

Inteligencia Artificial Neuronal

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura:	Inteligencia Artificial Neuronal
Clave de la asignatura:	ICC-1605
(Créditos) SATCA1	2-2-4
Carrera:	Ingeniería Electrónica

2.- PRESENTACIÓN

Caracterización de la asignatura

La asignatura contribuye en el desarrollo de habilidades y capacidades en el planteamiento y solución de problemas, que se relacionen con el diseño de sistemas de control inteligente en el área de las redes neuronales. Así mismo, facilita al estudiante el uso de software de computadora para la comprensión del comportamiento de los sistemas neuronales a través de la simulación, para plantear y predecir el funcionamiento óptimo bajo distintas condiciones de operación.

La estructura de la asignatura tiende a cubrir los aspectos del conocimiento básico en redes neuronales, en donde se explica los diferentes componentes de un sistema neuronal, pasando por el análisis y el diseño de sistemas neuronales hasta llegar a las aplicaciones.

La asignatura se relaciona de manera directa con materias como: Automatización, Control, Instrumentación e Inteligencia Artificial I; específicamente en los temas de control PID clásico, Controladores Lógicos Programables (PLC), modelado de sistemas, medición de variables en procesos. Por último las competencias específicas que se relacionan son: interpreta conceptos básicos de control, analiza la estabilidad de un sistema, diseña controladores clásicos, representa sistemas dinámicos en variables de estado, analiza controlabilidad de sistemas aplicando retroalimentación de estados.

Intención Didáctica

El programa de la asignatura se divide en cuatro unidades, presentando contenidos conceptuales de la asignatura a partir de la primera unidad; en donde

se plantea una introducción de los conceptos para su ubicación dentro de los marcos conceptuales de la teoría de control con redes neuronales.

Se propone la práctica de los conocimientos y el análisis mediante una actividad de revisión de artículos de investigación para encontrar relación de los conceptos estudiados con las investigaciones recientes además de generar el hábito en los estudiantes a revisar información a partir de formatos estándares de reporte de resultados de las nuevas investigaciones.

En concordancia con las nuevas propuestas por competencias se propone la formalización de los conceptos a partir de experiencias concretas; se busca que el estudiante adquiera el concepto en forma concreta y sea a través de la observación, la reflexión y la discusión que se dé la formalización; la solución de casos prácticos se hará después de este proceso.

El enfoque sugerido para la materia requiere que las actividades prácticas promuevan el desarrollo de habilidades para la experimentación, tales como: identificación, manejo y control de variables y datos relevantes; planteamiento de hipótesis; trabajo en equipo; asimismo, propicien procesos intelectuales como inducción-deducción y análisis-síntesis con la intención de generar una actividad intelectual compleja; por esta razón varias de las actividades prácticas se han descrito como actividades previas al tratamiento teórico de los temas, de manera que no sean una mera corroboración de lo visto previamente en clase, sino una oportunidad para conceptualizar a partir de lo observado. En las actividades prácticas sugeridas, es conveniente que el profesor busque sólo guiar a sus alumnos para que ellos hagan la elección de las variables a controlar y registrar. Para que aprendan a planificar, que no planifique el profesor todo por ellos, sino involucrarlos en el proceso de planeación.

Se sigue además las actividades de investigación documental, que permitan al estudiante hacer una revisión de material de libros o de la red en donde se reporten resultados teóricos o experimentales sobre aplicaciones de los principios estudiados, de manera que el estudiante encuentre que los conocimientos tienen una relación de impacto en las necesidades y realidades sociales.

Además de proponer una dinámica de puesta en común sobre los temas revisados en clase y su relación con los usos domésticos de la teoría, en donde los estudiantes detecten la aplicación de los principios teóricos estudiados, ya que una forma de reforzar el conocimiento, es mediante la relación de lo estudiado con la experiencia cotidiana.

3. PARTICIPANTES EN EL DISEÑO Y SEGUIMIENTO CURRICULAR DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Instituto Tecnológico de Cd. Guzmán Jal.	Instituto Tecnológico de Cd. Guzmán	

4. COMPETENCIAS A DESARROLLAR

Competencia general de la asignatura
<ul style="list-style-type: none"> ➤ Diseñar e implementar sistemas de control neuronal en aplicaciones de ingeniería de campo, laboratorio y simulación.
Competencias específicas
<ul style="list-style-type: none"> ➤ Identifica los usos de los sistemas neuronales en la solución de problemas de la realidad social para su implementación efectiva. ➤ Adquiere las habilidades en el uso de las herramientas para el diseño neuronal para su uso en laboratorio y campo. ➤ Adquiere las habilidades en el uso de las herramientas para el modelado e identificación con redes neuronales para entender sus principios de funcionamiento y aplicación a problemáticas de ingeniería. ➤ Adquiere las habilidades en el uso de las herramientas para el diseño de controles neuronales en el entorno de las aplicaciones de ingeniería de campo, laboratorio y simulaciones.
Competencias genéricas
<ul style="list-style-type: none"> ➤ Capacidad de análisis y síntesis ➤ Capacidad de organizar y planificar ➤ Conocimientos básicos de la carrera ➤ Comunicación oral y escrita ➤ Habilidad en manejo de la computadora ➤ Habilidad para buscar y analizar información proveniente de fuentes diversas ➤ Solución de problemas ➤ Toma de decisiones.

5. COMPETENCIAS PREVIAS DE OTRAS ASIGNATURAS

Competencias previas
<ul style="list-style-type: none"> ➤ Conceptos básicos de control.

- Estabilidad de sistemas.
- Interpretar gráficas de respuesta en frecuencia y el tiempo para diseñar compensadores.
- Diseño de controladores clásicos.
- Representar sistemas dinámicos en variables de estado analizando su controlabilidad aplicando retroalimentación de estados.

6. TEMARIO

Temas		Subtemas	Literatura
NO.	Nombre		
1	Fundamentos de Redes Neuronales	1.1 Historia y primeros resultados en redes neuronales. 1.2 Aplicaciones de las redes neuronales como sistemas de inteligencia artificial. 1.3 Conceptos y fundamentos de las redes neuronales.	[5],[6],[7]
2	Implementación de redes neuronales como aproximadores de funciones y mapeos	2.1 Algoritmos de entrenamiento y "retropropagación". 2.2 Separación de datos mediante un "perceptron". 2.3 Aproximación de funciones en R2 (curvas) y R3 (superficies). 2.4 Redes neuronales multicapa.	[1],[2],[3],[4],[5],[6]
3	Redes neuronales retroalimentadas para el modelado de sistemas dinámicos	3.1 Modelado de sistemas dinámicos. 3.2 Modelado de un sistema eléctrico. 3.3 Modelado de un sistema mecánico. 3.4 Modelado de un sistema electromecánico.	[3],[4],[5],[6],[7]
4	Control Neuronal	4.1 Control Neuronal NARMA L2 4.2 Control Neuronal NARMA L2 para un sistema eléctrico. 4.3 Control Neuronal NARMA L2 para un sistema mecánico. 4.4 Control Neuronal NARMA L2 para un sistema electromecánico.	[3],[4],[5],[6],[7]

7. ACTIVIDADES DE APRENDIZAJE

Competencia específica y genéricas	
Identifica los usos de los sistemas neuronales en la solución de problemas de la realidad social.	
Tema	Actividades de aprendizaje
Fundamentos de Redes Neuronales	<ul style="list-style-type: none"> ➤ Revisión de formatos estándar de reporte de nuevos resultados científicos. ➤ Realización de un ensayo sobre aplicaciones de los sistemas neuronales. ➤ Análisis de videos sobre aplicaciones de sistemas neuronales. ➤ Análisis de simulaciones de sistemas de sistemas neuronales.
Competencia específica y genéricas	
Adquiere las habilidades en el uso de las herramientas para el diseño neuronal.	
Tema	Actividades de aprendizaje
Implementación de redes neuronales como aproximadores de funciones y mapeos	<ul style="list-style-type: none"> ➤ Revisión de la sintaxis de los comandos para el diseño neuronal. ➤ Revisión de los parámetros y características de los bloque funcionales de Matlab para el diseño neuronal. ➤ Revisión de las interfaces de usuario para el diseño neuronal. ➤ Análisis de videos sobre aplicaciones de sistemas neuronales. ➤ Análisis de simulaciones de sistemas de sistemas neuronales.
Competencia específica y genéricas	
Adquiere las habilidades en el uso de las herramientas para el modelado e identificación con redes neuronales.	
Tema	Actividades de aprendizaje
Redes neuronales retroalimentadas para el modelado de sistemas	<ul style="list-style-type: none"> ➤ Revisión de las interfaces de usuario para la identificación con

dinámicos	<p>redes neuronales.</p> <ul style="list-style-type: none"> ➤ Revisión de los parámetros y características de los bloques funcionales de Matlab para el diseño con redes neuronales. ➤ Análisis de videos sobre aplicaciones de sistemas neuronales. ➤ Análisis de simulaciones de sistemas de sistemas neuronales. ➤ Análisis de manuales de especificación técnica de sistemas que incluyen sistemas neuronales.
Competencia específica y genéricas	
Adquiere las habilidades en el uso de las herramientas para el diseño de controles neuronales.	
Tema	Actividades de aprendizaje
Control Neuronal.	<ul style="list-style-type: none"> ➤ Revisión de las interfaces de usuario para la solución de problemas de control con redes neuronales. ➤ Análisis de videos sobre aplicaciones de sistemas neuronales. ➤ Análisis de simulaciones de sistemas de sistemas neuronales.

