

## CONTROL NEURONAL PREDICTIVO POR LINEALIZACIÓN INSTANTÁNEA

Rosalba Lamanna<sup>1</sup>, Raquel Gimón<sup>1</sup>

<sup>1</sup>Universidad Simón Bolívar. Departamento de Procesos y Sistemas. Caracas 1081A, Venezuela.

**Resumen:** Se desarrolla un esquema de Control Predictivo Generalizado (GPC) basado en un modelo neuronal del proceso, y se aplica en un reactor de neutralización de laboratorio. El modelo neuronal se obtiene previamente por identificación, y se linealiza en cada iteración del algoritmo de control. De esta manera se combina la capacidad de ajuste de la redes neuronales con la eficiencia de cálculo del GPC, consiguiéndose un sistema con las ventajas del control predictivo pero extendido a sistemas no lineales, que se comporta con precisión y rapidez ante perturbaciones. Copyright © 2007 CEA-IFAC

**Palabras clave:** Control Neuronal, Control Predictivo, Redes Neuronales, Linealización Instantánea, Control de pH..

### 1. INTRODUCCIÓN

Como es bien sabido, son muchas las aplicaciones de las redes neuronales en ingeniería. En especial en el modelado y en el control de procesos, las redes neuronales son elementos esenciales en muchos esquemas de control inteligente, ya que por su naturaleza no lineal y por su capacidad de aprendizaje, permiten la extensión de técnicas avanzadas de automatización a sistemas complejos y de dinámicas inciertas y/o variantes (Hunt *et al.*, 1992; Nørgaard, 1998; Lin *et al.*, 1999). Los sistemas de control que utilizan redes neuronales en su diseño son generalmente esquemas basados en modelo. En ellos la red neuronal puede representar el modelo del proceso que se controla (esquemas de Diseño Indirecto), o bien la red puede ser directamente el controlador (esquemas de Diseño Directo).

Este trabajo, enmarcado en un proyecto de investigación sobre el comportamiento de diversos esquemas de control neuronal, se dedica al desarrollo de un control neuronal predictivo (de

diseño indirecto) que luego se aplica a la regulación del pH en un reactor de laboratorio.

En la literatura se presentan diversas estrategias que han sido aplicadas para solucionar el problema de control de pH, las cuales van desde controladores lineales clásicos hasta controladores adaptativos basados en modelos no lineales de diferentes tipos (Ali, 2001; Waller y Toivonen, 2002; Galindo, Margaglio y Lamanna, 2000). El Control Predictivo Basado en Modelos (CPBM) (Camacho y Bordons, 1995; García, Prett y Morari, 1989) ha alcanzado un nivel muy significativo de aceptabilidad industrial en aplicaciones prácticas de control de procesos. Es una estrategia de control que se basa en el cálculo de las variables manipuladas por optimización de una cierta función de costo, en donde se utiliza de forma explícita un modelo (lineal) del proceso, para predecir el valor de las variables controladas a lo largo de un horizonte temporal especificado por el usuario, calculándose el valor de las variables manipuladas para hacer que en este horizonte, las variables controladas estén en sus valores de referencia.



Desde el punto de vista de aplicaciones industriales el *CPBM*, tiene algunas ventajas importantes sobre otras técnicas de control, que van desde la posibilidad de tratar en forma sencilla sistemas multivariados con distinto número de entradas y salidas, y de introducir restricciones en las variables de entrada y salida, hasta el hecho de que es conceptualmente simple de comprender, y que puede utilizarse en sistemas de dinámica difícil, respuesta inversa, etc.

La idea del esquema de control propuesto en este trabajo es usar el algoritmo de Control Predictivo Generalizado (Clarke, Mohtadi y Tuffs, 1987), aprovechando todas sus ventajas, pero adaptándolo a sistemas no lineales, mediante el uso de un modelo no lineal del proceso, realizado con una red neuronal. Esto es posible sometiendo a la red neuronal a un proceso de linealización en cada iteración del algoritmo de control. Así se elimina el problema de optimización no lineal iterativa que hace engorrosa o poco viable la aplicación en tiempo real de muchos de los esquemas de control neuronal predictivo propuestos en la literatura (Liu *et al.*, 1998; Zamarreño y Vega, 1999; Lamanna *et al.*, 2000).

Las pruebas en tiempo real del comportamiento del algoritmo desarrollado, se realizan en un sistema de control de pH constituido por un reactor continuo, donde ocurre la neutralización de una base débil con un ácido fuerte a escala de laboratorio. Si bien es cierto que este proceso es una simplificación de la típica situación industrial de control de pH donde se tratan fluidos que son mezclas de muchos ácidos y bases diferentes, el mismo constituye un sistema no lineal interesante y a la vez económico de implantar y de operar.

