



UNIVERSIDAD
DE PIURA

REPOSITORIO INSTITUCIONAL
PIRHUA

IDENTIFICACIÓN DE MODELOS WIENER Y HAMMERSTEIN APLICADOS AL DISEÑO DE UN CONTROL PID PREDICTIVO DE PH

Irene Alvarado-Tabacchi

Piura, diciembre de 2018

FACULTAD DE INGENIERÍA

Máster en Ingeniería Mecánico-Eléctrica con Mención en Automática y
Optimización

Alvarado, I. (2018). *Identificación de modelos Wiener y Hammerstein aplicados al diseño de un control PID predictivo de pH* (Tesis de Máster en Ingeniería Mecánico-Eléctrica con Mención en Automática y Optimización). Universidad de Piura. Facultad de Ingeniería. Piura, Perú.



Esta obra está bajo una licencia

[Creative Commons Atribución-NoComercial-SinDerivar 4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/)

[Repositorio institucional PIRHUA – Universidad de Piura](#)

UNIVERSIDAD DE PIURA

FACULTAD DE INGENIERÍA



**Identificación de modelos Wiener y Hammerstein aplicados al diseño de
un control PID predictivo de pH**

**Tesis para optar el Grado de Máster en Ingeniería Mecánico-Eléctrica
con mención en Automática y Optimización**

Irene Alvarado Tabacchi

Asesor: Dr. Ing. William Ipanaqué Alama

Piura, diciembre 2018

A Dios y a la Virgen

A mi familia y amigos, por todo su apoyo para hacer
realidad este sueño

Prólogo

El desarrollo de esta tesis, en concordancia con el objetivo del laboratorio de sistemas automáticos de control, busca contribuir con la aplicación del conocimiento para alcanzar mejoras en la calidad de los productos, ahorro energético, valor agregado, de manera de satisfacer los requerimientos del mercado cada vez más competitivo. Esto exige la mejora de los parámetros de producción y la reducción de costos. En el ámbito del control automático de procesos, se han desarrollado muchas alternativas tanto a nivel de investigación como de aplicación industrial. Se utilizan técnicas de control automático que van desde las técnicas PID como las desarrolladas con tecnología avanzada: lógica difusa, predictivo, inteligencia artificial, redes neuronales, control robusto, etc.

El tema de este estudio es una iniciativa que plantea aprovechar las características y prestaciones de los algoritmos avanzados de predicción, combinados con la metodología de los controles ampliamente conocidos. En este sentido, hay técnicas de mejoras de prestaciones, como los PID adaptativos, control IMC, Predictor de Smith, que son de sintonización relativamente simple, pero requieren de modelos o la caracterización de éste con pruebas empíricas, con la limitación que se comportan muy bien en el caso de modelos lineales. Esta limitación es lo que origina la necesidad de contar con modelos no lineales que además permitan su aplicación en soluciones de control de procesos que combinen de una manera relativamente simple las características y bondades de ambas metodologías, esto es: PID con características predictivas.

Los modelos orientados a bloques, para el caso del control de pH, han sido estudiados en distintos trabajos de investigación de control no lineal, por lo que se abre un panorama que facilite su implementación en las técnicas de control antes mencionadas en las que se requiere contar con un modelo que mejor se adapte al proceso que se desea controlar. La contribución de esta tesis se ha orientado a ello: desarrollar modelos no lineales para la mejora en el control de procesos; y aplicar un control predictivo con características PID.

Como resultado de la tesis se ha publicado un artículo SCOPUS: “*Wiener Predictive control for a pH Neutralization plant*” en el 2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON) (Anexo 1). Además se tuvieron dos estancias de investigación una en Alemania y otra en la Universidad Católica de Chile. Estas experiencias aportaron mayor conocimiento y experiencia en el trabajo desarrollado.

Quisiera expresar mi agradecimiento por el apoyo de la Universidad de Piura, por la formación académica, personal y profesional en todos estos años de servicio y mi agradecimiento a la facultad de Ingeniería que ha facilitado mi dedicación al desarrollo del proyecto de tesis y en especial a mi profesor asesor Dr. William Ipanaqué por su comprensión, su constante apoyo y su aliento para no abandonar este reto, hasta llegar a hacer realidad este anhelo.

Asimismo mi agradecimiento a Innóvate Perú en el marco del proyecto PIAP-3-P-1054-14 financiado por Innóvate Perú y la Universidad de Piura, que ha motivado la investigación de identificación y control de procesos no lineales, como es el caso del proceso de digestión anaerobia, en la que uno de los parámetros a controlar para favorecer la producción de biogás es el valor de pH que se debe mantener alrededor del valor neutro.

Resumen

Esta tesis propone analizar la aplicación del control de procesos no lineales en bioprocesos, siendo el pH uno de los parámetros a controlar.

En el caso de un proceso no lineal como el de una planta de neutralización de pH se han estudiado los métodos de identificación orientados a bloques, para obtener modelos Wiener y Hammerstein. Estos modelos se han usado en la implementación de un control predictivo PID basado en la estructura de un control MPC (Model Predictive Control), dando lugar a un algoritmo de control denominado WPID predictivo y HPID predictivo según se use un modelo Wiener o Hammerstein.

Los métodos de identificación y los algoritmos de control estudiados se han aplicado a una plataforma de simulación de una planta no lineal de neutralización de pH implementada en MATLAB. Los algoritmos de control predictivo WPIDp y HPIDp desarrollados se han comparado con técnicas clásicas de diseño PID y de control MPC, evidenciándose una notable mejora, en especial en el algoritmo WPIDp.

Se han publicado parte de los resultados obtenidos en un artículo SCOPUS: “Wiener Predictive control for a pH Neutralization plant” en el 2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON) (Anexo 1).

Tabla de contenido

Introducción	1
Capítulo 1 Métodos de identificación de modelos no lineales orientados a bloques	3
1.1. Estado del arte de las técnicas de identificación lineal	4
1.1.1. Modelos de bloques lineales	5
1.1.1.1. Función de transferencia racional	5
1.1.1.2. Respuesta impulsiva FIR (Finite Impulse Response)	7
1.1.1.3. Respuesta a escalón FSR (Finite Step Response)	7
1.1.1.4. Bases ortonormales	8
1.1.1.5. Función Laguerre	12
1.1.1.6. Representación en espacio de estado	14
1.1.2. Modelos de bloques no lineales	14
1.1.2.1. Funciones no lineales	15
1.1.2.2. Modelos Volterra	17
1.1.2.3. Modelos NARX y NARMAX	19
1.1.2.4. Modelos Hammerstein	19
1.1.2.5. Modelos Wiener	21
1.1.2.6. Modelos Hammerstein-Wiener	22
1.2. Métodos de identificación	22
1.2.1. Identificación en espacio de estado con método N4SID y descomposición SVD	23
1.2.1.1. Identificación del modelo Hammerstein	23
1.2.1.2. Identificación del modelo Wiener	26
1.2.2. Identificación usando bases ortonormales y descomposición SVD	28
1.2.2.1. Identificación modelos Hammerstein	28
1.2.2.2. Identificación modelos Wiener	31

Capítulo 2 Diseño de simulador de proceso de neutralización de pH	35
2.1. Estado del arte de los simuladores de pH en la literatura	35
2.2. Estado del arte de control de plantas de neutralización de pH	43
2.3. Descripción del proceso a modelar	46
2.3.1. Conceptos básicos en procesos de neutralización de pH	46
2.3.1.1. Definición ácido base	46
2.3.1.2. Equilibrio iónico del agua	47
2.3.1.3. Definición de pH	47
2.3.1.4. Reacciones de neutralización	48
2.3.1.5. Valoración ácido-base	49
2.3.2. Reacciones de neutralización del proceso a simular	50
2.4. Diseño del simulador de pH.....	52
 Capítulo 3 Obtención de los modelos Wiener y Hammerstein a partir de un simulador.....	 59
3.1. Señales de prueba para identificación.....	59
3.1.1. Pruebas recomendadas	60
3.1.1.1. Duración del experimento	63
3.1.1.2. Selección tiempo de muestreo	63
3.1.2. Diseño de los experimentos para la planta de pH	63
3.2. Pruebas para la identificación	65
3.2.1. Ensayos preliminares para la identificación.....	70
3.2.1.1. Análisis de respuesta a variaciones escalón	70
3.2.2. Prueba PRS y obtención de los datos E/S	71
3.3. Identificación lineal	72
3.4. Identificación modelos Wiener y Hammerstein	73
3.4.1. Método N4SID con descomposición SVD	73
3.4.2. Método con bases ortonormales y descomposición SVD	77
3.4.3. Modelos Wiener y Hammerstein usando intrucción nlhw de MATLAB	79
 Capítulo 4 Controlador PID predictivo para pH	 85
4.1. Controlador MPC predictivo con estructura en espacio de estado	86
4.1.1. MPC con modelo en espacio de estado.....	87
4.2. Control PID predictivo.....	92
4.3. Control PID predictivo con modelo Wiener (WPIDp)	96
4.3.1. Control PID predictivo con modelo Wiener (WPIDp) y control MPC con modelo Wiener (WMPC)	98
4.3.2. Control PID predictivo con modelo Wiener (WPIDp) y control MPC para modelo lineal	101

4.3.3. Comparación control WPIDp y control PI clásico.....	103
4.4. Control PID predictivo con modelo Hammerstein (HPIDp)	107
4.4.1. Control PID predictivo con modelo Hammerstein y control MPC para modelo lineal	109
4.4.2. Comparación control HPIDp y control PI clásico.....	110
Conclusiones y recomendaciones	111
Bibliografía.....	113
Anexos.....	121
Anexo A. Artículo presentado en el 2017 CHILEAN Conference on Electrical, Electronics, Engineering, Information and Communication Technologies, 2017, Chile	123
Anexo B Programas de identificación y control	130

Lista de Tablas

Tabla 1.	Modelos paramétricos de identificación	6
Tabla 2.	Condiciones de operación nominal	64
Tabla 3	Ganancia del proceso ante variaciones escalón a la entrada de la planta	69
Tabla 4	Caracterización de la respuesta a escalón	70
Tabla 5	Índices de <i>performance</i> de los modelos Wiener y lineal	73
Tabla 6.	Índice de <i>performance</i> de los modelos Hammerstein y lineal	76
Tabla 7.	Modelos Wiener y Hammerstein método N4SID con SVD	76
Tabla 8.	Índices de <i>performance</i> de modelos Wiener y Hammerstein con bases ortonormales	78
Tabla 9.	Modelos Wiener y Hammerstein con bases ortonormales.....	78
Tabla 10.	Parámetros característicos de controladores predictivos PIDp y MPC para sistema masa-resorte	95
Tabla 11.	Parámetros característicos de controladores PID y MPC predictivos con modelo Wiener para planta de pH.....	101
Tabla 12.	Parámetros del control PID y WPIDp para planta de pH	106

Lista de Figuras

Figura 1.	Estructura equivalente modelo lineal con bases ortonormales	10
Figura 2.	Diagrama de bloques del sistema Laguerre	13
Figura 3.	Aproximación lineal y cúbica de una serie de 5 pares ordenados de datos	16
Figura 4.	Aproximación por tramos	17
Figura 5.	Aproximación Piecewise Linear de la función $\sin(2x)$ con: a) $n=5$ puntos de quiebre y b) $n=10$ puntos de quiebre	17
Figura 6.	Modelo Hammerstein.....	19
Figura 7.	Modelo Wiener	21
Figura 8.	Modelo Hammerstein-Wiener.....	22
Figura 9.	Modelo Hammerstein en espacio de estado	24
Figura 10.	Modelo Wiener en espacio de estado.....	27
Figura 11.	Modelo no lineal Hammerstein.....	29
Figura 12.	Modelo Hammerstein con bases ortonormales	29
Figura 13.	Modelo no lineal Wiener.....	31
Figura 14.	Modelo Wiener con bases ortonormales.....	32
Figura 15.	Proceso de neutralización de pH.....	36
Figura 16.	Esquema de la planta de neutralización de pH de la UCSB	37
Figura 17.	Esquema del proceso de neutralización de pH.....	40
Figura 18.	Representación de planta de neutralización de pH en Simulink.....	40
Figura 19.	Planta piloto de neutralización de pH Centro Universitario da FEI	41
Figura 20.	Planta piloto de School Engineering, University of Warwick.	41
Figura 21.	Modelo matemático del modelo de la planta de neutralización de pH.	43
Figura 22.	Controlador PI digital del proceso de pH.....	44
Figura 23.	Diagrama controlador híbrido lógica difusa y PID	45
Figura 24.	Curvas de titulación ácido con base fuerte.....	50
Figura 25.	Esquema planta de neutralización de pH	51
Figura 26.	Modelo del proceso de neutralización en Simulink para variaciones de solución base	53
Figura 27.	Variaciones de pH ante variaciones de caudal solución base (mL/s)	54
Figura 28.	Modelo del proceso de neutralización en Simulink para variaciones de ácido.....	54

