

NIS: UNA HERRAMIENTA COMPUTACIONAL PARA LA IDENTIFICACIÓN DE SISTEMAS DINÁMICOS EN LAZO CERRADO EMPLEANDO REDES NEURONALES ARTIFICIALES

Zambrano Alejandro¹, Collazo Víctor², Troncone Numan², Rodríguez Jesús²

(Recibido febrero 2011, Aceptado junio 2012)

¹Sidor

²CORPOELEC

arzambrano@sidor.com, ntroncone@corpoelec.gob.ve

Resumen: El presente trabajo muestra el diseño e implementación de una herramienta computacional para identificación de sistemas dinámicos mediante aplicación de la tecnología de las Redes Neuronales Artificiales. El Neuro-Identificador de Sistemas (NIS), presenta dos componentes principales: un hardware de estímulo y adquisición de señales, y una interfaz humano-máquina, que ha sido desarrollada bajo arquitectura PC empleando la herramienta MATLAB®. El NIS permite obtener un modelo aproximado que se ajusta a la dinámica de la planta real, que posteriormente puede ser evaluado en lazo abierto. El NIS genera automáticamente funciones de transferencia de Primer Orden Más Tiempo Muerto (P.O.M.T.M.), y funciones de transferencia de segundo orden si la respuesta del sistema es sub-amortiguada. El sistema ha sido aplicado en la identificación de modelos bajo entorno de simulación y de circuitos eléctricos de primer y segundo orden. Los modelos obtenidos satisfacen las hipótesis, obteniéndose un factor de correlación cuadrática multivariable de 0.9670 en la estimación de un sistema de primer orden.

Palabras clave: Identificación de Sistemas Dinámicos/ Redes Neuronales Artificiales/ Interfaz Hombre-Máquina.

NIS: A COMPUTER TOOL FOR CLOSED-LOOP DYNAMICAL SYSTEM IDENTIFICATION THROUGH ARTIFICIAL NEURAL NETWORKS

Abstract: This paper shows the design and implementation of a computational tool for dynamical system identification by applying the technology of Artificial Neural Networks. The Systems Neuro-Identifier (NIS), has two main components: a stimulation and data acquisition hardware, and a human-machine interface. The NIS automatically generates first order transfer functions plus dead time (F.O.P.D.T.) and second order transfer functions if the system response is sub-damped. The tool has been applied in the identification of models under simulation environment and electrical circuits of first and second order. The models obtained matched the hypotheses regarding the dynamics of the plants identified, obtaining a quadratic multivariable correlation factor of 0.9670 in the first order identified systems.

Keywords: Dynamical System Identification/ Artificial Neural Networks/ Human-Machine Interface.

I. INTRODUCCIÓN Y ANTECEDENTES

Las redes neuronales artificiales constituyen una excelente herramienta para el aprendizaje de relaciones complejas a partir de un conjunto de ejemplos. Gracias a esto se ha generado un gran interés en la utilización de redes neuronales en la identificación de sistemas dinámicos con no linealidades desconocidas. Además, debido a sus capacidades de aproximación, así como a sus inherentes características de adaptabilidad, las redes neuronales artificiales presentan una importante alternativa en el modelado de sistemas dinámicos no lineales [1]. El objetivo principal del presente trabajo es mostrar el diseño e implementación de una herramienta

computacional denominada Neuro-Identificador de Sistemas (NIS), que permita estimar la función de transferencia ó identificación, mediante aplicación de la tecnología de redes neuronales artificiales, de sistemas lineales invariantes en el tiempo (LTI) y sistemas con no linealidades desconocidas, de una sola entrada y una sola salida (SISO) que se encuentren en un lazo de control realimentado.

La motivación del proyecto es la implementación de un software y su conexión a una interfaz de hardware para la adquisición de señales del sistema dinámico a ser identificado, que permita la obtención de modelos matemáticos de sistemas electro-mecánicos que ayuden

en la detección de desajustes y permitan la compensación de controladores de posición de paletas directrices en el esquema de gobernación de velocidad en máquinas hidrogeneradoras de centrales hidroeléctricas. En este caso, mantener la producción de energía eléctrica es fundamental, por lo que no se pueden desconectar este tipo de controladores y aplicar técnicas tradicionales de modelado, como la identificación empírica realizando pruebas en el sistema en lazo abierto y ajustando los parámetros de la respuesta temporal a una función de transferencia [2]. El desgaste mecánico en este tipo de máquinas ocasiona cambios importantes en la función de transferencia que, en algunos casos, pueden ser no linealidades que afectan el desempeño de los compensadores PI y PID presentes en el esquema de regulación.

Desde el punto de vista de la teoría de control, según [3], un sistema dinámico o proceso está formado por un conjunto de elementos relacionados entre sí que ofrecen señales de salida en función de señales o datos de entrada. Es importante resaltar el hecho de que no es necesario conocer el funcionamiento interno, o cómo actúan entre sí los diversos elementos, para caracterizar el sistema. Para ello, sólo se precisa conocer la relación que existe entre la entrada y la salida del proceso que realiza el mismo (principio de caja negra). El aspecto más importante de un sistema es el conocimiento de su dinámica, es decir, cómo se comporta la señal de salida frente a una variación de la señal de entrada. Un conocimiento preciso de la relación entrada-salida permite predecir la respuesta del sistema y seleccionar la acción de control adecuada para mejorarla. En sentido práctico, el conocimiento de dicha relación permite además, detectar posibles malfuncionamientos producto de desgastes o averías, en el caso de sistemas mecánicos o electrónicos, a través del análisis y comparación de las respuestas del sistema en diferentes instantes en el tiempo de vida útil de dichos sistemas.

Diversos avances se han suscitado en materia de caracterización y obtención de modelos que representan la dinámica y la relación entrada-salida de sistemas. El modelado empírico de procesos contempla análisis en el dominio temporal y en el dominio de la frecuencia, que se realizan utilizando datos de entrada y salida obtenidos experimentalmente a partir de la excitación del proceso que se desea identificar bajo una condición nominal de operación. En este sentido, [4] constituye el antecedente más representativo de la investigación que se documenta en estas páginas, ya que trata la identificación de un sistema en la empresa venezolana CORPOELEC (anteriormente EDELCA): el sistema de gobernación de una unidad hidrogeneradora, que según el autor "... el sistema completo es bastante complejo, con varios lazos de retroalimentación...". Pese a ésta y otras dificultades, la reducción de los elementos que conforman dicho sistema a una representación en diagrama de bloques resumida, que contempla al proceso como una relación entrada-salida, permitió simplificar la identificación.

Mediante ensayos escalón realizados bajo velocidad nominal, y con aplicación de perturbaciones inferiores al -3%, el autor logra obtener un modelo aproximado del sistema, mediante la implementación directa de un algoritmo denominado por el autor "método desarrollado por la USB". La obtención del modelo matemático permitió, entonces, re-entonar el controlador de gobernación de velocidad de la unidad hidrogeneradora, que en su oportunidad, para el año 2007, presentaba fuertes oscilaciones que comprometían la estabilidad del sistema.

Pese al éxito de la solución vista en [4], aplicada a una unidad hidrogeneradora, el método utilizado para la obtención del modelo matemático debió adaptarse a las particularidades de sistema en el momento, y no se contaba con una herramienta computacional de propósito específico para tal fin. Una de las principales razones por la cual se realiza modelado empírico de procesos y sistemas dinámicos es el hecho de que la obtención de un modelo fundamental, basado en principios físicos o químicos (según sea la naturaleza del proceso) resulta impráctico en la realidad, por lo que se acude a la obtención de parámetros que representen el comportamiento del sistema y se ajusten a una ecuación matemática. En otros casos, el modelado debe aplicarse cuando se desconoce total o parcialmente la ecuación del proceso y se requiere su obtención para bien conocer la relación entre la(s) variable(s) del proceso, o para entonar o adaptar controladores.

