

## EVALUACIÓN DE MODELOS NEURONALES DESTINADOS A OPERAR SISTEMAS INTELIGENTES DE CONTROL

Juan F. Giró<sup>a y b</sup>, Sandra Olariaga<sup>a</sup> y José E. Stuardi<sup>b</sup>

<sup>a</sup> *Departamento de Ingeniería de Sistemas de Información, Facultad Regional Córdoba, Universidad Tecnológica Nacional, Córdoba, Argentina, [juanfgiro@gmail.com](mailto:juanfgiro@gmail.com), <http://www.frc.utn.ar>*

<sup>b</sup> *Departamento de Estructuras, Facultad de C. E. F. y N., Universidad Nacional de Córdoba, Casilla de Correo 916, 5000 Córdoba, Argentina, [jstuardi@gmail.com](mailto:jstuardi@gmail.com), <http://www.efn.uncor.edu>*

**Palabras clave:** neurocontroladores inteligentes, control adaptativo, redes neuronales artificiales.

**Resumen.** La actividad desarrollada en torno al control de ensayos de motores de combustión interna y su simulación ha permitido comprobar que una adecuada implementación de soluciones simples, tal como es el caso de los controladores PID, puede producir resultados aceptables: rápida respuesta, y cierta precisión y confiabilidad. No obstante, las dificultades que encierra la determinación de sus constantes de operación y la necesaria coordinación de las acciones de control son una seria limitación. Este problema no es nuevo y su solución estimuló diversas líneas de investigación, encuadradas como sistemas adaptativos de control y precursoras del control inteligente. Todo sistema de control inteligente debe ser autónomo y la presencia de una red neuronal en condiciones adecuadas le asegura al sistema cierto grado de autonomía. Así, a través de redes neuronales se implementan controladores capaces de operar sobre sistemas en principio desconocidos, y de responder a cambios en la dinámica de los mismos a lo largo del tiempo, sin mayor información que la obtenida de su experiencia con el proceso. Sin embargo, persisten aún muchas dudas referidas a qué arquitecturas y técnicas de entrenamiento son más adecuadas para obtener la mejor respuesta de estos modelos. En este trabajo se muestra la conveniencia de incorporar el conocimiento sobre el problema tratado como un criterio central en la selección de los modelos neuronales más apropiados en cada caso. Se presentan resultados obtenidos al evaluar diferentes redes destinadas a representar la dinámica directa e inversa de un dispositivo y se extraen recomendaciones, que facilitarán el desarrollo de modelos neuronales de sistemas de control para ensayos de motores de combustión interna y su simulación.

## 1 INTRODUCCIÓN

La actividad desarrollada en torno al control de ensayos de motores de combustión interna y su simulación ha permitido comprobar que una adecuada implementación de soluciones simples, tal como es el caso de los controladores PID, puede producir resultados aceptables; es decir, obtención de respuestas rápidas con cierta precisión y confiabilidad. Esto es así a pesar de tratarse de sistemas de múltiples entradas y salidas (MIMO) donde, desde un punto de vista teórico, la técnica de control empleada no sería la apropiada para asegurar estos buenos desempeños.

Cabe aquí acotar que las aplicaciones a las que se hace referencia se caracterizan por demandar cambios en las condiciones de operación en intervalos de tiempo muy breves y presentar mucha sensibilidad a las variables de control. Estos casos pueden llevar con facilidad a que se desarrollen respuestas oscilantes, generalmente inaceptables, y aun a la destrucción de las instalaciones por sobrevelocidad. Es así que a la hora de seleccionar un controlador es muy importante tener en cuenta las mencionadas características, las que distinguen claramente a esta aplicación de otras cuasi-estacionarias, como es el caso del control de temperaturas en hornos y de otros procesos industriales.

Otro aspecto a considerar es que, por tratarse de sistemas MIMO, a la necesidad de ajustar adecuadamente las variables de control se agrega la necesidad de atender la coordinación entre las mismas, lo que implica la elección de un camino cada vez que en el espacio de estados hay que trasladarse desde un punto a otro. Aquí podrían distinguirse dos condiciones de control, que son denominadas de regulación y de seguimiento, según que en este espacio de estados el objetivo quede representado por un punto fijo o por uno que sea variable en el tiempo. Cualquiera sea el caso, si se consideran las restricciones que la aplicación impone a las condiciones de operación, los posibles caminos en el espacio de estados quedan limitados por hipersuperficies, que son las que Utkin denominó “superficies de deslizamiento” (Utkin, 1977).

Para la implementación de estas unidades de control, con capacidad para cumplir secuencias de ensayos seleccionando caminos compatibles con estas superficies de deslizamiento, normalmente se recurre a sistemas conmutados (Branicky, 1998). En estos casos una máquina de estados finitos irá seleccionando la opción de operación más apropiadas para cada condición, lo que conduce a una forma de implementación de los denominados sistemas de control de estructura variable (DeCarlo, 1988).

Sin embargo, como contrapartida a los buenos desempeños ya mencionados al comienzo, es necesario reconocer las dos importantes limitaciones de este tipo de sistemas de control: 1) deben individualmente determinarse las constantes de operación de los controladores PID para todas las combinaciones de señales de control y opciones de realimentación y 2) debe definirse la función de transición para la máquina de estados finitos que comande la redefinición de la estructura de control. Ambas son tareas complejas, pero de diferente naturaleza.

En efecto, la definición de las constantes de las unidades PID puede encontrar un punto de partida en las recomendaciones de Ziegler y Nichols (Ziegler, 1942) y sus posteriores numerosas variantes. Así es que esta tarea, aunque laboriosa, se ve en alguna medida facilitada. Por el contrario, las condiciones de conmutación a ser incorporadas en la función de transición son muy específicas de cada tipo de problema y deben apoyarse en experiencias obtenidas en casos similares, ya que es muy difícil que puedan preverse en forma teórica. Nótese además que, tanto las constantes PID como la función de transición, deben ser ajustadas cada vez que cambien las condiciones del sistema, y, en el caso aquí tratado, esto ocurre cuando se cambia el tipo de motor o la regulación del mismo.