Figura 29.	Variaciones de pH ante variaciones de caudal de solución ácida (mL/s)	55
Figura 30.	Variaciones de pH ante variaciones de concentración solución base (mol/L).....	55
Figura 31.	Variaciones de pH ante variación de concentración solución ácida (mol/L).....	56
Figura 32.	Modelo del proceso de neutralización para ácido sulfúrico diprótico	56
Figura 33.	Bloque no lineal del modelo diprótico de pH	57
Figura 34.	Comportamiento del pH en ambos modelos.....	57
Figura 35.	Diagrama del proceso de identificación	60
Figura 36.	Señal de prueba escalera	61
Figura 37.	Señal multinivel	61
Figura 38.	Ruido blanco coloreado	62
Figura 39.	Señal de entrada multisenoidal con desfase.....	62
Figura 40.	Respuesta ante variaciones escalón de la variable manipulable (caudal de la solución base).....	66
Figura 41.	Ampliación respuesta variación de pH: a) pH menor de 5, b) $10 < \text{pH} < 11$	66
Figura 42.	Simulador para entrada variaciones caudal base	67
Figura 43.	Característica estática de pH.....	67
Figura 44.	Rango reducido de variación caudal solución base	68
Figura 45.	Curva de pH para rango reducido de variación de caudal base alrededor del valor neutro.....	68
Figura 46.	Variación porcentual del pH ante variaciones de caudal base a) Rango $2.5 \leq \text{pH} \leq 11$, b) Rango $4.5 \leq \text{pH} \leq 9.5$	69
Figura 47.	Respuesta ante variación en caudal de la solución base	70
Figura 48.	Señal de entrada para prueba de identificación	71
Figura 49.	Datos entrada/salida.....	71
Figura 50.	Respuesta a datos estimación del modelo lineal.....	72
Figura 51.	Respuesta a datos de validación del modelo lineal.....	72
Figura 52.	Respuesta modelo Wiener con datos de validación.....	73
Figura 53.	Respuesta a escalón del modelo Wiener y proceso	74
Figura 54.	Característica del bloque no lineal del modelo Wiener : a) Inversa del bloque no lineal $N^{-1}(\cdot)$, b) Bloque no lineal $N(\cdot)$	74
Figura 55.	Respuesta modelo Hammerstein a datos de validación.....	75
Figura 56.	Características no lineal del modelo Hamerstein.....	75
Figura 57..	Respuesta del modelo Hammerstein y del proceso	76
Figura 58.	Respuesta modelo Wiener en bases ortonormales.....	77
Figura 59.	Respuesta modelo Hammerstein en bases ortonormales de Segundo orden	78

Figura 60.	Característica no lineal: a) modelo Wiener, b) Modelo Hammerstein	79
Figura 61.	Comparación modelos con datos de estimación	80
Figura 62.	Comparación modelos con datos de validación	81
Figura 63.	Respuesta a escalón para una variación de 0.1 mL/s del caudal base.....	81
Figura 64.	Respuesta del modelo m_4 con variación de 0.1 mL/s en el caudal base	82
Figura 65.	Respuesta del modelo m_3 con a) datos de estimación, b) datos de validación.....	82
Figura 66.	Estrategia de control predictivo basado en modelo	87
Figura 67.	Control MPC con espacio de estado aumentado.....	92
Figura 68.	Control PID predictivo.....	95
Figura 69.	a) Respuesta MPC, b) Respuesta PID predictivo.....	96
Figura 70.	a) Esquema control con modelo Wiener del proceso, b) Modelo Wiener.....	97
Figura 71.	Diagrama del control WPID predictivo de la planta de pH	97
Figura 72.	Salida control WPID predictivo para modelo SVD ante variaciones del horizonte de predicción.	98
Figura 73.	Respuesta WMPC ante variaciones de N_p	99
Figura 74.	Respuesta de control WPIDp y WMPC para horizonte de predicción $N_p = 10$	99
Figura 75.	Modelo control Wiener MPC para planta de neutralización de pH.....	100
Figura 76.	Comparación respuestas a variaciones de λ : a) Control WPIDp, b) Control WMPC.....	100
Figura 77.	Energía variable manipulable para $\lambda=100$: a) Control WPIDp, b) Control WMPC	101
Figura 78.	Esquema en Simulink del control MPC	102
Figura 79.	Comparación respuesta control WPIDp y MPC a grandes variaciones respecto al valor nominal del caudal de solución base	103
Figura 80.	Respuesta WPIDp y MPC pequeñas variaciones respecto al valor nominal del caudal de solución base con: a) $\lambda=100$, b) $\lambda=10$	103
Figura 81.	Respuesta a escalón a lazo abierto del proceso.....	104
Figura 82.	Comparación respuesta a escalón del proceso y modelo	104
Figura 83.	Respuesta del control PI ante variaciones del caudal base fuera del rango lineal.....	105
Figura 84.	Control PI ante pequeñas variaciones en el <i>setpoint</i>	106
Figura 85.	Control PID en Simulink	106
Figura 86.	Comparación control WPIDp vs PI con grandes variaciones del <i>setpoint</i>	107
Figura 87.	Comparación control WPIDp vs PI con pequeñas variaciones del <i>setpoint</i>	107
Figura 88.	Característica de la inversa del bloque no lineal modelo Hammerstein: a) Aproximación pchip, b) Aproximación pwlinear	108

Figura 89.	Implementación del control Hammerstein PID predictivo o HPIDp.....	108
Figura 90.	Control HPID predictivo para variaciones pequeñas respecto al valor nominal del caudal de solución base y con perturbación en el flujo ácido.....	109
Figura 91.	Respuesta de control HPIDp y MPC con perturbación en el flujo ácido ..	109
Figura 92.	Comparación control HPIDp y control PI ante pequeñas variaciones del <i>setpoint</i>	110
Figura 93.	Comparación HPIDp con ajuste en λ y PI.....	110

Introducción

El desarrollo del control avanzado tiene por objetivo afrontar problemas complejos, como es el caso de procesos con no linealidades. Así también, incorpora al algoritmo de control de las variables de un proceso, las exigencias de gestión de planta tales como ahorro de energía, manejo de restricciones. Una de las herramientas de control avanzado es el control predictivo basado en modelos Clarke et al., (1997), Ipanaque (2005), Ipanaque et al., (2011), Hernandez et al., (2014) que para sistemas lineales está ampliamente desarrollado, pero para el caso de procesos no lineales aún hay mucho trabajo por desarrollar. En ese sentido ya existen algunos trabajos a nivel de investigación y algunos ya son aplicados a plantas reales tales como: los controladores adaptivos, predictivos, redes neuronales Patric et al. (2010). Sin embargo, aún no hay metodologías ampliamente aceptadas que resulten eficientes para sistemas no lineales y que a la vez sean relativamente simples de aplicar. Por otro lado, se debe tener en cuenta que los controladores que son normalmente usados en la industria tales como los PID, García et al. (2012) por ser económicos con características de sintonización conocidas y actuación rápida, presentan limitaciones, sobre todo si se debe trabajar en un amplio rango de operación pues estos normalmente son sintonizados alrededor de un punto de operación de la planta y están basados en sintonización empírica. Además, tienen dificultades en procesos con fuertes retardos, respuesta inversa, interacciones, etc., presentes especialmente en la industria química, petroquímica y biológica. El objetivo de la tesis es la de encontrar modelos no lineales usando la estructura de modelos orientados a bloques y su aplicación en una estructura de control PID - con características predictivas- Kathebi (2001), Kokate (2011), aplicado a un proceso de neutralización de pH. Para ello la tesis se ha dividido en cinco capítulos, los cuales se resumen a continuación:

En el primer capítulo, se da una visión global de las técnicas de identificación haciendo especial énfasis en los métodos de identificación de modelos orientados a bloques Wiener y Hammerstein, Gomez, et al. (2004), Kozek and Jovanovic (2003), Sung and Lee (2004). En el segundo capítulo se hace el análisis de los procesos de neutralización y se diseña un simulador del proceso de neutralización de un ácido fuerte (ácido sulfúrico) y una base fuerte (soda cáustica), comparándose el comportamiento entre un modelo simplificado y el modelo completo, este último, teniendo en cuenta la característica diprótica del ácido sulfúrico (segunda disociación en general mucho más débil que la primera).

En el tercer capítulo se aplican los métodos de identificación vistos en el capítulo 1 usando el simulador de la planta diseñado en el capítulo anterior. El cuarto capítulo

abarca el estudio de los controladores PID predictivos en especial se describe el método que se aplica en el proyecto partiendo del algoritmo MPC basado en espacio de estados. Se analiza la estructura PID predictivo usando un modelo Wiener, denominada WPIDp (Wiener PID predictivo) y el modelo Hammerstein, denominado HPIDp (Hammerstein PID predictivo). Para el desarrollo del proyecto se ha elaborado una plataforma de simulación de una planta de pH considerando el ácido sulfúrico y la soda cáustica como sustancias a mezclarse en el proceso de neutralización, sobre el cual se aplican los algoritmos de identificación y de control. Para las pruebas se ha usado el modelo identificado del proceso diseñado en el simulador según la estructura Wiener y Hammerstein y se evalúan las prestaciones de un algoritmo PID con características predictivas comparando las prestaciones de este algoritmo con un controlador PID y un controlador MPC. En el quinto capítulo se dan algunas recomendaciones de trabajos futuros que se pueden realizar teniendo en cuenta algunas hipótesis que se usaron para simplificar los métodos evaluados en el proyecto pero que en base al análisis de los resultados obtenidos motivan a un estudio más exhaustivo de los algoritmos de identificación y control PID predictivos que permitan un ajuste en línea y aplicados a procesos multivariables.

Cabe mencionar que en el transcurso de la tesis se ha publicado un artículo SCOPUS: “*Wiener Predictive control for a pH Neutralization plant*” en el 2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON) (Anexo 1). Además, se tuvieron dos estancias de investigación una en Alemania y otra en la Universidad Católica de Chile. Estas experiencias aportaron mayor conocimiento y experiencia en el trabajo de esta tesis.

Los resultados presentados muestran las buenas prestaciones de la metodología propuesta tanto en lo que se refieren al modelo no lineal obtenido del proceso, como en el regulador PID predictivo propuesto. La ventaja que evidencia este regulador PID predictivo es que no requiere de plataformas complejas para poder implementarlo. Por último, se ha hecho un análisis de la conveniencia de cómo y cuándo usar, un modelo Wiener y un modelo Hammerstein.

Capítulo 1

Métodos de identificación de modelos no lineales orientados a bloques

La identificación de sistemas se refiere a la técnica experimental que permite determinar modelos de sistemas por ajuste de datos experimentales a una estructura de modelo determinada usando algoritmos de optimización. Vale decir que hay una teoría ampliamente desarrollada para identificación de modelos lineales y que funcionan bien si los procesos operan muy cerca del punto de funcionamiento alrededor del cual se obtuvieron los datos experimentales para su identificación. Sin embargo, cuando se tiene un amplio rango de operación, las hipótesis lineales ya no son válidas y se hace necesaria la estimación o identificación de un modelo no lineal con una estructura que permita capturar lo mejor posible el comportamiento del proceso que se está identificando.

En algunos casos, los modelos matemáticos se pueden deducir de los fenómenos que describen el proceso aplicando las leyes físicas, químicas, mecánicas, etc. que lo rigen (llamados también modelos de primeros principios o *first principles*). De lo contrario, se debe recurrir a otras técnicas de identificación que en el caso de procesos no lineales deben basarse en estructuras de modelos no lineales de caja- negra o caja-gris.

La teoría clave para la identificación de un sistema es encontrar el mejor modelo ajustable a las entradas, salidas y disturbios del sistema real. Si se tiene un conocimiento total de las leyes y magnitudes que gobiernan el sistema y, por tanto, son conocidos todos los parámetros, el modelo se denomina de caja blanca. Si el modelo se construye con solo datos de entrada/salida y sin conocimiento de las leyes que gobiernan el proceso entonces el modelo se denomina de caja negra. En muchos casos se tiene algún conocimiento de una parte del proceso o se conocen características específicas del mismo, por lo que esta información se usa para la modelación del proceso denominándose modelo de caja gris (Giri & Bai, 2010).

Muchas dinámicas de procesos pueden representarse mejor por modelos no lineales que por modelos lineales, debido a que describen mejor su comportamiento en un amplio rango de operación mientras que los lineales sólo aproximan el proceso alrededor de un punto de operación.

Uno de los modelos no lineales muy difundidos en las últimas décadas son los llamados modelos “orientados a bloques”, que consisten en la interconexión de un subsistema dinámico lineal invariante (LTI) y un bloque de no linealidad estática (sin memoria). Dentro de esta clase de modelos los más comunes son los modelos

Hammerstein y Wiener, los cuales han sido usados exitosamente para representar sistemas no lineales en varias aplicaciones prácticas.