En el contexto que motiva la presente investigación, debido a que sólo se pueden realizar mediciones de la entrada y la salida del sistema a identificar, y con la restricción en lazo cerrado, se propone la aplicación de redes neuronales para caracterizar la respuesta de sistemas dinámicos LTI, a través de una herramienta diseñada específicamente para tal fin. La aplicación de redes neuronales además, permitirá en el futuro la realización de pruebas para la identificación de sistemas no lineales [5] e identificación en línea.

Las redes neuronales artificiales son un paradigma de procesamiento inspirado en la forma en que funciona el sistema nervioso de los humanos, y está caracterizado por su capacidad de generalización, robustez y tolerancia a fallos [6]. La tecnología de las redes neuronales ha sido utilizada en problemas de generalización o aproximación de funciones de complejidad arbitraria, por lo que el diseño propuesto se vale de esta característica para representar la respuesta de un sistema cualquiera de una entrada y una salida, mediante el entrenamiento de una red neuronal tipo multicapa alimentada hacia adelante, a través del algoritmo Backpropagation [7].

Es notable destacar a [8]-[12], ya que presentan de manera formal la aplicación de redes neuronales alimentadas hacia adelante (feedforward), con memoria en la entrada (tapped delay) entrenadas fuera de línea mediante un algoritmo de retropropagación, como identificadores de sistemas dinámicos. Cabe resaltar que estos enfoques no

permiten la identificación de sistemas con parámetros variables en el tiempo debido a que la red es incapaz de ajustar sus pesos en línea. En esos casos, [13] y [14] proponen la utilización de redes neuronales de base radial (RBF) dinámicas, es decir, de tamaño variable que puede ser entrenada en línea e implementarse en tiempo real. En [15], se exponen algunas consideraciones no formales, basadas en simulaciones de sistemas no lineales, sobre la selección de las funciones de transferencia de las neuronas que conforman la RNA empleada en la identificación de los casos mostrados en dicho trabajo, que permiten una estimación de modelos con menor error. En estos trabajos no se muestran las realizaciones prácticas de los experimentos, ni su aplicación en la identificación de sistemas en lazo cerrado, sino que presentan enfoques más teóricos.

La herramienta desarrollada y presentada en este trabajo se vale del carácter adaptativo de las redes neuronales artificiales para almacenar conocimiento relativo a la dinámica del sistema a ser identificado. El NIS presenta dos componentes principales: un hardware de estímulo y adquisición de señales, y un software que ha sido desarrollado bajo arquitectura PC empleando las herramientas MATLAB/Simulink®, Neural Networks Toolbox™ y Data Acquisition Toolbox™. La herramienta permite obtener un modelo aproximado de primer orden más tiempo muerto o uno de segundo orden, que posteriormente puede ser evaluado en lazo abierto empleando cualquier herramienta computacional utilizada en el análisis de sistemas de control. Mediante el ajuste de pocos parámetros, el NIS permite identificar diversos sistemas LTI cuya función de transferencia es desconocida. En este artículo se muestran los resultados obtenidos en la fase de diseño y puesta en operación de la herramienta, aplicada a la identificación en entorno de simulación y de circuitos eléctricos utilizados para validar su funcionamiento.

El trabajo está estructurado de la siguiente manera: en la sección 2 se presenta todo el desarrollo de la investigación: los fundamentos teóricos de la identificación de sistemas y de las redes neuronales artificiales con entrenamiento supervisado, el método de identificación seleccionado y el diseño e implementación del NIS. En la sección 3 se presentan los resultados y discusión del funcionamiento del NIS en la identificación de sistemas dinámicos. En la sección 4 se presentan las conclusiones y en la sección 5 las referencias bibliográficas.

II. DESARROLLO

1. Identificación de Sistemas

La mayoría de los sistemas dinámicos pueden ser representados en forma discreta por un conjunto de ecuaciones [16]:

$$\begin{aligned} x_{(k+1)} &= \Gamma[x_{(k)}, u_{(k)}] \\ y_{(k)} &= \Psi[x_{(k)}] \end{aligned} \quad (1)$$

Donde x , u e y son secuencias de tiempo discreto. Cuando las funciones Γ y Ψ son desconocidas, estamos ante un problema de identificación.

Si el sistema descrito en (1) es lineal e invariante en el tiempo, las ecuaciones que describen al sistema se pueden expresar como:

$$\begin{aligned} x_{(k+1)} &= A.x_{(k)} + B.u_{(k)} \\ y_{(k)} &= C.x_{(k)} \end{aligned} \quad (2)$$

En este caso A, B y C son matrices y el sistema puede ser parametrizado por el conjunto (A, B y C). Para este tipo de sistemas, si el orden es conocido, se puede escoger la estructura del modelo y resulta un problema de estimación de parámetros.

En general un sistema no lineal descrito por (1) puede ser representado en términos de entrada-salida de la siguiente manera:

$$\begin{aligned} y_{(k+1)} &= f[y_{(k)}, y_{(k-1)}, \dots, y_{(k-n+1)}; \\ &u_{(k)}, u_{(k-1)}, \dots, u_{(k-m+1)}] \end{aligned} \quad (3)$$

El problema de la identificación consiste en seleccionar un modelo apropiado y ajustar sus parámetros de acuerdo a alguna ley adaptativa de forma que la respuesta del modelo ante una señal de entrada, o un conjunto de señales de entrada, se aproxime a la respuesta del sistema real ante esa misma entrada.

Específicamente, en la investigación se ha considerado obtener un modelo que represente la dinámica del sistema real, mediante redes neuronales artificiales, para luego analizar su curva de reacción y luego aplicar uno de los dos métodos de modelado empírico de procesos en el dominio temporal: a) modelado basado en la curva de reacción del proceso aproximando a una función de transferencia de P.O.M.T.M., o b) modelado basado en la curva de reacción del proceso aproximando a una función de transferencia de segundo orden cuando el sistema a modelar es sub-amortiguada. Tales métodos han sido seleccionados en función de la gran cantidad de sistemas reales cuya respuesta puede ser generalizada a modelos P.O.M.T.M. o de Segundo Orden. La entrada de prueba utilizada en ambos casos es el escalón, de allí que los ensayos que se hacen se les conocen como Pruebas de Cambio de Escalón [2].

2. Modelado Empírico de Procesos aproximándolos a una Función de Transferencia de Primer Orden Más Tiempo Muerto (P.O.M.T.M.)

La función de transferencia general utilizada en la estimación del modelo de primer orden más tiempo muerto, para este caso, está representada por [2]:

$$G_{(s)} = \frac{Ke^{-t_0s}}{\tau s + 1} \quad (4)$$

Donde:

K la ganancia de estado estable del proceso.

t_0 el tiempo muerto efectivo del proceso.

τ la constante de tiempo efectiva del proceso.

Las ecuaciones de interés, que han sido codificadas e incorporadas a la herramienta desarrollada en el proyecto de investigación, son las siguientes:

El valor de estado estable será:

$$y_{ss} = KA \quad (5)$$

De (5) obtenemos:

$$K = \frac{Y_{ss}}{A} \quad (6)$$

El método propone que t_0 y τ sean seleccionados de tal manera que la respuesta real y del modelo ante iguales condiciones de excitación coincidan en dos puntos en la región de máxima pendiente de su curva de reacción. Los dos puntos son:

$$t_1 = t_0 + \frac{\tau}{3} \quad (7)$$

$$t_2 = t_0 + \tau \quad (8)$$

Para localizar dichos puntos se utiliza la ecuación, resultado de la aplicación del método de fracciones parciales y la inversa de la transformada de Laplace a la función de transferencia de un sistema de primer orden:

$$c(t) = KA \left[1 - e^{-\frac{(t-t_0)}{\tau}} \right] u(t-t_0) \quad \forall \quad t > t_0 \quad (9)$$

Cuando $t = t_0 + \tau/3$ se tiene que:

$$y\left(t_1 = t_0 + \frac{\tau}{3}\right) = KA \left[1 - e^{-\frac{1}{3}} \right] = 0.283y_{ss} = 28.3\% y_{ss} \quad (10)$$

Y cuando $t = t_0 + \tau$ se tiene que:

$$y(t_2 = t_0 + \tau) = KA \left[1 - e^{-1} \right] = 0.632y_{ss} = 63.2\% y_{ss} \quad (11)$$

De (7) y (8), finalmente t_0 y τ se expresan como:

$$t_0 = t_2 - \tau \quad (12)$$

$$\tau = \left(\frac{3}{2}\right)(t_2 - t_1) \quad (13)$$

Si bien es cierto que esta aproximación es válida para sistemas en tiempo continuo, al muestrear las señales de entrada y salida del sistema a identificar bajo un criterio de sobremuestreo, y luego de un postprocesamiento o filtrado anti-aliasing, la respuesta del sistema en tiempo discreto puede aproximarse a la respuesta de un sistema de tiempo continuo, y así aplicarse la formulación antes descrita.

