

# ESCUELA POLITÉCNICA NACIONAL

## FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

### DISEÑO Y SIMULACIÓN DE DOS ESQUEMAS DE CONTROL INTELIGENTE CON REDES NEURONALES, APLICADOS A UN PROCESO DE NEUTRALIZACIÓN DE PH ALTAMENTE NO LINEAL

TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO  
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
“ELECTRÓNICA Y CONTROL”

**DIEGO ALEXIS ORTIZ CUNDAR**

[Diego.ortiz@epn.edu.ec](mailto:Diego.ortiz@epn.edu.ec)

**DIEGO PATRICIO VALDIVIEZO ANDINO**

[Diego.valdiviezo@epn.edu.ec](mailto:Diego.valdiviezo@epn.edu.ec)

**DIRECTOR: ING. GEOVANNY DANILO CHÁVEZ GARCÍA, PhD.**

[Danilo.chavez@epn.edu.ec](mailto:Danilo.chavez@epn.edu.ec)

DMQ, Febrero 2022

## **CERTIFICACIONES**

Nosotros, DIEGO ALEXIS ORTIZ CUNDAR y DIEGO PATRICIO VALDIVIEZO ANDINO declaramos que el trabajo de integración curricular aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.



---

**DIEGO ALEXIS ORTIZ CUNDAR**



---

**DIEGO PATRICIO VALDIVIEZO ANDINO**

Certifico que el presente trabajo de integración curricular fue desarrollado por DIEGO ALEXIS ORTIZ CUNDAR y DIEGO PATRICIO VALDIVIEZO ANDINO, bajo mi supervisión.

---

**GEOVANNY DANILO CHAVEZ**  
**DIRECTOR**

## **DECLARACIÓN DE AUTORÍA**

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como los productos resultantes del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

DIEGO ALEXIS ORTIZ CUNDAR

DIEGO PATRICIO VALDIVIEZO ANDINO

GEOVANNY DANILO CHÁVEZ GARCÍA

## DEDICATORIA

*A mis padres, Hugo y Zulay por su confianza, apoyo y esfuerzo dedicado en toda esta etapa de mi formación profesional.*

**Diego Valdiviezo.**

## AGRADECIMIENTO

*A “Dios”, que me permitió dar este paso muy importante dentro de mi vida.*

*A mis padres, Hugo y Zulay por ser los artífices de este sueño, y por su incansable muestra de amor y apoyo en todas las etapas de mi vida.*

*A mis hermanos, quienes siempre estuvieron presentes en cada momento y no perdieron su fe en alcanzar este logro.*

*A mi familia, mi esposa Michelle, y mi hija Annie, por ser la parte más importante de mi vida, por ser el pilar principal de apoyo y motivación, para seguir adelante en el alcance de cada uno de mis objetivos.*

*A mis familiares en general, en especial a mi tío, Patricio, quien me acogió en su hogar, durante todos esos años de mi etapa universitaria y supo guiarme hacia el alcance de esta meta.*

*A mis compañeros de la Escuela Politécnica Nacional, con cuales compartí una de las etapas más difíciles, pero más maravillosas de mi vida, gracias por compartir su conocimiento, cada una de sus experiencias de vida, y sobre todo por ser buenos amigos.*

*Finalmente quiero extender un agradecimiento infinito a todos quienes fueron mis profesores a lo largo de mi carrera, quienes son los formadores de este logro, en especial a los doctores Danilo Chávez y Oscar Camacho, quienes nos brindaron de la manera más desinteresada su apoyo en esta última etapa de la carrera.*

# ÍNDICE DE CONTENIDO

## TOMO II

	CERTIFICACIONES.....	I
	DECLARACIÓN DE AUTORÍA.....	II
	DEDICATORIA.....	III
	AGRADECIMIENTO.....	IV
	ÍNDICE DE CONTENIDO.....	V
	RESUMEN .....	VII
	ABSTRACT .....	VIII
	METODOLOGÍA .....	1
2.	2.1. DESCRIPCIÓN Y ANÁLISIS DE NO LINEALIDAD DEL REACTOR DE NEUTRALIZACIÓN.....	2
	2.1.1. Descripción del proceso .....	2
	2.1.2. Análisis de no linealidad.....	3
	2.2. APROXIMACION A FOPDT PROMEDIO DE LA PLANTA .....	10
	2.2.1. Cálculo del modelo de FOPDT promedio .....	11
	2.3. SINTONIZACIÓN DE CONTROLADOR PID CLÁSICO.....	12
	2.4. DISEÑO DEL ESQUEMA DE CONTROL CON PID NEURONAL.....	14
	2.5. DISEÑO DEL ESQUEMA DE CONTROL PID +RED NEURONAL EN PARALELO.....	19
4.	2.5.1. Selección de la red neuronal y base de datos.....	20
5.	2.5.2. Entrenamiento de la red neuronal.....	22
	REFERENCIAS BIBLIOGRÁFICAS .....	25
	ANEXOS.....	26
	ANEXO I .....	27
	ANEXO II .....	32
	ANEXO III .....	34
	ANEXO IV .....	35
	ANEXO V .....	37

## **TOMO II**

El presente, corresponde a la segunda parte, del trabajo de integración curricular “DISEÑO Y SIMULACIÓN DE DOS ESQUEMAS DE CONTROL INTELIGENTE CON REDES NEURONALES, APLICADOS A UN PROCESO DE NEUTRALIZACIÓN DE PH ALTAMENTE NO LINEAL” en el cual se muestran los capítulos 2, 4 y 5 correspondientes, a Metodología, Bibliografía y Anexos respectivamente, donde se detalla el procedimiento, fuentes de información utilizadas e información extra correspondientes al desarrollo al proyecto mencionado.

## RESUMEN

En el presente tomo, se muestra el detalle de la metodología empleada para el diseño, y simulación de los esquemas de control, PID (Control Proporcional, Integral y Derivativo) , clásico, PID Neuronal y PID+ Red Neuronal en paralelo, mismos que son empleados para el control del nivel pH, en el reactor de neutralización de pH, que se menciona en el capítulo 1 del tomo I del presente trabajo, se muestra una descripción y análisis de no linealidad de la planta (reactor de neutralización) , junto con las técnicas de sintonización de PID empleadas para este tipo de plantas, además del procedimiento utilizado para el entrenamiento, validación y pruebas de las respectivas redes neuronales empleadas en los 2 esquemas de control neuronal. Además, se muestra las referencias bibliográficas, y anexos, que corresponde a información extra y respaldos bibliográficos, correspondiente al desarrollo de este trabajo.

**PALABRAS CLAVE:** Reactor de Neutralización, PID Neuronal, Red Neuronal, Retroalimentación, Algoritmo de retro propagación, FOPDT, NARX.



## **ABSTRACT**

In this volume, the detail of the methodology used for the design and simulation of the control schemes, PID (Proportional, Integral and Derivative Control), classic, PID Neural and PID+ Neural Network in parallel, which are used for the control of the pH level, in the pH neutralization reactor, which is mentioned in chapter 1 of volume I of this work, a description and non-linearity analysis of the plant (neutralization reactor) is shown, along with the PID tuning techniques used for this type of plants, in addition to the procedure used for training, validation and testing of the respective neural networks used in the 2 neural control schemes. In addition, the bibliographical references and annexes are shown, which correspond to extra information and bibliographical support, corresponding to the development of this work.

**KEYWORDS:** Neutralization reactor, Neural PID, Neural Network, Feedforward, Back Propagation Algorithm, FOPDT, NARX.

## METODOLOGÍA

2. En el presente capítulo se muestra el desarrollo del producto final del proyecto de titulación, se indica mediante un esquema ordenado cada uno de los pasos seguidos, hasta obtener los resultados finales de los esquemas de control diseñados. El desarrollo del proyecto se fundamenta en una investigación descriptiva, ya que se diseña y evalúa distintos esquemas de control basados en redes neuronales para un proceso de neutralización de pH altamente no lineal, y finalmente obtener resultados del desempeño de dichos controladores mediante experimentación y comparación. La obtención de información requerida para el desarrollo del proyecto se obtuvo de artículos científicos, libros y páginas web académicos.

El presente trabajo en su totalidad, se conforma de cinco capítulos: el primero que hace referencia a un acercamiento al lector sobre el trabajo a realizar y una revisión bibliográfica de los conceptos necesarios para el desarrollo del mismo. El segundo capítulo, el cual se detalla en la presente sección, detalla la etapa de diseño, donde se hace una descripción del proceso y el análisis del mismo, desde el punto de vista de no linealidad, para luego realizar el diseño de un controlador clásico PID, el cual pasa a ser la base para el posterior diseño de los esquemas de control neuronal, tanto como para el PID neuronal, como para el esquema PID + Red neuronal en paralelo. El tercer capítulo corresponde al análisis de resultados obtenidos en las simulaciones de cada uno de los esquemas diseñados para el proceso y de la comparación de los mismos, además de las conclusiones y recomendaciones que se pueden dar de este análisis. En el cuarto capítulo se muestran las referencias bibliográficas utilizadas como fuente de información teórica, para el desarrollo del proyecto. Finalmente, el quinto y último capítulo que contiene información adicional que ya sea por extensión o relevancia fueron agregadas a esta sección denominada ANEXOS.

## 2.1. DESCRIPCIÓN Y ANÁLISIS DE NO LINEALIDAD DEL REACTOR DE NEUTRALIZACIÓN.

### 2.1.1. Descripción del proceso

Como se menciona en el capítulo 1 del tomo I, de este trabajo, el proceso sobre el cuál se diseñaran los controladores neuronales, es el de neutralización de pH, presentada en [1] el cual se caracteriza por su alta no linealidad con retardo de tiempo dominante y variable, el proceso consiste en la neutralización de un fluido a través de un tanque reactor, el cual es agitado de manera uniforme por lo que se considera una mezcla perfectamente homogénea de los flujos que ingresan al mismo, el flujo de corriente ácida ingresa por el ducto  $q_1(t)$  al tanque 2 del proceso para luego pasar a través de una válvula manual al reactor de neutralización, el flujo constante  $q_2(t)$  compuesto por bicarbonato de sodio permite mantener al nivel de Ph insensible a pequeños cambios de los flujos ácidos o básicos que ingresan al reactor, mientras que el flujo básico ingresa por el ducto  $q_3(t)$ , el mismo que constituye nuestra variable física a manipular para el control de nivel de pH del proceso, para finalmente tener nuestra mezcla saliendo del reactor por el ducto  $q_4(t)$  controlado por una válvula manual, donde a 6.745 metros se tiene un transmisor de pH con un rango de medición de entre 2.172 a 10.75 adimensional [2].

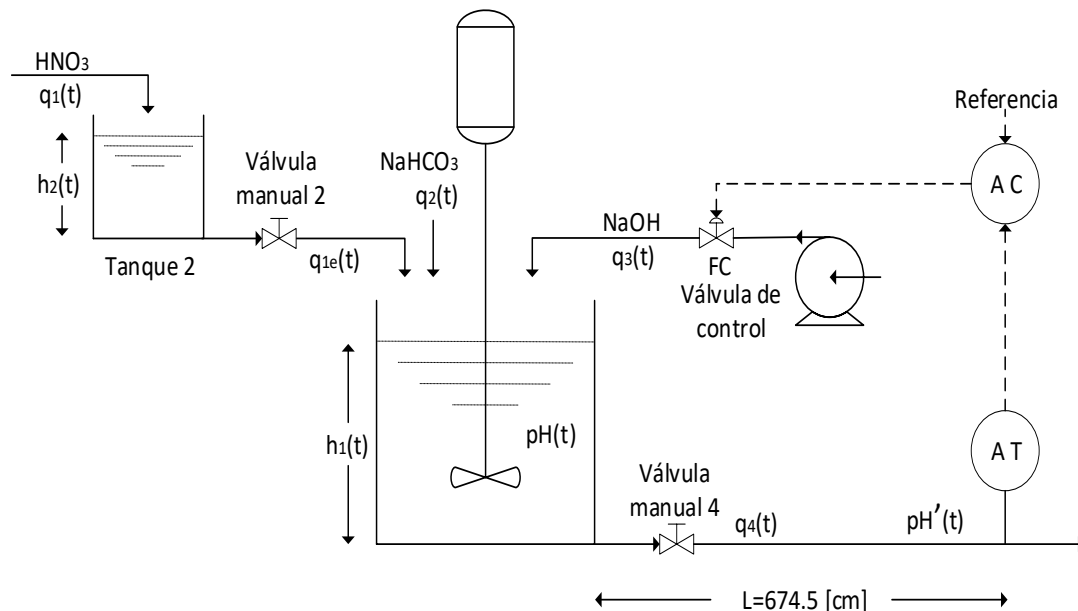


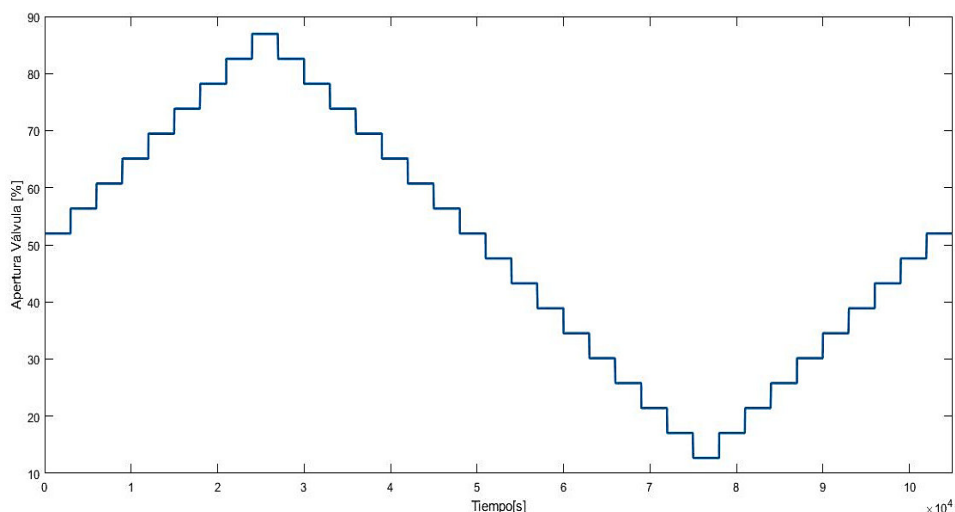
Figura 2.1 Esquema del proceso de neutralización de pH. Tomado de [1].