1.1. Estado del arte de las técnicas de identificación lineal

Para la identificación de los modelos no lineales con algunas de las estructuras Wiener o Hammerstein, se recurre en varios casos, a técnicas de identificación lineal que se complementan con algoritmos de modelación no lineal. Esto es posible sobre todo cuando dichas técnicas contemplan la identificación de manera desacoplada de los bloques lineal y no lineal, como es el caso de los modelos antes mencionados. El orden de la identificación de cada uno de dichos bloques dependerá de la herramienta y/o modelo de identificación que se use.

Los subsistemas lineales pueden ser:

- a. **Paramétricos:** Poseen una estructura y están descritos por un número finito de parámetros. Dentro de estos modelos se encuentran: la representación en espacio de estado, modelos de función de transferencia, modelos FIR (*Finite Impulse Response*), FSR (*Finite Step Response*), entre otros.
- b. **No paramétricos:** Requieren un número infinitos de parámetros y no tienen una estructura particular tal es el caso de la respuesta en frecuencia. Por lo general, se describe el proceso de manera gráfica, a partir de la cual se deducen también algunas características del proceso

Los elementos no lineales pueden ser paramétricos o no y además con memoria (la salida en un instante depende de la entrada actual y valores pasados) o sin memoria (la salida depende sólo de la entrada actual o presente en ese instante) (Giri & Bai, 2010). Los componentes del sistema pueden ser conectados de distintas maneras: en serie, paralelo, realimentados. La estructura más usada es la disposición en serie.

En el caso de controles predictivos la selección de la estructura adecuada de un modelo no lineal para representar un sistema dinámico es un paso crucial en el diseño de un control MPC no lineal o NMPC. En esa línea se han desarrollado modelos no lineales usando una variedad de técnicas, incluyendo los modelos de primeros principios y métodos empíricos (modelos de caja negra o *black-box*), que son válidos globalmente y pueden predecir sistemas dinámicos sobre un amplio rango de operación. Sin embargo, encontrar un modelo fiable basado en primeros principios es una tarea complicada que requiere tiempo y esfuerzo, por lo que en ese sentido los modelos de caja negra o *black-box* no lineales tienen ciertas ventajas sobre los anteriores aunque para estos no resulta fácil la selección de una estructura del modelo que describa las dinámicas no lineales válidas en un amplio rango de operación. Aun así, desde un punto de vista más práctico, es una herramienta más atractiva para el desarrollo de controladores no lineales como el NMPC.

En el caso de los modelos no lineales orientados a bloques, siempre existe la necesidad de identificar al menos un bloque no lineal y un bloque lineal, por lo que en la práctica sirve tener un conocimiento *a priori* y cualitativo del tipo de no linealidad que tiene el proceso para elegir el método y estructura que mejor se ajuste al proceso.

1.1.1. Modelos de bloques lineales

Para describir un sistema dinámico lineal invariante en el tiempo (LTI: *Linear and Time Invariant*), causal y estable, se deben cumplir las siguientes condiciones:

- a. Lineal:** Se puede aplicar el principio de superposición. Una característica de los sistemas lineales es que ante una entrada nula la salida también es nula.
- b. Invariante:** Sus parámetros no cambian con el tiempo
- c. Causal:** Depende únicamente de las entradas presentes y pasadas y de las salidas pasadas.
- d. Estable:** Toda entrada acotada provoca una salida acotada. Se denomina también sistema BIBO (*Bounded Input Bounded Output*).

La dinámica de un sistema LTI discreto se puede representar por ecuaciones diferencias con coeficientes constantes o por medio de la respuesta impulsiva $G(q)$.

En un sistema con una entrada dada $u(t)$, la salida $y(t)$ puede expresarse como (Neyra, 2009):

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t-k) \quad t = 0,1,2, \dots \quad (1.1)$$

La función de transferencia $G(q)$ del sistema descrito por (1.1) viene dada por:

$$G(q) = \sum_{k=1}^{\infty} g(k)q^{-k} \quad (1.2)$$

y la salida del sistema se define como:

$$y(t) = G(q)u(t) \quad (1.3)$$

Si $G(q)$ es estable, entonces:

$$\sum_{k=1}^{\infty} |g(k)| < \infty \quad (1.4)$$

Aunque un sistema puede determinarse únicamente por su respuesta impulsiva, ésta no se utiliza muchas veces en la práctica, pues en general tiene una serie infinita de términos. Por el contrario se prefiere describir un sistema por su función de transferencia $G(q)$ caracterizado por un número finito de parámetros, como se describe a continuación.

1.1.1.1. Función de transferencia racional

En la función de transferencia racional definida en (1.2) los coeficientes del numerador y denominador son los parámetros a determinar. Una manera formal y común de expresarla es como una relación de polinomios tal y como se describe a continuación.

$$G(q) = \frac{b_0 q^{-1} + \dots + b_{n_b} q^{-n_b}}{1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}} = \frac{B(q)}{A(q)} \quad (1.5)$$

A partir de la función de transferencia se puede escribir la ecuación diferencia:

$$y(t) + a_1 y(t-1) + \dots + a_{n_a} y(t-n_a) = b_1 u(t-1) + \dots + b_{n_b} u(t-n_b) \quad (1.6)$$

La estructura descrita en (1.5) es muy simplificada, en una situación real siempre hay presencia de ruido. El modelo del ruido, por lo general, se define en términos del llamado ruido blanco que es una función en la que los valores en el tiempo no tienen relación alguna. En términos estadísticos contiene todas las frecuencias y por ello tiene una densidad espectral constante. Por lo general el ruido se representa por medio de un filtro del ruido blanco a lo que se denomina ruido coloreado.

Dependiendo de cómo se modela el ruido se puede describir el modelo de forma paramétrica con estructuras definidas en la Tabla 1, en las que se asume el ruido aditivo. En dichas expresiones $e(t)$ es el ruido blanco y $A(q)$, $B(q)$, $C(q)$, $F(q)$ y $D(q)$ son polinomios en q^{-1} , como se describe a continuación:

$$A(q^{-1})y(k) = \frac{B(q^{-1})}{F(q^{-1})}u(k) + \frac{C(q^{-1})}{D(q^{-1})}e(k) \quad (1.7)$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}$$

$$B(q^{-1}) = b_1 q^{-1} + \dots + b_{n_b} q^{-n_b}$$

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}$$

$$D(q^{-1}) = 1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}$$

$$F(q^{-1}) = 1 + f_1 q^{-1} + \dots + f_{n_f} q^{-n_f}$$

Tabla 1. Modelos paramétricos de identificación

Modelo		Estructura
AR	Auto-Regressive	$y(k) = \frac{1}{A(q^{-1})} e(k)$
MA	Moving Average	$y(k) = C(q^{-1})e(k)$
ARX	Auto- Regressive model with eXogeneous variables	$y(k) = \frac{B(q^{-1})}{A(q^{-1})}u(k) + \frac{1}{A(q^{-1})}e(k)$
ARMA	Auto-Regressive Moving Average	$y(k) = \frac{C(q^{-1})}{A(q^{-1})}e(k)$
ARMAX	Auto-Regressive Moving Average with eXogeneous variables	$y(k) = \frac{B(q^{-1})}{A(q^{-1})}u(k) + \frac{C(q^{-1})}{A(q^{-1})}e(k)$
OE	Output Error	$y(k) = \frac{B(q^{-1})}{A(q^{-1})}u(k) + e(k)$
BJ	Box-Jenkins	$y(k) = \frac{B(q^{-1})}{F(q^{-1})}u(k) + \frac{C(q^{-1})}{D(q^{-1})}e(k)$

La entrada $u(k)$ de (1.7) también puede contener un retardo n_k , tal y como se considera en MATLAB, de modo que en su lugar se escribe $u(k - n_k)$.

1.1.1.2. Respuesta impulsiva FIR (Finite Impulse Response)

En este caso la salida es una función lineal de los parámetros en el que $A(q^{-1}) = 1$, esto es:

$$y(t) = h_1 u(t - 1) + \dots + h_{n_b} u(t - n_b) \quad (1.8)$$

donde n_b representa la longitud de datos en el tiempo de la respuesta impulsiva para un sistema estable. Los parámetros se pueden determinar aplicando el criterio de minimización del error cuadrático usando regresión lineal.

La ecuación (1.8) se puede escribir de la forma:

$$y(t) = \sum_{i=1}^{n_b} h_i u(t - i) \quad (1.9)$$

Conocido como modelo de convolución, donde h_i es la salida muestreada cuando el proceso se excita con un impulso unitario durante el período de muestreo y los datos se truncan en n_b valores. Para el modelo discretizado resulta la expresión:

$$y(k) = \sum_{i=1}^{n_b} h_i u(k - i) \quad (1.10)$$

Para procesos multivariados, cada salida refleja el efecto de las M entradas en la siguiente forma:

$$y(k) = \sum_{j=1}^M \sum_{i=1}^{n_b} h_i^{j_i} u^{ij}(k - i) \quad (1.11)$$

1.1.1.3. Respuesta a escalón FSR (Finite Step Response)

Es similar al modelo anterior, excepto que la señal aplicada a la entrada es un escalón.

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t - i) \quad (1.12)$$

Para sistemas estables, la respuesta truncada (discretizando el modelo) está dada por:

$$y(k) = y_0 + \sum_{i=1}^N g_i \Delta u(k-i) \quad (1.13)$$

donde g_i son los valores de la salida muestreada para la entrada escalón y $\Delta u(k) = u(k) - u(k-1)$. En la ecuación (1.13), el valor y_0 puede considerarse igual a cero sin perder generalidad. Este modelo tiene las mismas ventajas y desventajas del anterior.

1.1.1.4. Bases ortonormales

Los modelos paramétricos se pueden describir con un número infinito de parámetros, por ello, su uso está restringido sólo para identificación y control. En algunos métodos de identificación, bajo ciertas condiciones se logra reducir el número de parámetros a determinar, una de ellas por ejemplo consiste en usar bases ortonormales.

La condición de ortonormalidad de estos modelos permite reducir el número de parámetros dado que estos se pueden estimar independientemente uno del otro. Del conjunto de bases ortonormales los más usados son los modelos basados en filtros de Laguerre (usado para el caso de polos reales) o de Kautz (para polos complejos).

La función de transferencia de un modelo lineal con bases generalizadas ortonormales, (Gómez & Baeyens, 2004 a) se puede expresar de la siguiente manera:

$$H(z) = \sum_{j=0}^{p-1} b_j B_j(z) \quad (1.14)$$

siendo p el número de polos y b_j los coeficientes de la base ortonormal. Se tiene además:

$$B_0(z) = \frac{\sqrt{1 - |\xi_0|^2}}{z - \xi_0} \quad (1.15)$$

$$B_j(z) = \frac{\sqrt{1 - |\xi_j|^2}}{z - \xi_j} \prod_{i=0}^{j-1} \left(\frac{1 - \bar{\xi}_i z}{z - \xi_i} \right) \quad \text{para } j \geq 1 \quad (1.16)$$

siendo ξ_i el polo i y $\bar{\xi}_i$ su complejo conjugado.

La salida discreta puede expresarse como:

$$y(k) = \sum_{j=0}^{p-1} b_j B_j(z) x(k) + v(k) \quad (1.17)$$

y debe cumplirse:

$$\sum_{k=0}^{\infty} B_i(k) B_j(k) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases} \quad (1.18)$$

para que la base generada por las funciones $B_j(q)$ sea ortonormal.

En Gómez et al (2004) se formula un modelo lineal usando bases ortonormales racionales con polos fijos (OBFP) que permite incorporar polos en ubicaciones arbitrarias, de manera que la estima se realiza escogiendo la mejor ubicación de los polos (por lo general se parte de los polos hallados de la representación lineal). Los coeficientes de las bases son estimadas usando el método de descomposición en valores singulares. Es de nuestro interés la representación del modelo encontrado en bases ortonormales a la estructura en espacio de estado, por lo que siguiendo lo planteado por (Gómez y Baeyens, 2005b) se representan los filtros que forman dicha base ortonormal, de la siguiente forma:

$$Q_0(z) \triangleq \frac{1}{z - \xi_0} \quad (1.19)$$

$$Q_j(z) \triangleq \frac{1 - \xi_{j-1} z}{z - \xi_j}, \quad j \geq 1 \quad (1.20)$$

De esa manera se encuentra una estructura equivalente (Figura 1) a partir de la cual se puede determinar una realización mínima en espacio de estado para un sistema SIMO con m salidas, propuesta en (Gómez & Baeyens, 2004 a).