La Figura 1 permite explicar el método de modelado empírico expuesto:

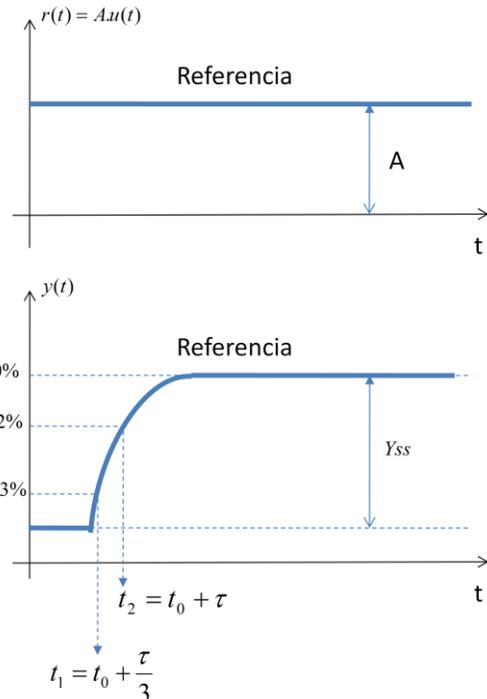


Figura 1. Gráfico del método de modelo empírico de sistemas P.O.M.T.M.

3. Redes Neuronales Artificiales con Entrenamiento Supervisado

El cerebro humano contiene aproximadamente 10^{11} elementos intensamente conectados llamados neuronas [17]. Para efectos de la obtención de su modelo artificial, estas neuronas tienen tres componentes principales: las dendritas, el cuerpo o soma de la neurona y el axón. Las dendritas son redes receptoras similares a un árbol de fibras nerviosas que transportan señales eléctricas a la parte interior del cuerpo de la neurona. El cuerpo de la célula suma y umbraliza estas nuevas señales. El axón es una única fibra larga que transporta la señal desde el cuerpo de la célula hacia otras neuronas. El punto de contacto entre un axón de una célula y una dendrita de otra célula es llamado sinapsis. El arreglo de las neuronas y las fuerzas de las sinapsis individuales están determinados por un complejo proceso químico, que establece la función de la red neuronal.

Las redes neuronales artificiales son redes interconectadas, que operan en paralelo mediante procesadores simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real, del mismo modo que lo hace el sistema nervioso central. Las redes neuronales responden a un modelo biológico de interconexión entre elementos de procesamiento denominados neuronas artificiales. El aprendizaje en los seres vivos, en particular los animales, se realiza por ensayo y error, mediante ejemplos o demostraciones. Las redes neuronales biológicas ajustan dinámicamente parámetros internos (pesos y tendencias) que rigen la representación de la información o conocimiento, adoptando la capacidad de generalizar respuestas ante eventos nunca antes suscitados. De manera análoga, se representan estos procesos, mediante algoritmos o circuitos electrónicos.

Mediante la herramienta NNT de MATLAB [18], se han implementado las funciones que permiten la parametrización de las redes neuronales aplicadas en la identificación del sistema, y por lo tanto no es objetivo del trabajo, codificar las ecuaciones relativas a la arquitectura de RNA.

4. Método de Identificación empleando Redes Neuronales Artificiales

El proceso de identificación de sistemas mediante aplicación de redes neuronales considerado en el desarrollo de la aplicación, según [16], responde al diagrama mostrado en la Figura 2.

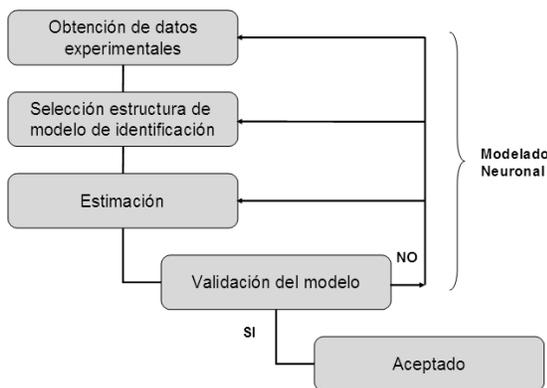


Figura 2. Fases del modelado neuronal.

Se hará especial hincapié en la selección de la estructura de modelo de identificación, en la estimación y en la validación.

El modelado empírico de procesos se fundamenta, esencialmente, en el ajuste de funciones de transferencia generales, de primer o segundo orden, mediante un conjunto de operaciones realizadas sobre datos experimentales extraídos del sistema real que se quiere identificar, por lo que el problema de la identificación de sistemas no lineales puede ser abordado mediante la aplicación de la tecnología de las redes neuronales

artificiales, específicamente a través de los modelos con entrenamiento supervisado. Mediante información de la entrada y la salida de un sistema con función de transferencia desconocida, una estructura de identificación basada en redes neuronales artificiales puede realizar el mapeo de un conjunto de vectores de entrada asociados a unas clases de salidas, y representar a través de sus parámetros internos (pesos y tendencias) la dinámica de dicho sistema. El Dr. K. Narendra [8]-[9] presenta un estudio formal sobre identificación y control de sistemas dinámicos empleando redes neuronales con memoria en la entrada. Dado que el sistema real es desconocido, se debe asumir que corresponde a un tipo determinado, y que un modelo parametrizado basado en (3) puede en teoría representar el comportamiento de entrada – salida de cualquier sistema de ese tipo. La estructura de modelo de identificación implementada se muestra en la Figura 3.

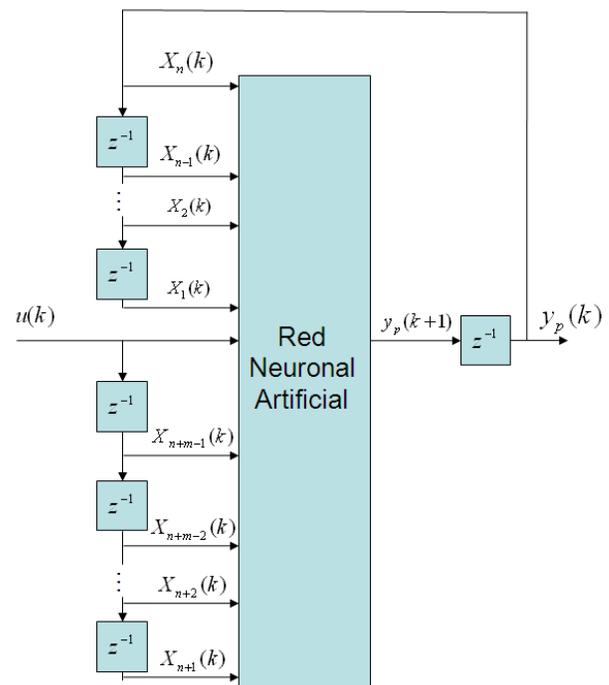


Figura 3. Estructura de identificación implementada.

