



IDENTIFICACIÓN Y CONTROL NEURONAL PARA UN MOTOR DE INDUCCIÓN LINEAL

Luis Alberto Esparza García¹ y Bernardino Castillo Toledo²

¹Universidad Autónoma de Nayarit, Ciudad de la cultura Amado Nervo S/N, ²CINVESTAV, Unidad Guadalajara, Apartado Postal 31-438, Plaza La Luna, Guadalajara², Jalisco, C.P. 45091, Mexico; e-mail: lesparza@gdl.cinvestav.mx y toledo@gdl.cinvestav.mx.

Un motor lineal es un motor eléctrico destinado a producir la fuerza en una línea recta. En este diseño, la fuerza es producida por un campo magnético en movimiento lineal que actúa sobre los conductores en el campo. Cualquier conductor, ya sea un bucle, una bobina o simplemente una pieza de metal de la placa, que se coloca en este campo tendrá las corrientes de Foucault inducida en ella creando así un campo magnético opuesto, de conformidad con la ley de Lenz. Los dos campos opuestos se repelen entre sí, creando de este modo el movimiento como el campo magnético barre a través del metal.

Muchos diseños se han propuesto para los motores lineales, separándose en dos grupos, los motores lineales de baja aceleración y alta aceleración. Los motores de baja aceleración lineal son aptos para los trenes de levitación magnética y otras aplicaciones de transporte terrestres. Los motores lineales de alta aceleración son normalmente diseñados para acelerar un objeto a una velocidad muy alta y luego soltarlo, como montañas rusas. Los motores de alta aceleración son por lo general motores de inducción lineal de CA (LIM) con una fase activa de tres bobinados en un lado del espacio de aire y una placa conductora pasiva en el otro lado. Los motores de baja aceleración, alta potencia y alta velocidad son por lo general del tipo

de motor síncrono lineal (LSM), con un devanado activo en un lado del espacio de aire y un conjunto de polos de imanes alternativos en el otro lado. Estos imanes pueden ser imanes permanentes o imanes energizados. Un ejemplo de uso de estos motores es el caso del Shanghai Transrapid.

Las redes neuronales se han establecido como una metodología adecuada para la aproximación de funciones no lineales; debido a esto son muy empleadas; por esta razón son implementadas en la identificación de sistemas no lineales para obtener modelos con dinámica muy cercana a la del sistema real (*Rovithakis and Chistodoulou, 2000*).

Los requerimientos para una red neuronal son menores en comparación con los procedimientos clásicos de identificación. Comparando ambos procedimientos desde el punto de vista de entrada y estructura, observamos que el usuario debe especificar los datos para el proceso de entrada-salida idénticamente en ambos tipos de procedimientos. En cuanto a la estructura, en los procedimientos clásicos el usuario debe especificar la característica de la estructura de la relación entre las entradas y las salidas, a veces muy complejas, mientras que para la



identificación usando las redes neuronales solo se debe especificar la topología de la red. La red neuronal no intenta asumir ninguna estructura de la relación entrada-salida, la red crea la relación de las entradas a las salidas basándose en los ejemplos proporcionados durante el entrenamiento. Lo que se le pide a la red en este caso, es que internamente identifique un modelo capaz de reproducir valores numéricos que se asemejen a los valores de algún sistema físico (*Pedro, Francisco, Miguel, Hugo, 2008, p. 80*).

Un identificador es un dispositivo que tiene la función de hacer que tanto la señal de entrada como su respuesta sean o parezcan idénticas. La función de dicho identificador en el caso que se presenta, realiza la función de copiar la respuesta del MIL, el cual es sometido a una excitación exterior, tratando al sistema a copiar como una caja negra, la cual no conocemos ni su composición ni su respuesta a la excitación exterior. El mecanismo encargado de copiar el funcionamiento del MIL, es una red neuronal entrenada con un algoritmo de entrenamiento específico. Los pesos son los estados a estimar y la salida de la red es la medición requerida. Y como referencia, para que la red calcule el error que está cometiendo, se toma la diferencia entre la salida de la red y la salida del MIL.

Una vez que termine el aprendizaje, el sistema identificador, será capaz de reproducir la respuesta, si la entrada del dispositivo identificador es la señal de entrada con la cual se ha entrenado la red. La interface con el usuario, se hará a través de una tarjeta de adquisición de datos NI-USB-6008 y el software Matlab® donde el usuario podrá seleccionar los diferentes parámetros para el entrenamiento de la red neuronal, y ver los resultados de dicho

entrenamiento, para comprobar que el entrenamiento se está realizando convenientemente.

Debido a que los sistemas dinámicos se encuentran a nuestro alrededor, las técnicas de identificación de sistemas han venido teniendo una gran relevancia en áreas muy diversas del conocimiento (Ingenierías, economía, biotecnología, etc.), donde se necesita de un modelo lo más parecido al sistema para fines de análisis, predicción, simulación, diseño y control.