El presente documento está organizado en cinco secciones. En la sección 2 se realiza una breve descripción de la planta de neutralización. En la sección 3 se obtiene un modelo neuronal del sistema por identificación a partir de datos de operación. En la sección 4 se desarrolla la propuesta de un algoritmo de GPC que utiliza como modelo una linealización instantánea de la red neuronal identificada. Finalmente, en la sección 5 se presentan los resultados obtenidos en el control de una planta de neutralización de laboratorio. Las conclusiones se resumen en la sección 6.

## 2. DESCRIPCIÓN DE LA PLANTA

El sistema de reacción química, que se muestra en la figura 1, está constituido por un reactor continuo e isotérmico, donde ocurre un proceso de neutralización en medio acuoso. Los reaccionantes son: una base débil (Acetato de Sodio), que constituye el fluido a neutralizar, y que se alimenta

al reactor con flujo de entrada  $F_b$ , y un ácido fuerte (Ácido Clorhídrico), que es el agente de titulación, cuyo flujo de entrada  $F_a$ , se manipula para lograr el pH requerido.

Figura 1. Sistema de Neutralización de pH

El reactor es de volumen constante y de composición homogénea. Los instrumentos más importantes son las bombas dosificadoras de cada flujo de entrada de reaccionantes, que permiten su regulación ya sea de forma manual o por medio de señales eléctricas; y el medidor-transductor de pH (pH-ímetro) que utiliza un electrodo sumergido en la mezcla, y que produce una señal, también eléctrica, proporcional al pH. Equipos accesorios son dos tanques de almacenamiento de los reactivos y un tercer tanque para la recolección de los productos de la reacción.

## 3. IDENTIFICACIÓN DEL MODELO NEURONAL

Se considera la familia de modelos no lineales de una entrada una salida de la forma:

$$y(t) = g[y(t-1), \dots, y(t-n); u(t-1), \dots, u(t-m)] \quad (1)$$

donde  $g$  es una función no lineal desconocida que se desea aproximar.

Para la realización de la función “ $g$ ” se escogen redes neuronales multicapa no retroalimentadas, por su flexibilidad y sus excelentes propiedades de aproximación. El uso de estas redes para identificación, se remonta a más de una década (Norgaard, 1998) y hay numerosos trabajos publicados sobre estas aplicaciones (Narendra y Parthasarathy, 1990; Lin *et al.* 1999).

El predictor neuronal a usar es entonces:

$$\hat{y}(t) = g[\varphi(t)] \quad (2)$$

donde:

$\varphi(t)$  es el vector de entradas:

$$\varphi(t) = [y(t-1), \dots, y(t-n); u(t-1), \dots, u(t-m)] \quad (3)$$



$y(t) = pH(t)$  (variable de salida) y  
 $u(t) = V_a(t)$  (variable manipulada: voltaje  
 equivalente al flujo de ácido).

Las redes neuronales multicapa son diseñadas y ajustadas con ayuda del *Neural Networks Toolbox* de Matlab®, Versión 7.1 Release 12. Se realizan numerosos diseños variando el vector de entradas de la red y las amplitudes de las entradas, el número de neuronas de la capa intermedia y los propios algoritmos de aprendizaje supervisado de la red.

La conformación del vector de entrada (número de entradas y salidas anteriores) se determina durante el proceso de identificación. Para ello se calculan los cocientes de Lipschitz para diversas combinaciones de datos entrada-salida en el vector de entradas (He y Asada, 1993). Estos cocientes tienen valores grandes cuando los datos de entrada son insuficientes y disminuyen al agregar más datos de instantes anteriores al vector de entradas. Por otra parte, los cocientes de Lipschitz no disminuyen significativamente si se agregan al vector de entradas más datos de los necesarios. Finalmente se obtiene:

$$\varphi(t) = [pH(t-1) \quad V_a(t-1)]$$

La arquitectura interna de la red se toma finalmente como de una sola capa intermedia, con funciones de activación  $[f_1(\cdot), \dots, f_{NI}(\cdot)]$  tangente hiperbólica y una capa de salida  $[F(\cdot)]$  lineal, de manera que el modelo neuronal queda:

$$pH_{est}(t) = \sum_j W_j \tanh\left(\sum_k w_{jk} \varphi_k(t) + w_{j0}\right) + W_0 \quad (4)$$

Los parámetros que deben ser estimados corresponden a los pesos o coeficientes ( $w_{jk}$ ) que ponderan las entradas de la red neuronal ( $\varphi$ ) y los pesos ( $W_j$ ) que ponderan la entrada a la capa de salida de la red (ver figura 2).