Sea $X(k)$ el vector de entradas hacia la red neuronal, la cual almacenará mediante sus parámetros internos, la dinámica del sistema a identificar. En este modelo, las entradas de la red neuronal son la entrada del sistema real $u(k)$, las entradas $X_{n+m-j}(k)$ que representan estados anteriores de la misma señal de entrada $u(k)$; y la salida del sistema, representada como entrada a la red neuronal por $X_1(k)$ y sus respectivos valores en estados anteriores, $X_n(k)$. El número de entradas retardadas a considerar depende directamente del orden estimado del sistema a identificar. La capacidad de la red neuronal de almacenar un comportamiento dinámico realmente es otorgada a través de la presentación de estados anteriores, tanto de la entrada del sistema a identificar, como de su salida. Esto se conoce como memoria a la entrada (tapped delay).

Como se verá en el apartado de diseño, el modelo seleccionado es el de una red neuronal tipo perceptrón multicapa con algoritmo de entrenamiento Levenberg-Marquardt Backpropagation [7] y [18]. Los elementos arquitectónicos de la red son variables. Las ecuaciones relativas a los algoritmos de entrenamiento, se encuentran codificadas en la NNT de MATLAB.

5. Diseño del Neuro-Identificador de Sistemas

A continuación se presentan las fases de diseño e implementación de la herramienta desarrollada.

5.1. Diseño General

El diseño del Neuro-Identificador de Sistemas, ha sido realizado considerando que el producto final debe ser una herramienta de uso sencillo, con una interfaz gráfica de usuario que permita el acceso a todas las funcionalidades en consonancia con la estrategia de identificación, y con elementos software y hardware de buen rendimiento computacional. El diseño ha sido subdividido en dos grandes etapas que pueden apreciarse en la Figura 4.

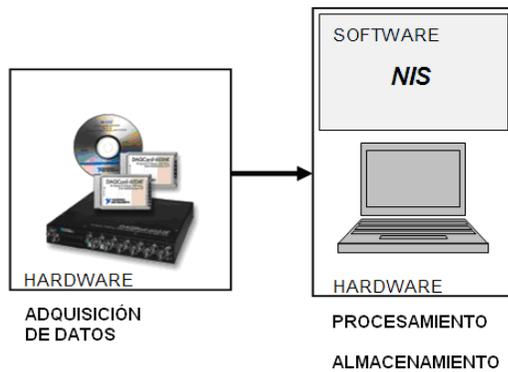


Figura 4. Diseño del Neuro-Identificador de Sistemas.

La etapa de software comprende la configuración del origen de datos del modelo, esto es, la configuración de las tarjetas de adquisición de datos con las que el usuario captura los datos experimentales, o bien la selección y configuración de un modelo MATLAB/Simulink® para ensayos basados en simulación. Dicha etapa contempla, en el mismo orden del esquema de identificación mediante redes neuronales visto en la Figura 2, opciones para llevar a cabo cada una de las fases de dicho esquema, así como también las operaciones típicas sobre archivos asociados al NIS, y la ayuda. La interfaz gráfica ha sido desarrollada bajo el módulo GUIDE™, de MATLAB®.

Para la etapa de hardware, se ha utilizado un computador portátil como elemento de procesamiento y almacenamiento, y una tarjeta de adquisición de datos junto a un equipo terminal de interconexión entre las señales eléctricas provenientes del sistema a identificar y la tarjeta mencionada. El software permite seleccionar y configurar los parámetros principales de la tarjeta de adquisición: frecuencia de muestreo, canales de entrada y salida, y tiempo de adquisición. El NIS está provisto de una interfaz con funciones de la Data Acquisition Toolbox™ de MATLAB®, lo que permite dar prioridad

en el software, a las interrupciones causadas por la tarjeta de adquisición de datos al sistema operativo embebido en el computador portátil.

5.2. Diagrama de Flujo del Software

El diagrama de flujo del software y sus distintos módulos se muestran en la Figura 5.

En el diagrama de flujo presentado en la Figura 5 se aprecian los seis módulos principales que integran el software de la herramienta implementada. Los módulos se encuentran delimitados por un recuadro que representa la interfaz gráfica de usuario principal, o GUI principal.

El bloque función Selección Origen de Datos del Modelo, permitirá seleccionar la fuente de los datos del sistema que se requiera identificar; existen dos posibilidades: origen de los datos desde un modelo Simulink, y origen de los datos desde una Tarjeta de Adquisición de Datos (DAQ). Para el primer caso, se desplegará una ventana que permite seleccionar y cargar un modelo Simulink cualquiera creado por el usuario. Si por el contrario se escoge la opción de origen desde DAQ, se desplegará una ventana que permite seleccionar el modelo de tarjeta, y una ventana que permite desplegar la información del dispositivo, en tiempo de ejecución del software.

El bloque Captura de Datos (1) puede ejecutar dos vías distintas, dependiendo del origen de datos seleccionado en el bloque anterior. En el caso de origen de datos desde modelo Simulink, la captura de datos realiza la acción de ejecutar la simulación del modelo seleccionado por el usuario. Para el segundo caso, cuando el origen de datos es una DAQ, la captura de datos despliega una nueva ventana de Adquisición de Datos, que luego debe ser parametrizada para su ejecución.

El bloque Acondicionamiento de Señales (2) permite, mediante una interfaz gráfica, aplicarle un acondicionamiento con Spline Cúbica tras ingresar un factor de atenuación, a las variables adquiridas en la etapa anterior. A través de éste método, es posible eliminar señales espurias no deseadas, adquiridas por lo general mediante una DAQ. Esta técnica permite eliminar el efecto de cuantización o discretización de las señales muestreadas, a efectos de aplicar asunciones propias de sistemas dinámicos en tiempo continuo.

El bloque Configuración de Red Neuronal Artificial (3) permite seleccionar la estructura de identificación que el usuario requiere utilizar para identificar un determinado sistema dinámico; luego, una ventana permitirá seleccionar y parametrizar la arquitectura de la red neuronal artificial. El conjunto de entrenamiento de la red neuronal es conformado por un vector de muestras de la entrada del sistema a identificar, y un vector de muestras de la salida del sistema a identificar. Este conjunto de entrenamiento se constituye automáticamente vía software.

El bloque Identificación de Sistemas (4) permite ejecutar el entrenamiento de la red neuronal, mostrando en tiempo

real la curva de error de la clasificación del conjunto de entrenamiento realizada por la red.

El bloque Validación (5) genera un bloque Simulink que representa a la red neuronal previamente entrenada. Luego se carga un modelo Simulink para validar y comparar el desempeño de la red neuronal con datos reales de validación del sistema identificado. El modelo permite también realizar una prueba escalón a la red neuronal.

Por último, el bloque Aproximación de Función de Transferencia (6) realiza un procesamiento gráfico de la curva de reacción del sistema identificado según las ecuaciones presentadas en la sección 2.1 de este artículo, grafica dicha curva y estima los parámetros necesarios para construir una función de transferencia de P.O.M.T.M. o una función de transferencia de segundo orden.