Las técnicas de control actuales, requieren de modelos matemáticos cada vez más exactos para el análisis y el diseño. En muchos casos, tales modelos no pueden ser obtenidos en forma sencilla y económica a partir de las leyes que rigen cada proceso. Aquí es donde la identificación de sistemas dinámicos juega un papel importante, ya que es una herramienta capaz de proporcionar los métodos necesarios para obtener de manera relativamente sencilla los modelos matemáticos buscados con un alto grado de exactitud.

El motor de inducción lineal no cuenta con un modelo que represente su dinámica y una manera de obtenerlo es a través de la identificación.

Retomando lo anteriormente mencionado y teniendo la necesidad de obtener un modelo que represente lo más exacto posible al MIL y conociendo las propiedades de las redes neuronales recurrentes se propone diseñar una red neuronal recurrente de alto orden discreta para identificar al motor de inducción lineal, una vez obtenido esto se desarrollará una ley de control.

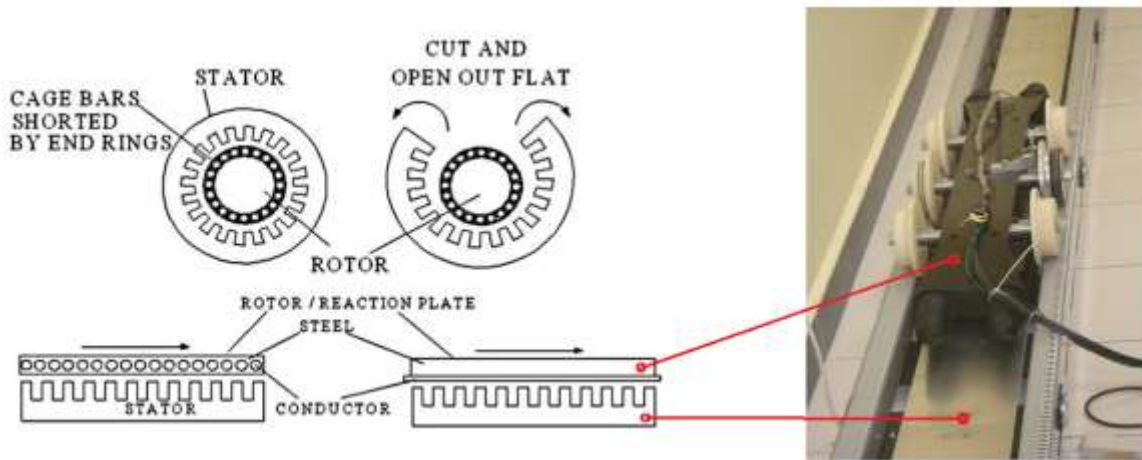


FIGURA 1. Construcción del Motor de Inducción Lineal

Red Neuronal Recurrente de alto Orden Discreta.

Considere el siguiente sistema no lineal discreto:

$$x(k+1) = F(x(k), u(k)) + D(w(k)) \quad (1)$$

donde $x(k) \in \mathcal{R}^n$ es el vector de estados de la planta, $u(k) \in \mathcal{R}^m$ es la entrada de control y $F \in \mathcal{R}^n \times \mathcal{R}^m \rightarrow \mathcal{R}^n$ es una función no lineal, $D(w(k)) \in \mathcal{R}^n$ son las perturbaciones del sistema.

Considere el problema de aproximar al sistema anterior por la siguiente red neuronal de alto orden (Rovithakis et al, 2000):

$$X_i(k+1) = w_i^{*T} z_i(x(k), u(k)) + \epsilon_{z_i}, i = 1 \dots, n \quad (2)$$

donde $X_i (i = 1, \dots, n)$ es el i -ésimo estado de la red neuronal, $x(k)$ es el vector de estado de la planta, ϵ_{z_i} es el error de aproximación acotado el cual puede ser reducido incrementando el número de pesos ajustable (Kosmatopoulos, Polycarpou, Christodolou and Inoannou,

1995). Suponga que existe un vector de pesos ideal w_i^* tal que $\|\epsilon_{z_i}\|$ puede ser minimizado sobre un conjunto compacto $\Omega_{z_i} \subset \mathcal{R}^{L_i}$. El vector de pesos ideal w_i^* es una cantidad artificial requerida para propósitos de análisis (Rovithakis et al, 2000). Se supone que este vector existe y es constante pero desconocido.

Para identificar el sistema en esta tesis, se utilizará la siguiente red neuronal de alto orden discreta, cuya estructura se muestra en la figura 18.

$$X_i(k+1) = w_i^T z_i(x(k), u(k)), \quad i = 1 \dots, n \quad (3)$$

donde $X_i (i = 1, 2 \dots, n)$ es el estado de la i -ésima neurona, $x(k)$ es el vector de estados de la planta, w_i es el vector de pesos adaptados en línea, y es un estimado para $w_i^* z_i(x(k), u(k))$ está definido como



$$z_i(x(k), u(k)) = \begin{bmatrix} z_{i_1} \\ z_{i_2} \\ \vdots \\ z_{i_{L_i}} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} \psi_{i_j}^{d_{i_j}(1)} \\ \prod_{j \in I_2} \psi_{i_j}^{d_{i_j}(2)} \\ \vdots \\ \prod_{j \in I_{L_i}} \psi_{i_j}^{d_{i_j}(L_i)} \end{bmatrix} \quad (4)$$

con L_i se refiere al número de conexiones de alto orden, $\{I_1, I_2, \dots, I_{L_i}\}$ una colección de subconjuntos no ordenados de dimensión $\{1, 2, \dots, n+m\}$, $d_{i_j}(k)$ enteros no negativos y ψ_i definido como

$$\psi_i = \begin{bmatrix} \psi_{i_1} \\ \vdots \\ \psi_{i_n} \\ \psi_{i_{n+1}} \\ \vdots \\ \psi_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(x_1(k)) \\ \vdots \\ S(x_n(k)) \\ u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \quad (5)$$