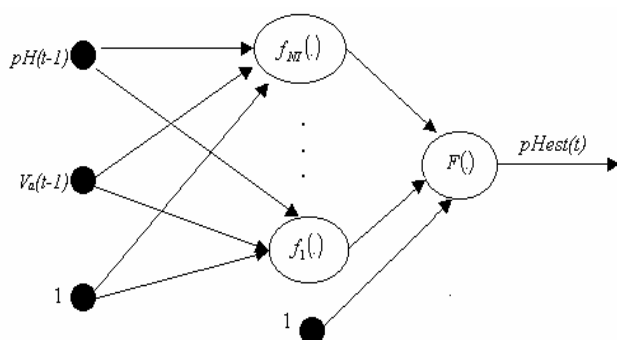


Figura 2 Red neuronal de identificación directa

Los datos entrada-salida necesarios para la identificación se obtienen con las variaciones de la

salida del proceso ( $pH$ ) ante cambios sinusoidales en la entrada principal ( $V_a$ ) con amplitudes que van desde 2V hasta 5V, en los dos puntos de operación que se muestran en la tabla 1 (el flujo de base se considera como una posible variable de perturbación).

Tabla 1 Condiciones en los puntos de operación

Prueba	Reactivo	Concentración (Molar)	Flujo (ml/min)
1	CH <sub>3</sub> COOH	0.019	82.1
	HCl	0.002	53.2
2	CH <sub>3</sub> COOH	0.018	48.1
	HCl	0.005	53.2

En la tabla 2 se muestran resultados de la identificación usando diferentes algoritmos de ajuste de las redes neuronales, reportándose el error cuadrático medio obtenido sobre el conjunto de datos de entrenamiento, EQMA, y el error obtenido utilizando conjuntos de datos nuevos o de prueba, EQMV, que se usan para la validación de las redes. Estos errores, llamados EQMA error de ajuste, y EQMV error de validación, se definen como:

$$EQMA = \frac{1}{N} \sqrt{\sum_{i=1}^N (y_{Ai} - \hat{y}_i)^2} \quad (5)$$

$$EQMV = \frac{1}{M} \sqrt{\sum_{i=1}^M (y_{Vi} - \hat{y}_i)^2} \quad (6)$$

donde:

$N$ : Número de elementos del conjunto de datos de entrenamiento.

$y_{Ai}$ : Salidas del conjunto de entrenamiento.

$M$ : Número de elementos del conjunto de datos de prueba.

$y_{Vi}$ : Salidas del conjunto de prueba.

Tabla 2 Resultados obtenidos a partir de diferentes tipos de aprendizaje

Prueba	Métodos	EQMA	EQMV
1	Gradiente	0.2%	0.24%
	BFGS	0.019%	0.015810%
	Levenberg Maquardt	0.018%	0.015807%
2	Gradiente	0.06%	0.06%
	BFGS	0.003%	0.002%
	Levenberg Maquardt	0.003%	0.004%

La comparación entre la disminución relativa de los errores de ajuste y de validación permite también determinar el número óptimo de neuronas de la red en su capa intermedia ( $NI$ ). Los resultados



de las numerosas pruebas realizadas no se muestran aquí por cuestión de espacio, pero la mejor estructura para la red de identificación resultó ser dos nodos de entrada (2), ocho (8) nodos en la capa intermedia y un (1) nodo de salida.

#### 4. CONTROL NEURONAL PREDICTIVO POR LINEALIZACIÓN INSTANTÁNEA

##### 4.1 Linealización Instantánea de un Modelo Determinístico

Para adaptar a procesos no lineales el esquema de Control Predictivo Generalizado (GCP, Clarke, Mohtadi y Tuffs, 1987) se propone en esta sección efectuar en línea con el sistema de control una linealización del modelo neuronal utilizando la técnica de *linealización instantánea* propuesta por Sørensen (1994).

Valores estimados de la salida de la planta pueden ser obtenidos a partir del modelo neuronal dado en la ecuación (2).

Para obtener la linealización de la función,  $g(\cdot)$ , se toma el vector de regresión general:

$$\varphi(t) = [y(t-1), \dots, y(t-n), u(t-d), \dots, u(t-d-m)]^T \quad (7)$$

y se interpreta como el “estado” del sistema a linealizar, y luego, la linealización de  $g$  alrededor del valor actual  $\varphi(t)$  será:

$$\tilde{y}(t) = -a_i \tilde{y}(t-1) - \dots - a_n \tilde{y}(t-n) + b_0 \tilde{u}(t-d) + \dots + b_m \tilde{u}(t-d-m) \quad (8)$$

donde:

$$a_i = \left. \frac{\partial g(\varphi(t))}{\partial y(t-i)} \right|_{\varphi(t)=\varphi(\tau)} \quad (9)$$

$$b_i = \left. \frac{\partial g(\varphi(t))}{\partial u(t-d-j)} \right|_{\varphi(t)=\varphi(\tau)} \quad (10)$$

$i = 1, \dots, n$ .

