sistemas dinámicos.

Las redes neurodifusas recurrentes pueden ser construidos de la misma forma que las redes neurodifusas. Por lo tanto son construidas basándose en una red neuronal recurrente o en un sistema difuso recurrente. En [7] y [4] se realiza un estudio más

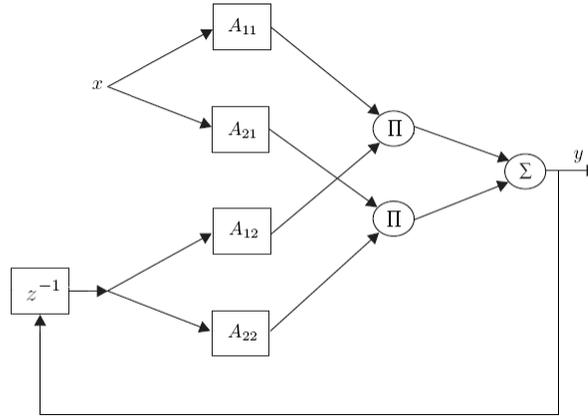


Figura 3.11: Red neurodifusa recurrente

detallado de las redes neurodifusas recurrentes.

En la figura 3.11 se muestra una sencilla red neurodifusa recurrente, la segunda entrada de esta red es la salida en el tiempo anterior.

3.5 Aproximadores Universales

Las redes neuronales y los sistemas difusos son aproximadores universales; esto significa que pueden aproximar a cualquier función suave con un grado de precisión arbitrario, dentro de un intervalo cerrado. Esta cualidad justifica el uso de los perceptrones multicapa en la resolución de problemas de aproximación de funciones.

Teorema de aproximación universal. Para cualquier función real y continua g definida en un conjunto compacto $U \subset \mathbb{R}^n$ y un número real $\varepsilon > 0$, existe un sistema difuso o una red neuronal tal que

$$\sup_{x \in U} |f(x) - g(x)| < \varepsilon. \quad (3.24)$$

La prueba de este teorema se puede encontrar en [24], [14], [1] y en [20]

Este teorema justifica el uso de las redes neuronales y los sistemas difusos en la resolución de problemas de aproximación de funciones e identificación de sistemas. Hay que notar que este teorema no hace referencia a la complejidad de la red neuronal o del sistema difuso, ni a la forma de obtenerlos.

Capítulo 4

Red Neurodifusa Recurrente

Entrada-Salida

Como ya se mencionó en la sección 1.2, el principal objetivo del actual trabajo es la identificación de un sistema usando una red neurodifusa recurrente entrada-salida. La red que usaremos es propuesta por Gonzalez-Olvera y Tang en [2] y [3]. Este capítulo está dedicado a estudiar la estructura y funcionamiento de dicha red neurodifusa recurrente.

4.1 Estructura de la Red

Consideremos un sistema SISO (por las siglas en inglés de *single-input single-output*) con entrada u_k y salida y_k , modelado por la siguiente ecuación en diferencias

$$y_{k+1} = f_d(y_k, \dots, y_{k-n+1}, u_k, \dots, u_{k-m+1}). \quad (4.1)$$

Además hacemos las siguientes suposiciones acerca del sistema:

- El sistema es estable en el sentido que una entrada acotada producirá una salida acotada.
- Conocemos de antemano el número de retrasos en la entrada y en la salida, n y m , respectivamente.
- La función f_d es continua.
- Contamos con N número de pares de datos entrada-salida, $\{u^i, y^i\}$.

La meta es encontrar una red neurodifusa recurrente con parámetros θ

$$\hat{y}_{k+1} = f(\hat{y}_k, \dots, \hat{y}_{k-n+1}, u_k, \dots, u_{k-m+1}, \theta), \quad (4.2)$$

tal que

$$\sup_k |\hat{y}_k(\cdot) - y_k(\cdot)| < \varepsilon, \quad (4.3)$$

donde $\varepsilon > 0$ es lo más pequeño posible.

Para tales fines se propone una red neurodifusa con n_R reglas

R_i : Si \hat{y}_k es A_i entonces

$$\begin{aligned} \hat{y}_{k+1} &= [c_{i1}, \dots, c_{in}] \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + [h_{i1}, \dots, h_{im}] \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}, \\ &= c_i^T z_k + h_i^T \xi_k, \end{aligned} \quad (4.4)$$

donde

$$c_i = \begin{bmatrix} c_{i1} \\ \vdots \\ c_{in} \end{bmatrix}, \quad z_k = \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix}, \quad h_i = \begin{bmatrix} h_{i1} \\ \vdots \\ h_{im} \end{bmatrix}, \quad \xi_k = \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}, \quad (4.5)$$

\hat{y}_k es el valor estimado de la señal de salida,

$$A_i(x) = \mu = e^{-\sigma_i^2(x-s_i)^2}, \quad (4.6)$$

son las funciones de pertenencia de la parte antecedente de las reglas difusas y la desdifusión se realiza por medio de un promedio ponderado.

El sistema puede ser descrito por las siguientes ecuaciones

$$\begin{aligned}\hat{y}_{k+1} &= (\varphi_k^1 c_1^T + \dots + \varphi_k^{n_R} c_{n_R}^T) z_k + \\ & (\varphi_k^1 h_1^T + \dots + \varphi_k^{n_R} h_{n_R}^T) \xi_k, \\ &= (\varphi_k^T \otimes z_k^T) \text{vect}(C^T) + (\varphi_k^T \otimes \xi_k^T) \text{vect}(H^T),\end{aligned}\quad (4.7)$$

donde \otimes es el producto de Kronecker, $\text{vect}(\cdot)$ es el operador vectorización,

$$C = \begin{bmatrix} c_1^T \\ \vdots \\ c_{n_R}^T \end{bmatrix}, \quad H = \begin{bmatrix} h_1^T \\ \vdots \\ h_{n_R}^T \end{bmatrix}, \quad \varphi_k = \begin{bmatrix} \varphi_k^1 \\ \vdots \\ \varphi_k^{n_R} \end{bmatrix}, \quad (4.8)$$

$$\varphi_k^i = \frac{A_i(\hat{y}_k)}{\sum_{j=i}^{n_R} A_j(\hat{y}_k)}. \quad (4.9)$$

El producto de Kronecker está definido por

$$Q \otimes P = \begin{bmatrix} q_{11}P & \cdots & q_{1n}P \\ \vdots & \ddots & \vdots \\ q_{m1}P & \cdots & q_{mn}P \end{bmatrix}, \quad (4.10)$$

donde Q es una matriz de $m \times n$ y P es una matriz de $k \times l$.

La vectorización está definida por

$$\text{vect}(Q) = [q_{11}, \dots, q_{m1}, q_{12}, \dots, q_{m2}, \dots, q_{1n}, \dots, q_{mn}]^T, \quad (4.11)$$

Los parámetros a identificar son:

$$C, \quad H, \quad \sigma = \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_{n_R} \end{bmatrix}, \quad \varsigma = \begin{bmatrix} \varsigma_1 \\ \vdots \\ \varsigma_{n_R} \end{bmatrix}. \quad (4.12)$$

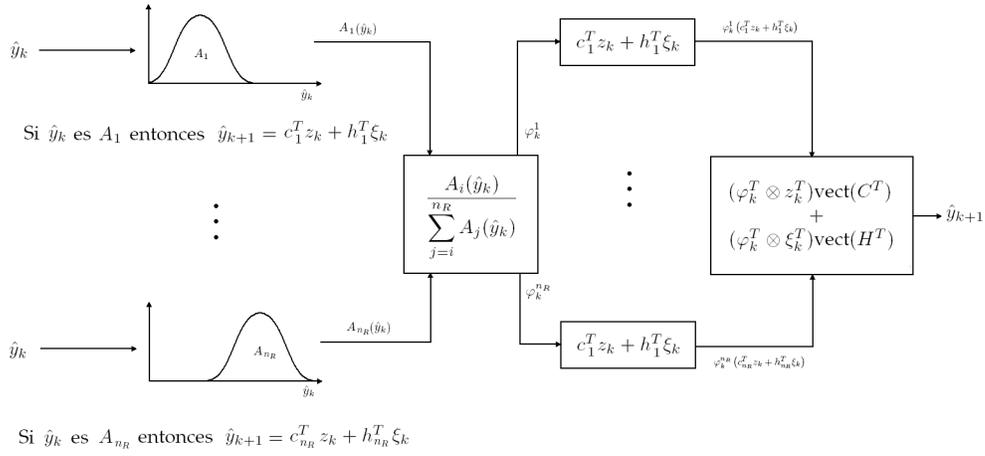


Figura 4.1: Red neurodifusa

La aproximación al sistema real con salida y_{k+1} está dada por

$$\begin{aligned}
 y_{k+1} &= f_1(C, \sigma, \varsigma, z_k, \xi_k) + f_2(H, \sigma, \varsigma, z_k, \xi_k) + \bar{\epsilon}_k, \\
 &= f(C, H, \sigma, \varsigma, z_k, \xi_k) + \bar{\epsilon}_k,
 \end{aligned} \tag{4.13}$$

El error de aproximación $\bar{\epsilon}_k$ será mínimo ($\epsilon_k = \min_{\theta} \bar{\epsilon}_k$) si los valores óptimos de C, H, σ y ς son encontrados, entonces

$$y_{k+1} = f(C^*, H^*, \sigma^*, \varsigma^*, z_k, \xi_k) + \epsilon_k. \tag{4.14}$$

Es importante notar que la red neurodifusa es un modelo no lineal del sistema. Cada regla difusa contiene un modelo lineal del sistema en su parte consecuente, cada uno de estos modelos lineales aproxima al sistema en alguna región de operación. La red combina los modelos lineales para crear un modelo global del sistema.

La figura 4.1 es la representación gráfica de la estructura de la red neurodifusa.

4.2 Linealización de los parámetros antecedentes

Para poder entrenar a la red es conveniente contar con un modelo de la misma que sea lineal con respecto a sus parámetros. Este no es caso de la red descrita anteriormente, ya que si bien es lineal con respecto a los parámetros de la parte consecuente, existen relaciones no lineales de los parámetros de la parte antecedente. Si asumimos que los parámetros están cerca de un mínimo local para los parámetros antecedentes, entonces, podemos linealizar a la red alrededor de este punto.

Para el proceso de linealización obtendremos la expansión de la red óptima (la que usa a los parámetros óptimos θ^*) en su serie de Taylor, alrededor de los parámetros subóptimos θ .

$$f(\theta^*) = f(\theta) + (\theta^* - \theta)^T \left. \frac{\partial f}{\partial \theta^*} \right|_{\theta^* = \theta} + R(\theta, \theta^*), \quad (4.15)$$

donde

$$\theta = \begin{bmatrix} \text{vect}(C^T) \\ \text{vect}(H^T) \\ \sigma \\ \varsigma \end{bmatrix}, \quad (4.16)$$

y $R(\theta, \theta^*)$ son los términos de mayor orden de la expansión. Ya que contamos con mediciones de la salida podemos considerar que $z_k = [y_k, \dots, y_{k-n+1}]^T$.

Debido a que f es lineal con respecto a los parámetros consecuentes C y H , la expansión puede ser escrita de la siguiente forma

$$\begin{aligned} f(\theta^*, z_k, \xi_k) &= f(C, H, \sigma, \varsigma, z_k, \xi_k) \\ &+ f(C^* - C, H^* - H, \sigma, \varsigma, z_k, \xi_k) \\ &+ (\sigma^* - \sigma)^T \left. \frac{\partial f}{\partial \sigma^*} \right|_{\theta} \\ &+ (\varsigma^* - \varsigma)^T \left. \frac{\partial f}{\partial \varsigma^*} \right|_{\theta} + R(\theta, \theta^*). \end{aligned} \quad (4.17)$$

Considerando que $y_{k+1} = f(C^*, H^*, \sigma^*, \varsigma^*, z_k, \xi_k) + e_k$ y que $\hat{y}_{k+1} = f(C, H, \sigma, \varsigma, z_k, \xi_k)$, se define el error de aproximación como

$$\begin{aligned}
\tilde{y}_{k+1} &= y_{k+1} - \hat{y}_{k+1} = (\sigma^* - \sigma)^T g_k + (\varsigma^* - \varsigma)^T h_k \\
&\quad + (\varphi_k^T \otimes z_k^T) \text{vect}((C^* - C)^T) \\
&\quad + (\varphi_k^T \otimes \xi_k^T) \text{vect}((H^* - H)^T) + \vartheta_k, \\
&= y_{k+1} - \hat{y}_{k+1} = \tilde{\sigma}^T g_k + \tilde{\varsigma}^T h_k \\
&\quad + (\varphi_k^T \otimes z_k^T) \text{vect}(\tilde{C}^T) + (\varphi_k^T \otimes \xi_k^T) \text{vect}(\tilde{H}^T) + \vartheta_k,
\end{aligned} \tag{4.18}$$

donde

$$\vartheta_k = R + \epsilon_k, \quad \rho_k = \left. \frac{\partial f}{\partial \sigma^*} \right|_{\theta}, \quad \varrho_k = \left. \frac{\partial f}{\partial \varsigma^*} \right|_{\theta}, \tag{4.19}$$

$$\tilde{C} = C^* - C, \quad \tilde{H} = H^* - H, \quad \tilde{\sigma} = \sigma^* - \sigma, \quad \tilde{\varsigma} = \varsigma^* - \varsigma. \tag{4.20}$$

En las ecuaciones (4.21) y (4.22) se muestra el calculo de la derivada parcial de f_1 con respecto de σ , las derivadas parciales con respecto de ς se pueden obtener de manera similar.

$$\left. \frac{\partial f_1}{\partial \sigma^*} \right|_{\sigma} = \frac{\partial \varphi^T}{\partial \sigma} C z(k) \tag{4.21}$$

$$\frac{\partial \varphi^T}{\partial \sigma} = 2 \begin{bmatrix} \frac{\sigma_1 (y_k - \varsigma_1)^2}{\sum_j \mu_j} \mu_1 \\ \vdots \\ \frac{\sigma_n (y_k - \varsigma_n)^2}{\sum_j \mu_j} \mu_n \end{bmatrix} \otimes \begin{bmatrix} \mu_1 & \cdots & \mu_n \end{bmatrix} - 2 \text{diag} \left(\begin{bmatrix} \frac{\sigma_1 (y_k - \varsigma_1)^2}{\sum_l \mu_l} \mu_1 & \cdots & \frac{\sigma_n (y_k - \varsigma_n)^2}{\sum_l \mu_l} \mu_n \end{bmatrix} \right) \tag{4.22}$$

La ecuación (4.18) es lineal con respecto a sus parámetros. Esto se ve más claramente si se reescribe de la siguiente forma

$$\tilde{y}_{k+1} = G_k \tilde{\theta} + \vartheta_k = G_k \tilde{\theta}_k + \vartheta_k, \tag{4.23}$$

donde

$$\tilde{\theta} = \theta^* - \theta, \quad G_k = \begin{bmatrix} \varphi_k^T \otimes z_k \\ \varphi_k^T \otimes \xi_k^T \\ \rho_k \\ \varrho_k \end{bmatrix}. \quad (4.24)$$

4.3 Entrenamiento de la Red Usando el Filtro de Kalman

Para entrenar a la red neurodifusa se requiere de un modelo lineal con respecto a los parámetros de la misma, además para que el modelo lineal sea valido es necesario inicializar el valor de θ cerca de un mínimo local de la función de costo, esto provocará que los valores de $(\theta^* - \theta)$ y de R sean pequeños. El algoritmo para la inicialización de θ se discute en la sección 4.6.

Si $(\theta^* - \theta)$ y R son pequeños el modelo linealizado de la red se puede mostrar como

$$\hat{\theta}_{k+1} = \hat{\theta}_k, \quad (4.25)$$

$$\hat{y}_{k+1} = G_k \hat{\theta}_k - \bar{\epsilon}_k. \quad (4.26)$$

Existen perturbaciones como R y $\bar{\epsilon}_k$ cuyo efecto debe ser tomado en cuenta. Se puede usar un algoritmo de entrenamiento basado en el filtro de Kalman, como el descrito en la sección 3.1.4.

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_k (y^k - G_k \hat{\theta}_{k-1}), \quad (4.27)$$

$$K_k = \frac{P_{k-1} G_k^T}{R_2 + G_k P_{k-1} G_k^T}, \quad (4.28)$$

$$P_k = P_{k-1} - \frac{P_{k-1} G_k^T G_k P_{k-1}}{R_2 + G_k P_{k-1} G_k^T} + R_1. \quad (4.29)$$

4.4 Entrenamiento de la Red Usando el Gradiente

También podemos usar un método basado en el gradiente para entrenar a la red. La aplicación de un método basado en el gradiente no requiere de un modelo lineal con respecto a los parámetros. Pero, debido las razones explicadas en la sección 3.1.5, si es necesario inicializar a la red cerca de un buen mínimo local de la función de costo.