En (25) $u(k) = [u_1(k) \ u_2(k) \ \dots \ u_m(k)]^T$ es el vector de entrada a la red neuronal, y $S(\cdot)$ está dado por $S(x(k)) = \frac{\alpha}{1+e^{-\beta x(k)}} + \gamma$, $\beta > 0$ (6) con

$$S(x(k)) \in [\gamma, \alpha + \gamma]$$

Filtro de Kalman Extendido

El filtro de Kalman formulado como un sistema lineal en el espacio de estados, provee una solución al problema lineal de filtrado óptimo. Se aplica a ambientes estacionarios y no estacionarios. La solución es recursiva, de tal forma que cada actualización del estado estimado es calculado a partir del estimado anterior y del nuevo dato de entrada. Esto significa que no es necesario almacenar todos los datos anteriores de tiempo (Haykin, 2001, p. 5).

El filtro de Kalman, suponiendo un sistema dinámico lineal, estima el estado con ruido aditivo blanco, tanto en el estado como en la salida. Sin embargo, en muchas ocasiones el modelo es no lineal, por lo que se extiende el filtro de Kalman a través de un proceso de linealización. El filtro resultante es conocido como filtro de Kalman extendido (FKE) (Haykin, 2001, p. 123).

Considérese un sistema dinámico no lineal descrito por el siguiente modelo en espacio de estado

$$w(k+1) = f(k, w(k)) + \mu(k) \quad (7)$$

$$y(k) = h(k, w(k)) + v(k) \quad (8)$$

Con $w(k) \in \mathfrak{R}^n$ el vector de estado del sistema, $y(k) \in \mathfrak{R}^p$ la salida medible, con $\mu(k) \in \mathfrak{R}^n$ y $v(k) \in \mathfrak{R}^p$ ruidos independientes, gaussianos y matrices de covarianza $Q(k)$ y $R(k)$ respectivamente, $f(k, z(k))$ denota la función matricial no lineal de transición y $h(k, z(k))$ denota la función matricial de medición no lineal (Haykin, 2001, p. 124).

La idea del FKE consiste en linealizar el modelo en espacio de estados de la ecuación (7) y (8) a cada instante de tiempo alrededor del estado estimado más reciente (Haykin, 2001).

Los cálculos de las matrices $F_{k+1,k}$ y $H(k)$ se realizan como sigue:

$$F_{k+1,k} = \frac{\partial f(k, w(k))}{\partial w} \quad (9)$$

$$H(k) = \frac{\partial h(k, w(k))}{\partial w} \quad (10)$$

Cuando han sido evaluadas las matrices $F_{k+1,k}$ y $H(k)$ se aplican las ecuaciones del FKE (Haykin, 2001, p. 128).

Para el entrenamiento de redes neuronales basadas en el filtro de Kalman, los pesos son los estados a estimar y la salida de la



red es la medición requerida. El error entre la red neuronal y la salida de la planta es considerada como ruido aditivo blanco. Debido a que la red neuronal es no lineal se requiere un FKE (Haykin, 2001), (Alanís, 2004), (Alanís, 2006).

El objetivo del entrenamiento del Filtro de Kalman Extendido es encontrar los pesos óptimos que minimicen el error de predicción. El FKE se describe por

$$w_i(k+1) = w_i(k) + \eta_i K_i(k) e_i(k) \quad (11)$$

$$K_i(k) = P_i(k) H_i(k) M_i(k) \quad i = 1, \dots, n \quad (12)$$

$$P_i(k+1) = P_i(k) - K_i(k) H_i^T(k) P_i(k) + Q_i(k) \quad (13)$$

con

$$M_i(k) = [R_i(k) + H_i^T(k) P_i(k) H_i(k)]^{-1} \quad (14)$$

$$e_i(k) = x_i(k) - X_i(k) \quad (15)$$

donde $e_i(k) \in \mathfrak{R}$ es el respectivo error de identificación, $P_i(k) \in \mathfrak{R}^{L_i \times L_i}$ es la matriz de covarianza del error de estimación al paso k , $w_i(k) \in \mathfrak{R}^{L_i}$ es el vector de pesos (estado) adaptados en línea, $x_i(k)$ es el i -ésimo estado de la planta, $X_i(k)$ es el i -ésimo estado de la red neuronal, $K_i(k) \in \mathfrak{R}^{L_i}$ es el vector de ganancias de Kalman, $Q_i(k) \in \mathfrak{R}^{L_i \times L_i}$ es la matriz de covarianza del ruido de estimación, $R_i(k) \in \mathfrak{R}$ es la covarianza del ruido de medición y $H_i(k) \in \mathfrak{R}^{L_i}$ es un vector, y se obtiene como la derivada del estado de la red neuronal $X_i(k) \in \mathfrak{R}$ con respecto a los pesos de la red neuronal $w_{ij}(k)$, y está dado por:

$$H_{ij} = \left[\frac{\partial X_i(k)}{\partial w_{ij}(k)} \right]^T \quad (16)$$

donde $i = 1, \dots, n$ y $j = 1, \dots, L_i$. Usualmente P_i y Q_i son inicializados como

matrices diagonales, con entradas $P_i(0)$ y $Q_i(0)$, respectivamente. Es importante notar que

$$\begin{aligned} \|H(k)\| &\leq \bar{H} \\ \|K(k)\| &\leq \bar{K} \\ \|P(k)\| &\leq \bar{P} \end{aligned} \quad (17)$$

La identificación entre la planta y la red neuronal se establece en el siguiente Teorema (para detalles ver Alanís, 2007).

Teorema. La RHONN (23) entrenada con el filtro de Kalman extendido para identificar el sistema no lineal (1) asegura que el error de identificación (15) es semiglobalmente uniformemente últimamente acotado (SGUUA), y los pesos de la RHONN permanecen acotados.

Modelo Matemático

Se refiere a un servomecanismo con un motor lineal. El modelo del motor tiene que ser establecido en la base de las siguientes ecuaciones describiendo la dependencia eléctrica y física del motor. Podemos decir que la forma es idéntica con el modelo estándar de la maquina CD (Eliska, 2009, p.156).

Ecuación de voltaje del motor:

$$U_0 = R * i + L \left(\frac{di}{dt} \right) + U_i \quad (18)$$

Donde:

U_0 : es el voltaje de la entrada

R : es la resistencia

L : es la inductancia

U_i : es el voltaje inducido

i : es la corriente de armadura

Ecuación de movimiento:

$$F_H = F_Z + m \left(\frac{dv}{dt} \right) \quad (19)$$

Donde:

F_H : es la fuerza conducida

F_Z : es la fuerza de carga

M : es la masa total de la armadura y carga

De las ecuaciones previas el modelo del motor tiene que ser establecido:



$$F_M(s) = \frac{v(s)}{U_0(s)} = \frac{\left(\frac{1}{K_1}\right)}{\left(\frac{s^2}{\Omega_0^2} + \left(\frac{2\xi s}{\Omega_0}\right) + 1\right)} \quad (20)$$

Donde:
 v: es la velocidad
 U₀: es el voltaje de la entrada
 K₁: es la constante del motor
 ξ: es el factor de amortiguamiento
 Ω₀: frecuencia natural no amortiguada

Identificador Neuronal

Un identificador con una red neuronal es aplicado a los datos adquiridos de LIM; se supone que todas las variables de estado son medibles. Para la identificación del LIM se utiliza una red neuronal recurrente de alto orden discreta (DRHONN por su nombre en inglés) en su representación serie- paralelo (ver figura 3) como:

$$\hat{p}_0(k+1) = w_1 S(p_0(k)) + w_2 S(v_0(k)) + w_3 u(k) \quad (21)$$

$$\hat{v}_0(k+1) = w_4 S(p_0(k)) + w_5 S(v_0(k)) + w_6 u(k) \quad (22)$$

donde $\hat{p}_0(k)$ identifica la posición, $\hat{v}_0(k)$ identifica la velocidad lineal y los pesos $w_{ij}(k)$ son actualizados por (11,12 y13).

El entrenamiento es realizado con el filtro de Kalman extendido, para la identificación se usa una configuración serie-paralelo como se muestra en la figura 3.

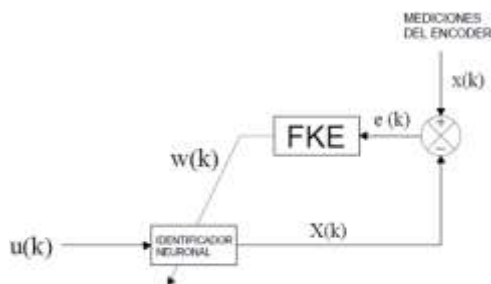


Figura 3. Esquema de identificación.

Todos los estados de la red neuronal son inicializados en cero. Las matrices de covarianza son inicializadas diagonales, como P(0)=1100, R(0)=900, Q(0)=100 para el desplazamiento, P(0)=10, R(0)=1, Q(0)=0.1 para la velocidad del LIM. Como se puede notar se cumple la condición del filtro de Kalman requiere que P > R > Q.

Diseño del Controlador

La variable a controlar es la velocidad $\hat{v}_0(k)$. El objetivo de control es el seguimiento de una referencia variante en el tiempo para la velocidad lineal. Se asume que todas las variables de estado son medibles.

El esquema que se utiliza se muestra en la figura 4.

