

CONTROL ADAPTATIVO CON RED NEURONAL DE ELMAN Y MODOS DESLIZANTES

RESUMEN

Se presenta un esquema de control adaptativo, donde se emplea una red neuronal de Elman la cual identifica la planta con un aprendizaje por medio de planos deslizantes. Estos planos también sirven para actualizar los parámetros del controlador, lo cual le da robustez al sistema de control. El esquema es aplicable tanto a plantas lineales como no lineales.

PALABRAS CLAVES: Control adaptativo, red neuronal de Elman, modos deslizantes.

ABSTRACT

An Elman neural network is used in an adaptive control scheme. The network is used for the plant identification with sliding planes learning approach. The sliding planes are also used to actualize the controller parameters, therefore giving robustness to the control system. The scheme is applied to linear as non linear plants.

KEYWORDS: Adaptive control, Elman neural network, sliding modes.

DIDIER GIRALDO BUITRAGO

MEE Stevens Tech., USA
Facultad de Ingeniería Eléctrica
Universidad Tecnológica de Pereira
dgiraldo@utp.edu.co

JOSÉ GABRIEL HOYOS

Estudiante maestría en
Ingeniería Eléctrica - UTP
Profesor Tec. Electrónica
Universidad del Quindío
josega@telesat.com.co

1. INTRODUCCIÓN

El problema del diseño de controladores para plantas con parámetros variantes, incertidumbres en el modelo y perturbaciones externas puede ser tratado con la técnica del control por modo deslizante [1],[2],[3]. Recientemente varios autores han propuesto técnicas donde se combinan las redes neuronales y la técnica de modos deslizantes [4], [5].

La red neuronal recurrente de Elman [7], identifica y predice una muestra delante de la señal de salida, con esta señal se obtiene el error del modelo de la planta, el cual se utiliza para calcular el plano deslizante del modelo. El aprendizaje de la red, que aprende el modelo de la planta, lo realiza con la técnica de planos deslizantes, estos planos también sirven para actualizar los parámetros del controlador, el cual es una versión digital de un control PID.

2. ESQUEMA DE CONTROL NEURO-ADAPTIVO

Una vía para implementar un control con red neuronal es usando el principio de control un paso adelante. En éste, un predictor en lazo cerrado, monitorea las entradas y salidas

de la planta y predice la salida futura una muestra adelante.

El controlador un paso adelante, monitorea las entradas pasadas y las actuales, así como la salidas pasadas y manipula la entrada de control de la planta de tal forma que la predicción de la salida sea la salida deseada. El control un paso adelante lineal y su contraparte adaptativa, está descrita en Goodwin y Sin [8]. El control por red neuronal para plantas no lineales es similar. Un red neuronal como modelo de la planta es usada para predicción.

El controlador (usualmente otra red neuronal) monitorea entradas y salidas de la planta y manipula la entrada de control para calcular la predicción de la salida, basado en el modelo interno del sistema el cual fue identificado y almacenado por la red neuronal como modelo de la planta.

En la figura 1 se muestra el esquema utilizado, el cual se puede aplicar a plantas no lineales y con incertidumbres en los parámetros. Esta estructura es general en el sentido en que puede ser fácilmente extendida a otros diseños de control, como por ejemplo modelo de referencia, o podría adicionarse algún criterio de optimización, convirtiéndolo en control óptimo.