$j = 0, \dots, m$ .

y

$$\begin{aligned} \tilde{y}(t-i) &= y(t-i) - y(\tau-i) \\ \tilde{u}(t-i) &= u(t-i) - u(\tau-i) \end{aligned} \quad (11)$$

Separando la parte de la expresión que contiene el vector  $\varphi(t)$ , el modelo lineal aproximado puede ser expresado de manera alternativa por la siguiente expresión:

$$y(t) = [1 - A(z^{-1})]y(t) + z^{-d}B(z^{-1})u(t) + \zeta(\tau) \quad (12)$$

donde el término de desviación o polarización,

$\zeta(\tau)$ , viene dado por:

$$\begin{aligned} \zeta(\tau) &= y(\tau) + a_1 y(\tau-1) + \dots + a_n y(\tau-1) - b_0 u(\tau-d) - \dots \\ &\quad - b_m u(\tau-d-m) \end{aligned} \quad (13)$$

y los coeficientes  $\{a_i\}$  y  $\{b_i\}$  han sido agrupados en  $A$  y  $B$ , ambos polinomios del operador retardo  $z^{-1}$ :

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_n z^{-n} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \dots + b_m z^{-m} \end{aligned} \quad (14)$$

En consecuencia el modelo aproximado dado por (12) puede ser interpretado como un modelo lineal afectado por ruido o perturbaciones,  $\zeta(\tau)$ , dependientes del punto de operación.

Dada la estructura de la red neuronal multicapa utilizada para modelar el sistema de neutralización, el  $pH$  estimado,  $pH_{est}$ , en el instante  $t$  está definido por la siguiente ecuación:

$$pH_{est}(t) = \sum_{j=1}^8 W_j \tanh \left( \sum_{k=1}^2 w_{jk} \varphi_k(t) + w_{j0} \right) + W_0 \quad (15)$$

donde:

$$\varphi_1(t) = pH(t-1)$$

$$\varphi_2(t) = V_a(t-1)$$

Las derivadas de la salida con respecto a las entradas  $\varphi_1(t)$  y  $\varphi_2(t)$  son calculadas según las siguientes ecuaciones:

$$\frac{\partial pH_{est}(t)}{\partial pH(t-1)} = \sum_j W_j w_{j1} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right) \quad (16)$$

$$\frac{\partial pH_{est}(t)}{\partial V_a(t-1)} = \sum_j W_j w_{j2} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right) \quad (17)$$

de manera que los coeficientes del modelo linealizado pueden calcularse a partir de:

$$a_1 = \sum_j W_j w_{j1} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right) \quad (18)$$

$$b_1 = \sum_j W_j w_{j2} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right)$$

##### 4.2 Control Neuronal Predictivo por Linealización Instantánea (CNPLI)

El Control Predictivo Generalizado basado en un modelo auto-regresivo se puede aplicar entonces al sistema no lineal calculando en cada iteración los polinomios  $A(z^{-1})$  y  $B(z^{-1})$  por linealización



instantánea del modelo neuronal. Las perturbaciones no medibles corresponden al término de desviación o polarización,  $\zeta(\tau)$ , de la ecuación (12). El algoritmo de control se implanta según el esquema que se muestra en la figura 3, y contiene los siguientes pasos:

1. Leer  $pH(t)$ .
2. Actualizar el vector de observaciones de la red directa:

$$\varphi(t) = \begin{bmatrix} pH(t-1) \\ V_a(t-1) \end{bmatrix}$$

3. Calcular los coeficientes del modelo lineal partir de las siguientes ecuaciones

$$a_1 = \sum_j W_j w_{j1} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right)$$

$$b_1 = \sum_j W_j w_{j2} \left( 1 - \tanh^2 \left( \sum_k w_{jk} \varphi_k(t) + w_{j0} \right) \right)$$

4. Construir los polinomios:  $A(z^{-1})$  y  $B(z^{-1})$

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_n z^{-n}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \dots + b_m z^{-m}$$

5. Determinar las predicciones óptimas

( $\hat{y} = \Gamma \Delta u + \Phi$ ), donde:

$\hat{y} \equiv [\hat{y}(t+1) \ \hat{y}(t+2) \ \dots \ \hat{y}(t+N_2)]^T$  son las predicciones hasta el horizonte de predicción  $N_2$ .

$$\Delta u = [\Delta u(t) \ \Delta u(t+1) \ \dots \ \Delta u(t+N_u-1)]^T$$

son las señales de control hasta el horizonte de control,  $N_u$ , y

$$\Phi = [\varphi(t+d) \ \varphi(t+d+1) \ \dots \ \varphi(t+N_2)]^T$$

Para ello:

- 5.1 Determinar los polinomios  $E_j(z^{-1})$ ,  $F_j(z^{-1})$  y  $G_j(z^{-1})$ . Haciendo  $E_1 = 1$  y calculando los demás coeficientes de los polinomios a partir de las siguientes ecuaciones:

$$F_j(z^{-1}) = \frac{1 - (1 - z^{-1})A(z^{-1})E_j(z^{-1})}{z^{-j}}$$

$$E_j = E_{j-1}(z^{-1}) + F_{j-1,0} * z^{-(j-1)}$$

$$G_j(z^{-1}) = B(z^{-1}) * E_j(z^{-1})$$

- 5.2 Determinar la matriz  $\Gamma$  y el vector  $\Phi$

$$\Gamma = \begin{bmatrix} g_0 & \cdot & \cdot & \cdot & 0 \\ g_1 & g_0 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ g_{N_2} & g_{N_2-1} & \cdot & \cdot & g_{N_2-N_u+1} \end{bmatrix}_{(N_2+1) \times N_u}$$

$$\Phi = \begin{bmatrix} z^1(G_1(z^{-1}) - g_0 - g_1 z^{-1})\Delta u(t) + F_1 y(t) \\ z^2(G_2(z^{-1}) - g_0 - g_1 z^{-1} - g_2 z^{-2})\Delta u(t) + F_2 y(t) \\ \dots \\ z^{N_2}(G_{N_2}(z^{-1}) - g_0 - g_1 z^{-1} - \dots - g_{N_2} z^{-N_2})\Delta u(t) + F_{N_2} y(t) \end{bmatrix}$$

6. Obtener la variación del vector de control óptimo,  $\Delta u(t)$ , por minimización del funcional de costo  $J$ :

$$J(N_2, N_u) = \sum_{j=1}^{N_2} \eta [s(t+j) - \hat{y}(t+j)]^2 + \sum_{i=1}^{N_u} \lambda [\Delta u(t+i-1)]^2$$

que, descrito en forma vectorial, y sustituyendo las predicciones calculadas en el paso (5), queda:

$$J = 0.5 * \Delta u^T * 2 * (\eta \Gamma^T * \Gamma + \lambda * I) * \Delta u + (\Phi - \underline{s})^T * \Gamma * \Delta u$$

donde:

$N_2$  = Horizonte de predicción.

$N_u$  = Horizonte de Control.

$\eta$  y  $\lambda$ : parámetros de sintonización que ponderan el comportamiento futuro de la planta y el controlador.

$s(t)$ : Trayectoria de referencia y

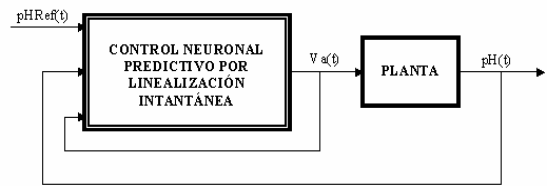
$$\underline{s} = [pH_{Ref}(t+1) \ pH_{Ref}(t+2) \ \dots \ pH_{Ref}(t+N_2)]^T$$

La optimización del funcional  $J$  se puede hacer utilizando la función *quadprog* disponible en Matlab®, Versión 7.1 Release 12, para obtener:

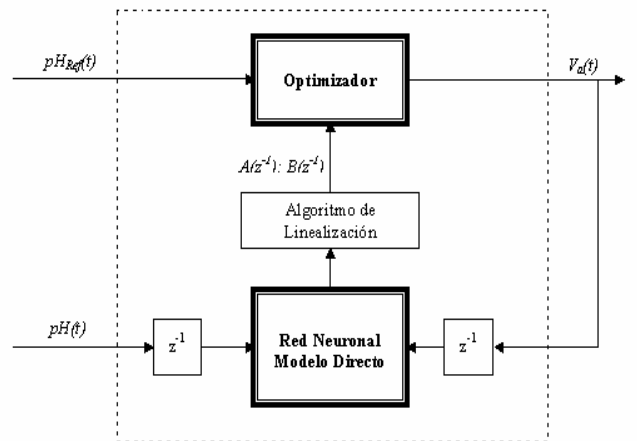
$$\Delta u(t) = \text{quadprog}(\eta \Gamma^T * \Gamma + \lambda * I, (\Phi - \underline{s})^T * \Gamma, \underline{a}, \underline{b})$$

7. Calcular la señal de control óptima, utilizando la primera componente del vector,  $\Delta u(t) = \Delta V_a(t)$ .

$$V_a(t) = V_a(t-1) + \Delta V_a(t)$$



(a) Esquema de Control



(b) Detalle: Control Neuronal por Linealización Instantánea

Figura 3. Control Neuronal Predictivo por Linealización Instantánea

Una de las ventajas de esta estrategia, sobre la estrategia de control predictivo neuronal reportado generalmente en la literatura que usa directamente



en el proceso de optimización una red neuronal como modelo del sistema, es la posibilidad de implantarla en tiempo real, ya que siendo el proceso de optimización mucho más sencillo, el tiempo de cálculo de la señal de control está muy por debajo de los límites que impone el tiempo de muestreo del sistema (Liu, Kadiramanathan y Billings, 1998; Zamarreño y Vega, 1999; Yu y Yu, 2003; Parlos, Parthasarathy y Atiya, 2001). Algunos resultados se presentan en la siguiente sección.