El modelo matemático y obtención de las ecuaciones diferenciales que describen nuestro proceso se muestran en el ANEXO I, donde además se muestra el punto de operación de nuestra planta y la tabla de valores de cada uno de las variables y parámetros involucrados en el análisis matemático del proceso alrededor de nuestro proceso.

### 2.1.2. Análisis de no linealidad.

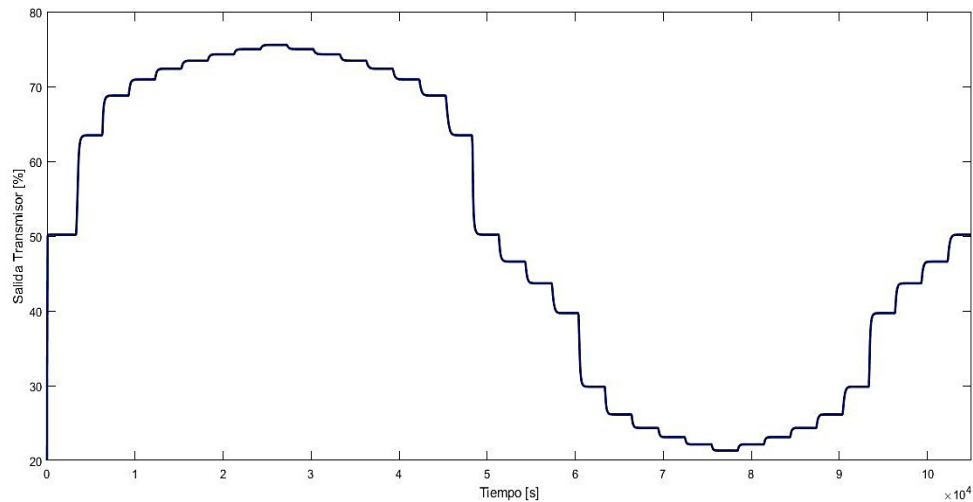
Partiendo del modelado matemático de nuestra planta, se puede definir a nuestro sistema como un sistema no lineal debido a la característica de las ecuaciones diferenciales que describen el proceso, las cuales son de tipo no lineal, considerando esta parte y el procedimiento sugerido por Iglesias [3] el cual consiste en analizar los cambios de parámetros de una serie de aproximaciones FOPDT (Planta de primer orden con retardo de tiempo) por sus siglas en inglés, estas aproximaciones y estimación de parámetros se las hace en base a la curva de reacción de la salida de nuestra planta en lazo abierto, frente a pequeñas variaciones en la señal de entrada, que corresponde al porcentaje de apertura de la válvula de flujo básico.

En la figura 2.2 se muestran los cambios realizados en la señal de entrada partiendo del punto de operación de la misma, el cual, tal como se indica en el ANEXO I, se encuentra en 52%, a partir de ahí se realizan cambios sucesivos del 4.37% tanto en subida como bajada cada 3000 [s], en un rango del 12.6 al 87% de apertura de la válvula.



**Figura 2.2** Secuencia cambios en porcentaje en la entrada de la planta.

Una vez definidos los cambios a realizar en la entrada de la planta en la figura 2.3 se muestra la respuesta de la planta, frente a los respectivos cambios.

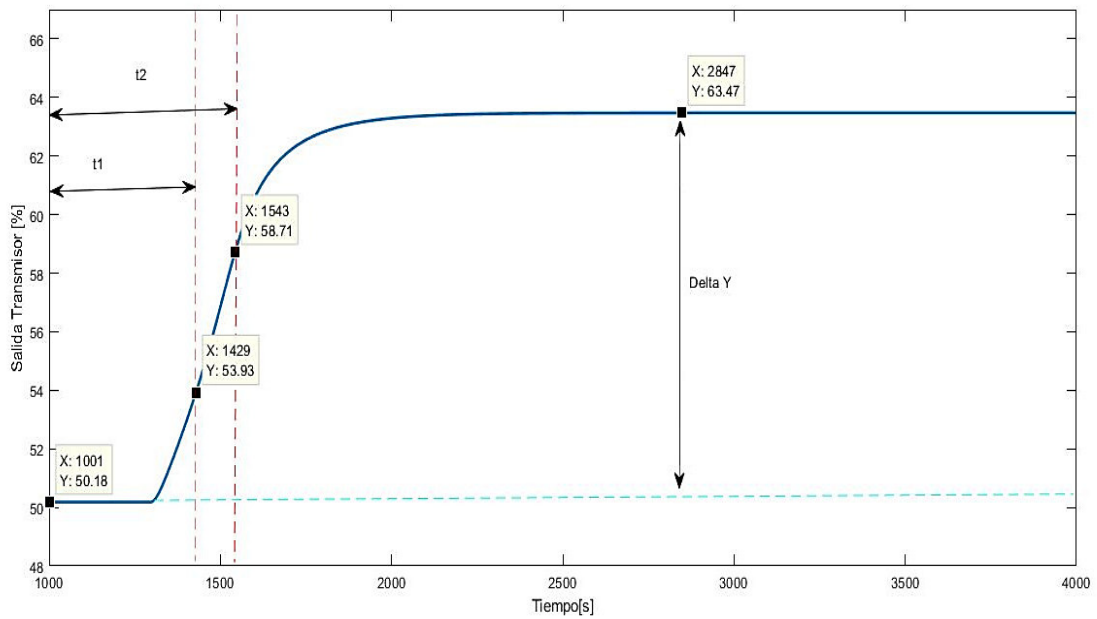


**Figura 2.3** Salida de la planta ante cambios de referencia mostrados en la figura 2.2

En la gráfica anteriormente mostrada son evidentes los cambios de respuesta que presenta la planta ante distintos puntos de operación, para corroborar esto, se realiza la aproximación FOPDT en cada uno de los puntos de respuesta de la planta, mediante el método de curva de reacción descrito previamente en el marco teórico de este trabajo.

Se usan como ejemplo de cálculo, 2 puntos específicos que son uno de subida que va del 52 al 56.368% y otro de bajada que va de la misma forma de 60.736% al 56.368% de apertura de la válvula, el cual se debe repetir para cada una de las respuestas obtenidas, ante las variaciones introducidas al sistema, para finalmente mostrar en una tabla de resumen los parámetros FOPDT que se obtuvieron en cada uno de los puntos de operación.

A continuación, en la figura 2.4 se muestra la obtención de datos de la curva de reacción para el cambio en subida, de donde a partir de estos datos y usando las ecuaciones 1.2, 1.3 y 1.4 se calcula la FOPDT en este punto.



**Figura 2.4** Curva de reacción de la planta ante un cambio de 52 a 56.37% en la entrada.

Reemplazando los datos obtenidos en la figura anterior se tiene que:

Ganancia K:

$$K^+ = \frac{63.47 - 50.18}{56.37 - 52} = 3.03 \quad (2.1)$$

Constante de tiempo  $\tau$  :

$$\tau^+ = 1.5(543 - 429) = 171 \quad (2.2)$$

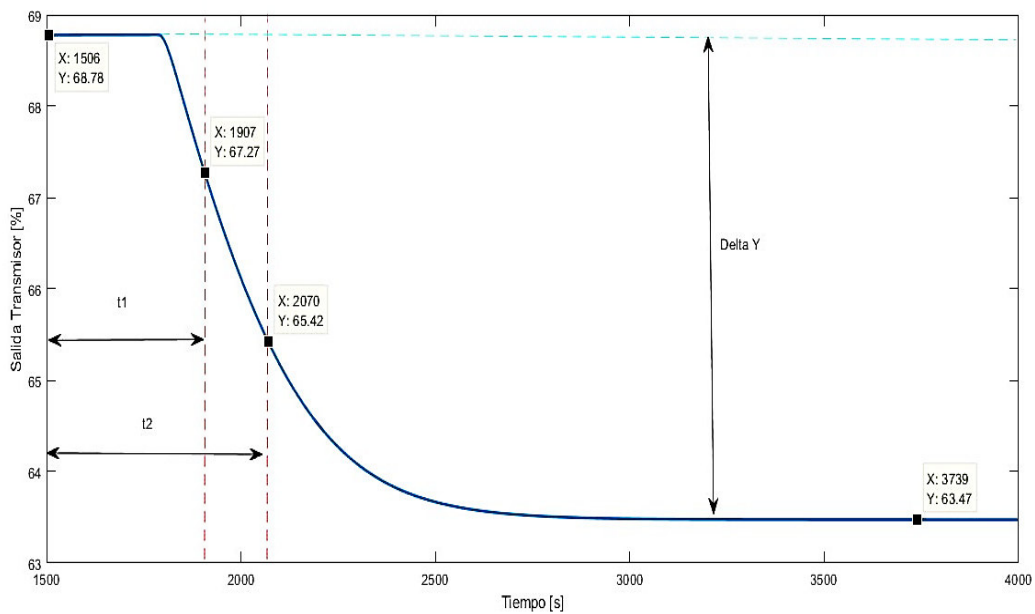
Retardo de tiempo  $t_0$  :

$$t_0^+ = 543 - 171 = 372 \quad (2.3)$$

Entonces, el modelo de FOPDT ante un cambio positivo se indica en la Ecuación

$$G(s)^+ = \frac{3.03}{171s + 1} e^{-372s} \quad (2.4)$$

De la misma forma que se realiza el cálculo de la FOPDT ante un cambio positivo, se ejemplifica el cálculo para un cambio negativo, partiendo de los datos proporcionados por la figura 2.5.



**Figura 2.5** Curva de reacción de la planta ante un cambio de 60.74 a 56.37% en la entrada.

Reemplazando los datos obtenidos en la figura anterior se tiene que:

a) Ganancia K:

$$K^- = \frac{63.47 - 68.78}{56.37 - 60.74} = 1.22 \quad (2.5)$$

b) Constante de tiempo  $\tau$  :

$$\tau^- = 1.5(570 - 407) = 244.5 \quad (2.6)$$

c) Retardo de tiempo  $t_0$  :

$$t_0^- = 570 - 244.5 = 325.5 \quad (2.7)$$

Entonces, el modelo de FOPDT ante un cambio positivo se indica en la Ecuación

$$G(s)^+ = \frac{1.22}{244.5s + 1} e^{-325.5s} \quad (2.8)$$

Como se puede observar en las ecuaciones 2.4 y 2.8 los parámetros de las plantas FOPDT cambian sus parámetros, tal como se lo esperaba, debido a la característica de no linealidad de la planta.

Tal como se muestra en los ejemplos de cálculo anteriores, se procede de la misma manera en los demás puntos de operación de la planta, pero en este caso se automatiza el proceso de cálculo empleando un script (archivo de código .m) desarrollado en el mismo Matlab, los códigos empleados para obtención de parámetros, tanto en cambios de referencia positivos y negativos, se muestran en el ANEXO II.

Para resumir los parámetros obtenidos a partir de los scripts, en cada punto de operación, se realiza la Tabla 2.1.