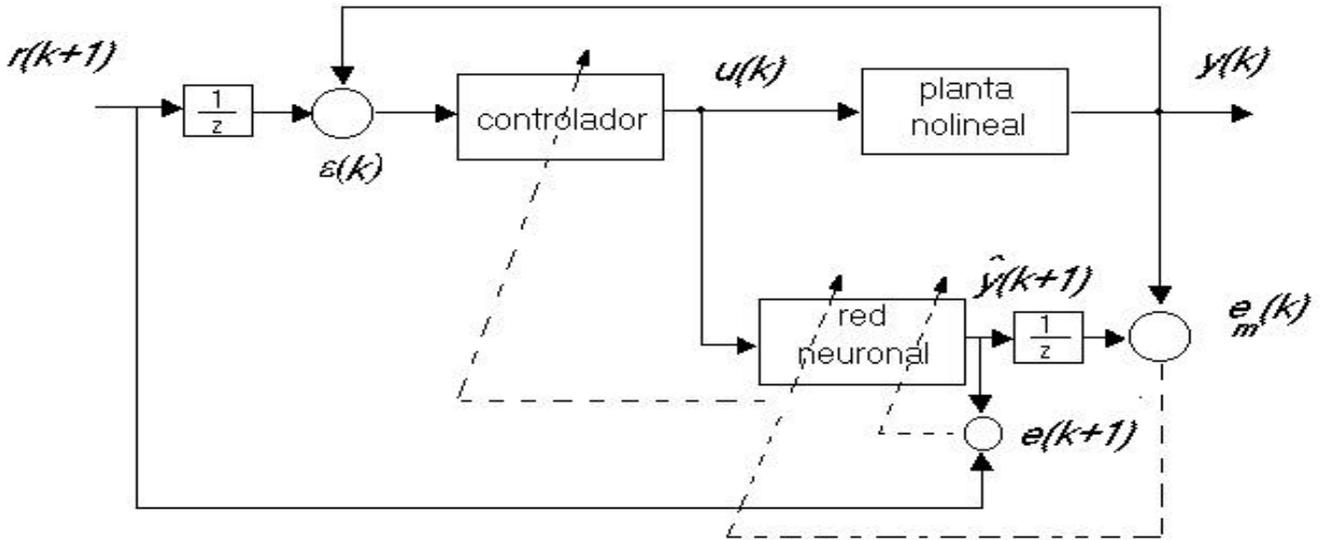


Figura 1. Esquema del sistema implementado.

Se debe notar que el error usado para la adaptación del controlador será el error de la propia señal de control (Error de comando); aunque éste normalmente no esta disponible, ya que no se tiene la señal de control deseada. En este artículo se estimará el error de comando y se llamará error de comando virtual, su cálculo se hará usando la predicción del modelo de la planta:

$$e_u(k+1) = \frac{\partial \hat{y}(k+1)}{\partial u(k)} e(k+1) \quad (1)$$

donde $e(k+1) = r(k+1) - \hat{y}(k+1)$ y $\partial \hat{y}(k+1) / \partial u(k)$ es un estimado del Jacobiano del sistema, el cual se obtiene usando la red neuronal como modelo.

3. MODELO PREDICTIVO DE LA PLANTA

Las redes neuronales pueden ser clasificadas como unidireccionales y recurrentes. En las redes unidireccionales, las neuronas están conectadas de tal forma que las señales fluyen en una sola dirección de la entrada hacia la salida. En las redes recurrentes existen conexiones tanto hacia delante como hacia atrás a lo largo de las cuales se propagan las señales en direcciones opuestas.

Las redes unidireccionales han sido aplicadas a la identificación con éxito, aunque con algunos inconvenientes como gran numero de neuronas en la capa de entrada, lo que conlleva a susceptibilidad al ruido y aprendizaje lento. Debido a su estructura, las redes recurrentes no sufren de los anteriores inconvenientes. La red de Elman [7] es uno de los tipos más simples de red recurrente la cual puede ser entrenada también usando el algoritmo estándar de retro-propagación.

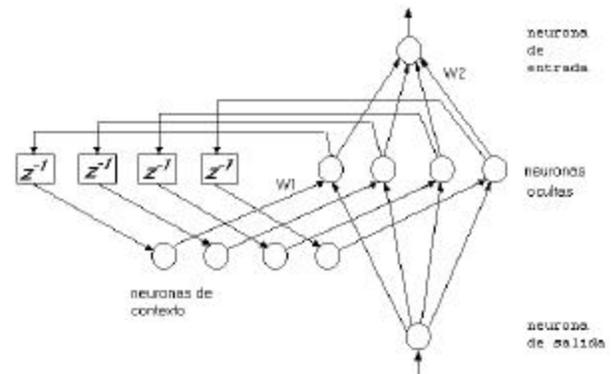


Figura 2. Red de Elman modificada.

La figura 2 muestra la red de Elman modificada. Ésta se utiliza para obtener el modelo de la planta y predecir su salida.

La estructura de la red consiste de una neurona de entrada, una de salida, un número de neuronas ocultas e igual número de neuronas de contexto. Las salidas de las neuronas ocultas son conectadas a las unidades de contexto, lo cual puede ser considerado como unidades de memoria de un paso de retardo.

Las salidas de las neuronas de contexto hacen parte de las entradas de las neuronas ocultas. Solo a las conexiones hacia adelante se les modificarán los pesos y los pesos de las conexiones recurrentes (de las ocultas a las de contexto) serán iguales a uno. Así como un sistema dinámico representa una relación entre la salida presente y las entradas pasadas, la introducción de realimentación en las neuronas de contexto representa la posibilidad de la red de Elman de modelar sistemas.