8. PRÁCTICAS.

<ul style="list-style-type: none"> ➤ Clasificación de datos en R2 y R3 con una red neuronal tipo perceptron. ➤ Ajuste de planos y superficies con una red neuronal multicapa: el caso de aprendizaje de una tabla de multiplicar. ➤ Ajuste de funciones en R3 con una red neuronal multicapa. el caso de la función cono. ➤ Diseño de un control PID clásico para un motor de CD. ➤ Identificación de sistemas dinámicos con redes neuronales: el caso de un

- levitador magnético y un sistema masa-resorte-amortiguador.
- Control neuronal NARMA-L2 de un levitador magnético.

9. PROYECTO INTEGRADOR

- Desarrollar un control neuronal para un proceso físico de temperatura.

10. EVALUACIÓN POR COMPETENCIAS

- Evaluación diagnóstica:
 - Mediante un ejercicio de exposición, informe escrito o examen oral determinar el grado de dominio de las competencias previas.
- Evaluación formativa:
 - Considerar la participación en las actividades programadas en la materia
 - Tareas de investigación y ejercicios.
 - Participación / exposiciones en clase.
 - Elaboración de artículos.
 - Prácticas en clase o extra clase.
 - Exámenes escritos y frente a la computadora.
 - Elaboración de diagramas a bloques utilizando Simulink.
 - Elaboración de informes sobre actividades de aprendizaje.
 - Visitas a la industria y centros de investigación
- Evaluación sumativa:
 - Prácticas
 - Exámenes
 - Proyecto Integrador
 - Elaboración de trabajos fuera de clase (tareas, investigaciones, lecturas, etc.)

11. FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.

7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.
1. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

RELACION PRÁCTICAS Y COMPETENCIAS A DESARROLLAR

- 1.- Clasificación de datos en R2 y R3 con una red neuronal tipo perceptron.
Competencia específica a desarrollar: Entrena una red neuronal para realizar una clasificación de datos en R2 y R3 para eficientar su desempeño.
- 2.- Ajuste de planos y superficies con una red neuronal multicapa: el caso de aprendizaje de una tabla de multiplicar.
Competencia específica a desarrollar: Entrena una red neuronal para aprender una tabla de multiplicar y lograr la mayor eficiencia para permitir que el cálculo sea lo más exacto posible.
- 3.- Ajuste de funciones en R3 con una red neuronal multicapa: El caso de la función cono.
Competencia específica a desarrollar: Entrena una red neuronal para realizar un ajuste de curva en R3 y obtener el mejor índice de desempeño.
- 4- Diseño de un control PID clásico para un motor de CD.
Competencia específica a desarrollar: Diseña un control PID clásico en MATLAB Simulink, para controlar un motor de corriente directa y mejorar sus índices de desempeño.
- 5.- Identificación de sistemas dinámicos con redes neuronales: El caso de un levitador magnético y un sistema masa-resorte-amortiguador.
Competencia específica a desarrollar: Identifica la dinámica de sistemas lineales mediante la interfaz de usuario de control neuronal NARMA-L2 de Matlab Simulink, para obtener el mejor índice de eficiencia.
- 6.- Control neuronal NARMA-L2 de un levitador magnético.
Competencia específica a desarrollar: Diseña un control neuronal NARMA-L2 de Matlab Simulink, para controlar un levitador magnético y mejorar sus índices de desempeño.

PROGRAMA DEL CURSO INTELIGENCIA ARTIFICIAL NEURONAL

Temario

Fundamentos de Redes Neuronales

- 1.1 Historia y primeros resultados en redes neuronales.
- 1.2 Aplicaciones de las redes neuronales como sistemas de inteligencia artificial.
- 1.3 Conceptos y fundamentos de las redes neuronales.

Implementación de redes neuronales como aproximadores de funciones y mapeos

- 2.1 Algoritmos de entrenamiento y “retropropagación”.
- 2.2 Separación de datos mediante un “perceptron”.
- 2.3 Aproximación de funciones en R2 (curvas) y R3 (superficies).
- 2.4 Redes neuronales multicapa.

Redes neuronales retroalimentadas para el modelado de sistemas dinámicos

- 3.1 Modelado de sistemas dinámicos.
- 3.2 Modelado de un sistema eléctrico.
- 3.3 Modelado de un sistema mecánico.
- 3.4 Modelado de un sistema electromecánico.

Control Neuronal

- 4.1 Control Neuronal NARMA L2
- 4.2 Control Neuronal NARMA L2 para un sistema eléctrico.
- 4.3 Control Neuronal NARMA L2 para un sistema mecánico.
- 4.4 Control Neuronal NARMA L2 para un sistema electromecánico.

PRÁCTICA 1:

Clasificación de Datos en R2 Y R3 con una Red Neuronal Tipo Perceptron.

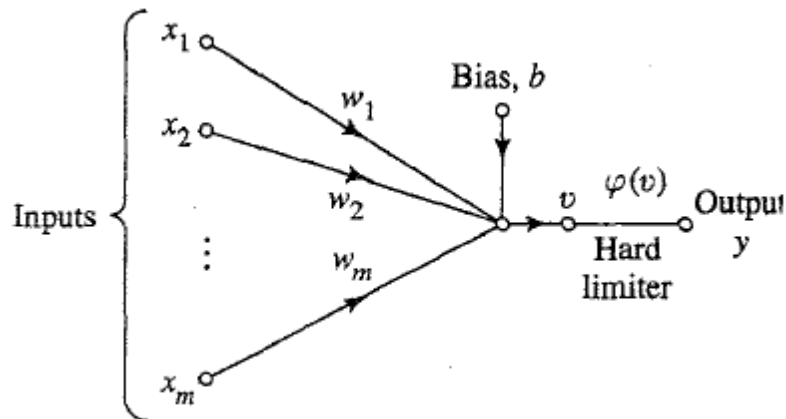
1.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Entrena una red neuronal para realizar una clasificación de datos en R2 y R3 para eficientar su desempeño.

1.2 INTRODUCCIÓN

PERCEPTRON

El Perceptron es la forma más simple de una red neuronal utilizada para la clasificación de patrones linealmente separables (patrones que se localizan en los lados opuestos de un hiperplano, también conocido como dicotomía). Básicamente consiste en una sola neurona con pesos ajustables y un umbral.

El algoritmo utilizado para ajustar los parámetros libres de esa red surgió como un procedimiento de aprendizaje de Rosenblatt (Rosenblatt, 1958) para su modelo de cerebro, el perceptron. La prueba de convergencia de dicho algoritmo es conocida como el teorema de convergencia del Perceptrón.



1.3 MATERIAL Y EQUIPO NECESARIO

1.3.1 Software

Matlab 6 o superior

1.3.2 Equipo

Computadora personal

1.4 METODOLOGÍA

Clasificación de datos en R2

1.4.1. En el editor de Matlab, escriba un programa para realizar la clasificación de datos en R2 con una red neuronal tipo Perceptron.

```
%Perceptron R2
clear all
P= [ +1 +3 -.5 -2
     +1 +1 +1 +1];
T= [1 1 0 0];
plotpv(P,T);
net= newp([-1 1;-1 1],1);
plotpv(P,T);
plotpc(net.IW{1},net.b{1});
net.trainparam.show=20;
net.trainparam.epochs=30;
net.trainparam.goal=0.01;
net=train(net,P,T);
plotpc(net.IW{1},net.b{1});
net.IW{1}
net.b{1}
```

1.4.2. Represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo Perceptron realizar la clasificación.

1.4.3. Modifique el programa agregando más datos a clasificar, represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo Perceptron realizar la clasificación.