Si escogemos como función de costo a

$$V_{k+1}(\theta) = \frac{1}{2}(\hat{y}_{k+1} - y_{k+1})^2, \quad (4.30)$$

donde \hat{y}_{k+1} se calcula usando la ecuación (4.7). El gradiente es

$$g = \frac{\partial V(\theta)}{\partial \theta} = (\hat{y}_{k+1} - y_{k+1})G_k, \quad (4.31)$$

donde θ y G_k se definen en las ecuaciones (4.16) y (4.24) respectivamente.

Conociendo al gradiente y habiendo elegido los valores de η y R podemos usar las ecuaciones (4.31) para ajustar el valor de θ .

$$\theta_{k+1} = \theta_k - \eta_k p_k, \quad p_k = R_k g_k. \quad (4.32)$$

4.5 Restricciones para los Parámetros

Durante el proceso de entrenamiento algunos parámetros pueden tomar valores indeseados. El primer comportamiento que debe ser evitado es que los parámetros ς tomen valores tales que coloquen a los conjuntos difusos fuera de la región de operación del sistema. El segundo comportamiento de este tipo es que los elementos de la primera columna de H tomen valores con signos distintos. La razón por la cual este último comportamiento debe ser evitado se explica en la siguiente sección. Para evitar este tipo de comportamientos indeseados se imponen restricciones a los

parámetros. Las restricciones que aquí se proponen son similares a las propuestas en [3].

Las restricciones se implementan por medio de funciones ϕ del tipo sigmoides. El parámetro x restringido se define como $x = \phi(\tilde{x})$. Las funciones ϕ deben cumplir con las siguientes características:

- $\phi(\tilde{x}) \in (a, b)$, $a \in \mathfrak{R}$, $b \in \mathfrak{R}$
- $\phi(\hat{x}) = \hat{x}$
- $\lim_{\tilde{x} \rightarrow \infty} \phi(\tilde{x}) = a$
- $\lim_{\tilde{x} \rightarrow -\infty} \phi(\tilde{x}) = b$
- $0 \leq \frac{\partial \phi}{\partial \tilde{x}} \leq 1$

donde a es la cota superior del parámetro restringido, b es la cota inferior del parámetro restringido y \hat{x} es el valor inicial del parámetro restringido.

Por ejemplo, si el valor inicial de ς_1 es 3, el valor máximo que puede tomar es 4 y el valor mínimo que puede tomar es 2; el parámetro restringido se puede definir como

$$\varsigma_1 = \tanh(\tilde{\varsigma}_1 - 3) + 3, \quad (4.33)$$

la gráfica de la función anterior se muestra en la figura 4.2 En este caso el entrenamiento se realizará sobre $\tilde{\varsigma}_1$ y la restricción garantiza que ς_1 permanecerá dentro de la región deseada.

Una vez impuestas las restricciones a los parámetros de la primera columna de H y a los parámetros ς las ecuaciones de la red se convierten en:

$$\begin{aligned} y_{k+1} &= f_1(C, \sigma, \zeta, z_k) + f_2(\bar{H}, \check{h}_{c1}, \sigma, \zeta, \bar{\xi}_k, u_k) + \bar{e}_k, \\ &= f(C, \bar{H}, \check{h}_{c1}, \sigma, \zeta, z_k, \bar{\xi}_k, u_k) + \bar{e}_k, \end{aligned} \quad (4.34)$$

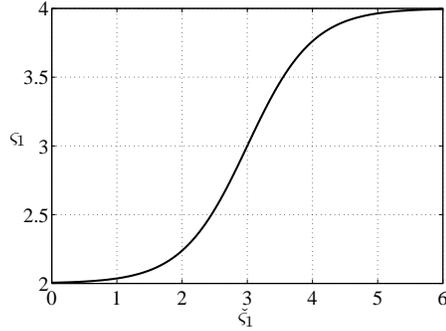


Figura 4.2: Parâmetro restringido

donde \bar{H} es la matriz H sin la primera columna

$$\bar{H} = [h_{c2}, h_{c3}, \dots, h_{cn_R}], \quad (4.35)$$

$$\bar{\xi} = [u_{k-1}, \dots, u_{k-m+1}]^T, \quad (4.36)$$

$h_{c1} = \phi_h(\check{h}_{c1})$ es la primera columna de la matriz H y $\varsigma = \phi_\zeta(\check{\zeta})$.

Expandiendo en series de Taylor obtenemos

$$\begin{aligned} f(\theta^*, z_k, \xi_k) &= f(C, H, \sigma, \zeta, z_k, \bar{\xi}_k, u_k) \\ &+ f(C^* - C, \bar{H}^* - \bar{H}, \sigma, \zeta, z_k, \bar{\xi}_k, u_k) \\ &+ (\check{h}_{c1}^* - \check{h}_{c1})^T \frac{\partial f}{\partial \check{h}_{c1}^*} \Big|_{\theta} \\ &+ (\sigma^* - \sigma)^T \frac{\partial f}{\partial \sigma^*} \Big|_{\theta} \\ &+ (\zeta^* - \zeta)^T \frac{\partial f}{\partial \zeta^*} \Big|_{\theta} + R(\theta, \theta^*). \end{aligned} \quad (4.37)$$

Finalmente

$$\frac{\partial f}{\partial \check{h}_{c1}^*} \Big|_{\theta} = \frac{\partial \phi_h}{\partial \check{h}_{c1}} \varphi^T u_k \quad (4.38)$$

y

$$\frac{\partial f_1}{\partial \zeta^*} \Big|_{\zeta} = \frac{\partial \varphi^T}{\partial \zeta} C z(k), \quad (4.39)$$

$$\begin{aligned}
\frac{\partial \varphi^T}{\partial \zeta} = & -2 \begin{bmatrix} \frac{\partial \phi_{\zeta_1}}{\partial \zeta_1} \frac{\sigma_1^2(y_k - \zeta_1)}{\sum_j \mu_j} \mu_1 \\ \vdots \\ \frac{\partial \phi_{\zeta_n}}{\partial \zeta_n} \frac{\sigma_n^2(y_k - \zeta_n)}{\sum_j \mu_j} \mu_n \end{bmatrix} \otimes \begin{bmatrix} \mu_1 & \cdots & \mu_n \end{bmatrix} \\
& + 2 \text{diag} \left(\begin{bmatrix} \frac{\partial \phi_{\zeta_1}}{\partial \zeta_1} \frac{\sigma_1^2(y_k - \zeta_1)}{\sum_l \mu_l} \mu_1 & \cdots & \frac{\partial \phi_{\zeta_n}}{\partial \zeta_n} \frac{\sigma_n^2(y_k - \zeta_n)}{\sum_l \mu_l} \mu_n \end{bmatrix} \right).
\end{aligned} \tag{4.40}$$

Es importante notar la similitud entre la ecuación anterior y la ecuación (4.22). El cálculo de las derivadas parciales no se complica demasiado al incorporar las restricciones, sólo hay que tomar en cuenta a los términos $\frac{\partial \phi_{\zeta}}{\partial \zeta}$.

4.6 Algoritmo de Inicialización de Parámetros

El algoritmo de inicialización de parámetros se propone con el objetivo de iniciar el entrenamiento de la red con un valor de θ tal que la función de costo se encuentre cerca de algún mínimo. Esto provoca que los valores de $(\theta^* - \theta)$ y de R sean pequeños.

En [2] y [3] se propone un algoritmo para inicializar los parámetros de la red que hace uso de ANFIS, pero es posible usar otros algoritmos que realicen tareas similares. El algoritmo que se propone y utiliza en este trabajo es el siguiente:

1. Excitar al sistema con señales pequeñas en amplitud pero con distintos offsets, de tal forma que hagan que las salidas del sistema se encuentren dentro ciertas regiones de operación acotadas.
2. Haciendo uso del método de los mínimos cuadrados recursivo, obtener modelos lineales, que identifiquen al sistema dentro cada una de las regiones de operación acotadas, con la estructura de los modelos lineales de las partes consecuentes de las reglas difusas de la red (ver ecuación (4.4)).
3. Inicializar los parámetros de C y H con valores de los parámetros c_i y h_i de los

modelos obtenidos en el paso anterior.

4. Inicializar los parámetros ς_i de los conjuntos A_i con valores que estén dentro de cada una de las regiones de operación acotadas.
5. Inicializar los parámetros σ_i de los conjuntos A_i con valores que sean congruentes con el número de modelos lineales obtenidos en el paso 2 y con el tamaño de las regiones de operación acotadas.

4.7 Control de Equivalencia Cierta Usando la Red

Una vez que la red neurodifusa ha sido entrenada y aproxima satisfactoriamente al sistema, la podemos usar para implementar un control de equivalencia cierta como el que se explico en la sección 3.2.2.

La red neurodifusa es descrita por la siguiente ecuación, presentada anteriormente en la ecuación (4.7),

$$\hat{y}_{k+1} = (\varphi_k^T \otimes z_k^T) \text{vect}(C^T) + (\varphi_k^T \otimes \xi_k^T) \text{vect}(H^T). \quad (4.41)$$

Haremos una pequeña modificación para que la ecuación (4.41) adquiriera la forma de la ecuación (2.53)

$$\hat{y}_{k+1} = (\varphi_k^T \otimes z_k^T) \text{vect}(C^T) + \varphi_k^T \bar{H} \bar{\xi}_k + \varphi_k^T h_{c1} u_k, \quad (4.42)$$

Si se cumple que $\varphi_k^T h_{c1} \neq 0$, se puede proponer el siguiente control de equivalencia cierta, similar al de la ecuación (2.54)

$$u_{ck} = \frac{1}{\varphi_k^T h_{c1}} (-(\varphi_k^T \otimes z_k^T) \text{vect}(C^T) - \varphi_k^T \bar{H} \bar{\xi}_k + \nu). \quad (4.43)$$

Si los parámetros h_{c1} tienen valores con signos distintos, no es posible garantizar que $\varphi_k^T h_{c1} \neq 0$.

Definimos a ν como

$$\nu = y_{(k+1)d} - k_1 e_k - k_2 e_{k-1} - \cdots - k_{p+1} e_{k-p}, \quad (4.44)$$

donde $y_{(k+1)d}$ es la salida deseada en el tiempo $k + 1$ y $e_k = y_k - y_{(k)d}$ es el error de seguimiento en el tiempo k . Para que el error tienda a cero es necesario que las raíces del polinomio $z^{p+1} + k_1 z^p + k_2 z^{p-1} + \cdots + k_{p+1}$ tengan valor absoluto menor a 1.

Si el error de aproximación ϵ_k de la ecuación (4.34) es cero, entonces se cumple que

$$\lim_{k \rightarrow \infty} e_k = 0, \quad (4.45)$$

tal y como ocurre en la ecuación (2.52). Si el error de aproximación ϵ_k no es cero pero es acotado, entonces el error de aproximación será acotado, tal y como ocurre en la ecuación (2.60).

Capítulo 5

Implementación

En este capítulo se presenta la validación experimental de los métodos de identificación y de los esquemas de control detallados en los capítulos anteriores. En la primera sección se hablará del equipo usado y como se conectó, en la sección posterior se muestran distintos modelos del sistema y finalmente se reporta el desempeño de varios esquemas de control. El objetivo principal es la obtención de un modelo del sistema con la estructura de la red neurodifusa mostrada en el capítulo 4 y la implementación de un esquema de control basado en esta red. Los demás modelos y esquemas de control se usan para compararlos con los basados en la red neurodifusa.

5.1 Equipo

El sistema electromecánico elegido fue el motor de corriente directa MT150F, parte del sistema para experimentación MS150 de Feedback. Además del motor MT150F fue necesario el uso de otros módulos del sistema MS150, una computadora personal (donde se realizaron los programas necesarios), y una tarjeta de adquisición de datos NI USB-6009, que permitió la comunicación entre la computadora y el sistema elec-

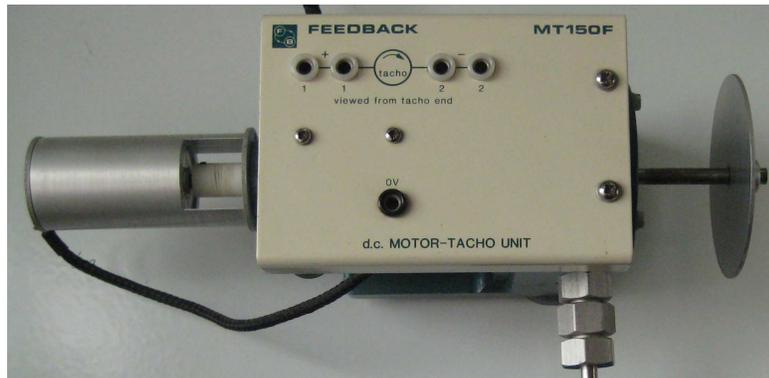


Figura 5.1: Motor MT150F.

tromecánico.

El paquete de computo Matlab se empleó para programar los algoritmos de identificación del sistema fuera de línea y para la simulación de los diversos esquemas de control. Para la implementación en tiempo real de los esquemas de control se eligió utilizar el lenguaje de programación Visual Basic, ya que permitía un manejo más sencillo de la tarjeta de adquisición de datos.

5.1.1 Motor y Módulos

El motor MT150F es un motor de corriente directa que forma parte del sistema MS150 de Feedback.inc. El sistema MS150 es un sistema desarrollado con fines didácticos; está formado por varios módulos y con él se pueden realizar diversas prácticas de control de velocidad y posición del motor MT150F. Los módulos del sistema MS150 que fueron utilizados son los siguientes:

- MT150F. Motor de corriente directa mostrado en la figura 5.1, tiene acoplado un taco-generator. El taco-generator genera una diferencia de potencial proporcional a la velocidad angular del motor. La ecuación que nos permite encontrar

la velocidad de la flecha del motor sabiendo la diferencia de potencial generada en el taco-generador, es la siguiente

$$V = 34.9D, \quad (5.1)$$

donde V es la velocidad angular en del motor rad/s y D es la diferencia de potencial en volts generada en el taco-generador del motor MT150F.

- PS150E. Es una fuente de poder con la que se alimentan los demás módulos del sistema.
- SA150D. Amplificador de potencia con el que se alimenta al motor, además este módulo permite realizar las conexiones necesarias para controlar el motor por armadura o por campo.
- OA150A Sumador inversor de ganancia unitaria hecho a partir de amplificadores operacionales 741.
- PA150C. Pre-amplificador que convierte la señal de control en dos señales que pueden ser introducidas al amplificador SA150D para controlar al motor.
- Freno magnético.

El freno magnético se usa para generar perturbaciones en el sistema. Al colocar el freno y el disco de aluminio conectado a la flecha del motor tal y como se muestra en la figura 5.3, el freno establece un campo magnético perpendicular al plano en el que se encuentra el disco. Cuando el motor gira el campo magnético del freno, se induce una diferencia de potencial entre el centro y la orilla del disco, la cual provoca una corriente eléctrica en el disco. La interacción entre la corriente y el campo magnético genera un par que se opone al movimiento rotacional de la flecha.

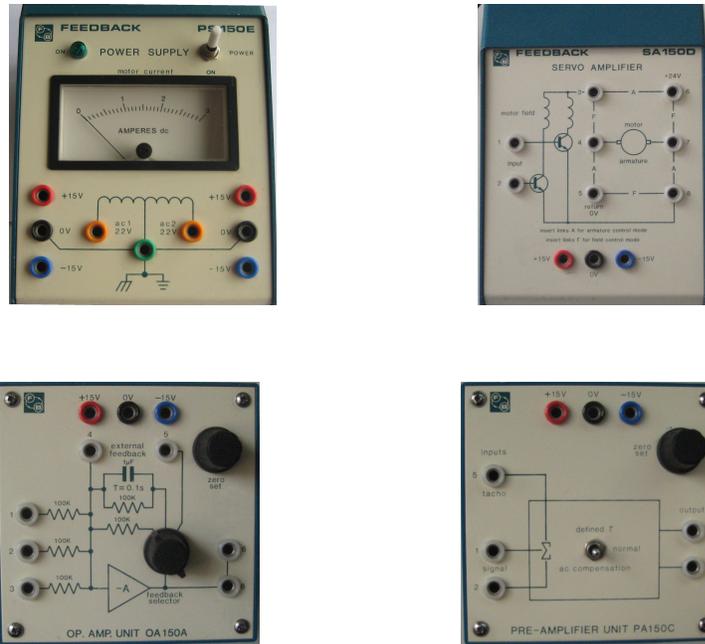


Figura 5.2: Módulos PS150E, SA150D, OA150A y PA150C.



Figura 5.3: Freno magnético.



Figura 5.4: Tarjeta NI USB-6009.

5.1.2 Tarjeta de Adquisición de Datos

La tarjeta de adquisición de datos usada es la tarjeta NI USB-6009 de National Instruments. Esta tarjeta se conecta con una computadora personal por medio del puerto USB, cuenta con puertos de entrada y salida tanto analógicos como digitales, referencias de voltaje de 2.5 y 5 volts y con contador digital. Para la realización de los experimentos de esta tesis no fue necesario usar los puertos digitales ni el contador digital.