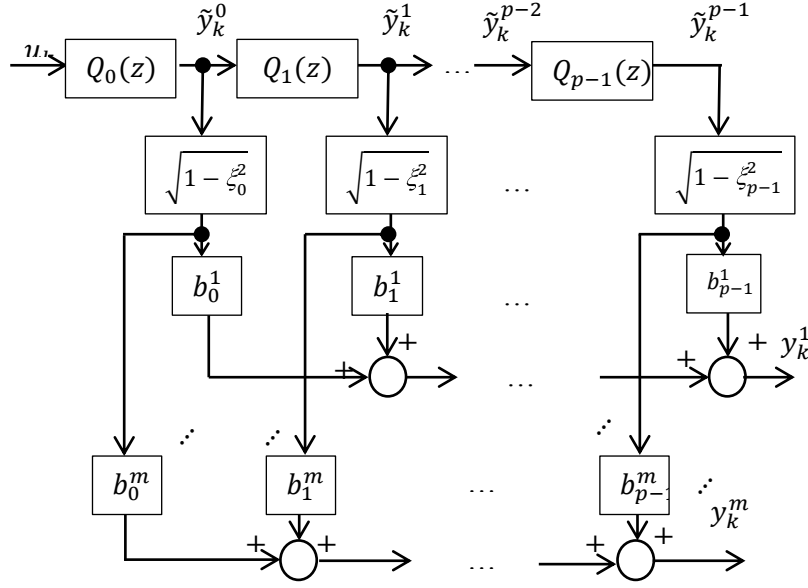


Figura 1. Estructura equivalente modelo lineal con bases ortonormales

A partir de esta estructura y considerando las definiciones dadas en (1.19) y (1.20) se pueden escribir las ecuaciones en espacio de estado:

$$x_{k+1}^0 = \xi_0 x_k^0 + u_k \quad (1.21)$$

$$\tilde{y}_k^0 = x_k^0 \quad (1.22)$$

que corresponde al primer filtro $Q_0(z)$, y

$$x_{k+1}^j = \xi_j x_k^j + (1 - \xi_j \xi_{j-1}) u_k^j \quad (1.23)$$

$$\tilde{y}_k^j = x_k^j - \xi_{j-1} u_k^j \quad (1.24)$$

que corresponde a los filtros Q_j . Las matrices asociadas a cada representación se escriben como:

$$(A^0, B^0, C^0, D^0) = (\xi_0, 1, 1, 0) \quad (1.25)$$

$$(A^j, B^j, C^j, D^j) = (\xi_j, (1 - \xi_j \xi_{j-1}), 1, -\xi_{j-1}) \quad (1.26)$$

Junto a dichas matrices se pueden definir:

$$x_k \triangleq [x_k^0, x_k^1, \dots, x_k^{p-1}]^T \quad (1.27)$$

$$\tilde{y}_k \triangleq [\tilde{y}_k^0, \tilde{y}_k^1, \dots, \tilde{y}_k^{p-1}]^T \quad (1.28)$$

$$y_k \triangleq [y_k^0, y_k^1, \dots, y_k^{p-1}]^T \quad (1.29)$$

$$\mathbf{b} \triangleq [b_0, b_1, \dots, b_{p-1}] \quad (1.30)$$

Por lo que ya se puede escribir en forma matricial la representación en espacio de estado del sistema completo, ordenando las ecuaciones descritas anteriormente:

$$A \triangleq \begin{bmatrix} \sqrt{1 - \xi_0^2} & 0 & \dots & 0 \\ 0 & \sqrt{1 - \xi_1^2} & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \dots & \sqrt{1 - \xi_{p-1}^2} \end{bmatrix} \quad (1.31)$$

$$A = \begin{bmatrix} A^0 & 0 & \dots & 0 \\ B^1 C^0 & A^1 & \dots & 0 \\ B^2 D^1 C^0 & B^2 C^1 & \dots & 0 \\ B^3 D^2 D^1 C^0 & B^3 D^2 C^1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ B^{p-1} D^{p-2} \dots D^1 C^0 & B^{p-1} D^{p-2} \dots D^2 C^1 & \dots & A^{p-1} \end{bmatrix} \quad (1.32)$$

$$\tilde{C} = \begin{bmatrix} C^0 & 0 & \dots & 0 \\ D^1 C^0 & C^1 & \dots & 0 \\ D^2 D^1 C^0 & D^2 C^1 & \dots & 0 \\ D^3 D^2 D^1 C^0 & D^3 D^2 C^1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ D^{p-1} D^{p-2} \dots D^1 C^0 & D^{p-1} D^{p-2} \dots D^2 C^1 & \dots & C^{p-1} \end{bmatrix} \quad (1.33)$$

Por último, se tiene:

$$B = [1 \ 0 \ 0 \ \dots \ 0]^T \quad (1.34)$$

$$C = b A \tilde{C} \quad (1.35)$$

y $D = 0$, por lo que la realización mínima en espacio de estado se puede expresar por:

$$x_{k+1} = A x_k + B u_k \quad (1.36)$$

$$y_k = C x_k + D u_k \quad (1.37)$$

1.1.1.5. Función Laguerre

El modelo SISO de un sistema lineal, usando un filtro Laguerre (Mahmoodi, Poshtan, Jahed-Motlagh, & Montazeri, 2009), viene dada por:

$$\hat{y}(z) = \left(\sum_{i=1}^N c_i L_i(z) \right) u(z) \quad (1.38)$$

$$L_i(z) = \sqrt{(1-a^2)T} \frac{(1-az)^{i-1}}{(z-a)^i} \quad (1.39)$$

donde $L_i(z)$ es un filtro de orden i de Laguerre, N el número de filtros de Laguerre usados para el desarrollo del modelo, a ($-1 < a < 1$) el parámetro del filtro Laguerre que normalmente es un polo real, T el periodo de muestreo, $\hat{y}(z)$ la salida del modelo y $u(z)$ es la entrada manipulada. Como se observa de la definición, se puede decir que el filtro de Laguerre es un caso particular de la bases ortonormales definidas en (1.14).

Estos filtros son utilizados como bases de la modelación de sistemas no lineales, combinándolos con otros kernels, por ejemplo, se puede usar un kernel de Volterra o un polinomio para la componente no lineal de un modelo. En este caso tal y como se desarrolla en (Medina, 2011), (Gruber & Bordons, 2007), (Sahar, Krishman, Rao, & Patwardhan, 2004), se puede definir el vector de estado:

$$L(k) = [l_1(k), l_2(k), \dots, l_N(k)]^T \quad (1.40)$$

donde $l_i(k)$ representa la salida del filtro Laguerre en el instante de muestreo k a la entrada $u(k)$. La red de filtros Laguerre se puede expresar en una realización de espacio de estado discreto:

$$L(k+1) = \Phi(a)L(k) + \Gamma(a)u(k) \quad (1.41)$$

donde $u(k)$ es la entrada del sistema, $\Phi(a)$ es una matriz triangular inferior $N \times N$ definida por:

$$\Phi(a) = \begin{bmatrix} a & 0 & 0 & 0 & 0 \\ (1-a^2) & a & 0 & 0 & 0 \\ -a(1-a^2) & (1-a^2) & a & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ (-1)^N a^{N-2} (1-a^2) & (-1)^{N-1} a^{N-3} (1-a^2) & \dots & \dots & a \end{bmatrix} \quad (1.42)$$

y $\Gamma(a)$ es un vector N dimensional:

$$\Gamma(a) = \left[\sqrt{(1-a^2)T}, -a\sqrt{(1-a^2)T}, \dots, (-a)^{N-1}\sqrt{(1-a^2)T} \right]^T \quad (1.43)$$

Para el modelo lineal dado por la ecuación (1.41), la salida puede expresarse como una suma ponderada de los estados según:

$$\hat{y}(k) = C^T L(k) \quad (1.44)$$

donde los elementos C son los coeficientes del filtro de Laguerre, esto es

$$C = [c_1, c_2, \dots, c_N]^T \quad (1.45)$$

En la Figura 2 se muestra el diagrama de bloques del sistema Laguerre para un sistema SISO (Medina, 2011), que es similar a la estructura representada en la Figura 1.

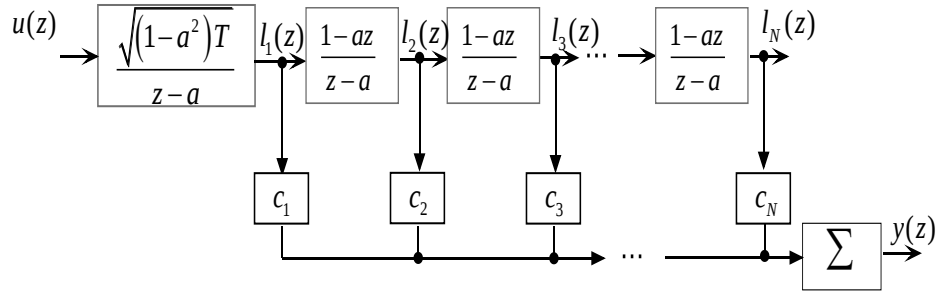


Figura 2. Diagrama de bloques del sistema Laguerre

Para el desarrollo de un modelo no lineal SISO con la estructura Wiener (primer bloque lineal y segundo bloque no lineal), se puede construir un mapeo estado- salida de tal manera que la salida del modelo queda representada como:

$$\hat{y}(k) = \Psi[x(k)] \quad (1.46)$$

donde $\Psi(\cdot)$ es una función no lineal sin memoria y $x(k)$ sería la variable de entrada del bloque de filtros Laguerre en el caso del modelo Wiener.

El paso clave en el desarrollo del modelo Laguerre es estimar el parámetro a y seleccionar el número de redes Laguerre N . Los datos de la respuesta escalón pueden usarse para generar una estima inicial del parámetro del filtro a . Si τ es la constante de tiempo, el polo discreto del filtro de Laguerre puede encontrarse como:

$$a = e^{-T/\tau} \quad (1.47)$$

En (Mahmoodi, Montazeri, Poshtan, & Jahed-Motlagh, 2007), (Mahmoodi, Poshtan, Jahed-Motlagh, & Montazeri, 2009) este parámetro se obtiene a partir del polo preseleccionado real en lugar de obtenerlo por optimización ya que mejora la velocidad y precisión del algoritmo de estima.

No obstante esta selección puede llevar a un número de filtros Laguerre muy elevado o a dinámicas de segundo orden no amortiguadas.

1.1.1.6. Representación en espacio de estado

El modelo en espacio de estado tiene la siguiente representación en tiempo continuo:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (1.48)$$

$$y(t) = Cx(t) + Du(t) \quad (1.49)$$

donde $x(t)$ son las variables de estado, A la matriz del sistema, B la matriz de entrada y C la matriz de salida, $y(t)$ la salida del proceso, $u(t)$ la entrada. Dicho modelo en el dominio discreto, sería:

$$x(k+1) = Ax(k) + Bu(k) \quad (1.50)$$

$$y(k) = Cx(k) + Du(k) \quad (1.51)$$

Este modelo tiene como ventaja la posibilidad de emplearse para procesos multivariados de una manera más directa pero tiene la desventaja de que en algunas ocasiones las variables de estado carecen de significado físico. El modelo en tiempo discreto es el que usaremos en el algoritmo de control predictivo, por sus características recursivas.

1.1.2. Modelos de bloques no lineales

De la literatura analizada se puede evidenciar que no en pocos casos, se usan modelos Wiener y Hammerstein (Gómez & Baeyens 2004 a), (Gómez & Baeyens 2004 b), (Bordons, 2000), (Henson & Seborg, 1994), (Mahmoodi et al, 2009), (Purizaga, 2007), para procesos de neutralización de pH, que es la aplicación que se estudia en este trabajo. El proceso por sus características de no linealidad facilita la validación de las distintas técnicas de identificación estudiadas para los modelos orientados a bloques. Lo que queda claro es que estos modelos tienen amplia aplicación en procesos químicos, biológicos, procesamiento de señales y control.

Dependiendo de la ubicación de cada uno de los elementos: lineal y no lineal se encuentran estructuras o modelos denominados: Wiener o Hammerstein, como también una combinación de los mismos como son los modelos Hammerstein-Wiener, Wiener-Hammerstein.

Para estos modelos se recurren a distintas alternativas, desde una simple aproximación polinómica de la no linealidad hasta la aplicación de técnicas de aproximación más complejas, lo cual dependerá también del tipo de no linealidad

que presente el proceso. Los modelos que revisaremos en este acápite son los que se suelen usar con cierta simplicidad en el tiempo. Hay otras alternativas que permiten la identificación de este bloque usando técnicas en la frecuencia, pero no son objeto de este trabajo.

Para la mayoría de modelos Wiener o Hammerstein, ha resultado especialmente útil la aproximación del bloque no lineal como una función polinómica, hipótesis que se asume si no se tiene conocimiento del tipo de no linealidad presente. Sin embargo, hay otras alternativas que se están usando, en especial, en procesos químicos.

Hay algunos factores importantes a tener en cuenta para su elección:

- El tipo de no linealidad que presente el sistema, es decir, si se trata de una no linealidad continua o una discontinua.
- Uso del modelo con propósitos de control, el cual requiere de condiciones matemáticas como la existencia de la inversa de la no linealidad.