1.4.4. Modifique el programa para llevar a la red neuronal a realizar la clasificación en la mayor cantidad de épocas posibles.

1.4.5. Represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo perceptron realizar la clasificación.

Clasificación de datos en R3

1.4.6. En el editor de Matlab, escriba un programa para realizar la clasificación de datos en R3 con una red neuronal tipo perceptron.

```
%Perceptron R3
clear all
NA1=[0 0 0 0 1 1 1 1 .5 1 1];
B1=[1 1 0 0 1 1 0 0 .5 1 1];
C1=[0 1 0 1 0 1 0 1 .5 1 1];
t=[ 0 0 0 1 0 0 0 0 0 1 0];
```

```

T=[NA1;B1;C1];
web=newp([0 1;0 1;0 1],1);
web.trainParam.epochs=10;
web.trainParam.goal=0;
web=train(web,T,t);
pesosf=web.IW{1,1}
umbral=web.b{1}
%pause;
plotpv(T,t)
hold on
%pause;
plotpc(lasi,umbral)
view([2 4 10])
grid
punto1=[1;1;1]
lasificación=sim(web,punto1)

```

1.4.7. Represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo perceptron realizar la clasificación.

1.4.8. Modifique el programa agregando más datos a clasificar, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo perceptron realizar la clasificación.

1.4.9. Modifique el programa agregando más datos a clasificar, represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal tipo Perceptron realizar la clasificación.

1.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que en algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

1.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo, cada vez que realice una modificación importante que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

1.7 CUESTIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones cree usted que es posible usar este tipo de estructura de red neuronal?

¿Qué otro tipo de técnicas es posible aplicar para obtener resultados similares?

¿Qué tipo de tecnología utilizaría usted para integrar una solución a un problema de campo con este tipo de técnicas?

1.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.
7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.
8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

1.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

1.10 REPORTE DEL ALUMNO

1.10.1 Desarrollar un reporte manuscrito con los siguientes puntos

1.10.2 La portada debe contener los siguientes datos:

Nombre de la escuela

Nombre de la materia

Nombre del profesor
Nombre del estudiante
Número de control
Número de la práctica
Nombre de la práctica
Lugar y fecha

1.10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: título, autores, editorial, año, numero de las paginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año

PRÁCTICA 2:

Ajuste de Planos y Superficies con una Red Neuronal Multicapa: El Caso de Aprendizaje de una Tabla de Multiplicar.

2.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Entrena una red neuronal para aprender una tabla de multiplicar y lograr la mayor eficiencia para permitir que el cálculo sea lo más exacto posible.

2.2 INTRODUCCION

Las redes neuronales artificiales (RNA) se definen como sistemas de mapeos no lineales cuya estructura se basa en principios observados en los sistemas nerviosos de humanos y animales. Constan de un número grande de procesadores simples ligados por conexiones con pesos. Las unidades de procesamiento se denominan neuronas. Cada unidad recibe entradas de otros nodos y genera una salida simple escalar que dependen de la información local disponible, guardada internamente o que llega a través de las conexiones con pesos.

Una red neuronal se caracteriza por los siguientes elementos:

1. Un conjunto de unidades de procesamiento o neuronas.
2. Un estado de activación para cada unidad, equivalente a la salida de la unidad.
3. Conexiones entre las unidades, generalmente definidas por un peso que determina el efecto de una señal de entrada en la unidad.
4. Una regla de propagación, que determina la entrada efectiva de una unidad a partir de las entradas externas.
5. Una función de activación que actualiza el nuevo nivel de activación basándose en la entrada efectiva y la activación anterior.
6. Una entrada externa que corresponde a un término determinado como bias para cada unidad.
7. Un método para reunir información, correspondiente a la regla del aprendizaje.
8. Un ambiente en el que el sistema va a operar, con señales de entrada e incluso señales de error.

2.3 MATERIAL Y EQUIPO NECESARIO

2.3.1 Software

Matlab 6 o superior

2.3.2 Equipo

Computadora personal

2.4 METODOLOGÍA

Tabla de multiplicar del número 2

2.4.1. En el editor de Matlab, escriba un programa para que una red neuronal tipo feed forward aprenda la tabla de multiplicar del 2.

```
%Entrenamiento de una neurona para aprender la Tabla de Multiplicar del 2.  
%Tabla de multiplicar red backpropagation  
clear all;  
P=[2 2 2 2 2 2 2 2 2 2;  
 1 2 3 4 5 6 7 8 9 10];  
D=[2 4 6 8 10 12 14 16 18 20];  
%Red neuronal y una fail forward  
net=newff([-3 3;-3 3],[10 10 1], {'tansig', 'tansig', 'purelin'},'trainlm');  
net.trainParam.show=20;  
net.trainParam.epochs=1000;  
net.trainParam.goal=1e-9;  
net=train(net, P, D);  
salida=sim(net,P)
```

2.4.2. Anote los resultados y represente gráficamente la ejecución del programa, señale los pesos utilizados y el número de épocas que le tomo a la red neuronal de retropropagación realizar la clasificación.

2.4.3. Modifique el programa cambiando el número de épocas para provocar que el entrenamiento de la red neuronal falle.

```
net.trainParam.epochs=__;
```

2.4.4. Regrese a las 1000 épocas y modifique el error permitido a $1e-1$ y verifique que pasa.

```
net.trainParam.goal=1e-1;
```

2.4.5. Modificar el programa con épocas = 1000, error en $1e-9$, elimine un tansig y modifique el número de capas ocultas a [2 1].

```
net=newff([-3 3;-3 3],[2 1], {'tansig', 'purelin'},'trainlm');  
net.trainParam.epochs=1000;  
net.trainParam.goal=1e-9;
```

2.4.6. Modifique el programa variando el error y el número de épocas para tener un mejor rendimiento, realice varias pruebas (mínimo 2).

2.4.7. En la ventana de comandos de MatLab ingresas a la interfaz de usuario utilizando el comando “nntool”.Para llevar a cabo el entrenamiento de una red neuronal para que aprenda la tabla de multiplicar del número 2.

2.4.8. En la interfaz de usuario diseñe una red neural con 3 capas

2.4.9. Ingrese los datos a la red neuronal para poder llevar a cabo el entrenamiento para que pueda aprender la tabla de multiplicar del 2.

2.4.10. Ejecute el entrenamiento y reporte los resultados obtenidos

2.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

2.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo cada vez que realice una modificación importante, que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

2.7 CUESTIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones cree usted que es posible usar este tipo de estructura de red neuronal?

¿Qué otro tipo de técnicas es posible aplicar para obtener resultados similares?

¿Qué tipo de tecnología utilizaría usted para integrar una solución a un problema de campo con este tipo de técnicas?

2.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.

4. Dorf, R. (2010). Modern Control Engineering. 12ª Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.

7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.

8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

2.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

2.10 REPORTE DEL ALUMNO

2.10.1 Desarrollar un reporte manuscrito con los siguientes puntos

2.10.2 La portada debe contener los siguientes datos:

- Nombre de la escuela
- Nombre de la materia
- Nombre del profesor
- Nombre del estudiante
- Número de control
- Número de la práctica
- Nombre de la práctica
- Lugar y fecha

2.10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: título, autores, editorial, año, número de las páginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año

PRÁCTICA 3:

Ajuste de Funciones en R3 con una Red Neuronal Multicapa: El Caso de la Función Cono.

3.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Entrena una red neuronal para realizar un ajuste de curva en R3 y obtener el mejor índice de desempeño.

3.2 INTRODUCCION

RED NEURONAL MULTICAPA

Las redes neuronales tipo perceptrón multicapa (MLP, por su nombre en inglés Multilayer perceptron), han sido aplicadas satisfactoriamente para resolver muy diversos y difíciles problemas por medio del algoritmo conocido como retropropagación; este algoritmo consta de dos etapas:

1. Etapa hacia adelante

Se fijan los parámetros de la red y se presenta una señal de entrada a la red, que se propaga hacia adelante para producir la salida.