El puerto analógico de entrada cuenta con un convertidor analógico-digital de aproximaciones sucesivas de 14 bits y puede leer datos en un rango de hasta $\pm 20 V$. La tarjeta puede adquirir 48000 muestras/s. El puerto analógico de salida cuenta con un convertidor digital-analógico de aproximaciones sucesivas de 12 bits y su rango de salida es de 0 a 5 V. La tarjeta puede escribir 300 muestras/s.

5.1.3 Conexión del Motor con la Computadora

El tiempo de muestreo con el que es usado en cada uno de los experimentos es de 5 ms, tiempo necesario para la lectura de un dato por medio de la tarjeta de

adquisición datos, el procesamiento para la generación de un nuevo dato de salida y la escritura del dato de salida en la tarjeta de adquisición de datos.

Dado que la diferencia de potencial máxima que se genera en el taco-generador es de $13 V$, el rango de $\pm 20 V$ del puerto analógico de entrada resulta adecuado. La diferencia de potencial que es leída en el taco-generador contiene ruido (parte de ese ruido se debe al alambrado). Para eliminar este ruido se usa un filtro digital que está programado en cada uno de los programas que leen la diferencia de potencial generada en el taco-generador.

El filtro que se eligió usar es un filtro Butterworth pasabajas de primer orden, con una frecuencia de corte de $53 Hz$. Este filtro fue diseñado para dejar pasar las señales a las frecuencias de operación del sistema y para atenuar el ruido causado entre otros factores por la señal de $60 Hz$ de la instalación eléctrica. La ecuación en diferencias de dicho filtro es la siguiente

$$y_i = 0.5236x_i + 0.5236x_{i-1} + 0.0472y_{i-1}, \quad (5.2)$$

donde y es la señal filtrada y x es la señal no filtrada.

La señal de control, que es generada por medio del software de acuerdo con los distintos esquemas de control, puede tomar valores menores que cero; ya que la el rango del puerto de salida analógico es de 0 a $5 V$ es necesario de adecuar esta señal de algún modo.

Para poder ser mandada por el puerto analógico de salida la señal de control se multiplica por -1 y se le suma un offset de $2.5 V$, por medio de software. Además el rango de la señal de control se restringe a $[-0.23, 0.23] V$. Este rango se escogió ya que para valores fuera de este intervalo el módulo pre-amplificador PA150C se satura. La referencia de $2.5 V$ de la tarjeta NI USB-6009 y dos de los módulos sumadores inversores OA150A son usados para recuperar la señal de control que se introduce

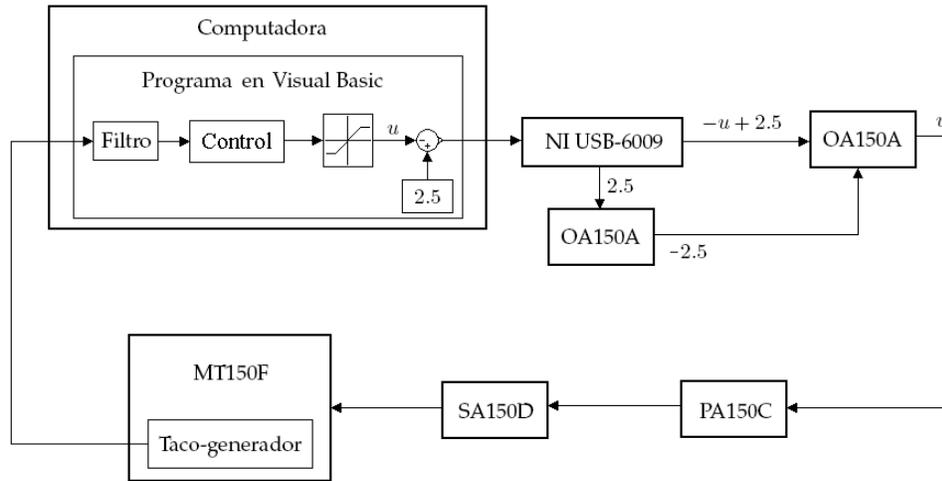


Figura 5.5: Conexión del motor con la computadora.

al módulo PA150C. En la figura 5.5 se muestra un diagrama de bloques de como se conecta el motor con la computadora.

5.2 Identificación del Sistema Electromecánico

En esta sección se muestran los resultados de la identificación del sistema electromecánico usando las técnicas descritas en las secciones 3.1 y 3.3. El sistema electromecánico es un sistema SISO cuya entrada es el la diferencia de potencial que se aplica al módulo PA150C y cuya salida es la diferencia de potencial *normalizada* generada en el taco-generador que, como ya se explico anteriormente, es lineal con respecto a la velocidad del motor. El módulo SA150D se conecta de modo que se realice un control por armadura del motor.

La máxima velocidad a la que puede girar el motor es aproximadamente 453.7 rad/s , por lo tanto la diferencia de potencial máxima que se genera en el taco-generador es de 13 V . Para facilitar el manejo de las cantidades durante el proceso

de identificación y control, la diferencia de potencial generada en el taco-generador se divide entre $13 V$ para obtener la diferencia de potencial normalizada que es adimensional.

5.2.1 Identificación del Sistema Electromecánico Usando la Respuesta al Escalón

El primer modelo que es usado para identificar el sistema electromecánico es un modelo lineal de primer orden. El objetivo es encontrar una función de transferencia como la que se muestra en la ecuación 2.2 usando el método descrito en la sección 3.1.1.

La respuesta del sistema a una entrada $0.15u_{-1}(t)$ (escalón con amplitud de $0.15 V$) se muestra con línea sólida en la figura 5.6. Inspeccionando la gráfica de la figura 5.6 se determinó el valor que deben de tomar las constantes k y τ , por lo tanto la función de transferencia del modelo es

$$G(s) = \frac{4.03}{0.492s + 1}. \quad (5.3)$$

La respuesta del modelo a la entrada $0.15u_{-1}(t)$ se muestra con línea punteada en la figura 5.6. Se puede ver que el la función de transferencia de la ecuación 5.3 es un modelo que aproxima aceptablemente al sistema.

La figura 5.7 muestra las respuestas del sistema electromecánico (línea sólida) y del modelo de primer orden (línea punteada) cuando la entrada es $0.075u_{-1}(t)$ y $0.23u_{-1}(t)$. La capacidad del modelo para describir al sistema se ve disminuida cuando la entrada es distinta de $0.15u_{-1}(t)$, se pueden ajustar de nuevo los parámetros para que el modelo aproxime el comportamiento del sistema cuando la entrada es $0.075u_{-1}(t)$ o $0.23u_{-1}(t)$, pero los nuevos modelos no aproximan adecuadamente al sistema cuando le entrada es $0.15u_{-1}(t)$.

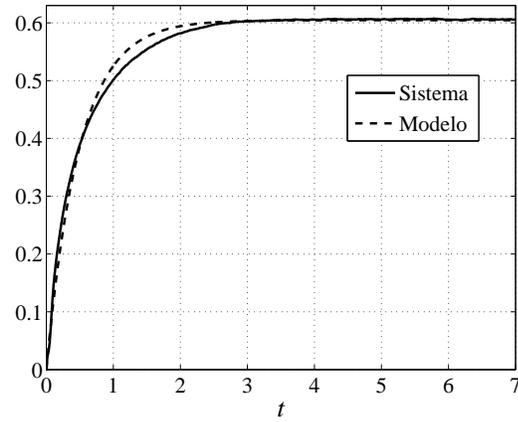


Figura 5.6: Respuesta del sistema electromecánico y del modelo de primer orden a $0.15u_{-1}(t)$.

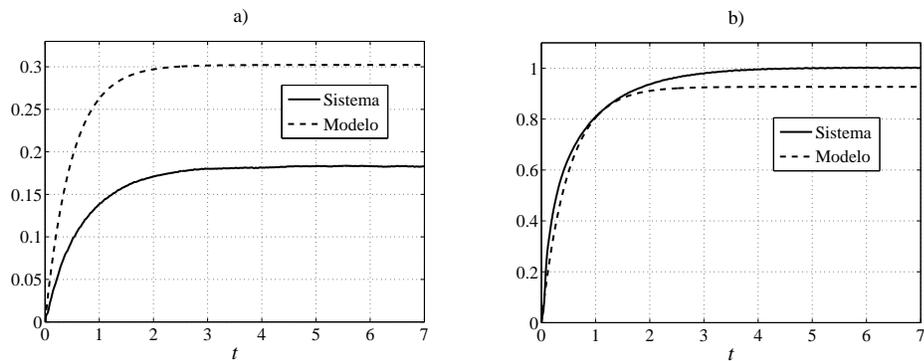


Figura 5.7: a) Respuesta del sistema electromecánico y del modelo a $0.075u_{-1}(t)$. b) Respuesta del sistema electromecánico y del modelo a $0.23u_{-1}(t)$.

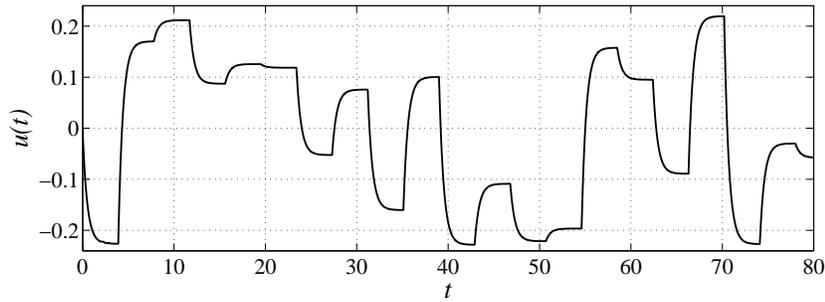


Figura 5.8: Entrada de excitación.

5.2.2 Identificación del Sistema Electromecánico Usando Mínimos Cuadrados

En aras de obtener un modelo más confiable del sistema electromecánico, fue usado el método de los mínimos cuadrados recursivo descrito en la sección 3.1.3. El modelo que se encontró usando este método fue un modelo lineal en tiempo discreto.

La estructura del modelo lineal en tiempo discreto es la siguiente

$$\begin{aligned}
 \hat{y}_{k+1} &= c_1 \hat{y}_k + \dots + c_n \hat{y}_{k-n+1} + h_1 u_k + \dots + h_m u_{k-m+1}, \\
 &= [c_1, \dots, c_n] \begin{bmatrix} \hat{y}_k \\ \vdots \\ \hat{y}_{k-n+1} \end{bmatrix} + [h_1, \dots, h_m] \begin{bmatrix} u_k \\ \vdots \\ u_{k-m+1} \end{bmatrix}, \quad (5.4)
 \end{aligned}$$

donde \hat{y} es la salida, u es la entrada y c_i y h_j son parámetros que tienen que ser estimados.

Para obtener un conjunto de pares de datos entrada-salida $\{u^i, y^i\}$ y poder aplicar el método de los mínimos cuadrados recursivo, se excitó al sistema con la entrada mostrada en la figura 5.8 y se midió su salida.

El número de retrasos en la entrada m y el número de retrasos en la salida n que debería tener el modelo no eran conocidos. Se estimaron los parámetros para

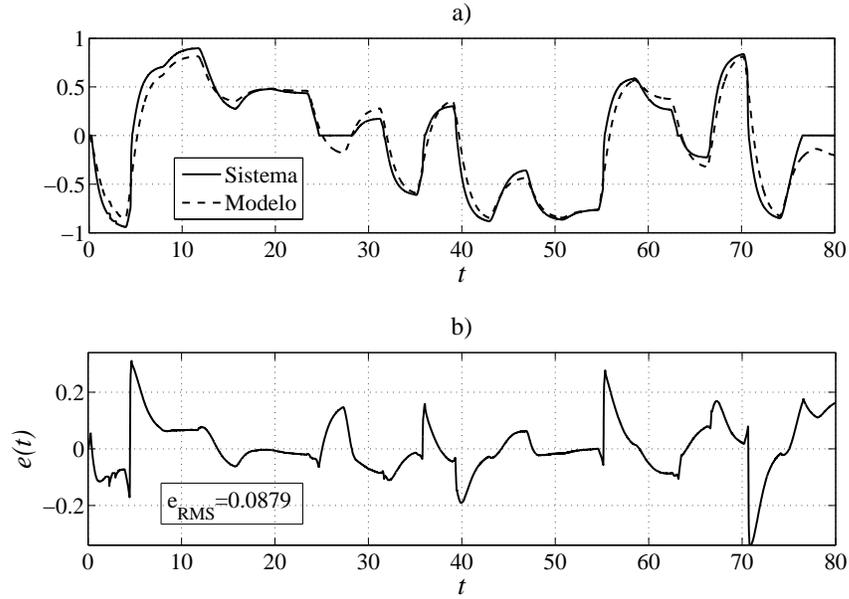


Figura 5.9: a) Respuesta del sistema electromecánico y del modelo lineal en tiempo discreto a la entrada de excitación. b) Error de aproximación.

distintos modelos, cada uno con distintos valores de m y n , el modelo que mejor aproximó al sistema es aquel en el que $n = 3$ y $m = 2$. Los valores de los parámetros obtenidos son

$$\begin{aligned} c_1 &= 2.2208, & c_2 &= -1.5311, & c_3 &= 0.3098, \\ h_1 &= 0.0323, & h_2 &= -0.0305, \end{aligned} \quad (5.5)$$

por lo tanto el modelo en tiempo discreto del sistema es

$$\hat{y}_{k+1} = \begin{bmatrix} 2.2208, -1.5311, 0.3098 \end{bmatrix} \begin{bmatrix} \hat{y}_k \\ \hat{y}_{k-1} \\ \hat{y}_{k-2} \end{bmatrix} + \begin{bmatrix} 0.0323, -0.0305 \end{bmatrix} \begin{bmatrix} u_k \\ u_{k-1} \end{bmatrix}. \quad (5.6)$$

Las salidas del sistema electromecánico (línea sólida) y del modelo en tiempo discreto (línea punteada) cuando la entrada de excitación es la que se muestra en la

figura 5.8 se muestran en la figura 5.9, junto con el error de aproximación. El error de aproximación tiene un valor rms de 0.0879.

5.2.3 Identificación del Sistema Electromecánico Usando la Red Neurodifusa

El último y más elaborado modelo que fue empleado para identificar al sistema electromecánico es la red neurodifusa recurrente entrada-salida descrita en la sección 4.1. Para entrenar dicha red se usó el filtro de Kalman, tal y como se describe en la sección 4.3. Para poder entrenar a la red fueron usados los pares de datos entrada-salida $\{u^i, y^i\}$ que se obtuvieron cuando se excitó al sistema con la entrada mostrada en la figura 5.8.

En número de reglas difusas que tiene la red con la que se identificó al sistema es $n_R = 3$. Cada regla difusa tiene en su parte consecuente un modelo lineal en tiempo discreto, los números de retrasos en la entrada y en la salida se escogieron como $n = 3$ y $m = 2$, ya que estos valores mostraron ser adecuados en la sección anterior.

Para inicializar los parámetros de la red cerca de algún buen mínimo de la función de costo, se siguió el algoritmo descrito en la sección 4.6. Los parámetros que se obtuvieron siguiendo este algoritmo son los siguientes:

$$C = \begin{bmatrix} 2.894 & -2.8341 & 0.9124 \\ 2.8609 & -2.5287 & 0.7977 \\ 2.1351 & -1.4191 & 0.2783 \end{bmatrix}, \quad H = \begin{bmatrix} 0.0041 & -0.0015 \\ 0.05 & -0.0234 \\ 0.032 & -0.036 \end{bmatrix}, \quad (5.7)$$

$$\sigma = \begin{bmatrix} 5 \\ 25 \\ 5 \end{bmatrix}, \quad \varsigma = \begin{bmatrix} -0.8 \\ 0 \\ 0.8 \end{bmatrix}. \quad (5.8)$$

El valor inicial de P fue $P = 10^8 I$ y los valores de R_1 y R_2 fueron escogidos como $10^{-8} I$ y 0.995 respectivamente.

Las primeras veces que fue realizado el proceso de entrenamiento los parámetros ς_i adquirieron valores con valores absolutos demasiado grandes, lo cual coloca a los conjuntos difusos fuera de la región de operación del sistema. Para evitar este detalle se decidió fijar estos parámetros, de tal suerte que siempre tendrían los siguientes valores $\varsigma_1 = -1$, $\varsigma_2 = 0$ y $\varsigma_3 = 1$. También fue mantenido estático $\sigma_2 = 25$, esto se hizo para que el conjunto A_2 fuera muy estrecho (además de estar centrado en 0 gracias a que $\varsigma_2 = 0$) y la red tuviera un mejor desempeño al identificar el sistema en regiones donde se presenta fricción seca y en regiones donde la entrada es muy pequeña y produce una salida igual a cero. Este último tipo de comportamiento se puede observar en la figura 5.9 a) cerca del segundo 26.

