

SIMULACIÓN E IDENTIFICACIÓN DE SISTEMAS DINÁMICOS MEDIANTE REDES NEURONALES ENTRENADAS CON EL MÉTODO DE RETROPROPAGACIÓN DE ERRORES Y TEACHER FORCING

V. Leonardo Paucar
Lpaucar@ieee.org

Marcos J. Rider
mjriderr@ieee.org

André L. Morelato
morelato@dsee.fee.unicamp.br

**Facultad de Ingeniería Eléctrica y Electrónica, Universidad Nacional de Ingeniería
Departamento de Ingeniería Eléctrica, Universidade Federal do Maranhão
Departamento de Sistemas de Energía Eléctrica, Universidade Estadual de Campinas**

RESUMEN

En el presente artículo se presenta la descripción y resultados de la aplicación del algoritmo para la simulación e identificación de sistemas dinámicos no lineales mediante redes neuronales artificiales (RNA) entrenadas con el método de retropropagación de errores (BP - back-propagation) y el procedimiento teacher forcing (BPTF). Fueron analizadas varias configuraciones de redes neuronales de dos capas de neuronas, una escondida y la otra de salida. Las redes neuronales propuestas han sido aplicadas a dos sistemas de prueba, el sistema dinámico del péndulo doble y el motor de inducción de tercer orden. Los resultados obtenidos permiten estimar que las redes neuronales que adoptan BPTF son bastante útiles para la simulación e identificación de sistemas dinámicos no lineales, principalmente durante los primeros pasos de tiempo posteriores a los períodos con los cuales fueron entrenadas las redes neuronales en estudio.

ABSTRACT

In this paper it is presented the description and application results of an algorithm for simulation and identification of non-linear dynamic systems using artificial neural networks (ANN) trained with the method of error back-propagation with teacher forcing (BPTF). There have been analyzed several configurations of neural nets of two layers, one hidden and the other one in the output. The proposed artificial neural networks have been applied to simulation and identification of double pendulum non-linear dynamic system and to the third-order model of the induction motor. Results obtained from those applications indicate that artificial neural networks using BPTF constitute a very useful method for non-linear dynamic systems simulation and identification, at least for the first steps after the training

INTRODUCCIÓN

En las décadas recientes diversas técnicas de inteligencia artificial (IA) han comenzado a aplicarse en áreas de la ingeniería eléctrica y electrónica así como en otras áreas del conocimiento. Una de las técnicas IA que ha recibido bastante atención son las redes neuronales artificiales (RNA) las cuales desde la popularización del método de entrenamiento de back-propagation error (BP) o retropropagación de errores en 1986 [2, 3, 4, 13, 11], comenzaron a ser utilizadas ampliamente para resolver los más variados problemas, tanto de naturaleza estática como dinámica. [3, 4, 7]

Las redes neuronales pueden clasificarse en dos grandes grupos: las redes no recurrentes (feedforward) y las recurrentes o realimentadas. En teoría el primer grupo representa la posibilidad de realizar mapeamientos de funciones no lineales de una manera estática, mientras que en el otro grupo surge la opción de representar sistemas dinámicos no lineales realimentados. [2, 3, 4, 6, 11]

El problema de identificación de sistemas es uno de los mayores desafíos dentro de la teoría de los sistemas dinámicos y de control. A pesar de que existen varias referencias sobre la caracterización de las

funciones no lineales, solamente algunas técnicas específicas pueden ser aplicadas en la identificación de sistemas dinámicos prácticos. [5, 10, 12]. Aplicaciones más modernas indican que las redes neuronales vienen siendo aplicadas para el modelamiento de sistemas dinámicos no lineales [1]. Una de las ventajas de las redes neuronales artificiales es que se favorece el modelamiento implícito y sin tanta complicación de los sistemas dinámicos e independiente de la complejidad de los modelos matemáticos involucrados, ello también debido en parte a que la implementación con la tecnología computacional actual no es muy difícil ni demorada. [2, 5, 10, 11]

La gran capacidad que tienen las redes neuronales de perceptrons en multicamadas para la aproximación de funciones no lineales usando el algoritmo de entrenamiento de retropropagación de errores (BP por back-propagation error) torna factible su uso en la identificación, simulación y control de los sistemas dinámicos. [6, 2]

En este trabajo se detallan las consideraciones para la aplicación del método back-propagation usando el procedimiento teacher forcing (BPTF) o BP con profesor forzado, el cual permite, vía la utilización de algunos retardos (*delays*), que las RNA multicamada sean utilizadas para identificar y simular sistemas dinámicos no lineales. Los resultados obtenidos indican la posibilidad de una previsión adecuada de algunos estados futuros de los sistemas motivo de aplicación. El sistema dinámico no lineal del péndulo doble fue utilizado para mostrar el algoritmo BPTF y su utilidad en las identificaciones realizadas. La otra aplicación que aborda el caso del motor de inducción de tercer orden se orienta hacia una aplicación más práctica en simulación e identificación de parámetros de las máquinas eléctricas y los sistemas de potencia.

DINÁMICA DEL PÉNDULO DOBLE

Previamente a la presentación del algoritmo de back-propagation con teacher forcing, de manera similar a un ejemplo tutorial será descrita la dinámica del péndulo doble (ver Fig. 1) modelado en el dominio del espacio de estados, y desarrollado usando un conjunto de ecuaciones diferenciales de primer orden. El péndulo doble consiste de dos masas (esferas) sostenidas por sendos brazos de longitudes que pueden

condiciones iniciales correspondientes a posiciones y velocidades angulares, estas últimas normalmente son cero. Por efecto del propio peso de las masas, el péndulo doble comienza a oscilar en torno a su eje y cada esfera tendrá una evolución que dependerá de las condiciones iniciales y de los valores de los parámetros tales como la masa de las esferas y longitudes de los brazos.