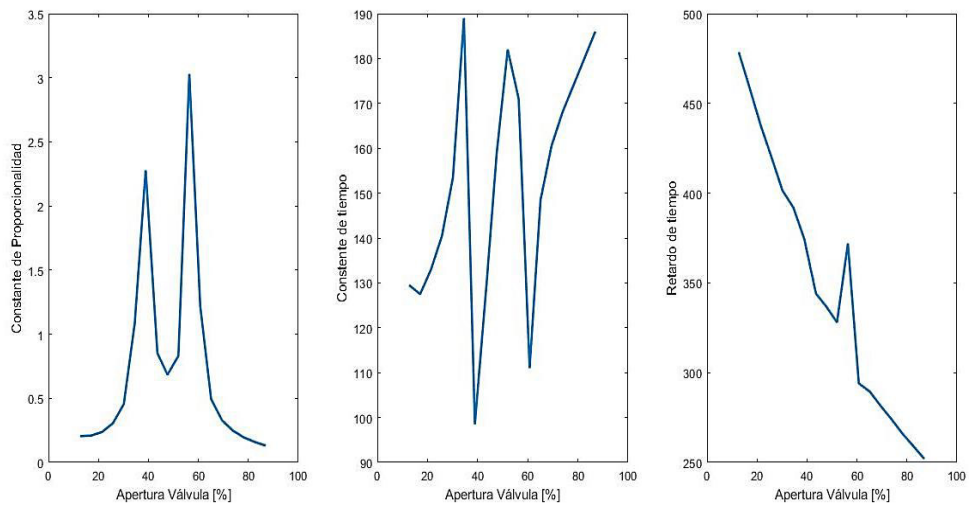


**Tabla 2.1** Valores de parámetros para modelos FOPDT.

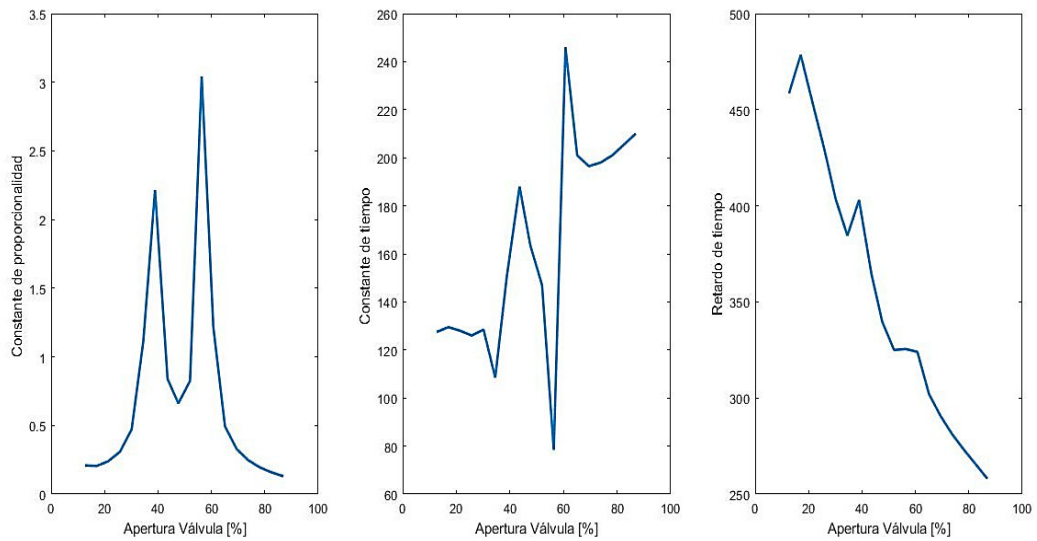
<b>Modelo FOPDT</b>	<b>Cambio en la entrada de la planta <math>m(t)</math> [%]</b>	<b>Ganancia <math>K</math> [adimensional]</b>	<b>Constante de tiempo <math>\tau</math> [s]</b>	<b>Retardo de tiempo <math>t_0</math> [s]</b>	<b>Relación <math>\frac{t_0}{\tau}</math> [adimensional]</b>
1	52 – 56.37	3.03	171	372	2.175
2	56.37 – 60.74	1.22	111	294	2.648
3	60.74 – 65.10	0.494	148.5	289.5	1.949
4	65.1 – 69.47	0.328	160.5	281.5	1.754
5	69.47 – 73.84	0.247	168	274	1.631
6	73.84 – 78.21	0.195	174	266	1.529
7	78.21 – 82.58	0.159	180	259	1.439
8	82.58 – 86.94	0.131	186	252	1.355
9	86.94 – 82.58	0.130	210	258	1.229
10	82.58 – 78.21	0.158	205.5	265.5	1.292
11	78.21 – 73.84	0.194	201	273	1.358
12	73.84 – 69.47	0.246	198	281	1.419
13	69.47 – 65.10	0.328	196.5	290.5	1.478
14	65.10 – 60.74	0.493	201	302	1.502
15	60.74 – 56.37	1.215	246	324	1.317
16	56.37 – 52	3.043	78.5	325.5	4.093
17	52 – 47.63	0.824	147	325	2.211
18	47.63 – 43.26	0.661	163.5	339.5	2.076
19	43.26 – 38.90	0.839	188	364	1.943
20	38.90 – 34.53	2.214	151	403	2.668
21	34.53 – 30.16	1.115	108.5	384.5	3.544
22	30.16 – 25.79	0.472	128.5	403.5	3.140
23	25.79 – 21.42	0.309	126	430	3.412
24	21.42 – 17.06	0.240	128	454	3.549
25	17.06 – 12.69	0.205	129.5	478.5	3.695
26	12.69 – 17.06	0.209	127.5	458.5	3.596
27	17.06 – 21.42	0.237	133	438	3.293
28	21.42 – 25.79	0.305	140.5	420	2.989

29	25.79 – 30.16	0.455	153.5	401.5	2.615
30	30.16 – 34.53	1.085	189	392	2.074
31	34.53 – 38.9	2.278	98.5	374.5	3.802
32	38.9 – 43.26	0.853	130	344	2.646
33	43.6 – 47.63	0.684	159	337	2.119
34	47.63 – 52	0.831	182	328	1.802

De acuerdo a los valores que se muestran en la tabla 2.1 , se puede observar como todos los parámetros de las FOPDT van variando en función del cambio realizado en la entrada de la planta, para evidenciar de manera más ilustrativa como varían tanto la constante de tiempo, el retardo y la constante de proporcionalidad, se muestran las figuras 2.6 y 2.7, donde en se muestran la variación en cambios de subida y bajada respectivamente de cada uno de los 3 parámetros de la FOPDT (Constante de proporcionalidad, constante de tiempo y tiempo de retardo).



**Figura 2.6** Variación de parámetros de las FOPDT de la planta ante cambios de ascendentes.



**Figura 2.7** Variación de parámetros de las FOPDT del plante ante cambios de descendentes.

En las figuras 2.6 y 2.7 se puede observar como los parámetros van cambiando en función del porcentaje de apertura de la válvula de control, donde se muestra una relación no definida entre éstas, a excepción del parámetro de retardo de tiempo el cuál se aproxima a una relación lineal decreciente con la apertura de la válvula.

## 2.2. APROXIMACION A FOPDT PROMEDIO DE LA PLANTA

Como primera parte del proceso de desarrollo de los controladores, se tiene la aproximación de a una función de transferencia de primer orden con retardo de tiempo (FOPDT), de la planta, esto como información fundamental, para el diseño de un PID clásico el cual servirá como base para el desarrollo de los esquemas de control neuronal.

Para el cálculo de la FOPDT de la planta, utilizamos un promedio general de todos los valores de los parámetros obtenidos en la tabla 2.1, esto con el fin de obtener una función de transferencia única, sobre la cual se utilizarán métodos clásicos de sintonización del PID para el proceso.

### 2.2.1. Cálculo del modelo de FOPDT promedio

A través de las fórmulas básicas de la media aritmética, tenemos:

a) Determinación de la ganancia promedio.

$$K_{prom} = \frac{\sum_{i=1}^{34} K_i}{26} \quad (2.9)$$

$$K_{prom} = \frac{9.05}{34} = 0.2657$$

b) Determinación de la constante de tiempo promedio.

$$\tau_{prom} = \frac{\sum_{i=1}^{34} \tau_i}{26} \quad (2.10)$$

$$\tau_{prom} = \frac{5206.74}{34} = 153.13$$

c) Determinación del retardo de tiempo promedio.

$$t_{0prom} = \frac{\sum_{i=1}^{34} t_{0i}}{34} \quad (2.11)$$

$$t_{0prom} = \frac{12001.32}{34} = 352.98$$

Entonces, el modelo de FOPDT promedio se muestra en la Ecuación 2.12.

$$G(s) = \frac{0.2657}{153.13s + 1} e^{-352.98s} \quad (2.12)$$

### 2.3. SINTONIZACIÓN DE CONTROLADOR PID CLÁSICO.

En esta sección, se procesa a realizar el diseño del controlador tipo PID clásico para nuestra planta de neutralización de pH utilizando la información obtenida en la sección 2.2, que como se mencionó anteriormente será el punto de partida para el posterior diseño de los controladores neuronales, es por ello de la importancia de mostrar la metodología empleada para el diseño de este primer esquema de control empleado para la planta en estudio.

En primera instancia para el diseño del PID se procede a determinar el método de sintonización más adecuado y conveniente para el proceso, el cual se lo realiza en base a las características que muestra el modelo lineal FOPDT de la planta. Teniendo en cuenta que el modelo obtenido, representado en la ecuación 2.12, se escoge el método de sintonización propuesto por Dahlin mostrado en [4], el cual funciona muy bien en comparación a otros métodos de sintonización en modelos con retardo de tiempo dominante, es decir en aquellos cuya relación entre el retardo y constante de tiempo del modelo es mayor a uno, ( $t_0/\tau > 1$ ). Para el diseño del PID por medio de Dahlin se deben calcular los parámetros correspondientes a este, tales como la constante de proporcionalidad  $K_p$ , el tiempo integral  $\tau_i$  y el tiempo de derivativo  $\tau_d$ , para esto se usan las ecuaciones propuestas por el autor del método [4], mostradas en la tabla 1.1.

Empleando las fórmulas propuestas en la tabla 1.1 se tiene:

- a) Cálculo de la constante de control proporcional.

$$K_p = \frac{\tau_{prom}}{2 * K_{prom} * t_{0Prom}} \quad (2.13)$$

$$K_p = \frac{153.13}{2 * 0.2657 * 352.98} = \mathbf{0.8163}$$

- b) Cálculo de la constante de tiempo integral.

$$\tau_i = \tau \quad (2.14)$$

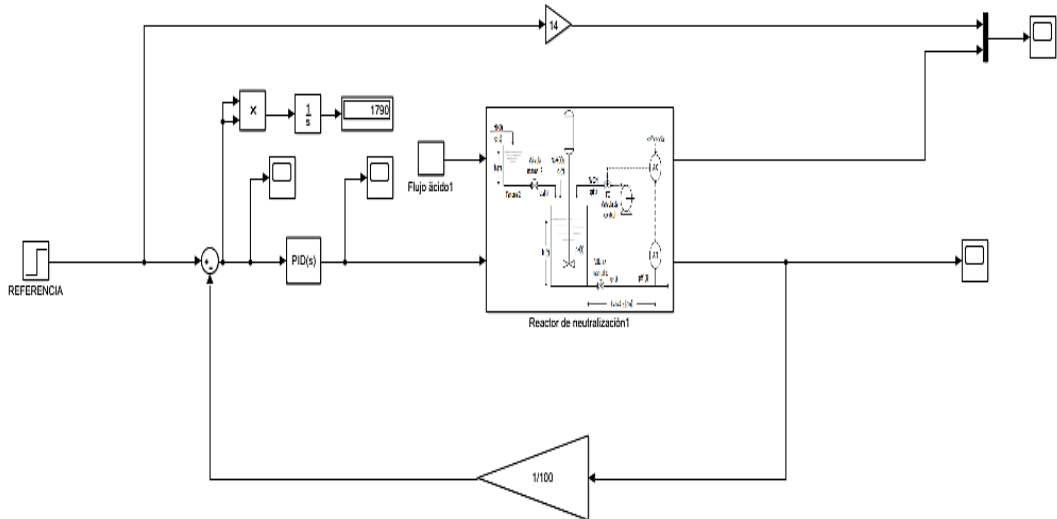
$$\tau_i = \mathbf{153.13}$$

c) Cálculo de la constante de tiempo derivativo

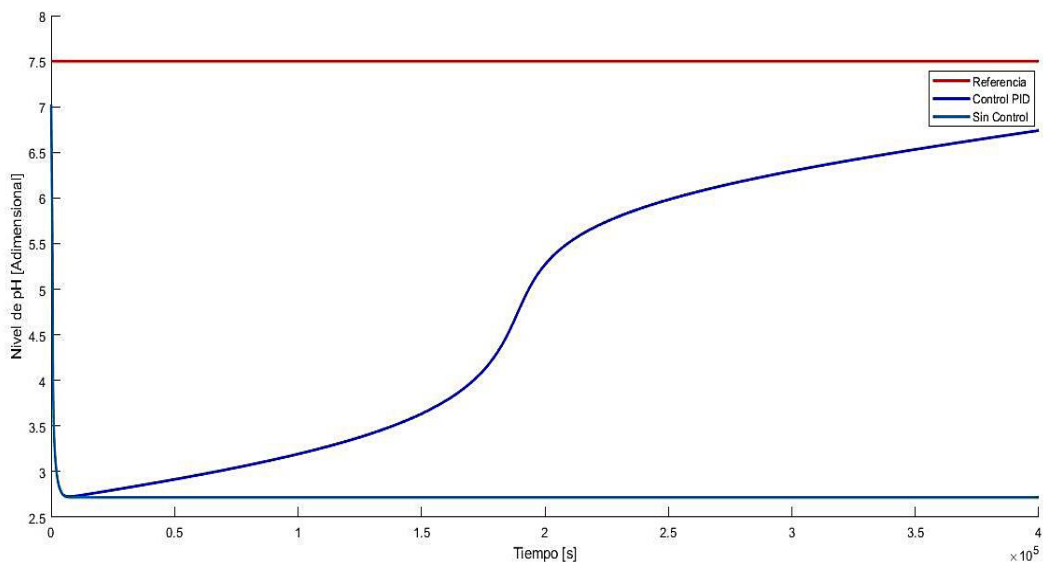
$$\tau_d = \frac{t_0}{2} \quad (2.15)$$

$$\tau_d = \frac{352.98}{2} = 176.49$$

El diagrama de bloques implementado en Simulink y la respuesta obtenida del lazo de control con el PID diseñado, se muestra en la figuras 2.8 y 2.9 respectivamente, donde mediante un bloque PID se cargan los parámetros de las constantes de proporcionalidad, tiempo integral y tiempo derivativo, obtenidos en el diseño, además dentro del bloque PID se carga un parámetro de filtrado para la señal de control, esto con el objetivo de mitigar las oscilaciones introducidas por la acción de control derivativa, haciendo que la acción de control no sea muy brusca, el parámetro con el que se trabaja para este filtro es de 0.75. La simulación de funcionamiento del PID se la realiza para una referencia de un nivel de pH igual a 7.5, sin perturbaciones, y durante un tiempo de 40000 segundos.