2. Etapa hacia atrás

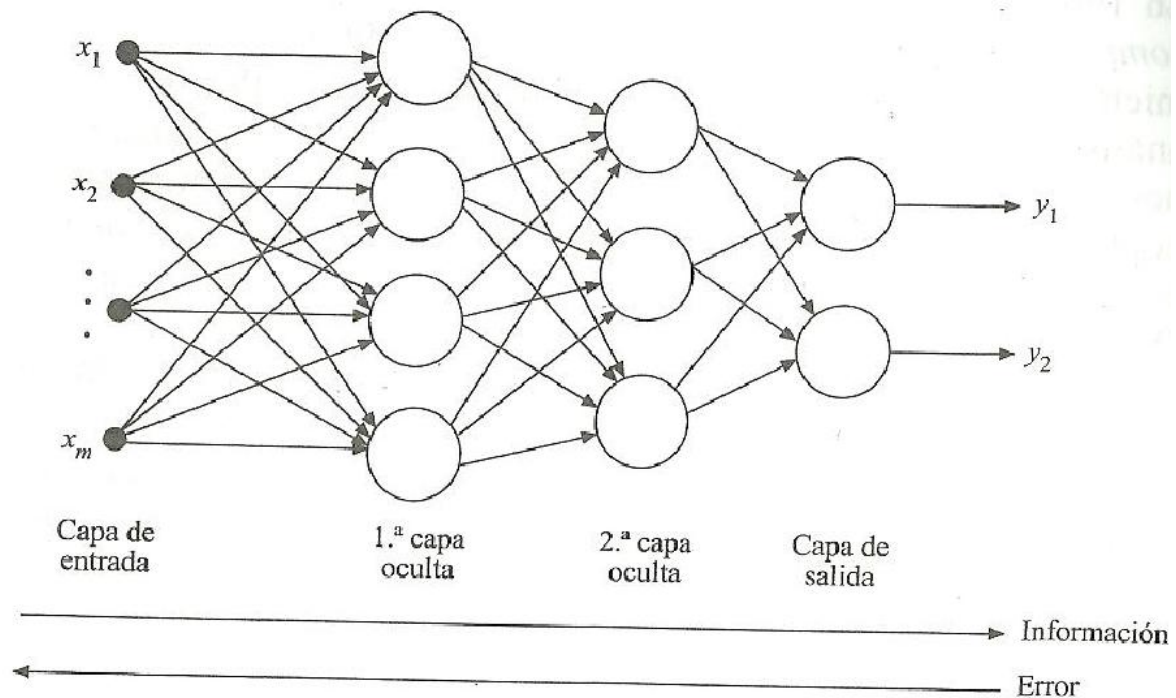
El error entra la salida deseada y la red se propaga hacia atrás. Los parámetros de la red se modifican para minimizar el cuadrado de dicho error.

El MLP tiene tres características distintivas:

1. El modelo de cada neurona de la red incluye una función de activación no lineal. Lo importante aquí es que la linealidad es suave (en cualquier punto existen todas sus derivadas).
2. La red contiene una o más capas ocultas que no son parte de las entradas o las salidas de la red. Estas neuronas ocultas permiten que la red aprenda tareas complejas por la extracción progresiva de las características principales de los patrones de entrada.

3. La red presenta altos grados de conectividad, determinados por las sinapsis de la propia red.

La combinación de estas características junto con la habilidad de aprender de la experiencia a través del entrenamiento del MLP dan como resultado un gran potencial de computación.



3.3 MATERIAL Y EQUIPO NECESARIO

3.3.1 Software

Matlab 6 o superior

3.3.2 Equipo

Computadora personal

3.4 METODOLOGÍA

El caso de la función Cono

3.4.1. En el editor de Matlab, escriba un programa para que una red neuronal multicapa realice la proyección en R3 de un cono.

```
clear all;  
xc=5;  
yc=5;  
h=6;  
r=4;
```

```

[x,y]=meshgrid(1:0.5:9,1:0.5:9);
z=h-(h/r)*sqrt((x-xc).^2+(y-yc).^2);
[r,c]=size(z)
for i=1:r
    for j=1:r
        if z(i,j)<0
            z(i,j)=0;
        end
    end
end
surf(x,y,z);
axis([0 10 0 10 -5 10]);
P=[];D=[];
for i=1:r
    for j=1:r
        P=[P, [x(i,j);y(i,j)]];
        D=[D, z(i,j)];
    end
end
net=newff([-1 10;-1 10],[4 10 1],{'tansig','tansig','purelin'},'trainlm');
net.trainParam.show=20;
net.trainParam.epochs=1000;
net.trainParam.goal=1e-6;
net=train(net, P, D);
salida=sim(net,P);
aproximacion=zeros(r,r);
k=1;
for i=1:r
    for j=1:r
        aproximacion(i,j)=salida(k);
        k=k+1;
    end
end
figure(2);
surf(x,y,aproximacion);
axis([0 10 0 10 -5 10]);

```

3.4.2 Corra el programa y anote las columnas y filas que este arroja. Tome impresión de la imagen del cono generada.

3.4.3 Modifique el programa cambiando el error a -12 y la primera capa de épocas a 10. (Estos datos se modifican cambiando los siguientes renglones del programa original).

```

net=newff([-1 10;-1 10],[10 10 1],{'tansig','tansig','purelin'},'trainlm');
net.trainParam.show=20;
net.trainParam.epochs=1000;

```

```
net.trainParam.goal=1e-12;
```

3.4.4 Anote el número de renglones y filas resultantes, así como anexe la imagen generada.

3.4.5 Modifique el número de apocas a 5, dejando todos los demás valores iguales.

```
net=newff([-1 10;-1 10],[10 10 1],{'tansig','tansig','purelin'},'trainlm');  
net.trainParam.show=20;  
net.trainParam.epochs=5;  
net.trainParam.goal=1e-12;
```

3.4.6 Anote el número de renglones y filas resultantes, así como anexe la imagen generada.

3.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

3.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo cada vez que realice una modificación importante, que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

3.7 CUESTRIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones cree usted que es posible usar este tipo de estructura de red neuronal?

¿Qué otro tipo de técnicas es posible aplicar para obtener resultados similares?

¿Qué tipo de tecnología utilizaría usted para integrar una solución a un problema de campo con este tipo de técnicas?

3.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.

7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.
8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

3.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

3.10 REPORTE DEL ALUMNO

3. 10.1 Desarrollar un reporte manuscrito con los siguientes puntos

3. 10.2 La portada debe contener los siguientes datos:

Nombre de la escuela
Nombre de la materia
Nombre del profesor
Nombre del estudiante
Número de control
Número de la práctica
Nombre de la práctica
Lugar y fecha

3. 10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: titulo, autores, editorial, año, numero de las paginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año

PRÁCTICA 4:

Diseño de un Control PID Clásico para un Motor de CD.

4.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Diseña un control PID clásico en MATLAB Simulink, para controlar un motor de corriente directa y mejorar sus índices de desempeño.

4.2 INTRODUCCIÓN

Controlador PID

Es interesante señalar que más de la mitad de los controladores industriales que se usan hoy en día utilizan esquemas de control PID o PID modificado. Los controladores PID analógicos son, principalmente, de tipo hidráulico, neumático, electrónico, eléctrico o sus combinaciones. En la actualidad, muchos de éstos se transforman en formas digitales mediante el uso de microprocesadores.

Debido a que casi todos los controladores PID se ajustan en el sitio, en la literatura se han propuesto muchos tipos diferentes de reglas de sintonización, que permiten llevar a cabo una sintonización delicada y fina de los controladores PID en el sitio. Asimismo, se han desarrollado métodos automáticos de sintonización y algunos de los controladores PID poseen capacidad de sintonización automática

en línea. Actualmente se usan en la industria formas modificadas del control PID, tales como el control I-PD y el control PID con dos grados de libertad.

Es posible obtener muchos métodos prácticos para una conmutación sin choque (desde la operación manual hasta la operación automática) y una programación del aumento.

La utilidad de los controles PID estriba en que se aplican en forma casi general a la mayoría de los sistemas de control. En el campo de los sistemas para control de procesos, es un hecho bien conocido que los esquemas de control PID básicos y modificados han demostrado su utilidad para aportar un control satisfactorio, aunque tal vez no aporten un control óptimo en muchas situaciones específicas.

4.3 MATERIAL Y EQUIPO NECESARIO

4.3.1 Software

Matlab 6 o superior

4.3.2 Equipo

Computadora personal

4.4 METODOLOGÍA

Controlador PID

4.4.1. En el MARTLAB, Simulink, diseñe un control clásico para un motor de corriente directa (C.D.), basándose en la figura 4.1.

