

## DEFINICIÓN Y EVALUACIÓN DE MODELOS NEURONALES PARA SISTEMAS DE CONTROL

Juan Giró<sup>1,2</sup>, Sandra Olariaga<sup>1</sup>, Nancy Paez<sup>1</sup>, Alejandro Garcia<sup>1</sup> y Alejandro Vuirli<sup>3</sup>

<sup>1</sup>*Departamento de Ingeniería de Sistemas de Información, Facultad Regional Córdoba,  
Universidad Tecnológica Nacional, <http://www.frc.utn.ar>*

<sup>2</sup>*Departamento de Estructuras, Facultad de Ciencias Exactas, Físicas y Naturales,  
Universidad Nacional de Córdoba, <http://www.efn.uncor.edu>*

<sup>3</sup>*Facultad de Ciencias Exactas, Físicas y Naturales,  
Universidad Nacional de Córdoba, <http://www.efn.uncor.edu>*

**Resumen:** Las redes neuronales artificiales ofrecen un enorme potencial que viene siendo muy bien aprovechado en el diseño de controladores de manipuladores robóticos y otras muchas aplicaciones. Sin embargo, persisten aún muchas dudas referidas a las arquitecturas y técnicas de entrenamiento que son más adecuados para obtener la mejor respuesta de estos modelos. En este trabajo se pone énfasis en la conveniencia de incorporar conocimiento sobre el problema tratado, como criterio central en la selección de los modelos neuronales más apropiados en cada caso. Se presentan luego los resultados obtenidos al evaluar redes simples destinadas a representar la dinámica directa e inversa de un dispositivo.

Keywords: Redes Neuronales, Adaline, Perceptrones, Neurocontroladores

### 1. INTRODUCCIÓN

Las unidades de control modernas son sistemas computacionales en los que el software viene adquiriendo cada vez mayor preponderancia, en muchos casos tomando la forma de sistemas embebidos. Paralelamente, se asiste a una creciente complejidad de los sistemas a controlar y cada vez mayores exigencias de autonomía y adaptabilidad. Esto es así en todos los campos de aplicación, y lo es muy especialmente en el caso de la robótica, donde se comprueba un proceso de gran difusión de manipuladores en aplicaciones industriales y de precisión.

Todos estos factores han llevado a replantear las técnicas de control tradicionales y al desarrollo de otras nuevas, que pueden enmarcarse dentro del llamado control inteligente. En este campo, se pueden citar los sistemas expertos, el control borroso, el control basado en algoritmos genéticos, las redes neuronales y diversas combinaciones entre ellos.

Concentrando la atención en las redes neuronales artificiales, es necesario señalar que son dos los aspectos que justifican el gran interés que despiertan 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.

En este trabajo se hace una breve referencia a los aspectos que deben ser considerados al seleccionar e implementar modelos neuronales, para luego concentrar la atención en los resultados obtenidos al representar la dinámica directa e inversa de un dispositivo a través de modelos simples. Para ello, se pone énfasis en la conveniencia de incorporar el conocimiento sobre el problema tratado como criterio central en la selección de los modelos neuronales más apropiados en cada caso.

## 2. 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. Los aspectos que deben considerarse son los siguientes:

### 2.1. 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). Nótese que el sobredimensionamiento de una red puede favorecer el proceso de aprendizaje y también resentir su capacidad de generalización.

### 2.2. Funciones de activación

Si bien la función de activación más utilizada es la sigmoïdal, en algunos casos es más conveniente la tangente hiperbólica o la función lineal. Pham ha comprobado la conveniencia de utilizar una u otras, o aún una combinación de ellas (híbridas), según sea el grado de no linealidad de la función representada (Pham, 1997).

### 2.3. Definición de pesos iniciales

Tradicionalmente, se asignan valores aleatorios a los pesos de una red para iniciar el proceso de entrenamiento, aunque 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.