Otro detalle que fue evitado fue que los parámetro h_{i1} tomaran valores con signos distintos. Si los parámetros h_{i1} tienen signos distintos, el valor de $\varphi_k^T h_{c1}$, necesario para realizar un control de equivalencia cierta (ver ecuación (4.43)); puede llegar a ser cero. Esto provoca que la señal de control u_{ck} , generada por el control de equivalencia cierta basado en la red, se indetermine.

Los parámetros que se obtuvieron al finalizar el entrenamiento de la red son los siguientes:

$$C = \begin{bmatrix} 2.8603 & -2.7465 & 0.8859 \\ 2.8714 & -2.7525 & 0.8810 \\ 2.1083 & -1.3171 & 0.2083 \end{bmatrix}, \quad H = \begin{bmatrix} 0.0022 & -0.0013 \\ 0.0141 & -0.0140 \\ 0.0376 & -0.0357 \end{bmatrix}, \quad (5.9)$$

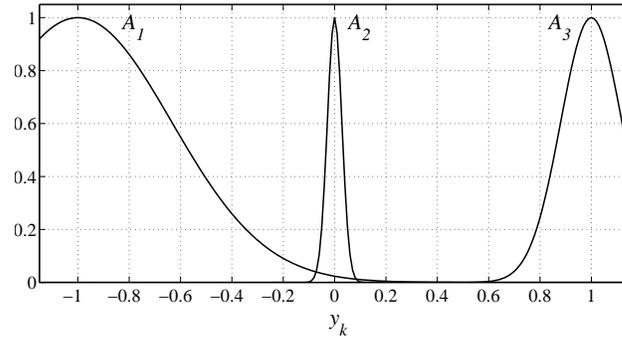


Figura 5.10: Conjuntos difusos A_j .

$$\sigma = \begin{bmatrix} 1.9361 \\ 25 \\ 5.9324 \end{bmatrix}, \quad \varsigma = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}. \quad (5.10)$$

Los conjuntos difusos de la parte antecedente de las tres reglas difusas de la red neurodifusa se muestran en la figura 5.10. Las salidas del sistema electromecánico (línea sólida) y de la red neurodifusa (línea punteada) cuando la entrada de excitación es la que se muestra en la figura 5.8 se muestran en la figura 5.11, junto con el error de aproximación. El error de aproximación tiene un valor RMS de 0.0693.

Identificación del Sistema Electromecánico Usando la Red Neurodifusa con Restricciones

Para evitar que los parámetros ς tomen valores con valores absolutos grandes y que los parámetros h_{1i} tomen valores con signos distintos, se implementan restricciones como las descritas en la sección 4.5.

Los valores iniciales de los parámetros ς y h_{1i} fueron tomados de las ecuaciones (5.7) y (5.8). Aplicando las restricciones, los parámetros ς y h_{1i} quedan definidos de

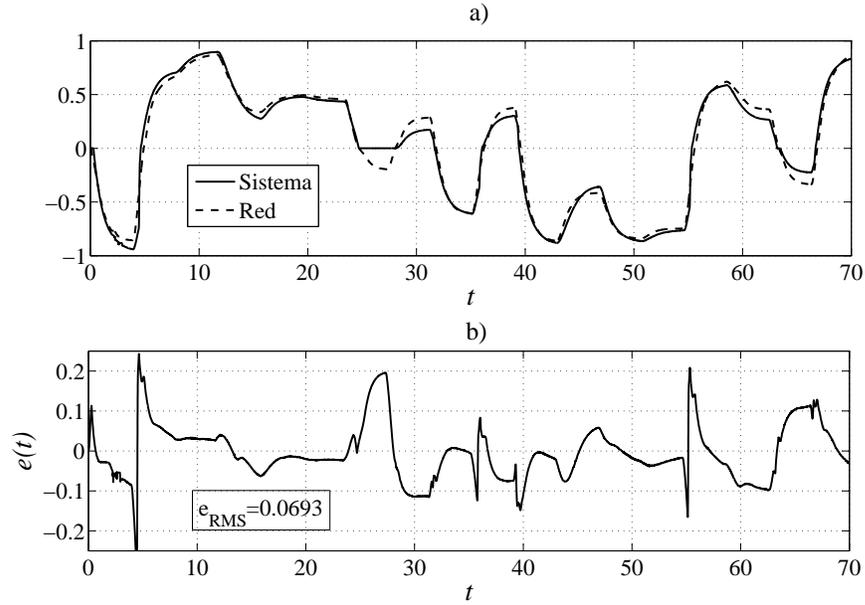


Figura 5.11: a) Respuesta del sistema electromecánico y de la red neurodifusa a la entrada de excitación. b) Error de aproximación.

la siguiente manera:

$$\varsigma_1 = 0.4 \tanh(\check{\zeta}_1 + 0.8) - 0.8, \quad (5.11)$$

$$\varsigma_2 = 0.4 \tanh(\check{\zeta}_2), \quad (5.12)$$

$$\varsigma_3 = 0.4 \tanh(\check{\zeta}_1 - 0.8) + 0.8, \quad (5.13)$$

$$h_{11} = 0.004 \tanh(\check{h}_{11} - 0.0041) + 0.0041, \quad (5.14)$$

$$h_{21} = 0.049 \tanh(\check{h}_{21} - 0.05) + 0.05, \quad (5.15)$$

$$h_{31} = 0.031 \tanh(\check{h}_{31} - 0.032) + 0.032. \quad (5.16)$$

Por lo tanto, se garantiza que:

$$\varsigma_1 \in (-1.2, -0.4), \quad \varsigma_2 \in (-0.4, 0.4), \quad \varsigma_3 \in (0.4, 1.2),$$

$$h_{11} \in (0, 0.0082), \quad h_{21} \in (0, 0.1), \quad h_{31} \in (0, 0.064).$$

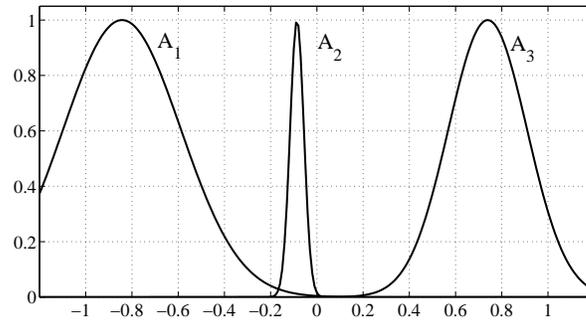


Figura 5.12: Conjuntos difusos A_i de la red restringida.

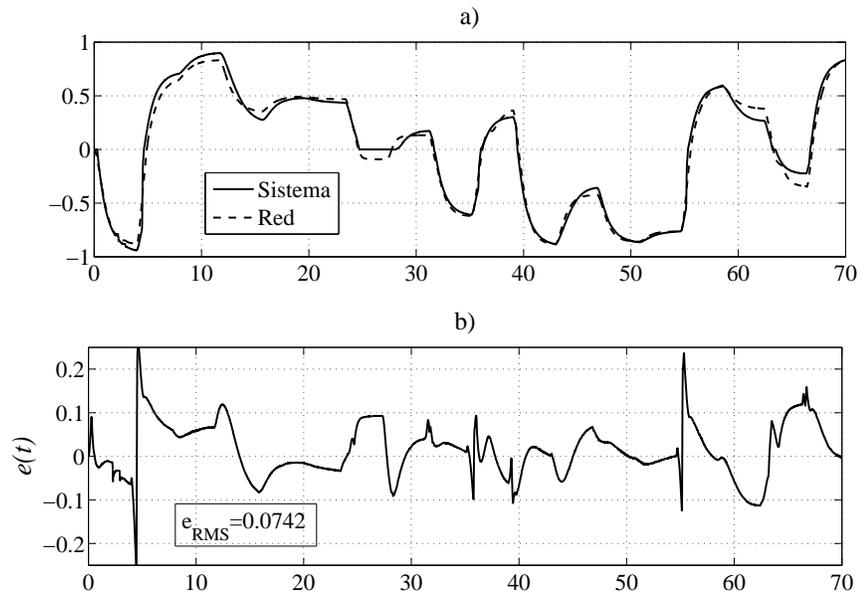


Figura 5.13: a) Respuesta del sistema electromecánico y de la red neurodifusa con restricciones. b) Error de aproximación.

Los valores iniciales de los parámetros fueron los de las ecuaciones (5.7) y (5.8). En este entrenamiento el parámetro σ_2 no fue mantenido estático. Los parámetros que se obtuvieron al finalizar el entrenamiento de la red con restricciones son los siguientes:

$$C = \begin{bmatrix} 2.9075 & -2.8413 & 0.9335 \\ 2.7275 & -2.483 & 0.7552 \\ 2.2601 & -1.5487 & 0.2884 \end{bmatrix}, \quad H = \begin{bmatrix} 0.0029 & -0.0012 \\ 0.0452 & -0.0446 \\ 0.033 & -0.0324 \end{bmatrix}, \quad (5.17)$$

$$\sigma = \begin{bmatrix} 2.784 \\ 23.73 \\ 4.162 \end{bmatrix}, \quad \varsigma = \begin{bmatrix} -0.843 \\ 0.086 \\ 0.739 \end{bmatrix}. \quad (5.18)$$

Los conjuntos difusos de la parte antecedente de las tres reglas difusas de la red neurodifusa con restricciones se muestran en la figura 5.12. Las salidas del sistema electromecánico (línea sólida) y de la red neurodifusa con restricciones (línea punteada) cuando la entrada de excitación es la que se muestra en la figura 5.8 se muestran en la figura 5.13, junto con el error de aproximación. El error de aproximación tiene un valor RMS de 0.0742.

5.2.4 Comparación entre Modelos

Para realizar una comparación adecuada de los cuatro modelos anteriores fue obtenida la respuesta del modelo lineal de primer orden en tiempo continuo de la ecuación (5.3) a la entrada es la mostrada en la figura 5.8.

En la figura 5.14 se muestran las salidas de los cuatro modelos (obtenidos en las tres secciones anteriores) cuando la entrada es la que se muestra en la figura 5.8, com-

paradas con la salida del sistema electromecánico, además se muestran los errores de aproximación con sus respectivos valores RMS.

El valor RMS del error de aproximación es un buen indicador de la capacidad de aproximación de cada modelo (ver [2] y [3]). Entre menor sea el valor RMS del error de aproximación mejor será la capacidad del modelo para aproximar al sistema. El modelo con el valor RMS del error de aproximación más grande es el modelo lineal de primer orden. El valor RMS del error de aproximación más pequeño se encuentra cuando se usa la red neurodifusa sin restricciones.

El valor RMS del error de aproximación de la red neurodifusa con restricciones es más grande que el de la red neurodifusa sin restricciones, en este caso complicar un poco más el entrenamiento no mejoro el desempeño de la red. Pero si observamos las figuras 5.12 y 5.10 podemos ver que cuando existen restricciones los conjuntos difusos permanecen mejor colocados dentro de la región de operación del sistema.

En la figura 5.14 podemos observar que los picos del error de aproximación se presentan, principalmente, cuando hay picos en la derivada de la señal de entrada, o dicho con otras palabras, cuando existen cambios más bruscos con la señal de entrada.

5.3 Control del Sistema Electromecánico

En esta sección se valorarán diversas estrategias de control, PI, controles difusos, control por asignación de polos y control de equivalencia cierta usando la red neurodifusa. En todos los casos, se analiza su respuesta ante una señal cuadrada, ante un señal que es una suma de escalones y ante una señal escalón cuando existe una perturbación generada por medio de del freno magnético.

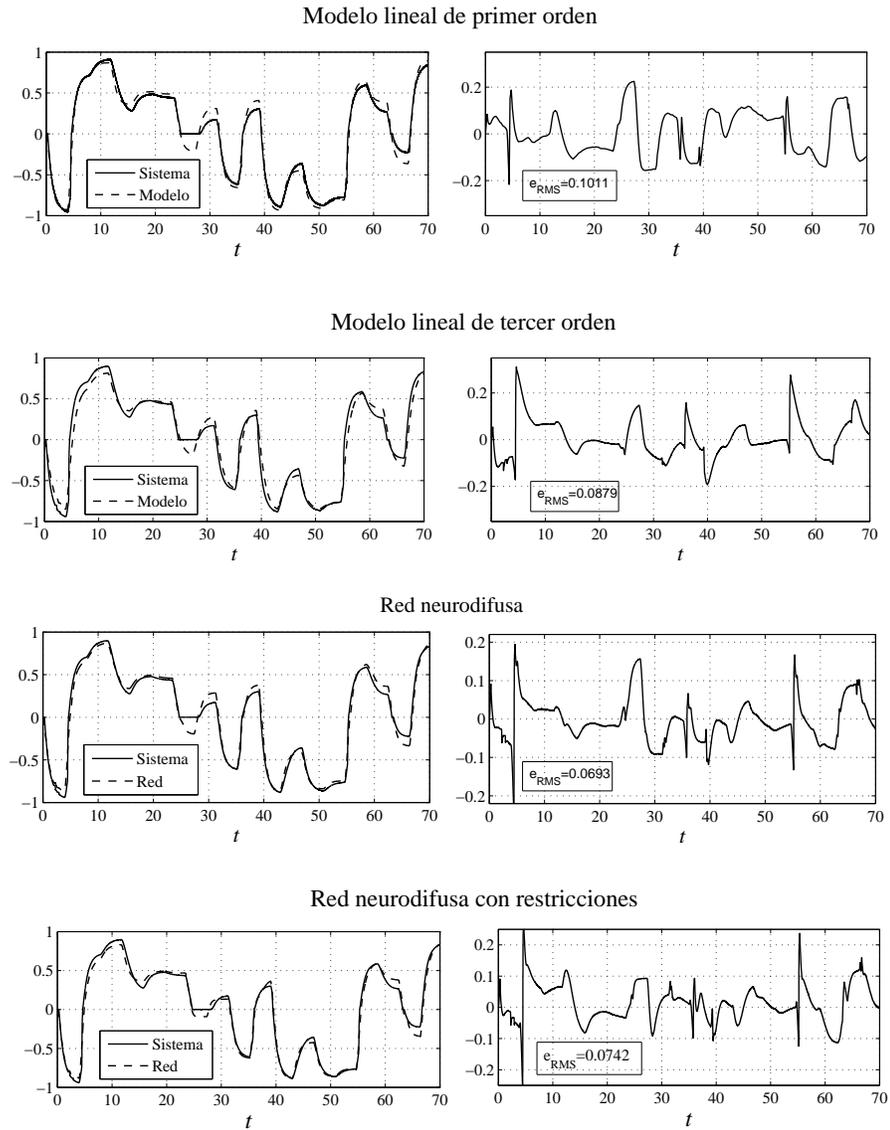


Figura 5.14: Repuesta de los cuatro modelos.

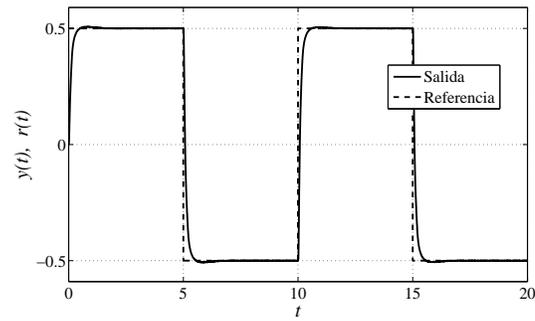


Figura 5.15: Respuesta del sistema en lazo cerrado con un controlador PI a una señal de referencia cuadrada.

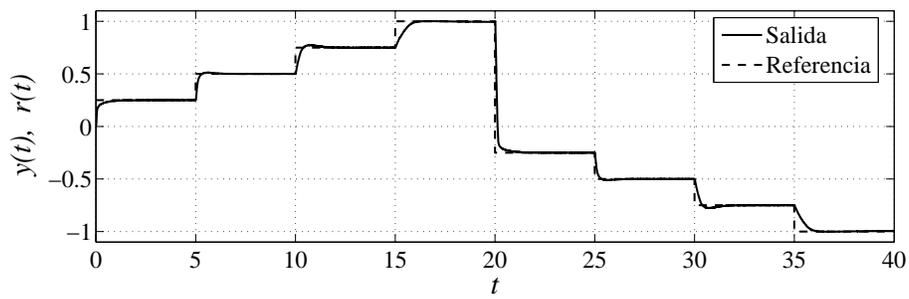


Figura 5.16: Respuesta del sistema en lazo cerrado con un controlado PI cuando la referencia es una suma de escalones.

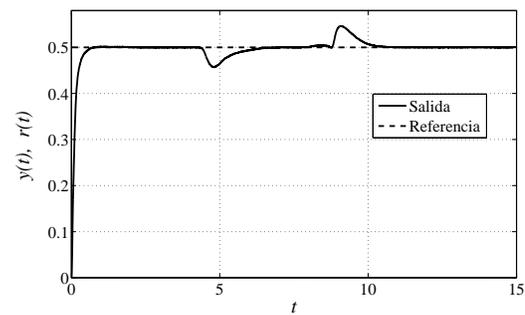


Figura 5.17: Respuesta del sistema con un controlador PI cuando hay una perturbación.