El punto de partida es que se puede usar una función de expansión, con funciones base o parámetros. La estructura principal es como sigue (Hagenblad, 1999):

$$y = \sum_{i=1}^n f_i g_i(x) \quad (1.52)$$

Se debe distinguir entre dos tipos:

- a. Funciones en las que se fijan algunas funciones base predeterminadas g_i , donde los coeficientes f_i son los únicos parámetros a determinar.
- b. Funciones en las que $g_i(x) = g_i(x, \eta)$. Como se observa contiene además el parámetro η que puede afectar tanto la forma como la posición de las funciones base.

El primer caso es una particularidad del segundo en donde la salida es una función lineal de sus parámetros. Para estimar los parámetros se pueden usar métodos de regresión lineal con la ventaja que es una única estima y el método es rápido.

1.1.2.1. Funciones no lineales

Algunas alternativas para la representación de las funciones no lineales se describen a continuación (Purizaga, 2007), (Hagenblad, 1999)

- a) ***Series de potencia o representación polinomial:*** Son útiles solamente para sistemas cuya no linealidad es pequeña o cuando no se conoce el tipo de no linealidad. La función base tiene la forma:

$$g_i(x) = x^i, \quad i = 1, 2, \dots \quad (1.53)$$

La desventaja de estas series, es que la aproximación en zonas lejanas al punto específico resulta pobre y por ello se recomienda su uso entorno a un punto específico.

- b) **Funciones *splines*:** Son funciones cuya expresión matemática varía por intervalos. Un *spline* es una curva diferenciable definida en porciones mediante polinomios que modelan no linealidades discontinuas, como son la precarga, saturación, zona muerta, ganancia variable, entre otras. Por ejemplo, un *spline* cúbico está formado por polinomios cuadráticos, los cuales cambian en los llamados puntos de quiebre.
- ***Spline* grado 1:** Son funciones polinomiales de grado 1, es decir funciones $f(x) = ax + b$ que se encargan de unir cada par de coordenadas mediante una recta.

Una función *spline* de grado 1 que interpole los datos consiste en unir cada uno de los puntos (pares coordenados) mediante segmentos de recta, como se ilustra en la Figura 3, en la que se grafican las aproximaciones lineal y cúbica de un conjunto de puntos usando en este último caso la instrucción *spline* de MATLAB.

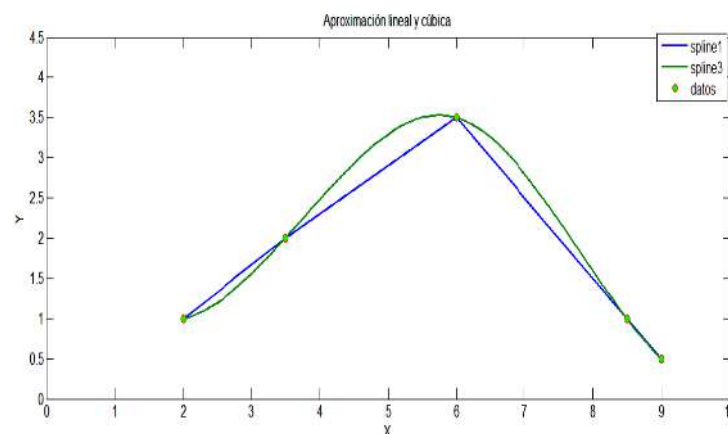


Figura 3. Aproximación lineal y cúbica de una serie de 5 pares ordenados de datos

- ***Spline* de segundo orden:** Los polinomios $P(x)$ a través de los que se construye el *spline* tiene la forma $P(x) = ax^2 + bx + c$.
- ***Spline* cúbico o de tercer orden:** Cada polinomio $P(x)$ a través del que se construyen los *splines* en cada conjunto de puntos tienen la forma $P(x) = ax^3 + bx^2 + cx + d$.
- **Función a tramos o *piecewise*:** Se emplean para aproximar funciones genéricas, constituyen una herramienta poderosa en la representación de no linealidades. Estas funciones permiten un tratamiento sistemático y exacto de funciones de aproximación (Giri & Bai, 2010). Un ejemplo de esta aproximación se muestra en la Figura 4.

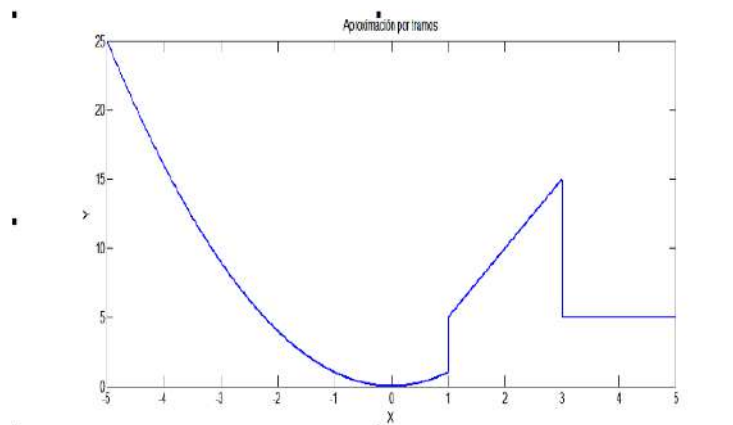


Figura 4. Aproximación por tramos
Fuente: Elaboración propia

Un caso particular de estas funciones son las conocidas como *piecewise linear* o definidas a tramos lineales. La particularidad es que estas funciones unen los llamados puntos de quiebre con tramos de rectas. Son especialmente útiles para no linealidades continuas y que permiten su aproximación de manera lineal. En el caso de la aplicación a un proceso de neutralización de pH suelen usarse con pocos puntos de quiebre y así reducir el número de parámetros a determinar brindando una muy buena aproximación.

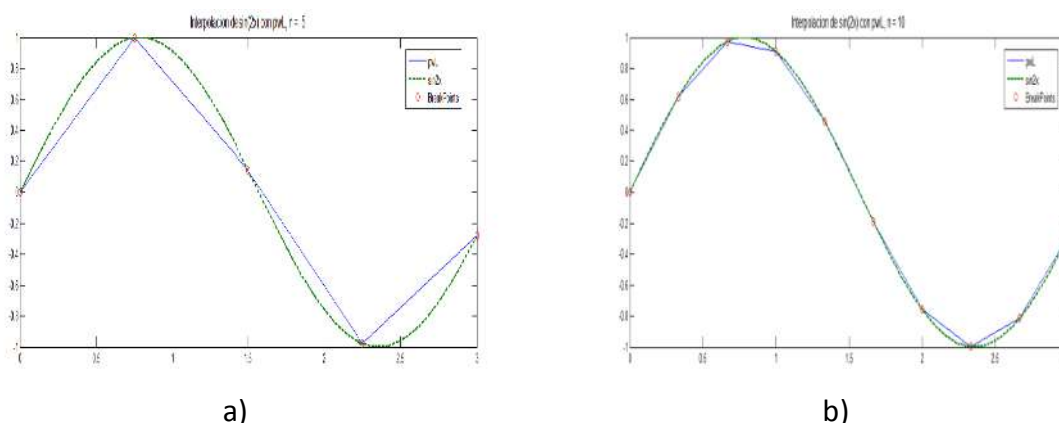


Figura 5. Aproximación Piecewise Linear de la función $\text{sen}(2x)$ con: a) $n=5$ puntos de quiebre y b) $n=10$ puntos de quiebre
Fuente: Elaboración propia

Si la función a aproximar es $Y = f(x)$ entonces para un intervalo de x comprendido entre dos puntos de quiebre (x_i, y_i) y (x_{i-1}, y_{i-1}) aproximando la función en dicho intervalo por una recta se puede escribir:

$$Y = y_{i-1} + \frac{y_i - y_{i-1}}{x_i - x_{i-1}}(x - x_{i-1}) \quad (1.54)$$

1.1.2.2. Modelos Volterra

Un sistema ligeramente no lineal estático con señal de entrada $u(t)$ y salida $y(t)$ puede describirse por medio de las series de Taylor:

$$y(t) = \sum_{i=0}^n c_i u^i(t) = c_0 + c_1 u(t) + c_2 u^2(t) + \dots \quad (1.55)$$

Los modelos en series de Volterra fueron desarrollados por primera vez en 1930, demostrándose que cualquier función continua no lineal puede describirse por las series de funciones ponderadas de Volterra, que en cierta manera son consideradas como una generalización del modelo de respuesta a impulso, pero para descripción de sistemas no lineales.

La descripción en tiempo discreto de los sistemas por medio de las series de Volterra, se puede expresar como:

$$\begin{aligned} y(k) = & h_0 + \sum_{\mu_1=0}^k h_1(\mu_1) u(k - \mu_1) + \sum_{\mu_1=0}^k \sum_{\mu_2=0}^k h_2(\mu_1, \mu_2) u(k - \mu_1) u(k - \mu_2) + \\ & \dots = \sum_{i=0}^n \dots \sum_{\mu_i=0}^k h_i(\mu_1, \dots, \mu_i) \prod_{j=1}^i u(k - \mu_j) \end{aligned} \quad (1.56)$$

Estas series son caracterizadas por sus Kernels de Volterra, es decir las funciones $h_n(\mu_1, \dots, \mu_n)$, $n = 0, 1, 2, \dots$ en el caso discreto.

Una serie de Volterra, a pesar de su amplia aplicabilidad en la representación de sistemas no lineales tiene algunas limitaciones, entre ellas la necesidad de un gran número de parámetros para ajustar un sistema no lineal sencillo, lo que conlleva a un gran esfuerzo computacional (Mahmoodi, Poshtan, Jahed-Motlagh, & Montazeri, 2009). Esto se debe a que las series de Volterra mapean las entradas pasadas para una salida actual $y(k)$.

Según (Gruber & Bordons, 2007) por lo general la expansión de Volterra se trunca, usando polinomios ortogonales generalizados de Laguerre o de Kautz para caracterizar los kernels, aunque se pueden usar otras funciones kernels. Si se hace, por ejemplo, una expansión con una serie de segundo orden y horizonte de truncamiento N , se puede escribir (1.56) como:

$$y(k) = \sum_{i=0}^N h_1(i) u(k - i) + \sum_{i=0}^N \sum_{j=0}^N h_2(i, j) u(k - i) u(k - j) \quad (1.57)$$

Si h_i se proyecta sobre un espacio formado por una base de polinomios de Laguerre ortogonales y linealmente independientes, se puede reducir el número de parámetros a determinar y el modelo se puede simplificar un poco más.

1.1.2.3. Modelos NARX y NARMAX

Los algoritmos de identificación paramétrica: NARX (*Nonlinear Auto-Regressive model with eXogeneous variables*) y NARMAX (*Nonlinear Auto-RegressiveMoving Average model with eXogeneous variables*) son una extensión de sus equivalentes lineales ARX y ARMAX cuyas funciones de transferencia fueron descritas en la Tabla 1. Estos modelos son ampliamente utilizados por ser capaces de describir una gama de sistemas no lineales (Da Silva, 2009).

El *modelo* NARX se puede describir como:

$$y(k) = \mathcal{N}[y(k-1), \dots, y(k-n_a), u(k-1), \dots, u(k-n_b)] \quad (1.58)$$

y el modelo NARMAX

$$y(k) = \mathcal{N}[y(k-1), \dots, y(k-n_a), u(k-1), \dots, u(k-n_b), e(k), e(k-1), \dots, e(k-n_c)] \quad (1.59)$$

En las dos ecuaciones n_a , n_b y n_c representan los retardos de la salida, entrada y del ruido respectivamente y $\mathcal{N}$ es una función no lineal con grado de no linealidad l .

Los *modelos* NARX y NARMAX pueden representarse por medio de una función polinomial o racional, siendo la representación polinomial la más ampliamente usada.

1.1.2.4. Modelos Hammerstein

Los modelos *Hammerstein* constan de la conexión en cascada de un bloque no lineal $\mathcal{N}(\cdot)$, seguido por el bloque lineal $G(q)$ como en la Figura 6.