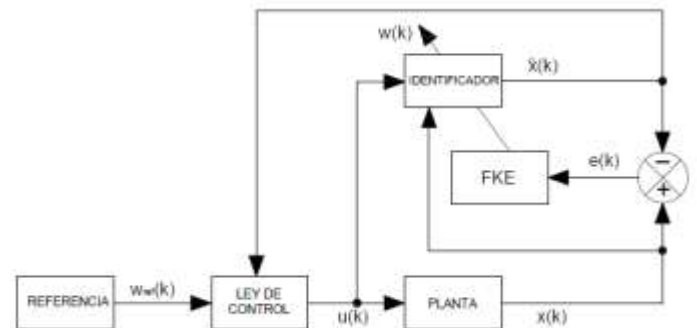


Figura 4. Esquema de control del LIM basado en un identificador.

Ley de control. Para alcanzar los objetivos de control de velocidad, se define el vector de error de seguimiento:

$$e(k) = \hat{X}(k) - W_{ref}(k) \quad (23)$$

donde e(k) es el error entre la salida de la red neuronal y la referencia deseada, respectivamente.



Evaluando un paso adelante

$$\begin{aligned} e(k+1) &= \widehat{v}_0(k+1) - W_{ref}(k) \quad (24) \\ e(k+1) &= w_4 S(p_0(k)) + w_5 S(v_0(k)) \\ &\quad + w_6 u(k) - W_{ref}(k) \\ &= ke(k) \end{aligned}$$

De aquí se obtiene la ley de control

$$\begin{aligned} u(k) &= w_6^{-1} \left(ke(k) - w_4 S(p_0(k)) \right. \\ &\quad \left. - w_5 S(v_0(k)) \right. \\ &\quad \left. + W_{ref}(k) \right) \quad (25) \end{aligned}$$

donde $0.9 < k < 1$.

los datos de entrada-salida de éste. El modelo propuesto que se utilizó para la identificación es el de (21) y (22). La estructura utilizada es la propuesta en la figura 3. La señal que se utilizó es una combinación de los comandos pulso generador y repetidor de secuencia, estas dos señales se suman antes de aplicarlo a LIM. Los parámetros de las señales son: amplitud = 1, periodo = 2, ancho de pulso = 50, retraso de fase = 0; valores del vector de salida = [1 0.8 0.9 0.6 0.7 0.5 0.8], tiempo de muestreo = -1, respectivamente. Los resultados de la simulación son los siguientes:

Resultados de Simulaciones

Las simulaciones de la identificación del LIM se realizaron en Simulink®, utilizando

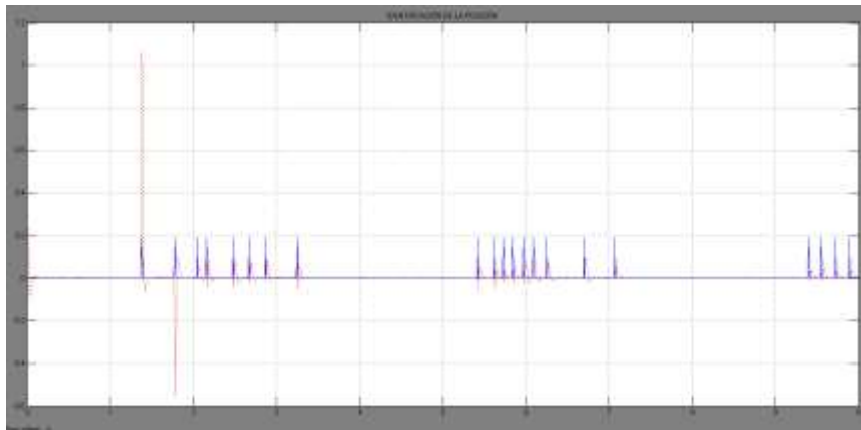


FIGURA 5. Simulación de la identificación de la posición.