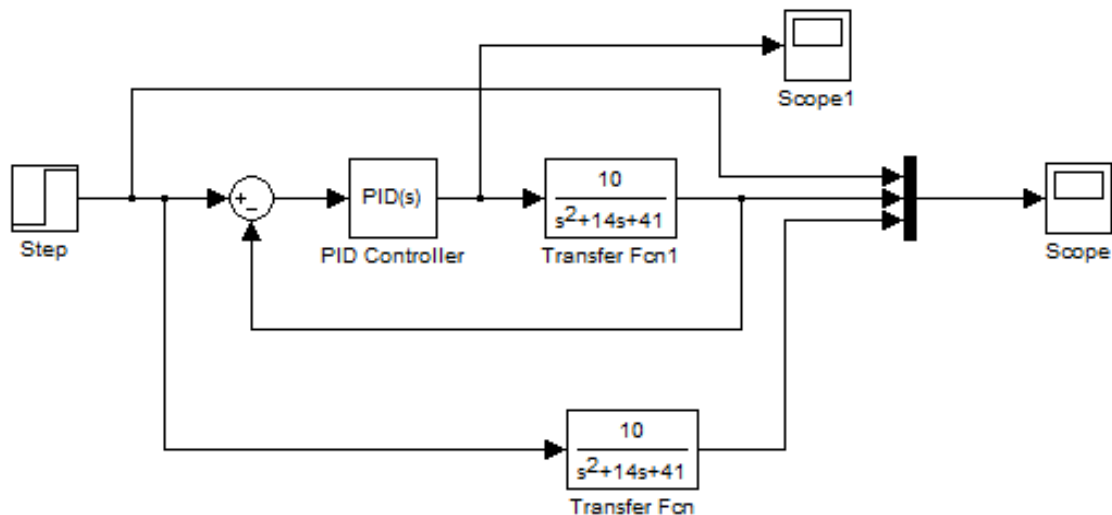


Fig. 4.1

4.4.2. Ajuste las constantes del controlador a $P=7.6$; $I=32$; $D=0.08$, así como un numerador de 10 y coeficientes en denominador de 1, 14 y 41. Figura 4.2 y 4.3

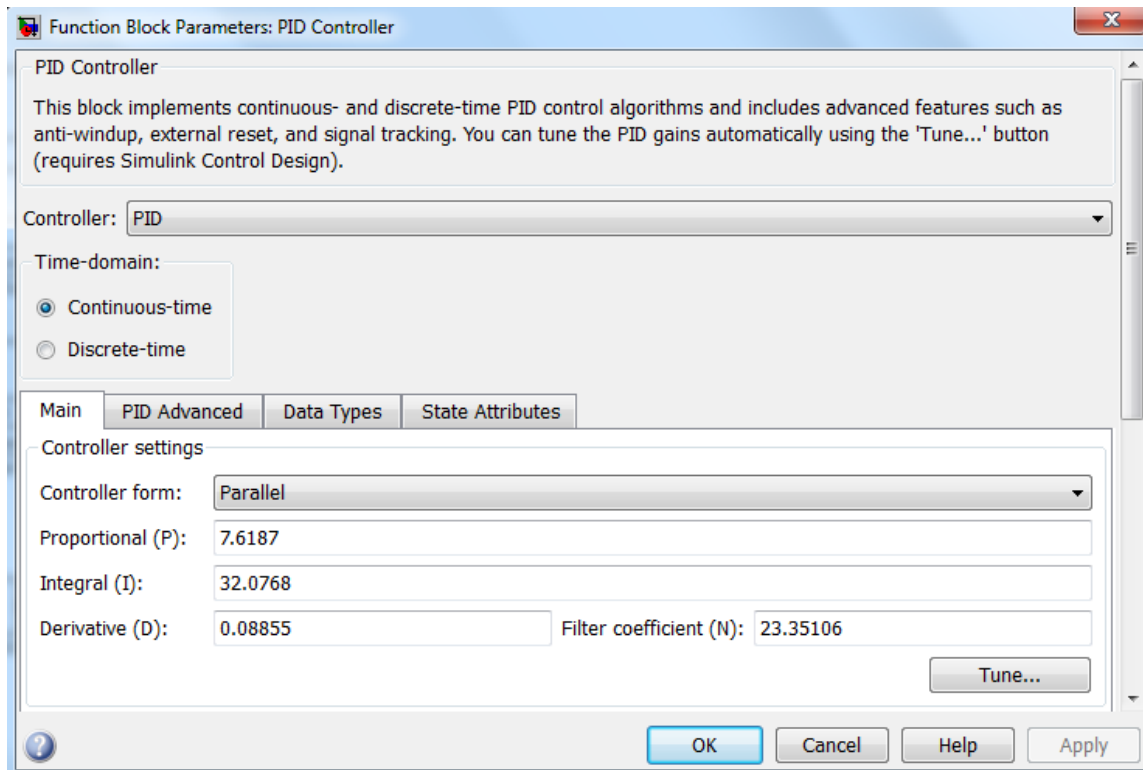


Fig 4.2

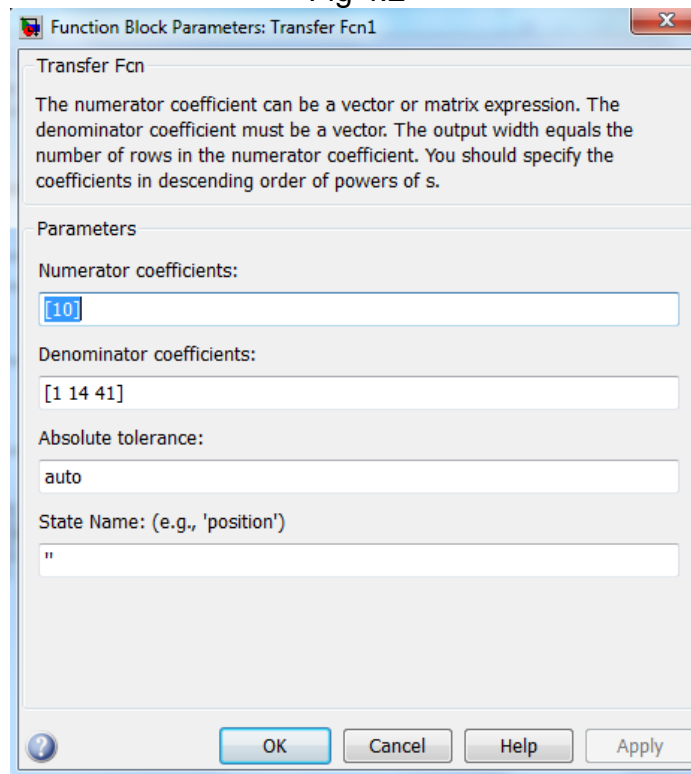


Fig 4.3

4.4.3. Corra el programa y visualice la respuesta a una entrada escalón unitario.

4.4.4. Modifique el programa agregando el diagrama mostrado en la fig. 4.4

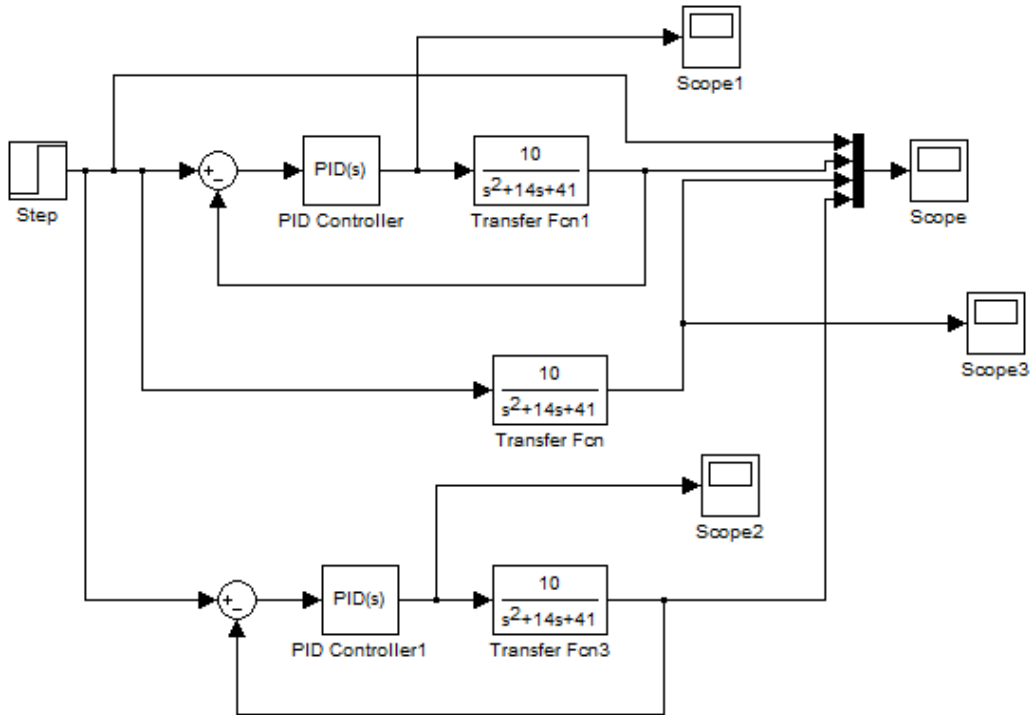


Fig. 4.4

4.4.5. Ajuste las constantes del controlador a P=120; I=20; D=0.96, así como un numerador de 10 y coeficientes en denominador de 1, 14 y 41. Fig 4.5