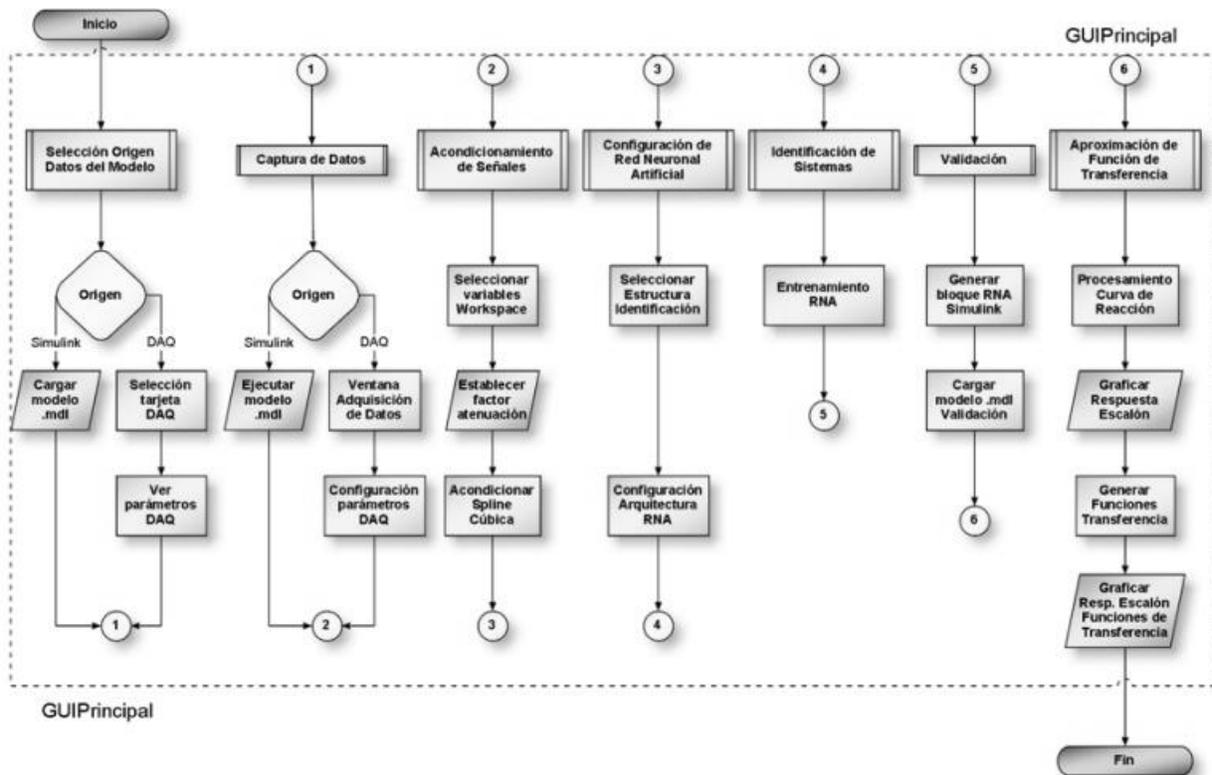


Figura 5. Diagrama de flujo del software del NIS.

5.3. Hardware de Adquisición y Procesamiento de Datos

La Figura 6 muestra el hardware para adquisición de datos utilizados, así como también el hardware requerido para el almacenamiento, procesamiento y ejecución del NIS.

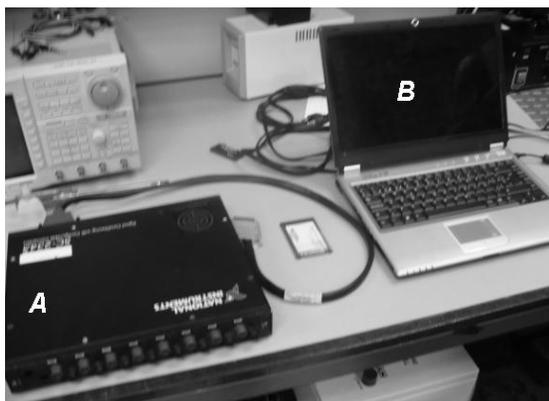


Figura 6. Hardware del NIS.

El hardware de adquisición (Fig. 6 - A) está conformado por la tarjeta de adquisición DAQ Card 6062E, cuya máxima frecuencia de muestreo es 500 kS/s. El hardware de estímulo de señales de excitación para las plantas a ser identificadas, es un generador de señales programable del fabricante Yokogawa, modelo FG300, cuya máxima frecuencia de salida para señales programadas por diskette de $3^{1/2}$ es 15 kHz. El hardware de procesamiento (Fig.6 - B), almacenamiento y ejecución del software es un computador portátil del fabricante Síragon, modelo Canaima 3050P, con procesador Intel Pentium M 2.1 GHz., 1 Gb de RAM y sistema operativo Microsoft Windows XP. Los requerimientos de software para la ejecución del NIS son: MATLAB 7.0 o superior, Drivers NI-DAQ.

5.4. Acabado Final

El aspecto definitivo de la ventana principal del software de la herramienta desarrollada se muestra en la Figura 7.

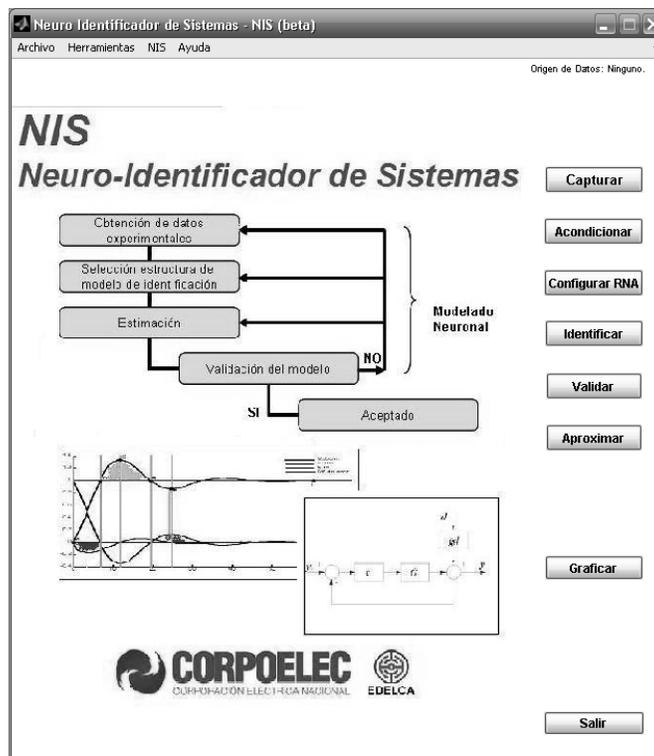


Figura 7. Ventana principal del software.

Este software sirve como Interfaz Gráfica de Usuario para la parametrización de la herramienta NIS; esto es, su hardware y el conjunto de ecuaciones a fin de realizar la identificación de sistemas dinámicos según las fases de modelado neuronal. Un conjunto de menús adicionales permiten realizar operaciones sobre archivo, acceder a la ayuda de la herramienta, y parametrizar el hardware de adquisición de datos.

III. RESULTADOS Y DISCUSIÓN

Para validar el correcto funcionamiento de la herramienta diseñada y evaluar su desempeño general, se identificaron sistemas bajo el entorno Simulink, y se identificaron

circuitos electrónicos de laboratorio.

1. Identificación de Sistemas en Entorno MATLAB/Simulink®

La Figura 8 muestra el diagrama de bloques, con un sistema cuya función de transferencia ha sido seleccionada arbitrariamente; dicho sistema ha sido identificado mediante obtención de datos de su entrada y salida, mediante el osciloscopio de Simulink. El sistema se encuentra en un lazo cerrado, y fue excitado con una señal aleatoria. Esta prueba ha permitido validar el funcionamiento de los módulos software que conforman a la herramienta NIS.

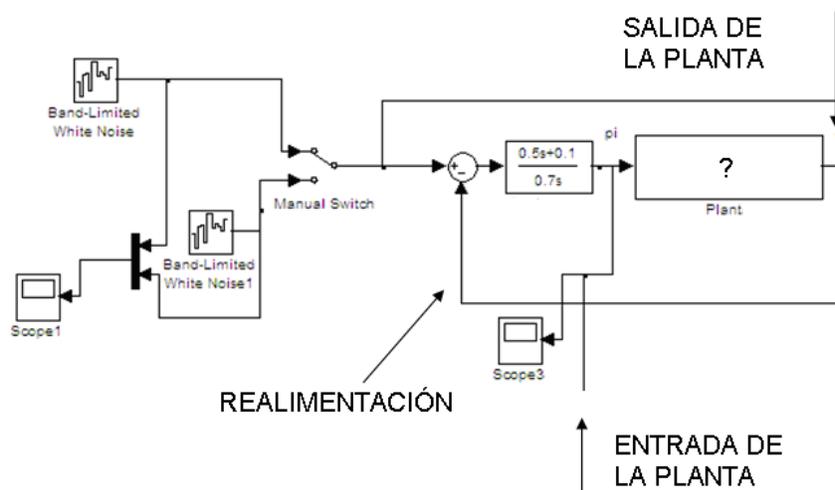


Figura 8. Sistema desconocido – modelo Simulink.