**Figura 2.8** Diagrama de bloques implementado en Simulink con PID diseñado.



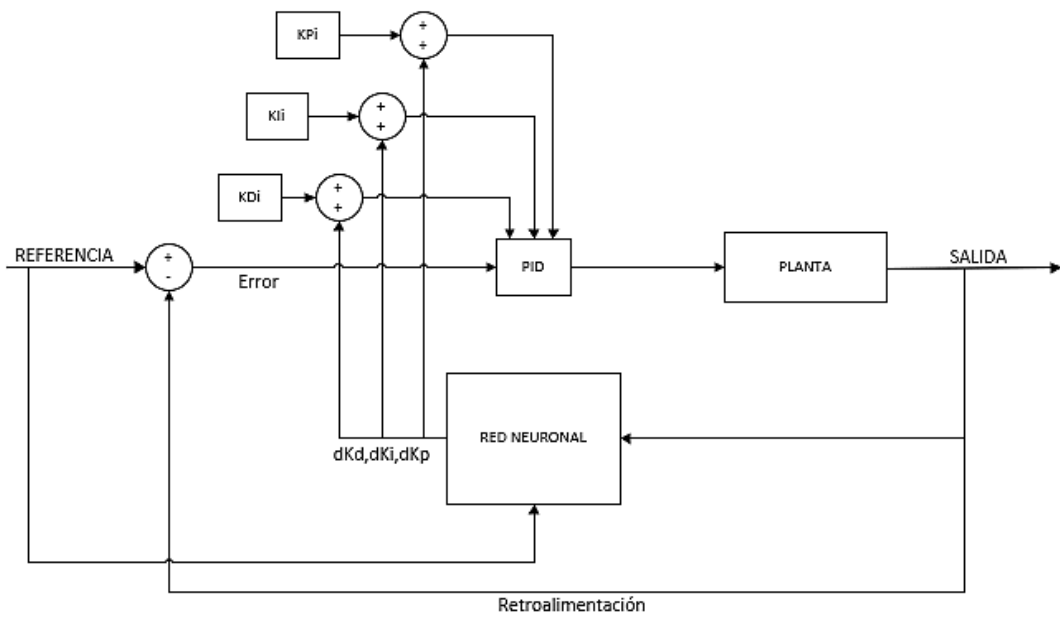
**Figura 2.9** Respuesta de pH con el controlador PID diseñado.

Como se puede observar en la figura 2.9, el PID diseñado no logra satisfacer las necesidades de control, pues si bien se ve que tiende a estabilizarse en el punto de referencia el tiempo que tomaría, es ahí la necesidad de implementar esquemas de control más eficientes como los esquemas con redes neuronales.

## 2.4. DISEÑO DEL ESQUEMA DE CONTROL CON PID NEURONAL.

El esquema de control neuronal, tipo PID Neuronal, denominado así por su autor Omatu Shigeru [5], se basa en el entrenamiento de una red neuronal con retroalimentación del error, programada con un algoritmo de retro propagación del error, en donde se tiene como entrada el valor de la referencia y la salida de nuestra planta, y como salida se tiene pequeños valores de corrección de los parámetros del PID, los cuales irán modificando de manera automática los valores iniciales de los parámetros del PID clásico, en función del error que ingrese a la red neuronal, a excepción del parámetro de filtrado del PID, que se exceptúa en este esquema de control.[6]

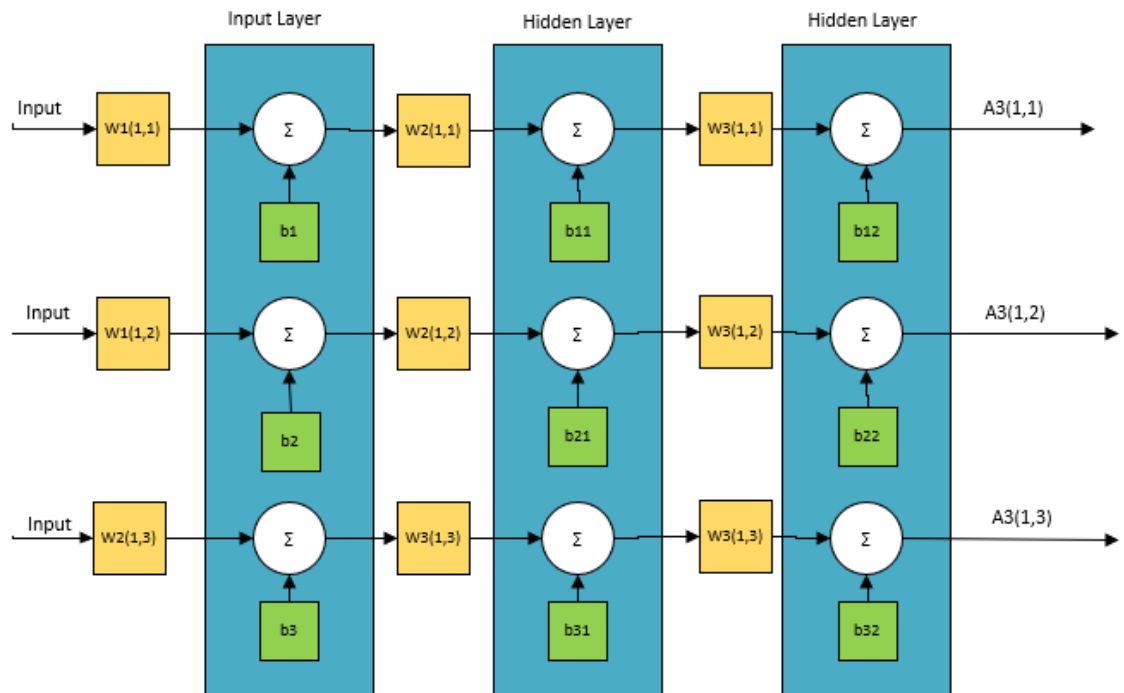
El esquema general diseñado se muestra en la figura 2.10, donde se puede observar lo mencionado en el párrafo anterior.



**Figura 2.10** Esquema general de control con PID Neuronal.

El algoritmo que se diseña para este esquema está basado en el entrenamiento en línea de una red neuronal feedforward con algoritmo de retro propagación del error, cuyas funciones de activación son de tipo lineal, con un total de tres capas, donde se tiene la capa de entrada y dos capas ocultas, una de las cuales hace como salida de la red, mientras que la otra es en la que se ejecutan las funciones de activación y a la cual se le van asignado los pesos correspondientes para la minimización del error de salida de la red, la capa de salida pasa a ser la misma capa oculta ya que se toman como salidas los pesos actualizados de esta como resultados del entrenamiento de la red. La estructura de la red neuronal utilizada para el PID neuronal se muestra en la figura 2.11.





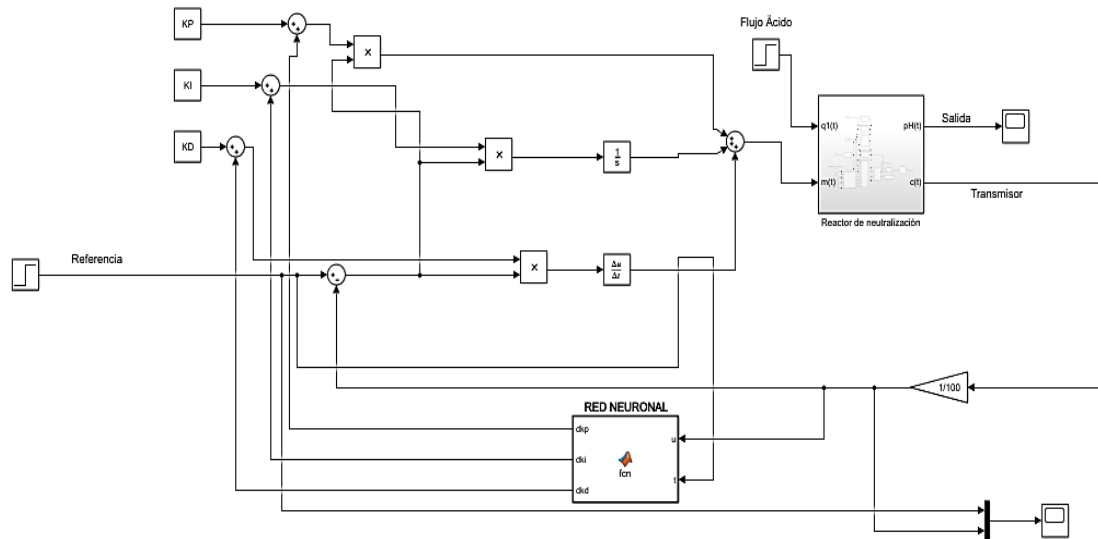
**Figura 2.11** Esquema general de red neuronal a entrenar en PID neuronal.

Como se menciona en el capítulo I, las redes feedforward (alimentación hacia adelante) por sus siglas en inglés, son las más recomendadas cuando se desea realizar interpolación de datos no lineal, con un error mínimo, es por ello que, al tener respuestas no lineales en el reactor de neutralización de pH, se procede a usar este tipo de red neuronal para una auto sintonización inteligente del PID de la planta.

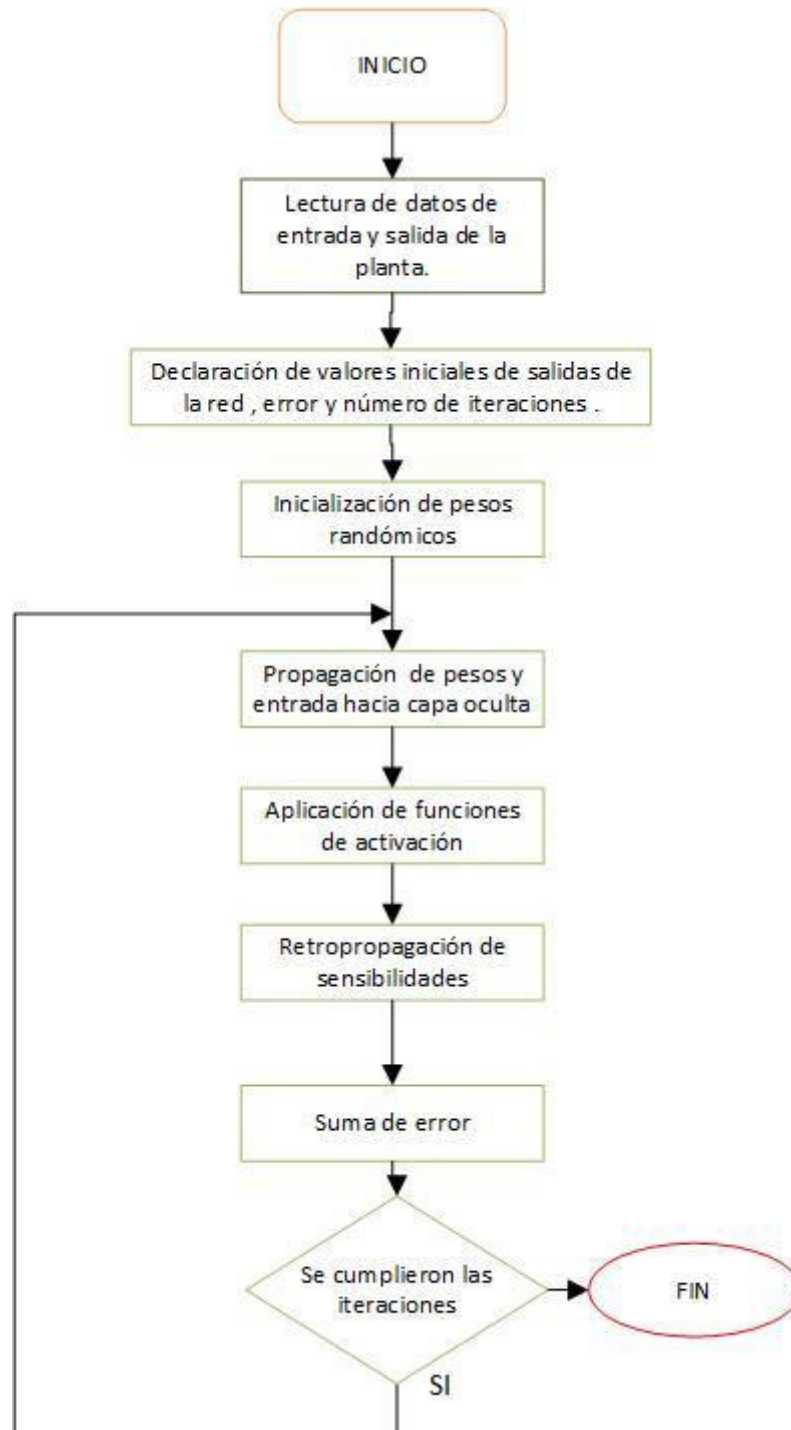
La red neuronal es implementada mediante código en una función de Matlab dentro de Simulink, la programación de la red neuronal se la realiza básicamente con la implementación de las ecuaciones 1.8, 1.9 con función de activación lineal y 1.10, en un lazo de iteraciones que representa el número de épocas o repeticiones para la actualización de pesos de la red y minimización del error, debido a que lo que se espera es el mínimo error posible en la red neuronal se toman como salida los pesos finales obtenidos, los cuales pasan a ser los factores de corrección que se agregan a los correspondientes valores iniciales del PID diseñado en la sección 2.3.

El diagrama de bloques implementado en Simulink se muestran en la figura 2.12, donde se puede identificar el bloque de función de Matlab donde se realizó la programación correspondiente al entrenamiento de la red feedforward para el ajuste de parámetros del PID diseñado en 2.3.

En la figura 2.13 se muestra el diagrama de flujo empleado en el bloque de funciones para el entrenamiento de la red neuronal, mientras que el código es descrito en el ANEXO III.