### 2.4. Proceso de entrenamiento

Es indudable que el método de Backpropagation presentado por Rumelhart se convirtió en un clásico, al estimular una verdadera revolución en el campo de las redes neuronales (Rumelhart, 1986). 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.

## 3. EL PROBLEMA Y SUS SOLUCIONES

Se ha llegado así a establecer, en forma resumida, la conveniencia de emplear redes neuronales 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, en buena 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 su esclarecimiento, requiere 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, Kolmogorov, 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 un problema, como ocurre con las unidades de control, parecería recomendable

aplicar este conocimiento para definir modelos tan específicos como sea posible.

Las ecuaciones consideradas 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. Entre otras, torques externos, fuerzas de inercia, fricción, gravitatorias y las originadas en las aceleraciones centrípetas y de Coriolis. Si bien la complejidad geométrica de los dispositivos, las fuerzas de rozamiento de variada naturaleza y los accionamientos electromecánicos 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 representarlas.

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 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

### 3.1. Sistemas Lineales

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

$$M\ddot{x} + C\dot{x} + Kx = u \quad (1)$$

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:

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

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

$$\dot{x}_{t+\Delta t} = [11x_{t+\Delta t} - 18x_t + 9x_{t-\Delta t} - 2x_{t-2\Delta t}] / (6\Delta t) \quad (3)$$

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

Si se considera la ecuación de equilibrio dinámico (1) en el instante  $t+\Delta t$ , se reemplaza la aceleración y velocidad en las expresiones (3) y (4), y finalmente se agrupan los términos, se llega a una ecuación que suele ser atribuida a Houbolt y que tiene la siguiente forma general (Houbolt, 1950):

$$x_{t+\Delta t} = A^{-1}[u_{t+\Delta t} + Bx_t + Dx_{t-\Delta t} + Ex_{t-2\Delta t}] \quad (5)$$

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 (2) 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 (5) 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 (5), lo que conduce a:

$$u_{t+\Delta t} = Ax_{t+\Delta t} - Bx_t - Dx_{t-\Delta t} - Ex_{t-2\Delta t} \quad (6)$$

que puede también ser 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 2, 5 o 6:

$$x_{t+\Delta t} = W_0 + W_1 x_t + W_2 \dot{x}_t + W_3 \ddot{x}_t \quad (2')$$

$$x_{t+\Delta t} = V_0 + V_1 u_{t+\Delta t} + V_2 x_t + V_3 x_{t-\Delta t} + V_4 x_{t-2\Delta t} \quad (5')$$

$$u_{t+\Delta t} = Z_0 + Z_1 x_{t+\Delta t} + Z_2 x_t + Z_3 x_{t-\Delta t} + Z_4 x_{t-2\Delta t} \quad (6')$$

donde  $W_i$ ,  $V_i$  y  $Z_i$  son matrices que representan los pesos sinápticos de los modelos neuronales. 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.

### 3.2. 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:

$$M \ddot{x} + C(\dot{x}, x) \dot{x} + K(x) x = u \quad (7)$$

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:

$$x_{t+\Delta t} = G u_{t+\Delta t} + H_0 x_t + H_1 x_{t-\Delta t} + \dots + H_n x_{t-n\Delta t} \quad (8)$$

y su modelo de dinámica inversa por

$$u_{t+\Delta t} = G^{-1} [x_{t+\Delta t} - H_0 x_t - H_1 x_{t-\Delta t} - \dots - H_n x_{t-n\Delta t}] \quad (9)$$

Las matrices  $G$  y  $H$  son seguramente una combinación lineal de  $M$ ,  $C$  y  $K$ , 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 también ser necesario incorporar funciones de activación no lineales.
- No es fácil establecer relaciones entre las matrices de pesos sinápticos y las matrices de las ecuación 8 y 9, por lo que los modelos neuronales no siempre conducirán a la identificación de 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 con modelos de elevada complejidad.