Las señales de entrada y salida de la planta, que fueron muestreadas utilizando el osciloscopio de Simulink, se muestran en las Figuras 9 y 10.

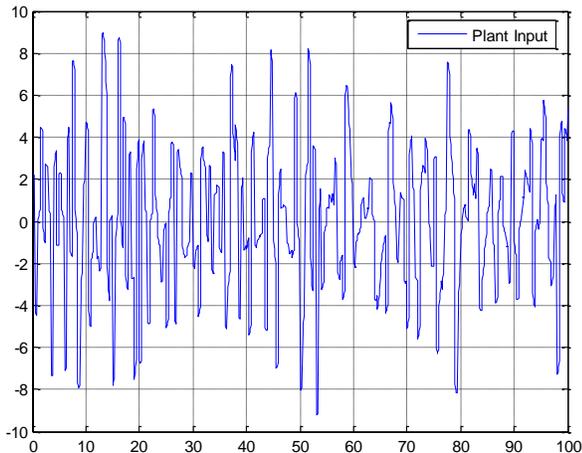


Figura 9. Entrada muestreada de la planta – modelo Simulink.

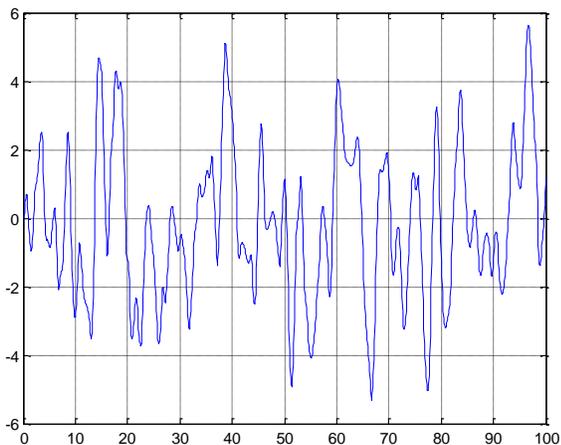


Figura 10. Salida muestreada de la planta – modelo Simulink.

La cota de error alcanzada por el algoritmo de entrenamiento Levenberg – Marquardt Backpropagation, aplicado a la arquitectura de red neuronal establecida para la identificación del circuito de primer orden, en este caso ha sido 3.85194×10^{-10} en 1551 épocas de entrenamiento. La Figura 11 muestra el gráfico de validación del modelo de red neuronal, donde se somete a la misma señal aleatoria, tanto a la planta como a la red neuronal artificial previamente entrenada, en condición de lazo abierto.

La Figura 12 muestra la respuesta de la planta y de la red neuronal en lazo abierto, ante una entrada escalón de amplitud 1. El error en la aproximación se puede asumir, considerando que la obtención de un modelo empírico con un error inferior al 20% es tolerable en la mayoría de los casos. Para reducir el error se hace necesario ejecutar el entrenamiento de la red neuronal en un mayor número de

épocas, o bien ajustar parámetros del algoritmo de entrenamiento como la tasa de aprendizaje.

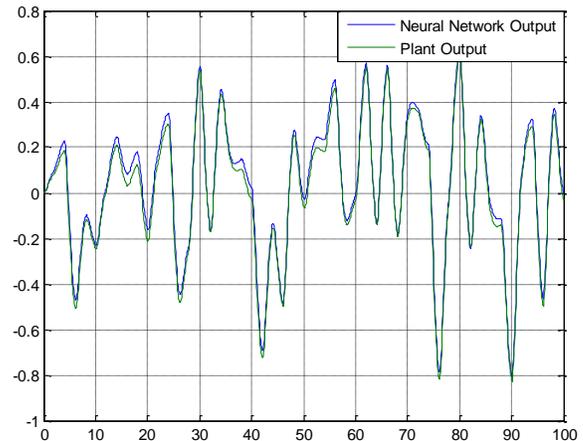


Figura 11. Respuesta de la red neuronal y del sistema ante la misma entrada aleatoria.

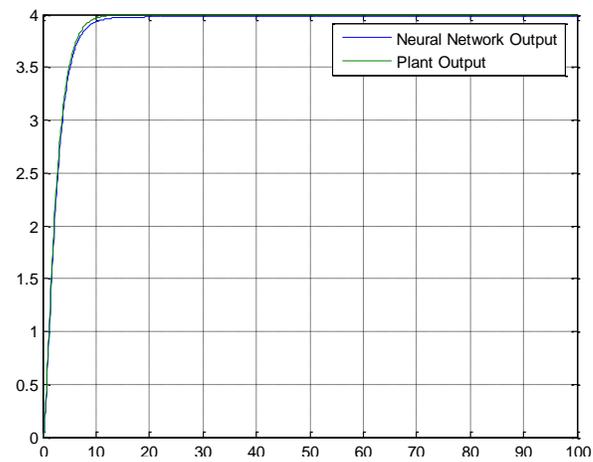


Figura 12. Respuesta de la red neuronal y del sistema ante entrada escalón.

Las Figuras 13 y 14 muestran respectivamente, la función de transferencia aproximada por el Neuro-Identificador y la salida del modelo Simulink.

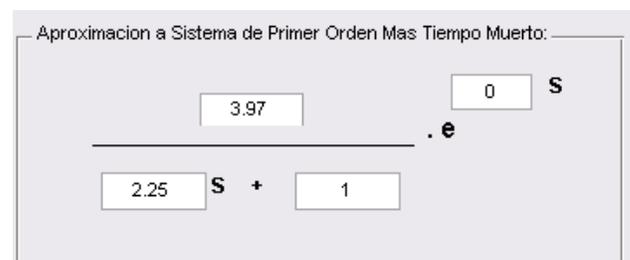


Figura 13. Función de transferencia aproximada por el NIS.

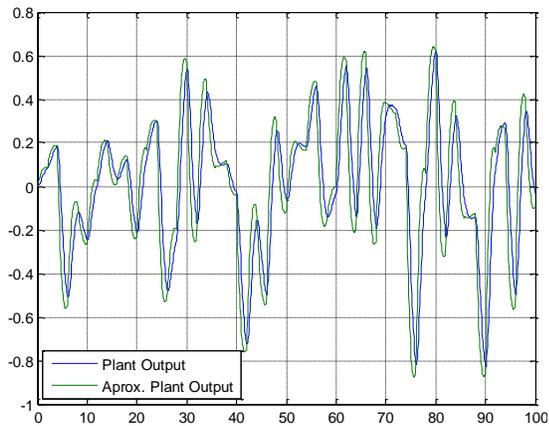


Figura 14. Salida del sistema Simulink y salida aproximada por el NIS.

2. Identificación de una Planta de Primer Orden

La Figura 15 muestra el circuito de primer orden que ha sido identificado mediante obtención de datos experimentales en laboratorio. Esta prueba ha permitido probar el NIS completo: su software y su hardware.

Las señales de entrada y salida de la planta, que fueron muestreadas utilizando el hardware de adquisición de datos, se muestran en la Figuras 16 y 17. La frecuencia de muestreo de las señales fue seleccionada excitando la entrada del circuito y observando hasta que punto la frecuencia atenúa la salida de la planta. En este caso, la frecuencia de muestreo ha sido de 80 Hz.