**Figura 2.12** Diagrama de bloques implementado para control con PID Neuronal.



**Figura 2.13** Diagrama de flujo para entrenamiento de red neuronal para PID Neuronal.

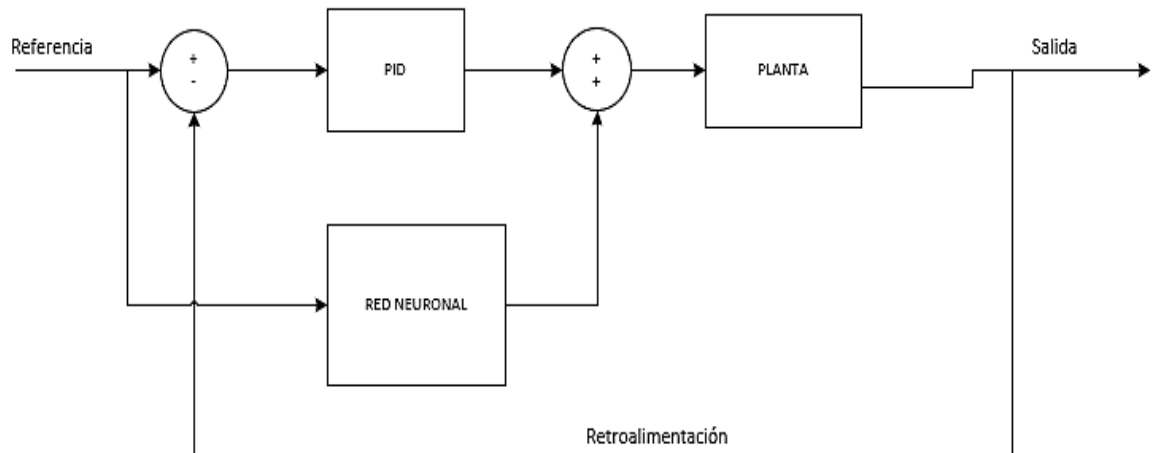
Una vez definido el esquema de control, se procede a comprobar los resultados del mismo, los cuales son obtenidos mediante simulación en distintas situaciones como pequeños cambios de referencia e introducción de perturbaciones, y para el análisis del mismo se hace una comparación de índices de desempeño como ISE y TVu, mencionados en la sección 1.4 del presente trabajo, con respecto a los mostrados por el esquema de control con PID clásico bajo las mismas condiciones de simulación. Dicho análisis y resultados se muestran en el capítulo 3.

## **2.5. DISEÑO DEL ESQUEMA DE CONTROL PID +RED NEURONAL EN PARALELO.**

En este esquema, se plantea un sistema de control basado en el PID clásico sintonizado previamente en la sección 2.3, el cual se encuentra trabajando en paralelo con una red neuronal entrenada previamente, por lo que podría considerarse como una red neuronal offline, es decir solamente procesa los datos de entrada en función de los parámetros del entrenamiento previo que se hizo, la cual es una de principales diferencias del esquema anterior de PID neuronal, el cual por su parte se caracteriza por realizar un entrenamiento en línea de la base de datos que ingresan a la red neuronal.

La principal característica de este esquema de control es que la red neuronal actúa como ente de colaboración a la actividad de control realizada por el PID clásico [7], ante situaciones donde la señal de control que este envía hacia el actuador del proceso. (Válvula), no es la suficiente para el cumplir los requerimientos de control deseados, actuando así la red neuronal como un comando parcial de control del sistema

El esquema general de control planteado se muestra en la figura 2.14, donde se puede evidenciar como las entradas a la red neuronal son los valores de referencia de la planta, y cuya salida se suma a la señal de control del PID clásico.



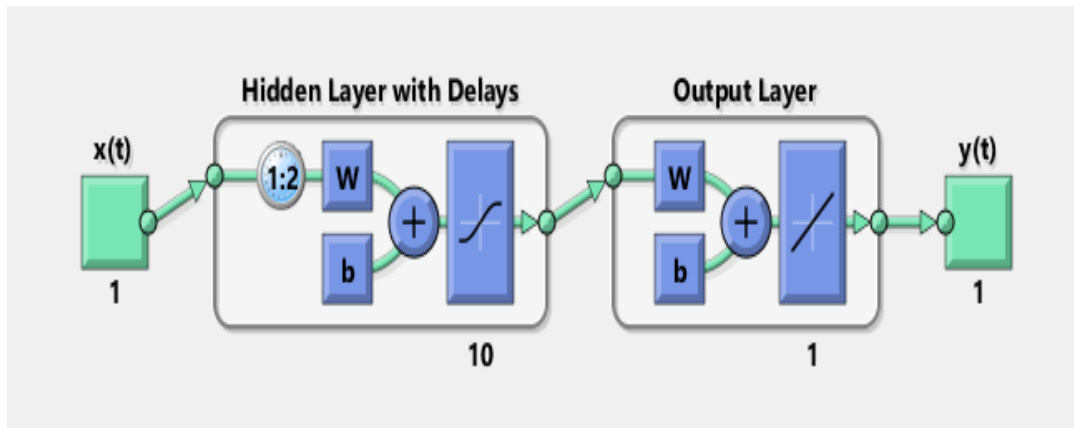
**Figura 2.14** Esquema general de control con PID+ Red Neuronal

Como se mencionó anteriormente este esquema neuronal se caracteriza por poseer una red neuronal previamente entrenada, por lo que, para el diseño e implementación de este esquema, se debe primero realizar la selección del tipo de red neuronal a entrenar y la base de datos con la que se va a entrenar la misma, para luego mediante el toolbox (caja de herramientas) de Matlab realizar la configuración más óptima de la red, para el sistema de control, a continuación, se muestra el procedimiento a seguir para el entrenamiento de la red.

### 2.5.1. Selección de la red neuronal y base de datos.

Para la selección de la red neuronal y la base de datos a utilizar, lo primero que se debe considerar es cuál va a ser el objetivo de la misma, el cual es obtener una señal de control de compensación a la del PID clásico, que permita a la válvula de regulación de flujo básico realizar la acción de apertura o cierre porcentual adecuado, para obtener los niveles de pH requerido por la referencia. Teniendo en cuenta lo anterior, se considera una red que permita interpolar una función no lineal de la manera más exacta precisa por lo que se considera una red neuronal recurrente, con algoritmo de retro propagación, considerando que este tipo de redes neuronales permiten predecir el valor futuro de una función basada a experiencias anteriores[8] , y que si tenemos esto en el esquema de comando en la planta, se podrá tener un mejor resultado en la acción de control, por lo que se considera una tipo de red neuronal especial denominado NARX (Nonlinear, Autorregresive with External Input) por sus siglas en inglés, la cual se caracteriza por sus aplicaciones en

ejercicios de predicción e interpolación de funciones no lineales, utilizando funciones de activación tipo tangente sigmoide.[9] .La estructura general de la red neuronal seleccionada se muestra en la figura 2.15. La estructura y funcionamiento de la redes tipo NARX se muestran más a detalle en el ANEXO IV.



**Figura 2.15** Estructura general de red neuronal NARX seleccionada para el esquema de control.

Por otro lado , para la selección de datos de entrada de la red, se selecciona al error entre la entrada y la salida del sistema de control funcionando únicamente con el PID diseñado en 2.3 , simulado con 4 cambios de subida y 4 de bajada en la referencia, mientras que para los datos de salida, se selecciona la señal de control obtenida mediante la simulación de esquemas de control funcionando con sus respectivos PID sintonizados en cada uno de los puntos de referencia seleccionados, por lo que se toma en cuenta los datos de la tabla 2.1 para realizar una primera sintonización de cada PID, para luego ajustarlos hasta obtener la mejor respuesta y desempeño posible de los controladores diseñados, para la sintonización de los PID se utiliza el mismo método que se usó en la sección 2.3, cuyos resultados se muestran en la tabla 2.2, junto con los cambios de referencia donde se mostró mejores resultados, cabe recalcar que estos datos son obtenidos de la simulación a través del bloque To Workspace, considerando que estos deben ser de las mismas dimensiones y obtenidos bajo las mismas condiciones, es decir con cambios de referencia en el mismo orden y al mismo tiempo.

**Tabla 2.2** Parámetros de PID's seleccionados para obtención de la señal de control deseada para datos de salida de la red neuronal.

Cambio de Referencia [Nivel pH]	Cambio de entrada de la planta [% Apertura Válvula]	PID		
		$K_p$ [Adimencional]	$\tau_i$ [s]	$\tau_d$ [s]
7.025 – 7.5	52 – 56.37	0.8	171	186
7.5 – 8.0	56.37 – 60.74	0.155	111	147
8.0 – 8.5	60.74 – 65.10	0.52	148.5	144.75
8.5 – 9.0	65.10 – 69.46	0.869	160.5	140.75
7.025 – 6.5	52 – 47.63	0.274	147	162.5
6.5 – 6.0	47.63 – 43.26	0.364	163.5	169.75
6.0 – 5.5	43.26 – 38.90	0.308	188	182
5.5 – 5.0	38.90 – 34.54	0.85	151	201.5

Una vez que se simulan los esquemas, tanto el PID promedio de la sección 2.3 con los cambios de referencia de la tabla 2.2 y con los respectivos PID de la misma, se toman los datos antes mencionados y se procede a entrenar la red neuronal en el toolbox de Matlab.

### 2.5.2. Entrenamiento de la red neuronal.

Una vez guardada la base de datos a entrenar en la red neuronal se procede a realizar el entrenamiento de la misma, cabe mencionar que la cantidad de datos obtenidos, está en función del tiempo que simularon los esquemas de donde se obtuvieron los mismo, los cuales por defecto se toman 10 datos cada segundo.

El entrenamiento de la red neuronal se lo hace desde el toolbox “Neural Time Series” (ntstool) el cuál es una herramienta de entrenamiento de redes neuronales de series de tiempo de Matlab, para acceder al mismo lo podemos hacer digitando en la ventana de comandos, ‘ntstool’, donde se nos muestra 3 tipos de redes en serie de tiempo a seleccionar como se muestra en la figura 2.16, donde se selecciona la red “Nonlinear Input-Output”, que es el tipo de red seleccionado previamente, para luego realizar una secuencia de pasos guiados por la herramienta, donde primero se selecciona los datos de entrada y

salida de la red y se los configura como vectores de columnas, luego se configura el porcentaje de datos que se utilizarán para el entrenamiento, validación y prueba de la red neuronal, dicha distribución de porcentajes por recomendación es del 70, 15 y 15% respectivamente [9], se continúa con la configuración número de retardos que pueden ser 1 o 2 , esto como análisis de experiencias pasadas de la red para la predicción a futuro, además del número de capas ocultas de nuestra red, las cuales son en función del error medio cuadrático , ya que no existe un criterio definido para esto, sino más bien depende de los datos entrenados, es por ello que se inicia entrenando la red con una capa oculta y se llega hasta 10 capas ocultas, ya que la carga computacional es mayor para un mayor número de capas ocultas, después de la comprobación del error medio cuadrático y la mediación con la carga computacional se determinó que con tres capas ocultas el entrenamiento presenta buenos resultados. Toda esta etapa de entrenamiento de la red se muestra en el ANEXO V.

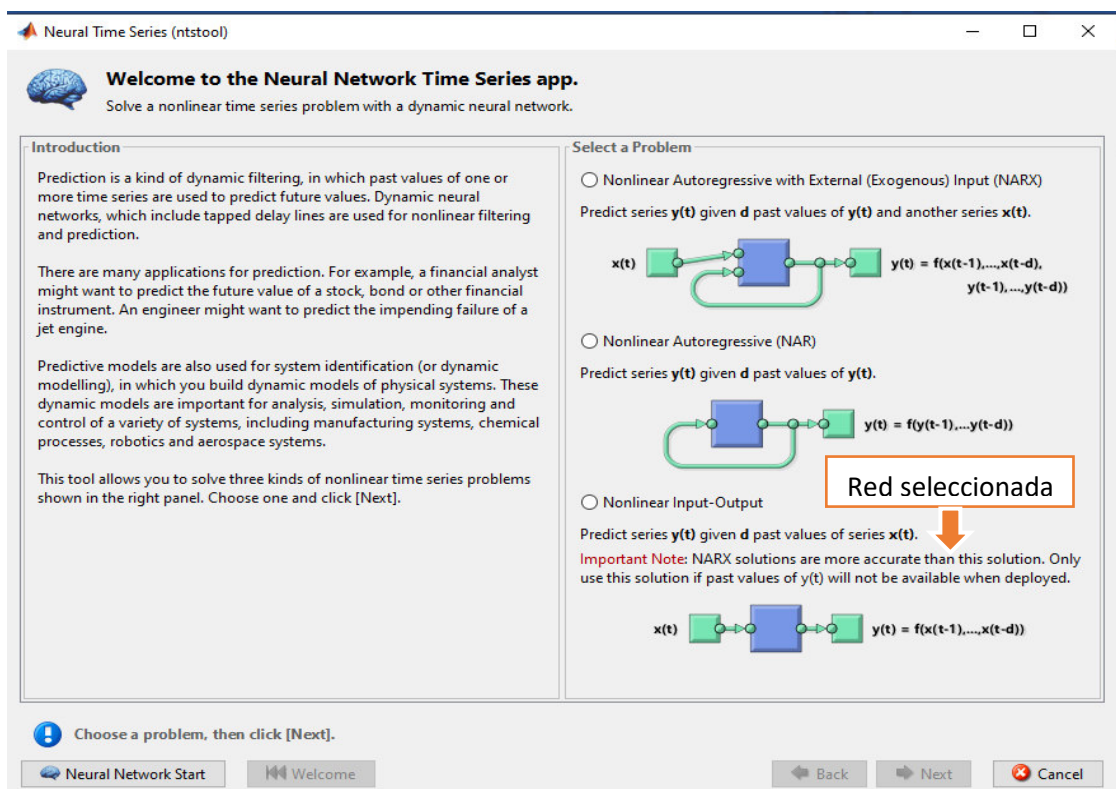
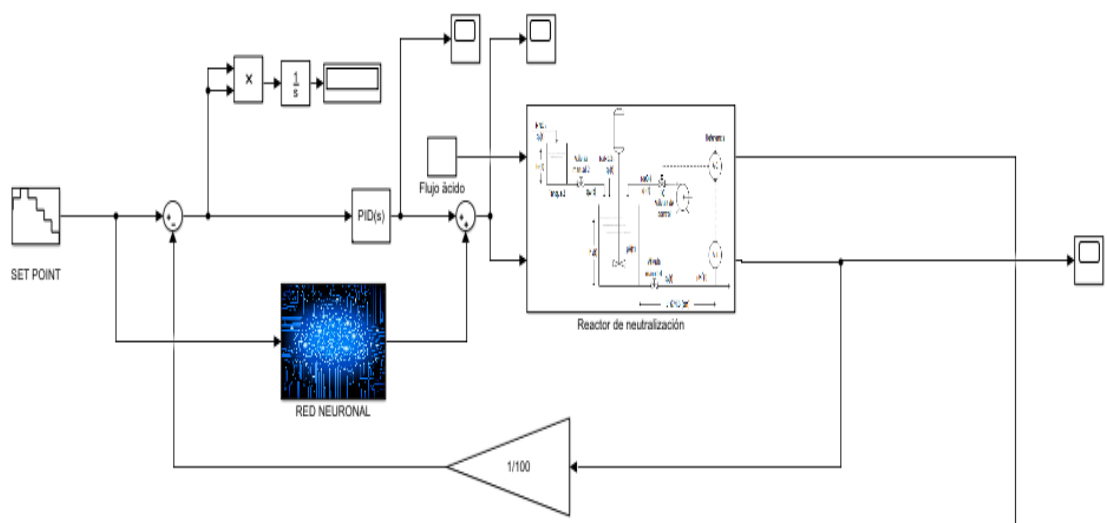


Figura 2.16 Neural Times Series de Matlab



Una vez terminado el proceso de configuración de la red neuronal para su entrenamiento se entrena la misma, para lo cual se usa como algoritmo de entrenamiento el algoritmo de retro propagación del error y como algoritmo de optimización de Levenberg Marquadt el cuál se caracteriza por tener un término de parada en las iteraciones de cálculo de pesos, cuando la generalización deja de mejorar, es decir cuando el error cuadrático medio de las muestras de validación indica un aumento [9]. Finalmente se pasan los resultados obtenidos, es decir nuestra red ya entrenada, a un bloque de función de simulink y se procede a comprobar el esquema de control conectado la red en paralelo al PID teniendo como entrada a la referencia y su salida se suma a la acción de control del PID. En la figura 2.17 se muestra el esquema simulado junto con la red neuronal entrenada.