La dinámica del sistema del péndulo doble puede representarse por el siguiente conjunto de ecuaciones diferenciales no lineales de primer orden:

$$\dot{x}_1 = x_2 \quad (1)$$

$$\dot{x}_2 = -\frac{g}{L_1} \sin x_1 + \frac{m_2}{m_1} \frac{g}{L_1} \cos x_3 \sin(x_3 - x_1) \quad (2)$$

$$\dot{x}_3 = x_4 \quad (3)$$

$$\dot{x}_4 = -\frac{g}{L_2} \sin x_3 \quad (4)$$

siendo:

$$x_1 = \theta_1$$

$$x_2 = \dot{\theta}_1$$

$$x_3 = \theta_2$$

$$x_4 = \dot{\theta}_2$$

La integración numérica de las ecuaciones (1) a (4) puede ser realizada usando por ejemplo el método explícito Runge-Kutta de cuarto orden, como se hizo en este trabajo para generar las trayectorias mostradas en la Fig. 2.

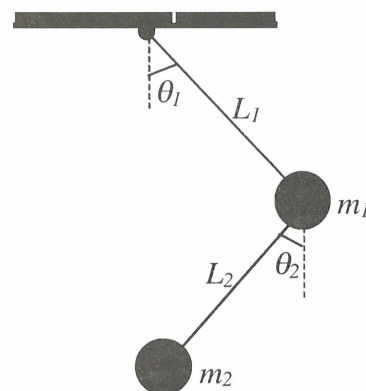


Fig. 1 Péndulo doble usado como sistema dinámico no lineal en las aplicaciones de prueba.

En la figura 2 se presentan algunas simulaciones considerando los parámetros $m_1 = 0.1$ kg, $m_2 = 0.2$ kg, $L_1 = 0.2$ m, $L_2 = 0.4$ m. Además de ello fueron consideradas las siguientes condiciones iniciales: $x_1 = x_3 = 45^\circ$; $x_2 = x_4 = 0$.

Las simulaciones mostradas en la Fig. 2 indican la no linealidad del sistema dinámico. Otras simulaciones usando diferentes condiciones iniciales pueden alterar el movimiento del péndulo y consecuentemente pueden ser generadas otras curvas trayectorias.

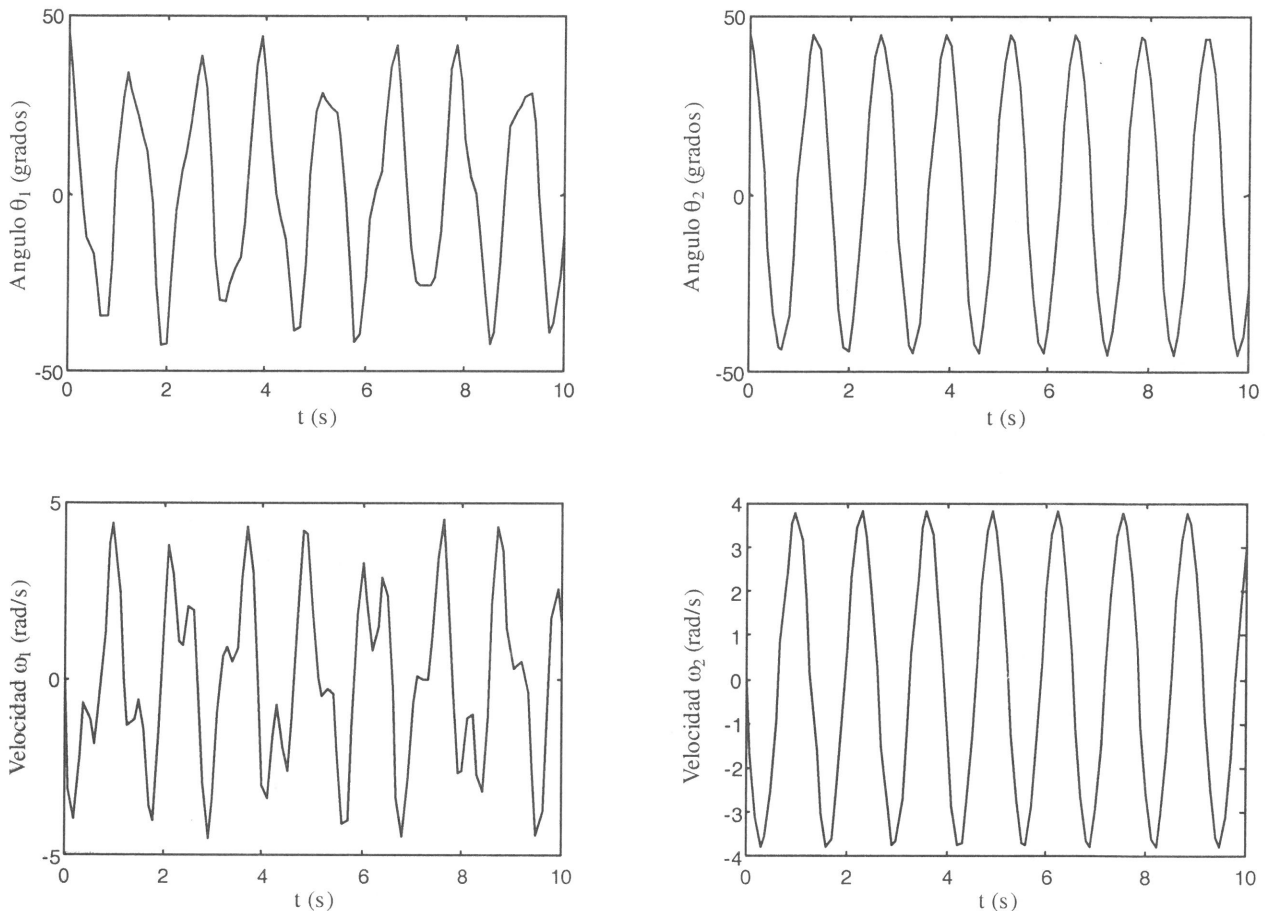


Fig. 2 Simulaciones de la dinámica del péndulo doble mostrando las posiciones angulares q_1 y q_2 y las velocidades w_1 y w_2 cuando las condiciones iniciales são: $x_1 = x_3 = 45^\circ$, $x_2 = x_4 = 0$.