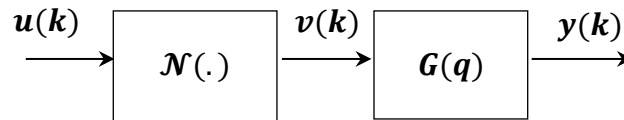


Figura 6. Modelo Hammerstein
Fuente: Elaboración propia

En el *modelo* Hammerstein la señal intermedia $v(k)$ se obtiene mapeando $u(k)$ a través de la función $\mathcal{N}$, esto es:

$$v(k) = \mathcal{N}(u(k)) \quad (1.60)$$

Luego

$$v(k-i) = \mathcal{N}(u(k-i)), \text{ para } i = 1, \dots, n_u \quad (1.61)$$

Si el bloque dinámico lineal se representa con un modelo ARX que se obtiene a partir de las medidas de entrada y salida, $v(k)$ y $y(k-j)$, se puede escribir:

$$y(k) = \sum_{j=1}^{n_a} \theta_j y(k-j) + \sum_{i=1}^{n_b} \sigma_i v(k-i) \quad (1.62)$$

donde:

- n_a : retardo de la salida del modelo ARX
- n_b : retardo de la entrada del modelo ARX
- θ_j : parámetros relacionados a cada regresor de salida del modelo ARX
- σ_i : parámetros relacionados a cada regresor de entrada del modelo ARX

Sustituyendo (1.61) en (1.62) se obtiene

$$y(k) = \sum_{j=1}^{n_a} \theta_j y(k-j) + \sum_{i=1}^{n_b} \sigma_i \mathcal{N}(u(k-i)) \quad (1.63)$$

En la práctica la señal intermedia $v(k)$, no está disponible. Como ya se ha mencionado antes, usualmente se expresa el bloque no lineal del modelo *Hammerstein* en forma polinomial en relación a los datos de entrada y salida del bloque, $u(k)$ y $v(k)$, respectivamente. En este caso la salida del bloque no lineal tiene la forma:

$$v(k) = \alpha_1 u(k) + \alpha_2 u^2(k) + \dots + \alpha_l u^l(k) \quad (1.64)$$

siendo k el instante de tiempo, $v(k)$ la pseudo-salida del bloque no lineal, $u(k)$ la variable de entrada y α_i ($i = 1, \dots, l$) son los coeficientes del polinomio l siendo este último el grado de no linealidad del modelo *Hammerstein*.

El modelo *Hammerstein* en su forma paramétrica puede escribirse como:

$$A(q^{-1})y(k) = B(q^{-1}) \sum_{i=1}^l \alpha_i u^i(k-d) + \varepsilon(k) \quad (1.65)$$

siendo d el tiempo muerto. Sustituyendo la ecuación (1.64) en la ecuación (1.65) se obtiene:

$$A(q^{-1})y(k) = B(q^{-1})v(k-d) + \varepsilon(k) \quad (1.66)$$

1.1.2.5. Modelos Wiener

En los modelos Wiener los bloques están dispuestos a la inversa de los modelos Hammerstein, como en la

Figura 7. Es importante considerar que en estos modelos algunos algoritmos de identificación requieren que la no linealidad estática sea invertible. Esta restricción está relacionada con el hecho de que la señal que conecta ambos subsistemas no es medible.

La señal de salida del modelo Wiener se obtiene por mapeo de la señal intermedia $v(k)$, a través de la función $\mathcal{N}(\cdot)$, tal que:

$$y(k) = \mathcal{N}(v(k)) \quad (1.67)$$

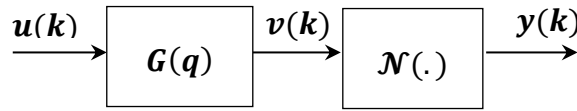


Figura 7. Modelo Wiener
Fuente: Elaboración propia

Como la señal $v(k)$ no está disponible, puede estimarse a través de la inversa de la función $\mathcal{N}(\cdot)$. Por tanto, esta función debe ser invertible para que el modelo pueda ser estimado.

El bloque dinámico lineal puede ser representado por un modelo ARX, determinado por el par de datos entrada y salida $u(k)$ y $v(k)$, respectivamente (Da Silva, 2009) como se muestra en la siguiente ecuación:

$$v(k) = \sum_{j=1}^{n_x} \theta_j v(k-j) + \sum_{i=1}^{n_u} \sigma_i u(k-i) \quad (1.68)$$

Reemplazando (1.68) en (1.67) se tiene:

$$y(k) = \mathcal{N} \left(\sum_{j=1}^{n_x} \theta_j v(k-j) + \sum_{i=1}^{n_u} \sigma_i u(k-i) \right) \quad (1.69)$$

Si la función $\mathcal{N}(\cdot)$ es invertible, como de hecho es requisito del modelo Wiener, se puede escribir

$$\mathcal{N}(\cdot)^{-1} = \mathcal{N}^{-1}(\cdot) \quad (1.70)$$

Si se considera un retraso j , tal que $j = 1, \dots, n_y$, se puede escribir:

$$v(k-j) = \mathcal{N}^{-1}(y(k-j)) \quad (1.71)$$

Reemplazando (1.71) en (1.69) se obtiene:

$$y(k) = \mathcal{N} \left(\sum_{j=1}^{n_x} \theta_j \mathcal{N}^{-1}(y(k-j)) + \sum_{i=1}^{n_u} \sigma_i u(k-i) \right) \quad (1.72)$$

1.1.2.6. Modelos Hammerstein-Wiener

En algunas aplicaciones resulta más adecuada la combinación de los modelos anteriormente descritos. En la Figura 8, se muestra un esquema en donde el bloque lineal se ubica entre dos bloques no lineales uno a la entrada y otro a la salida, denominado modelo Hammerstein-Wiener. En varios casos se asume que la no linealidad de entrada $N_1(\cdot)$ y la de salida $N_2(\cdot)$ son funciones polinómicas, como las descritas en la ecuación (1.64) aunque no es una condición necesaria.

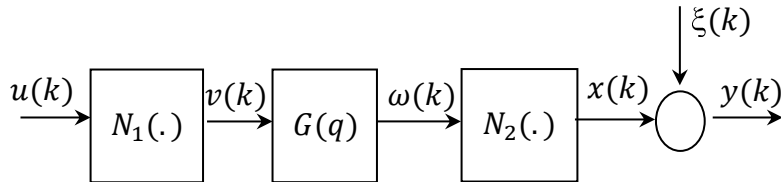


Figura 8. Modelo Hammerstein-Wiener
Fuente: Elaboración propia

En esta estructura es importante considerar que el bloque dinámico lineal se asume asintóticamente estable. Por otro lado, no es requisito que los bloques no lineales tengan inversa; pero en el caso que su uso sea en control predictivo es una condición necesaria, de manera que facilite el cálculo de los parámetros del control.

En general, se puede afirmar que la mayor dificultad del problema de identificación por un lado reside en el hecho que las señales internas v y ω no son accesibles para su medición y, por otro lado, en la forma de las no linealidades de las entradas y salidas; que por lo general son parcial o completamente desconocidas. Además la presencia de ruido $\xi(t)$, representa también otro problema a considerar en dicho proceso de identificación. Por simplicidad es frecuente asumir que el ruido es ergódico¹, pues se cumple que $\gamma_k = 0$, para $k \neq 0$ y de valor estacionario cero (Brouiri, y otros, 2014).

1.2. Métodos de identificación

En cualquiera de las estructuras orientadas a bloques que se han mencionado es necesaria la identificación de los elementos no lineales y lineales. En primer lugar, describiremos de manera muy sencilla algunos de los elementos no lineales que se suelen

¹ Un proceso estacionario es ergódico si los parámetros calculados en un conjunto de realizaciones (promedios de un conjunto) son iguales que los parámetros estadísticos calculados en una única realización (promedios temporales). Para una serie temporal se requiere como condición que $\lim_{k \rightarrow \infty} \gamma_k = 0$, siendo γ_k la covarianza.

utilizar para los modelos Wiener y Hammerstein y explicaremos a continuación los métodos de identificación del bloque lineal y no lineal usados en esta tesis.

Algunos de estos métodos permiten la identificación directa en una sola etapa de ambos bloques y otros realizan la identificación en dos etapas. En este caso el orden de la identificación dependerá de los elementos no lineales y lineales que se usarían. En el capítulo 3 se explicarán a detalle los métodos de identificación usados para una planta de neutralización de pH y las validaciones y análisis de cada uno de ellos.

1.2.1. Identificación en espacio de estado con método N4SID y descomposición SVD

En los últimos años se han desarrollado nuevos métodos de identificación que permitan encontrar modelos fiables de sistemas lineales invariantes en el tiempo LTI (*Linear Time Invariant*) en espacio de estado a partir de datos de entrada-salida y que requieren poca carga computacional, sin necesidad de procedimientos de optimización iterativos. Entre estos métodos se encuentran los conocidos como métodos de Identificación de sistemas en Espacio de Estado basados en Subespacios, 4SID (*Subspace-based State Space System IDentification*).

La principal herramienta computacional en dichos métodos es la descomposición QR y de valores singulares. Aunque hay una teoría bien desarrollada para métodos en subespacios de sistemas LTI, no la hay para sistemas no lineales. En (Gómez, 2005 a) se propone un algoritmo que consiste en dos pasos básicos. El primer paso es un algoritmo en subespacio (lineal) estándar aplicado a un sistema equivalente lineal cuyas entradas (respectivamente salidas) son versiones filtradas (por las funciones no lineales que describen las no linealidades estáticas) de las entradas originales (respectivamente salidas), mientras el segundo paso se trata de un problema de minimización de norma-2 que es resuelto por medio de la descomposición SVD. A continuación, explicaremos dichos métodos tratados con profundidad en (Gómez, 2005a)

1.2.1.1. Identificación del modelo Hammerstein

Como se ha visto en 1.2 un modelo Hammerstein es representado por un elemento no lineal en cascada con un sistema lineal LTI cuya representación en espacio de estado discreto es:

$$x_{k+1} = Ax_k + Bv_k + \omega_k \quad (1.73)$$

$$y_k = Cx_k + Dv_k + n_k \quad (1.74)$$

Donde para un sistema SISO se tiene $y_k \in \mathcal{R}$, $x_k \in \mathcal{R}^n$, $v_k \in \mathcal{R}$, $\omega_k \in \mathcal{R}^n$ y $n_k \in \mathcal{R}$, son la salida, estado, entrada, ruido del proceso y ruido en la medición de salida del sistema LTI en el instante k , respectivamente, y donde A, B, C y D son las matrices del sistema de dimensiones apropiadas.

Como explica Gómez (2005) se asume que el bloque sin memoria no lineal se puede describir por una combinación lineal de funciones bases de la forma:

$$v_k = \mathcal{N}(u_k) = \sum_{i=1}^r \alpha_i g_i(u_k) \quad (1.75)$$

donde $g_i(\cdot): \mathcal{R}, (i = 1, \dots, r)$, son las funciones base conocidas, $\alpha_i \in \mathcal{R}$, $(i = 1, \dots, r)$ son matrices de parámetros desconocidos y $u_k \in \mathcal{R}$ es el vector de entrada del modelo Hammerstein en el instante k . Como ya se ha mencionado, se usarán polinomios como funciones base, aunque también pueden usarse otras funciones generadas por funciones de traslación y dilatación de funciones madre (p.e. *wavelets*, o funciones *Radial Basis*) o métodos *piecewise linear*, entre otros.

El objetivo de la identificación es estimar las matrices o vectores de parámetros desconocidos $\alpha_i, (i = 1, \dots, r)$, además de las matrices A, B, C y D caracterizando, las partes no lineal y lineal del sistema y el orden del modelo n , a partir de los N pares de datos $\{u_k, y_k\}_{k=1}^N$ tomados de las medidas de entrada-salida del proceso.

Por simplicidad representaremos el modelo Hammerstein con el esquema de la Figura 9. Esta representación es válida tanto para sistemas SISO como para sistemas multivariables o MIMO.

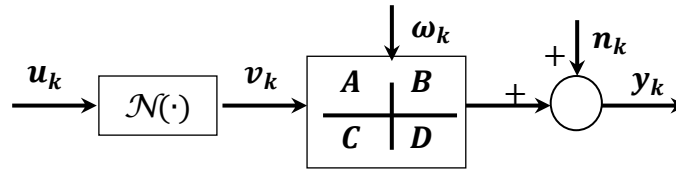


Figura 9. Modelo Hammerstein en espacio de estado
Fuente: Elaboración propia

Con el objetivo de ilustrar el método, se desestiman los efectos de los disturbios que se consideran nulos. Reemplazando la ecuación (1.75) en (1.73) y (1.74) se puede escribir:

$$x_{k+1} = Ax_k + \sum_{i=1}^r B\alpha_i g_i(u_k) + \omega_k \quad (1.76)$$

$$y_k = Cx_k + \sum_{i=1}^r D\alpha_i g_i(u_k) + n_k \quad (1.77)$$

De (1.76) y (1.77) se puede ver que la parametrización no es única pues la misma descripción se puede obtener con matrices de parámetros: $B\beta, D\beta$ y $\beta^{-1}\alpha_i$ para una matriz no singular $\beta \in \mathcal{R}$. Gómez (2005) explica que se deben imponer restricciones adicionales que permitan encontrar uno a uno los parámetros a identificar, esto es, para que el sistema sea identificable. Por esta razón se normalizan las matrices de parámetros α_i , asumiendo que

$\|[\alpha_1 \ \alpha_2 \ \cdots \ \alpha_r]^T\|_2 = 1$, de modo que la parametrización (1.76) y (1.77) es única. Como se verá más adelante esto es posible usando la descomposición en valores singulares de la matriz de parámetros a determinar con el criterio de mínimos cuadrados.

