

UNIVERSIDAD AUTONOMA DE NUEVO LEON
FACULTAD DE INGENIERIA MECANICA Y ELECTRICA



**MODELADO DE SISTEMAS
MEDIANTE REDES NEURONALES**

T E S I S

**QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERIA ELECTRICA
ESPECIALIDAD EN CONTROL**

PRESENTA:

EFRAIN ALCORTA GARCIA

MONTERREY, N. L.

FEBRERO DE 1992

TM

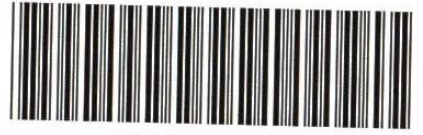
Z5853

.M2

FIME

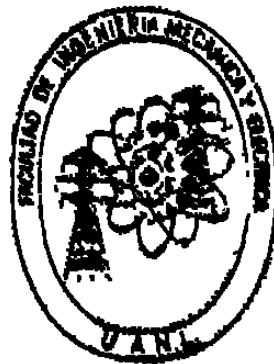
1992

A4



1020070622

UNIVERSIDAD AUTONOMA DE NUEVO LEON
FACULTAD DE INGENIERIA MECANICA Y ELECTRICA
DIVISION DE ESTUDIOS SUPERIORES



**MODELADO DE SISTEMAS
MEDIANTE REDES NEURONALES**

T E S I S

**QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERIA ELECTRICA
ESPECIALIDAD EN CONTROL**

PRESENTA

EFRAIN ALCORTA GARCIA

MONTERREY, N. L.

FEBRERO DE 1992

UNIVERSIDAD AUTONOMA DE NUEVO LEON

FACULTAD DE INGENIERIA MECANICA Y ELECTRICA

DIVISION DE ESTUDIOS SUPERIORES

MODELADO DE SISTEMAS MEDIANTE REDES NEURONALES

TESIS

QUE PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS EN INGENIERIA ELECTRICA

ESPECIALIDAD EN CONTROL

PRESENTA:

EFRAIN ALCORTA GARCIA

MONTERREY, N. L.

FEBRERO DE 1992

A MIS PADRES

Mario René Alcorta Montemayor

Aracelia García de Alcorta

Con agradecimiento y respeto

A MIS HERMANOS

Maria Aracelia

Mario René

Luciano

Antonio Aradio

Con cariño

AGRADECIMIENTOS

Al Dr. Edgar N. Sánchez C. por su apoyo y paciencia en el desarrollo de este trabajo, los cuales permitieron llevarlo a buen término.

Al Dr. Joaquin Collado M. por sus sugerencias y comentarios, siempre valiosos en la elaboración de este trabajo.

Al Dr. Rafael Kelly M. con quien iniciara mis estudios de posgrado y a quien debo buena parte de los conocimientos y motivación que ahora tengo.

A Ma. de la Luz de la Rosa por su ayuda en la mecanografía de este trabajo y en general a todos y cada uno de mis profesores y compañeros del programa Doctoral en Ingeniería Eléctrica por su amistad y compañerismo.

Al Consejo Nacional de Ciencia y Tecnología por su apoyo en la realización de este trabajo.

CONTENIDO

INTRODUCCION

Cap. 1 REDES NEURONALES	1
1.1 Reseña histórica	1
1.2 Algunas estructuras de redes neuronales	4
Cap. 2 PRELIMINARES	11
2.1 Preliminares matemáticos	11
2.2 El problema de aproximación	20
Cap. 3 APROXIMACION DE FUNCIONES	22
3.1 Redes neuronales como aproximadores	23
3.2 Construcción de la aproximación	26
3.2.1 Aproximación por mínimos cuadrados	27
3.2.2 La mejor aproximación	34
3.2.3 Filtro de Kalman extendido	39
3.3 Dimensionalidad	43
Cap. 4 MODELADO DE SISTEMAS	49
4.1 Utilización de redes neuronales estáticas	49
4.1.1 Identificación paramétrica	50
4.1.2 Identificación del estado estacionario	52
4.2 Utilización de redes neuronales dinámicas	53
4.2.1 El operador de Hopfield	54

4.2.2 Aproximación de operadores	55
4.2.3 Conjetura acerca de la aproximación de operadores a través del operador de Hopfield	56
4.2.4 Acerca de la veracidad de la conjetura	59
Cap. 5 CONCLUSIONES Y TRABAJOS FUTUROS	62
REFERENCIAS	64

INDICE DE FIGURAS

1.1 Analogía entre una neurona y su modelo matemático.	3
1.2 Red neuronal con una capa de procesamiento y función de activación sigmoideal.	5
1.3 RN1CP de una sola entrada.	6
1.4 Red neuronal multicapa con una capa de entrada, dos capas ocultas y una capa de procesamiento.	7
1.5 Representación compacta de la RNM	8
1.6 Diagrama de bloques de la red neuronal de Hopfield.	9
1.7 Representación de la red neuronal de Hopfield de n estados.	10
2.1 Representación gráfica del teorema de la proyección para R^3	15
3.1 Esquema de aproximación fuera de línea	29
3.2 Esquema de aproximación en presencia de ruido	40
3.3 Representación de un vector en el espacio euclidiano tridimensional	44
3.4a Gráfica de las salidas, ejemplo 1	45
3.4b Gráfica del error, ejemplo 1	45
3.5a Gráfica de las salidas con u_2 , ejemplo 1	45
3.5b Gráfica del error con u_2 , ejemplo 1	46
3.6a Gráfica de las salidas con u_1 , ejemplo 2	46
3.6b Gráfica del error con u_1 , ejemplo 2	46
3.7a Gráfica de las salidas con u_2 , ejemplo 2	47
3.7b Gráfica del error con u_2 , ejemplo 2	47

3.8a Gráfica de las salidas, ejemplo 3	47
3.8b Gráfica del error, ejemplo 3	48
3.9a Gráfica de las salidas, ejemplo 4	48
3.9b Gráfica del error, ejemplo 4	48
4.1 Esquemas que han sido propuestos para la identificación de sistemas no lineales	51
4.2 Representación de la red neuronal de Hopfield modificada.	55
4.3 Gráfica de la salida del sistema y de la red neuronal	61
4.4 Evolución del error en el tiempo	61

INTRODUCCION

El tema de *redes neuronales* recientemente ha sido sujeto de amplia investigación, tal y como se puede constatar en las publicaciones especializadas. Este auge en el estudio de *redes neuronales* ha sido impulsado principalmente por sus características tanto de velocidad de procesamiento, como de flexibilidad. Lo anterior unido a los avances tecnológicos permiten su utilización en diversos campos como reconocimiento de patrones, computación paralela, etc.

Las características mencionadas motivan la idea de tratar de incorporar las *redes neuronales* como herramienta útil a las técnicas del control automático, en específico en lo que se refiere al modelado e identificación de sistemas. Dada la gran importancia que tiene para control el obtener una representación matemática de sistemas desconocidos a partir de mediciones entrada-salida, las características mencionadas de las *redes neuronales* hacen más atractiva su utilización; el presente trabajo se enfoca precisamente en su posible uso en este campo.

La presentación de la tesis está organizada como sigue:

En el capítulo 1 se muestra una breve reseña histórica además se describen las *redes neuronales* en las que se basa este trabajo. En el capítulo 2 se repasan algunos conceptos matemáticos que serán de utilidad más adelante, así como el planteamiento del

problema de aproximación de funciones mediante redes neuronales.

En el capítulo 3 y 4 se discuten las principales aportaciones del presente trabajo: la propuesta de uso del gradiente descendente en continuo para el ajuste de los pesos, el replanteamiento del problema de aproximación mediante redes neuronales en el contexto de espacio de Hilbert, y finalmente en el 4, el planteamiento del problema de aproximación de operadores (análogo al de modelado de sistemas) utilizando redes neuronales dinámicas y la elaboración de una conjetura acerca de su uso, la cual es respaldada por simulaciones. Además se repasan las principales antecedentes con respecto a la aproximación de funciones y modelado de sistemas.

Finalmente, en el capítulo 5 se resumen las conclusiones y se dan algunas sugerencias para trabajos futuros.

Capítulo 1

Redes Neuronales

1.1 RESEÑA HISTORICA

El término red neuronal es usado para describir varios modelos de elementos básicos (neuronas) altamente interconectados, que forman una burda aproximación del sistema nervioso biológico. Estas ofrecen una alternativa muy interesante al cálculo secuencial tradicional.

El concepto de red neuronal no es nuevo, la idea fue concebida originalmente como un modelo del cerebro humano [1], la finalidad era comprender y explicar como funcionaba éste. Ya en el siglo XIX Williams James realiza una descripción del funcionamiento del cerebro; sin embargo, no es sino hasta 1943 cuando McCulloch y Pitts dan formalismo a esta descripción. Como resultado se obtiene un modelo matemático (Neuronal M-P) para describir la actividad nerviosa del cerebro.

En 1958 Rosenblatt publicó como una neurona M-P puede ser usada para clasificar cierto conjunto de patrones. Esto vino a incrementar el interés que ya se tenía sobre redes neuronales; sin embargo, en la década de los sesenta varios hechos sirvieron para apagar este entusiasmo:

- 1) Se llegó a pensar que era posible construir partes del cerebro, cosa que además de ser falsa era muy mal vista.
- 2) Existía una gran confusión y a algunos científicos les irritaba las afirmaciones

CAPITULO 1:REDES NEURONALES

irresponsables hechas acerca de los posibles logros de las redes neuronales.

3) Los modelos no podían ser llevados fácilmente a la práctica debido principalmente a las limitaciones tecnológicas.

4) Minsky y Papert mostraron que un perceptrón¹ de una capa no podía formar la operación lógica OR exclusivo, además conjeturaron que esto era cierto para toda combinación de perceptrones.

Afortunadamente la conjetura de Minsky y Papert resultó ser falsa, como luego se probó en la década de los setentas; así mismo, los avances en electrónica (VLSI) facilitaron la implementación de los modelos. Todo esto provocó un incremento en el interés, tanto en el estudio como en la puesta en práctica de los modelos de redes neuronales.

En 1984 J. J. Hopfield [2] propone un modelo dinámico de red neuronal. No fue el primero ni el único modelo de este tipo; sin embargo, ha alcanzado gran popularidad debido al éxito en aplicaciones como memoria asociativa y en la solución a problemas de programación lineal [1].

En 1988 aparecen una serie de resultados en lo referente a aproximación de funciones mediante redes neuronales [3-7], sin embargo en su mayoría estos resultados no son constructivos. Basados en estos resultados algunos autores intentan construir la aproximación formalmente [7-10].

Los primeros modelos trataron de construirse haciendo analogía a las neuronas

¹ Estructura que modela a una neurona, muy popular en la década de los sesentas.

biológicas (fig. 1.1) [1].

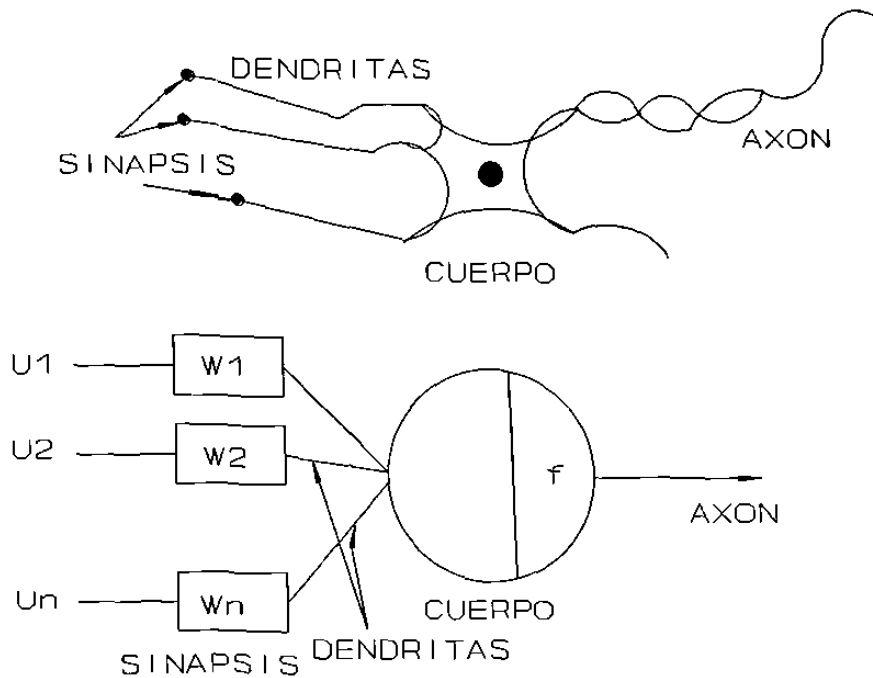


Figura 1.1 Analogía entre una neurona y su modelo matemático.

En la figura 1.1 se observan las diferentes partes de la neurona; la sinapsis es un punto de conexión ponderado; la dendrita conecta una sinapsis al cuerpo; en el cuerpo se suman las entradas ponderadas y se aplican como argumento a una función (llamada función de activación) generalmente no lineal; finalmente el axon conecta el cuerpo de la neurona con alguna otra.

La conexión masiva de neuronas se le denomina red neuronal; algunas de las clasificaciones son basadas en la forma en que se interconectan. Actualmente el estudio

CAPITULO 1:REDES NEURONALES

de las redes neuronales ha seguido dos caminos distintos; el primero a través de neuro fisiólogos; los cuales siguen tratando de modelar y comprender el funcionamiento del cerebro; el segundo seguido por ingenieros y gente del área de la computación, los cuales tratan de incorporar los modelos del cerebro, hechos por los neurofisiólogos, a la ingeniería y a la computación [11]. Este trabajo se clasifica dentro del segundo grupo.

Algunos temas en los que se aplican redes neuronales son:

- i) Clasificación de patrones.
- ii) Memorias asociativa.
- iii) Descomposición de señales.
- iv) Solución a problemas de programación lineal.
- v) Control de robots.
- vi) Aproximación de funciones.
- vii) Identificación de sistemas dinámicos.

1.2 ESTRUCTURAS DE REDES NEURONALES

A continuación se describen las estructuras de redes neuronales, que se utilizarán en este trabajo.