**Figura 2.17** Diagrama de bloques simulado para esquema PID+ Red Neuronal.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] E. Anchitipan, "Diseño, simulación y comparación de esquemas de control adaptativo aplicados a plantas no lineales de parámetros variables y retardo de tiempo dominante," Aug. 2020, Accessed: Aug. 10, 2021. [Online]. Available: <http://bibdigital.epn.edu.ec/handle/15000/21075>
4. [2] E. Anchitipan and O. Camacho, "Processes with Variable Parameters and Dominant Time Delay: A Fuzzy Adaptive Control Approaches Comparison," 2021, pp. 181–195. doi: 10.1007/978-3-030-72208-1\_14.
- [3] E. Iglesias, "Using fuzzy logic to enhance control performance of sliding mode control and dynamic matrix control," Tesis Doctoral, University of South Florida, South Florida, 2006.
- [4] V. Alfaro, "Metodos de Sintonización de PID que funcionan como reuladores" Revista de la Universidad de Costa Rica, vol. 10, N° 1 y 2, pp 5-6, Enero 2002, San José-Costa Rica.
- [5] OMATU, S. Neuro-control and Its Applications to Electric Vehicle Control. *IWANN'09 Proceedings of the 10th Interntaional Work-Conference on Aritificial Neural Networks: Part II: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living*. 2009, pp. 1 – 12.
- [6] J. D. R. Antolines, D. F. C. Quiroga, and M. Á. P. Pachón, "Implementación de neurocontroladores en línea: tres configuraciones, tres plantas," *Ingeniería y universidad*, vol. 16, no. 1, pp. 163–182, 2012.
- [7] Kamalasadán, S. y Ghandakly, A. A Neural Network Parallel Adaptive Controller for Dynamic System Control. *IEEE Transactions on Instrumentation and Measurement*, 2007, vol. 56, núm 5, pp. 1786 – 1796.
- [8] A. Marino and F. Neri, "PID Tuning with Neural Networks," 2019, pp. 476–487. doi: 10.1007/978-3-030-14799-0\_41.
- [9] T. Pröll and M. N. Karim, "Model-predictive pH control using real-time NARX approach," *AIChE Journal*, vol. 40, no. 2, pp. 269–282, 1994, doi: 10.1002/aic.690400207.
- [10] T. K. Gustafsson y K. V. Waller, «Dynamic modeling and reaction invariant control of pH», *Chem. Eng. Sci.*, vol. 38, n.º 3, pp. 389–398, 1983.
- [11] K. V. Waller y P. M. Makila, «Chemical reaction invariants and variants and their use in reactor modeling, simulation, and control», *Ind. Eng. Chem. Process Des. Dev.*, vol. 20, n.º 1, pp. 1–11, 1981.
- [12] S. Narendra ,K Parthasarathy , " Identification and Control of Dynamical Systems Using Neural Networks", *IEEE TRANSACTIONS ON NEURAL NETWORKS*, VOL. I, NO. I, MARCH 1990.
- [13] «RNR NARX», *Numerentur.org*, Recuperado de: [RNR NARX – Numerentur.org](http://Numerentur.org)
- [14] R. Fonseca, P. Gómez" *Redes Neuronales no lineales con entradas externas NARX*", National Institute of Astrophysics, Optics and Electronics, Mexico DC, 2009.

## **ANEXOS**

En el presente capítulo se muestra el desarrollo de la información adicional mencionada dentro del capítulo dos del trabajo, el cual contiene los siguientes ítems.

### **5.**

ANEXO I. Desarrollo del análisis matemático del reactor de neutralización de pH.

ANEXO II. Código de Matlab para obtención de parámetros FOPDT

ANEXO III. Código de Matlab para entrenamiento de red neuronal empleada en el esquema de PID neuronal

ANEXO IV. Detalle de la estructura y funcionamiento de redes neuronales NARX.

ANEXO V. Detalle del proceso de entrenamiento de la red neuronal a través del toolbox Neural Series Time de Matlab.

## ANEXO I

Tal como se mencionó en la sección 2.1 en este anexo se detalla el análisis matemático del reactor de neutralización de pH, para lo cual se deben considerar las siguientes características físicas del proceso.

- El volumen del líquido en el reactor se mantiene dentro de los límites del mismo, es decir, varía sin desbordarse del reactor.
- Los flujos que llegan al reactor son mezclados de manera perfecta y homogénea.
- El sistema tanque y tubería se encuentran bien aislados, por lo que no existe intercambio de calor desde el exterior, ni entre ellos.
- Al momento de la mezcla se considera una completa solubilidad de los iones de los compuestos químicos involucrados.
- Se considera como perturbación del sistema al flujo de corriente ácida  $q_1(t)$ . que ingresa al tanque 2.

Teniendo en cuenta las consideraciones anteriores, se procede a plantear las ecuaciones físicas y químicas que describen el proceso, partiendo así del análisis dinámico físico del mismo.

En primera instancia se consideran las ecuaciones químicas obtenidas mediante el balance de reacciones producidas en el reactor.[1]



Teniendo en cuenta el cociente de reacción, determinan las constantes químicas de equilibrio en el proceso.[1]

$$K_{a1} = \frac{[HCO_3^-][H^+]}{[H_2CO_3]} \quad (5.4)$$

$$K_{a2} = \frac{[CO_3^{2-}][H^+]}{[HCO_3^-]} \quad (5.5)$$

$$K_w = [H^+][OH^-] \quad (5.6)$$

El equilibrio químico es modelado por medio de la introducción de dos reacciones invariantes  $W_a$  y  $W_b$  para cada flujo de entrada [10], [11]. La reacción  $W_a$  es una cantidad relacionada con la carga de los iones, mientras que  $W_b$  indica la concentración del ion  $CO_3^{2-}$ . Las reacciones invariantes son las siguientes:

$$W_{ai} = [H^+]_i - [OH^-]_i - [HCO_3^-]_i - 2[CO_3^{2-}]_i \quad (5.7)$$

$$W_{bi} = [H_2CO_3]_i + [HCO_3^-]_i + [CO_3^{2-}]_i \quad (5.8)$$

Donde  $i$  va desde 1 a 4 y representa los flujos que actúan en el proceso.

A través de las ecuaciones de las constantes de equilibrio y de las reacciones invariantes se puede hallar una expresión para determinar el nivel de pH, Ecuación 5.9 y 5.10 [1].

$$W_b \frac{\frac{K_{a1}}{[H^+](t)} + \frac{2K_{a1}K_{a2}}{[H^+](t)^2}}{1 + \frac{K_{a1}}{[H^+](t)} + \frac{K_{a1}K_{a2}}{[H^+](t)^2}} + W_a + \frac{K_w}{[H^+](t)} - [H^+](t) = 0 \quad (5.9)$$

$$pH(t) = -\log[H^+](t) \quad (5.10)$$

A continuación, se presenta las ecuaciones que describen el comportamiento del proceso bajo las consideraciones mencionadas anteriormente.

Realizando el balance masa en el tanque 2, se tiene la ecuación 5.11 [1]

$$q_1(t)\rho - q_{1e}(t)\rho = A_2\rho \frac{dh_2(t)}{dt} \quad (5.11)$$

De la misma manera para el reactor de neutralización, se realiza el balance de masa, con lo que se obtiene la ecuación 5.12. [1]

$$q_{1e}(t)\rho + q_2(t)\rho + q_3(t)\rho - q_4(t)\rho = A_1\rho \frac{dh_1(t)}{dt} \quad (5.12)$$

Una vez determinadas las ecuaciones de balance de masa, se procede a detallar las fórmulas de flujo de las válvulas del flujo de salida  $q_4(t)$  y de la de entrada al reactor  $q_{1e}(t)$  teniendo las ecuaciones 5.14 y 5.13 respectivamente.[1]

$$q_{1e}(t) = C_{V2}\sqrt{h_2(t)} \quad (5.13)$$

$$q_4(t) = C_{V4}(h_1(t))^n \quad (5.14)$$

Empleando el principio de conservación de masa relacionada con las reacciones químicas invariantes para el reactor de neutralización y tanque 2.[1]

$$q_{1e}(t)W_{a1} + q_2(t)W_{a2} + q_3(t)W_{a3} - q_4(t)W_{a4}(t) = A_1 \frac{d(h_1(t)W_{a4}(t))}{dt} \quad (5.15)$$

$$q_{1e}(t)W_{b1} + q_2(t)W_{b2} + q_3(t)W_{b3} - q_4(t)W_{b4}(t) = A_1 \frac{d(h_1(t)W_{b4}(t))}{dt} \quad (5.16)$$

Las ecuaciones que caracterizan la dinámica del proceso se detallan a continuación

Considerando el retardo del tiempo introducido por el transmisor de pH, se tiene.[1]

$$pH'(t) = pH(t - t_0(t)) \quad (5.17)$$

Donde le retardo de tiempo es.[1]

$$t_0(t) = \frac{LA\rho}{q_4(t)} \quad (5.18)$$

La señal del transmisor de pH, está definida por:[1]

$$\frac{dc(t)}{dt} = \frac{1}{\tau_T} [K_T pH'(t) - c(t)] \quad (5.19)$$

Y finalmente se representa la ecuación de variación de flujo de la válvula de control de flujo básico.[1]

$$\frac{dq_3(t)}{dt} = \frac{1}{\tau_V} [K_V m(t) - q_3(t)] [1] \quad (5.20)$$

Donde: [1]

- $q_1(t)$  : Flujo de la corriente ácida, ml/s.
- $q_{1e}(t)$  : Flujo de la corriente ácida de salida desde el tanque 2, ml/s.
- $q_2(t)$  : Flujo del corriente buffer, ml/s.
- $q_3(t)$  : Flujo de la corriente básica, ml/s.
- $q_4(t)$  : Flujo de la corriente de salida, ml/s.
- $pH(t)$  : Potencial de hidrogeniones del líquido en el reactor, adimensional.
- $pH'(t)$  :  $pH(t)$  considerando el retardo de tiempo  $t_0(t)$ , adimensional.
- $t_0(t)$  : Retardo de tiempo, s.
- $\rho$  : Densidad de los flujos, ml/cm<sup>3</sup>.
- $A_1$  : Sección transversal del reactor de neutralización de pH, cm<sup>2</sup>.
- $A_2$  : Sección transversal del tanque 2, cm<sup>2</sup>.
- $h_1(t)$  : Nivel del líquido en el reactor de neutralización de pH, cm.
- $h_2$  : Nivel del líquido en el tanque 2, cm.
- $C_{V2}$  : Coeficiente del flujo de la válvula manual 2, (ml/s)/cm<sup>1/2</sup>.
- $C_{V4}$  : Coeficiente del flujo de la válvula manual 4, (ml/s)/cm<sup>0.607</sup>.
- $n$  : Coeficiente de la válvula manual 4.
- $c(t)$  : Señal de salida del transmisor de pH en una escala de 0 a 100, %.
- $m(t)$  : Salida del controlador de 0 a 100, %.
- $\tau_T$  : Constante de tiempo del sensor de pH, s.

- $\tau_V$  : Constante de tiempo de la válvula de control, s.  
 $K_T$  : Ganancia del sensor de pH, %.  
 $K_V$  : Ganancia de la válvula de control, (ml/s)/%CO.  
 $A$  : Sección transversal de la tubería, cm<sup>2</sup>.  
 $L$  : Longitud de la tubería, cm.