Así mismo los términos definidos en las sumatorias de las ecuaciones (1.76) y (1.77) se pueden escribir en forma matricial haciendo $\tilde{B} \triangleq [B\alpha_1 \ \cdots \ B\alpha_r]$, $\tilde{D} \triangleq [D\alpha_1 \ \cdots \ D\alpha_r]$ y $U_k \triangleq [g_1(u_k)^T \ \cdots \ g_r(u_k)^T]^T$, de modo que se puede escribir:

$$x_{k+1} = Ax_k + \tilde{B}U_k + \omega_k \quad (1.78)$$

$$y_k = Cx_k + \tilde{D}U_k + v_k \quad (1.79)$$

Las ecuaciones (1.78) y (1.79) se pueden interpretar como una realización de espacio de estado de un sistema LTI equivalente cuya entrada U_k es la versión filtrada de la entrada original u_k , (por la hipótesis de funciones base conocidas $g_i(\cdot)$, Gómez & Baeyens, 2005). A partir de aquí se puede usar cualquier algoritmo de identificación en espacio de estado (tales como N4SID de Van Overschee y de Moor, el algoritmo MOESP por Verhaegen o el algoritmo CVA de Larimore) para estimar $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ y $\tilde{D}$ de las matrices A , B , C y D respectivamente, de los datos entrada-salida.

En (Gómez y Baeyens, 2005b) se resumen los elementos de la representación del bloque lineal. El sistema correspondiente al modelo Hammerstein SISO se modela como un sistema de múltiples entradas y una salida y expandiendo las expresiones de $\tilde{B}$ y $\tilde{D}$ se puede escribir:

$$\tilde{B} = \begin{bmatrix} b_1\alpha_1 & \cdots & b_1\alpha_r \\ \vdots & \ddots & \vdots \\ b_r\alpha_1 & \cdots & b_r\alpha_r \end{bmatrix} = B\alpha^T \quad (1.80)$$

$$\tilde{D} = [d\alpha_1 \ \cdots \ d\alpha_r] = D\alpha^T \quad (1.81)$$

Se tiene entonces la matriz de parámetros (se recuerda que las matrices A y C fueron estimadas directamente de la aplicación del algoritmo N4SID):

$$[\tilde{B}^T \ \tilde{D}^T]^T \triangleq \Theta_{BD} = \begin{bmatrix} B \\ D \end{bmatrix} \alpha^T \quad (1.82)$$

En sentido de la norma-2 se pueden encontrar las estimas $\hat{B}$, $\hat{D}$ y $\hat{\alpha}$ como sigue:

$$(\hat{B}, \hat{D}, \hat{\alpha}) = \arg \min_{B, D, \alpha} \left\{ \left\| \hat{\Theta}_{BD} - \begin{bmatrix} B \\ D \end{bmatrix} \alpha^T \right\|_2^2 \right\} \quad (1.83)$$

Del álgebra lineal el problema de mínimos cuadrados planteado se puede resolver usando la descomposición de valores singulares SVD. Para sistemas en los que la matriz definida en (1.83) tiene número de filas mayor

que el de columnas o a la inversa, se puede aplicar la descomposición SVD económica, esto es:

$$\hat{\Theta}_{BD} = U_s \Sigma_s V_s^T = \sum_{i=1}^s \sigma_i u_i v_i^T \quad (1.84)$$

donde Σ_s es una matriz diagonal que contiene los s valores singulares no nulos σ_i , $i=1, \dots, s$) de $\hat{\Theta}_{BD}$ en orden decreciente, y las matrices $U_s = [u_1 \ u_2 \ \dots \ u_s] \in R^{(n+1) \times s}$ y $V_s = [v_1 \ v_2 \ \dots \ v_s] \in R^{r \times s}$ contienen solo las primeras s columnas de las matrices unitarias $U \in R^{(n+1) \times (n+1)}$ y $V \in R^{r \times r}$ que corresponden a la descomposición total SVD. Luego las matrices que minimizan la norma dada en (1.83) se encuentran como:

$$\left(\begin{bmatrix} \hat{B} \\ \hat{D} \end{bmatrix}, \hat{a} \right) = (U_1 \Sigma_1, V_1) \quad (1.85)$$

donde $\Sigma_1 = \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_p\}$, $U_1 \in R^{(n+1) \times 1}$ y $V_1 \in R^{r \times 1}$ que se obtienen de la partición de la descomposición (1.84) escrita en la forma:

$$\hat{\Theta}_{BD} = [U_1 \ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \quad (1.86)$$

Se puede verificar que el error de aproximación $\left\| \hat{\Theta}_{BD} - \begin{bmatrix} \hat{B} \\ \hat{D} \end{bmatrix} \hat{a}^T \right\|_2^2 = \sigma_{1+1}^2$. El vector (o matriz en el caso MIMO) de coeficientes α hallado de esta forma cumple la condición de norma unitaria, pues las componentes columna de V_1 son vectores unitarios.

Se hace notar que para el cálculo de los parámetros dados en (1.85) se ha aplicado la simplificación descrita en (Golub & Van Loan, 1996), por lo que dada una matriz $A \in R^{1 \times n}$:

$$k < r = \text{rank}(A) \quad (1.87)$$

Se puede aproximar la descomposición SVD de A como:

$$A_k = \sum_{i=1}^k \sigma_i u_i v_i^T \quad (1.88)$$

Donde A_k es una buena aproximación de A siempre y cuando $\sigma_k \gg \sigma_{k+1}$. Esta condición se cumple si el modelo encontrado con el método de identificación usado, en nuestro caso el método N4SID, nos da una buena aproximación de ajuste a los datos E/S usados.

1.2.1.2. Identificación del modelo Wiener

En (Gómez et al, 2005) se hace el planteamiento para la identificación del modelo Wiener usando la técnica N4SID (*Numerical algorithms for*

Subspace State Space System IDentification) con descomposición SVD. Los primeros pasos descritos en 1.2.1.1 para la identificación del modelo Hammerstein también se necesitan en este caso. La diferencia ahora radica en la ubicación de los bloques lineal y no lineal y, por tanto, en las señales que se definen para la identificación de estos bloques y los parámetros y matrices que se determinarían.

Para describir el método usaremos el diagrama de bloques para el modelo Wiener en espacio de estado mostrado en la Figura 10.

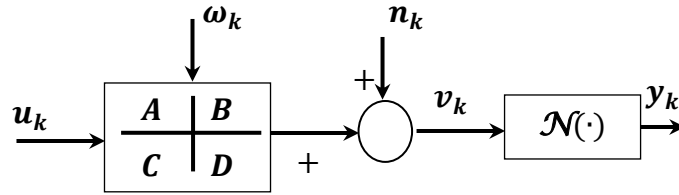


Figura 10. Modelo Wiener en espacio de estado
Fuente: Elaboración propia

donde $x_k \in \mathcal{R}^n$, $u_k \in \mathcal{R}$, $v_k \in \mathcal{R}$, $\omega_k \in \mathcal{R}^n$ y $n_k \in \mathcal{R}$ representan el estado, la entrada, salida del sistema LTI lineal y los correspondientes ruidos del proceso en el instante k .

La representación del bloque lineal en espacio de estado se puede escribir como:

$$x_{k+1} = Ax_k + Bu_k + \omega_k \quad (1.89)$$

$$v_k = Cx_k + Du_k + n_k \quad (1.90)$$

Asumiendo que el bloque no lineal descrito por la función $\mathcal{N}(\cdot) \in \mathcal{R}$ es invertible, su inversa $\mathcal{N}^{-1}(\cdot)$ puede describirse como:

$$\mathcal{N}^{-1}(y_k) = \sum_{i=1}^r \alpha_i g_i(y_k) = v_k \quad (1.91)$$

donde $g_i(\cdot) \in \mathcal{R}$, $(i = 1, \dots, r)$ son funciones base conocidas y $\alpha_i \in \mathcal{R}$, $(i = 1, \dots, r)$ son parámetros desconocidos. Con esta representación la ecuación (1.90) puede escribirse de la siguiente manera:

$$\alpha Y_k \triangleq \sum_{i=1}^r \alpha_i g_i(y_k) = Cx_k + Du_k + n_k \quad (1.92)$$

donde $\alpha \triangleq [\alpha_1, \dots, \alpha_r]$, $Y_k \triangleq [g_1^T(y_k), \dots, g_r^T(y_k)]^T$, siendo $y_k \in \mathcal{R}$ la salida del bloque no lineal.

El modelo Wiener entonces se puede escribir como:

$$x_{k+1} = Ax_k + Bu_k + \omega_k \quad (1.93)$$

$$Y = \tilde{C}x_k + \tilde{D}u_k + \tilde{n}_k \quad (1.94)$$

con $\tilde{C} \triangleq \alpha^\dagger C$, $\tilde{D} \triangleq \alpha^\dagger D$, $n_k \triangleq \alpha^\dagger n$ y donde $\alpha^\dagger$ es la pseudoinversa izquierda de α . También será necesario imponer la condición $\|\alpha\|_2 = 1$.

Así como en el caso del modelo Hammerstein se puede aplicar el algoritmo de identificación en subespacio de estado N4SID, MOESP o CVA para encontrar las estimas de las matrices del sistema $A, B, \tilde{C}, \tilde{D}$ a partir de los datos de entrada-salida. El problema es ahora estimar las matrices C, D y α . Se considera que el mejor ajuste se encuentra con la condición:

$$(\hat{C}, \hat{D}, \hat{\alpha}^\dagger) = \arg \min_{C, D, \alpha^\dagger} \left\{ \left\| \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} - \alpha^\dagger \begin{bmatrix} C & D \end{bmatrix} \right\|_2^2 \right\} \quad (1.95)$$

La solución para este problema de minimización se encuentra con la descomposición SVD de manera que:

$$\begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} = U_s \Sigma_s V_s^T = \sum_{i=1}^r \sigma_i u_i v_i^T \quad (1.96)$$

y la partición de esta descomposición conduce a:

$$\begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \quad (1.97)$$

con $\Sigma_1 = \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_m\}$, siendo $m = 1$ para un sistema SISO, $U_1 \in R^{r \times 1}$ y $V_1 \in R^{(n+1) \times 1}$ y, por último, se encuentra C, D y α como $\begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} = \Sigma_1 V_1^T$ y $\hat{\alpha} = U_1^\dagger$.

1.2.2. Identificación usando bases ortonormales y descomposición SVD

Otra de las alternativas planteadas para la identificación de bloques lineales y que se puede aplicar en la identificación de modelos orientados a bloques es la de usar bases ortonormales, como las descritas en 1.1.1.4, con la ventaja que se requiere de una menor parametrización.

1.2.2.1. Identificación modelos Hammerstein

En (Gómez et al, 2005) y (Zhang, Wang, Xiang, & Li, 2006) se presenta una alternativa de modelos Hammerstein usando bases ortonormales. Se asume que se tienen N bases ortonormales para describir la componente lineal del proceso y que se describe la componente no lineal por medio de funciones polinomiales.

El modelo Hammerstein representado en la Figura 11 se puede escribir de la siguiente manera:

$$y(k) = G(z^{-1})\mathcal{N}(u(k)) + n(k) \quad (1.98)$$

Donde para un sistema SISO: $u(k) \in \mathcal{R}^n$, $v(k) \in \mathcal{R}^n$, $y(k) \in \mathcal{R}$, $n(k) \in \mathcal{R}$ son la entrada del sistema, salida intermedia, salida del sistema y ruido de salida en el instante k , respectivamente.

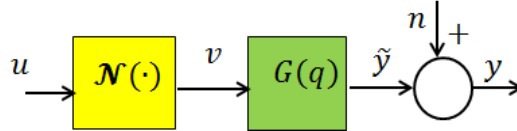


Figura 11. Modelo no lineal Hammerstein
Fuente: Elaboración propia

La salida puede escribirse de la siguiente manera:

$$y(k) = \left(\sum_{l=1}^N b_l B_l(z^{-1}) \right) \left(\sum_{i=1}^r a_i g_i(u(k)) \right) + n(k)$$

$$= \sum_{l=1}^N \sum_{i=1}^r b_l a_i B_l(z^{-1}) (g_i(u(k))) + n(k)$$

(1.99)

Donde $g_i(\cdot) \in \mathcal{R}$, $(i = 1, \dots, r)$ son funciones bases no lineales conocidas y $a_i \in \mathcal{R}$ son vectores de parámetros desconocidos en el caso de un sistema SISO (matrices en el caso del sistema MIMO) y $B_l(z^{-1})$ puede ser cualquier base ortonormal racional (ver 1.1.1.4), N es la longitud del truncamiento.

El diagrama de bloques de la Figura 12 ilustra el modelo Hammerstein con bases ortonormales.