1.2.1 Red Neuronal con una capa de procesamiento (RN1CP)

Esta estructura de red neuronal es de las más básicas, por su simplicidad; generalmente es utilizada para realizar mapeos de \mathbf{R}^n a \mathbf{R} . Mapeos a \mathbf{R}^m pueden

CAPITULO 1:REDES NEURONALES

realizarse utilizando tantas redes neuronales como m elementos tenga la salida.

Una RN1CP de k elementos de procesamiento está compuesta por n entradas. Para cada nodo las entradas son ponderadas (multiplicadas por un peso de ponderación w_{ij}) y sumadas. A esta suma se le agrega un valor de umbral (θ) y el resultado se aplica como argumento a una función denominada función de activación, generalmente no lineal. La salida de la red se obtiene al sumar la salida ponderada por β , de cada elemento de procesamiento (fig.1.2.)

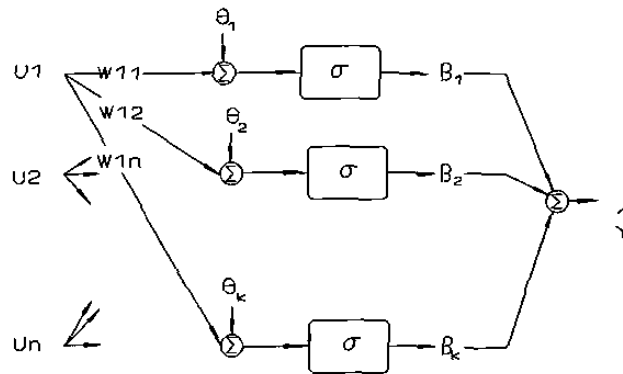


Figura 1.2 Red neuronal con una capa de procesamiento y función de activación sigmoidea.

La ecuación que describe a esta red es:

$$\hat{Y} = \sum_{i=1}^k \beta_i \sigma \left(\sum_{j=1}^n w_{ij} u_j + \theta_i \right)$$

Generalmente en este trabajo se utilizará la red con una sola entrada, es decir los mapeos serán de \mathbf{R} a \mathbf{R} (fig. 1.3).

$$\hat{Y} = \sum_{i=1}^K \beta_i \sigma (w_i u + \theta_i)$$

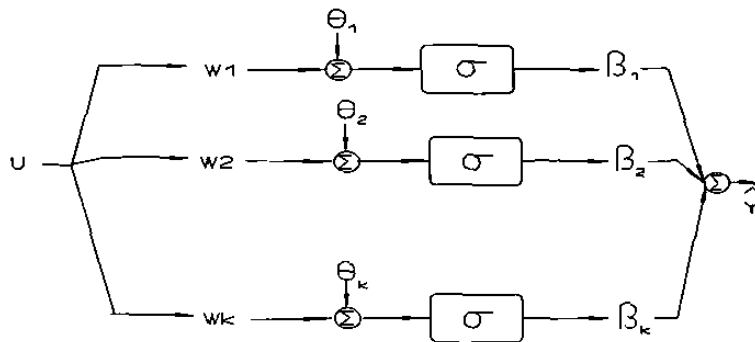


Figura 1.3 RN1CP de una sola entrada.

1.2.2 Red Neuronal Multicapa

La Red Neuronal Multicapa (RNM) es una generalización de la RN1CP. Básicamente puede desarrollar el mismo tipo de mapeos [15]; sin embargo, aquí entran en juego un número mayor de parámetros e interconexiones. Una Red Neuronal Multicapa con una capa de entrada, una de salida y dos ocultas se muestra en la figura 1.4.

Para simplificar la notación se denotan las ponderaciones de cada capa por las matrices $W^1 \in \mathbf{R}^{p \times m}$, $W^2 \in \mathbf{R}^{q \times p}$, $W^3 \in \mathbf{R}^{m \times q}$, Γ^1 , Γ^2 , y Γ^3 son operadores matriciales diagonales, los elementos γ de la diagonal son las funciones de activación, que actúan sobre la suma ponderada de las salidas de la capa anterior. Generalmente la función de activación es no lineal. Típicamente se utiliza la

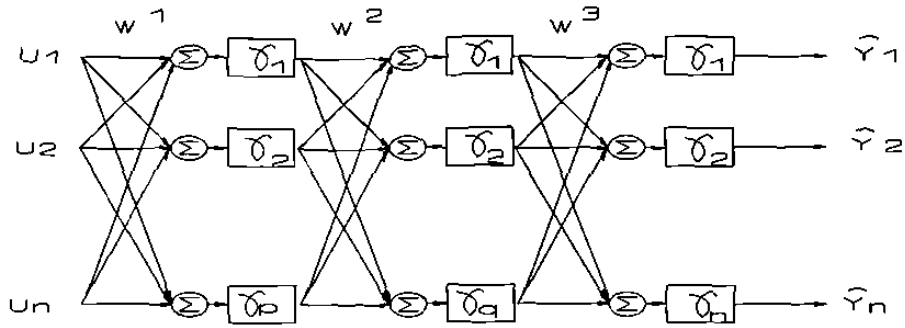


Figura 1.4 Red neuronal multicapa con una capa de entrada, dos capas ocultas y una capa de salida

función de activación sigmoideal $\gamma_i(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$, al igual que en la RN1CP; cada capa

de la red puede ser representada como:

$$N_1[u] = \Gamma^1[W^1u]$$

donde W^1u representa el producto de la matriz de ponderaciones por el vector de entradas; $\Gamma(W^1u)$ describe la aplicación del vector de entradas ponderado como argumento de la función de activación. El mapeo entrada-salida de la red puede representarse como:

$$Y = N[u] = \Gamma^3[W^3\Gamma^2[W^2\Gamma^1[W^1u]]] = N_3N_2N_1[u]$$

$$Y \in R^m \quad u \in R^n$$

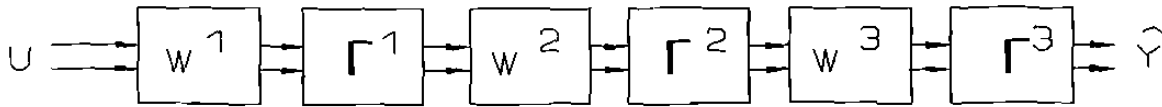


Figura 1.5 Representación compacta de la RNM

1.2.3 Red Neuronal de Hopfield (RNH)

La red neuronal de Hopfield [2] fue propuesta en 1984; tiene la particularidad de que su comportamiento es descrito por una ecuación diferencial y no una algebraica como en las anteriores (RN1CP, RNM). Constituye por lo tanto una red neuronal dinámica.

Este modelo fue originalmente utilizado para asociar patrones influenciados por ruido [2]. La idea fue la de incluir cada patrón a recordar (patrón de entrada) en una condición inicial, y relacionar cada patrón al que se quería asociar (patrón de salida) con un punto de equilibrio asintóticamente estable; así la salida de la red convergerá al punto de equilibrio (patrón de salida) en cuya región de atracción fuera colocada la condición inicial (patrón de entrada).

La ecuación diferencial matricial que describe el comportamiento de la RNH está dada por (ver figura 1.6):

$$\frac{dz}{dt} = -A * Z + W * g(Z) + U \quad Z(0) \in \mathbb{R}^n$$

$$Y = g(Z)$$

$$g(Z) = \begin{bmatrix} \text{tgh}(z_1) \\ \vdots \\ \text{tgh}(z_n) \end{bmatrix}; \quad U = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}; \quad A = \text{diag}\{\alpha_i\} \quad i=1, \dots, n$$

La matriz $A \in \mathbb{R}^{n \times n}$ es diagonal definida positiva; $W \in \mathbb{R}^{n \times n}$ es la matriz de pesos de interconexión; $U \in \mathbb{R}^n$ es la entrada; $Z(0)$ es la condición inicial; $g(Z)$ es una función vectorial con elementos sigmoidales del tipo tangente hiperbólica (figura 1.6).

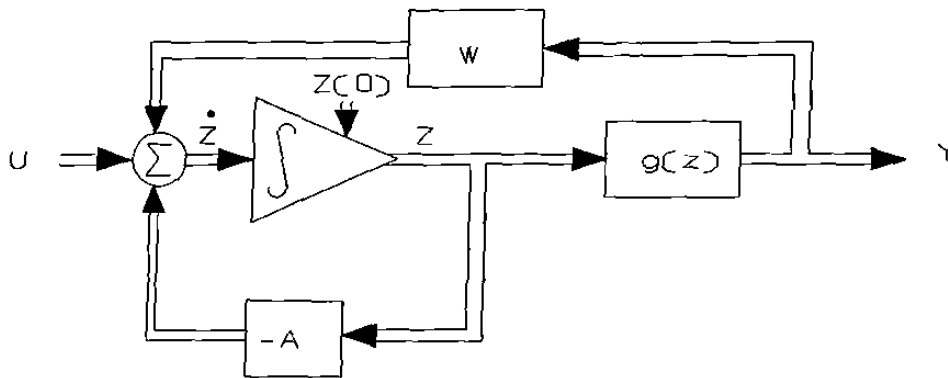


Figura 1.6 Diagrama a bloques de la red neuronal de Hopfield

En el desarrollo original presentado en [2], el vector U es considerado constante y los elementos de las matrices A y W son dadas en terminos de resistencias y capacitancias, la salida Y es definida como la tangente hiperbólica del estado de tal forma que queda de su misma dimensión. Como se puede observar la red neuronal de Hopfield es un sistema de ecuaciones diferenciales no lineales acopladas, una representación en

diagrama de bloques se muestra en la figura 1.7.

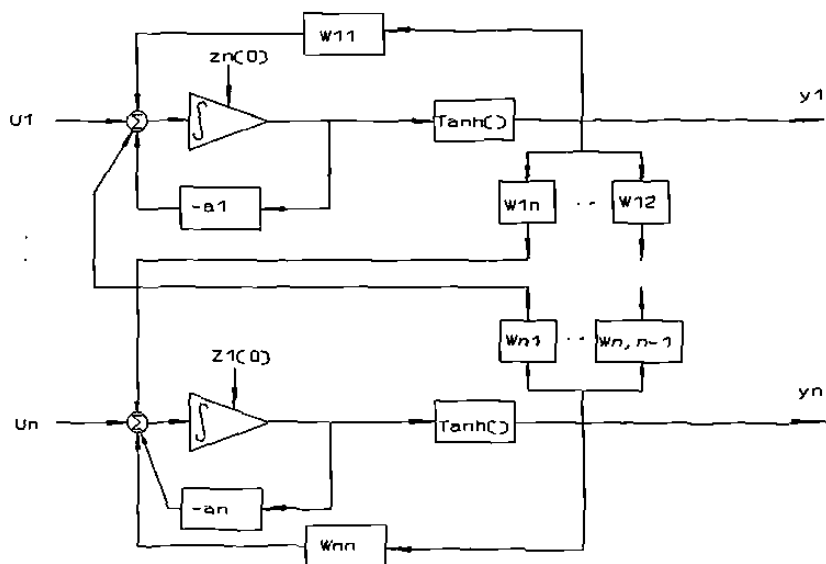


Figura 1.7 Representación de la red neuronal de Hopfield de n estados.

CAPITULO 2

PRELIMINARES

En este capítulo se describen definiciones, teoremas y conceptos que serán de utilidad en capítulos posteriores. Para este fin se consideran dos secciones: la primera se compone de preliminares matemáticos; en la segunda se plantea el problema de aproximación de funciones mediante redes neuronales.

2.1 PRELIMINARES MATEMATICOS

El contenido fue recopilado de varias fuentes, principalmente de [12], [13], [14], [15], [16]; la intención es revisar material que servirá de base para hacer más claro el contenido de los siguientes capítulos.

Teorema 1. Independencia lineal.- Una condición necesaria y suficiente para que el conjunto de vectores $\{ x_1, x_2, \dots, x_n \}$ sea linealmente independientes es que la

expresión $\sum_{k=1}^n \alpha_k x_k = 0$ implique $\alpha_k = 0 \forall k = 1, \dots, n$.

Prueba: Ver [12]

CAPITULO 2: PRELIMINARES

Def. 1. Base.- Un conjunto S de vectores linealmente independientes se denomina base de un espacio lineal X , si el conjunto de vectores S genera todo el espacio X ; es decir cualquier elemento $x \in X$ puede ser representado unívocamente como una combinación lineal de los elementos de S .

Un espacio lineal X se denomina de dimensión finita si tiene una base finita (un conjunto S con un número finito de elementos); de otra forma se denomina de dimensión infinita.

Def. 2. Métrica.- Métrica o distancia es una función no negativa $d()$ definida sobre un conjunto X que cumple con:

1) $d(x, y) = 0$ si y solo si $x=y$

2) $d(x, y) = d(y, x)$

3) $d(x, z) \leq d(x, y) + d(y, z)$

$\forall x, y, z \in X$.

Def. 3. Norma.- Una funcional $\| \cdot \|: X \rightarrow \mathbf{R}$ definida sobre el espacio lineal X es una norma si cumple con las siguientes propiedades:

1) $\|x\| \geq 0 \quad \forall x \in X$

$$2) \quad |\alpha x| = |\alpha| |x| \quad \forall x \in X, \quad \forall \alpha \in \mathbb{R}, \quad |\alpha| \text{ valor absoluto}$$

$$3) \quad |x+y| \leq |x| + |y| \quad \forall x, y \in X$$

Def. 4. **Espacios $L_{p[a,b]}$.**- Los espacios $L_{p[a,b]}$ son espacios lineales normados cuyos elementos son funciones $f: \mathbb{R} \rightarrow \mathbb{R}$ integrables en el sentido de Lebesgue (funciones continuas excepto probablemente en un conjunto de medida cero, una explicación más amplia puede verse en [13]) que cumplen con:

$$L_{p[a,b]} = \left\{ f(t): \mathbb{R} \rightarrow \mathbb{R} \mid \left(\int_a^b |f(t)|^p dt \right)^{\frac{1}{p}} < \infty \right\}$$

Def. 5. **Secuencia de Cauchy.**- Una secuencia $\{ x_n \}$ en un espacio lineal normado, se dice Cauchy si $\|x_n - x_m\| \rightarrow 0$ cuando $n, m \rightarrow \infty$; equivalentemente existe un $\epsilon > 0$ y un N tal que $\|x_n - x_m\| < \epsilon \quad \forall n, m > N$.

Def. 6. **Espacio de Banach.**- Un espacio lineal normado se denomina espacio de Banach ó espacio lineal normado completo si toda secuencia de Cauchy converge.

Teorema 2 .- En un espacio de Banach un subconjunto es completo si y sólo si el subconjunto es cerrado.