## 4. RESULTADOS

Para brindar un ejemplo sobre las consideraciones anteriores se seleccionó un sistema elástico de un grado de libertad en dos variantes, una lineal y otra no lineal. Las respuestas de estos sistemas fueron integradas por el método de Runge Kutta de 4° orden, con un paso de

integración de 0,01 seg. a lo largo de 25 seg, y son representadas en la Figura 1.

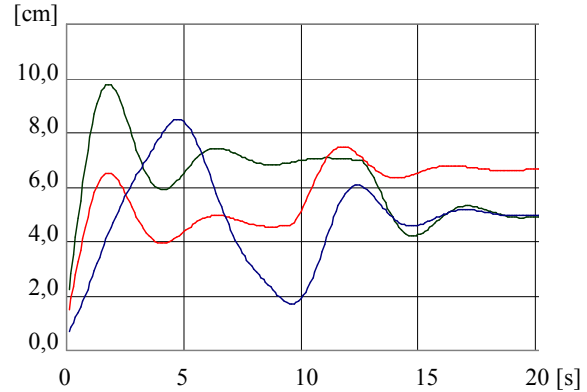


Figura 1. Respuestas del sistema elástico lineal ante las tres condiciones diferentes de carga utilizadas en los estudios.

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 "Pmxc", donde el sufijo x identifica la cantidad de unidades en la única capa oculta utilizada.

En el caso de un modelo cinemático directo lineal, se representó la ecuación (2) que prevé la determinación de la

próxima posición del sistema a partir de su última condición de desplazamiento, velocidad y aceleración. Las redes multicapa de perceptrones utilizadas tenían una capa oculta de cinco, diez y quince unidades. Todos los modelos tenían tres unidades de entrada y se usaron intervalos de tiempo entre 0,1 y 0,35 segundos.

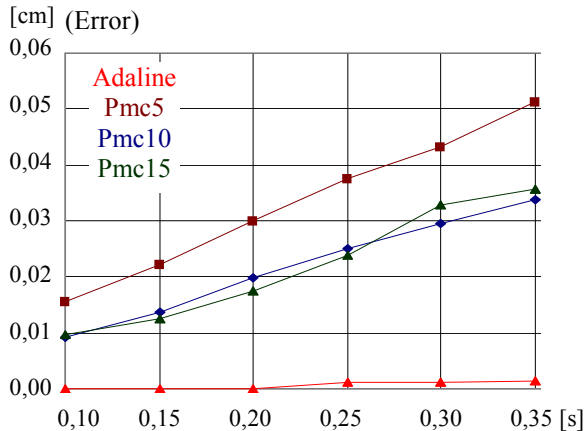


Figura 2. Errores cometidos por diferentes configuraciones de redes al reproducir el modelo cinemática lineal (Ec.2).

Los resultados representados en la Figura 2 permitieron comprobar el excelente desempeño de la unidad Adaline simple, donde su comportamiento muestra también una mayor independencia del intervalo de tiempo que el de los otros modelos.

El modelo dinámico directo, que representa la ecuación (5), está destinado a anticipar el próximo valor de desplazamiento a partir de la fuerza aplicada y valores anteriores del desplazamiento. Aquí se estudiaron casos con tres y hasta nueve unidades de entrada ( $N_e$ ), donde esta cantidad mínima proviene de las fórmulas más elementales de diferenciación numérica. En todos los casos se consideró un intervalo de tiempo de 0,1 seg y las redes de perceptrones tenían una capa oculta con siete, nueve y trece unidades. Los resultados son representados en la Figura 3.