De acuerdo a Iglesias [14] en la tabla 5.1 se presentan los valores de operación en estado estacionario del reactor de neutralización de pH.[1]

**Tabla 5.1** Valores de operación estacionaria para el reactor de neutralización de pH

Variable	Valor	Variable	Valor
$q_1$	16.6 ml/s	$K_T$	7.1429 %
$q_{1e}$	16.6 ml/s	$K_V$	0.3 (ml/s)/%CO
$q_2$	0.55 ml/s	$A$	6.01031 cm <sup>2</sup>
$q_3$	15.6 ml/s	$L$	674.5 cm
$q_4$	32.75 ml/s	$[q_1]$	0.003M HNO <sub>3</sub>
Referencia	7.025	$[q_2]$	0.03M NaHCO <sub>3</sub>
$pH$	7.025	$[q_3]$	0.003M NaOH + 0.00005M NaHCO <sub>3</sub>
$\rho$	1 ml/cm <sup>3</sup>	$K_{a1}$	$4.47 \times 10^{-7}$
$A_1$	207 cm <sup>2</sup>	$K_{a2}$	$5.62 \times 10^{-11}$
$A_2$	42 cm <sup>2</sup>	$K_w$	$1 \times 10^{-14}$
$h_1$	25.5 cm	$W_{a1}$	0.003 M
$h_2$	3 cm	$W_{b1}$	0 M
$C_{V2}$	9.584 (ml/s)/cm <sup>1/2</sup>	$W_{a2}$	-0.03 M
$C_{V4}$	4.5861 (ml/s)/cm <sup>0.607</sup>	$W_{b2}$	0.03 M
$n$	0.607	$W_{a3}$	$-3.05 \times 10^{-3}$ M
$c$	50.18 %	$W_{b3}$	$5 \times 10^{-5}$ M
$m$	52 %	$W_{a4}$	$-4.36 \times 10^{-4}$ M
$\tau_T$	15 s	$W_{b4}$	$-5.276 \times 10^{-4}$ M
$\tau_V$	6 s		



## ANEXO II

A continuación, se muestra el código empleado para la obtención de parámetros FOPDT de manera automática.

- Código para cambios de subida.

```
clc
n=2
VI=52*(1+0.084*(n)); %% Valor final entrada
VF=52*(1+0.084*(n+1)); %% Valor inicial entrada
sim('AproximacionesFOPDT'); %Simulación.
yf=max(SalidaTransmisor); %Valor final salida
yi=SalidaTransmisor(10889,1);%% Valor inicial salida
deltay=(yf-yi); %%Variación de la salida
K=deltay/(VF-VI) %% Constante de ganancia.
Yt1=0.283*deltay+yi %%valor que alcanza el 28.32%
Yt2=0.632*deltay+yi %%valor que alcanza el 63.2%
for i=1:4001
    if(abs(SalidaTransmisor(i)-Yt1)<=0.01)
        t1=i %% Tiempo que alcanza el 28.32%
    end
    if(abs(SalidaTransmisor(i)-Yt2)<=0.01)
        t2=i %% Tiempo que alcanza el 28.32%
    end
end
t1=t1-1500;
t2=t2-1500;
tao=1.5*(t2-t1); %% Costante de tiempo
to=t2-tao; %% Retardo de tiempo
r=to/tao; %% Relación entre tao/to
```

- Código empleado para cambios descendentes.

```

clc
n=2
VI=52*(1-0.084*(n)); %% Valor final entrada
VF=52*(1-0.084*(n+1)); %% Valor inicial entrada
sim('AproximacionesFOPDT'); %Simulación.
yi=max(SalidaTransmisor); %Valor final salida
yf=SalidaTransmisor(10889,1);%% Valor inicial salida
deltay=(yf-yi); %%Variación de la salida
K=deltay/(VF-VI) %% Constante de ganancia.
Yt1=0.283*deltay+yi %%valor que alcanza el 28.32%
Yt2=0.632*deltay+yi %%valor que alcanza el 63.2%
for i=1:4001
    if(abs(SalidaTransmisor(i)-Yt1)<=0.01)
        t1=i %% Tiempo que alcanza el 28.32%
    end
    if(abs(SalidaTransmisor(i)-Yt2)<=0.01)
        t2=i %% Tiempo que alcanza el 28.32%
    end
end
end
t1=t1-1500;
t2=t2-1500;
tao=1.5*(t2-t1); %% Costante de tiempo
to=t2-tao; %% Retardo de tiempo
r=to/tao; %% Relación entre tao/to

```

## ANEXO III

En el presente anexo se muestra el código empleado para el entrenamiento online de la red neuronal utilizada en el esquema de control con PID neuronal.

```
function [dkp,dki,dkd] = fcn(u,t)
error=u-t
dkp=0;
dki=0;
dkd=0;
sum=0;
n1 = 3;
n2 = 3;
ep = 1;
W1 = 2*ep*rand(n1,1) - ep;
b1 = 2*ep*rand(n1,1) - ep;
W2 = 2*ep*rand(n2,n1) - ep;
b2 = 2*ep*rand(n2,1) - ep;
W3 = 2*ep*rand(1,n2) - ep;
b3 = 2*ep*rand - ep;

alfa = 0.001;
for Epocas = 1:2000
    sum = 0;
    %q = randi(Q);
    % Propagación de la entrada hacia la salida
    a1 = (W1*t + b1);
    a2 = (W2*a1 + b2);
    a3 = W3*a2 + b3;
    % Retropropagación de la sensibilidades
    e = u-a3;
    s3 = -2*1*e;
    s2 = diag((1-a2).*a2)*W3'*s3;
    s1 = diag((1-a1).*a1)*W2'*s2;
    % Actualización de pesos sinapticos y polarizaciones
    W3 = W3 - alfa*s3*a2';
    b3 = b3 - alfa*s3;
    W2 = W2 - alfa*s2*a1';
    b2 = b2 - alfa*s2;
    W1 = W1 - alfa*s1*t';
    b1 = b1 - alfa*s1;
    % Sumando el error cuadratico
    sum = e^2 + sum;
end
dkp=W1(1)
dki=W2(2)
dkd=W3(3)
```

## ANEXO IV

En el presente anexo, se hace una explicación más a detalle de la estructura y funcionamiento de las redes neuronales No-Lineales Autorregresivas con entradas Exógenas (NARX).

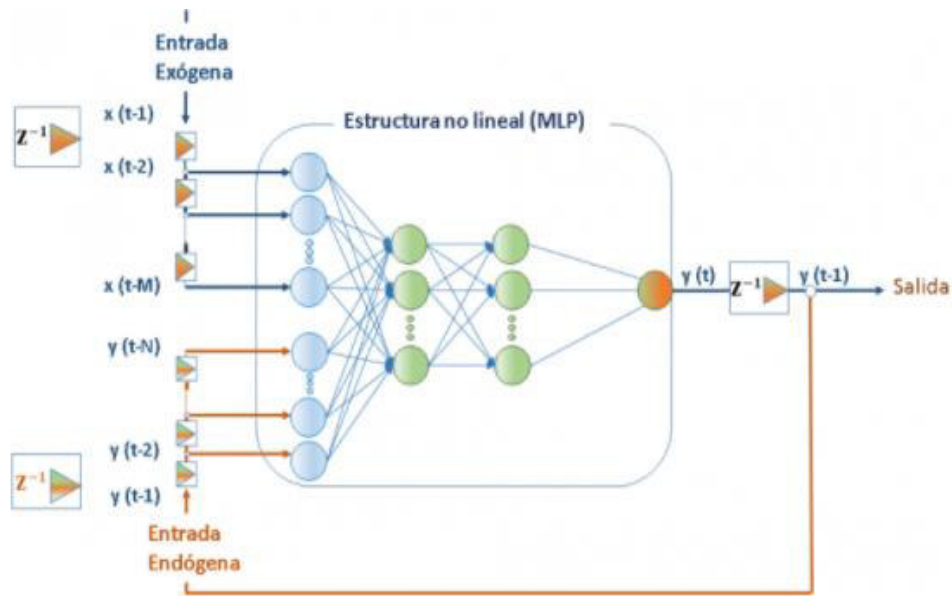
### **Arquitectura.**

Las redes NARX fueron propuestas como estructuras de identificación de sistemas en 1990 por Kumpati S. Narendra y Kannan Parthasarathy [12]. Su arquitectura se basa en una red dinámica recurrente, es decir el cálculo de la salida depende del valor anterior de la misma, hecho que la diferencia de las redes feedforward las cuales su valor de salida dependen únicamente de sus entradas, teniendo en cuenta esto se puede decir que una red recurrente es una red que crea series de tiempo, de ahí que se las denomina en muchas bibliografías como redes neuronales de series de tiempo, la principal ventaja de estas redes neuronales son su dinámica, la cual permite trabajar como herramienta de aproximación universal y como herramienta de predicción es de ahí su utilidad dentro del control de procesos como estructuras de identificación. En la figura 5.1 se muestra la arquitectura de la redes NARX, donde se puede observar la capa de entradas exógenas junto con un serie de tiempo que crea los retardos en la entrada, luego se pasa a la capa oculta donde se tiene un modelo Multicapa Perceptrón (MLP) [13] con funciones de activación no lineal (Sigmoide) para luego dar paso a la capa de salida, donde también se crea un retardo del dato de salida para luego ser devuelto como entrada para el recálculo de la salida.

### **Propiedades:**

- Su salida depende de valores actuales, pasados y en ocasiones futuros de sus entradas y salidas.
- Son las más utilizadas en identificación y predicción de datos.
- Presentan una potente herramienta de cálculo con precisión casi perfecta para la aproximación de datos.
- Las redes NARX pueden mediante la configuración adecuada pueden simular a las redes neuronales recurrente totalmente conectadas, siempre y cuando la función de activación sea la adecuada [14].

- Tienen la desventaja de presentar una desaceleración lineal, es decir implica que si una red neuronal recurrente normal computa una tarea de interés en tiempo T, entonces el tiempo total utilizado por su red NARX es (N+1)T [14].



**Figura 5.1** Arquitectura red neuronal NARX. Tomado de [13]

**Función matemática.**

Tal como lo indica en su arquitectura la salida de la red NARX viene dada por una función no lineal de sus entradas y salida tanto actuales como anteriores [14], teniendo así que:

$$y(t+n) = q(y(t+n-1), \dots, y(t); u(t+n), \dots, u(t)) \tag{5.21}$$

Función no lineal

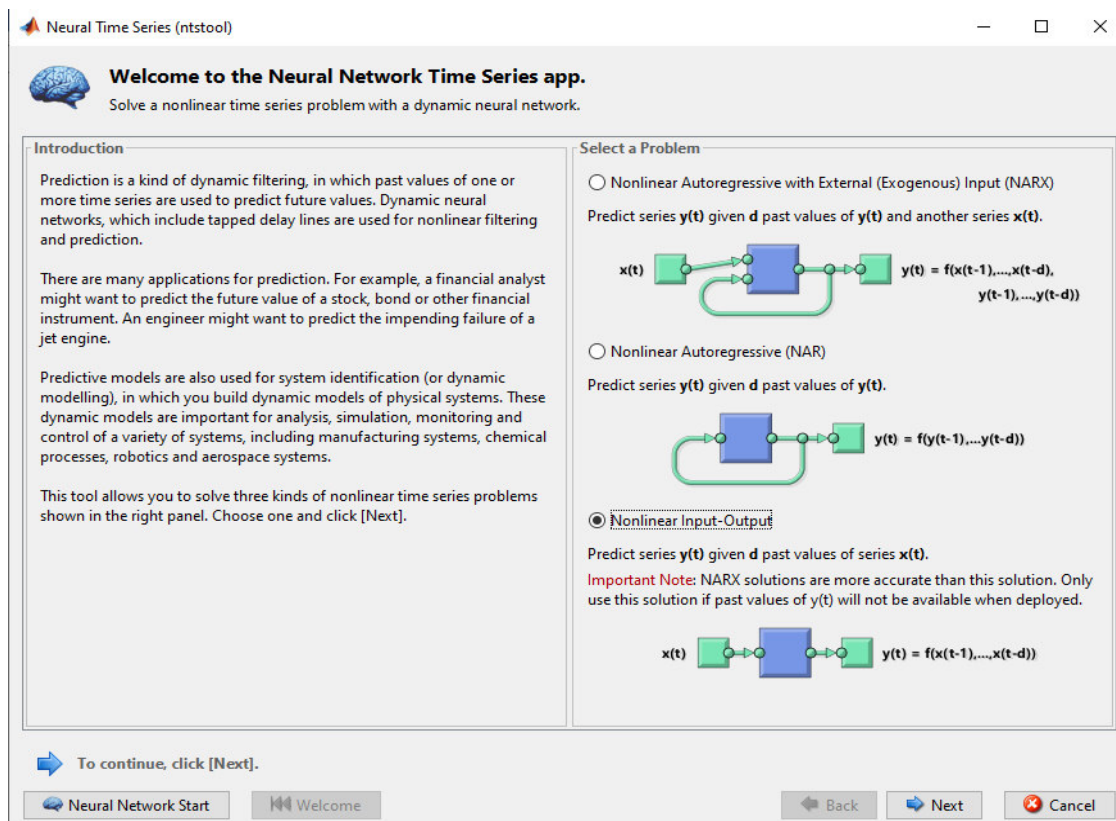
Entradas Exógenas

Observaciones

## ANEXO V

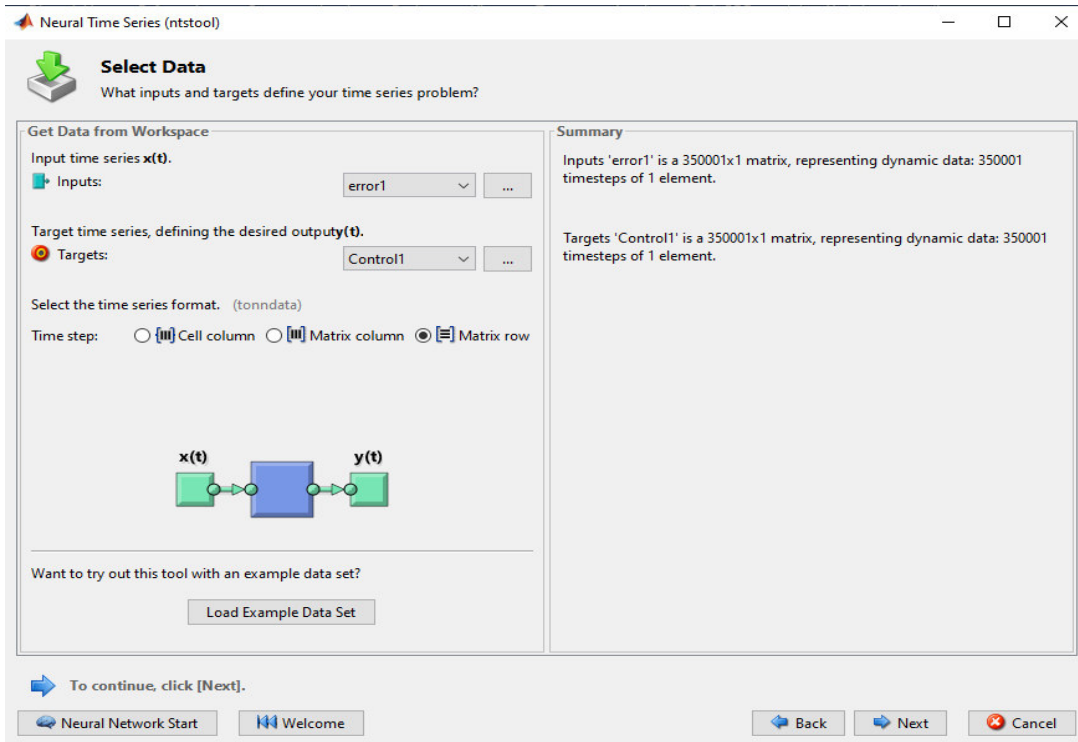
A continuación, se muestra los pasos a seguir para el entrenamiento de una red neuronal feedforward tipo NARX a través del toolbox Neural Time Series de Matlab, una vez que ya tiene en el Workspace nuestra base de datos.