5.3.1 Control PI

El primer esquema de control que fue implementado fue un controlador PI. Fue elegido un controlador PI dado que el sistema en lazo cerrado sin control no tiene problemas de con el sobrepaso, pero si los tiene con el error en estado estable. El cómputo de la integral fue realizado usando la aproximación trapezoidal de la integral.

Las ganancias con las que se obtuvo el mejor desempeño del sistema en lazo cerrado fueron $K_p = 1$ y $K_i = 2$. La respuesta del sistema en lazo cerrado cuando la señal de referencia es una señal cuadrada con amplitud de 0.5 se muestra en la figura 5.15. La figura 5.16 muestra la respuesta del sistema en lazo cerrado cuando la referencia es una suma de señales escalón. Es importante recordar que la salida del sistema electromecánico es la diferencia de potencial *normalizada* generada en el taco-generador, por lo tanto cuando la salida del sistema alcanza la referencia de 0.5, la velocidad angular del motor es 226.85 rad/s .

La figura 5.17 muestra cual es la salida del sistema cuando hay una perturbación, producida por el freno magnético, y la señal de referencia es $0.15u_{-1}(t)$ (escalón con amplitud de 0.5). El freno es aplicado alrededor del segundo 5 y es retirado alrededor del segundo 10.

5.3.2 Control Difuso Tipo Mamdani

El segundo esquema de control que fue implementado en el sistema electromecánico fue un controlador difuso tipo Mamdani. En este esquema de control se coloca un sistema difuso tipo Mamdani, como el descrito en la sección 2.2.4 y representado en la figura 2.8, en el lugar del bloque "control" de la figura 3.2.

Las variables lingüísticas que son ponderadas en la parte antecedente de las reglas

difusas son el error (er) y la integral del error (ie). Fueron planteados seis conjuntos difusos, tres para cada variable lingüística. Los conjuntos difusos para el error son *grande negativo* (EGN), *pequeño* (EP) y *grande positivo* (EGP); estos conjuntos se muestran en la figura 5.18. Por otro lado los conjuntos difusos para la integral error son *grande negativa* (IGN), *pequeña* (IP) y *grande positiva* (IGP); estos conjuntos se muestran en la figura 5.19.

$$EGN(er) = \frac{1}{1 + e^{-100(-er-0.01)}} , \quad (5.19)$$

$$EP(er) = e^{-100er^2} , \quad (5.20)$$

$$EGP(er) = \frac{1}{1 + e^{-100(er-0.01)}} , \quad (5.21)$$

$$IGN(ie) = \frac{1}{1 + e^{-100(-ie-0.02)}} , \quad (5.22)$$

$$IP(ie) = e^{-300ie^2} , \quad (5.23)$$

$$IGP(ie) = \frac{1}{1 + e^{-100(ie-0.02)}} . \quad (5.24)$$

El universo de discurso de la señal de control (salida del sistema difuso) fue definido como $[-0.3, 0.3]$ V. Recordemos que el valor de la señal de control está restringido al intervalo $[-0.23, 0.23]$ V; si el universo de discurso se hubiera definido de esta manera, las salidas del sistema difuso hubieran sido distintas ya que estas se calcularon usando el método del centroide. Los conjuntos difusos para la señal de control (u) también son tres y son *grande negativa* (UGN), *pequeña* (UP) y *grande positiva* (UGP); estos conjuntos se muestran en la figura 5.20.

$$UGN(ie) = \frac{1}{1 + e^{-50(-u-0.2)}} , \quad (5.25)$$

$$UP(ie) = e^{-400u^2} , \quad (5.26)$$

$$UGP(ie) = \frac{1}{1 + e^{-50(u-0.2)}} . \quad (5.27)$$

La base de conocimiento usada es la siguiente:

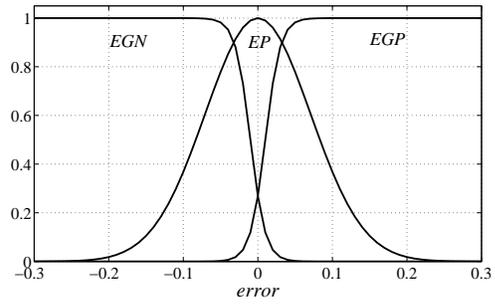


Figura 5.18: Conjuntos difusos para el error.

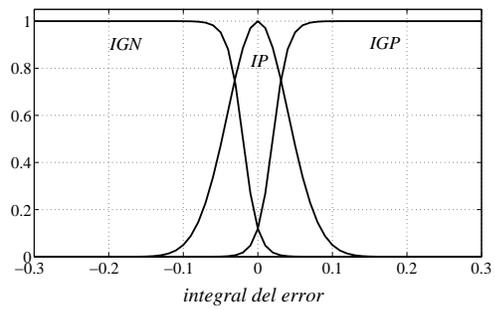


Figura 5.19: Conjuntos difusos para la integral del error.

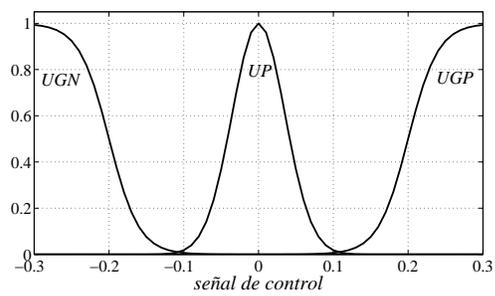


Figura 5.20: Conjuntos difusos para la señal de control.

- R_1 : Si el *error* es grande negativo y la *integral del error* es grande negativa entonces u es grande negativa.
- R_2 : Si el *error* es grande negativo y la *integral del error* es pequeña entonces u es grande negativa.
- R_3 : Si el *error* es grande negativo y la *integral del error* es grande positiva entonces u es grande negativa.
- R_4 : Si el *error* es pequeño y la *integral del error* es grande negativa entonces u es grande negativa.
- R_5 : Si el *error* es pequeño y la *integral del error* es pequeña entonces u es pequeña.
- R_6 : Si el *error* es pequeño y la *integral del error* es grande positiva entonces u es grande positiva.
- R_7 : Si el *error* es grande positivo y la *integral del error* es grande negativa entonces u es grande positiva.
- R_8 : Si el *error* es grande positivo y la *integral del error* es pequeña entonces u es grande positiva.
- R_9 : Si el *error* es grande positivo y la *integral del error* es grande positiva entonces u es grande positiva.

El proceso de agregación es realizado usando la t-norma mínimo, la activación es realizado usando la t-norma producto algebraico, para la acumulación se usa la t-conorma máximo y la desdifusión es realizado usando el método del centroide.

La respuestas del sistema en lazo cerrado con un controlador difuso tipo Mamdani cuando las señales de referencia son una señal cuadrada con amplitud de 0.5,

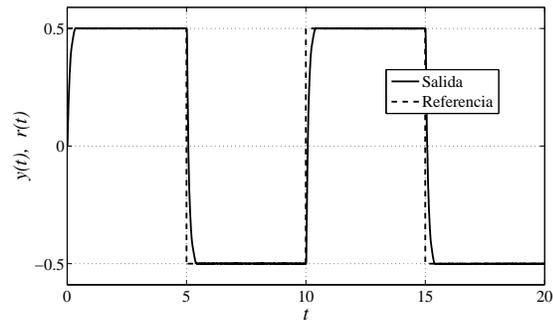


Figura 5.21: Respuesta del sistema en lazo cerrado con un controlador difuso tipo Mamdani a una señal de referencia cuadrada.

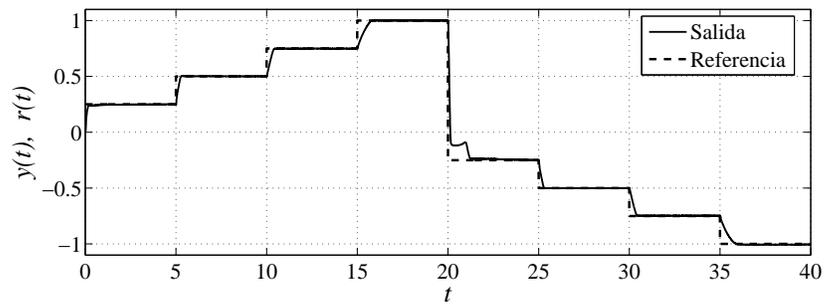


Figura 5.22: Respuesta del sistema en lazo cerrado con un controlador difuso tipo Mamdani cuando la referencia es una suma de escalones.

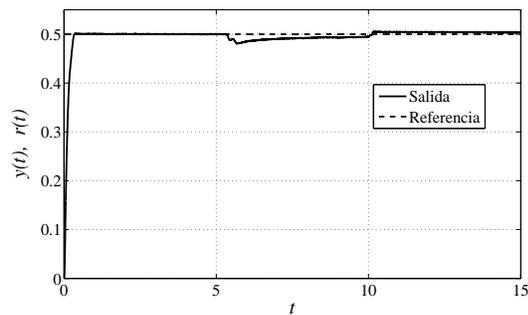


Figura 5.23: Respuesta del sistema con un controlador difuso tipo Mamdani cuando hay una perturbación.

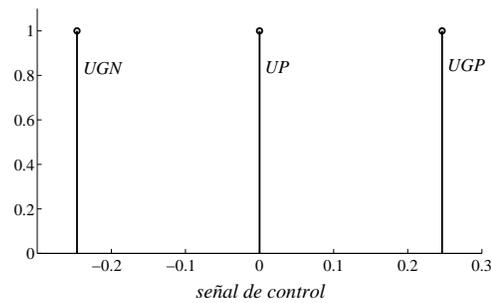


Figura 5.24: Conjuntos tipo singleton para la señal de control.

una suma de señales escalón y $0.15u_{-1}(t)$ (aplicando el freno magnético aproximadamente al mismo tiempo que en el caso del controlador PI); se muestran en las figuras 5.21, 5.22 y 5.23 respectivamente.

5.3.3 Control Difuso Tipo Singleton

El proceso más complejo de implementar en un controlador difuso tipo Mamdani es la desdifusión el cual puede ser computacionalmente pesado. El proceso de desdifusión es mucho más simple cuando el sistema difuso es de tipo singleton.

Ya que el controlador difuso tipo Mamdani de la sección anterior hizo que el sistema electromecánico en lazo cerrado tuviera un desempeño bastante satisfactorio, el controlador difuso tipo singleton que fue implementado pretendió ser lo más parecido al controlador difuso tipo Mamdani. Lo único que fue modificado fueron los tres conjuntos difusos de la señal de salida (ver figura 5.20), los cuales se cambiaron por tres conjuntos tipo singleton. Para que los dos controladores fueran lo más parecidos posibles, los nuevos conjuntos tipo singletons se colocaron en la posición en la que se encuentran los centroides de los conjuntos difusos de la figura 5.20. Los conjuntos difusos tipo singleton obtenidos se muestran en la figura 5.24.

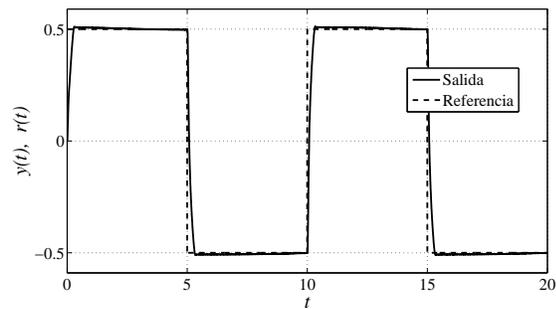


Figura 5.25: Respuesta del sistema en lazo cerrado con un controlador difuso tipo singleton a una señal de referencia cuadrada.

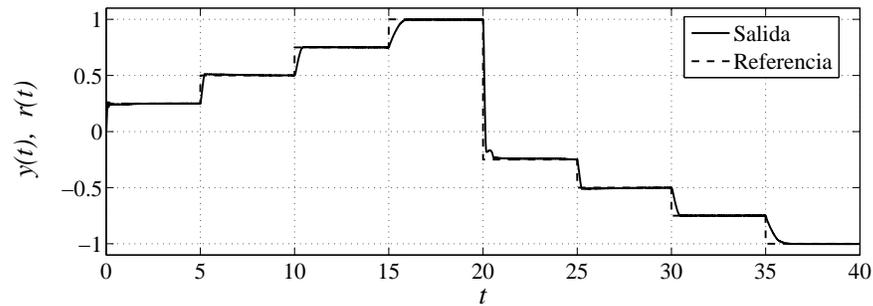


Figura 5.26: Respuesta del sistema en lazo cerrado con un controlador difuso tipo singleton cuando la referencia es una suma de escalones.

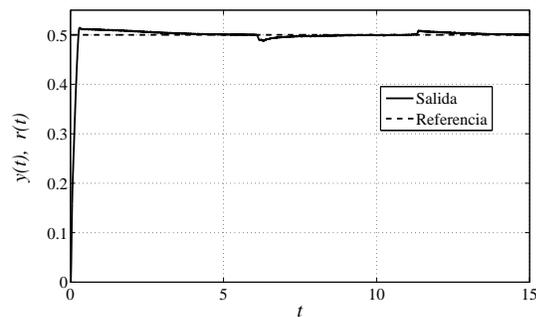


Figura 5.27: Respuesta del sistema con un controlador difuso tipo singleton cuando hay una perturbación.

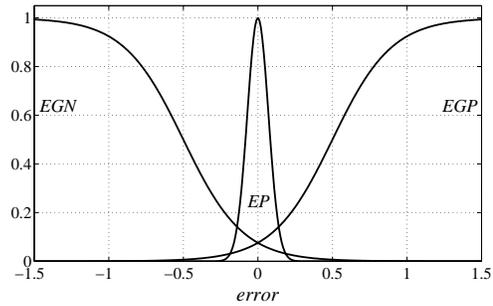


Figura 5.28: Conjuntos difusos para el error.

Las respuestas del sistema en lazo cerrado con un controlador difuso tipo singleton cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de señales escalón y $0.15u_{-1}(t)$ (aplicando el freno magnético aproximadamente al mismo tiempo que en los dos casos anteriores); se muestran en las figuras 5.25, 5.26 y 5.27 respectivamente.

5.3.4 Control Difuso Tipo Takagi-Sugeno

El siguiente esquema de control que fue implementado fue un controlador difuso tipo Takagi-Sugeno. En este esquema de control se coloca un sistema difuso tipo Takagi-Sugeno, como el descrito en la sección 2.2.5 en el lugar del bloque "control" de la figura 3.2.

La variable lingüística que fue ponderada en la parte antecedente de las reglas difusas fue el error (er). Los conjuntos difusos para el error son *grande negativo* (EGN), *pequeño* (EP) y *grande positivo* (EGP); estos conjuntos se muestran en la figura 5.28.

$$EGN(er) = \frac{1}{1 + e^{-5(-er-0.5)}} \quad , \quad (5.28)$$

$$EP(er) = e^{-100er^2} \quad , \quad (5.29)$$

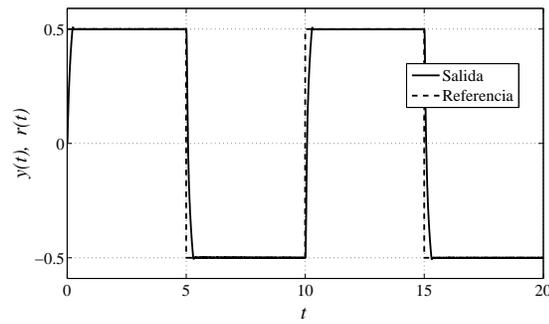


Figura 5.29: Respuesta del sistema en lazo cerrado con un controlador difuso tipo Takagi-Sugeno a una señal de referencia cuadrada.

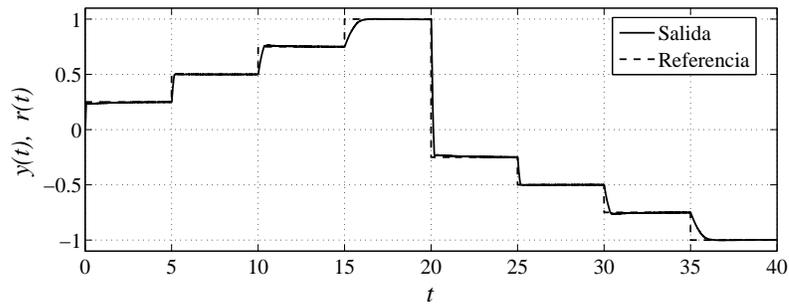


Figura 5.30: Respuesta del sistema en lazo cerrado con un controlador difuso tipo Takagi-Sugeno cuando la referencia es una suma de escalones.