BACK-PROPAGATION CON TEACHER FORCING

Usando retardos o *delays*, el procedimiento teacher forcing introduce dinámica en el entrenamiento de RNA back-propagation evitando saturaciones en el proceso de realimentación. En la Fig. 3 se sugiere cómo realizar el entrenamiento de la red de perceptrons multicamada. El método teacher forcing consiste en ingresar los valores de salida deseados dentro de las unidades de entrada a medida que la red funciona; consecuentemente el profesor o *teacher* obliga a las unidades de salida a tener los estados correctos.

Independiente del método de entrenamiento, la minimización del error cuadrático medio, $e^2(n)$, es un esquema de adaptación en lazo abierto puesto que durante el entrenamiento la salida real de la RNA no se realimenta a la entrada. En la literatura sobre control y procesamiento de señales, ese método de entrenamiento es referenciado como la formulación de la ecuación de error, mientras que en la literatura sobre RNA se conoce como teacher forcing. [5, 3] En la Fig. 3 la llave A indica la posición para entrenamiento cuando la llave B es la opción para usar la red neuronal. El esquema de adaptación en malla abierta o posición de entrenamiento (posición A),

representa la previsión para adelante. Luego de que el entrenamiento de la red neuronal se ha completado, el desempeño de generalización de la red neuronal se evalúa con una predicción recursiva autónoma o esquema de adaptación en malla cerrada (posición B). La formulación de un sistema dinámico no lineal en el dominio del espacio de estados es la siguiente:

$$x_k = f(x_{k-1}, u_{k-1}) \quad (5)$$

$$y_k = g(x_k) \quad (6)$$

Para reproducir la evolución de un sistema dinámico no lineal se requiere el conjunto de condiciones iniciales del sistema.

Se usan atrasos de tiempo para simular la discretización de la dinámica del sistema. El procedimiento teacher forcing mantiene la salida de la red neuronal muy próximo del camino deseado y parece acelerar el entrenamiento y aprendizaje.

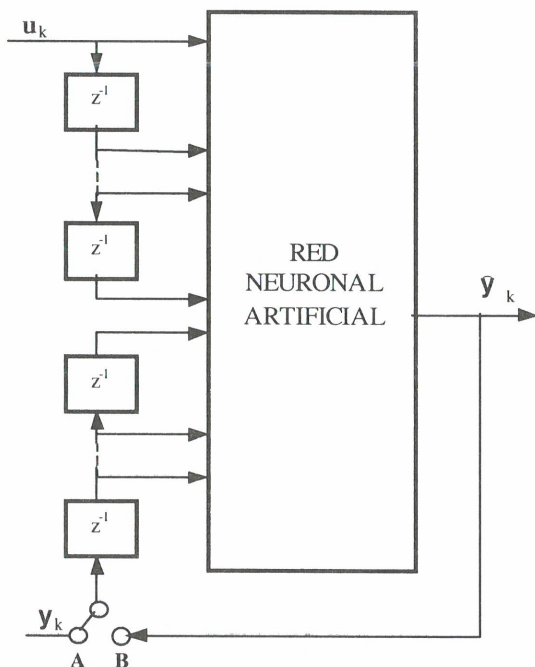


Fig. 3 Arquitectura del método back-propagation con teacher forcing.

La red neuronal artificial quedará correctamente entrenada solamente dentro del intervalo (región de atracción) que envuelve el conjunto de condiciones iniciales. Una previsión de corto plazo se calcula iterativamente al realimentarse la secuencia de predicciones de un paso para adelante, calculadas por la red, a la entrada.

Después de la inicialización no hay entradas externas aplicadas a la red neuronal. La predicción es considerada como de corto plazo porque normalmente es posible mantener una identificación confiable durante un número limitado de pasos de tiempo futuros. El comportamiento de la red neuronal es muy sensible a las condiciones iniciales y la calidad de las predicciones se degradan rápidamente una vez que se pasan un cierto número de pasos de tiempo futuros [5, 13, 6].

Los pasos de tiempo futuros se refieren a aquellos instantes de tiempo posteriores al período de tiempo considerado durante el entrenamiento de la red neuronal. Así por ejemplo, si una red fue entrenada durante 5 segundos, los pasos de tiempo futuros corresponden a cualquier instante posterior a 5 s. La red neuronal con entrenamiento de retropropagación de errores y profesor forzado identifica muy bien un sistema dinámico hasta algunos instantes posteriores a los 5 s, luego pierde calidad progresivamente.

RESULTADOS DE LA IDENTIFICACIÓN CON EL PÉNDULO DOBLE

El péndulo doble es el primer sistema dinámico no lineal para probar el método back-propagation con teacher forcing. La red neuronal no recurrente fue entrenada para reproducir el movimiento del péndulo dada una cierta condición inicial. Fue desarrollado un programa computacional en lenguaje C que implementa el BPTF para una red de varias capas escondidas de neuronas. El programa computacional lee la arquitectura de la red, la tasa de aprendizaje, el momentum y el número de atrasos de tiempo deseados. Para el entrenamiento fue prevista la realización de super épocas que puedan asegurar un entrenamiento completo para esas condiciones. Luego del entrenamiento la RNA puede identificar o reproducir la dinámica del sistema no lineal a partir de una condición inicial dentro del intervalo considerado en el entrenamiento.