Prueba: Ver [12]

CAPITULO 2: PRELIMINARES

Def. 7. Espacio Pre-Hilbert.- Un espacio pre-Hilbert es un espacio lineal sobre el cual se define un producto interior $\langle \cdot, \cdot \rangle: X \times X \rightarrow \mathbf{R}$, que satisface los siguientes axiomas

$$\forall x, y \in X \quad y \quad \forall \lambda \in \mathbf{R} \text{ (C)}.$$

P1.- $\langle x, y \rangle = \overline{\langle y, x \rangle}$, ————— *significa complejo conjugado*

P2.- $\langle x+y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$

P3.- $\langle \lambda x, y \rangle = \lambda \langle x, y \rangle$

P4.- $\langle x, x \rangle \geq 0, \quad \langle x, x \rangle = 0 \iff x=0$

Def. 8. Ortogonalidad.- Dos vectores $x, y \in X$ (pre-Hilbert) se denominan ortogonales si el producto interior $\langle x, y \rangle$ es igual a cero

Def. 9. Base Ortonormal.- Un conjunto de vectores S en un espacio pre-Hilbert \mathcal{H} se dice base ortonormal de \mathcal{H} , si S es una base de \mathcal{H} y además si $\forall s_i, s_j \in S$ con $i \neq j$, $\langle s_i, s_j \rangle = 0$ y $\|s_i\| = 1$.

Def 10. Espacio de Hilbert.- Todo espacio pre Hilbert que sea completo en la norma definida a partir del producto interior se denomina espacio de Hilbert.

CAPITULO 2: PRELIMINARES

Teorema 3. Teorema de la proyección.- Sea \mathcal{H} un espacio de Hilbert y M un subespacio cerrado de \mathcal{H} . Para cualquier vector $x \in \mathcal{H}$ existe un único vector $m_0 \in M$ tal que $\|x - m_0\| \leq \|x - m\|$ para todo $m \in M$. Aún más, una condición necesaria y suficiente para que $m_0 \in M$ sea un vector único y minimizante es que $\langle x - m_0, m \rangle = 0 \quad \forall m \in M$.

Prueba: Ver [12]

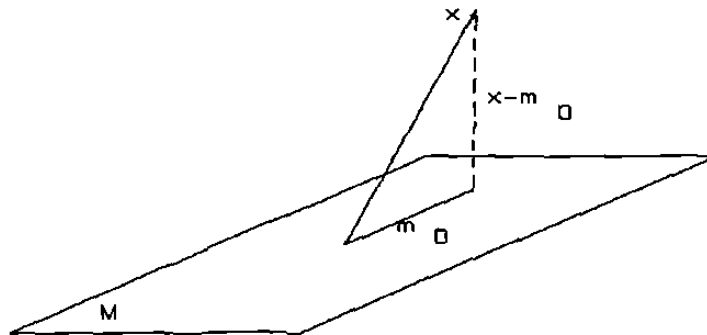


Figura 2.1 Representación gráfica del teorema de la proyección para \mathbb{R}^3

Una manera de encontrar m_0 es mediante la solución de las ecuaciones normales del problema de aproximación, las cuales resultan al aplicar las condiciones del teorema de la proyección.

Obtención de las ecuaciones normales

Suponer que se tiene $M \subset \mathcal{H}$, y que forma, generado por y_1, y_2, \dots, y_n un subespacio cerrado (pues M es de dimensión finita) de \mathcal{H} ; entonces dado $X \in \mathcal{H}$ y sabiendo que

CAPITULO 2: PRELIMINARES

podemos escribir un elemento de M como: $m = \sum_{i=1}^n \alpha_i y_i$ entonces las condiciones del

teorema de la proyección se establecen como:

$$\langle X - \alpha_1 y_1 - \dots - \alpha_n y_n, y_j \rangle = 0, \quad j = 1, \dots, n$$

Sustituyendo para todo j se obtiene:

$$\langle y_1, y_1 \rangle \alpha_1 + \dots + \langle y_n, y_1 \rangle \alpha_n = \langle x, y_1 \rangle$$

$$\langle y_1, y_2 \rangle \alpha_1 + \dots + \langle y_n, y_2 \rangle \alpha_n = \langle x, y_2 \rangle$$

⋮

$$\langle y_1, y_n \rangle \alpha_1 + \dots + \langle y_n, y_n \rangle \alpha_n = \langle x, y_n \rangle$$

Las ecuaciones anteriores son denominadas ecuaciones normales para el problema de minimización. Estas se pueden poner en forma matricial como:

$$\begin{bmatrix} \langle y_1, y_1 \rangle & \langle y_2, y_1 \rangle & \dots & \langle y_n, y_1 \rangle \\ \langle y_1, y_2 \rangle & \langle y_2, y_2 \rangle & \dots & \langle y_n, y_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle y_1, y_n \rangle & \langle y_2, y_n \rangle & \dots & \langle y_n, y_n \rangle \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \langle x, y_1 \rangle \\ \langle x, y_2 \rangle \\ \vdots \\ \langle x, y_n \rangle \end{bmatrix}$$

el problema de aproximación queda resuelto al obtener los α_i , lo cual requiere que la matriz de Gram:

$$G = G(y_1, y_2, \dots, y_n) = \begin{bmatrix} \langle y_1, y_1 \rangle & \dots & \langle y_n, y_1 \rangle \\ \vdots & \ddots & \vdots \\ \langle y_1, y_n \rangle & \dots & \langle y_n, y_n \rangle \end{bmatrix}$$

tenga inversa.

Proposición 1. El determinante de la matriz $G\{y_1, \dots, y_n\}$ es diferente de cero \Leftrightarrow los vectores (y_1, \dots, y_n) son linealmente independientes.

Prueba: Ver [12].

El cálculo de los coeficientes para formar m_0 se obtienen al invertir la matriz de Gram y multiplicar esta inversa por el vector, cuyos elementos se forman con los productos interiores entre el vector a aproximar y los elementos del conjunto M . Sin embargo, el camino más común para encontrar estos coeficientes es mediante la ortonormalización de Gram-Schmidt, pues calcular la inversa de la matriz de Gram, cuando se tienen dimensiones grandes pueden presentar problemas numéricos. El procedimiento de ortonormalización de Gram-Schmidt es presentado en la prueba del siguiente teorema.

Teorema 4. Teorema de Gram-Schmidt [12].- Sea $\{X_i\}$ una secuencia finita o contable de vectores linealmente independientes en un espacio pre-Hilbert X . Entonces existe una secuencia ortonormal $\{e_i\}$ tal que, para cada n el espacio generado por los n primeros e_i es el mismo que el generado por los n primeros X_i .

Prueba: Para el primer vector tomemos:

$$e_1 = \frac{X_1}{\|X_1\|}$$

el cual obviamente genera el mismo espacio que X_1 .

Para X_2 primero se calcula

$$Z_2 = X_2 - \langle X_2, e_1 \rangle e_1$$

$$e_2 = \frac{Z_2}{\|Z_2\|}$$

Por cálculo directo se verifica que $\langle e_2, e_1 \rangle = 0$. El vector Z_2 no puede ser cero puesto que X_2 y e_1 son linealmente independientes. e_2 y e_1 generan el mismo espacio que X_1 y X_2 desde que X_2 puede ser expresado como combinación lineal de e_1 y e_2 . Los demás e_i son definidos por inducción como:

$$Z_n = X_n - \sum_{i=1}^{n-1} \langle X_n, e_i \rangle e_i$$

$$e_n = \frac{Z_n}{\|Z_n\|}$$

De nuevo se puede verificar que $\langle Z_n, e_i \rangle = 0 \quad \forall i < n$, y Z_n es no cero pues es combinación lineal de vectores linealmente independientes. Por inducción se observa que el espacio generado por los e_i es similar al generado por los X_i .



Si el espacio M del teorema de la proyección estuviera formado por vectores

ortonormales

$$\langle e_i, e_j \rangle = 0 \quad \forall i \neq j \quad \text{y} \quad \langle e_i, e_i \rangle = 1$$

$$\alpha_i = \langle X, e_i \rangle$$

(Los α_i son llamados coeficientes de Fourier).

Entonces:

$$m_0 = \hat{Y} = \sum_{i=1}^n \langle X, e_i \rangle e_i$$

es la mejor aproximación de X .

Para complementar la descripción del problema de aproximación es necesario introducir las siguientes definiciones.

Def. 11. **Subconjunto denso** [12],[16].- Un subconjunto D de un espacio lineal normado X se dice denso en X si para cada elemento $x \in X$ y cada $\epsilon > 0 \exists d \in D$ con

$$\|x - d\| < \epsilon$$

Si D es denso en X entonces hay puntos de D arbitrariamente cercanos a cada $x \in X$.

CAPITULO 2: PRELIMINARES

Def. 12. **Espacio separable** [12],[16].- Un espacio lineal normado se dice separable si contiene un subconjunto denso contable.

Def. 13. **Excitación persistente** [13].- Un vector $\phi(t): \mathbf{R}_+ \rightarrow \mathbf{R}^n$ excita persistente si existen constantes α_1, α_2 y $\delta > 0$ tales que:

$$I\alpha_2 \geq \int_{t_0}^{t_0+\delta} \phi(\tau)\phi(\tau)^T d\tau \geq I\alpha_1 \quad \forall t_0 \geq 0$$

2.2 EL PROBLEMA DE APROXIMACIÓN

El problema de aproximación planteado en esta sección se refiere a la aproximación de funciones mediante redes neuronales. En el siguiente capítulo se abordara este problema con más detalle.

La siguiente formulación del problema es la usual y se encuentra en varios trabajos recientes ([14], [15]).

Problema de aproximación

Sea $f(x)$ una función continua con $x \in \mathbf{R}$ y $F(x, v)$ una red neuronal cuyos parámetros se representan por v ; el problema consiste en determinar (si es que existen) los parámetros $v' \in \mathcal{P}$ tal que:

CAPITULO 2: PRELIMINARES

$$\partial(F(v^*, x), f(x)) \leq \epsilon \quad \forall v \in \mathcal{P}$$

para algún $\epsilon > 0$, donde \mathcal{P} es el conjunto de valores admisibles para v y $\partial(\cdot, \cdot)$ es una métrica.

Como se verá en el siguiente capítulo los parámetros v^* de la red neuronal $F(x, v)$ sí existen. También se repasarán los resultados reportados acerca de la existencia de la aproximación, así como los métodos para construir la aproximación.

CAPITULO 3

Aproximación de Funciones

Dentro de la teoría de aproximación de funciones existen varios resultados, basados en diferentes tipos de polinomios [13], series de potencias, etc. Como ejemplo pueden citarse los polinomios de Chebyshev de primer y segundo tipo, los de Legendre, los de Hermite, y los de Laguerre. En este capítulo se analizan aproximaciones basadas en redes neuronales.

La utilización de redes neuronales para la aproximación de funciones fue demostrada en 1988 cuando varios autores publicaran resultados al respecto, para diferentes clases de redes neuronales. Algunos de estos autores fueron Cybenko [3], Funahashi [4], Hornik [5], Stinchcombe y White [6], Carroll y Dickinson [7].

A continuación, se revisarán primero algunos de los principales resultados a cerca de utilizar redes neuronales para la aproximación de funciones; enseguida se procederá a repasar algunos intentos para construir la solución. Cabe aclarar que no hay metodología que asegure que se obtendrá la mejor aproximación en algún sentido (Cuando se calculan los pesos minimizando el error cuadrático de la salida, por ejemplo) para el problema ya planteado; esto debido a que, en general, el problema de aproximación por redes neuronales no tienen un mínimo único.

3.1 REDES NEURONALES COMO APROXIMADORES

En esta sección se resumen algunos resultados referentes al uso de redes neuronales como aproximadores; estos resultados se refieren a la red neuronal sigmoideal con una capa de procesamiento (RNS1CP). Se ha seleccionado esta estructura por ser la más simple. El siguiente teorema demostrado en [3] aborda el tema.

Teorema [3]: Sea $\sigma(\cdot)$ cualquier función sigmoideal continua. Entonces sumas de la forma:

$$G(U) = \sum_{j=1}^n \beta_j \sigma(w_j^T U + \theta_j)$$

son densas en $C(I_n)$ (espacio de funciones continuas con soporte en el hiperplano formado por I_n). En otras palabras, dada una función $F \in C(I_n)$ y $\epsilon > 0$, existe una $G(x)$, para la cual:

$$|G(U) - F(U)| < \epsilon \quad \forall U \in I_n$$

Prueba : Ver [3]

CAPITULO 3: APROXIMACION DE FUNCIONES

Otro teorema referente a lo mismo es:

Teorema [4]: Sea $\phi(U)$ una función continua no constante monotónicamente creciente y acotada, y sea K un subconjunto compacto de \mathbb{R}^n y $f(u_1, \dots, u_n)$ una función continua real sobre K . Entonces para todo $\epsilon > 0$ existe un entero N y constantes reales $\alpha_i, \gamma_i, w_{ij}$ ($i=1, \dots, n$) tal que:

$$\hat{f}(u_1, u_2, \dots, u_n) = \sum_{i=1}^N \alpha_i \phi \left(\sum_{j=1}^n w_{ij} u_j - \theta_i \right)$$

satisface:

$$\max_{u \in K} | f(u_1, \dots, u_n) - \hat{f}(u_1, \dots, u_n) | < \epsilon$$

En la siguiente tabla se resumen algunos de los resultados principales con los que se cuenta a la validez de las aproximaciones:

CAPITULO 3: APROXIMACION DE FUNCIONES

RN1CP	Espacio lineal en el que la RN1CP es densa	Referencias
σ Sigmoidal monotónica	$C_{[a,b]}$	[4] [5]
σ Sigmoidal continua	$C_{(I_n)}$	[3]
σ Sigmoidal continua $\sigma \in L_1(\mathbf{R}), \int \sigma(t)dt \neq 0$	$L^1(I_n)$	[3]
σ Sigmoidal continua	$L^2(I_n)$	[5] [7]
σ Sigmoidal monotónica	$L^p(I_n)$ $1 \leq p \leq \infty$	[5]
$\sigma \in L_1(\mathbf{R}, \lambda) \cap L_p(\mathbf{R}, \lambda)$ $\int \sigma d\lambda \neq 0$	$L_p(\mathbf{R})$	[6]
$\sigma \in L_1(\mathbf{R}, \lambda)$, continua $\int \sigma d\lambda \neq 0$	$C_{[a,b]}$	[6]

De la tabla anterior podemos observar que existen algunos resultados interesantes, tales como el de que la RNS1CP resulta ser densa en L_2 . Otro hecho importante es que la función de activación no tiene que ser sigmoidal y no necesariamente continua; basta con que pertenezca al espacio L_p donde se realiza la aproximación. Una observación

CAPITULO 3: APROXIMACION DE FUNCIONES

interesante es que para la RNH aún no se ha intentado buscar resultados análogos; de encontrarlos pueden ser muy importantes, pues se trata de una red neuronal dinámica y la aproximación sería en un espacio de operadores, esto será tratado más adelante en el capítulo 4.