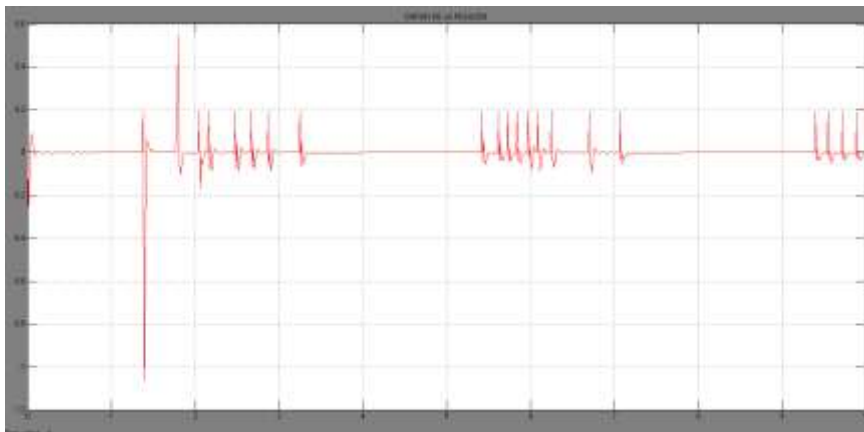


FIGURA 6. Simulación del error en la identificación de la posición

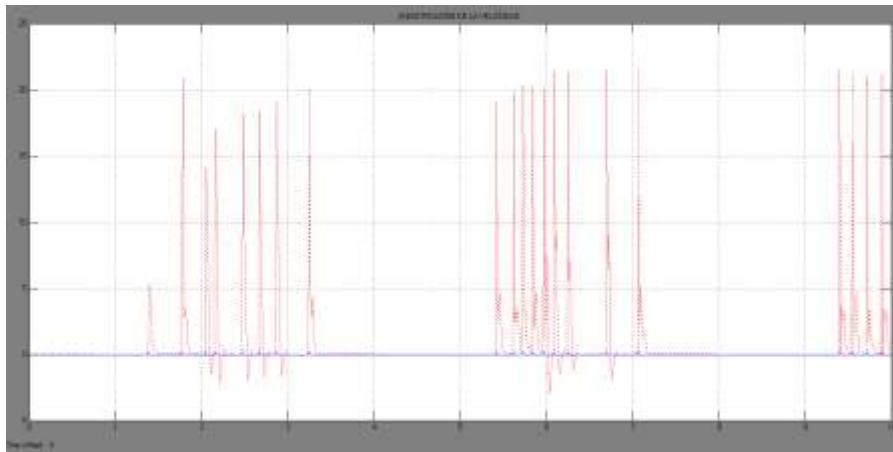


FIGURA 7. Simulación de la identificación de la velocidad

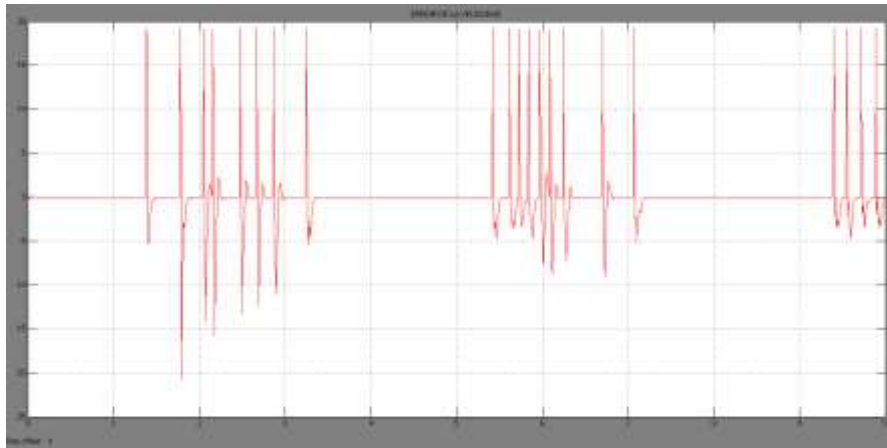


FIGURA 8. Simulación del error en la identificación de la velocidad.

Para la validación de la identificación neuronal, al LIM se le aplica una entrada escalón con los siguientes parámetros: tiempo de paso = 5, valor inicial = 0, valor

final =1 y tiempo de muestreo = 0.01. Se obtienen nuevos datos los cuales son aplicados al identificador dando los siguientes resultados:

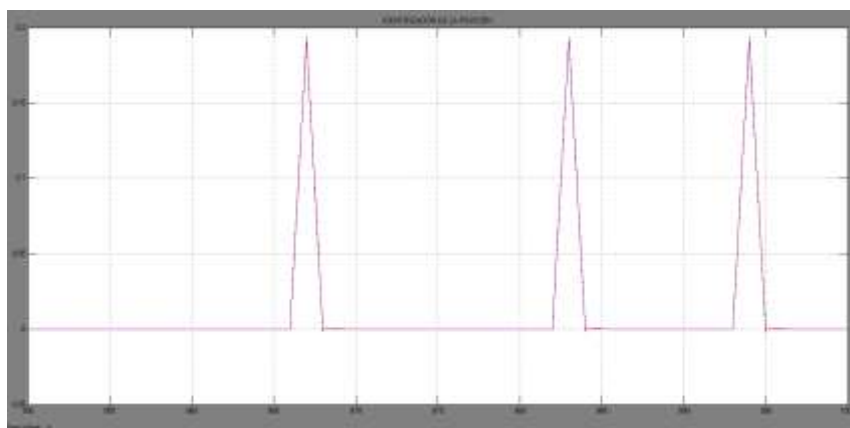


FIGURA 9. Simulación de la identificación de la posición.