## 5. RESULTADOS

La operación del sistema de control requiere el ajuste de cuatro parámetros: Horizonte de Predicción ( $N_2$ ), horizonte de control ( $N_u$ ) y los coeficientes que ponderan el comportamiento futuro de la planta ( $\eta$ ) y la señal de control ( $\lambda$ ). Diferentes pruebas en los dos puntos de operación descritos previamente, producen los resultados resumidos en las tablas 3 y 4, en cuanto a error estacionario  $e_{ss}$  y tiempo de establecimiento de la respuesta  $T_{ss}$ .

En la tabla 5 se muestran los resultados de diferentes experiencias con los valores que se estimaron más convenientes para los parámetros ajustables.

En las figuras 4 y 5 se presentan las respuestas de la planta a lazo cerrado, con  $\eta=0.5$  y  $\lambda=0.05$  para la primera prueba y  $\eta=0.5$  y  $\lambda=0.01$  para la segunda prueba, y los valores óptimos de  $N_2, N_u=2$

Tabla 3. Características de la respuesta al implantar el CPNLI con  $\eta=0.5$ ;  $\lambda=0.01$

Prueba ( $N_2=N_u$ )	$pH$	$pH_{Ref}$	$e_{ss}$ [%]	$T_{ss}$ [min]	
1	2	7.32	7.32	0.0%	9
	4	7.32	7.32	0.0%	8
	6	7.32	7.32	0.0%	8
2	2	7.33	7.32	0.2%	7
	4	7.32	7.32	0.0%	13
	5	7.20	7.32	2.0%	12

Tabla 4. Características de la respuesta al implantar el CPNLI con  $N_2=N_u=2$

Prueba	$\eta$	$\lambda$	$pH$	$pH_{Ref}$	$e_{ss}$ [%]	$T_{ss}$ [min]
1	0.5		7.32	7.32	0.0%	9
	1	0.05	7.32	7.32	0.0%	9
	5		7.32	7.32	0.0%	11
2	0.5		7.33	7.32	0.2%	7
	1	0.01	7.31	7.32	0.1%	3
	5		7.31	7.32	0.1%	8

Tabla 5. Características de la respuesta al implantar el CPNLI con  $N_2=N_u=2$ ;  $\eta=0.5$

Prueba	$\lambda$	$pH$	$pH_{Ref}$	$e_{ss}$ [%]	$T_{ss}$ [min]
1	0.01	7.32	7.32	0.0%	9
	0.05	7.32	7.32	0.0%	8
	0.1	7.32	7.32	0.0%	9
2	0.01	7.33	7.32	0.2%	7
	0.05	7.05	7.32	0.2%	15
	0.1	7.19	7.32	0.2%	11

También se realizaron pruebas para demostrar la efectividad del CNPLI en una situación diferente al punto de operación del sistema al sintonizar el controlador.

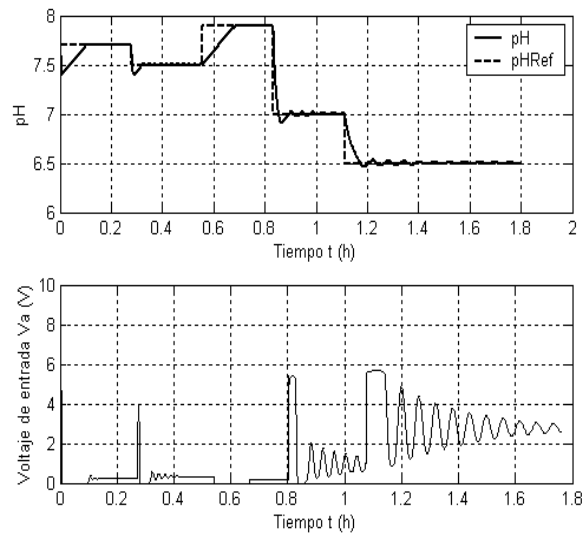


Figura 4. Desempeño del CNPLI. (Prueba 1).

Las figuras 6 y 7 muestran que las dos versiones del controlador responden de manera satisfactoria (aunque con cierto deterioro en la calidad de la respuesta transitoria) ante condiciones diferentes a las utilizadas en su sintonización.