Sin embargo, los resultados antes citados no dan pruebas constructivas. Esto significa que no se cuenta con una metodología que permita encontrar los pesos tales que se encuentre la solución que minimiza el error; las pruebas se relacionan con la existencia de solución. A pesar de esto se han hecho algunos intentos por encontrar la solución al problema de aproximación utilizando redes neuronales, las cuales serán repasados en la siguiente sección.

3.2 CONSTRUCCION DE LA APROXIMACION

Básicamente en esta sección serán revisados los trabajos [8], [10] y [18] y [19] referentes al tema de aproximación de funciones, así como el desarrollo de algunas ideas que giran alrededor de estos trabajos.

No es objetivo de este capítulo el de dar solución acerca de la construcción de la aproximación sino, el de revisar lo que se ha podido hacer al respecto y cuales son las

CAPITULO 3: APROXIMACION DE FUNCIONES

dificultades al buscar esta solución.

3.2.1 Aproximación por mínimos cuadrados: Primer planteamiento de Stubberud y Wabgaonkar.

La idea para este planteamiento fue introducida en [8] en 1990; para esto utilizan una RN1CP con función de activación sigmoideal. Algunos parámetros de la red (los que son argumento de la función de activación) son arbitrariamente asignados; después se calculan los parámetros restantes (β_i) que minimizan un criterio seleccionado.

La ecuación que describe a la red esta dada por:

$$\hat{Y} = \sum_{i=1}^N \beta_i \sigma \left(\sum_{j=1}^n w_{ij} u_j + \theta_i \right)$$

donde los valores w_{ij} y θ_i constantes, han sido seleccionados de manera arbitraria. Alguna asignación "natural" cuando no se tiene ninguna otra información puede ser el asignar como valor a w_{ij} alguno de los patrones de entrada [8]. Más adelante se describirán condiciones necesarias para que esta asignación permita la utilización del procedimiento de mínimos cuadrados.

CAPITULO 3: APROXIMACION DE FUNCIONES

En cuanto al número de nodos (neuronas) a utilizar, en [8] proponen que estos sean tantos como el número de pares entrada-salida a "recordar"; sin embargo, para un mapeo continuo de una función entrada-salida, aún en un dominio compacto (cerrado y acotado si se trata de \mathbb{R}^n), se tendrán un número infinito de nodos. La asignación del número de nodos tiene que hacerse por prueba y error, al igual que la de los valores del argumento de la función de activación, pues no existe procedimiento que permita escoger óptimamente estos.

Por simplicidad vamos a considerar la aproximación de una función escalar de una sola variable $Y = F(u)$, esto sin pérdida de generalidad pues las extensiones a funciones multivariadas son directas. Con esto la red a utilizar se reduce a:

$$\hat{Y} = \sum_{i=1}^N \beta_i \sigma(w_i u + \theta_i)$$

Tomando un número suficiente de nodos de la función a aproximar y sabiendo que la RN1CP es continua, para algún intervalo del dominio de la función a aproximar, solo se requiere utilizar puntos de entrada "ceranos" en todo este intervalo y unirlos a los correspondientes puntos de la salida para obtener una buena aproximación.

Tomando k pares entrada-salida y aplicándolos a la ecuación anterior podemos reescribirla como:

CAPITULO 3: APROXIMACION DE FUNCIONES

$$\hat{Y} = \begin{bmatrix} \sigma(w_1 u^1 + \theta_1) & \dots & \sigma(w_N u^1 + \theta_N) \\ \vdots & \ddots & \vdots \\ \sigma(w_1 u^k + \theta_1) & \dots & \sigma(w_N u^k + \theta_N) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_N \end{bmatrix} = \Phi B$$

CASO A.- Planteamiento fuera de línea

En este caso se considera la aproximación fuera de línea, en la cual se cuenta con mediciones de la forma $U_i - \hat{Z}_i$ y $U_i - Y_i$, es decir pares entrada-salidas ver figura 3.1.

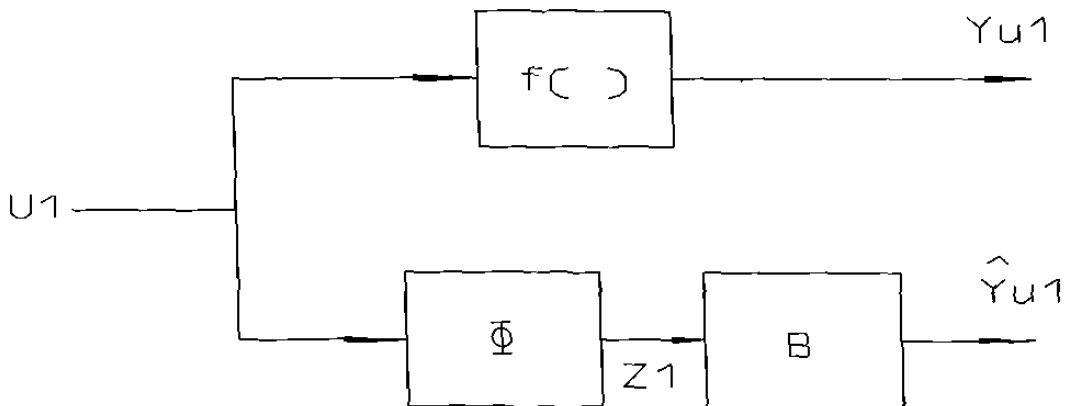


Figura 3.1 Esquema de aproximación fuera de línea

y se desea calcular **B** tal que:

CAPITULO 3: APROXIMACION DE FUNCIONES

$$Y = f(U) = \Phi B$$

$$\text{donde: } B = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$

De aquí se puede calcular los β_j , siempre y cuando se tenga un número suficiente de muestras distintas y que exista la pseudo inversa de la matriz de información Φ . Algunas condiciones necesarias para que ocurra esto son: que la señal de entrada no permanezca constante; los pesos $w_{ij} \neq w_{lp}$, $\theta_i \neq \theta_j$. Una vez asegurada la invertibilidad de Φ , utilizando mínimos cuadrados:

$$B = (\Phi^T * S * \Phi)^{-1} * \Phi^T * S * Y$$

donde s es una matriz de ponderación definida positiva. La utilización de mínimos cuadrados asegura que se minimiza el cuadrado del error.

Ejemplos

Como ejemplos se presentan simulaciones en las cuales se consideran funciones continuas.

Simulación 1.- En esta simulación se considera la función a aproximar: $Y = F(u) = \text{sen}(u)$, una red neuronal con 10 neuronas y 40 pares entrada-salida.

CAPITULO 3: APROXIMACION DE FUNCIONES

Los pesos seleccionados arbitrariamente se escogen con valores aleatorios entre -1 y 1, mientras que la señal de entrenamiento también es aleatoria pero con valores entre 0-5. Una vez encontrados los pesos restantes se obtienen las gráficas para dos entradas, una el tiempo $u_1 = t$ y $u_2 = \sin(10t)$ (figura 3.3 y 3.4).

Simulación 2.- La simulación 2 es realizada en las mismas condiciones que la primera; sin embargo, en este caso se utiliza otra función a aproximar; $y = f(u) = \sin(u) \cdot \cos(u)$. Los resultados son mostrados en la figura 3.5 y 3.6.

Al realizar simulaciones como las anteriores se observa que la acertada selección de un patrón de entrenamiento (pares $U_k - Y_k$) es muy importante. Esto es bien conocido dentro de la teoría de identificación, en la cual se sugieren señales ricas en componentes de frecuencia, por tal motivo fueron utilizados patrones aleatorios, los cuales parecen dar un mejor resultado.

CASO B.- Planteamiento en línea

La versión recursiva es dada por el algoritmo de mínimos cuadrados recursivos en discreto:

CAPITULO 3: APROXIMACION DE FUNCIONES

$$B_{k+1} = B_k + P_{k+1} \phi^T (Y - \phi^T B_k)$$

$$P_{k+1} = P_k + \phi (\phi^T P_k \phi + S)^{-1} \phi^T P_k$$

$$\hat{Y} = \phi^T B; \quad \phi = \begin{bmatrix} \sigma(w_1 u + \theta_1) \\ \vdots \\ \sigma(w_N u + \theta_N) \end{bmatrix}$$

Sin embargo, la implementación de las ecuaciones anteriores no es fácil, pues la ecuación para P_{k+1} tiene mal condicionamiento numérico, y requiere la utilización de las factorizaciones 'square root' para su implementación [20], [21].

Como alternativa, en este trabajo se ha modificado el planteamiento original de [8], utilizando el método del gradiente descendente en tiempo continuo. Este método resulta ser más sencillo y no tiene el problema numérico mencionado, con lo cual se puede implementar de manera directa, las ecuaciones que describen el algoritmo son:

$$Y = F(u)$$

$$\hat{Y} = \phi^T B \quad (*)$$

$$\dot{B} = -\gamma \phi (\phi^T B - f(u))$$

Al usar este método (gradiente descendente), cada vez que es aproximada una función, los parámetros β' son calculados. Estos valores permanecen acotados. Una ventaja de este método es que puede usarse para intervalos arbitrarios de tiempo, a diferencia del caso fuera de línea en el cual se tiene que volver a entrenar la red para

CAPITULO 3: APROXIMACION DE FUNCIONES

poder usarla en cada intervalo de tiempo (dependiendo del patrón de entrenamiento).

Ejemplos

Para mostrar la simplicidad del método se realizaron las siguientes simulaciones:

Simulación 3.- Considerar la función $f(t) = \text{sen}(t)$ del ejemplo 1, utilizando 10 neuronas y el cálculo de los pesos en línea por el método del gradiente descendente. Los resultados se muestran en las figuras 3.7, 3.8 y 3.9. Se presentan las salidas, los errores y algunos de los parámetros de adaptación para diferentes valores de la ganancia de adaptación. De los resultados se observa que el error decrece a medida que se aumenta la ganancia de adaptación; sin embargo, esta incrementa la sensibilidad del algoritmo y puede tener problemas si es muy grande.

Simulación 4.- En este segundo ejemplo se considera la función $f_4 = \text{Tanh}(t) + 0.1 \cdot \text{sen}(t)$ utilizando 10 neuronas y simulando para dos valores diferentes de la ganancia de adaptación.

Aquí cabe establecer dos preguntas difíciles de contestar:

¿Cuántas neuronas se requieren como mínimo para asegurar un error máximo?

¿Cómo asignar w_i^* y θ_i^* de manera óptima en algún sentido?

Más adelante se plantearán algunas reflexiones relacionadas con estas preguntas.

3.2.2 La Mejor Aproximación

A continuación se describe una de las aportaciones de el presente trabajo; se plantea "La mejor aproximación" de funciones mediante RNS1CP y el sentido en el que esta se establece, en base a un replanteamiento del problema de aproximación. Esto con la finalidad de poner las condiciones para el uso del Teorema de la proyección. Cabe aclarar que este replanteamiento no resuelve por completo el problema de aproximación; sin embargo, esto ayuda a explicar y fundamentar algunas ideas (como en que sentido se tiene la mejor aproximación, porque puede converger el algoritmo que busca los pesos, conocer la norma del error, etc.).

Definiendo cada elemento básico (neurona) como:

$$Z_i = \sigma (W_i^T u + \theta_i)$$

con $w_i \in \mathbf{R}^n$ y θ_i fijos, y la tangente hiperbólica como función de activación σ . Esta neurona pertenece a un espacio de Hilbert L_2 , como se demuestra en [5], [7]. Además un hecho importante resulta ser que combinaciones lineales de estos vectores (neuronas) son densas en L_2 [5], [7].

Replanteamiento del problema de aproximación.- Dadas n neuronas (elementos) $\{Z_i\} \in L_{2[a,b]}$ linealmente independientes y el subespacio cerrado $M \subset L_{2[a,b]}$ formado por un subconjunto $\{z_i\}$ de n neuronas y una función $Y = f(u) \in L_{2[a,b]}$, encontrar los valores

CAPITULO 3: APROXIMACION DE FUNCIONES

β_i de $\hat{Y} = \sum_{j=1}^n \beta_j z_j$ que hacen que \hat{Y} sea la mejor aproximación de Y :

$$\|f(u) - \sum_{i=1}^n \beta_i^* z_i\|_2 \leq \|f(u) - \sum_{i=1}^n \beta_i z_i\|_2$$

La norma del error esta dada por:

$$\|e\| = \|f(u) - \sum_{i=1}^n \beta_i z_i\|$$

Bajo este nuevo planteamiento el problema tiene solución única y se puede calcular metodológicamente, por la aplicación directa del teorema de la proyección. La solución es descrita por las ecuaciones normales.

Un procedimiento para encontrar la solución es:

- a) Se ortonormalizan los elementos Z_i de M , obteniendo los e_i
- b) Se calculan los coeficientes de Fourier $\langle e_i, f(u) \rangle$
- c) Se construye $\hat{Y} = \sum_{i=1}^n \langle e_i, f(u) \rangle e_i$

CAPITULO 3: APROXIMACION DE FUNCIONES

Por el teorema de Gram-Schmidt, se observa que el cálculo de los coeficientes de Fourier es equivalente al cálculo de los β_i para los vectores $\{z_i\}$ linealmente independientes. Otra alternativa puede ser la solución de las ecuaciones normales.

Algunas de las ventajas con las que se cuenta, al replantear el problema de aproximación de esta forma, son las siguientes:

- i) Se tiene un problema con solución única, garantizado por el teorema de la proyección.
- ii) Con Y conocida es posible conocer el error máximo que se tendrá.
- iii) Este error máximo resulta ser el mínimo alcanzable con el conjunto de neuronas M .
- iv) Con Y conocida, existe una metodología independiente de cualquier patrón de entrenamiento que encuentra la mejor aproximación para un conjunto de neuronas $\{z_i\}$, es decir los valores β_i .
- v) Si Y es desconocida, se pueden utilizar procedimientos recursivos para el

CAPITULO 3: APROXIMACION DE FUNCIONES

cálculo de los β_p , y este se explica en la sección 3.2.1.