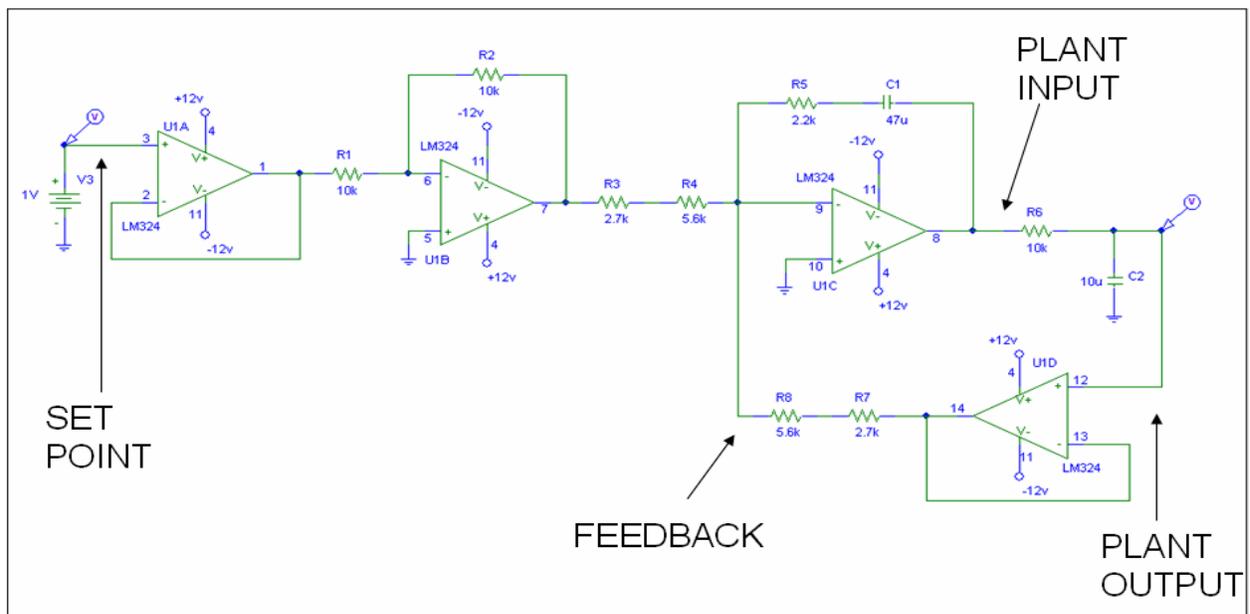


Figura 15. Circuito de primer orden.

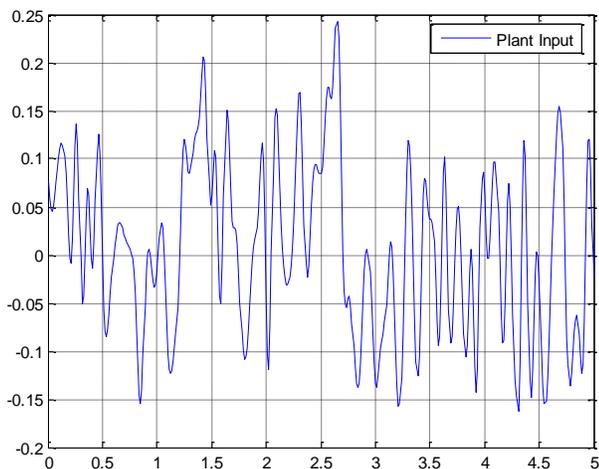


Figura 16. Entrada muestreada del circuito de primer orden.

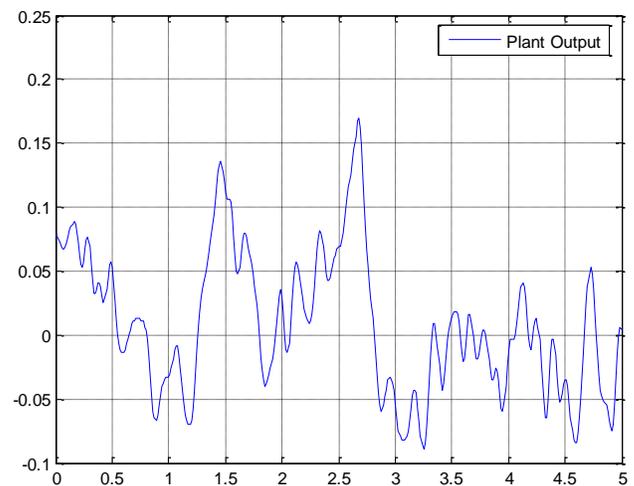


Figura 17. Salida muestreada del circuito de primer orden.

La cota de error alcanzada por el algoritmo de entrenamiento Descenso de Gradiente, aplicado a la arquitectura de RNA establecida para la identificación del circuito de primer orden, ha sido 2.98×10^{-6} en 10000 épocas de entrenamiento. Para reducir el error se hace necesario ejecutar el NIS en una arquitectura de cómputo de mayor velocidad, a fin de que el algoritmo converja más rápidamente.

La Figura 18 muestra una entrada de excitación aleatoria, mientras que la Figura 19 muestra el gráfico de validación del modelo de red neuronal, donde se somete a la misma señal aleatoria, tanto a la planta como a la red neuronal artificial previamente entrenada, en condición de lazo abierto.

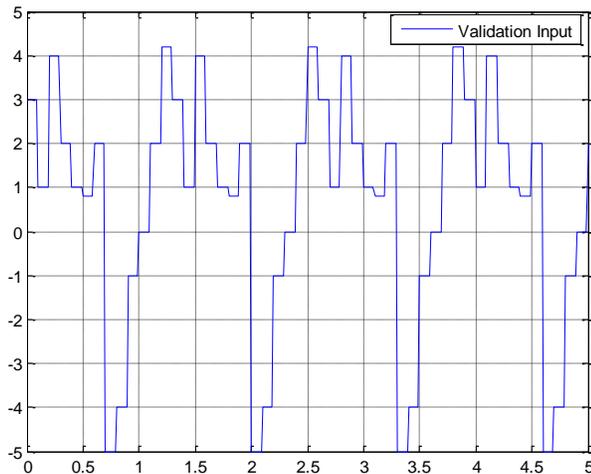


Figura 18. Entrada de validación para el modelo obtenido.

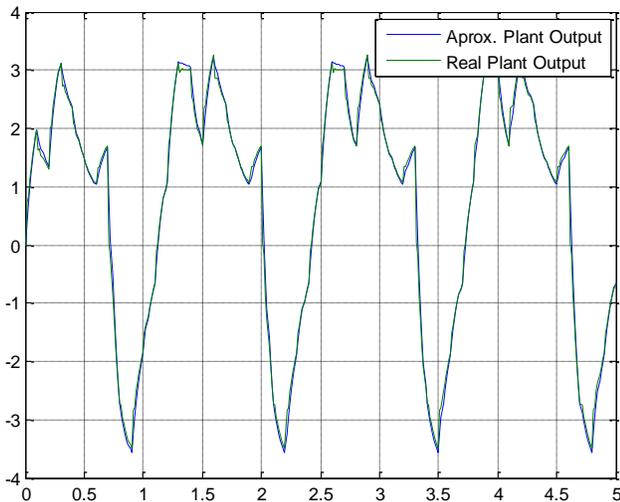


Figura 19. Respuesta de la red neuronal y del circuito ante la misma entrada aleatoria.

La Figura 20 muestra la respuesta de la planta y de la red neuronal en lazo abierto, ante una entrada escalón.

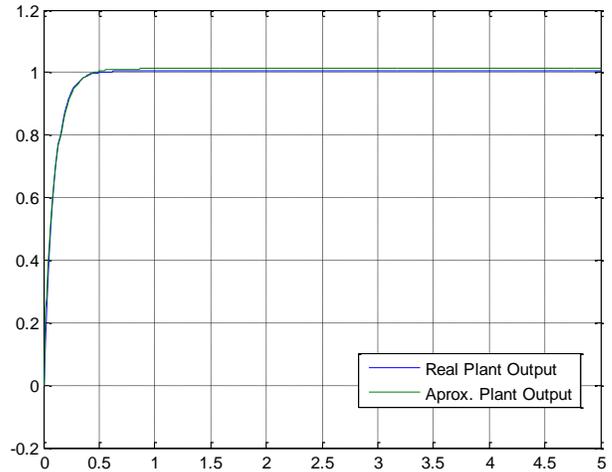


Figura 20. Respuesta de la red neuronal y de la planta, en lazo abierto, ante entrada escalón.