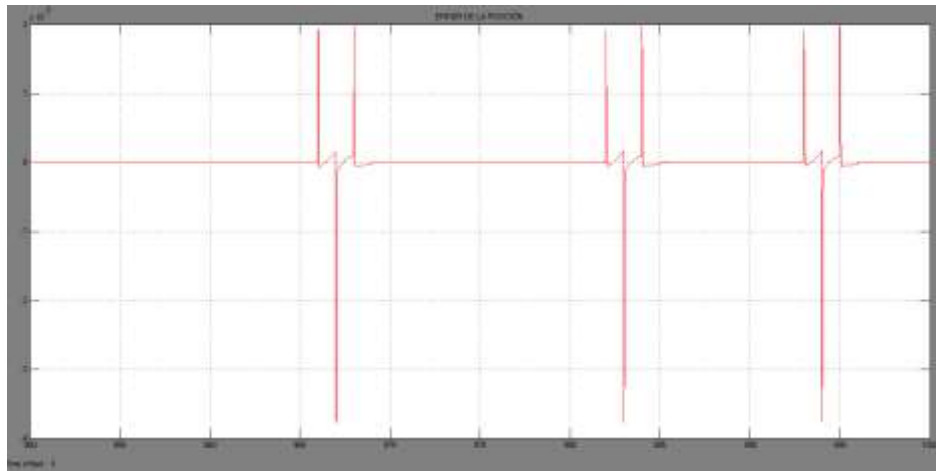


FIGURA 10. Simulación del error en la identificación de la posición

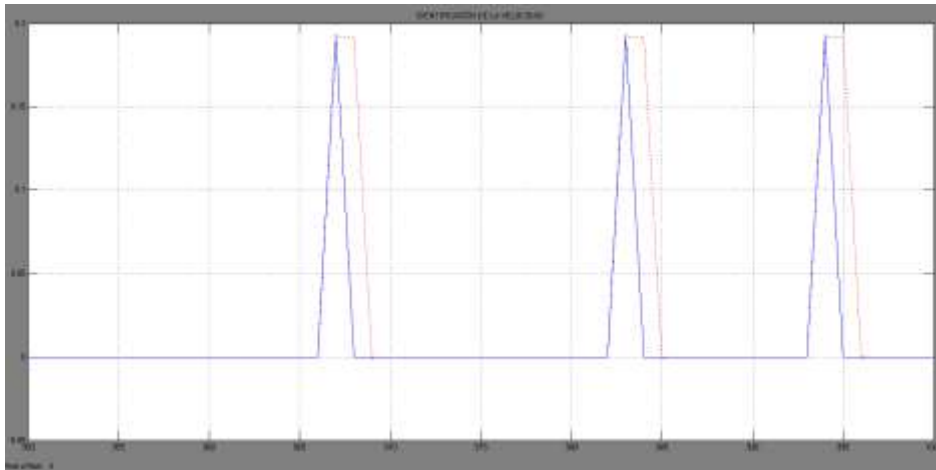


FIGURA 11. Simulación de la identificación de la velocidad

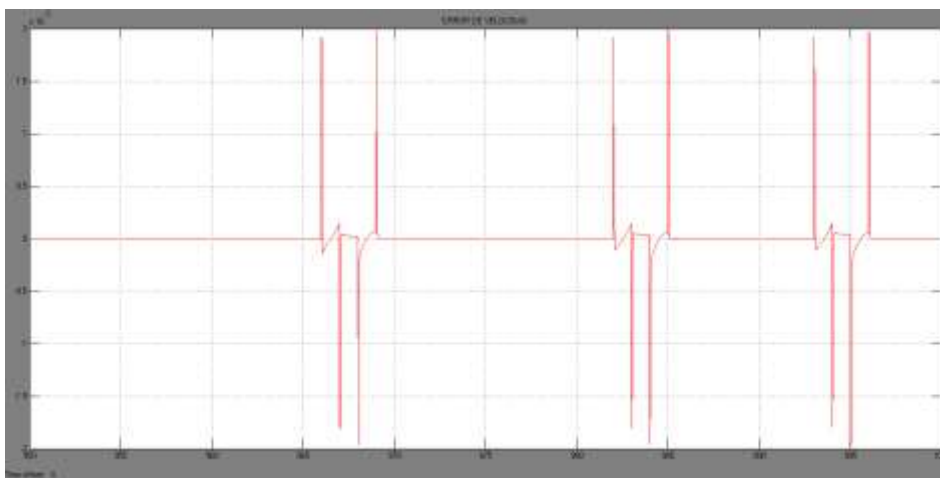


FIGURA 12. Simulación del error en la identificación de la velocidad



Para evaluar el comportamiento del controlador diseñado y la respuesta del sistema en lazo cerrado, se realizaron simulaciones para un Motor de Inducción Lineal.

Los Parámetros del motor lineal son:

- Masa (m) = 45 g
- Frecuencia = 100 Hz
- Longitud = 10.5 cm
- Tiempo (t) = 176 ms
- Voltaje (V) = 6.2 volts
- Inductancia (L) = 41 mH

- Resistencia (R) = 74 ohm
- Constante del motor = 10 v/ms^{-1}
- Velocidad (v) = 0.6 m/s
- Las condiciones de simulación son:
 - Tiempo de simulación: 10 segundos.
 - Tiempo de muestreo: 1 milisegundos

El control es aplicado hasta un segundo para permitir al identificador aprender la dinámica de la planta.

La referencia para la velocidad lineal es una señal constante con valor de 12 y tiempo de muestreo de 1.