Ha sido propuesta una arquitectura RNA de dos capas escondidas con 50 neuronas cada una y 4 neuronas adicionales en la capa de salida, 8 delays para cada unidad de salida realimentada a la correspondiente unidad de entrada (ver Fig. 3). Ambas posiciones angulares q_1 y q_2 , y las velocidades angulares w_1 y w_2 , han sido los correspondientes a las unidades de salida.

Durante el proceso de entrenamiento fueron usadas 20 condiciones iniciales aleatorias dentro del intervalo de 30 a 60 grados. Como criterio de convergencia fue fijado un error cuadrático medio de $1e-6$ para los 50 puntos de entrenamiento dentro del intervalo de tiempo predefinido de 5 s.

En la Fig. 4 se muestran los datos de entrenamiento de los estudios de simulación en el sistema de prueba del péndulo doble, tomando en cuenta todos los datos de velocidades y posiciones angulares para el entrenamiento. Esos gráficos fueron obtenidos para 20 condiciones iniciales aleatorias dentro del intervalo de 30 a 60 grados. El tiempo de simulación total para aplicación de la red neuronal entrenada fue 10 s.

Ese hecho se justifica debido al movimiento periódico del péndulo doble.

Por otra parte en la Fig. 5 son mostrados los errores simples indicados en porcentaje para las posiciones y velocidades angulares. Esos errores dan una indicación de que para algunos pasos futuros inmediatamente después de los 5 s, la red neuronal fue capaz de prever la dinámica del péndulo doble con una precisión bastante aceptable.

En vista de que fueron considerados 8 atrasos (0.7 s), este período no presenta error, luego el error va aumentando en cada predicción hasta que en cierto momento es bastante grande.

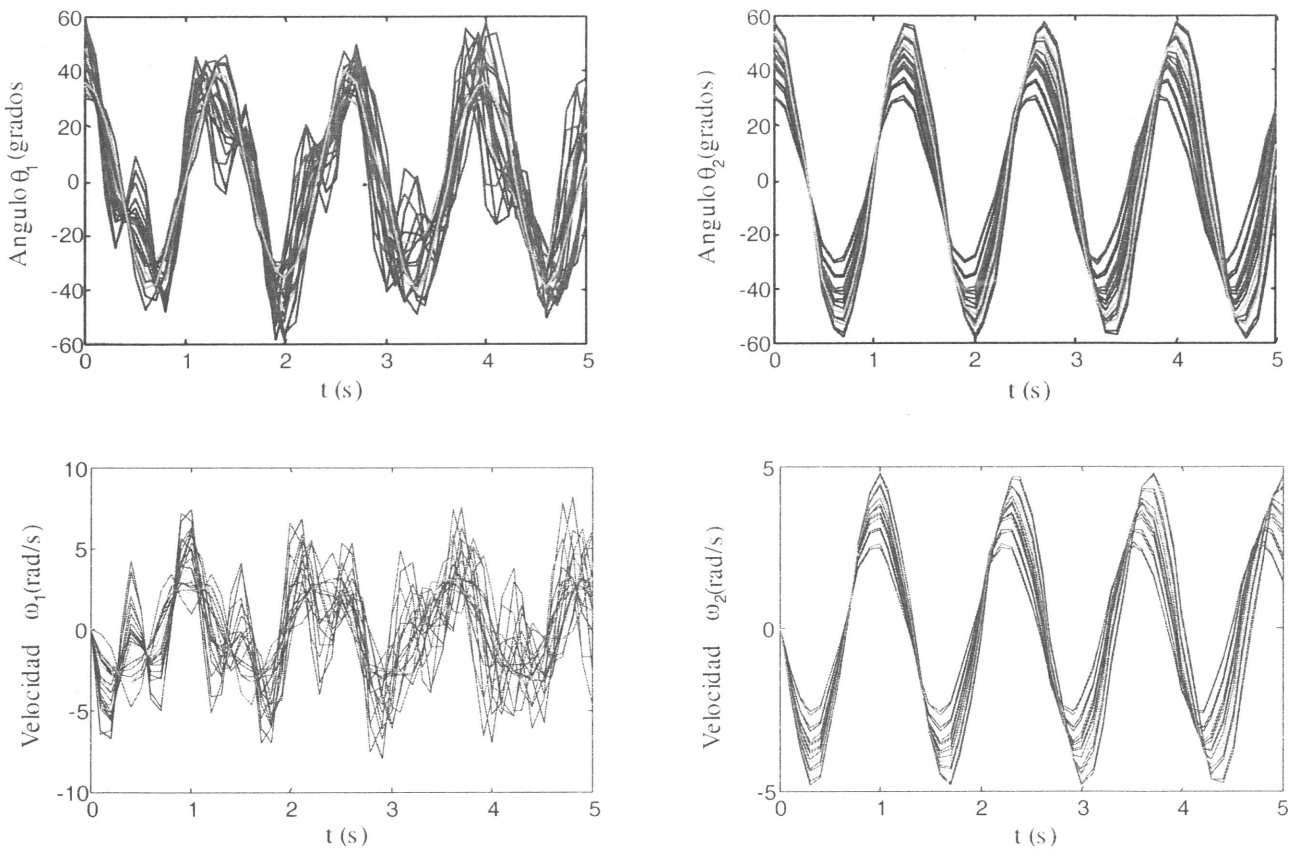


Fig. 4 Datos de entrenamiento del péndulo doble.