Cabe aclarar que con este replanteamiento no se hacen contribuciones a las matemáticas; este enfoque mediante el uso de herramientas del análisis funcional, ha podido explicar o aclarar algunos problemas existentes (i y v) y plantear nuevos resultados (ii, iii y iv) en el uso de redes neuronales.

De nuevo se insiste en que el problema originalmente planteado aún no ha sido resuelto y las dos preguntas hechas en lo referente al número y características (como determinar los valores w_{ij} y θ_i que determinan a las neuronas) de las neuronas que se utilizan en la aproximación siguen abiertas. Reflexiones acerca de las dificultades que se tienen al intentar responder a éstas, se ofrecen en la sección 3.3.

Como aclaración del procedimiento anterior se incluye el siguiente ejemplo.

Ejemplo 3.5. Considerar la función $Y = \text{Sen}(t)$ y el conjunto cerrado $M = \{Z_1, Z_2, Z_3, Z_4, Z_5\}$ (el cual define un subespacio cerrado en L_2) donde:

$$Z_1 = \text{tgh}(0.05t + 0.05)$$

$$Z_2 = \text{tgh}(0.15t + 1)$$

$$Z_3 = \text{tgh}(0.25t + 1.5)$$

$$Z_4 = \text{tgh}(0.35t + 2)$$

$$Z_5 = \text{tgh}(0.45t)$$

CAPITULO 3: APROXIMACION DE FUNCIONES

Se desea encontrar los β_j con los que se alcanza la mejor aproximación de Y.

Dos procedimientos son posibles, el procedimiento de ortonormalización descrito anteriormente o la solución directa de las ecuaciones normales. En este caso, dado que solo se tienen 5 vectores y el posible uso de la computadora, se procedió a la solución directa de las ecuaciones normales. La matriz de Gram queda:

$$G(Z_1, Z_2, Z_3, Z_4, Z_5) = \begin{bmatrix} 7.2766 & 9.8340 & 10.208 & 10.285 & 9.5255 \\ 9.834 & 13.430 & 13.999 & 14.120 & 12.892 \\ 10.208 & 13.999 & 14.620 & 14.756 & 13.353 \\ 10.285 & 14.120 & 14.756 & 14.897 & 13.436 \\ 9.525 & 12.892 & 13.353 & 13.436 & 12.777 \end{bmatrix}$$

Los vectores B quedan:

$$B = \begin{bmatrix} -0.4827 \\ 10.378 \\ -24.317 \\ 16.104 \\ -1.557 \end{bmatrix} \quad B_1 = \begin{bmatrix} 17.523 \\ -96.478 \\ 341.82 \\ -252.56 \\ -7.2936 \end{bmatrix}$$

donde B_1 es un conjunto de parámetros calculado utilizando el procedimiento seguido en la sección 3.2.1 mediante mínimos cuadrados fuera de línea.

Para efecto de comparar la efectividad del método se obtienen las normas de los errores correspondientes:

$$|e| = 2.509 \quad |e_1| = 2.8$$

de donde se observa que se cumple lo predicho por el teorema de la proyección.

CAPITULO 3: APROXIMACION DE FUNCIONES

3.2.3 Filtro de Kalman extendido: Segundo planteamiento de Stubberud y Wabaonkar.

En esta sección se considera la aproximación de una función escalar $F: U \subset \mathbf{R}^n \rightarrow \mathbf{R}$, donde U es un subconjunto compacto de \mathbf{R}^n . El mapeo general $F: \mathbf{R}^n \rightarrow \mathbf{R}^m$ de cada componente de la salida puede ser tratado separadamente como m mapeos $\mathbf{R}^n \rightarrow \mathbf{R}$ [18].

La red neuronal utilizada para la aproximación esta caracterizada como sigue:

$$\hat{Y} = \sum_{i=1}^N \beta_i G\left(\sum_{j=1}^n w_j^i u_j + \theta_i\right) = h(z, u)$$

En este caso $G(\cdot)$ es la función de activación, que como se concluye en [6] no necesariamente tiene que ser sigmoideal cuando se quiere utilizar la red para aproximar funciones.

Una parametrización válida es:

$$G^i(t) = a_0^j + b_1^j \cos(2\pi b_2^j t) + a_1^j \sin(2\pi a_2^j t) \quad t \in \mathbf{R}$$

donde a_0 , a_1 , a_2 , b_1 y b_2 son los parámetros a estimar de la función de activación. Estos son independientemente de los parámetros w_j^i , θ_i de la red.

Se define un vector Z que contenga todos los parámetros a estimar, tanto los propios de la red como los de la función de activación. Sabiendo de [4] que existe un número de neuronas N y un vector de parámetros Z^* constantes que hace mínimo el error entre alguna función a aproximar y la red neuronal, entonces:

CAPITULO 3: APROXIMACION DE FUNCIONES

$$\dot{z}^* = 0$$

Así mismo considerando el problema de aproximación 2.1 añadiendo ruido a la salida de la red se completa el esquema presentado en [18] (figura 3.2).

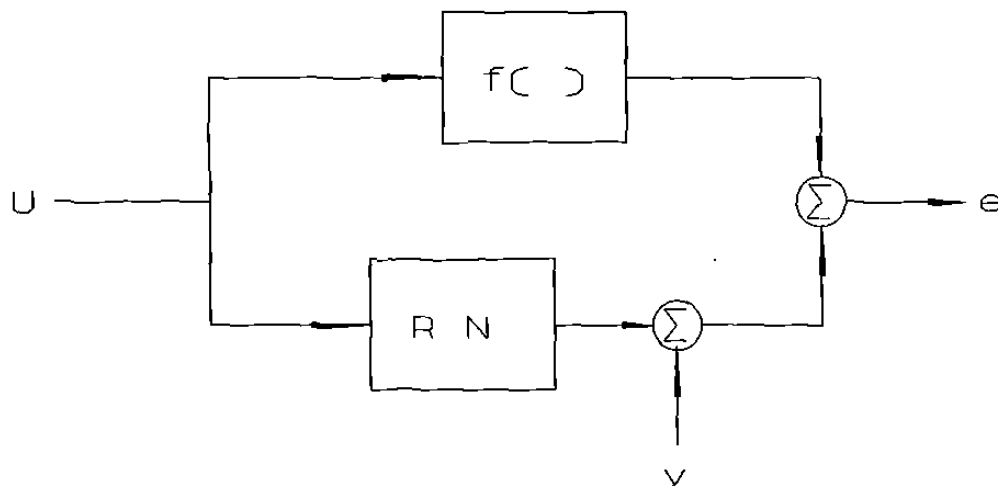


Figura 3.2 Esquema de aproximación en presencia de ruido

El ruido es gaussiano, con media cero y covarianza R . Enseguida se describirán las ecuaciones del filtro de Kalman extendido.

El filtro de Kalman extendido es un estimador de estado para una clase de sistemas no lineales; sin embargo, en general este filtro no es óptimo y dada la linealización bajo la que es llevado a cabo, el filtro puede diverger, con lo que la aplicación de este filtro debe llevarse a cabo con cuidado. [21], [22].

CAPITULO 3: APROXIMACION DE FUNCIONES

En resumen, las ecuaciones son las siguientes [23]:

Modelo del $X(t) = f(x(t),t) + W(t) ; W(t)$

Sistema $Y(t) = g(x(t),t) + v(t) ; v(t)$

Condiciones $X(0)$

iniciales

correlación $E(W(t) V^T(\tau)) = 0 \quad \forall t, \tau$

del ruido

Ecuaciones del $\dot{\hat{x}} = f(\hat{x}(t), t) + K(t) * [Y(t) - g(\hat{x}(t), t)]$

estado estimado

Ecuación de la $\dot{P}(t) = F(\hat{x}(t), t) P(t) + P(t) F^T(\hat{x}(t), t) + Q$

covarianza del $-P(t) H^T(\hat{x}(t), t) R^{-1}(t) H(\hat{x}(t), t) P(t)$

error

CAPITULO 3: APROXIMACION DE FUNCIONES

Ecuación de la ganancia del filtro

$$K(t) = P(t) H^T(\hat{x}(t), t) R^{-1}(t)$$

Donde:

$$F(\hat{x}(t), t) = \left. \frac{\partial f(x(t), t)}{\partial x(t)} \right|_{x(t) = \hat{x}(t)}$$

$$H(\hat{x}(t), t) = \left. \frac{\partial g(x(t), t)}{\partial x(t)} \right|_{x(t) = \hat{x}(t)}$$

La aplicación de las ecuaciones anteriores a la estimación del estado Z^* queda:

$$\hat{Z}^* = PH^T R^{-1} [Y + v - h(\hat{Z}^*, u)]$$

$$\dot{P} = -PH^T R^{-1} H P$$

$$h(\hat{Z}^*, u) = \hat{Y} \quad y \quad H = \left. \frac{\partial h(Z^*, u)}{\partial Z^*} \right|_{Z^* = \hat{Z}^*}$$

donde K es la ganancia del filtro, P es la matriz de aproximación de la varianza del error, R es la covarianza del ruido u.

Este planteamiento del filtro de Kalman extendido resulta ser prometedor, sin embargo, como ya se mencionó su uso es restringido y requiere estudios futuros. Además su implementación tiene de nuevo algunos problemas numéricos semejantes a los de mínimos cuadrados recursivos ([20] [22]).

3.3 Dimensionalidad

En esta sección se resumen algunas de las reflexiones hechas (las cuales son resultado del estudio y simulación de los modelos de redes neuronales vistos) acerca de la dificultad para seleccionar el número de neuronas así como los parámetros de estas, de tal forma que resulten óptimas, en algún sentido.

Considerando el espacio euclidiano tridimensional y un vector arbitrario en éste, (figura 3.3), se deben escoger 2 vectores de entre 3 posibles sobre los ejes X, Y ó Z tal que se logre la mejor aproximación de R. Claramente no resulta lo mismo la selección de vectores en el plano XZ que en el plano XY ó YZ, pues la proyección ortogonal sobre cada plano es diferente. En este espacio euclidiano de dimensión 3 tenemos el mismo número de elecciones; sin embargo, el problema se complica cuando podemos escoger vectores arbitrarios y no precisamente sobre los ejes. Esto es en cuanto a que neuronas escoger; en cuanto al número también se tienen dificultades pues un par de vectores en el plano Z resulta ser peor aproximación que uno solo sobre el eje Y.

Todo lo anterior ocurre sobre un espacio de dimensión finita, pero el espacio en los que se definen las redes neuronales son de dimensión infinita ($C_{[a,b]}$, $L_p_{[a,b]}$) con lo que el problema se vuelve aún más complicado.

CAPITULO 3: APROXIMACION DE FUNCIONES

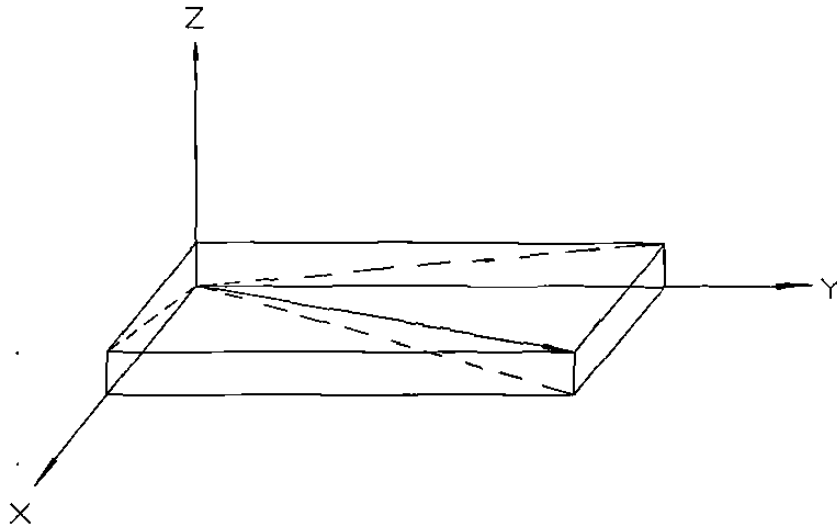


Figura 3.3 Representación de un vector en el espacio euclidiano tridimensional

Posibles respuestas a las preguntas P1 y P2 pudieran ser encontradas considerando que las redes neuronales resultaron ser densas en un espacio de Hilbert separable, de aquí alguna solución puede encontrarse al descifrar la manera en el que las "neuronas" se pueden numerar.

CAPITULO 3: APROXIMACION DE FUNCIONES

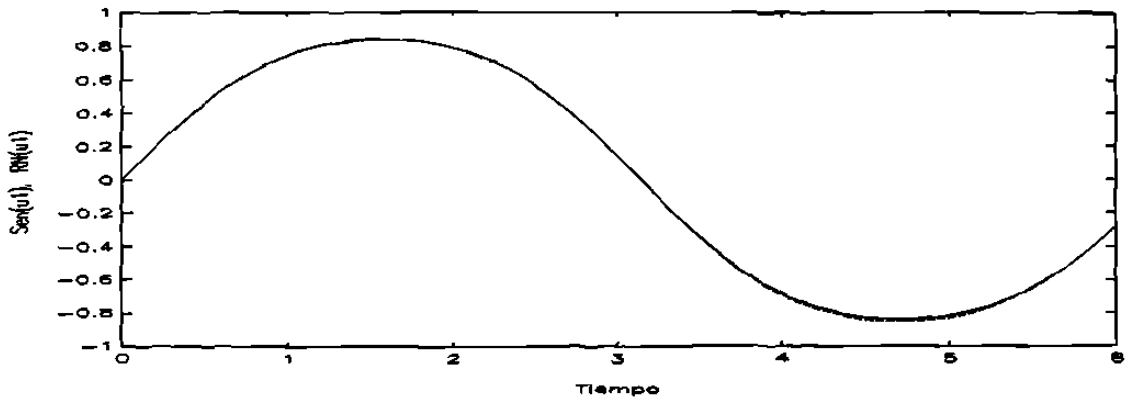


Figura 3.4a Gráfica de las salidas, ejemplo 1.