Es obvio que el problema descripto no es nuevo y fue oportunamente advertido por numerosos investigadores, lo que estimuló la presentación de diversas propuestas, muchas de las cuales quedan encuadradas como sistemas adaptativos de control y pueden ser consideradas precursoras del control inteligente. La denominación de control inteligente se reserva para una generalización del concepto del control, que incluye capacidades de adaptación y aprendizaje, y que es adecuado para operar sobre gran cantidad de variables. Para su implementación, las redes neuronales artificiales ocuparon progresivamente un lugar cada vez más destacado, hasta alcanzar una participación central en estas técnicas destinadas al control autónomo de procesos dinámicos no lineales. De esta manera se fue estableciendo un nuevo y vasto dominio en el que intervienen la teoría de control, la inteligencia artificial y la ingeniería de software.

## 2 NEUROCONTROLADORES

Así, la creciente complejidad de los sistemas a controlar y las cada vez mayores exigencias sobre los resultados esperados han llevado a replantear las técnicas de control disponibles y a la aparición de otras nuevas, que pueden enmarcarse dentro del ya mencionado control inteligente. En este campo, se pueden citar los sistemas expertos, el control borroso, el control basado en algoritmos genéticos, las redes neuronales artificiales y diversas combinaciones entre ellos.

En este trabajo se concentrará la atención en las redes neuronales artificiales, y para comenzar es necesario señalar que son dos los aspectos que justifican el gran interés que despiertan estas redes en las aplicaciones de control. El primero se refiere a su aptitud para aproximar funciones altamente no lineales, lo que permite modelar sistemas complejos y construir controladores para los mismos. El segundo es su capacidad de aprendizaje, lo que hace posible alcanzar un elevado grado de autonomía.

Al considerarse el primer aspecto, ha sido demostrado que toda red que disponga de por lo menos una capa oculta y contenga en ella una adecuada cantidad de unidades convenientemente entrenadas, adquirirá el carácter de aproximador universal. Esto es, será capaz de reproducir cualquier función continua y no lineal, que pueda estar definida en cierto hiperespacio, y así permitir la implementación de controladores capaces de operar sobre sistemas en principio desconocidos.

Con respecto al segundo aspecto, se debe observar que las redes neuronales artificiales son sistemas masivamente paralelos, que materializan su capacidad de aprendizaje a través de la modificación de los pesos de interconexión entre sus neuronas. Más aún, existe la posibilidad de alterar en forma dinámica la topología de una red, es decir, su número de capas, la cantidad de neuronas en cada capa o la manera como éstas son interconectadas, lo que ofrece otras formas menos convencionales de aprendizaje. Cualquiera sea el caso, las redes pueden adecuar dinámicamente su desempeño para responder a cambios en el sistema o su entorno a lo largo del tiempo, sin mayor información que la obtenida de su experiencia con el proceso.

Para la implementación de neurocontroladores estas redes ofrecen numerosas opciones, destacándose los modelos que se basan en funciones de base radial y en el perceptrón multicapa. Una de las principales diferencias entre ambos modelos son las funciones de activación empleadas, ya que en las redes de base radial las funciones de activación son locales, normalmente de tipo Gausiano, mientras que en las redes de perceptrones las funciones son globales, normalmente sigmoides. Estas últimas constituyen a su vez la base para las llamadas redes recurrentes.

Nótese que se puede obtener un marco unificador para el control neuronal al considerar el entrenamiento de la red como un problema de optimización no lineal, en el cual se trata de encontrar una representación óptima que minimice una función objetivo en el hiperespacio de pesos de la red. Así, la clave para distinguir las diferentes posibilidades en el diseño de

neurocontroladores está dada por el rol que desempeñe la red neuronal en la función objetivo. Básicamente, puede hacerse referencia a dos roles principales, que son los que corresponden al modelado directo y el modelado inverso de las características dinámicas del objeto a ser controlado (*Planta*).

Conceptualmente, el comportamiento de todo sistema discreto dinámico  $S$  puede expresarse como:

$$S: x(t+1) = f(u(t), x(t)) \quad (1)$$

$$y(t) = \varphi(x(t)) \quad (2)$$

donde  $f$  denota la función de transición de estados,  $x(t)$  representa el vector de estado del sistema,  $y(t)$  el vector de salida y  $u(t)$  la acción de control en ese mismo instante. La función  $\varphi$  relaciona los vectores de estado y de salida, expresando el concepto de observabilidad.

La ecuación (1) puede ser rescrita,

$$u(t) = C(x(t), x(t+1)) \quad (3)$$

donde  $C$  es una forma de modelo inverso de  $S$ , que representa la controlabilidad del sistema, y cuya misión es generar una acción apropiada que permita cumplir cierta trayectoria en el espacio de estados, en un intervalo de tiempo finito. Nótese que, estrictamente hablando, la referencia a un modelo inverso no se corresponde con el concepto de inversa matemática, sino más bien a una relación entre entradas y salidas expresadas en un sentido apropiado.

Alternativamente, la acción de control puede ser redefinida por una función  $C'$  que opere a partir del vector de salida  $y(t)$  y de su valor deseado  $z(t)$  en determinado instante:

$$u(t) = C'(y(t), z(t+1)) \quad (4)$$

Finalmente, en cualquier instante  $t$  el error cometido quedará representado por la diferencia entre ambos vectores:

$$\varepsilon(t) = y(t) - z(t) \quad (5)$$

donde el objetivo de la acción de control es procurar que el comportamiento de  $|\varepsilon|$  sea monotónicamente descendente y tienda rápidamente a cero.