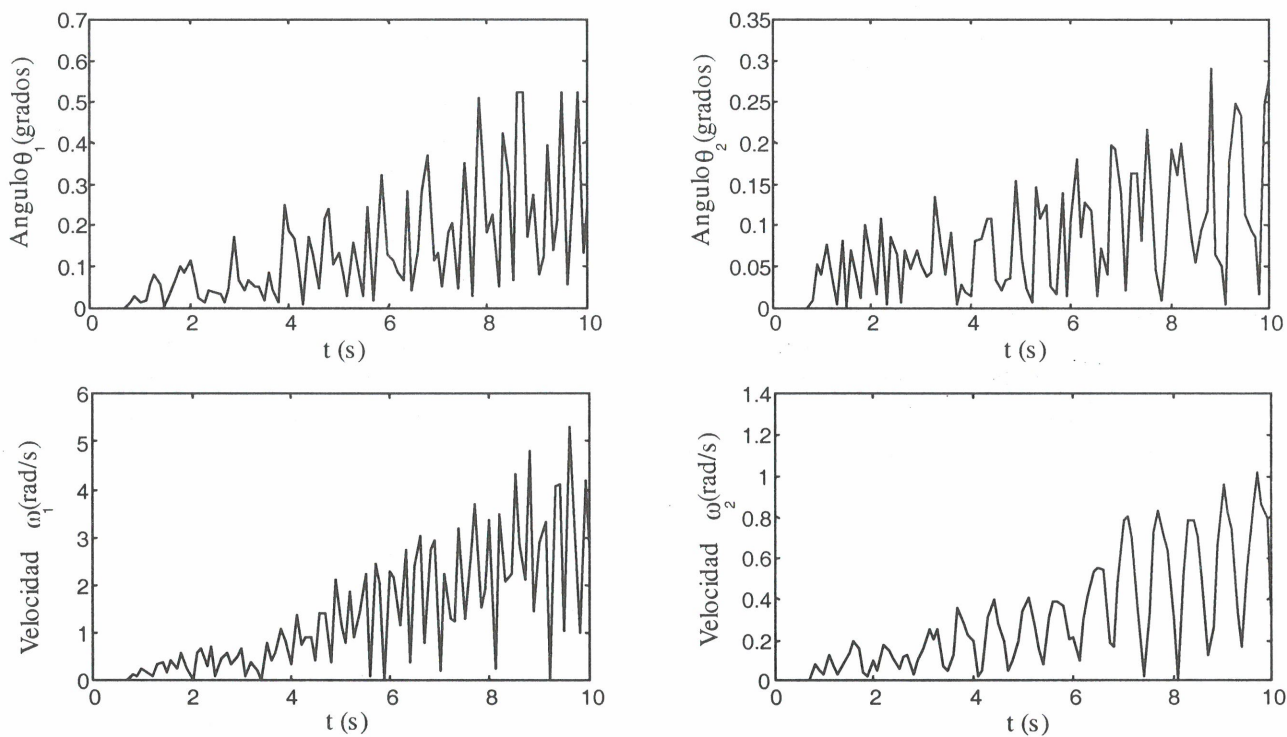
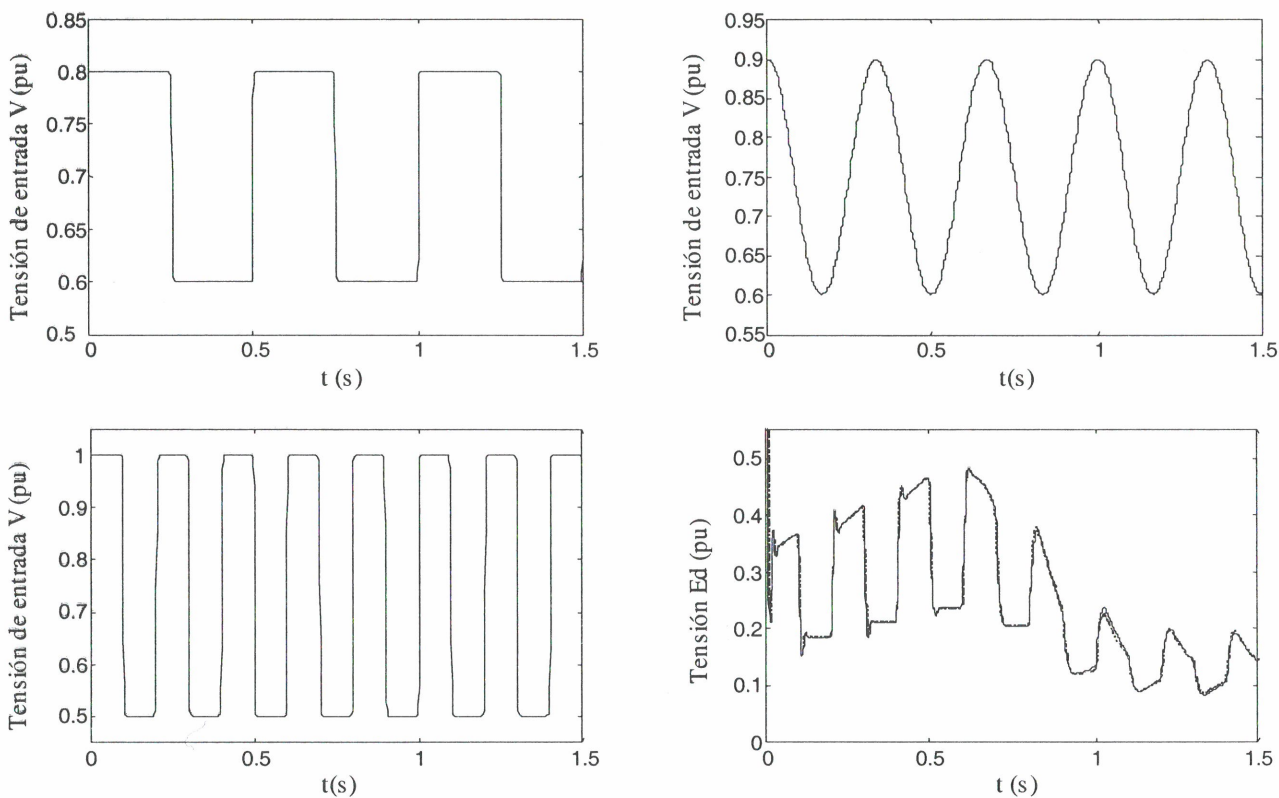


Fig. 5 Error simple de la prueba de la RNA para el péndulo doble.