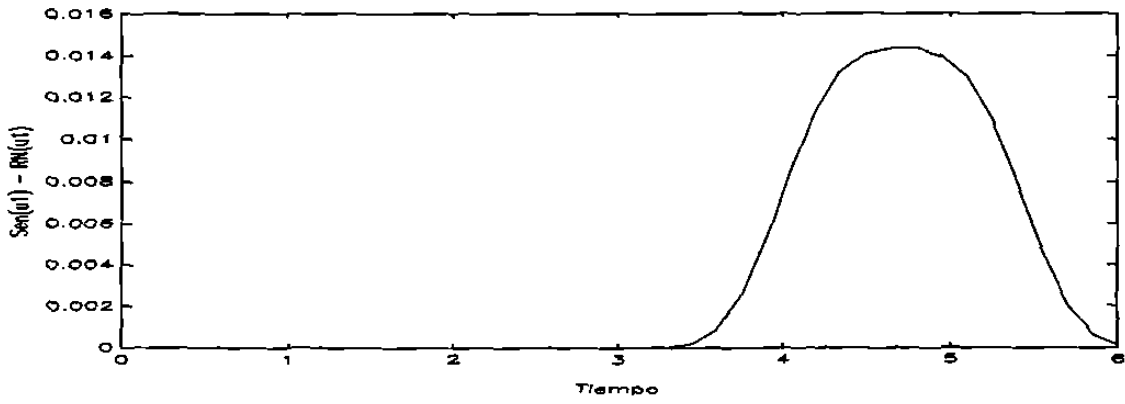


Figura 3.4b Gráfica del error, ejemplo 1

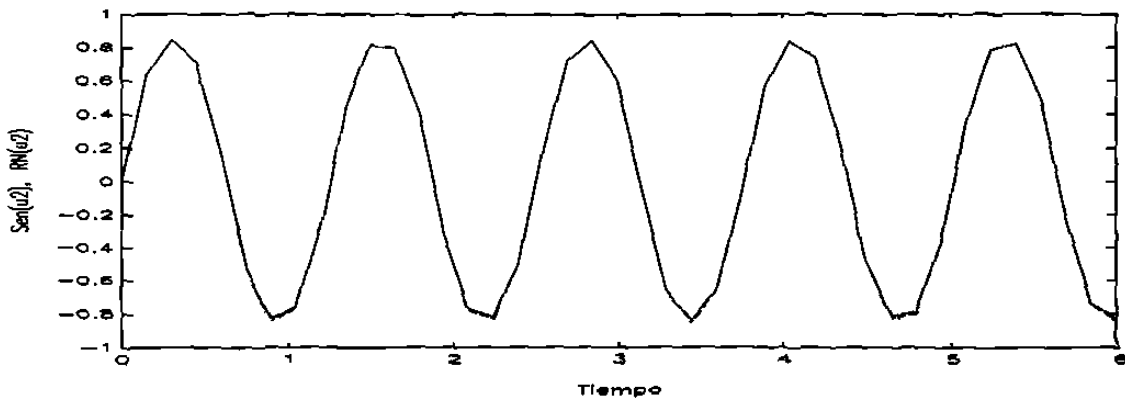


Figura 3.5a Gráfica de las salidas, ejemplo 1.

CAPITULO 3: APROXIMACION DE FUNCIONES

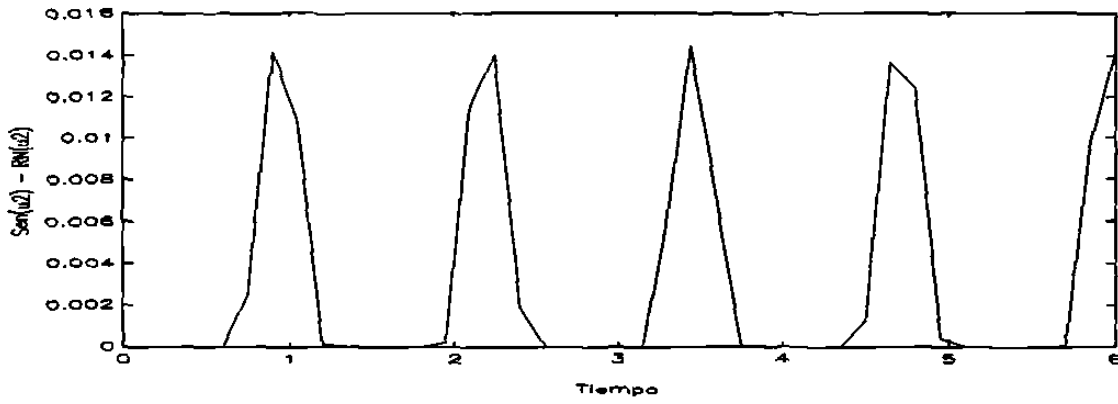


Figura 3.5b Gráfica del error con u_2 , ejemplo 1.

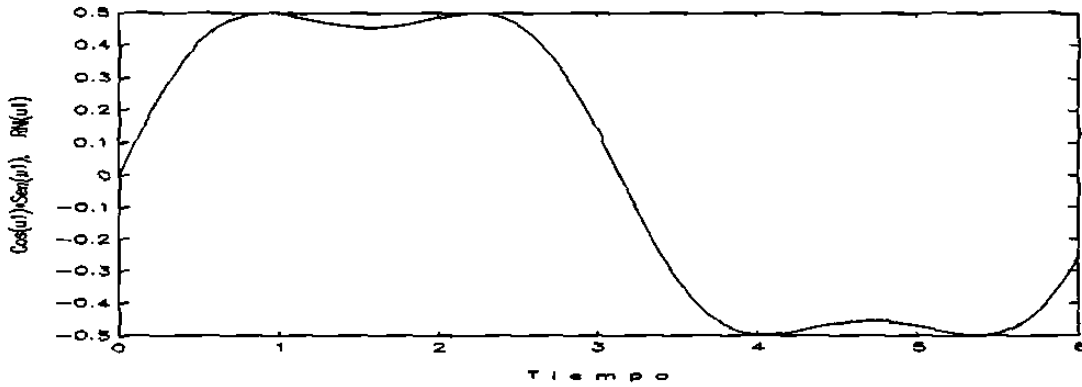


Figura 3.6a Gráfica de las salidas con u_1 , ejemplo 2.

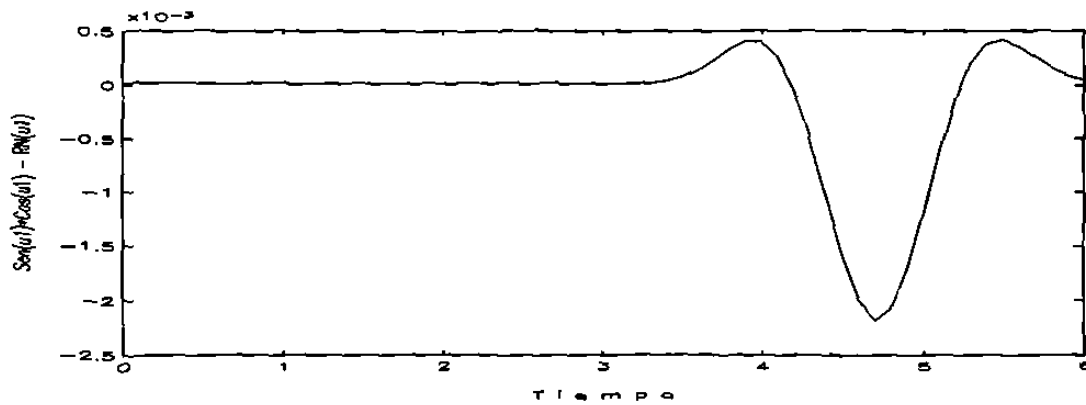


Figura 3.6b Gráfica del error con entrada u_1 , ejemplo 2.

CAPITULO 3: APROXIMACION DE FUNCIONES

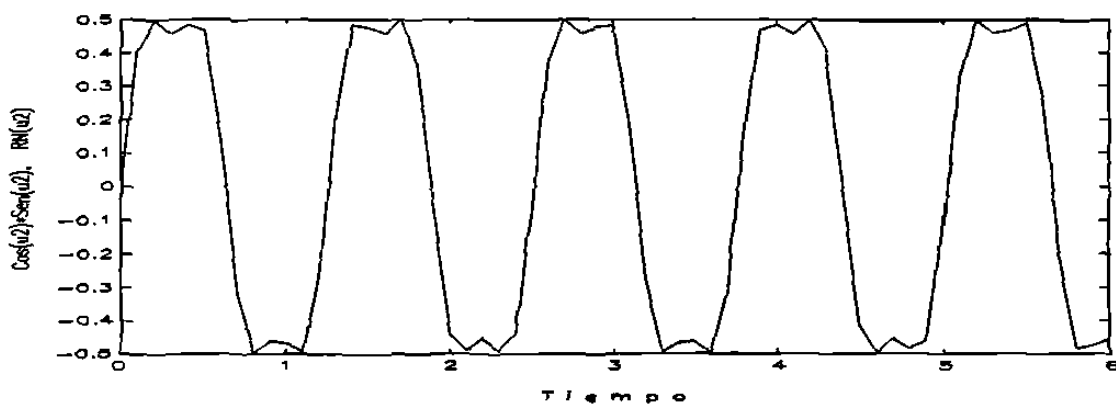


Figura 3.7a Gráfica de las salidas con u_2 , ejemplo 2.

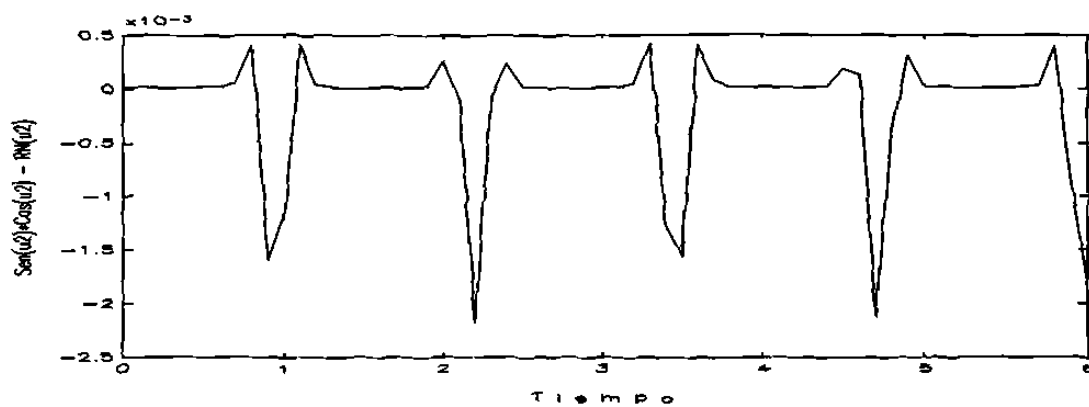


Figura 3.7b Gráfica del error con u_2 , ejemplo 2.

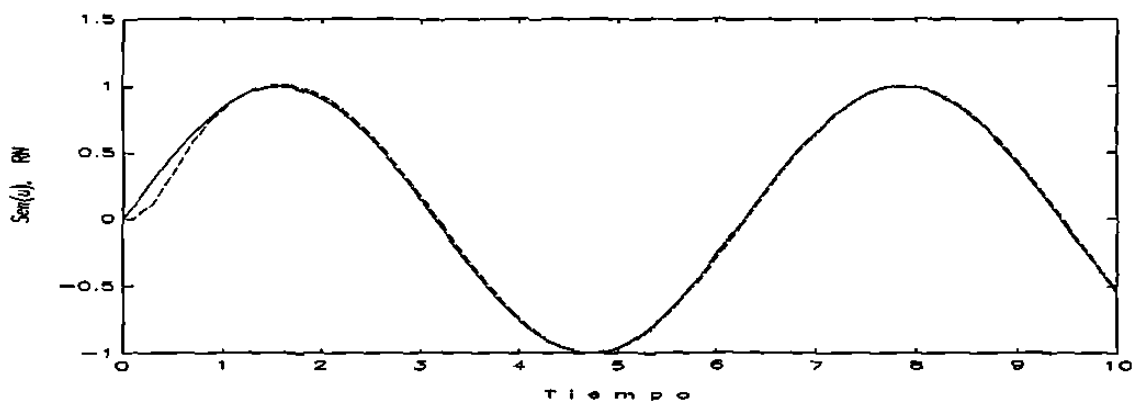


Figura 3.8a Gráfica de las salidas, ejemplo 3.

CAPITULO 3: APROXIMACION DE FUNCIONES

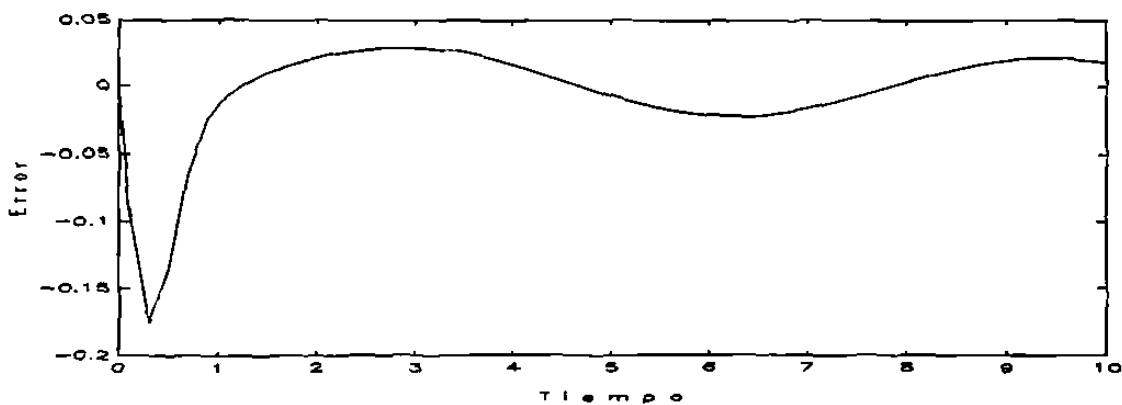


Figura 3.8b Gráfica del error, ejemplo 3.