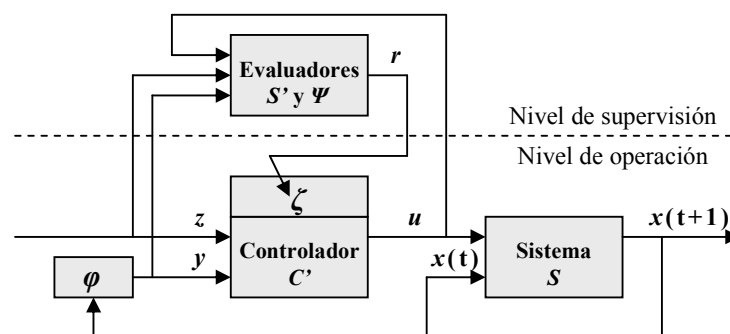


Figura 1: Esquema de un sistema de control adaptativo

Obsérvese que, por tratarse de un sistema adaptativo, habrá que prever la alteración de la función  $C'$  ante cambios en el entorno, de manera de asegurar en todo momento el cumplimiento del objetivo del sistema. El proceso de adaptación será entonces conducido por un vector de referencia  $r(t)$  tal que:

$$r(t) = \Psi(u(t), y(t), e(t)) \quad (6)$$

luego

$$C'(t) = \zeta(r(t)) \quad (7)$$

donde  $\Psi$  representa la inteligencia del sistema y  $\zeta$  su capacidad de aprendizaje. Esto es presentado en el esquema de la [Figura 1](#), donde  $\mathcal{S}$  representa un modelo del objeto controlado  $\mathcal{S}$ , cuya finalidad es anticipar su conducta en diferentes condiciones de operación.

En estos casos, las redes neuronales artificiales ofrecen una opción conveniente para la representación del modelo de la unidad de control  $\mathcal{C}$  y la dinámica directa del propio sistema controlado  $\mathcal{S}$ . Para ello, deben considerarse los numerosos aspectos que son propios de la selección e implementación de modelos neuronales, y ponerse particular atención en sus limitaciones.

### 3 LIMITACIONES DE LOS MODELOS NEURONALES

Los buenos desempeños que exhiben las redes neuronales en aplicaciones de control deben ser contrastados con sus debilidades. En su mayoría, estas últimas están referidas a aspectos particulares de su concepción y entrenamiento, donde se presentan múltiples posibilidades y se carece de recomendaciones definitivas que permitan seleccionar la arquitectura y métodos más convenientes para resolver cada caso. Por ello, y en forma resumida, se revisan a continuación los numerosos aspectos que deben considerarse para implementar una red multicapa de perceptrones.

#### 3.1 Selección de la arquitectura de la red

Se refiere al número más conveniente de capas ocultas, número de neuronas y la conectividad entre ellas. Según Kolmogorov una única capa oculta con un mínimo número de neuronas permite identificar cualquier función continua ([Valishevsky, 1998](#)). Sin embargo otros autores, tales como Hertz recomendaron la utilización de dos capas ocultas, como mínimo, para alcanzar determinado nivel de precisión ([Hertz, 1994](#)). Es necesario aquí advertir que el sobredimensionamiento de una red puede favorecer el proceso de aprendizaje, aunque al contribuir a la presencia de mínimos locales en su función error, puede también entorpecerlo. Asimismo, el sobredimensionamiento de la red resiente su capacidad de generalización.

#### 3.2 Funciones de activación

La función de activación más utilizada es la sigmoideal, aunque en algunos casos es más conveniente la tangente hiperbólica, la función lineal o una combinación de ellas, dando lugar a esquemas híbridos. Esto está en relación con el grado de no linealidad de la función representada y varios autores han estudiado la conveniencia de utilizar unas u otras ([Pham, 1997](#)).

#### 3.3 Definición de pesos iniciales

Tradicionalmente, se asignan valores aleatorios a los pesos de una red para iniciar el proceso de entrenamiento. Es probable que esta práctica tenga por único respaldo la ley del mínimo esfuerzo, ya que son numerosos los autores que han demostrado su inconveniencia, tal como lo hicieron Nguyen y Widrow ([Nguyen, 1990](#)). En efecto, iniciar el proceso de entrenamiento con valores aleatorios en los pesos puede conducir a una situación en la que numerosas neuronas contribuyan muy poco al desempeño de la red, y su actualización durante el proceso de entrenamiento sea inútil. Más aún, esta asignación no permite asegurar que se dispondrá de un número suficiente de unidades para representar una función en todo su dominio de utilización.

### 3.4 Proceso de entrenamiento

Es indudable que el método de Backpropagation presentado por Rumelhart (Rumelhart, 1986) se convirtió en un clásico, al estimular una verdadera revolución en el campo de las redes neuronales. Sin embargo, es necesario destacar que este algoritmo presenta puntos débiles muy importantes, como son una muy lenta convergencia y la posibilidad de quedar atrapado en los mínimos locales de la función error. Para superar estas dificultades se han propuesto diversas variantes, como son el método Rprop (Riedmiller, 1993), BPVS y ABP (Plagianakos, 2002) o el denominado Quasi-Newton (Beigi and Li, 1990), que conduce a procesos de entrenamiento más rápidos y seguros.

## 4 EL PROBLEMA Y SUS SOLUCIONES

En las secciones anteriores se han establecido, en forma resumida, los roles que desempeñan las redes neuronales artificiales en la implementación de unidades de control modernas. Al mismo tiempo, se reconoce que la adecuada implementación de tales redes requiere de la precisa definición de numerosos parámetros y para esta tarea no parece disponerse de criterios que puedan considerarse definitivos. Estas dos ideas centrales, que son en alguna medida contrapuestas, son destacadas por la literatura especializada y arrojan ciertas dudas sobre las ventajas de la aplicación de estos modelos conexionistas. Queda así planteado un problema que, para esclarecerlo, es necesario comenzar por hacer algunas importantes distinciones:

- a) Las redes neuronales artificiales tienen dos aplicaciones principales, que son la identificación de patrones y la representación de funciones. Se trata de problemas muy diferentes, aunque en ambos casos se emplean modelos neuronales que son muchas veces del mismo tipo. Sin embargo, las experiencias en uno de estos campos no pueden ser trasladadas al otro y esto no siempre es advertido.
- b) Aún dentro del mismo campo de la representación de funciones se presentan dos casos que son esencialmente distintos, como es el de representar fenómenos absolutamente desconocidos y el de modelar sistemas que responden a principios físicos claramente establecidos. En el primer caso se trata de verdaderos modelos de cajas negras, solo expresables a través de sus relaciones de entrada-salida, mientras que el segundo debe ser reconocido como un problema de identificación de parámetros.