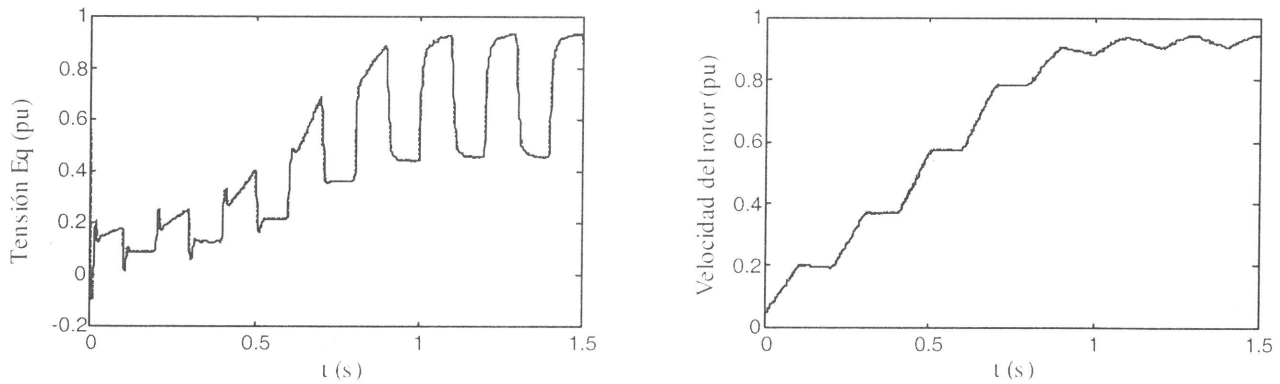


Fig. 6 Datos de entrenamiento para la máquina de inducción.

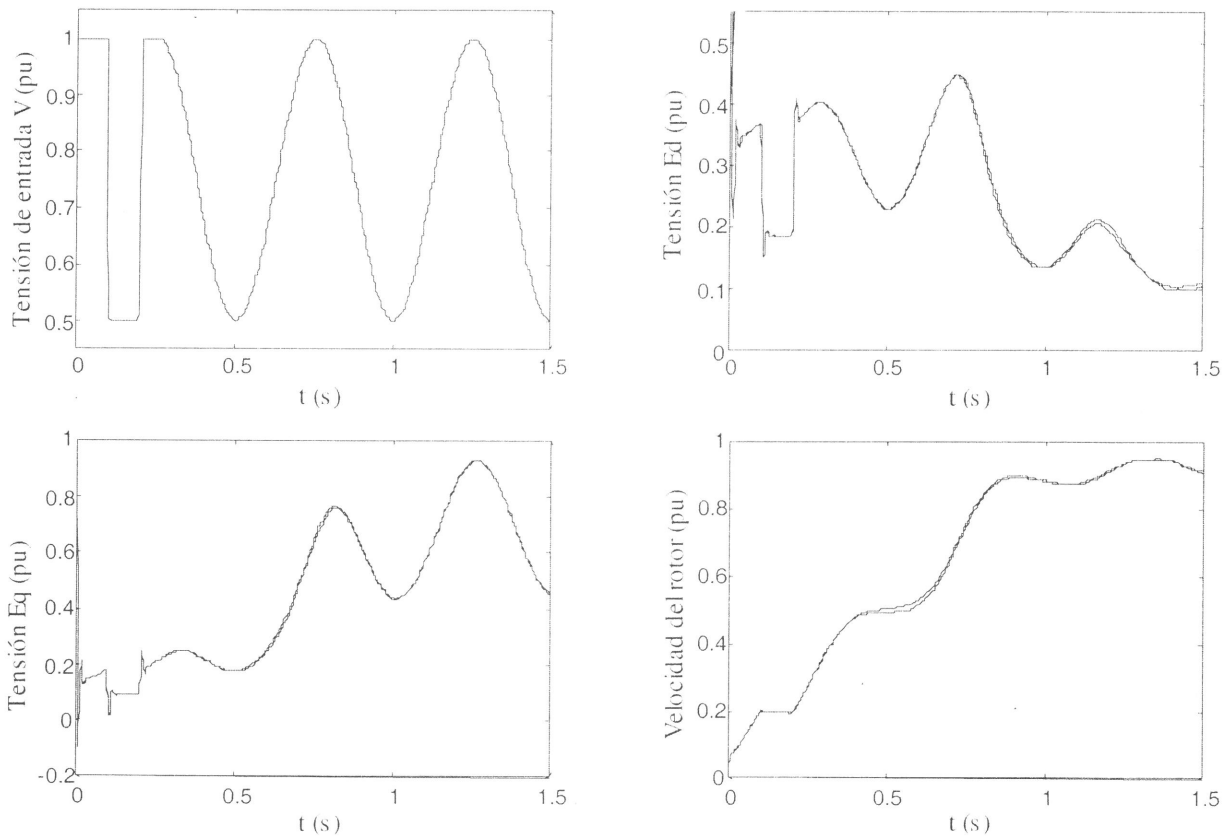


Fig. 7 Resultados de la máquina de inducción considerando la identificación con RNA y con ecuaciones diferenciales.