Al principio del entrenamiento, las salidas de las neuronas de contexto son inicializadas en cero (para función tangente hiperbólica) o en 0.5 (para sigmoïdal). La salida de la neurona j de contexto es:

$$x_{cj}(k+1) = \alpha x_{cj}(k) + x_j(k) \quad (2)$$

donde $x_{cj}(k)$ y $x_j(k)$ son las salidas de la neurona j-ésima de contexto y oculta respectivamente y α es la ganancia de realimentación.

El valor de α debe estar entre 0 y 1 y se tomará fijo para todas las realimentaciones.

Es de anotar que el comportamiento de la red de Elman depende altamente del valor escogido para α .

La salida de la red neuronal puede ser determinada como:

$$\hat{y}(k+1) = f \left\{ \sum_{i=1}^q W2_i f_H \left[\sum_{j=1}^n W1_{ij} I_j(k) \right] \right\} \quad (3)$$

Donde q es el número de neuronas en la capa oculta (= 4), n es la suma de el número de entradas (= 1) y cuatro es el número de unidades de contexto, $W1_{ij}$ es el peso de la neurona i en la capa oculta a la entrada j, $W2_i$ es el peso de la neurona oculta a la de salida, $I(k)$ es un conjunto de datos consistente de la señal de control $u(k)$ y las salidas de las neuronas de contexto en el mismo instante k, $f_H(\cdot)$ es la función de activación para el nodo oculto i, y $f(\cdot)$ es la función de activación lineal del nodo de salida.

El Jacobiano estimado del sistema es usado para calcular el error de comando "virtual" el cual puede ser determinado por la aplicación directa de la regla de la cadena:

$$\begin{aligned} \frac{\partial \hat{y}(k+1)}{\partial u(k)} &= \sum_{i=1}^q \frac{\partial \hat{y}(k+1)}{\partial f_H(x_i)} \frac{\partial f_H(x_i)}{\partial x_i} \frac{\partial x_i}{\partial u(k)} \\ &= \sum W2_i [1 - f_H(x_i)] f_H'(x_i) W1_{i1} \end{aligned} \quad (4)$$

con $f_H(\cdot)$ la función sigmoïdal.

Si $f_H(\cdot)$ es la función tangente hiperbólica, (4) se convierte en:

$$\frac{\partial \hat{y}(k+1)}{\partial u(k)} = \sum_{i=1}^q W2_i [1 - f_H^2(x_i)] \left(\frac{1}{2} \right) W1_{i1} \quad (5)$$

4. CONTROLADOR

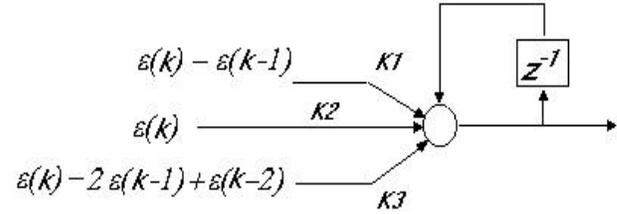


Figura 3. Controlador.

Un controlador del tipo PID adaptativo es utilizado y se muestra en la figura 3. Es posible además implementar otros controladores.

La señal de control $u(k)$ se calcula como:

$$u(k) = u(k-1) + K_1(k)h_1(k) + K_2(k)h_2(k) + K_3(k)h_3(k) \quad (6)$$

donde $K1, K2, K3$ son los parámetros del controlador $e(k) = r(k) - y(k)$ es el error de realimentación,

$$h_1(k) = e(k) - e(k-1)$$

$$h_2(k) = e(k)$$

$$h_3(k) = e(k) - 2e(k-1) + e(k-2)$$

y $y(k), r(k)$ son la salida real y la señal deseada en el instante k.

5. ALGORITMO DE APRENDIZAJE POR MODO DESLIZANTE

Si el entrenamiento nunca es desactivado, el esquema de la figura 1 podría ser en principio un controlador adaptativo para sistemas variantes en el tiempo.

El diseño del control por planos deslizantes se divide en dos fases. En la primera, una superficie deslizante se define para producir un comportamiento entrada/salida. En la segunda, los pesos son actualizados de forma que satisfagan las condiciones de seguimiento en la superficie deslizante. Para la capa de salida del modelo y para la salida del controlador, las superficies deslizantes son definidas en (7) y (8) respectivamente:

$$S_m = \dot{e}_m + I_1 e_m \quad (7)$$

$$S_c = \dot{e}_u + I_2 e_u \quad (8)$$

donde $e_m(k) = y(k) - \hat{y}(k)$ es el error del modelo y $I_1, I_2 > 0$, son constantes.