Como ya fue anticipado, Kolgomorov, Herz y otros muchos investigadores procuraron definir las condiciones mínimas que debía cumplir una red neuronal para ser capaz de representar cualquier función continua. Sin embargo, es válido poner en duda la conveniencia de recurrir a modelos tan generales cuando se trata de representar fenómenos conocidos. En efecto, al ser posible anticipar la forma general de las expresiones diferenciales que gobiernan el problema, como ocurre con las unidades de control, parecería recomendable aplicar este conocimiento para definir modelos tan específicos como sea posible.

Estas ecuaciones corresponden al equilibrio dinámico de un sistema de varios grados de libertad, que incluye desplazamientos y rotaciones, en el que intervienen fuerzas de variada naturaleza. A los torques externos necesarios para provocar los movimientos deseados, deben agregarse las fuerzas de inercia, fricción, gravitatorias y las originadas en las aceleraciones centrípetas y de Coriolis, muchas de las cuales adquieren particular importancia debido a la alta velocidad con que operan estos equipos. Si bien la complejidad geométrica de los dispositivos, la presencia de fuerzas de rozamiento de variada naturaleza y los accionamientos de origen electromecánico hacen difícil la completa formulación de estas ecuaciones, no debe olvidarse que responden a principios de la mecánica que son ampliamente conocidos. Es

decir, de manera alguna se trata de cajas negras y el conocimiento disponible debería ser suficiente o por lo menos de gran ayuda a la hora de definir la arquitectura de las redes encargadas de representarlos.

La idea de que la arquitectura de las redes neuronales debe estar claramente inspiradas en el conocimiento sobre la naturaleza de los fenómenos representados fue presentada por Rudolph y Haecker, quienes aseguran en sus trabajos que éste es el camino para obtener modelos simples, eficientes y fáciles de entrenar (Rudolph, 1998 y Haecker, 2001). Más aún, Rudolph recomendó la utilización del teorema  $\pi$  de Buckingham (Buckingham, 1914) para garantizar la homogeneidad dimensional de las expresiones representadas.

Pasando a consideraciones más específicas, el objeto de los modelos neuronales en el caso de las unidades de control es anticipar la próxima condición del sistema al cabo de un cierto intervalo de tiempo (modelo directo), o las fuerzas necesarias para alcanzar ciertas condiciones de operación (modelo inverso). Para definir tales modelos se analizan sistemas lineales y no lineales.

#### 4.1 Sistemas Lineales

El sistema de ecuaciones diferenciales de equilibrio dinámico que gobierna el problema mecánico del tipo estudiado tiene la forma general:

$$M \ddot{x} + C \dot{x} + K x = u \quad (8)$$

donde  $M$ ,  $C$  y  $K$  representan las matrices de inercia, amortiguamiento y rigidez respectivamente. En caso de conocerse, en un cierto instante, la posición, velocidad y aceleración del sistema, la predicción de su nueva posición al cabo de un intervalo de tiempo  $\Delta t$  puede hacerse a partir de simples consideraciones cinemáticas. En efecto, si se asume que la aceleración se mantiene constante en el intervalo, se tendrá:

$$x(t+\Delta t) = x(t) + \dot{x}(t) \Delta t + 0.5 \ddot{x}(t) \Delta t^2 \quad (9)$$

Sin embargo, es poco frecuente que se conozca la velocidad y aceleración del sistema. Para resolver estos casos puede recurrirse a fórmulas de diferenciación numérica para expresar la velocidad y aceleración en función de los desplazamientos en intervalos anteriores. Por ejemplo:

$$\dot{x}(t+\Delta t) = [11 x(t+\Delta t) - 18 x(t) + 9 x(t-\Delta t) - 2 x(t-2\Delta t)] / (6 \Delta t) \quad (10)$$

$$\ddot{x}(t+\Delta t) = [2 x(t+\Delta t) - 5 x(t) + 4 x(t-\Delta t) - x(t-2\Delta t)] / (\Delta t^2) \quad (11)$$

Si se considera la ecuación de equilibrio dinámico (8) en el instante  $t+\Delta t$ , se reemplaza la aceleración y velocidad en las expresiones (10) y (11), y finalmente se agrupan los términos, se llega a una ecuación que suele ser atribuida a Houbolt (Houbolt, 1950) y que tiene la siguiente forma general:

$$x(t+\Delta t) = A^{-1} [u(t+\Delta t) + B x(t) + D x(t-\Delta t) + E x(t-2\Delta t)] \quad (12)$$

donde las matrices  $A$ ,  $B$ ,  $D$  y  $E$  quedan expresadas como una combinación lineal de las matrices  $M$ ,  $C$  y  $K$ .

Nótese que la expresión (9) permite anticipar la futura posición del sistema a partir de una combinación lineal del desplazamiento, velocidad y aceleración en un intervalo de tiempo anterior. De la misma forma, la expresión (12) hace lo propio combinando linealmente las posiciones de intervalos anteriores. También debe notarse que la cantidad de términos de esta

última expresión dependerá de las fórmulas de diferenciación numérica que se adopten. Luego, se deduce fácilmente que un conjunto de unidades Adaline (Adaptive Linear Element) serán suficientes para resolver este tipo de problema. Más precisamente, serán necesarias tantas unidades Adaline como grados de libertad tenga el sistema representado.

Para expresar la dinámica inversa del sistema, es decir determinar las fuerzas que deben aplicarse para alcanzar ciertas condiciones cinemáticas, solo será necesario reordenar la ecuación (12), lo que conduce a:

$$\mathbf{u}(t+\Delta t) = \mathbf{A} \mathbf{x}(t+\Delta t) - \mathbf{B} \mathbf{x}(t) - \mathbf{D} \mathbf{x}(t-\Delta t) - \mathbf{E} \mathbf{x}(t-2\Delta t) \quad (13)$$

que será también representada por un conjunto de unidades Adaline.