Extensas simulaciones y usando un número diferente de atrasos indican que un número mayor de atrasos mejora el comportamiento de la red neuronal entrenada, para la predicción de un mayor número de pasos futuros. La utilización de pocos atrasos no mejora la capacidad de predicción de la red neuronal de más pasos futuros. Redes neuronales con un mayor número de neuronas puede mejorar el entrenamiento

hasta cierto punto, sin embargo existen limitaciones para redes grandes. Las características y consideraciones del algoritmo de de entrenamiento back-propagation estándar son válidas para el método teacher forcing.

LA DINÁMICA DEL MOTOR DE INDUCCIÓN

El algoritmo de redes neuronales BPTF será probado con el sistema dinámico del motor de inducción. Los datos de entrada y salida para la simulación se han obtenido con el modelo de tercer orden del motor, el cual se obtiene a partir del modelo de quinto orden despreciando los efectos transitorios del estator de la máquina de inducción. La representación matemática del modelo de tercer orden del motor de inducción se muestra en las siguientes ecuaciones diferenciales [8, 9]:

$$T_o' \frac{dE_q'}{dt} = -\frac{X}{X'} E_q' + \frac{X - X'}{X'} V + T_o' w_s (v - 1) E_d' \quad (7)$$

$$T_o' \frac{dE_d'}{dt} = -\frac{X}{X'} E_d' - T_o' w_s (v - 1) E_q' \quad (8)$$

$$2H \frac{dv}{dt} = \frac{VE_d'}{X} - T_m \quad (9)$$

En las ecuaciones (7) a (9) v es la velocidad normalizada del rotor ($v = \omega/\omega_s$), ω_s es la velocidad

síncrona (377 rad/s). La constante de inercia del rotor es representada por H , T_o' es la constante de tiempo transitoria a circuito abierto; X' , X son las reactancias síncrona y transitoria. Las variables de la tensión del rotor en los ejes d-q son representadas por E_d' y E_q' . T_m es el torque mecánico y V es la tensión terminal.

Los parámetros del motor considerados en el trabajo son: $X = X' = 2.36$ pu (por unidad), $H = 0.7916$ s, $T_o' = 0.1557$ s y $T_m = 1.0$ pu.

En el modelo de la red neuronal para representar el modelo de la máquina de inducción de tercer orden, la tensión de entrada V , las tensiones E_d' , E_q' y la velocidad v del rotor son las entradas al modelo de la red neuronal. La salida de la red neuronal son las tensiones rotóricas en los ejes d-q y la velocidad normalizada del rotor.

La estructura de la red que representa la máquina de inducción de tercer orden es una red neuronal de perceptrons multicamada con 32 entradas, 3 capas escondidas de 25 neuronas cada una, 8 atrasos para cada unidad de salida realimentando a la respectiva

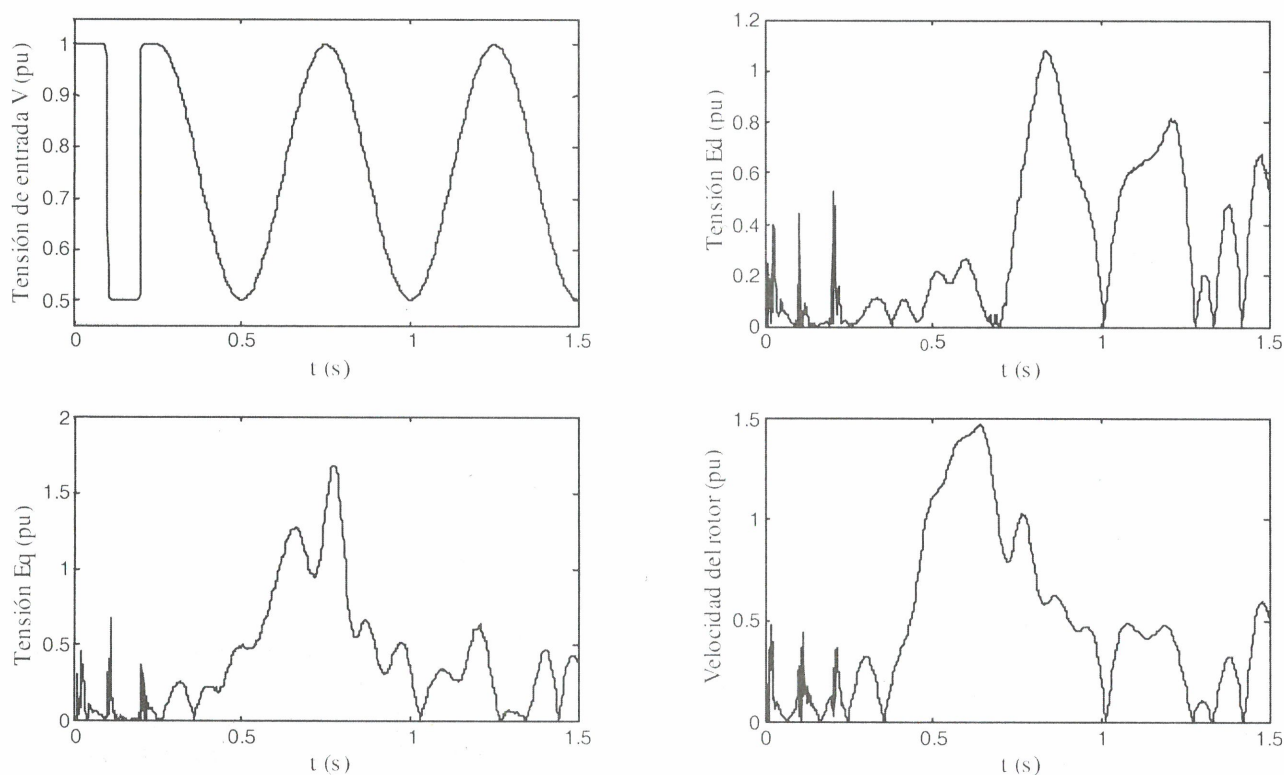


Fig. 8 Error simple de la prueba de la RNA para la máquina de inducción de tercer orden