Para la capa oculta de la red neuronal como modelo, la superficie es definida como:

$$S_H = \dot{e}_H + I_H e_H \quad (9)$$

donde $e_H = (1/2)e_m^2$ y $I_H > 0$.

Y considerando que las funciones de activación de ambas estructuras son lineales, la adaptación de los pesos para la red neuronal como modelo y el controlador puede ser obtenida por (10) y (11):

$$\Delta W_{2_i} = \mathbf{a}_1 \text{sign}(S_m) |e_m| f_H(x_i) \quad (10)$$

$$\Delta K_{1,2,3} = \mathbf{a}_2 \text{sign}(S_c) |e_u| h_{1,2,3} \quad (11)$$

con $\mathbf{a}_1, \mathbf{a}_2 > 0$.

La actualización para la capa oculta es dada en (12) para una función de activación tangente hiperbólica:

$$\Delta W_{1_{ij}} = \frac{\mathbf{b} \text{sign}(S_H) |e_H|}{\frac{1}{2}(e_m W_{2_i} + \mathbf{h}) [1 - f_H^2(x_i)]} I_j \quad (12)$$

donde \mathbf{h} es una pequeña constante para prevenir la división por cero.

Para prevenir el efecto de “chattering”, se puede utilizar la siguiente fórmula, en vez de la función signo:

$$\mathbf{u}(s) = \frac{S}{|S| + \mathbf{d}} \quad (13)$$

donde \mathbf{d} es una constante positiva pequeña.

Lo ideal sería tener I_1, I_2 grandes para garantizar convergencia rápida, pero esto puede implicar problemas de convergencia e inestabilidad. Por esto los parámetros I_1, I_2 , se deben calcular dentro de unos límites de acuerdo a (14) y (15):

$$I_1 = \max \left(\text{abs} \left\{ -\frac{1}{f_H(x_i)} \frac{\partial f_H(x_i)}{\partial t} \frac{|e_m|}{|e_m|} \right\} \right) \quad (14)$$

$$I_2 = \max \left\{ \frac{|\dot{e}_u|}{|e_u|} \right\} \quad (15)$$

6. SIMULACIONES

Se implementaron varias simulaciones para plantas lineales y no lineales, obteniendo buenos resultados, por ejemplo, para la planta no lineal tomada de [6]:

$$\dot{y}(t) + \text{par1} * |y(t)| y(t) = u(t) \quad (16)$$

y en su versión discreta:

$$y(k) = y(k-1) + T \cdot u(k-1) - T \cdot \text{par1} \cdot |y(k-1)| \cdot y(k-1) \quad (17)$$

Donde par1 es 0.8 para $0 \leq k \leq 900$ y 1.7 para $k > 900$.

En la figura 4 se muestra la señal de referencia y la salida de la planta. La señal de referencia es una función seno, $r(k) = \sin(2\pi k / 500)$, nótese el cambio en $k=900$ debido a la variación en par1 .

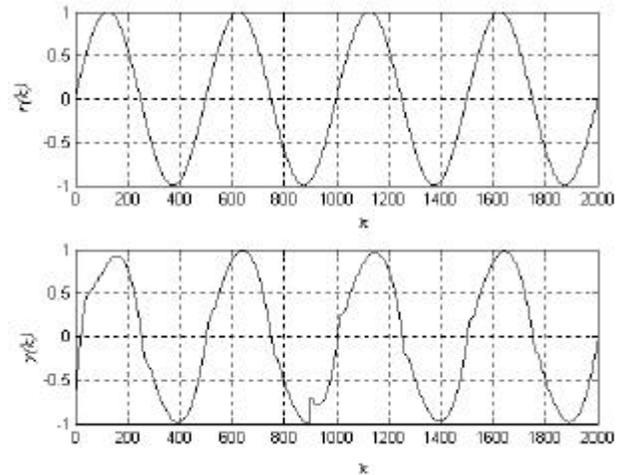


Figura 4. Señal de referencia y salida de la planta.

Para la anterior planta, se muestra también el valor absoluto del error del modelo (figura 5), la señal de control (figura 6) y los cambios de los parámetros del controlador (figura 7).