Una vez entrenadas, las unidades Adaline darán lugar a las siguientes expresiones, que se corresponden con las ecuaciones 9, 12 o 13:

$$\mathbf{x}(t+\Delta t) = \mathbf{W}_0 + \mathbf{W}_1 \mathbf{x}(t) + \mathbf{W}_2 \mathbf{x}'(t) + \mathbf{W}_3 \mathbf{x}''(t) \quad (9')$$

$$\mathbf{x}(t+\Delta t) = \mathbf{V}_0 + \mathbf{V}_1 \mathbf{u}(t+\Delta t) + \mathbf{V}_2 \mathbf{x}(t) + \mathbf{V}_3 \mathbf{x}(t-\Delta t) + \mathbf{V}_4 \mathbf{x}(t-2\Delta t) \quad (12')$$

$$\mathbf{u}(t+\Delta t) = \mathbf{Z}_0 + \mathbf{Z}_1 \mathbf{x}(t+\Delta t) + \mathbf{Z}_2 \mathbf{x}(t) + \mathbf{Z}_3 \mathbf{x}(t-\Delta t) + \mathbf{Z}_4 \mathbf{x}(t-2\Delta t) \quad (13')$$

donde  $\mathbf{W}_i$ ,  $\mathbf{V}_i$  y  $\mathbf{Z}_i$  son matrices que representan los pesos sinápticos de las redes. Como puede observarse, estas expresiones permiten determinar en forma directa la posición del sistema o las fuerzas requeridas al cabo de un intervalo de tiempo, según correspondan a un modelo directo o inverso. Además, igualando estas ecuaciones con las obtenidas a partir de las expresiones diferenciales se dispondrá de un medio apropiado para identificar los parámetros característicos del problema dinámico.

## 4.2 Sistemas No Lineales

Dentro del amplio espectro de formulaciones no lineales que pueden presentarse, hay algunos casos que son característicos en la representación de fenómenos mecánicos. Uno de estos casos es aquel en que los coeficientes de amortiguamiento dependen de la velocidad y posición del sistema, mientras que los elementos de la matriz de rigidez varían con la posición, y queda representado por la siguiente ecuación general:

$$\mathbf{M} \mathbf{x}'' + \mathbf{C}(\mathbf{x}', \mathbf{x}) \mathbf{x}' + \mathbf{K}(\mathbf{x}) \mathbf{x} = \mathbf{u} \quad (14)$$

Se trata de una ecuación no lineal por la presencia de términos en la que aparecen las velocidades y los desplazamientos multiplicados entre sí, como así también los cuadrados de estos últimos. Para resolverla, debe recordarse que toda función continua puede ser representada por una serie en la que aparece la propia función evaluada en un punto próximo y sus derivadas, como es el caso de la serie de Taylor. Si estas derivadas son a su vez expresadas con fórmulas de diferenciación numérica, se deduce que el comportamiento del sistema no lineal puede ser expresado por:

$$\mathbf{x}(t+\Delta t) = \mathbf{G} \mathbf{u}(t+\Delta t) + \mathbf{H}_0 \mathbf{x}(t) + \mathbf{H}_1 \mathbf{x}(t-\Delta t) + \dots + \mathbf{H}_n \mathbf{x}(t-n\Delta t) \quad (15)$$

y su modelo de dinámica inversa por

$$\mathbf{u}(t+\Delta t) = \mathbf{G}^{-1} [\mathbf{x}(t+\Delta t) - \mathbf{H}_0 \mathbf{x}(t) - \mathbf{H}_1 \mathbf{x}(t-\Delta t) - \dots - \mathbf{H}_n \mathbf{x}(t-n\Delta t)] \quad (16)$$



Las matrices  $\mathbf{G}$  y  $\mathbf{H}$  serán una combinación lineal de  $\mathbf{M}$ ,  $\mathbf{C}$  y  $\mathbf{K}$ , solo que difícilmente pueda conocerse la forma que toman estas relaciones ni la cantidad de términos necesarios para resolver cada caso. Aquí, como ocurrió con los sistemas lineales, se recurre a modelos neuronales para representar el comportamiento dinámico del sistema y se comprueban los siguientes aspectos particulares:

- Puede llegar a ser necesaria una red neuronal con una o más capas de unidades ocultas, según la severidad de la falta de linealidad del problema.
- Por este mismo motivo, puede ser necesario incorporar funciones de activación no lineales, tales como las funciones sigmoidales.
- No es fácil establecer relaciones entre las matrices de pesos sinápticos y las matrices  $\mathbf{G}$  y  $\mathbf{H}_i$  de los términos de la ecuación 15 o 16, por lo que los modelos neuronales no siempre conducirán a la identificación de los parámetros del sistema.

En estos casos, se recomienda comenzar con los mismos modelos utilizados para tratar problemas lineales similares, y darles progresivamente mayor generalidad según los resultados que se obtengan. Nunca se recomienda operar en sentido contrario, es decir comenzar adoptando modelos generales de elevada complejidad.

## 5 RESULTADOS

### 5.1 Caso presentado

Se seleccionó un sistema de segundo orden, lazo abierto y un grado de libertad con el fin de brindar ejemplos sobre las consideraciones anteriores. Para ello se entrenaron y validaron modelos neuronales directos e inversos que se corresponden con las ecuaciones (9), (12), (13) y (15). Para entrenar los modelos neuronales se utilizaron dos variantes de ese sistema, uno lineal y otro no lineal, y las respuestas correspondientes a tres condiciones en las que la fuerza aplicada tiene diferente variación en el tiempo. El comportamiento de los sistemas elásticos fue integrado por el método de Ruge Kutta de 4º orden, con un paso de integración de 0,01 seg. a lo largo de 25 seg. Una variante de este mismo método, de mayor orden, permitió comprobar que el error cometido estaba en un nivel aceptable. En la [Figura 2](#). se muestran las fuerzas aplicadas y las respuestas del sistema lineal.