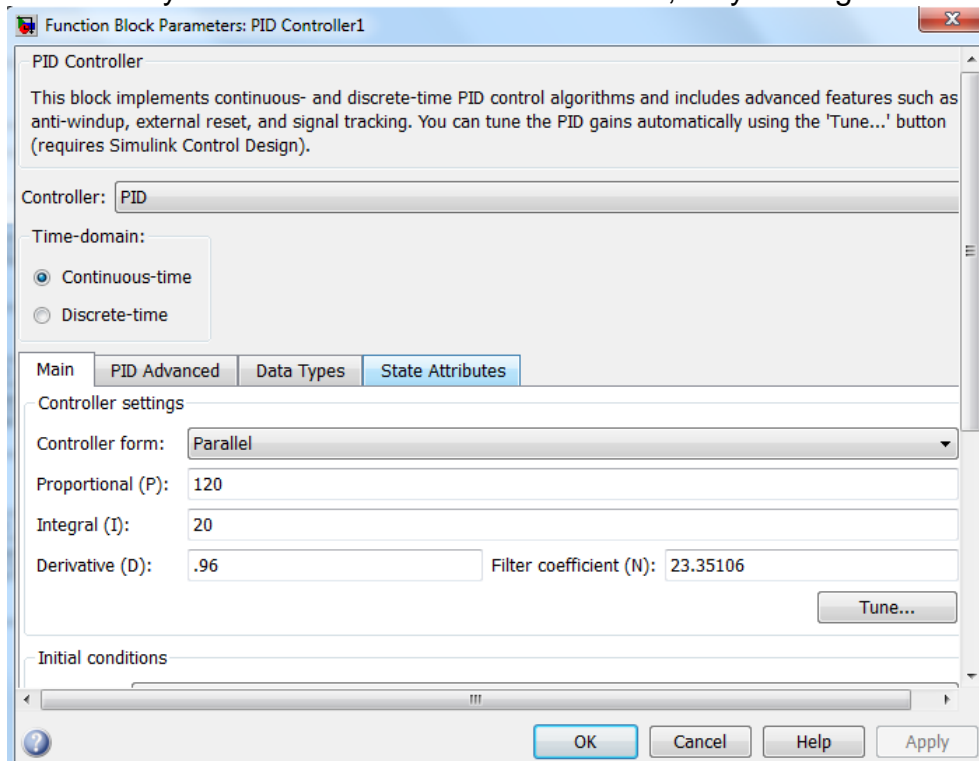


Fig 4.5

4.4.6. Corra el programa y visualice la respuesta a una entrada escalón unitario.

4.4.7. Modifique el valor de la constante integral del Controlador PID hasta perder la estabilización de la respuesta. Fig 4.6

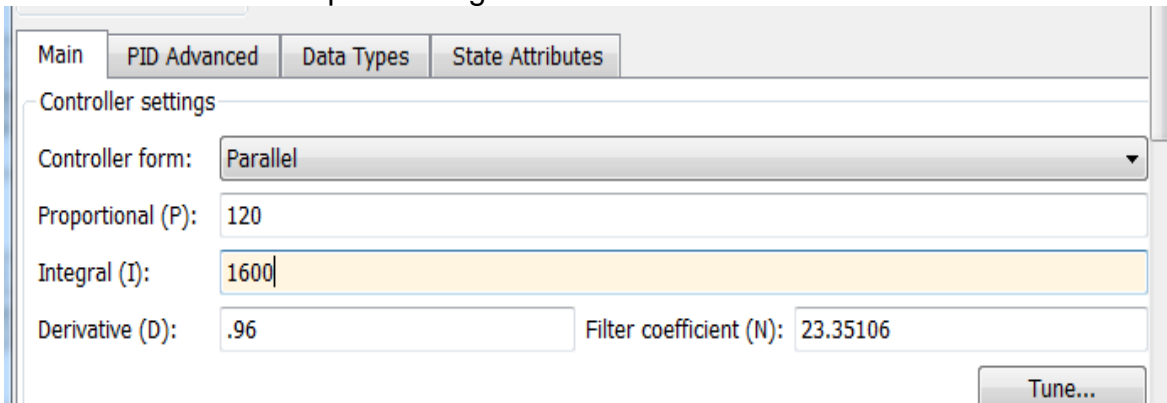


Fig 4.6

4.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

4.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo cada vez que realice una modificación importante, que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

4.7 CUESTIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones domésticas es posible instalar controles PID?

¿Con que otro tipo de tecnologías y principios físicos es posible construir controles PID?

4.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.

7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.

8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

4.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

4.10 REPORTE DEL ALUMNO

4.10.1 Desarrollar un reporte manuscrito con los siguientes puntos

4.10.2 La portada debe contener los siguientes datos:

Nombre de la escuela
Nombre de la materia
Nombre del profesor
Nombre del estudiante
Número de control
Número de la práctica
Nombre de la práctica
Lugar y fecha

4.10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas

y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: titulo, autores, editorial, año, numero de las paginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año

PRÁCTICA 5:

Identificación de Sistemas Dinámicos con Redes Neuronales: El Caso de un Levitador Magnético y un Sistema Masa-Resorte-Amortiguador.

5.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Identifica la dinámica de sistemas lineales mediante la interfaz de usuario de control neuronal NARMA-L2 de Matlab Simulink, para obtener el mejor índice de eficiencia.

5.2 INTRODUCCIÓN

IDENTIFICACIÓN

Para realizar un proceso de identificación, el primer paso es elegir un modelo de estructura de usar. Un modelo estándar que se utiliza para representar sistemas no lineales de tiempo discreto generales es el modelo autorregresivo promedio de movimiento no lineal (Narma):

$$y(k+d) = N[y(k), y(k-1), \dots, y(k-n+1), u(k), u(k-1), \dots, u(k-n+1)]$$

donde $u(k)$ es la entrada del sistema, e $y(k)$ es la salida del sistema. Para la fase de identificación, se puede entrenar una red neuronal para aproximar la función no lineal N . Este es el procedimiento de identificación utilizado para el controlador predictivo NN.