Primero ingresamos al “toolbox” por medio de “*ntstool*” en la ventana de comandos de Matlab o buscamos dentro de la pestaña APPS, el bloque Neural Net Time Series, de donde nos aparecerá la siguiente ventana. Ver Figura 5.1



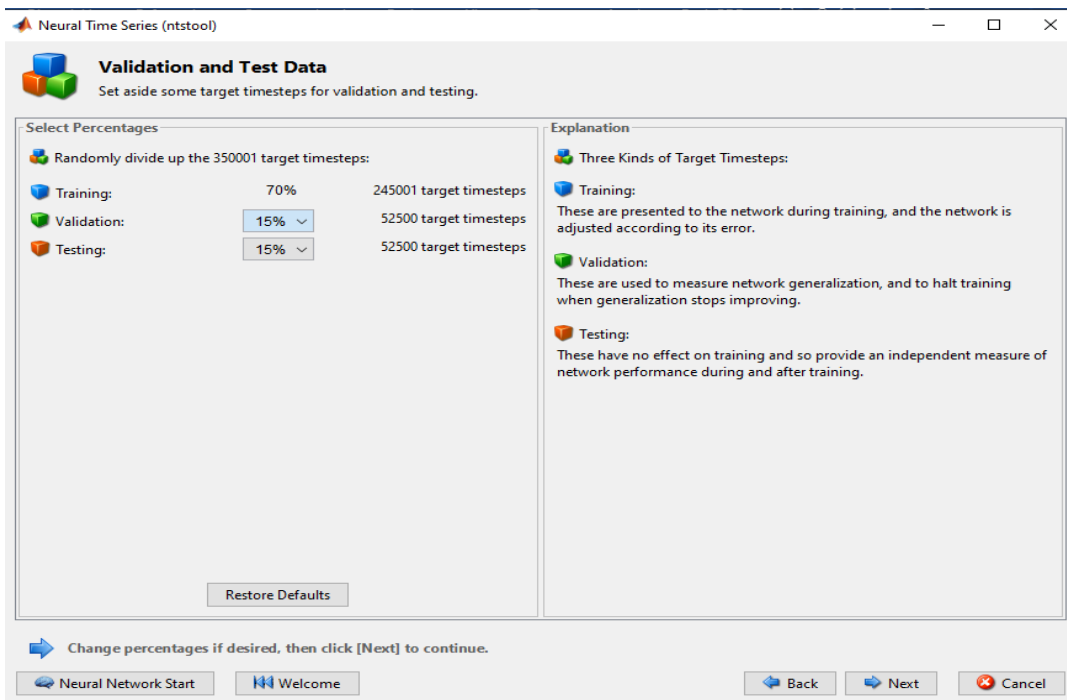
**Figura 5.1** Ventana principal del toolbox de Matlab, “Neural Time Series” .

Seleccionamos la red tipo NARX “Nonlinear input-output”, como se muestra en la figura 5.1, y presionamos NEXT, donde nos aparecerá la siguiente ventana. Ver figura 5.2



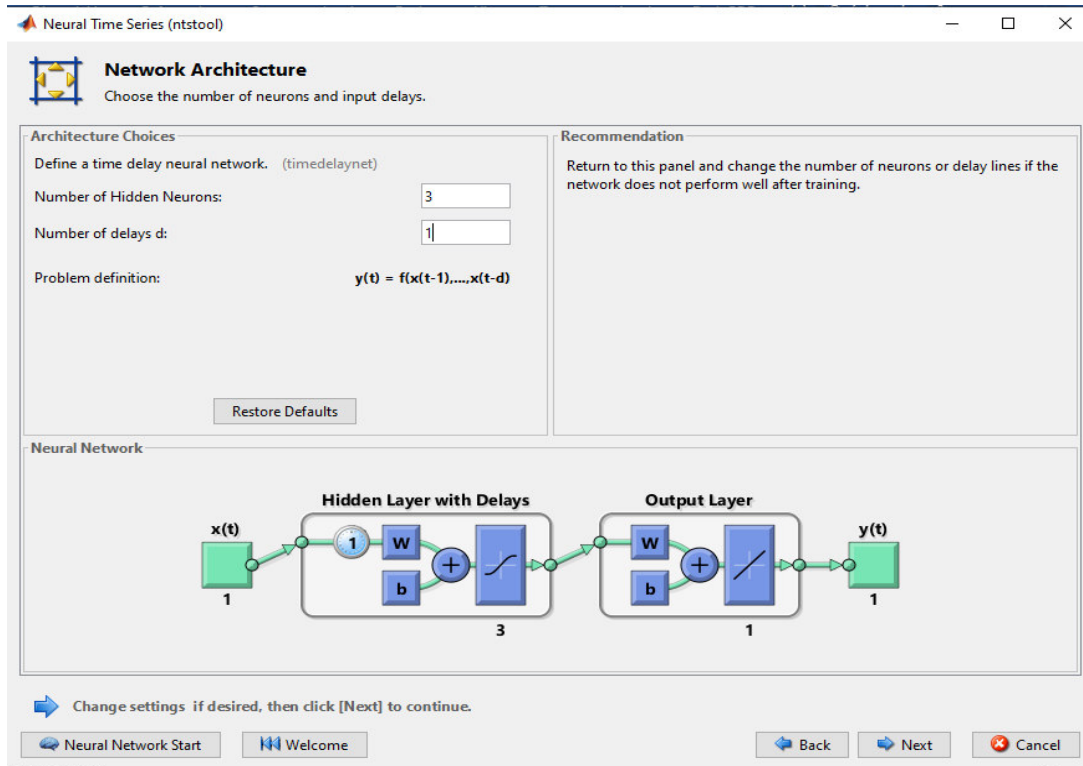
**Figura 5.2** Selección de datos para entrenamiento de la red neuronal NARX.

En la figura 5.2 se debe seleccionar nuestra base de datos, que en este caso es tomada como “error1” como entrada y “Control1” como salida, además se deberá seleccionar el formato de columnas para poder entrenar la red, luego presionamos NEXT nuevamente.



**Figura 5.3** Selección de porcentajes para entrenamiento y validación de la red.7

Una vez seleccionada la base de datos, tendremos que seleccionar los porcentajes de entrenamiento, validación y prueba de la red, para lo cuál dejamos los valores por defecto, ya que son los recomendados por la mayoría de bibliografías consultadas en el capítulo 2, los cuál se muestra en la figura 5.3, por lo que solo damos click en NEXT donde nos aparecerá una ventana como la mostrada en la figura 5.4



**Figura 5.4** Selección del número de capas ocultas y retardos de la red.

En esta parte debemos seleccionar el número de capas ocultas de nuestra red, la cuál no sigue una regla definida, por lo que se debe probar con distintos números de capas, en cuanto al retardo en nuestro caso se seleccionó un retardo de una muestra.

Luego de esto finalmente damos click en NEXT y procedemos a dar click en TRAIN (Entrenar) en la nueva ventana que aparece como se muestra en la figura 5.5.



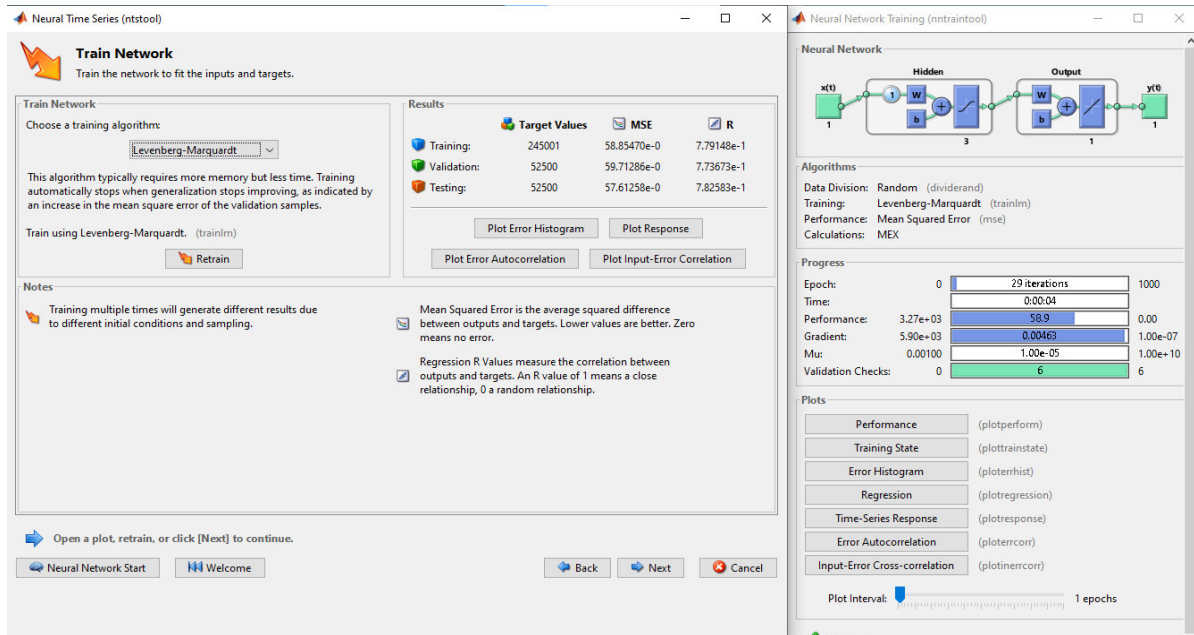


Figura 5.5 Entrenamiento de la red neuronal NARX creada previamente.

Una vez terminado el entrenamiento, el toolbox permite expresar la red neuronal creada y entrenada como una función o un diagrama de bloques de Simulink, por lo que en nuestro caso creamos un bloque de Simulink, como se puede observar en la figura 5.6.

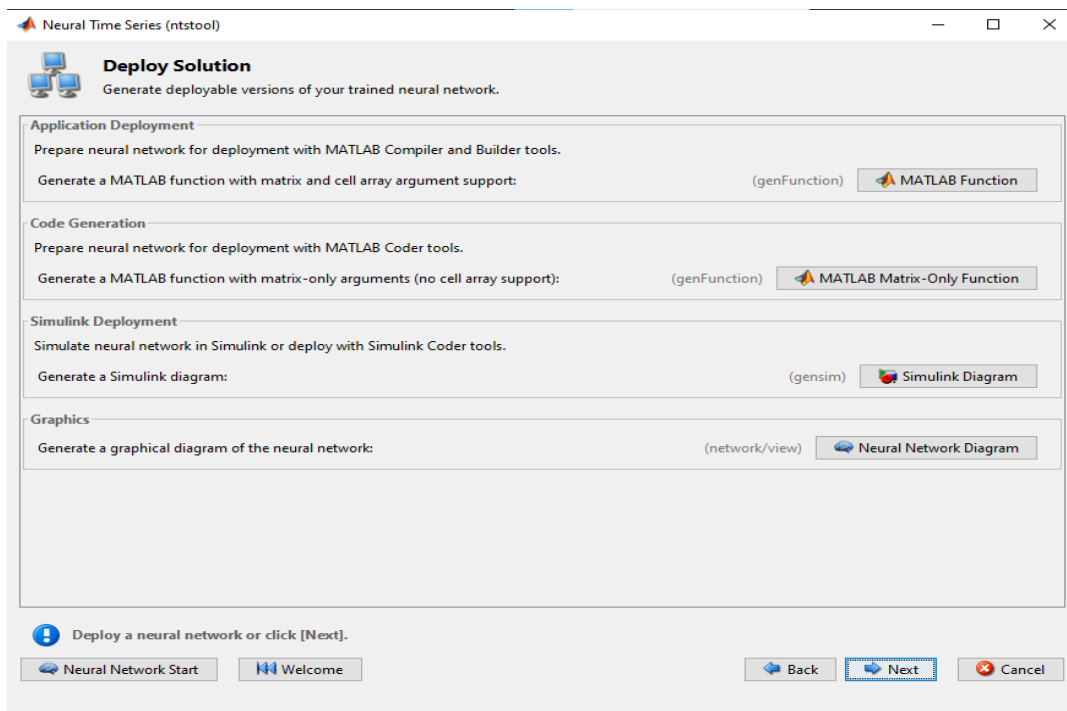


Figura 5.6 Expresión de resultados de entrenamiento de red neuronal.