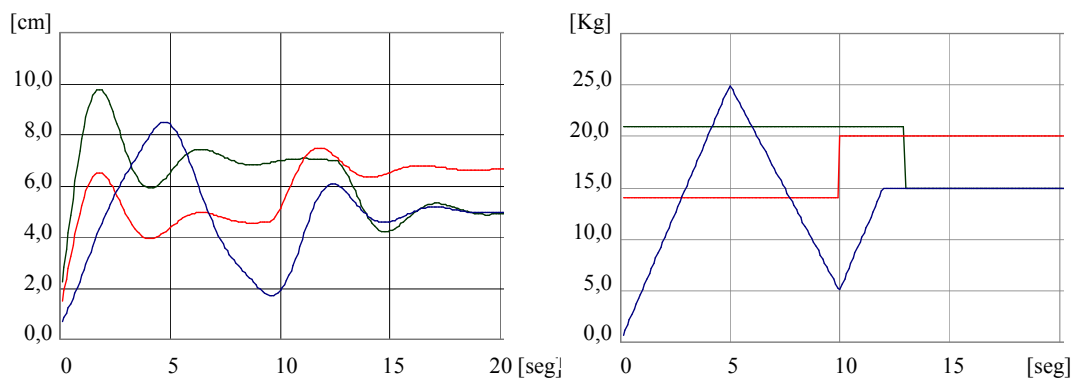


Figura 2 – Respuesta del sistema (izq) a las fuerzas aplicadas en escalón y en rampa (der)

En todos los casos, un perfil de fuerza y la correspondiente respuesta del sistema fueron tomados como conjunto de entrenamiento, mientras que los otros dos grupos de fuerzas-respuesta fueron utilizados como conjuntos de validación. Pudo comprobarse que podía emplearse cualquiera de los conjuntos para uno u otro fin, sin que esto incidiera en los resultados, lo que confirma que en el proceso de entrenamiento las redes aprendieron a reconocer las características del sistema. Es decir, no se trata de la memorización de un patrón de conducta conocido, sino de un verdadero proceso de identificación de parámetros.

Para disponer de un valor de comparación se calculó, en cada caso, la raíz media cuadrada (RMS) de las diferencias obtenidas durante el proceso de entrenamiento, promediándose los valores de las RMS correspondientes al conjunto de entrenamiento y al de validación. En todos los casos se limitó a 300 la cantidad de ciclos de ajuste.

Para el entrenamiento de las redes se implementó una versión básica del método de Backpropagation, con una única mejora que consistió en posibilitar el ajuste del factor de *aprendizaje*  $\eta$  durante el proceso de entrenamiento (Plagianakos, 2002). Por el contrario, al factor de *momentum*  $\beta$  se lo mantuvo constante. El objeto de emplear un algoritmo de entrenamiento tan básico fue el de disponer de la mayor sensibilidad para estudiar el desempeño de los diferentes modelos, ya que los algoritmos más sofisticados permiten disimular muchos de sus defectos.

En lo sucesivo, se hace referencia a resultados obtenidos con una unidad Adaline y se los compara con los de modelos más complejos de redes de perceptrones multicapa. Para estos últimos se emplea el identificador “Pmcx”, donde el sufijo  $x$  identifica la cantidad de unidades en la única capa oculta utilizada.

## 5.2 Modelo cinemático directo - Caso lineal

La ecuación (9) prevé la determinación de la próxima posición del sistema a partir de la última condición de desplazamiento, velocidad y aceleración conocida. Los resultados presentados se obtuvieron con una unidad Adaline y con tres variantes de redes Multicapa de Perceptrones, todas ellas con tres unidades de entrada. En el caso de las redes de perceptrones se adoptó una capa oculta con cinco, diez y quince unidades (Pmc5, Pmc10 y Pmc15), estudiándose casos con intervalos de tiempo entre 0,1 y 0,35 segundos.

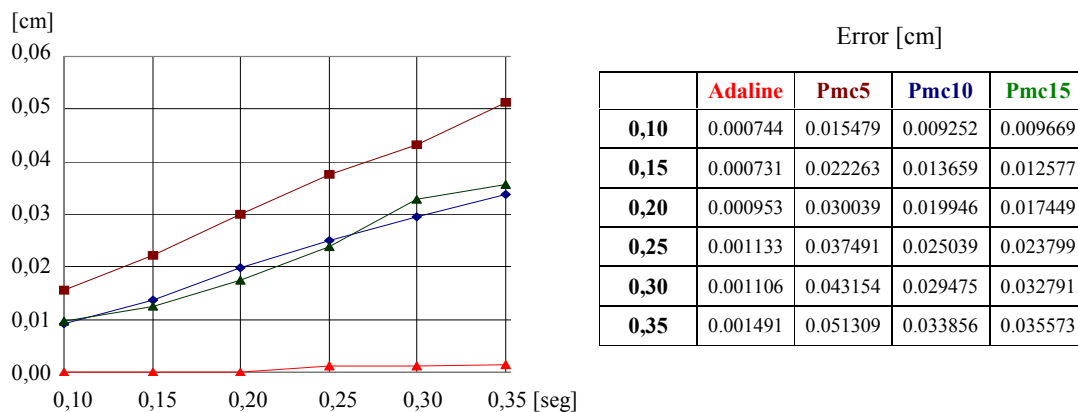


Figura 3 – Error medio cuadrático con distintos modelos e intervalos de tiempo

Este caso permitió comprobar el excelente desempeño de una unidad Adaline simple, donde sus resultados mostraron, además, una mayor independencia del intervalo de tiempo

que la obtenida con los otros modelos. En la [Figura 3](#) se muestra una tabla con los valores correspondientes al error medio cuadrático y su representación gráfica.

### 5.3 Modelo dinámico directo- Caso lineal

La ecuación (12) permite anticipar el próximo valor de desplazamiento a partir de la fuerza aplicada y valores anteriores del desplazamiento. Las fórmulas más elementales de diferenciación numérica requieren tres puntos para el cálculo de la derivada segunda, por lo que el segundo miembro de la ecuación (12) tendrá como mínimo tres términos (fuerza, desplazamiento actual y desplazamiento anterior). Por esta razón se estudiaron casos a partir de tres y hasta nueve unidades de entrada ( $N_e$ ). Para obtener los resultados presentados se usó una unidad Adaline y tres redes Multicapa de Perceptrones. Las redes de perceptrones tenían una capa oculta con siete, nueve y trece unidades (Pmc7, Pmc9 y Pmc13), y para todos los casos se consideró un intervalo de tiempo de 0,1 seg. Los resultados son tabulados y representados en la [Figura 4](#).