Si desea que la salida del sistema a seguir alguna trayectoria de referencia $y(k+d) = y_r(k+d)$, el siguiente paso es el desarrollo de un controlador no lineal de la forma

$$u(k) = G[y(k), y(k-1), \dots, y(k-n+1), y_r(k+d), u(k-1), \dots, u(k-m+1)]$$

El problema con el uso de este controlador es que si usted quiere entrenar una red neuronal para crear la función G para minimizar el error cuadrático medio, es necesario utilizar backpropagation dinámico ([NaPa91] o [HaJe99]). Esto puede ser bastante lento. Una solución, propuesta por Narendra y Mukhopadhyay [NaMu97], es el uso de modelos aproximados para representar el sistema. El controlador se utiliza en esta sección se basa en el modelo aproximado Narma-L2:

$$\hat{y}(k+d) = f[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] + g[y(k), y(k-1), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] \cdot u(k)$$

5.3 MATERIAL Y EQUIPO NECESARIO

5.3.1 Software

Matlab 6 o superior

5.3.2 Equipo

Computadora personal

5.4 METODOLOGÍA

NARMA-L2

5.4.1. Para correr el demo NARMA-L2, se siga los siguientes pasos:

5.4.1.1. Iniciar MATLAB®.

5.4.1.2. Correr el demo escribiendo en la ventana de commando de MATLAB la palabra "narmamaglev" (sin comillas). Este comando iniciara Simulink con el diagrama a bloques del control NARMA-L2. Figura 5.1

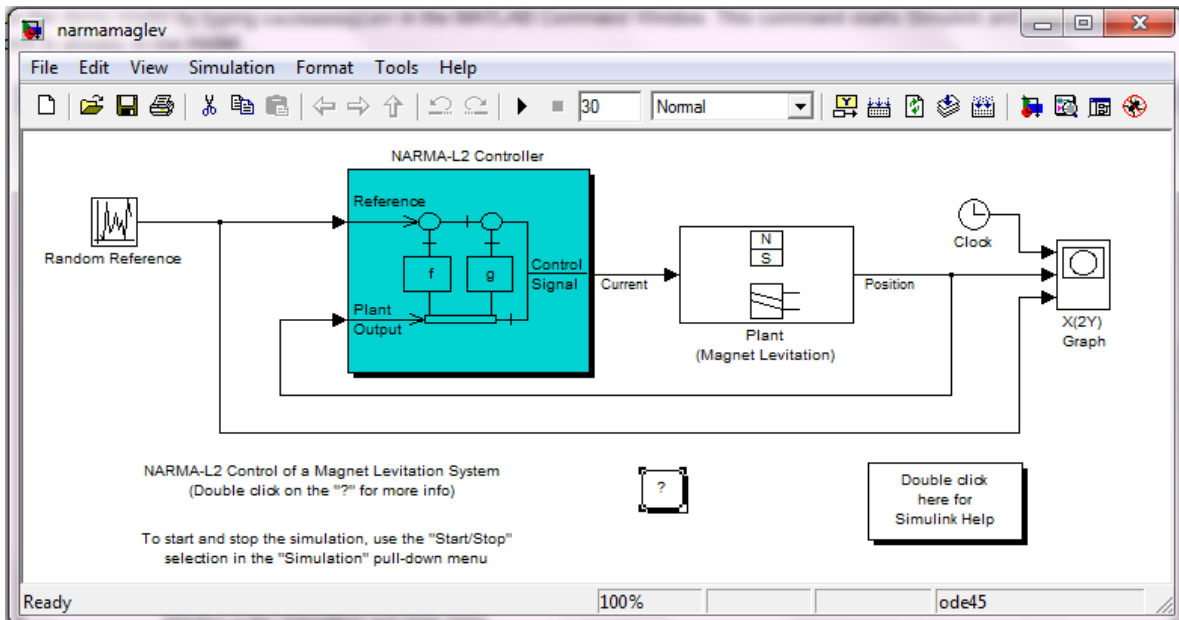


Fig 5.1

5.4.1.3. Da doble click en el bloque de control NARMA- L2. Esta acción abrirá una ventana en la cual puede modificar los valores originales de NARMA-L2. Fig 5.2

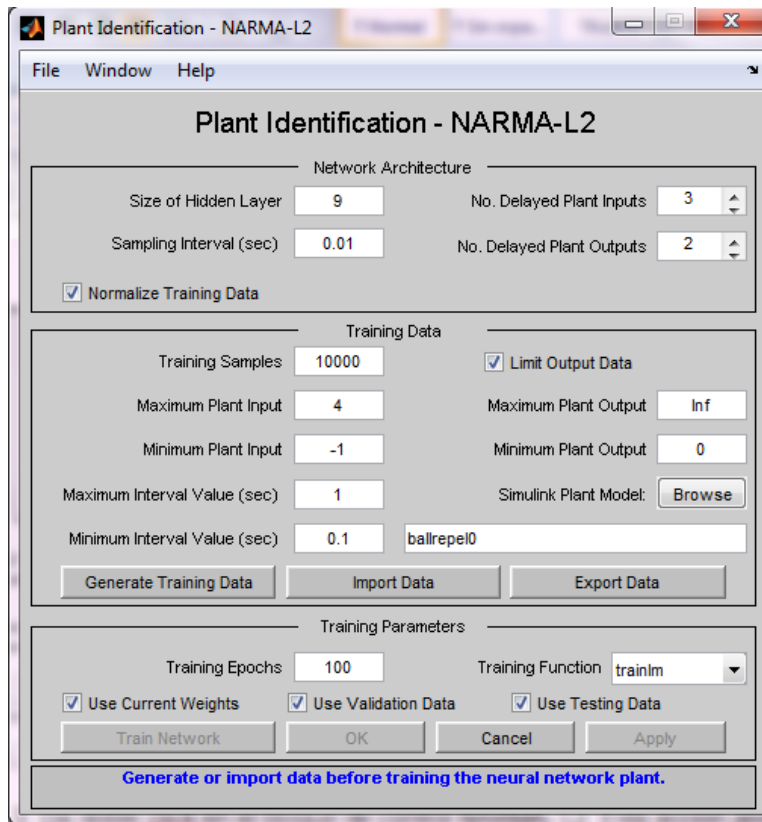


Fig 5.2

5.5.3. Cambia el número de neuronas ocultas a 50.

5.4.4. Realiza pruebas con 100 neuronas ocultas.

5.4.5. Finalmente regresa los valores a sus originales y describe el resultado obtenido.

5.4.6 Para cada una de las pruebas anteriores registre y reporte la ventana identificatoion.

5.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

5.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo cada vez que realice una modificación importante, que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

5.7 CUESTIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones cree usted que es posible usar esta técnica?
¿Qué otro tipo de técnicas es posible aplicar para obtener resultados similares?

5.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.
7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.
8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

5.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

5.10 REPORTE DEL ALUMNO

5.10.1 Desarrollar un reporte manuscrito con los siguientes puntos

5.10.2 La portada debe contener los siguientes datos:

Nombre de la escuela
Nombre de la materia
Nombre del profesor
Nombre del estudiante
Número de control
Número de la práctica
Nombre de la práctica
Lugar y fecha

5.10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: titulo, autores, editorial, año, numero de las paginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año

PRÁCTICA 6:

Control Neuronal NARMA-L2 de un Levitador Magnético.

6.1 COMPETENCIA ESPECÍFICA A DESARROLLAR: Diseña un control neuronal NARMA-L2 de Matlab Simulink, para controlar un levitador magnético y mejorar sus índices de desempeño.

6.2 INTRODUCCIÓN

NARMA-L2

El control neuronal descrito en esta práctica se denomina con dos nombres diferentes: retroalimentación de control de linealización y control Narma-L2.

Se denomina linealización por realimentación cuando el modelo de la planta tiene una forma particular. Esto se conoce como control de Narma-L2 cuando el modelo de la planta puede ser aproximada por la misma forma. La idea central de este tipo de control es transformar la dinámica de sistemas no lineales de la dinámica lineal mediante la cancelación de las no linealidades.

El controlador es simplemente un el modelo neuronal de la plante modificada, la cual es entrenada fuera de línea. La única desventaja de este método es de que la planta debe estar en la forma compañera o en una aproximación a esta forma.

6.3 MATERIAL Y EQUIPO NECESARIO

6.3.1 Software

Matlab 6 o superior

6.3.2 Equipo

Computadora personal

6.4 METODOLOGÍA

NARMA-L2

6.4.1. Para correr el demo NARMA-L2, se siga los siguientes pasos:

6.4.1.1. Iniciar MATLAB[®].

6.4.1.2. Correr el demo escribiendo en la ventada de commando de MATLAB la palabra "narmamaglev" (sin comillas). Este comando iniciara Simulink con el diagrama a bloques del control NARMA-L2. Figura 5.1