Finalmente, se explora el comportamiento del controlador ante perturbaciones en el flujo de base de entrada. Los resultados de dos experiencias se muestran en las figuras 8 y 9, donde se produjo un incremento de 36.8 ml/min (39%) en el flujo de base, en las dos condiciones de operación y prácticamente no se observaron alteraciones en la respuesta del sistema.



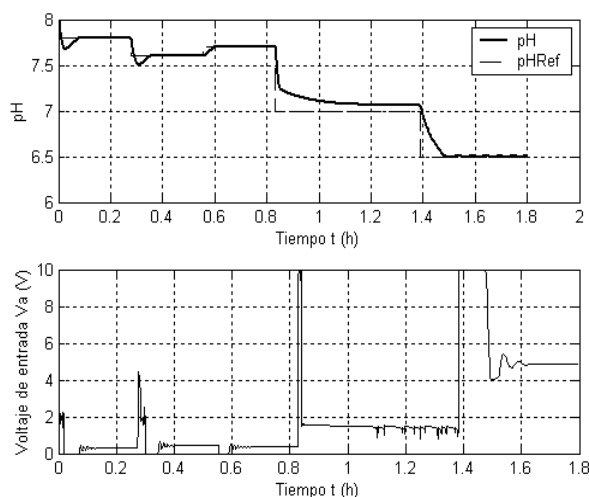


Figura 5. Desempeño del CNPLI. (Prueba 2).

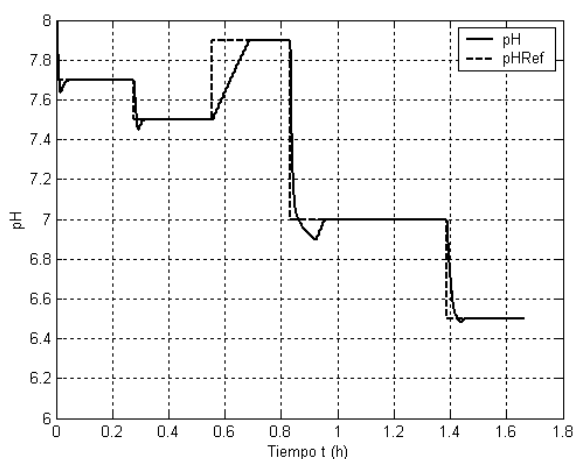


Figura 6. Desempeño del CNPLI ajustado para la prueba 2. (Prueba 1).

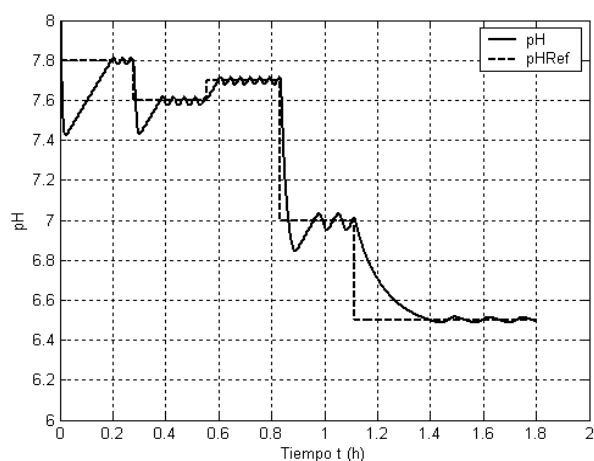


Figura 7. Desempeño del CNPLI ajustado para la prueba 1. (Prueba 2).

Figura 8. Rechazo a perturbaciones. Controlador CNPLI (Prueba 1)

## 6. CONCLUSIONES

Los controladores neuronales predictivos que comúnmente se reportan en la literatura, utilizan a la red neuronal para calcular el comportamiento futuro del sistema, sobre un horizonte de tiempo especificado. Las salidas estimadas pasan a un optimizador, el cual tiene como objetivo la minimización de un criterio de comportamiento para el cálculo de una adecuada señal de control. Se trata de un problema de optimización bastante complejo, que en muchos casos conlleva a que el esquema no se pueda implantar en tiempo real.

Figura 9. Rechazo a perturbaciones. Controlador CNPLI (Prueba 2)

Para subsanar esta dificultad, en este trabajo, se propone esta nueva estrategia de control predictivo neuronal por linealización instantánea CPNLI, que combina la ventaja de las redes neuronales, las cuales pueden constituir modelos identificados muy precisos de sistemas no lineales, y las numerosas ventajas de la estrategia de control predictivo generalizado que ha sido ampliamente estudiada y probada en la industria (Camacho y Bordons, 1995). Los coeficientes del modelo autoregresivo en el CPNLI, se extraen por la



linealización instantánea de un modelo neuronal, y son recalculados en cada iteración del algoritmo. Se trata entonces de un esquema de control neuronal adaptativo, pero con una estrategia de muy fácil implantación. En efecto, en la fase realizada fuera de línea sólo se requiere ajustar la red neuronal directa y los parámetros característicos del GPC (horizonte de control, horizonte de predicción, etc.). Adicionalmente, su “costo” computacional es bajo. El tiempo de cálculo oscila alrededor de 0.3 seg. siendo entonces 67% menor que el tiempo de muestreo de la planta de neutralización estudiada (1 seg.).