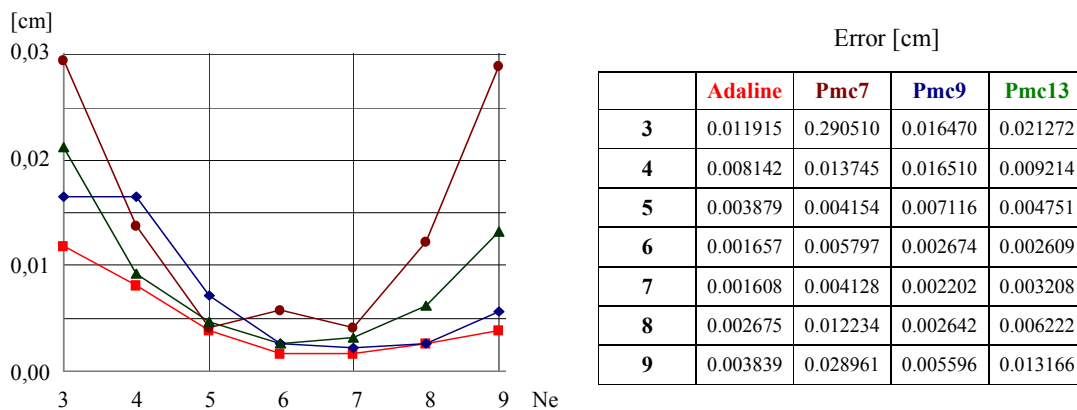


Figura 4 – Error obtenido con distintos modelos y cantidad de puntos anteriores

Este caso muestra que la unidad Adaline mantuvo su ventaja sobre las redes multicapa de perceptrones, aunque las diferencias entre los distintos modelos fue mucho menor que en el caso anterior. Es interesante notar que los mejores resultados se obtuvieron con cinco y seis unidades de entrada, lo que está en relación con la cantidad de términos que requieren las mejores fórmulas de diferenciación numérica para el cálculo de la derivada segunda. A partir de allí, la mayor complejidad de los modelos incidió negativamente en sus desempeños.

### 5.4 Modelos inversos - Caso lineal

Como ya fue anticipado, los modelos inversos tienen la finalidad de predecir la fuerza necesaria para alcanzar cierta condición cinemática. Esto puede hacerse con la ecuación (13) o recurriendo a la propia ecuación de equilibrio dinámico (ecuación 8). En este último caso, el proceso de entrenamiento de la red permite identificar con facilidad la masa, amortiguamiento y elasticidad del sistema.

Los modelos neuronales aquí empleados fueron una unidad Adaline y dos redes Multicapa de Perceptrones (Pmc5 y Pmc12), todas ellas con tres unidades de entrada. Se analizaron diferentes intervalos de tiempo, entre 0,1 y 0,35 seg, y los valores obtenidos se muestran en la [Figura 5](#).

Estos modelos confirmaron su aptitud para representar la dinámica inversa de un sistema y es necesario destacar el pequeño error obtenido en el cálculo de la fuerza, a pesar de tratarse

de casos en que se presentan cambios bruscos debido a escalones y rampas. En estos casos, la ventaja de la unidad Adaline se redujo conforme aumentó el intervalo de tiempo considerado.

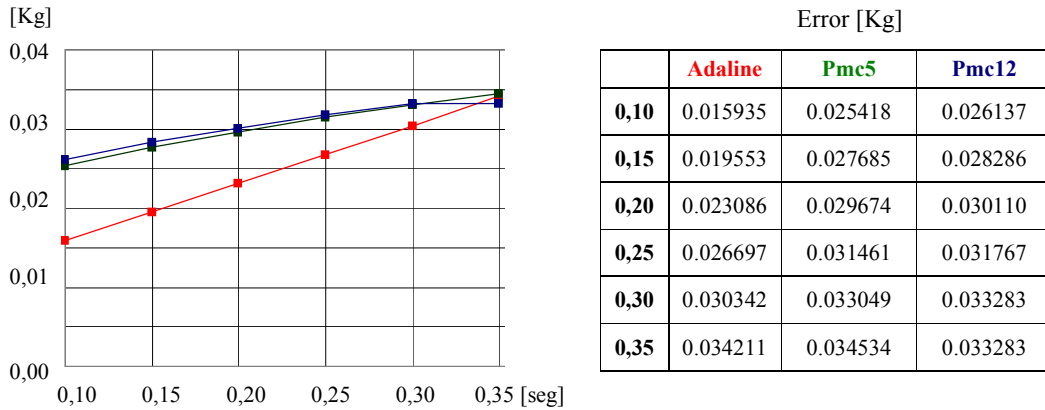


Figura 5 – Error medio cuadrático en el cálculo de la fuerza a ser aplicada

### 5.5 Modelo dinámico del caso no lineal

La respuesta de un sistema no lineal puede ser representada por una expresión tal como la ecuación (15), donde no puede anticiparse la cantidad de términos necesarios para mantener el error en niveles suficientemente bajos. Para este estudio se emplearon redes de perceptrones de una capa oculta con once y quince unidades (Pmc11 y Pmc15) y se consideró un intervalo de tiempo de 0,1 seg. Se presentan los resultados obtenidos en la Figura 6.