Las Figuras 21 y 22 muestran respectivamente, la función de transferencia aproximada por el Neuro-Identificador y la salida real del circuito de primer orden.

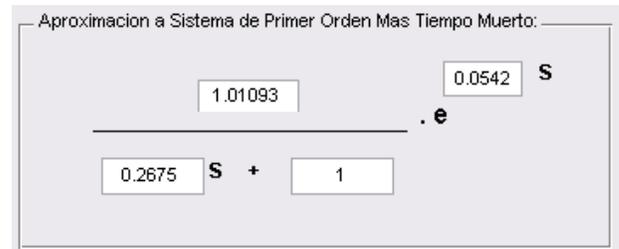


Figura 21. Función de transferencia aproximada por el NIS – caso circuito de primer orden.

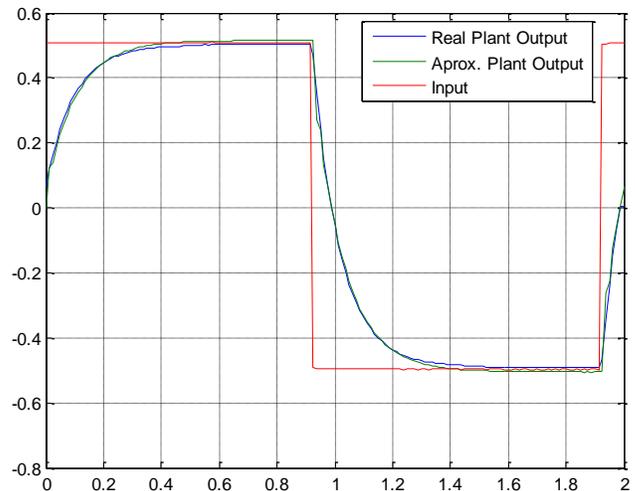


Figura 22. Salida del circuito de primer orden y salida de la planta aproximada por el NIS.

Cuantitativamente, el grado de confiabilidad del modelo se ha evaluado mediante un factor de correlación cuadrática multivariable. En probabilidad y estadística, la correlación indica la fuerza y la dirección de una relación

lineal entre dos variables aleatorias. Se considera que dos variables cuantitativas están correladas cuando los valores de una de ellas varían sistemáticamente con respecto a los valores homónimos de la otra: si tenemos dos variables (A y B) existe correlación si al aumentar los valores de A lo hacen también los de B y viceversa. Se ha utilizado la ecuación:

$$R_y^2 = 1 - \left| 1 - \frac{\sum_{i=1}^n \hat{Y}_i^2}{\sum_{i=1}^n Y_i^2} \right| \leq 1 \quad (14)$$

Que expresa la fuerza de la relación lineal entre el conjunto de datos de salida del modelo aproximado, \hat{y}_i , y el conjunto de datos de salida del sistema o planta real, y_i , ante una misma entrada cualquiera. El cuadrado del factor de correlación únicamente indica que es una expresión cuadrática. La ecuación ha sido ajustada de forma tal que, indistintamente del sentido que arroje la correlación, el factor siempre sea positivo e inferior o igual a la unidad, donde la unidad representa una correlación total. Valores superiores a 0.96 en el factor de correlación, indican claramente que el modelo aproximado está correlado con el modelo real. Así, para el modelo de primer orden aproximado se tiene:

$$R_y^2 = 0.9670 \quad (15)$$

Que, junto a la correspondencia visual mostrada en las gráficas anteriores, confirma la validez del modelo identificado con el NIS.

IV. CONCLUSIONES

1. El Neuro-Identificador de Sistemas hace uso efectivo de la metodología empleada para la identificación mediante Redes Neuronales Artificiales.
2. La herramienta implementada permite, mediante el ajuste de pocos parámetros, identificar sistemas dinámicos LTI - SISO. A través del experimento realizado, se demuestra que es válido utilizar el modelado neuronal en la identificación de sistemas LTI que se encuentren en un lazo de control realimentado.
3. En relación al desempeño, se observa que el error cuadrático medio obtenido en la fase de entrenamiento de la red neuronal debe ser inferior a 10^{-10} para obtener un modelo que permite obtener correlaciones cuadráticas multivariadas superiores a 0.96.
4. Por último, se demuestra que es posible después de obtener el modelo de red neuronal que representa la dinámica del sistema LTI, realizar pruebas escalón y, aplicando técnicas de modelado empírico de procesos sobre la red neuronal previamente entrenada, aproximar una función de transferencia.

V. REFERENCIAS

1. Gil R. Páez D., "Identificación de Sistemas Dinámicos utilizando Redes Neuronales RBF", Revista Iberoamericana de Automática e Informática Industrial, vol. 4, núm. 2, Comité Español de Automática, Madrid, España, pp. 32-42.
2. Sarmiento S., "Identificación Empírica de Modelos", UNEXPO Vicerrectorado Puerto Ordaz, Puerto Ordaz, Venezuela, Marzo, 2003. pp 173-188.
3. Ogata K., "Modern Control Engineering", Third Edition. Prentice Hall, 1998. 997 p.
4. Contreras R., "Tuning of Governance System of a 805 MVA Generator Unit at Guri Plant After 25 Years of Service", 1st Venezuelan Congress on Networks and Electrical Energy, Anzoátegui, Venezuela. November, 2007. pp. 1-9.
5. Hangos K. M., Bokor J., Szederkényi G., "Analysis and Control of Nonlinear Process Systems", Springer-Verlag London Limited, 2004.
6. Zambrano A., Pinto L., "Aplicación de las Redes Neuronales Artificiales en Procesadores Digitales de Señales: Caracterización de Sensores Infrarrojos", Revista Universidad, Ciencia y Tecnología, vol. 13, UNEXPO Vicerrectorado Puerto Ordaz, Puerto Ordaz, Venezuela, pp. 20-27.
7. Demuth H., Beale M., Hagan M., "Neural Network Design", PWS Publishing Company, Boston, EUA, 1996.
8. Narendra, K. S., Parthasarathy, K., "Backpropagation in Dynamical Systems containing Neural Networks", Technical Report 8905, Centre for Systems Science, Department of Electrical Engineering, Yale University, 1989.
9. Narendra, K. S., Parthasarathy, K., "Identification and Control of Dynamical Systems Using Neural Networks," IEEE Transactions on Neural Networks, Vol. 1, No. 1, pp. 4-27, 1990.
10. Kuschewsky G. J., Hui S., Zak S.H., "Application of feedforward neural networks to dynamical system identification and control", IEEE Transactions on Control Systems Technology, Vol. 1, No. 1, 1993, pp. 37-49.
11. Chi S.R., Shoureshi R., Tenorio M., "Neural Networks for System Identification", IEEE Control Systems Magazine, No. 10, 1990, p.p. 31-34.
12. Chen S., Billings S.A., Grant P.M., "Nonlinear System Identification using Neural Networks", International Journal of Control, Vol. 51, No. 6, 1990, p.p. 1191-1214.
13. Poggio T., Girosi F., "A Theory of Networks for Approximation and Learning", Artificial Intelligence Memo No. 1140, Massachusetts Institute of Technology, 1989.
14. Sanner R.M., Slotine J.E., "Stable Adaptive Control and Recursive Identification using Radial Gaussian

- Networks”, Proceedings of 30th on Decision and Control, Brighton, England, 1991.
15. Nechyba M., Xu Y., “Neural Network Approach to Control System Identification with Variable Activation Functions”, The Robotics Institute, Carnegie Mellon University, Pittsburg, EUA, 2000.
 16. Cañete J., “Modeling and Control using Neural Networks”, Malaga University, PhD Program on Mechatronics Engineering, Málaga, Spain. April, 2008.
 17. Troncone N., “Introducción a las Redes Neurales”. Trabajo de Ascenso, UNEXPO Vicerrectorado Puerto Ordaz. Puerto Ordaz, Venezuela. Diciembre, 2002. pp 10-50.
 18. The Mathworks Inc., “MATLAB 7.0 Neural Networks Toolbox Help Notes”, May, 2006.