Ante cambios en la referencia del controlador, la calidad de la respuesta transitoria se ve afectada a medida que las diferencias entre la planta y el modelo lineal aumentan. Esto ocurrió en las experiencias de laboratorio debido a alteraciones provocadas en la concentración de la solución de base de entrada. Situaciones de este tipo podrían mejorarse reajustando la red neuronal directa, con información obtenida en línea, cada cierto tiempo. Los resultados de respuesta estable son bastante precisos, con desviaciones estacionarias solo en presencia de errores del modelo. El comportamiento del CPNLI es excelente en el rechazo a perturbaciones (cambio en el flujo de base de entrada).

#### AGRADECIMIENTO

Los autores agradecen el financiamiento otorgado por FONACIT para este trabajo, bajo la subvención S1 99000111.

#### REFERENCIAS

- Ali, E. (2001). “pH Control using PI Control algorithms with automatic tuning method”. *Trans IChemE*, **Vol. 79**, Parte A, pp. 611-619.
- Clarke, D. W., Mohtadi y Tuffs, P. S. (1987). “Generalized Predictive Control – parts I and II”. *Automatica*, **23** (2), pp 137-160.
- Camacho E.F. and Bordons C. (1995). “*Model Predictive Control in the process industry*”. Springer, London.
- Galindo, J., Margaglio, E., Lamanna, R. y Uria, M. (2000). “Control difuso multi-modelo: Aplicación al control de pH en un reactor de laboratorio”. *Memorias de ESTYLF 2000* (X Congreso Español sobre Tecnologías y Lógica Fuzzy), pp. 49-54. Sevilla, España.
- Garcia, C.E., D.M. Prett and M. Morari (1989). “Model Predictive Control: Theory and Practice-A Survey,” *Automatica*, **25**, pp 335-348.
- He, X. y Asada, H. (1993). “A new method for identifying orders of input-output models for nonlinear dynamic systems”. *Proc. of the American Control Conference*. San Francisco, California.
- Hunt K.J., Sbarbaro D., Zbikoswski R. and Gawthrop P.J. (1992). “Neural Networks for Control Systems. A Survey”. *Automatica*, **28** (6), pp. 1083-1112.
- Lamanna R., Uria M., Jaramillo L. y Ricardo F. (2000). “Control neuronal predictivo de un reactor de laboratorio”. *Memorias del IX Congreso Latinoamericano de Control Automático*. Cali, Colombia. (En CDrom).
- Lin Jun-Shien, Jang Shi-Shang; Shieh Shyan-Shu y Subramanian, M. (1999). “Generalized Multivariable Dynamic Artificial Neural Network. Modeling for Chemical Processes”. *Ind. Eng. Chem. Res.*, **Vol. 38**, N° 12, pp. 4700-4711.
- Liu, G. P., Kadirkamanathan, V. y Billings, S. A. (1998): “Predictive control for non-linear systems using neural networks”. *Int. J. Control*. **Vol. 71**, N° 6. pp. 1119-1132.
- Narendra, K. y Parthasarathy, K. (1990). “Identification and control of dynamical systems using neural networks”. *IEEE Transactions on Neural Networks*, **Vol. 1**, N° 1, pp.4-27.
- Nørgaard, P. M. (1998). “*System identification and control with neural networks*”. Ph.D. Thesis. Departamento de Automatización, Universidad Técnica de Denmark.
- Parlos A. G., Parthasarathy S., Atiya A. F. (2001). “Neuro – Predictive Process Control Using On-Line Controller Adaptation”. *IEEE Transactions on Control Systems Technology*. **Vol. 9**, N° 5.
- Sørensen, O. (1994). “Neural Networks performing system identification for control applications” *Third Int. Conference on Artificial Neural Networks*. IEE Brighton, UK, pp.172-176.
- Waller, J. y Toivonen, H. (2002). “A neuro-fuzzy model predictive controller applied to a pH Neutralization process”. *Proc. of the 15th International Federation of Automatic Control IFAC World Congress*. España.
- Yu, D. L. y Yu, D. W. (2003). “A linear parameter-varying radial basis function model and predictive control of chemical reactor”. *International Journal of Systems Science*. **Vol. 34**, N° 14-15. pp. 747-761.
- Zamarreño, J. M. y Vega P. (1999). “Neural predictive control. Application to a highly non-linear system”. *Engineering Applications of Artificial Intelligence*. **12**, pp. 149-158.