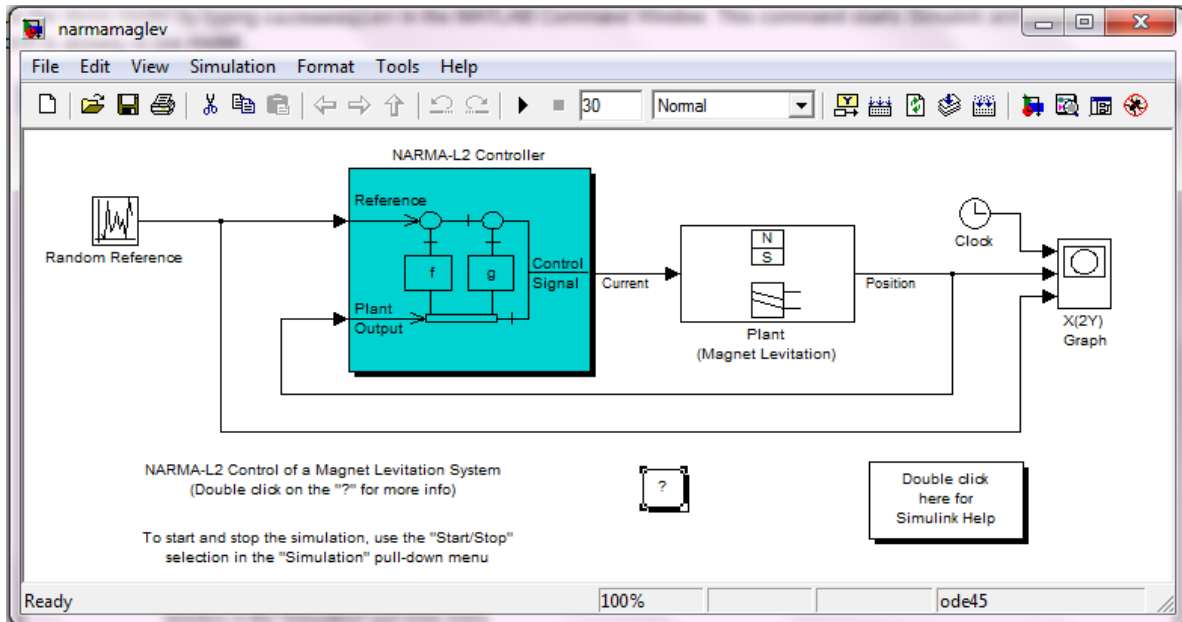


Fig 5.1

6.4.1.3. Da doble click en el bloque de control NARMA- L2. Esta acción abrirá una ventana en la cual puede modificar los valores originales de NARMA-L2. Fig 5.2

The 'Plant Identification - NARMA-L2' dialog box is shown with the following settings:

- Network Architecture:**
 - Size of Hidden Layer: 9
 - No. Delayed Plant Inputs: 3
 - Sampling Interval (sec): 0.01
 - No. Delayed Plant Outputs: 2
 - Normalize Training Data
- Training Data:**
 - Training Samples: 10000
 - Maximum Plant Input: 4
 - Minimum Plant Input: -1
 - Maximum Interval Value (sec): 1
 - Minimum Interval Value (sec): 0.1
 - Simulink Plant Model: ballrepel0
 - Limit Output Data
 - Maximum Plant Output: Inf
 - Minimum Plant Output: 0
- Training Parameters:**
 - Training Epochs: 100
 - Training Function: trainlm
 - Use Current Weights
 - Use Validation Data
 - Use Testing Data

Buttons: Generate Training Data, Import Data, Export Data, Train Network, OK, Cancel, Apply.

Footer: Generate or import data before training the neural network plant.

Fig 5.2

6.4.1.4. La simulación de NARMA-L2 es iniciada en la ventana de Simulink en el botón de Start, una vez iniciada la simulación se abrirá una ventana con la salida de la planta. Fig 5.3

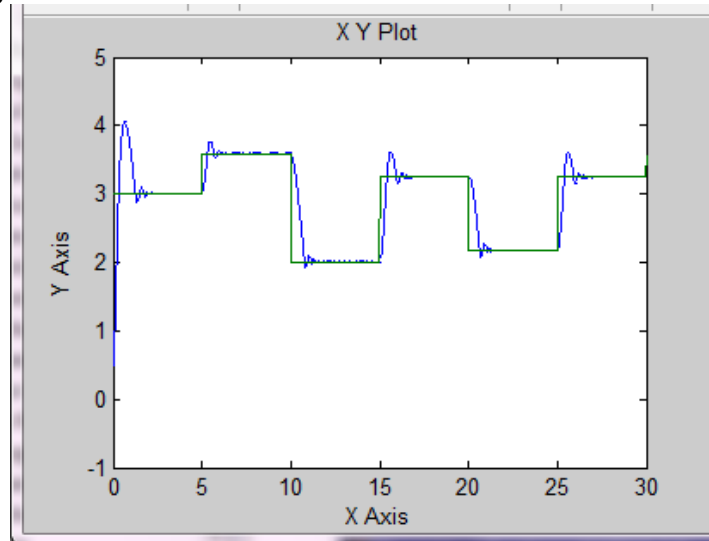


Fig 5.3

6.4.2. Una vez comprendido el funcionamiento de la función NARMA-L2, modifique el número de neuronas internas a 25.

6.5.3. Cambie el número de neuronas ocultas a 50.

6.4.4. Realice pruebas con 100 neuronas ocultas.

6.4.5. Finalmente regrese a los valores a sus originales y describe el resultado obtenido.

6.5 RECOMENDACIONES

Utilice un equipo de cómputo de capacidades estándar para poder realizar el entrenamiento de las redes, ya que algunos equipos de recursos reducidos pueden fallar en el entrenamiento debido al desbordamiento de la memoria.

6.6 OBSERVACIONES

Realice la grabación del diagrama o archivo de trabajo cada 15 minutos para prevenir que se borre debido a descuidos en el manejo del software o por pérdida de energía en el equipo de cómputo.

Realice la grabación de los archivos o diagramas renombrando cada nueva versión con un subíndice al final del archivo cada vez que realice una modificación importante, que suponga resultados distintos para el reporte de manera que pueda repetir todos los resultados encontrados en la práctica para futuras referencias.

6.7 CUESTRIONARIO DE REFLEXIÓN

¿En qué tipo de aplicaciones cree usted que es posible usar este tipo de control?

¿Cuáles son las ventajas que usted siguiere para este control respecto del control PID clásico?

6.8 FUENTES DE INFORMACIÓN

1. Ogata, K. (2009). Modern Control Engineering. 5th. Edition. Prentice Hall. USA.
2. Ogata, K. (2010). Ingeniería de Control Moderno. 5^a Edición. Prentice Hall. México.
3. Kuo, B. (2009). Automatic Control Systems. Prentice Hall. USA.
4. Dorf, R. (2010). Modern Control Engineering. 12^a Edición. Pearson Education. USA.
5. Ponce, P. (2010). Inteligencia Artificial con aplicaciones a la ingeniería. Alfaomega. México.
6. Haykin, S. (1998). Neural Networks. A Comprehensive Foundation. Prentice Hall.
7. Sánchez, E. (2006). Redes Neuronales, Conceptos Fundamentales y Aplicaciones a Control Automático. Prentice Hall.
8. Demuth H. Neural Network Toolbox, User's Guide, Matlab Mathworks

6.9 NORMAS DE SEGURIDAD

Realice la práctica en equipo de cómputo estándar y bajo control de mantenimientos frecuente.

6.10 REPORTE DEL ALUMNO

6.10.1 Desarrollar un reporte manuscrito con los siguientes puntos

6.10.2 La portada debe contener los siguientes datos:

- Nombre de la escuela
- Nombre de la materia
- Nombre del profesor
- Nombre del estudiante

Número de control
Número de la práctica
Nombre de la práctica
Lugar y fecha

6.10.3 Estructura del reporte

Competencia a desarrollar: Escriba la competencia a desarrollar en esta práctica, puede encontrarla en este manual el principio de la descripción de esta práctica.

Introducción: En un párrafo describa con sus propias palabras cual es el objetivo de la práctica.

Marco teórico: En una página describa con sus propias palabras a manera de ensayo los conceptos básicos que entendió de la información contenida en la introducción previa de esta práctica, puede usted utilizar el recurso de las graficas y los diagramas. Es obligatorio utilizar al menos una referencia bibliográfica para complementar su ensayo.

Desarrollo de la práctica: Comenta la secuencia de pasos que se realizaron para obtener los resultados de la práctica.

Resultados: Presenta el reporte de los resultados en la forma que el profesor te indique (graficas, tablas, figuras, diagramas etc.).

Conclusiones: En un párrafo describe cuáles son tus conclusiones acerca del desarrollo de la práctica y de los resultados, además de comentar brevemente alguno de los problemas presentados durante el proceso.

Bibliografía: Has una lista de las referencias bibliográficas usadas para consulta de acuerdo al siguiente formato:

Libro: titulo, autores, editorial, año, numero de las paginas consultadas

Artículo: nombre del artículo, autor(es), publicación, año.