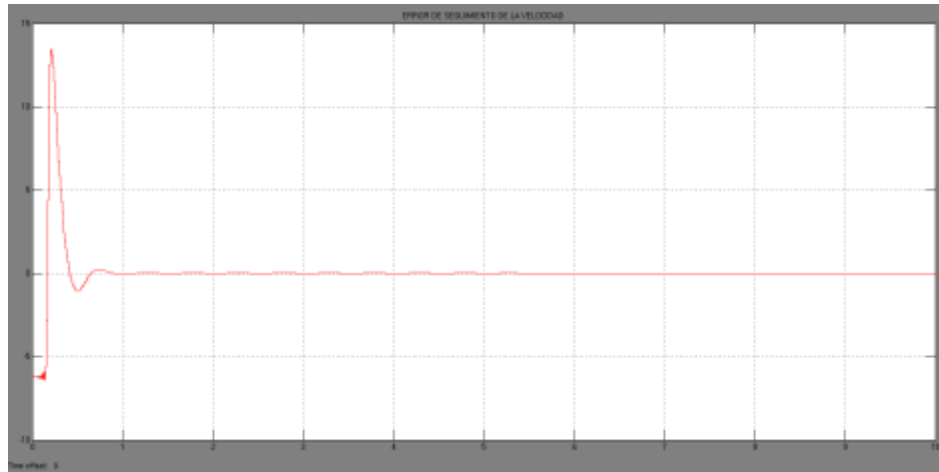


FIGURA 13. Simulación del error de seguimiento de la velocidad

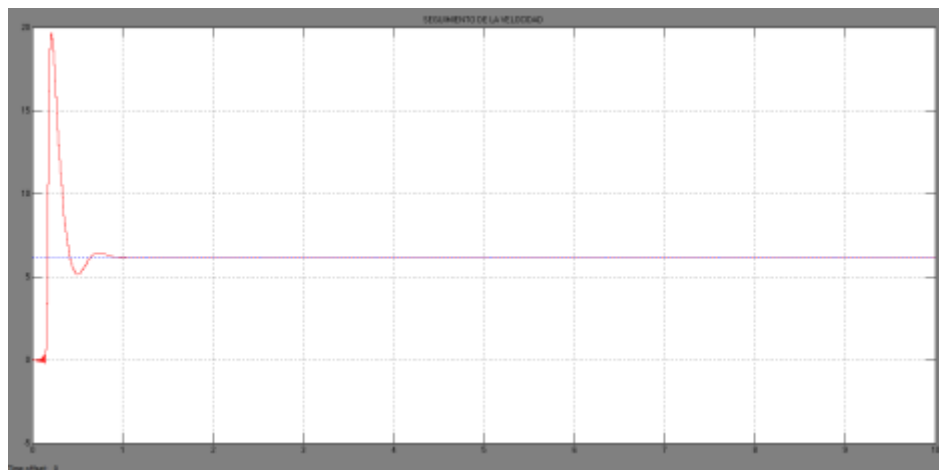


FIGURA 14. Simulación del seguimiento de la velocidad



Conclusiones

En esta investigación se desarrolló un sistema de control neuronal por retroalimentación de estados para el seguimiento de trayectoria deseada para el motor de inducción lineal.

Un identificador neuronal se diseñó para estimar el modelo matemático del motor de inducción lineal.

En ambos casos los resultados de simulación ilustran la efectividad y robustez del esquema de control propuesto.

Referencias

- A. Y. Alanís. Entrenamiento de redes neuronales con el filtro de kalman. Tesis de Maestría, Centro de Investigación y Estudios Avanzados del IPN, Guadalajara, México, 2004.
- A. Y. Alanís, E. N. Sánchez and A. G. Loukianov. "Discrete-time recurrent neural induction motor using Kalman learning". Proceedings of the International Joint Conference on Neural Network, July 2006.
- A. Y. Alanís. Discrete-time Neural Control: Application to Induction Motors. Tesis de Doctorado, Centro de Investigación y Estudios Avanzados del IPN, Guadalajara, México, 2007.
- Eliska Tejklová. "Realization of Servomechanism with Linear Motor in Matlab–Simulink Environment"; Department of applied Cybernetics, Faculty of Mechanical Engineering, Technical University of Liberec, Studenská 2, 461 17 Liberec, Czech Republic. June 9 – 12, 2009. P 155 - 157.
- G. A. Rovithakis and M. A. Chistodoulou. Adaptive Control with Recurrent High-Order Neural Networks. Springer Verlag, Berlin, Germany, 2000.
- Pedro Arafet Padilla, Francisco Chang Mumañ, Miguel Torres Alberto, Hugo Domínguez Abreu; "MÉTODOS DE IDENTIFICACIÓN DINÁMICA", Facultad de Ingeniería Eléctrica. Universidad de Oriente; Junio 2008.
- S. Haykin. Kalman Filtering and Neural Netorks. John Wiley and Sons, N.Y., U.S.A., 2001.
- E. B. Kosmatopoulos, M. M. Polycarpou and M. A. Christodolou and P. A. Inoannou. "High-order neural network structures for identification of dynamical systems". Vol. 6, No. 2, pp. 422-431, IEEE Trans. on Neural Networks, 1995.