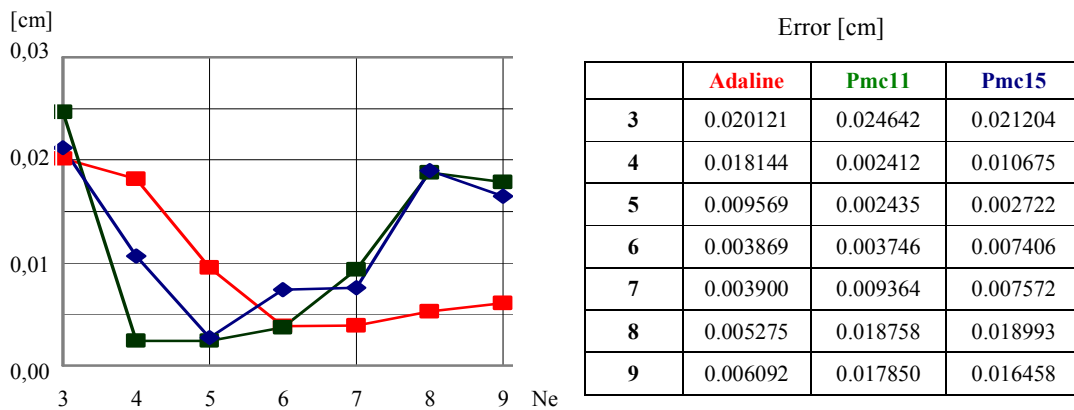


Figura 6 – Error obtenido con distintos modelos y cantidad de puntos anteriores

Obsérvese que la unidad Adaline y las redes de perceptrones demostraron muy buenos desempeños, ya que el nivel de error obtenido fue bajo a pesar de que el sistema estudiado presentaba una importante falta de linealidad. Sin embargo, como puede apreciarse, el comportamiento de los modelos neuronales fue más irregular que el obtenido al representar sistemas lineales. También debe mencionarse que otros modelos de perceptrones, con pocas unidades en la capa oculta, no alcanzaron a exhibir niveles suficientemente bajos de error en los ciclos de entrenamiento previstos.

## 5.6 Discusión de resultados

Los resultados obtenidos confirmaron la conveniencia de recurrir a modelos neuronales simples, que guarden directa relación con las expresiones que gobiernan los fenómenos representados. En todos los casos estudiados estos modelos simples tuvieron mejores desempeños que otros más complejos, con capas de unidades ocultas. Además, debe destacarse que las unidades Adaline presentaron menor sensibilidad a parámetros tales como el factor de aprendizaje  $\eta$  y el de momentum  $\beta$ , mientras que, por el contrario, las redes de perceptrones mostraron mayor sensibilidad a estos factores y obligaron a un ajuste más fino de sus valores. También se comprobó en todos estos casos la conveniencia de utilizar una función de activación lineal, aún en los modelos no lineales.

Como ya fue anticipado, estos desempeños contrastan con recomendaciones que se encuentran habitualmente en la literatura, que invitan a definir los modelos neuronales a partir de criterios heurísticos, prestando escasa atención al objeto de estudio.

## 6 CONCLUSIONES Y TRABAJO FUTURO

Si bien la conveniencia de emplear redes neuronales en la implementación de unidades de control está actualmente fuera de discusión, la falta de criterios objetivos y definitivos para la selección de tales modelos es fuente de incertidumbre. Esta situación aceptada como inevitable en muchos casos, no debería serlo, de modo alguno, en el estudio de aquellos problemas que responden a una formulación conocida.

En este trabajo se obtuvieron resultados que confirmaron las ventajas de recurrir al aporte del conocimiento sobre el fenómeno estudiado y de sus técnicas numéricas de resolución para orientar el desarrollo de los modelos neuronales, lo que implicó dejar de lado las recomendaciones heurísticas y de *caja negra*. En efecto, pudo comprobarse que con modelos muy simples es posible obtener aún mejores resultados que con otros más generales y complejos. Además, se posibilitó así un paralelismo entre los modelos neuronales y los modelos numéricos generales, lo que en muchos casos posibilitó la identificación de los parámetros del sistema. Esto permitió enfatizar la conveniencia de mantener los modelos tan simples como sea posible, confirmando que mayor complejidad no es necesariamente sinónimo de mejor desempeño.

Para continuar este trabajo se considerarán otros problemas de mayor complejidad y de varios grados de libertad, con la finalidad de demostrar que las ideas presentadas son generales y mantienen vigencia en todos los casos.

## REFERENCIAS

- Buckingham E. (1914). On physically similar systems: Illustration of the use of dimensional equations. *Phys. Review*, 4, 345-376.
- Beigi H. and Li C. (1990). A new set of learning algorithms for neural networks. *Proc. of ISMM Conference on Computer Applications in Design, Simulation and Analysis*, New Orleans, Louisiana.
- Branicky M. (1998). Multiple Lyapunov functions and other análisis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):475-482.
- DeCarlo R., Zak S. and Matthews G. (1988). Variable structure control of nonlinear multivariable systems: A tutorial. *Proceedings of the IEEE*, 76(3):212-232.

- Haecker J.; Rudolph S. (2001). On neural network topology design for nonlinear control, *Proceedings SPIE Aerosense 2001 Conference On Applications and Science of Computational Intelligence IV*, Orlando, Florida, April 16-20.
- Hertz J., Krogh A. and Palmer G. (1994). *Introduction to the theory of neural computation*. Addison-Wesley.
- Houbolt J. (1950). A recurrent matrix solution for the dynamic response of elastic aircraft. *Journal of Aeronautical Science*, Vol 17:540-550.
- Nguyen, D. and Widrow, B. (1990). Neural networks for self-learning control systems. *IEEE Control Systems Magazine*, 5:18-23.
- Pham D. and Xing L. (1997). *Neural networks for identification, prediction and control*. Springer Verlag, London.
- Plagianakos, Magoulas and Vrahatis (2002). Deterministic nonmonotone strategies for effective training of multilayer perceptrons. *IEEE Transactions on Neural Networks*, 13(6):1268-1284.
- Riedmiller, M. and Braun, H. (1993). A direct adaptive method for faster backpropagation learning: the Rprop algorithm. *Proceedings of the ICNN*, San Francisco.
- Rudolph S. (1998). On the Context of Dimensional Analysis in Artificial Intelligence, *Proceedings International Workshop on Similarity Methods*, University of Stuttgart, Germany, November 26-27, 147-162.
- Rumelhart D., Hinton G. and Williams R. (1986). Learning internal representations by error propagation. *Parallel distributed processing: Explorations in the microstructures of cognition*, MIT Press, I:318-362.
- Utkin V. (1977). Variable structure systems with sliding modes, *IEEE Transactions on Automatic Control*, AC-22(2):212-222.
- Valishevsky A. (1998). Comparative analysis of different approaches towards multilayer perceptron training, *Department of Computer Science*, University of Latvia, Riga.
- Ziegler J. and Nichols N. (1942). Optimum settings for automatic controllers, *Transactions of the ASME*, 64:759-768.