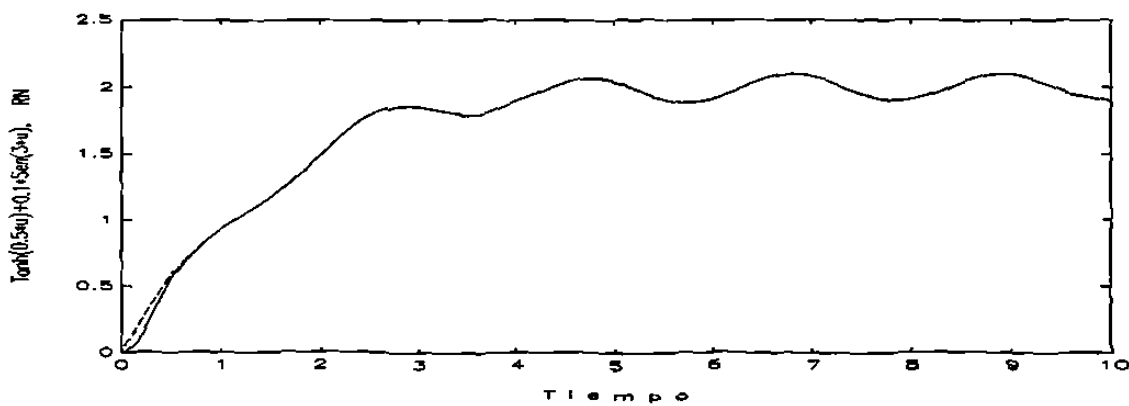


Figura 3.9a Gráfica de las salidas, ejemplo 4.

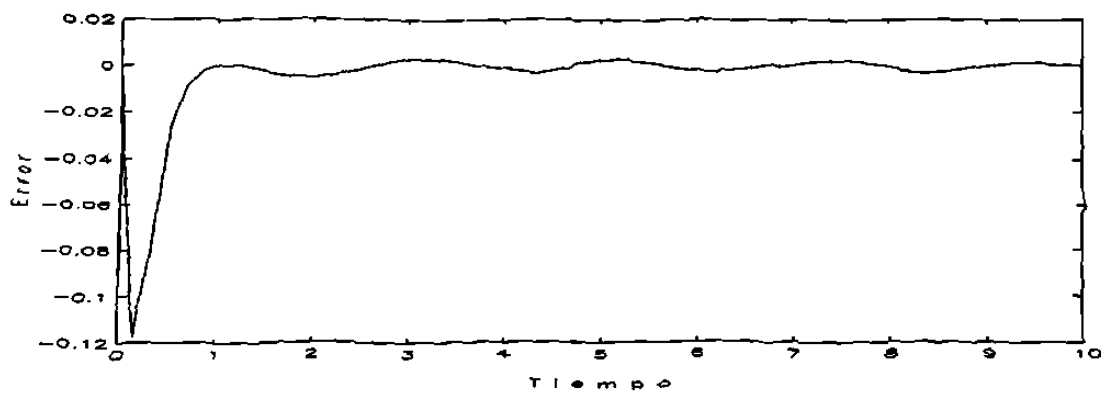


Figura 3.9b Gráfica del error, ejemplo 4.

Capítulo 4

Modelado de sistemas

En este capítulo se desarrolla la aportación principal de la tesis; el modelado de sistemas dinámicos mediante redes neuronales. El objetivo principal de este trabajo no es el de obtener contribuciones originales en el tema, sino el de clarificar los resultados existentes y establecer los problemas que aún continúan abiertos.

De nueva cuenta y como en el caso de aproximación de funciones las redes neuronales resultan ser una alternativa y no el único camino para el modelado de sistemas dinámicos no lineales; otra alternativa la presentan los operadores de Volterra. Sin embargo, el manejo de estos operadores resulta ser poco práctico. Recientemente varios autores han abordado el tema de identificación (Modelado) de sistemas dinámicos usando redes neuronales [16], [25] y [26]. Aunque aún no existe un resultado contundente en este sentido, si hay evidencias, vía simulaciones, referentes a su potencialidad.

4.1 Utilización las redes neuronales estáticas

En esta sección se resumen algunos intentos acerca del uso de redes neuronales estáticas para la aproximación (Modelado) de sistemas dinámicos. Estos intentos miden en cierto sentido el alcance de las redes estáticas al intentar este tipo de aproximaciones.

CAPITULO 4: MODELADO DE SISTEMAS

Como se vió en el capítulo anterior las redes neuronales estáticas tienen probadas cualidades para la aproximación de funciones. Sin embargo, cuando se trata de la aproximación de sistemas dinámicos resultan ser insuficientes. Esto se pone en evidencia en presencia de entradas constantes: el sistema dinámico puede tener una salida variante en el tiempo, mientras que la red tendrá una salida constante. A pesar de esto algunos autores han encontrado lugar para aplicarlas.

4.1.1 Identificación paramétrica

En esta subsección se describe un trabajo presentado en [16]. En este trabajo se sugieren estructuras de identificación, así como estructuras de controladores usando redes neuronales para el control adaptable de sistemas dinámicos desconocidos. Para esto se supone una planta estable con parametrización conocida, pero con los valores de los parámetros desconocidos, en este caso funciones.

La clase de sistemas dinámicos que pueden ser generados usando redes neuronales, pueden ser representados en términos de matrices de transferencia de sistemas lineales ($W(s)$) y operadores estáticos no lineales ($N(u)$) [16], formándose los sistemas con sumatorias; multiplicaciones por una constante e integradores. Basado en esto se consideran las siguientes estructuras (fig. 4.1)

En donde las funciones $f(\cdot)$ y $g(\cdot)$ son substituidas por RNM, suponiendo esta substitución posible. La utilización de estas estructuras requiere que se seleccionada una

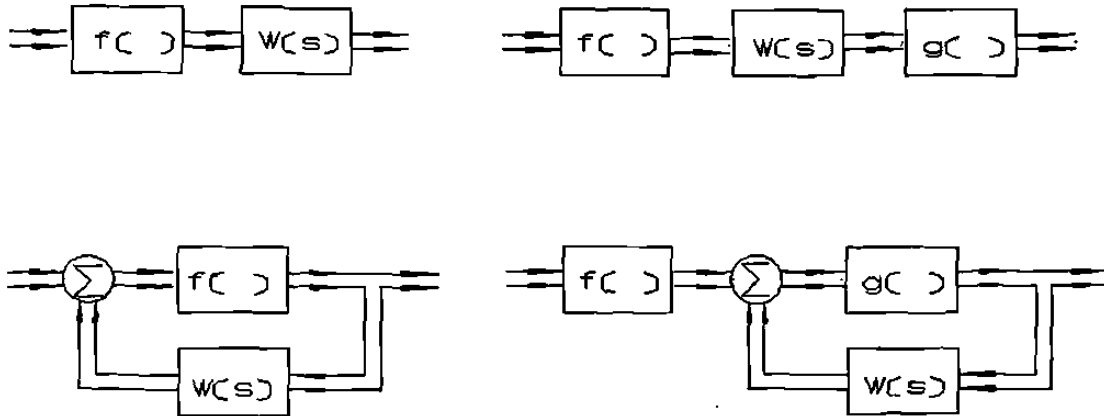


Figura 4.1 Esquemas que han sido propuestos para la identificación de sistemas no lineales

de ellas, lo cual no siempre resulta fácil.

Estos modelos conteniendo redes neuronales fueron motivados por modelos que han sido usados en la literatura de sistemas adaptables, para la identificación y control de sistemas lineales; y pueden ser considerados como una generalización de estos a sistemas no lineales [16].

La puesta en práctica de estos algoritmos requiere del cálculo de los parámetros de la red. Este cálculo puede llevarse a cabo mediante el procedimiento de retropropagación ó el de retropropagación dinámica; sin embargo, el uso de retropropagación dinámica, requiere un número muy grande de pasos para convergencia (entre 25,000 y 100,000) y nada asegura que se ha convergido a los parámetros deseados.

Finalmente, para el desarrollo de este enfoque se consideraron las siguientes hipótesis de trabajo:

- 1) La planta debe ser BIBO-estable.
- 2) El identificador es estable.

CAPITULO 4: MODELADO DE SISTEMAS

- 3) Se supone que las funciones a aproximar ($F(\cdot)$ y $g(\cdot)$) son aproximables por RNM arbitrariamente bien.
- 4) La planta se supone de fase mínima y grado relativo uno.

4.1.2. Identificación en estado estacionario

Este enfoque es utilizado en [19]; la idea central consiste en la observación de que la respuesta en estado estacionario puede ser representada a través de un operador estático.

Para lograr la representación se considera un operador estático continuo y acotado $f: E^r \rightarrow E^m$, el cual representa una relación estática:

$$z = f(u)$$

Se supone que Z y u son medibles. Para la identificación se requiere del uso de una red neuronal $f: E^r \rightarrow E^m$ de tal forma que.

$$\delta Z = f(u) - \hat{f}(u)$$

El objetivo es minimizar δz . La idea de aproximar el estado estacionario reduce el problema de identificación a uno de aproximación de funciones, el cual puede desarrollarse mediante algún procedimiento visto en el capítulo anterior, en particular el

del filtro de Kalman extendido.

A manera de comentario se cuestiona la utilidad de este tipo de aproximación, pues aproximaciones más interesantes deban incluir la dinámica del sistema. Esta aproximación es realmente llevada a cabo al rededor de puntos de equilibrio estables del sistema, un trabajo sobre el tema es desarrollado en [25] en el se pretende la regulación de sistemas dinámicos, de nuevo las restricciones son fuertes y se tienen dificultades con equilibrios inestables.

4.2 Utilización de redes neuronales dinámicas

Como se vio en la sección anterior, la aproximación de sistemas dinámicos utilizando redes neuronales estáticas, no es satisfactoria en general, y el uso de estas redes es restringido.

El motivo antes citado nos mueve a considerar el uso de redes neuronales dinámicas continuas como las de Hopfield, sugerido en [16]. Una proposición más concreta se encuentra en [26]. En lo que resta de este capítulo se propondrá una manera de utilizar las redes neuronales dinámicas de Hopfield para el modelado de sistemas dinámicos no lineales, donde solo hay información de las señales entrada-salida, así como el planteamiento de una conjetura ([26]) al respecto. Esto constituye la principal aportación de la tesis.

4.2.1. El operador de Hopfield

Considerando la red neurona de Hopfield descrita en el capítulo 2 tomando en cuenta una breve modificación sobre la salida se obtiene:

$$\frac{dZ}{dt} = -AZ + W g(Z) + U \quad Z(0) \in \mathbb{R}^n \quad U, Y \in \mathbb{R}^n \quad (4.1)$$

$$Y = B^T g(Z)$$

$$A = \text{diag}\{\alpha_i\}; \quad W = \begin{bmatrix} w_{11} & \dots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{n1} & \dots & w_{nn} \end{bmatrix}; \quad g(Z) = \begin{bmatrix} \text{tgh}(z_1) \\ \vdots \\ \text{tgh}(z_n) \end{bmatrix}; \quad U = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} \quad u_i = u_j \quad \forall ij$$

$$B^T = [\beta_1, \dots, \beta_n] ; A, W \in \mathbb{R}^{n \times n}$$

donde la salida ahora pertenece a $L_{p[a,b]}$ y no a $L^p_{p[a,b]}$, es decir, ahora es un escalar; el vector B se ha agregado con esta finalidad. Aunque esto incrementa el número de parámetros a estimar le da a la RNHm una estructura análoga a RNICP (fig. 4.2).

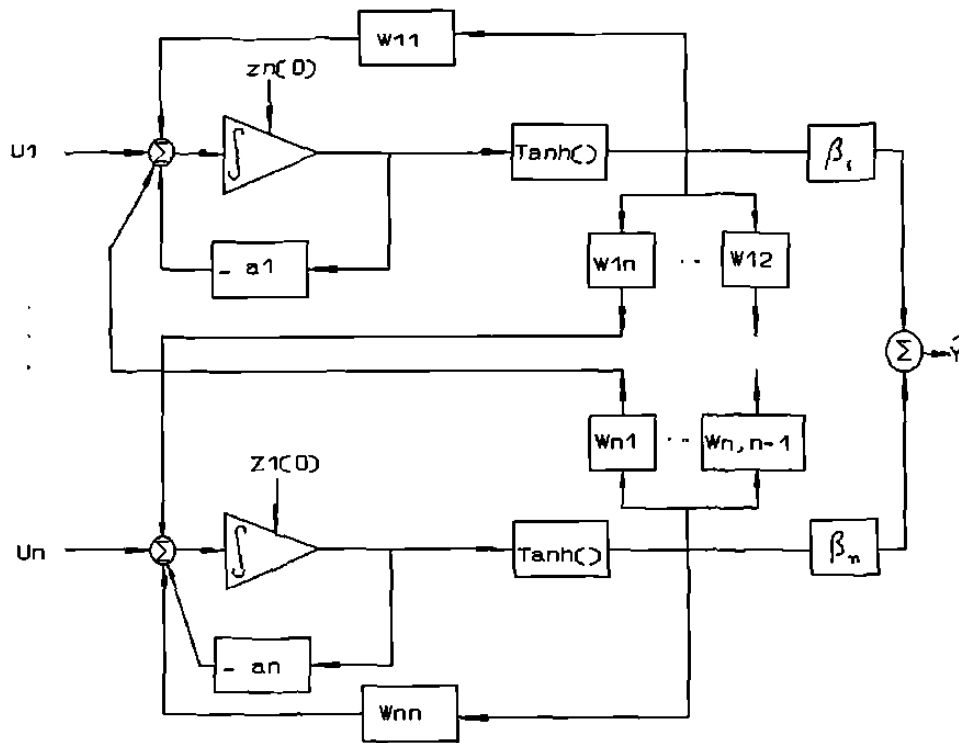


Figura 4.2 Representación de la red neuronal de Hopfield modificada

Def. 4.1. **Operador de Hopfield.**- Establecemos el operador de Hopfield como el operador asociado con la ecuación diferencial (4.1) y lo denotamos por: $O_H: M \subset L^p_{[a,b]} \rightarrow N \subset L^p_{[a,b]}$.

A continuación se presenta el uso de esta estructura para modelado.

4.2.2. Modelado de sistemas como aproximadores de operadores

El problema de modelado consiste en la representación matemática de sistemas. El modelo de un sistema puede ser expresado a través de un operador $p: U \rightarrow Y$, un

CAPITULO 4: MODELADO DE SISTEMAS

primer objetivo consiste en caracterizar la clase P a la cual pertenecen. Dada una clase p y el hecho de que $p \in P$, un segundo objetivo consiste en determinar una clase \hat{p} y un elemento $\hat{p} \in \hat{P}$ tal que \hat{p} aproxime a p en algún sentido deseado [16].