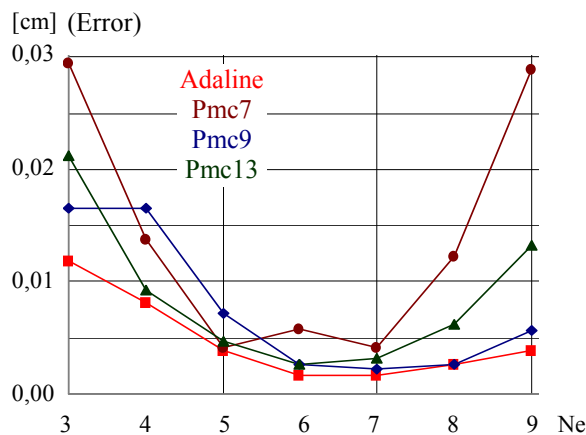


Figura 3. Errores al reproducir el modelo dinámico directo (Ec.5) con diferentes configuraciones de redes.

Este caso mostró que la unidad Adaline mantuvo su ventaja sobre las redes multicapa de perceptrones, aunque la 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.

Como ya fue anticipado, los modelos inversos tienen la finalidad de predecir la fuerza necesaria para alcanzar cierta condición cinemática. En este caso el modelo usado representó a la ecuación de equilibrio dinámico (1), donde el proceso de entrenamiento de la red permitió identificar la masa, amortiguamiento y elasticidad del sistema. Los modelos de perceptrones tenían una capa oculta con cinco y doce unidades, y se consideraron diferentes intervalos de tiempo, entre 0,1 y 0,35 seg. Se muestran los valores obtenidos en la Figura 4.

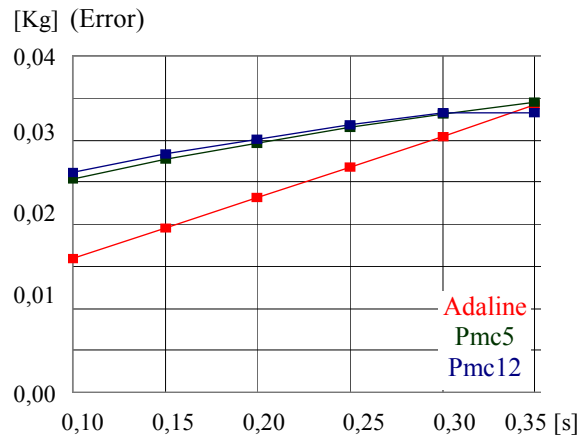


Figura 4. Diferencias obtenidas en el cálculo de las fuerzas aplicadas con el modelo dinámico inverso (Ec.1).

Aquí pudo confirmarse la aptitud de los modelos para representar la dinámica inversa de un sistema y, a pesar de los cambios bruscos que presentaban las fuerzas, éstas fueron determinadas con un error muy pequeño. El estudio muestra que la ventaja de la unidad Adaline se redujo conforme aumentó el intervalo de tiempo considerado.

Por último, para estudiar la respuesta de un sistema no lineal el modelo neuronal representó la ecuación (8), donde no puede anticiparse la cantidad de términos que serán necesarios. Se consideró un intervalo de tiempo de 0,1 seg. y los modelos de perceptrones tuvieron una capa oculta con once y quince unidades. Se representan los resultados en la Figura 5.

Obsérvese que la unidad Adaline y las redes de perceptrones mostraron muy buenos desempeños, a pesar de la importante falta de linealidad del sistema estudiado. Sin embargo, puede apreciarse que estas respuestas fueron más irregulares que las obtenidas con sistemas lineales.

Los resultados obtenidos confirman claramente la conveniencia de recurrir a modelos neuronales simples, que presenten directa relación con las expresiones que gobiernan los fenómenos representados. Además de estos buenos desempeños, las unidades Adaline presentaron menor sensibilidad a parámetros tales como el factor de aprendizaje  $\eta$  y el de momentum  $\beta$ . 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.