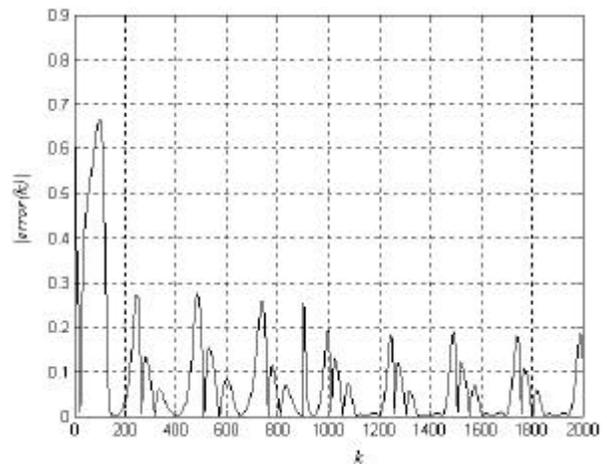


Figura 5. Error del modelo.

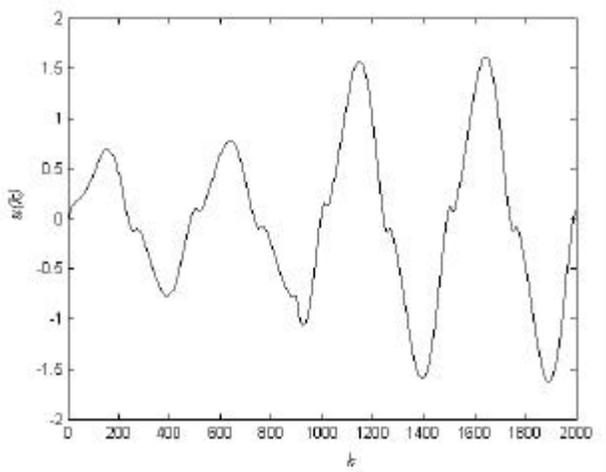


Figura 6. Señal de control.

En la figura 6 se observa cómo la señal de control aumenta para compensar el cambio en $par1$ y los parámetros del controlador (figura 7) presentan cambios en su pendiente.

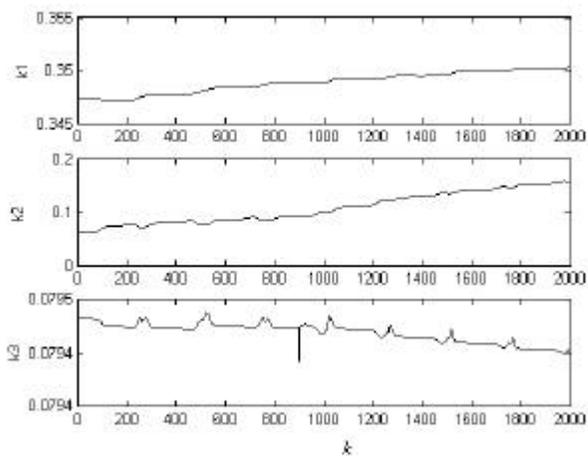


Figura 7. Variación de los parámetros del controlador, $k1$ (acción derivativa), $k2$ (acción proporcional), $k3$ (acción integral).

7. CONCLUSIONES

El esquema mostrado presenta un buen comportamiento en el control adaptativo de varias plantas, tanto lineales como no lineales. Aunque las redes recurrentes pueden presentar problemas de estabilidad, la técnica de planos deslizantes, le da robustez al control y elimina la inestabilidad, esto último faltaría por ser demostrado, pero a través de las simulaciones realizadas se puede ver que el esquema presenta buena estabilidad.

8. BIBLIOGRAFÍA

- [1] UTKIN V. I., Sliding Modes and Their Application in Variable Structure Systems, MIR Publishers, Moscow, 1978.
- [2] HUNG J.Y., W. GAO, J.C. HUNG, Variable Structure Control: A Survey, IEEE Trans. On Industrial Electronics, vol 40, N°1, pp. 2-22, 1993.
- [3] D. GIRALDO, I. TABARES, El Problema de Seguimiento en Modo Deslizante, SCIENTIA ET TECHNICA, No. 4, pp 43-49, octubre 1996.
- [4] TOPALOV A.V., O. KAYNAK, Online Learning in Adaptive Neurocontrol Schemes with a Sliding Mode Algorithm, IEEE Trans. On Sys. Man and Cybernetics, vol 31, no. 3, part.B, pp. 445-450, 2001.
- [5] YURKOVICH. HUNG, Variable Structure Control: A Survey, IEEE Trans. On Industrial Electronics, vol 40, N°1, pp. 2-22, 1993.
- [6] SLOTINE J.J., W. LI, Applied Nonlinear Control, Prentice Hall, 1991.
- [7] ELMAN J.L., Finding Structure in Time, Cognitive science, no. 14, 1990.
- [8] GOODWIN G.C., K.S. SIN, Adaptive filtering prediction and control., Prentice Hall, 1984.