Cuando p representa a sistemas dinámicos con equilibrios estables, los espacios U y Y son subconjuntos de $L_{\mathcal{P}}(0,T]$. El operador queda definido implícitamente por el par de funciones entrada-salida [16].

El problema de modelado se puede resumir como: encontrar un modelo matemático \hat{p} de p tal que

$$\|y - \hat{y}\| = \|Pu - \hat{P}u\| \leq \epsilon \quad \forall y \in Y \quad y \quad u \in U \quad \forall \epsilon > 0$$

donde $Y = Pu$ $\hat{Y} = \hat{P}U$

Este problema es semejante al de aproximar p por \hat{p} , tal que sea minimizada la norma entre la diferencia de las salidas cuando se aplica la misma entrada. Toda ϵ significa que \hat{p} se tendrá que aproximar arbitrariamente bien a p .

4.2.3 Conjetura acerca de la aproximación de operadores utilizando operadores de Hopfield

La conjetura que se discute establece las bases para la "extensión" de los resultados repasados al principio del capítulo 3 a la aproximación de operadores.

Planteando el problema de aproximación en términos de la red de Hopfield modificada queda:

CAPITULO 4: MODELADO DE SISTEMAS

Problema de aproximación de operadores.

Sea P un operador que representa a un sistema $P:U \rightarrow Y$ continuo, acotado, estable, y sea $O_H:U \rightarrow Y$ el operador de Hopfield establecido, el problema de aproximación consiste en determinar los parámetros W, A, B de O_H tal que:

$$\|Pu - O_H u\| < \epsilon$$

para algún ϵ arbitrario

La siguiente conjetura se plantea, acerca de la existencia de solución del problema de aproximación anterior.

Conjetura

Dada una ecuación diferencial con equilibrios estables

$$\dot{X} = f(x,t) \quad x(0) \in R \quad Y = h(x)$$

cuyo operador asociado es P y una RNH_m :

$$\dot{Z} = -AZ + WY + U \quad Z(0) \in R^N$$

$$\hat{Y} = B^T Y$$

y cuyo operador asociado es O_H .

Existen al menos un conjunto de valores N, A, W y B tales que:

CAPITULO 4: MODELADO DE SISTEMAS

$$\|y - \hat{y}\| < \epsilon \text{ ó } \|Pu - O_H \mu\| < \epsilon, \quad u \in U \subset L_{p,b}$$

para todo $\epsilon > 0$.

El problema planteado es semejante al problema 2.1, pero ahora en terminos de operadores y no de funciones. La conjetura habla acerca de la posibilidad de usar la RNHm para la aproximación de una clase de operadores asociados a sistemas dinámicos. La veracidad de la conjetura traería como consecuencia un resultado análogo a los citados en la sección 3.1; Es decir, que la combinación lineal de operadores resulte ser denso y separe puntos de algún espacio de operadores estables, quedando aún por resolver el problema de la adecuada selección de los parámetros.

Enseguida se muestran un ejemplo, el cual si bien es cierto no demuestran nada, si dan evidencia acerca de la posible veracidad de la conjetura. En este ejemplo, los pesos son encontrados utilizando el método de gradiente descendente aprovechando que estos se pueden parametrizar linealmente en la RNHm.

Para el ajuste de los pesos se utilizaron gradientes descendentes, sin embargo, pruebas formales de que estos son adecuados quedan como trabajos futuros.

$$\begin{aligned}\dot{w}_{ij} &= -\gamma_j * e * tgh(Z_i) \\ \dot{\beta}_i &= -\gamma_{\beta_i} * e * tgh(Z_i) \\ \dot{\alpha}_i &= -\gamma_{\alpha_i} * e * Z_i\end{aligned}$$

CAPITULO 4: MODELADO DE SISTEMAS

Ejemplo 4.1. Utilizando el operador de Hopfield, encontraran los valores N , A , W , B que hacen que O_H se aproxime a p en el sentido antes descrito, con:

$$\frac{dx}{dt} = -5*x + \text{sen}(x) + U$$

Para este ejemplo utilicemos un valor de, $N = 5$ y la búsqueda de α_i , W_i y β_i realizada a través de gradientes descendentes.

En este ejemplo se buscara minimizar el error a la salida, pero no se considera ningún ϵ , pues esto requiere de conocer soluciones explícitas de las ecuaciones diferenciales para su cálculo y verificación.

Los resultados se muestran gráficamente en las figuras 4.3 y 4.4.

4.2.4 Acerca de la veracidad de la conjetura

El problema de aproximar operadores no es nuevo. De hecho, fue utilizado para atacar el problema de robustez bajo el enfoque de H_∞ [27]; en comunicaciones se aproximan operadores utilizando sumatorias y filtros con retraso. Más recientemente apareció un trabajo [28], en el cual se aborda el problema de identificación de sistemas de control no-lineales. Para lo que se define un espacio vectorial de operadores V el cual es Hilbert y separable bajo un subespacio de Kernel reproducibile; utilizando operadores de Volterra, se construye solución a este problema de aproximación (fuera de línea).

CAPITULO 4: MODELADO DE SISTEMAS

Es nuestra convicción que la conjetura resultará verdadera. Desde nuestro punto de vista la prueba debe seguir los siguientes pasos:

- a) Los operadores $O_H \in V$
- b) El conjunto de operadores O_H es denso y separa elementos de V .

CAPITULO 4: MODELADO DE SISTEMAS

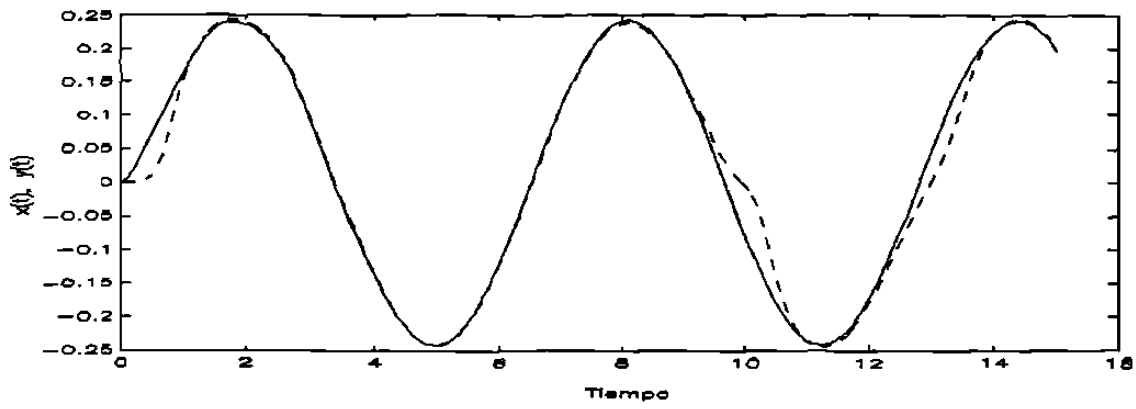


Figura 4.3 Gráfica de la salida del sistema y de la red neuronal.

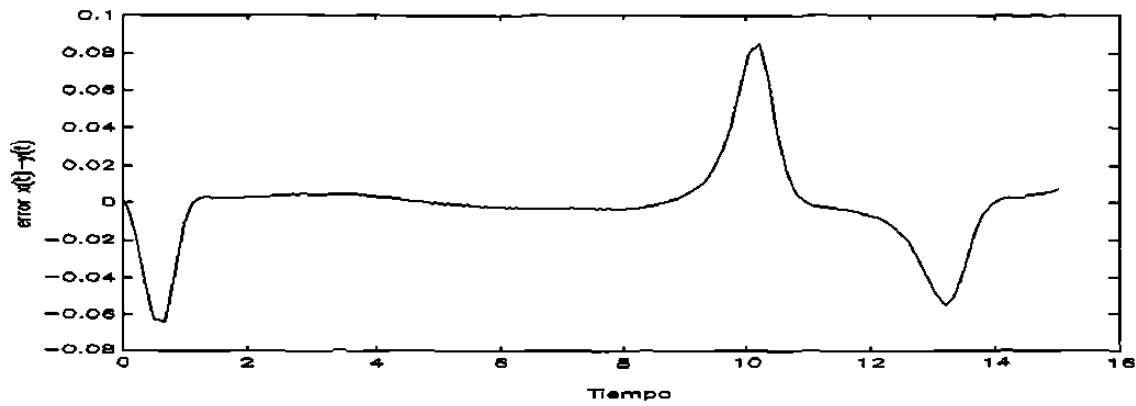


Figura 4.4 Evolución del error en el tiempo.

Capítulo 5.

Conclusiones y trabajos futuros

En este trabajo se ha dado un panorama de la utilización de redes neuronales como aproximadores de funciones y para modelar sistemas dinámicos.

Primero fueron consideradas las redes neuronales estáticas y su característica de funcionar como aproximadores universales; se repasaron los principales resultados existentes en cuanto a la posibilidad de la aproximación; después se resumieron los principales trabajos en lo referente a la construcción de la aproximación, se propuso un nuevo procedimiento, que si bien no es solución completa tiene algunas ventajas por ser un enfoque más general

Hacemos la observación de que el problema de aproximación de funciones mediante redes neuronales estáticas requiere aún de trabajo. Todavía la selección del número de 'neuronas' y los pesos de los argumentos de la función de activación óptimamente son problemas abiertos.

Posteriormente se considero el modelado de sistemas dinámicos; primero repasando algunos intentos hechos con redes neuronales estáticas, sin embargo, estas no son las más adecuadas al no contar con memoria se propuso una idea al respecto: basados en la red neuronal de Hopfield con una modificación a la salida se establece una

CAPITULO 5: CONCLUSIONES Y TRABAJOS FUTUROS

conjetura, la cual es respaldada por simulaciones.

La identificación de sistemas dinámicos requiere de estructuras más poderosas que las redes neuronales estáticas. Aquí el uso de redes neuronales dinámicas parece lo más razonable; sin embargo, hace falta sustentación teórica. Trabajos futuros deberán enfocarse a la obtención de esta, así como al desarrollo de metodologías para llevar a cabo la identificación o la estimación en estos términos (basados en la RNH ó en la RNHm).

Trabajos futuros deben orientarse a la búsqueda de soluciones a los problemas abiertos descritos. Como sugerencia se propone el estudio del principio del mínimo de pontryagin a la búsqueda de los pesos.

Referencias

- [1] P. J. W. Melsa, "*Neural Networks: A conceptual overview*", Tellabs Research Center, TRC-89-08, 1989.

- [2] J. J. Hopfield, "*Neural networks and physical systems with emergent collective computational abilities*" Proc. Nat. Acad. Sc. U.S. Vol. 79, April 1982.

- [3] G. Cybenko, "*Approximation by superposition of a sigmoidal function*", Math. Control, Signal and Systems, 1989.

- [4] K. Funahashi, "*On the approximate realization of continuous mappings by neural networks*", Neural Networks, Vol. 2, 1989.

- [5] K. Hornik, M. Stinchcombe, H. White, "*Multilayer feedforward networks are universal approximators*", Neural Networks, Vol. 2, 1989.

- [6] H. White, M. Stinchcombe, "*Universal approximation using feedforward networks with nonsigmoid hidden layer activation functions*", Proc. IJCNN, Washington, 1989, pp 613-617.

- [7] S. M. Carroll, B. W. Dickinson, "*Construction of neural networks using the Radon transform*", IJCNN, Washington, 1989, pp 607-611.

Referencias

- [8] A. Stubberud, H. Wabaonkar, "*Variational methods for neural networks*", LATINCON IEEE, Monterrey, México, Octubre de 1990.
- [9] E. Alcorta, E. N. Sánchez, "*Aproximación de funciones mediante redes neuronales*", X Congreso del AMCA, CINVESTAV, MEXICO, Nov. 1991.
- [10] E. Alcorta, E. N. Sánchez, "*The best function approximation by Neural Networks*", Sometido a IJCNN 92, Baltimore, junio 1992.
- [11] V. Vemuri, "*Artificial Neural Networks: theoretical concepts*", IEEE Computer society press, 1988.
- [12] D. G. Luenberger, "*Optimization by vector Space Methods*", New York, Wiley, 1969.
- [13] E. Kreyzig, "*Introductory Functional Analysis*", Wiley, 1978.
- [14] S. Sastry, M. Bodson, "*Adaptive control, stability, convergence and robustness*", Printice Hall, 1989.
- [15] Girosi, Poggio, "*Networks for approximation and learning*", Proceedings IEEE, Vol. 78, No. 9, Set. 1990.

- [16] K. S. Narendra, K. Parthasarathy, "*Identification and control of dynamical systems using neural networks*", IEEE Trans. on Neural Networks, March 1991.
- [17] R. Carelli, D. Patiño, B. Kuchen, "*Análisis de sistemas de control con redes neuronales para manipuladores robóticos*", V Congreso Latinoamericano de Control Automático, La Habana, Cuba, Febrero 1992.
- [18] H. Wabgaonkar, A. Stubberud, "*Approximation and estimation techniques for neural networks*", 29th CDC, Hawaii, Dec. 1991.
- [19] A. Stubberud, H. Wabgaonkar, "*Application of neural networks to the system identification problem*", CDC, Hawaii, Dec. 1991.
- [20] K. J. Astrom, "*Adaptive Control*", Prentice-Hall, 1989.
- [21] G. C. Goodwin, K. Sang Sin, "*Adaptive filtering, prediction and control*", Prentice-Hall, 1984.
- [22] H. W. Sorenson (Editor), "*Kalman Filtering: Theory and Application*", IEEE Press, 1985.

- [23] Arthur Gelb, "*Applied optimal control*", MIT press, 1986
- [24] P. R. Kumar, "*Neural Networks*", Escuela AMCA 1990, Cuernavaca, Morelos, México, Octubre 1990.
- [25] K. S. Narendra, A. U. Levin, "*Regulation of nonlinear dynamical systems using multiple neural networks*", ACC, Boston Jun., 1991.
- [26] E. Alcorta, E. N. Sanchez, "*Non-linear identification via neural networks*", Sometido al ASCASP-IFAC, Grenoble, France, Jun. 1992.
- [27] B. A. Francis, "*A course in H_{∞} Control Theory*", Lecture notes in control and information sciences, No. 88, Springer Verlag Berlin, 1987.
- [28] J. P. de Figueiredo, G. Chen, "*A new operator-theoretic approach to optimal identification of nonlinear control systems*", ACC, Boston, Jun., 1991.