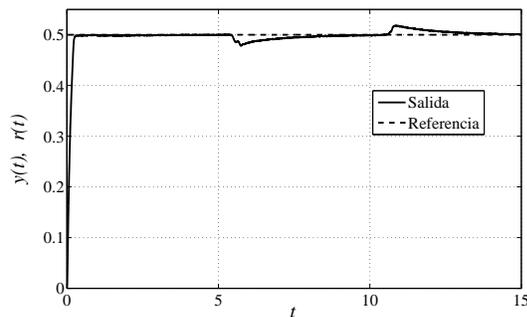


Figura 5.31: Respuesta del sistema con un controlador difuso tipo Takagi-Sugeno cuando hay una perturbación.

$$EGP(er) = \frac{1}{1 + e^{-5(er-0.5)}} \cdot \quad (5.30)$$

La base de conocimiento que fue empleada es la siguiente

- R_1 : Si el *error* (er) es grande negativo entonces $u = 15er + 2 \int er dt$
- R_2 : Si el *error* (er) es pequeño entonces $u = er + 2 \int er dt$
- R_3 : Si el *error* (er) es grande positivo entonces $u = 15er + 2 \int er dt$

Es importante notar que la regla R_2 contiene en su parte consecuente al controlador PI que ya ha sido usado para controlar al sistema electromecánico en la sección 5.3.1. Las reglas R_1 y R_3 contienen un controlador PI con una ganancia proporcional muy grande, que mejora el tiempo de subida pero que generaría un sobrepaso importante si no se usara en combinación con el controlador de la regla R_2 .

Las respuestas del sistema en lazo cerrado con un controlador difuso tipo Takagi-Sugeno, cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de señales escalón y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en las figuras 5.29, 5.30 y 5.31 respectivamente.

5.3.5 Control por Asignación de Polos

Los cuatro esquemas de control implementados en las secciones anteriores no hacen uso explícito de un modelo matemático del sistema para su diseño, aunque el contar con uno es de gran ayuda en la fase de simulación.

Para diseñar un control por asignación de polos se requiere contar un modelo en variables de estado del sistema. Partiendo de la función de transferencia de la ecuación (5.3), considerando a la velocidad angular del motor como único estado,

tomando en cuenta la ecuación (5.1) y recordando que la salida y del sistema electromecánico es la diferencia de potencial generada en el taco-generador dividida entre 13; podemos deducir el siguiente modelo en variables de estado del sistema electromecánico

$$\begin{aligned}\dot{x} &= 2.0325x + 3716.98u, \\ y &= 0.0022068x.\end{aligned}\tag{5.31}$$

El esquema usado es el que se muestra en la figura 2.5. Ya que solo hay un estado que realimentar, el vector K solo tiene un elemento (k). A las constantes k_a y k se les asignaron valores tales que la respuesta del sistema en lazo cerrado fuera estable, sobreamortiguada y lo más rápida posible; los valores escogidos fueron $k_a = 29$ y $k = 0.0064$. Por lo tanto el modelo del sistema en lazo cerrado es el siguiente

$$\begin{aligned}\begin{bmatrix} \dot{x} \\ \dot{x}_a \end{bmatrix} &= \begin{bmatrix} -25.7526 & 70508.62 \\ -0.0022068 & 0 \end{bmatrix} \begin{bmatrix} x \\ x_a \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r, \\ y &= [0.0022068 \ 0] \begin{bmatrix} x \\ x_a \end{bmatrix}.\end{aligned}\tag{5.32}$$

Puede comprobarse que los valores característicos de la matriz A del modelo anterior son -16.0703 y -9.6823 , los cuales tienen parte imaginaria igual a cero y parte real negativa. Aunque se pueden modificar los valores de k_a y k para que los valores característicos de la matriz A estén más alejados del eje imaginario, el desempeño del sistema real no mejora (y en algunos casos empeora) debido a que la señal de control solo puede tomar valores dentro del intervalo $[-0.23, 0.23]$.

Las respuestas del sistema en lazo cerrado con un control por asignación de polos cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de escalones y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en las figuras 5.32, 5.33 y 5.34 respectivamente.

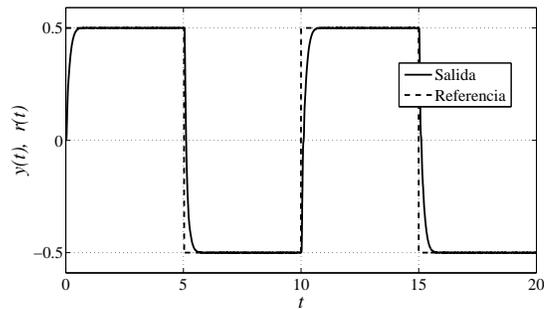


Figura 5.32: Respuesta del sistema en lazo cerrado con un control por asignación de polos a una señal de referencia cuadrada.

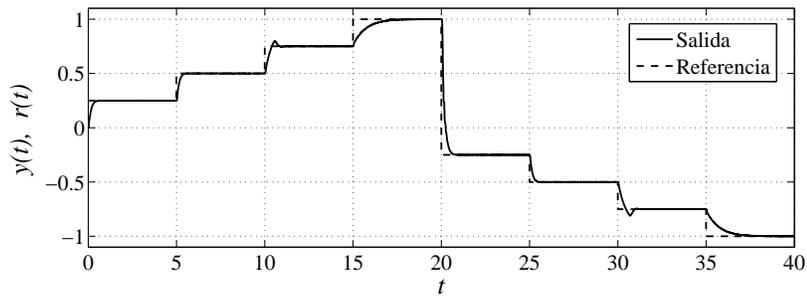


Figura 5.33: Respuesta del sistema en lazo cerrado con un control por asignación de polos cuando la referencia es una suma de escalones.

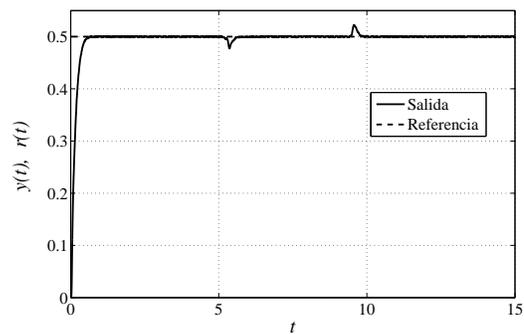


Figura 5.34: Respuesta del sistema con un control con un control por asignación de polos cuando hay una perturbación.

5.3.6 Control de Equivalencia Cierta Usando la Red Neurodifusa

En esta sección se muestra el desempeño del sistema en lazo cerrado usando controles de equivalencia cierta como los descritos en la sección 4.5.

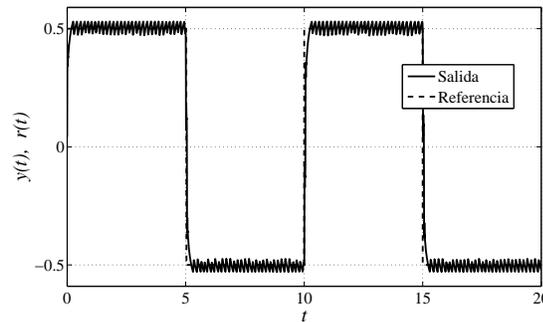


Figura 5.35: Respuesta del sistema con un control de equivalencia cierta, cuando $\nu = y(k+1)d$.

En la figura 5.35 se muestra la respuesta del sistema usando un control de equivalencia cierta (ver ecuación (5.35)) basado en la red neurodifusa, cuyos parámetros se muestran en las ecuaciones 5.33 y 5.34 y donde $\nu = y(k+1)d$. La señal de referencia es una señal cuadrada con amplitud de 0.5. El sistema alcanza a la referencia pero no puede mantenerse en ella. Este tipo de comportamiento es debido a que, en principio la señal de control en el tiempo k debería hacer que el sistema llegara a la referencia en el tiempo $k + 1$, por supuesto el tamaño de las señales de control requeridas para hacer tal seguimiento de la señal de referencia es muy grande y el sistema no las puede generar. Entonces errores de seguimiento muy pequeños provocan señales de control muy grandes que a su vez provocan que el sistema se pase de la referencia y así sucesivamente. Este comportamiento se evitó usando una ν diferente.

En la sección 5.2.3 se dijo que los parámetros h_{1i} deben ser del mismo signo para evitar que $\varphi_k^T h_{c1}$ se haga eventualmente cero e indetermina la señal de control. En

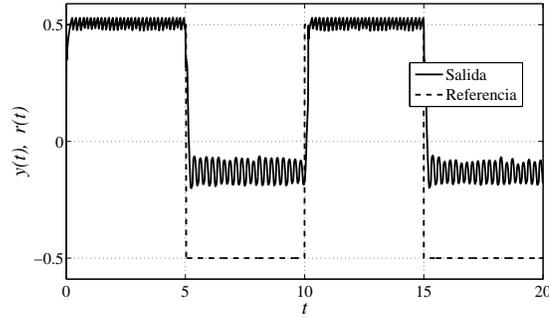


Figura 5.36: Respuesta del sistema con un control de equivalencia cierta, cuando $\nu = y_{(k+1)d}$ y los parámetros h_{1i} tienen signos distintos

la figura 5.36 se muestra lo que ocurre cuando se aplica un control de equivalencia cierta usando como modelo una red neurodifusa donde los parámetros h_{1i} son de signos distintos. La señal de referencia usada fue una señal cuadrada con amplitud de 0.5. Los parámetro de dicha red son los siguientes

$$C = \begin{bmatrix} 2.8989 & -2.8271 & 0.9278 \\ 2.7627 & -2.5327 & 0.7699 \\ 2.2601 & -1.5487 & 0.2884 \end{bmatrix}, \quad H = \begin{bmatrix} -0.0197 & 0.0209 \\ 0.0075 & -0.0073 \\ 0.033 & -0.0324 \end{bmatrix}, \quad (5.33)$$

$$\sigma = \begin{bmatrix} 4.6138 \\ 25 \\ 5.3902 \end{bmatrix}, \quad \varsigma = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}. \quad (5.34)$$

Cuando se usa una red que con los parámetros anteriores para aproximar al sistema electromecánico y la entrada de excitación es la que se muestra en la figura 5.8, el error de aproximación tiene un valor rms de 0.0547.

En la figura 5.36 podemos observar que el sistema nunca alcanza la referencia

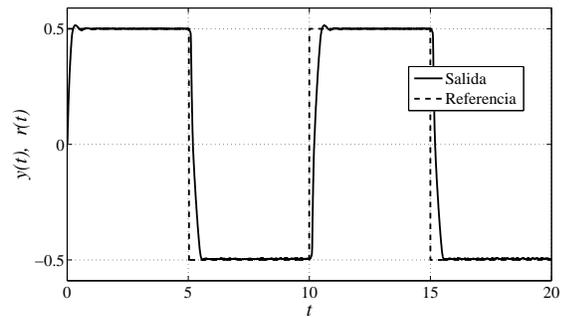


Figura 5.37: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta a una señal de referencia cuadrada.

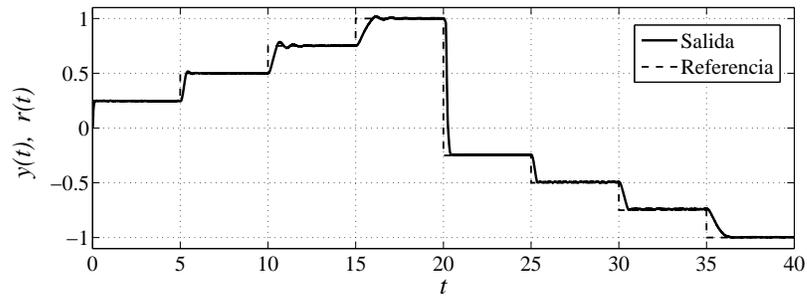


Figura 5.38: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta cuando la referencia es una suma de escalones.

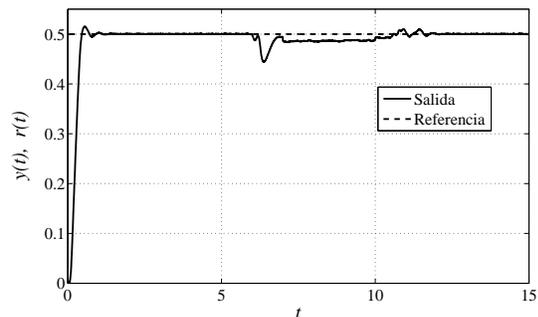


Figura 5.39: Respuesta del sistema con un control de equivalencia cierta cuando hay una perturbación.

de -0.5 y al intentarlo tiene un comportamiento errático alrededor de -0.15 . Analizando los parámetros podemos concluir que es en la zona alrededor de -0.15 en donde podría ocurrir que $\varphi_k^T h_{c1} = 0$. Evidentemente esto nunca ocurrió, ya que de haber ocurrido se hubiera generado un error en el programa que calculaba la señal de control. Lo que ocurre es que en algún momento $\varphi_k^T h_{c1}$ es positivo con un valor absoluto pequeño, provocando un señal de control positiva con un valor absoluto grande ; y en otro momento muy cercano, $\varphi_k^T h_{c1}$ es negativo con un valor absoluto muy pequeño, provocando que la señal de control negativa con un valor absoluto grande.

Para mejorar el desempeño del sistema se volvió a usar la red neurodifusa, cuyos parámetros se muestran en las ecuaciones 5.33 y 5.34 pero esta vez se definió

$$\nu = y_{(k+1)d} + 0.99e_k - 0.08e_{k-1}, \quad (5.35)$$

las raíces del polinomio $z^2 - 0.99z + 0.08$ son 0.9012 y 0.0888, por lo tanto el error de seguimiento (e) debe tender un valor pequeño. Si el error de identificación fuera cero, el error de seguimiento tendería a cero.

Las respuestas del sistema en lazo cerrado usando un control de equivalencia cierta basado en la red neurodifusa, cuyos parámetros se muestran en las ecuaciones 5.33 y 5.34, y en el cual ν se define por medio de la ecuación (5.35) cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de escalones y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en las figuras 5.40, 5.41 y 5.42 respectivamente.

Control de Equivalencia Cierta Usando la Red Neurodifusa con Restricciones

En la sección 5.2.3 se obtuvieron los parámetros de dos redes que modelan al sistema electromecánico. Ahora se aplica un control de equivalencia cierta que usa como

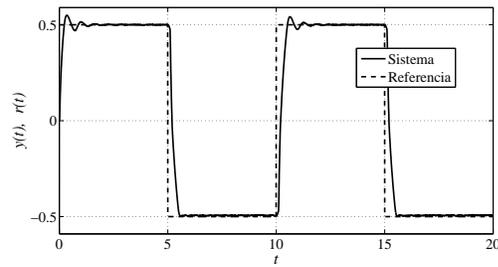


Figura 5.40: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta basado en la res neuronal con restricciones a una señal de referencia cuadrada.

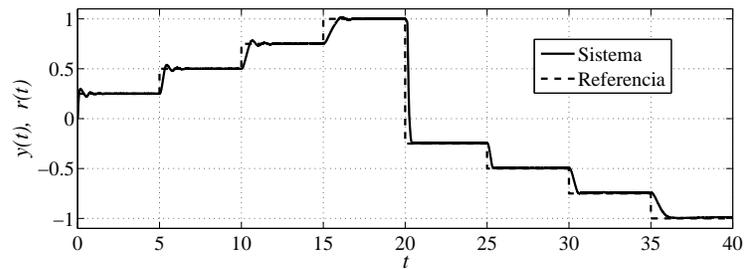


Figura 5.41: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta basado en la res neurodifusa con restricciones cuando la referencia es una suma de escalones.

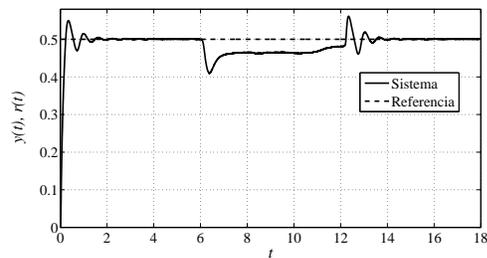


Figura 5.42: Respuesta del sistema con un control de equivalencia cierta basado en la res neurodifusa con restricciones cuando hay una perturbación.

modelo la red neurodifusa con restricciones, cuyos parámetros se muestran en las ecuaciones 5.17 y 5.18.

Las respuestas del sistema en lazo cerrado usando un control de equivalencia cierta basado en la red neurodifusa con restricciones y en el cual ν se define por medio de la ecuación (5.35) cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de escalones y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en las figuras 5.40, 5.41 y 5.42 respectivamente.

5.3.7 Control de Equivalencia Cierta Usando la Red Neurodifusa Estimando Parámetros en Línea

En los controles implementados en la sección anterior no logran que el sistema en lazo cerrado realice un seguimiento adecuado de la señal de referencia cuando existe la perturbación producida por el freno magnético. Para mejorar el desempeño del sistema cuando existe una perturbación fueron implementados controles de equivalencia cierta, similares a los de la sección anterior, pero con la diferencia de que los parámetros se *estiman en línea* usando un método basado en el gradiente.

El método que es usado para ajustar los parámetros es prácticamente igual al de la sección 4.4, solo que en este caso se ajustan sólo los parámetros contenidos en las matrices C y H . Esta decisión fue tomada para evitar que los parámetros σ_i y ς_i tomaran valores indeseados y por la complejidad que representa el cálculo de las parciales de la función de costo con respecto de los parámetros σ_i y ς_i . R y η fueron definidos como I y 0.000001 respectivamente. Y ν se define por medio de la ecuación (5.35).