unidad de entrada (ver Fig. 3). Tres diferentes tensiones de excitación V mostradas en la Fig.6 fueron aplicadas al modelo de motor de inducción para obtener los datos para el entrenamiento de la red neuronal. Las respuestas de la red neuronal artificial y del modelo de tercer orden del motor de inducción se muestran en la Fig. 6 y Fig.7. En la Fig.6 se tienen las tensiones E_d' , E_q' y la velocidad del rotor para la tercera tensión de excitación V aplicada para el entrenamiento. La curva sólida representa la respuesta del modelo de tercer orden y la curva restante indica la respuesta de la red neuronal. En las dos figuras citadas el error es bien pequeño y las curvas de los valores verdadero y calculado son casi coincidentes.

Después del entrenamiento la RNA representando al motor de inducción fue probada para la tensión de excitación V mostrada en la Fig. 7. Este tipo de tensión no fue considerado en el proceso de entrenamiento de la red neuronal artificial. Las simulaciones del modelo de tercer orden de la máquina de inducción y de la red neuronal se muestran en la Fig. 7.

En la Fig. 8 se grafican los errores simples en porcentaje para las tensiones E_d' , E_q' y la velocidad del rotor. Analizando los resultados se observa que son bastante próximos los resultados obtenidos con el modelo de tercer orden considerando ecuaciones diferenciales, y los resultados del modelo del motor de inducción usando redes neuronales artificiales. Los errores simples de prueba, después del entrenamiento, son del orden de 1% excepto durante los primeros pasos en que son menores de 0.5%. Se puede asegurar la eficacia de la metodología propuesta basada en RNA con teacher forcing para efectos de identificación de sistemas dinámicos no lineales.

Este modelo del motor de inducción usando redes neuronales puede capturar la dinámica de esta máquina eléctrica para diversas condiciones de operación.

CONCLUSIONES

El entrenamiento de redes neuronales artificiales con back-propagation adoptando el procedimiento teacher forcing permite identificar la evolución dinámica de los sistemas dinámicos no lineales.

Los resultados de las pruebas realizadas usando los sistemas dinámicos del péndulo doble y la dinámica del motor de inducción de tercer orden, revelan que la identificación y simulación usando el método BPTF pueden ser muy eficientes principalmente durante los

primeros pasos de la identificación posterior al período de tiempo usado durante la fase de entrenamiento de la RNA.

Estos resultados indican las ventajas del teacher forcing el cual debería ser usado en las diversas áreas de control y de identificación dentro de la ingeniería eléctrica.

Las redes neuronales del tipo back-propagation y con teacher forcing son entrenadas con mayor eficacia con los métodos de segunda orden, Levenberg-Marquardt, con el cual se consigue un entrenamiento con menos épocas y con tiempos de procesamiento menores que las versiones del back-propagation y con una mayor precisión en los resultados.

AGRADECIMIENTOS

Los autores agradecen a CAPES y CNPq por su apoyo para el desarrollo de esta investigación.

REFERENCIAS

1. Chu S.R., Shoureshi R. and Tenório M., "Neural Networks for System Identification", IEEE Control Systems Magazine, pp. 31-35, April 1990.
2. Fausett L., "Fundamental of Neural Networks. Architectures, Algorithms and Applications", Prentice-Hall, 1994.
3. Haykin S., "Neural Networks: a comprehensive foundation", 2nd.ed., Prentice Hall, 1998.
4. Hertz J.A., Krogh A. and Palmer R.G., "Introduction to the Theory of Neural Computation", Addison-Wesley Pub.Co., USA, 1991.
5. Narendra K.S. and Parthasarathy K., "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Transactions on Neural Networks, Vol.1, No.1, pp. 4-27, March 1990.
6. Pearlmutter B.A., "Dynamic Recurrent Neural Networks", Carnegie Mellon University, USA, 1990.
7. El-Sharkawi M. and Niebur D., "Artificial Neural Networks with Applications to Power Systems", IEEE PES special publication 96 TP 112-0, 1996.
8. Bostanci M., Koplowitz J. and Taylor C.W., "Identification of Power System Load

- Dynamics Using Artificial Neural Networks”, IEEE Transactions on Power Systems, Vol.12, No.4, pp.1468-1473, Nov. 1997.
9. Lesieutre B.C., Sauer P.W. and Pai M.A., “Development and Comparative Study of Induction Machine Based Dynamic P,Q, Load Models”, IEEE Transactions on Power Systems, Vol.10, No.1, pp.182-191, Feb. 1995.
 10. Page G.F., Gomm J.B. and Williams D., “Application of Neural Networks to Modeling and Control”, London: Chapman and Hall Limited, UK, 1993.
 11. Kröse, B.J.A., van der Smagt, P.P. “An Introduction to Neural Networks”, Department of Computer Systems, University of Amsterdam, 1993.
 12. Von Zuben F.J. “Redes Neurais Aplicadas ao Controle de Máquina de Indução”, Tesis de Maestría, Facultad de Ingeniería Eléctrica y Computación, Campinas-SP, Brasil, Oct. 1993.
 13. Rumelhart D.E. and McClelland J.L., eds. “Parallel Distributed Processing: *Explorations in the Microstructure of Cognition*”, Vol.1, Cambridge, MA: MIT Press, 1986.