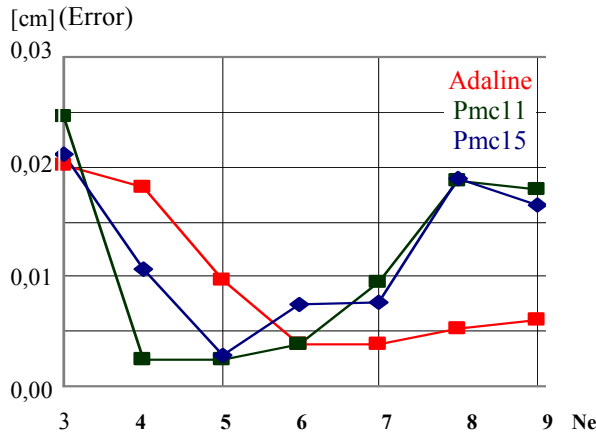


Figura 5. Representación del modelo no lineal (Ec.8) con diferentes configuraciones de redes: errores obtenidos.

## 5. CONCLUSIONES

La conveniencia de emplear redes neuronales en la implementación de unidades de control está actualmente fuera de toda discusión. Sin embargo, la ausencia de criterios objetivos y definitivos para la selección de tales modelos es siempre una fuente de incertidumbre. Si bien esto es inevitable en muchos casos, no debería serlo, de modo alguno, en el estudio de aquellos problemas que responden a una formulación conocida.

Así, 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. Además, se enfatizó la conveniencia de mantener los modelos tan simples como sea posible, ya que la mayor complejidad no es nunca sinónimo de mejor desempeño. Así quedó establecido un paralelismo entre los modelos numéricos generales y los modelos neuronales, que en muchos casos permite la identificación de los parámetros del sistema.

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.

## RECONOCIMIENTO

El trabajo presentado está inscripto en el proyecto "Evaluación de arquitecturas y técnicas de entrenamiento de Redes Neuronales Artificiales Multicapa de Perceptrones", que cuenta con el apoyo de la Agencia Córdoba Ciencia, del Gobierno de la Provincia de Córdoba (Expte. No. 0279-004549/2005).

## REFERENCIAS

- Buckingham E. (1914). On physically similar systems: Illustration of the use of dimensional equations. In: Phys. Review, 4, 345-376.
- Beigi, H. and Li, C. (1990). A New Set of Learning Algorithms for Neural Networks. In: Proc. of ISMM conference on Computer Applications in Design, Simulation and Analysis, New Orleans, Louisiana.
- Haecker J.; Rudolph S. (2001). On neural network topology design for nonlinear control, In: 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). In: Introduction to the Theory of Neural Computation. Addison-Wesley.
- Houbolt J. (1950). A recurrent matrix solution for the dynamic response of elastic aircraft. In: Journal of Aeronautical Science, Vol 17:540-550.
- Nguyen, D. and Widrow, B. (1990). Neural Networks for Self-Learning Control Systems. In: IEEE Control Systems Magazine, pp.18-23.
- Omidvar, O. and Elliott, D. (1997). Neural Systems for Control. Academic Press.
- Pham D. and Xing L. (1997). In: Neural networks for identification, prediction and control. Springer Verlag, London.
- Plagianakos, Magoulas and Vrahatis (2002). Deterministic Nonmonotone Strategies for Effective Training of Multilayer Perceptrons. In: IEEE Transactions on Neural Networks, Vol.13, No.6.
- Riedmiller, M. and Braun, H. (1993). A direct adaptive method for faster backpropagation learning: the Rprop algorithm. In: Proc. ICNN, San Francisco.
- Rudolph S. (1998). On the Context of Dimensional Analysis in Artificial Intelligence. In: Proceedings International Workshop on Similarity Methods, Univ. of Stuttgart, Germany, Nov. 26-27, 147-162.
- Rumelhart D., Hinton G. and Williams R. (1986). Learning internal representations by error propagation. In: Parallel Distributed Processing: Explorations in the Microstructures of Cognition, Vol.I, MIT Press, pp. 318-362.
- Valishevsky, A. (2001). Comparative Analysis of Different Approaches towards Multilayer Perceptron Training. In: Scientific Proceedings of Riga Technical University, pp 1-10.