Se eligió usar los parámetros de las ecuaciones 5.33 y 5.34 ya que el sistema en lazo cerrado usando un control de equivalencia cierta basado en estos parámetros se

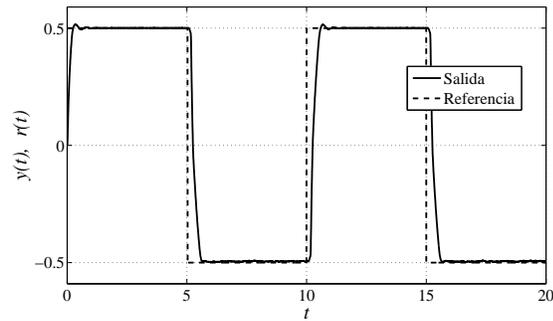


Figura 5.43: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta estimando parámetros en línea una señal de referencia cuadrada.

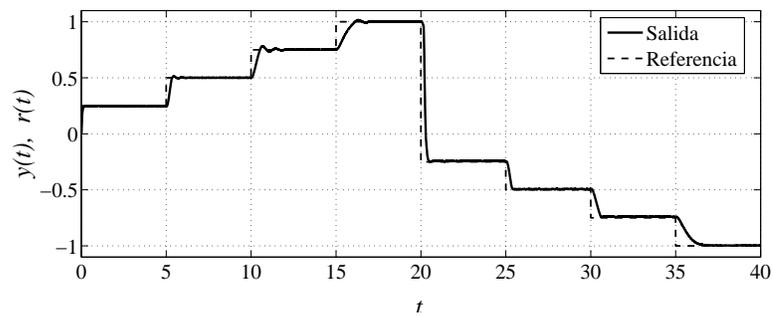


Figura 5.44: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta estimando parámetros en línea cuando la referencia es una suma de escalones.

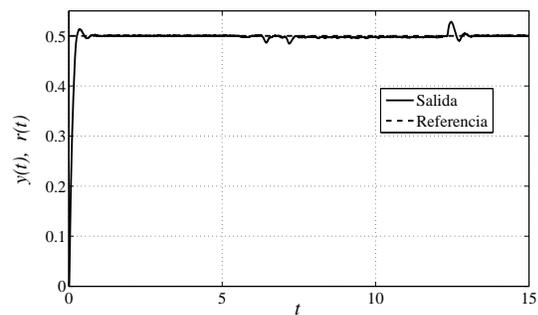


Figura 5.45: Respuesta del sistema con un control de equivalencia cierta estimando parámetros en línea cuando hay una perturbación.

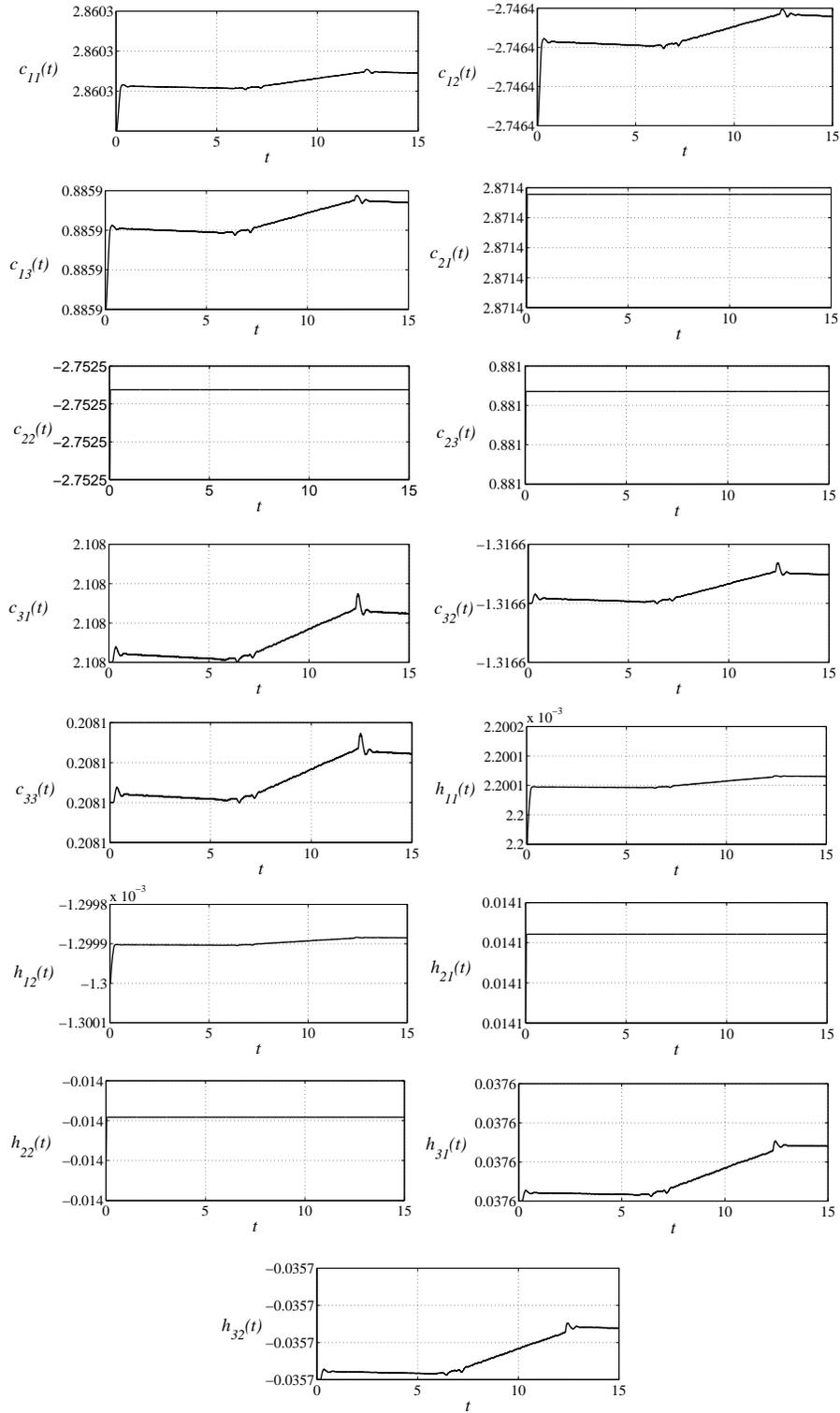


Figura 5.46: Evolución de los parámetros c_{ij} y h_{ij} .

desempeño mejor que cuando se usaron los parámetros de las ecuaciones 5.17 y 5.18. Además la red que tiene los parámetros de las ecuaciones 5.33 y 5.34 tiene un valor RMS del error de aproximación menor que la red con los parámetros 5.17 y 5.18.

Las respuestas del sistema en lazo cerrado usando el control de equivalencia cierta basado inicialmente en la red, cuyos parámetros se muestran en las ecuaciones 5.33 y 5.34, cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de escalones y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en las figuras 5.47, 5.48 y 5.49 respectivamente.

La evolución de los parámetros c_{ij} y h_{ij} , cuando la señal de referencia es un $0.15u_{-1}(t)$ y se aplica el freno magnético, se muestra en la figuras y 5.46. Es importante notar que los parámetros varían poco y lo hacen sobre todo al inicio y cuando es aplicado el freno.

Control de Equivalencia Cierta Usando la Red Neurodifusa con Restricciones Estimando Parámetros en Línea

Podría suceder que durante la operación del sistema, el método para estimar parámetros en línea provoca que los parámetros h_{i1} tengan signos distintos. Esta situación debe ser evitada para evitar que la señal de control se indetermina. Para evitar tal situación se imponen restricciones como las descritas en la sección 4.5.

Los valores iniciales de los parámetros h_{1i} fueron 0.0022, 0.0141 y 0.0376, respectivamente. Aplicando las restricciones, los parámetros h_{1i} quedan definidos de la siguiente manera:

$$h_{11} = 0.0022 \tanh(\check{h}_{11} - 0.0022) + 0.0022, \quad (5.36)$$

$$h_{21} = 0.0141 \tanh(\check{h}_{21} - 0.0141) + 0.0141, \quad (5.37)$$

$$h_{31} = 0.0376 \tanh(\check{h}_{31} - 0.0376) + 0.0376. \quad (5.38)$$

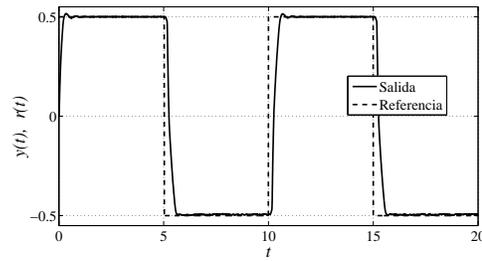


Figura 5.47: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta estimando parámetros, con restricciones a una señal de referencia cuadrada.

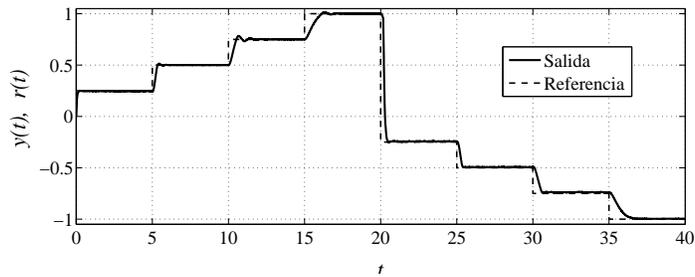


Figura 5.48: Respuesta del sistema en lazo cerrado con un control de equivalencia cierta estimando parámetros, con restricciones cuando la referencia es una suma de escalones.

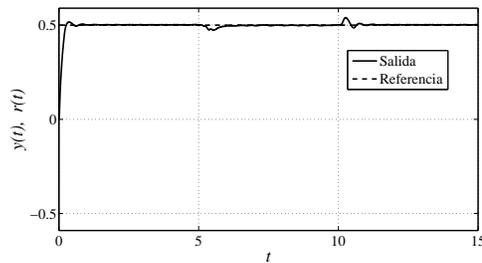


Figura 5.49: Respuesta del sistema con un control de equivalencia cierta estimando parámetros, con restricciones cuando hay una perturbación.

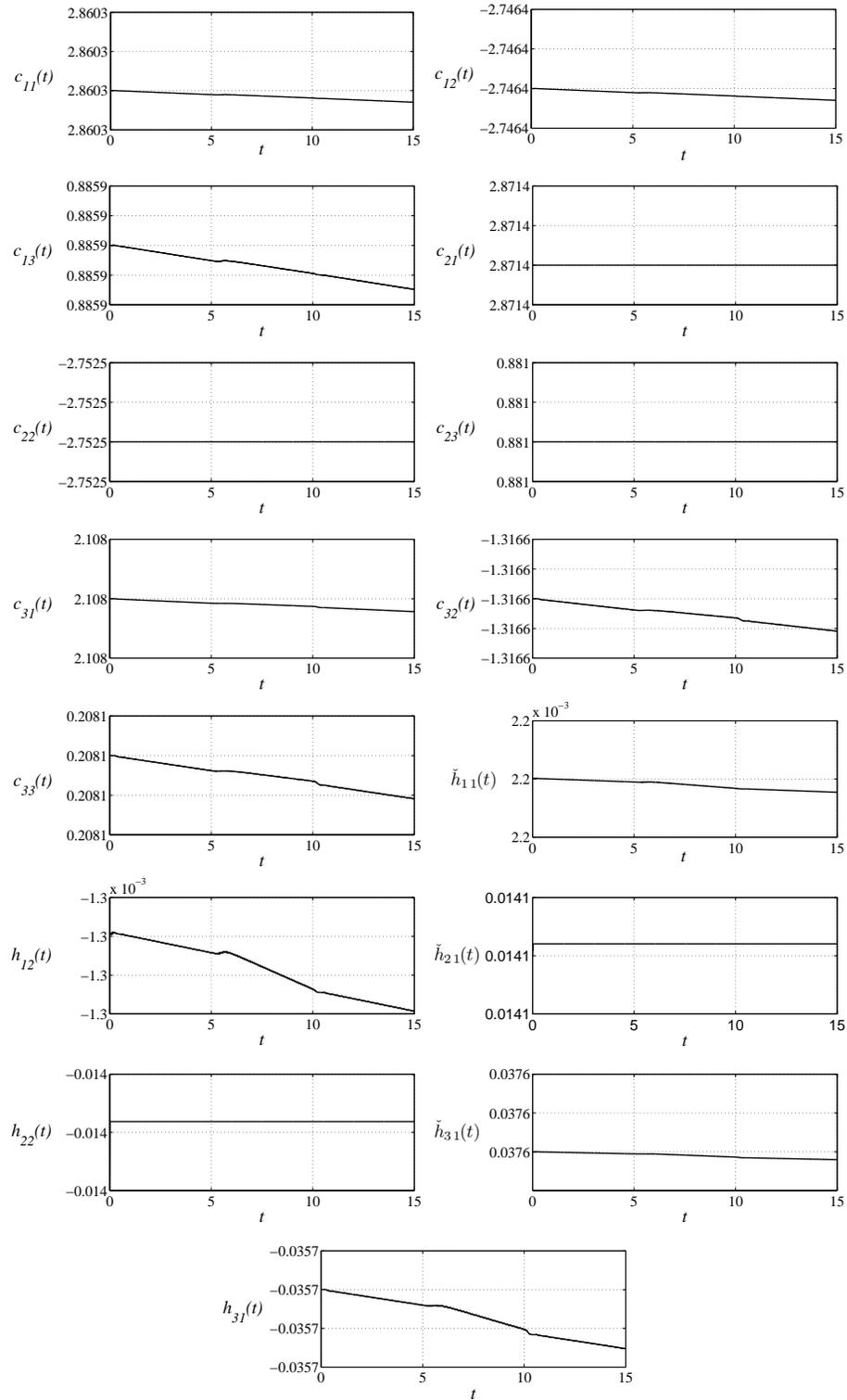


Figura 5.50: Evolución de los parámetros c_{ij} y h_{ij} , con restricciones a h_{i1}

Por lo tanto, se garantiza que:

$$h_{11} \in (0, 0.0044), \quad h_{21} \in (0, 0.0282), \quad h_{31} \in (0, 0.0752).$$

Las respuestas del sistema en lazo cerrado usando el control de equivalencia cierta basado en la red con restricciones, cuyos parámetros iniciales se muestran en las ecuaciones 5.33 y 5.34, cuando las señales de referencia son una señal cuadrada con amplitud de 0.5, una suma de escalones y $0.15u_{-1}(t)$ (aplicando el freno magnético); se muestran en la figura 5.50.

La evolución de los parámetros de la red con restricciones, cuando la señal de referencia es un $0.15u_{-1}(t)$ y se aplica el freno magnético, se muestra en la figura 5.50.

5.3.8 Comparación entre Esquemas de Control

En la figura 5.51 se muestran las respuestas del sistema a una entrada de referencia escalón cuando y además una perturbación es producida aplicando el freno magnético. Estas gráficas han sido escogidas para realizar la comparación del desempeño del sistema con los distintos esquemas de control ya que en ellas se puede observar la respuesta a una entrada escalón y la respuesta ante una perturbación.

Un detalle que no se puede apreciar en la figura 5.51 es que el sistema tiene distintas respuestas transitorias dependiendo no solo del controlador que tenga, sino también del valor de la señal de referencia. Este hecho es en parte imputable a alguna no linealidad del sistema electromecánico y a que la señal de control tiene que ser acotada.

Cuando se aplican los controladores difusos tipo Mamdani y tipo Takagi-Sugeno, el tiempo que le lleva al sistema alcanzar a la referencia es más pequeño que cuando se usan los otros controladores. Esto es debido, en el caso del control difuso tipo Takagi-

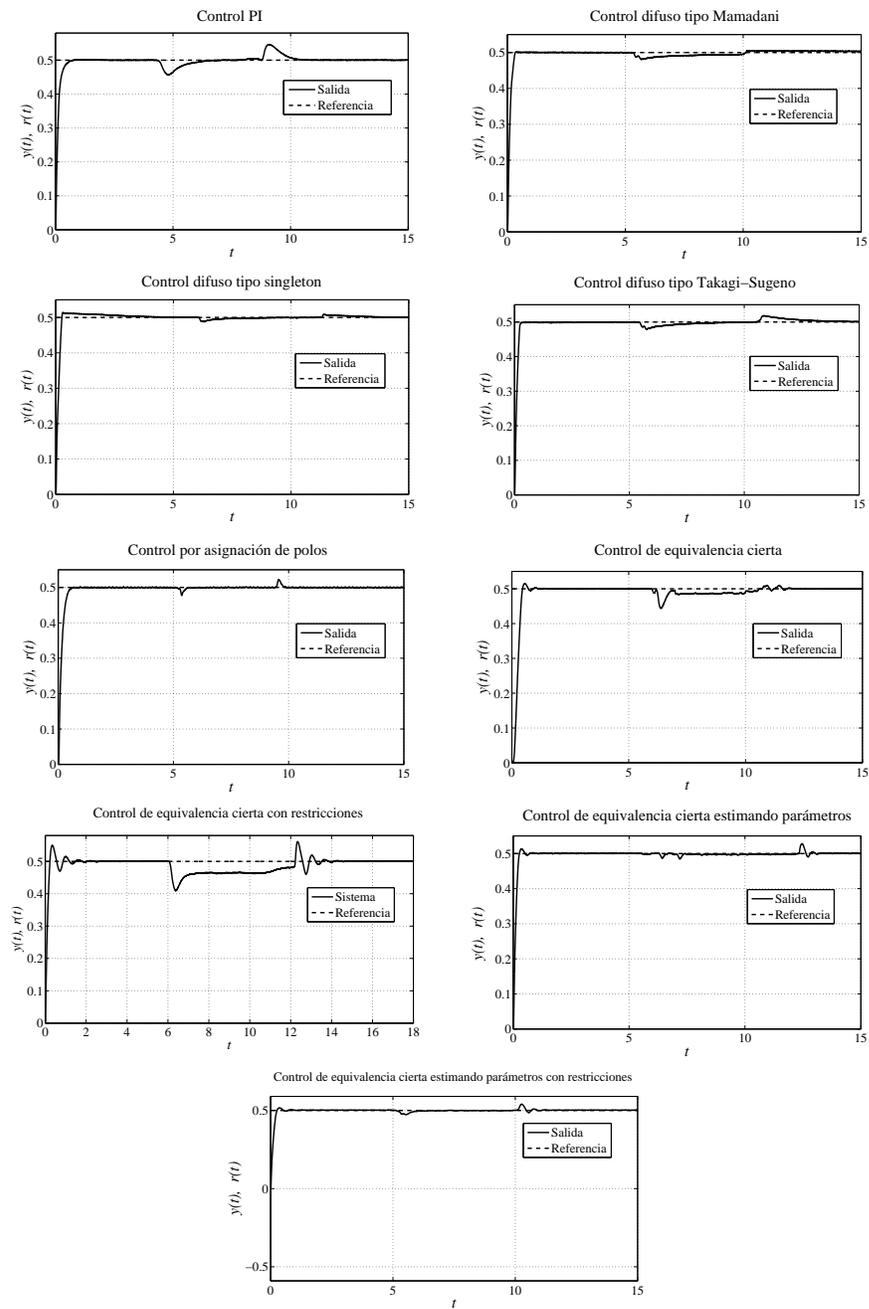


Figura 5.51: Respuesta del sistema con los distintos controles.

Sugeno, a que cuando el error de seguimiento es grande, actúa sobre el sistema un control PI con la ganancia proporcional grande. Cuando se usa el control difuso tipo Mamdani los conjuntos difusos fueron definidos de tal manera que la señal del control fuera grande hasta que el error de seguimiento fuera suficientemente pequeño. Si se usa el controlador difuso tipo singleton, la referencia se alcanza rápidamente pero existe un pequeño sobrepaso y el sistema tarda un poco en volver a alcanzar a la referencia.

Cuando se aplican los controladores de equivalencia se alcanza la señal de referencia bastante rápido pero existe un pequeño sobrepaso cuando la referencia es positiva. La dinámica que presenta el error de seguimiento esta determinada por ν , por lo tanto, si ν es escogida de manera distinta es posible modificar el comportamiento del sistema. Pero también es importante tener en cuenta las limitaciones propias del sistema, por ejemplo, que la señal de control debe ser acotada.

El control por asignación de polos logra que el sistema tenga una mejor respuesta, que cuando se aplican los otros controles, cuando se aplica el freno magnético. La capacidad del control por asignación de polos de eliminar los errores de seguimiento debidos a las perturbaciones provocadas por el freno, se debe a la acción del bloque integrador que incorpora en su diseño (ver figura 2.5).

Como es de esperarse, cuando se aplican los controles de equivalencia cierta el sistema no responde adecuadamente ante la perturbación. Esto se debe a que la ley de control depende muy estrechamente del modelo del sistema, y al aplicar el freno estamos modificando los parámetros. Este problema se aminora cuando los parámetros del modelo se adaptan en línea, a pesar de que los parámetros varían poco esto es suficiente para que el sistema se acerque más a la referencia cuando se aplica el freno. En este caso el error de seguimiento no se elimina como cuando se aplica el control por

asignación de polos, pero es menor que cuando los parámetros permanecen estáticos.

El hecho de imponer restricciones a los parámetros cuando estos son estimados en línea, no parece afectar el desempeño del sistema. Sin embargo no está de más incorporar dichas restricciones ya que esto da la certidumbre de que la señal de control nunca estará indefinida.

Capítulo 6

Conclusiones

En el primer capítulo se habló de la necesidad de contar con modelos de los sistemas con los que se trabaja, esta necesidad se hizo presente a lo largo del desarrollo de esta tesis. La sintonización de los controles PID y difusos se realizó con ayuda de simulaciones, y por lo tanto fue importante contar con un modelo que identificara adecuadamente al sistema. Cuando el modelo usado para dichas simulaciones es la red neurodifusa, los resultados arrojados por las simulaciones son más cercanos a los resultados obtenidos de forma experimental, que cuando se usan los modelos lineales. Por lo tanto la red neurodifusa nos proporcionó una mejor herramienta para diseñar esquemas de control.

El pequeño valor RMS del error de aproximación de la gráfica 5.11 (error de aproximación cuando se usa la red neurodifusa como modelo) indica que la capacidad de la red neurodifusa para aproximar al sistema es bastante satisfactoria. Este hecho aunado al desempeño satisfactorio de la red neurodifusa en las simulaciones, mencionadas en el párrafo anterior, nos da la seguridad de que este modelo puede ser usado para otras tareas donde se requiera de un modelo del sistema electromecánico.

Una vez que se ha demostrado experimentalmente que la red neurodifusa identifica satisfactoriamente a un sistema relativamente sencillo como lo es el sistema electromecánico, es válido pensar que la estructura de la red puede ser usada para identificar sistemas más complejos.

De igual manera puede abordarse el problema de controlar sistemas más complicados usando controles de equivalencia cierta basados en la red que los identifique. El uso del control de equivalencia cierta es conveniente sólo cuando se tiene una amplia confianza en la validez del modelo. El control de equivalencia cierta no es el único esquema de control basado en la red que puede ser implementado. Por ejemplo, se puede usar algún control predictivo basado en el modelo.

La red que se entrenó con restricciones tiene un error de aproximación con un valor RMS más grande que la que se entrenó manteniendo estáticos los parámetros ς , y los controles de equivalencia cierta estimando parámetros en línea con y sin restricciones funcionan de manera igualmente satisfactoria. A pesar de las dos razones anteriores podemos concluir que vale la pena implementar las restricciones, ya que estas garantizan que los parámetros sobre los cuales se aplican nunca tomaran valores indeseados.

Implementar las restricciones a los parámetros ς nos permite conservar el conocimiento previo que se tiene acerca del funcionamiento del sistema. Cuando las restricciones se aplican a los parámetros ς , se garantiza que los conjuntos difusos nunca estarán fuera de la región de operación; y cuando las restricciones se aplican a los parámetros h_{i1} , se garantiza que el control de equivalencia cierta siempre se podrá aplicar. Por otra parte, al implementar las restricciones a los parámetros ς , en vez de dejarlos estáticos, se aumenta la flexibilidad de la red.

Cuando existe una perturbación y se aplica un control de equivalencia cierta re-

alizando la estimación de parámetros en línea el sistema en lazo cerrado tiene una mejor respuesta que cuando los parámetros permanecen estáticos. La estimación en línea de los parámetros de la red puede ser de gran utilidad cuando se trabaja con sistemas variantes en el tiempo.

Tal como se menciona en el primer capítulo existe un compromiso entre la complejidad del modelo y su capacidad de aproximar al sistema. Como ya se ha mostrado, la red realiza una mejor identificación del sistema que los modelos lineales, pero el precio que hay que pagar por ello es la mayor complejidad del modelo y la mayor cantidad de parámetros que contiene el mismo. El modelo que ha de ser usado dependerá de las necesidades específicas de cada problema. Por ejemplo, el control por asignación de polos hace uso del modelo lineal de primer orden en su diseño, y provoca que el desempeño del sistema en lazo cerrado sea satisfactorio.

El control PI hace que la respuesta en lazo cerrado del sistema sea satisfactoria, en el sentido que la referencia se alcanza rápidamente y que hace que el sistema sea robusto ante las perturbaciones provocadas por el freno magnético. Los demás controles hacen que el sistema alcance más rápidamente que cuando se usa un control PI. Una vez más, el precio que hay que pagar por tener un mejor desempeño es el aumento de la complejidad del esquema de control.

El control PI fue el más sencillo de implementar y es más sencillo de sintonizar que los controles difusos. Podríamos decir que para controlar este sistema electromecánico basta un control PI. Pero para sistemas más complicados, un control PI puede ser insuficiente o inadecuado. Es en este tipo de casos cuando se justifica la implementación de un control más complicado, como un control difuso o de equivalencia cierta.

Los recursos computacionales con los que se contó fueron suficientes para la im-

plementación de los esquemas de control, pero si se hubiera ocupado un microcontrolador menos poderoso, seguramente el tiempo de muestreo de 5 ms no hubiera sido suficiente para realizar cálculos como la desfusión en el control difuso tipo Mamdani, o los productos de Kronecker en los controles de equivalencia cierta. Si nos enfrentáramos al problema de controlar al sistema electromecánico usando un microcontrolador menos poderoso tendríamos que sacrificar un poco del desempeño en aras de tener una menor complejidad. Por ejemplo, tendríamos la necesidad de migrar de un control difuso tipo Mamdani a un control difuso tipo singleton, para simplificar el cálculo de la desfusión.

Finalmente, podemos concluir que la red neurodifusa recurrentes descrita en el capítulo 4 es un recurso de gran utilidad para realizar tareas de identificación de sistemas no lineales en los que solo se dispone de los datos entrada-salida. La red permite incorporar el conocimiento previo del sistema, además de que nos permite tener una interpretación física de la misma como un conjunto de modelos lineales que juntos describen a un sistema no lineal.

Apéndice A

Programas

Los programas para la identificación del sistema electromecánico fueron desarrollados en Matlab y consistieron en la programación de las ecuaciones correspondientes, para cada método de identificación.

La implementación de los distintos esquemas de control se realizó con ayuda de programas desarrollados en Visual Basic. La figura A.1 muestra el diagrama de flujo de los programas que se realizaron para realizar la implementación de los esquemas de control. En estos programas, la señal de referencia es leída desde un archivo de texto, cada tiempo de muestreo se realiza la lectura de la velocidad, el cálculo y la escritura de la señal de control, y finalmente los datos son guardados en un archivo de texto de tal manera que queden disponibles para su posterior análisis.

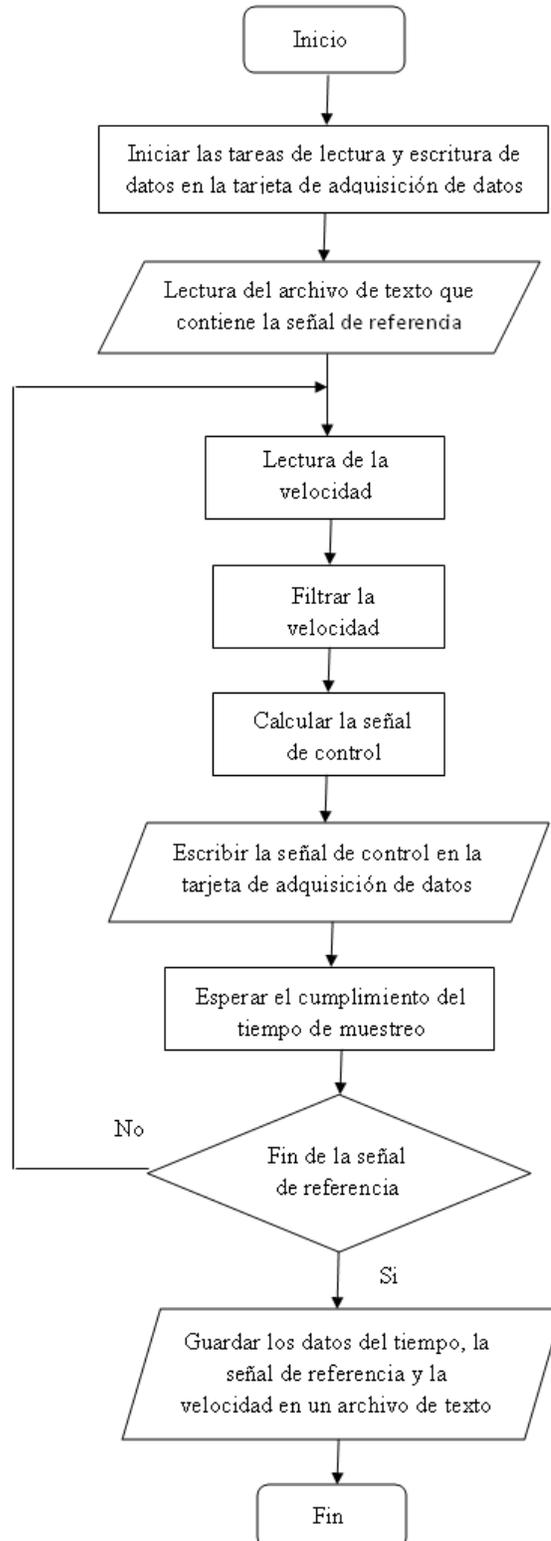


Figura A.1: Diagrama de flujo

Bibliografía

- [1] Hung T. Nguyen, Nadipuram R. Prasad, Carol L. Walker and Elbert A. Walker. *A first course in fuzzy and neuralcontrol*. Chapman & Hall/CRC, 2003.
- [2] Gonzalez Olvera M. and Tang Xu. *A new recurrent neurofuzzy network for identification of dynamic systems*. Fuzzy sets and systems. Vol 158, N° 10. Pags 10023 - 1035. Elsewe, 2007.
- [3] Gonzalez Olvera M. and Tang Xu. *Identification and control of non linear systems via a constrained input - output neurofuzzy network*, Book: *Advances in control systems theory and aplications*. University of science and technology of China. Gang Tao, to be published 2009.
- [4] Detlef D Nauck and Andreas Nurnberger. *The Evolution of Neuro-Fuzzy Systems*. NAFIPS, 2005.
- [5] Ryszard S. Gessing. *Whether the Spreaded Good Opinion About Fuzzy Controllers is Justified*. Proceedings of the 17th World Congress The International Federation of Automatic Control. Seoul, 2008.
- [6] Hassan Khalil. *Nonlinear systems*. Prentice Hall, 3rd edition 2002.
- [7] Oliver Nelles. *Nonlinear system identification*. Springer, Berlin, 2001.

- [8] P.J. Werbos. *Beyond regression: New tools for prediction and analysis in the behavioral sciences*. Harvard University, 1974.
- [9] L.A. Zadeh. *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers*. World Scientific Pub Co Inc, 1996.
- [10] K.S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Tras. Neural Networks*, 2(4): 303-314. 1:4-27, March 1990.
- [11] Kevin M. Passino. *Fuzzy control*. Addison Wesley Longman, Menlo, 1998.
- [12] Bart Kosko. *Fuzzy engineering*. Prentice Hall, Upper Saddle River, 1997.
- [13] Gloria Mata, Víctor M. Sanchez y Juan M. Gómez. *Análisis de sistemas y señales*. Facultad de Ingeniería, UNAM, México D.F., 2002.
- [14] Li-Xin Wang. *Adaptative fuzzy sistemes and control*. Prentice Hall, Englewood Cliffs, 1994.
- [15] Rolf Johanssons. *System modeling and identification*. Prentice Hall, Englewood Cliffs, 1993.
- [16] Katsuhiko Ogata. *Ingeniería de control moderna*. Pearson Prentice Hall, Madrid, 4ª edición, 2003.
- [17] Norman S. Nise. *Sistemas de control para ingeniería*. Cecs, México, 3ª edición, 2006.
- [18] Chi-Tsong Chen. *Linear sistem theory and desing*. Oxford University Press, New York, 1999.
- [19] Jean Jacques E. Slotine and Weiping Li. *Applied nonlinear control*. Prentice Hall, Englewood Cliffs, 1991.

- [20] Simon Haykin. *Neural networks*. Pearson Prentice Hall, New Dehli, 2^a edition, 2005.
- [21] A. Cichocki and R Unbehauen. *Neural networks for optimization and signal processing*. John Wuiley & Sons, Chichester, 1993.
- [22] Roland S. Burns. *Adevanced controlengineering*. Butterworth-Heinemann, New Dehli, 2001.
- [23] Jan Mikles and Miroslav Fikar. *Process modelling, identification and control*. Springer, 2007.
- [24] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, 2(4): 303-314. Springer, 1989.