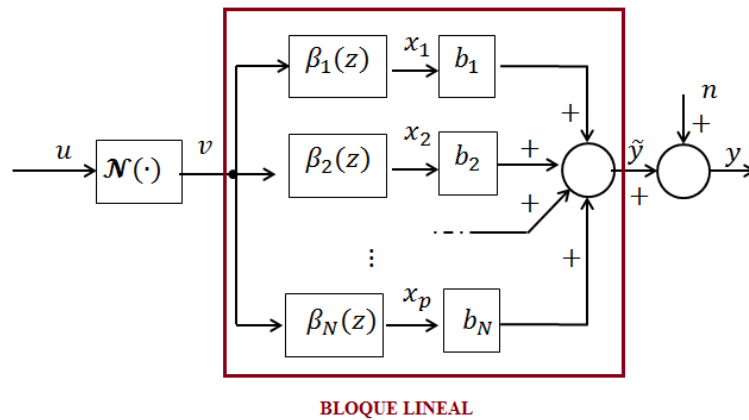


Figura 12. Modelo Hammerstein con bases ortonormales
Fuente: Elaboración propia

Con el objetivo de tener una única parametrización en a_i (o b_l), tal y como se explicó en 1.2.1.1, se establece la siguiente condición:

$$\|a_i\| = 1 \quad (i = 1, \dots, r) \quad (1.100)$$

Ahora se puede definir el vector de parámetros a determinar

$$\theta \triangleq [b_1 a_1, \dots, b_1 a_r, \dots, b_N a_1, \dots, b_N a_r]^T \quad (1.101)$$

$$\begin{aligned} \phi(k) \triangleq [& B_1 (g_1^T(u(k))), \dots, B_1 (g_r^T(u(k))), \dots \\ & B_N (g_1^T(u(k))), \dots, B_N (g_r^T(u(k)))]^T \end{aligned} \quad (1.102)$$

Con estas definiciones, la ecuación (1.94) se puede escribir en forma de regresión lineal como:

$$y(k) = \theta^T \phi(k) + n(k) \quad (1.103)$$

Si consideramos un conjunto S de datos entrada/salida, podemos definir:

$$Y_s \triangleq [y_1, \dots, y_s]^T, \Gamma_s \triangleq [\gamma_1, \dots, \gamma_s]^T, \Phi_s \triangleq [\phi_1, \dots, \phi_s]^T \quad (1.104)$$

por lo que ahora se puede escribir:

$$Y_s = \Phi_s^T \theta + \Gamma_s \quad (1.105)$$

Asumiendo que existe la inversa de las matrices, se sabe que la estima $\hat{\theta}$ de θ que minimiza los errores de predicción $\varepsilon_s = Y_s - \Phi_s^T \theta$, se encuentra usando la herramienta de optimización por mínimos cuadrados dada por:

$$\hat{\theta} = (\Phi_s \Phi_s^T)^{-1} \Phi_s Y_s \quad (1.106)$$

El problema ahora es estimar las matrices (o vector) de parámetros a_i ($i = 1, \dots, r$) y b_l ($l = 1, \dots, N$) de la estima $\hat{\theta}$.

Se puede escribir en forma matricial:

$$\Theta_{ac} \triangleq \begin{bmatrix} a_1^T b_1^T & a_1^T b_2^T & \dots & a_1^T b_N^T \\ a_2^T b_1^T & a_2^T b_2^T & \dots & a_2^T b_N^T \\ \vdots & \vdots & \vdots & \vdots \\ a_r^T b_1^T & a_r^T b_2^T & \dots & a_r^T b_N^T \end{bmatrix} = \mathbf{a} \mathbf{b}^T \quad (1.107)$$

en la que : $\mathbf{a} = [a_1, a_2, \dots, a_r]^T$, $\mathbf{b} = [b_1^T, b_2^T, \dots, b_N^T]^T$.

Se define $\theta = \text{blockvec}(\Theta_{ab})$, donde $\text{blockvec}(\Theta_{ab})$ es la matriz de bloques de columnas que resulta de colocar las columnas de Θ_{ab} una encima de la otra (Gómez y Baeyens 2004a). La estima $\hat{\Theta}_{ab}$ de Θ_{ab} se puede encontrar a partir de la estima $\hat{\theta}$ en (1.106).

El siguiente paso es estimar las matrices de parámetros $\mathbf{a}$ y $\mathbf{b}$ a partir de la estima $\hat{\Theta}_{ab}$ y una forma de encontrar dichos parámetros es resolviendo el problema de optimización:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = \arg \min_{\mathbf{a}, \mathbf{b}} \left\{ \|\hat{\Theta}_{ab} - \mathbf{a}\mathbf{b}^T\|_2^2 \right\} \quad (1.108)$$

En (Gómez, 2005) usando la técnica de descomposición SVD se resuelve dicho problema, haciendo:

$$\hat{\Theta}_{ab} = U\Sigma V^T \quad (1.109)$$

Las matrices $(\hat{\mathbf{a}}, \hat{\mathbf{b}})$ se pueden encontrar a partir de:

$$(\hat{\mathbf{a}}, \hat{\mathbf{b}}) = (U_1, V_1 \Sigma_1) \quad (1.110)$$

donde las matrices $U_1 \in R^{nr \times n}$, $V_1 \in R^{N \times n}$ y $\Sigma_1 = \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_n\}$ se determinan de la partición en la descomposición SVD económica de $\hat{\Theta}_{ab}$ definida como:

$$\hat{\Theta}_{ab} = [U_1 \quad U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \quad (1.111)$$

El error de aproximación se determina por:

$$\|\hat{\Theta}_{ab} - \hat{\mathbf{a}}\hat{\mathbf{b}}^T\|_2^2 = \sigma_{n+1}^2 \quad (1.112)$$

1.2.2.2. Identificación modelos Wiener

Siguiendo el mismo procedimiento visto para modelos Hammerstein también se puede encontrar el modelo Wiener de un sistema no lineal, usando el esquema dado en la Figura 13.

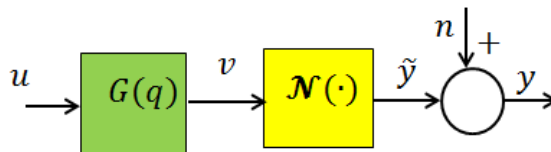


Figura 13. Modelo no lineal Wiener
Fuente: Elaboración propia

Podemos asumir en este caso que la matriz (o función) de transferencia $G(z)$ del subsistema lineal LTI es también representado por una base ortonormal de la forma:

$$G(z) = \sum_{i=1}^N b_i B_i(z^{-1}) \quad (1.113)$$

y para el subsistema no lineal, la función inversa de la no linealidad la podemos expresar de la siguiente manera:

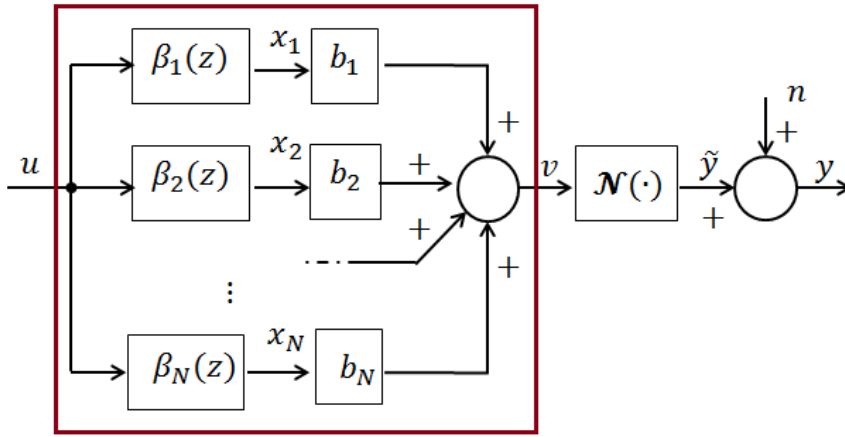
$$\mathcal{N}^{-1}(y_k) = \sum_{i=1}^r a_i g_i(y_k) \quad (1.114)$$

El diagrama de bloques del modelo Wiener con bases ortonormales se muestra en la Figura 14. La variable intermedia v_k se puede escribir como:

$$v_k = G(q)u_k \quad (1.115)$$

y también, simplificando la expresión de la salida sin considerar el efecto del ruido, se tiene que :

$$v_k = \mathcal{N}^{-1}(y_k) \quad (1.116)$$



BLOQUE LINEAL

Figura 14. Modelo Wiener con bases ortonormales
Fuente: Elaboración propia

Ahora $g_i(\cdot) \in \mathcal{R}, (i = 1, \dots, r)$ es el vector de parámetros conocidos y $a_i \in \mathcal{R}$ son las matrices de parámetros desconocidos.

Iguando las ecuaciones (1.115) y (1.116) se puede escribir:

$$g_1(y_k) + \sum_{i=2}^r a_i g_i(y_k) = \sum_{l=1}^N b_l B_l(z) u_k + n_k \quad (1.117)$$

O lo que es equivalente

$$g_1(y_k) = - \sum_{i=2}^r a_i g_i(y_k) + \sum_{l=1}^N b_l B_l(z) u_k + n_k \quad (1.118)$$

que ya es una regresión lineal y, por tanto, podemos definir:

$$\theta \triangleq [a_2, \dots, a_r, \dots, b_1, \dots, b_N]^T$$

$$\phi_k \triangleq [-g_2^T(y(k)), -g_3^T(y(k)), \dots, -g_r^T(y(k)), B_1(q)u_k^T, \dots, B_N(q)u_k^T]^T \quad (1.119)$$

En forma más compacta se tiene:

$$g_1(y_k) = \theta^T \phi_k + n_k \quad (1.120)$$

Como se ha explicado en el caso del modelo Hammerstein se puede estimar $\hat{\theta}$ a partir del criterio de minimización del error de predicción y considerando que se tienen S datos entrada/salida:

$$\hat{\theta} = (\phi_s \phi_s^T)^{-1} \phi_s Y_s \quad (1.121)$$

En este caso

$$Y_s \triangleq [g_1^T(y_1), g_2^T(y_2), \dots, g_1^T(y_s)]^T \quad (1.122)$$

$$\phi_s \triangleq [\phi_1, \dots, \phi_s] \quad (1.123)$$

En adelante se procede de manera similar a lo aplicado en el caso Hammerstein, recurriendo a la descomposición en valores singulares SVD, aunque la expresión de las ecuaciones que se minimizan para encontrar los parámetros resulta un poco más sencilla.

El bloque lineal viene dado por (1.113) pero en este caso se tiene más bien la expresión de la inversa del bloque no lineal:

$$\mathcal{N}^{-1}(y(k)) = \sum_{i=1}^r a_i g_i(y_k) \quad (1.124)$$

Por tanto, se encuentra la variable intermedia

$$v(k) = \sum_{l=1}^N b_l B_l(z) u_k = \sum_{i=1}^r a_i g_i(y_k) \quad (1.125)$$

Asumiendo $a_1 = 1$ por simplicidad, se puede despejar de (1.125):

$$g_1(y_k) = \phi^T(k) \theta \quad (1.126)$$

Con

$$\theta = [a_2, a_3, \dots, a_r, b_1, b_2, \dots, b_N]^T \quad (1.127)$$

$$\phi(k) = [-f_2(y(k)), \dots, f_r(y(k)) \quad \beta_1(q)u(k) \quad \dots \quad \beta_p(q)u(k)]^T \quad (1.128)$$

Aplicando el criterio de mínimos cuadrados para minimizar el error: $e(k) = g_1(y(k)) - \Phi^T(k)\theta$, se obtiene la estima de los parámetros desconocidos $\hat{\theta}$:

$$\hat{\theta} = (\Phi\Phi^T)^{-1}\Phi g \quad (1.129)$$

donde:

$$\Phi = [\phi^T(1); \quad \phi^T(2); \quad \dots \quad ; \phi^T(N)]^T \quad (1.130)$$

$$g \triangleq [g_1(y(1)) \quad g_1(y(2)) \quad \dots \quad g_1(y(N))]^T \quad (1.131)$$

Capítulo 2

Diseño de simulador de proceso de neutralización de pH

En muchos procesos de la industria química, agroindustria, alimentaria, farmacéutica, tratamiento de aguas residuales, etc. se requiere el control del pH. En procesos de neutralización se busca mantener el valor de pH alrededor del nivel neutro (en muchos casos $\text{pH}=7$). En (Kambale, George, & Zope, 2015), se hace un estudio de las aplicaciones de neutralización de pH y el control siendo que cada vez se vuelve necesario no sólo cumplir con los objetivos de control sino también con condiciones de seguridad y de protección ambiental. Se hace notar que en aplicaciones industriales el pH suele variar entre 2 y 10.

Una de las principales dificultades para el control del pH es que el proceso es inherentemente no lineal y con una gran sensibilidad alrededor del punto de neutralización. Además presenta ganancias variantes con el tiempo cuando las incertidumbres en los caudales y concentraciones de los agentes de neutralización están presentes. Muchos desechos industriales son principalmente alcalinos siendo un alto contaminante tanto del suelo como del mar, donde muchos desechos son arrojados. El control de pH determina la buena operación de un proceso, en (Priyanka, George, & Mobeen, 2015) se indica por ejemplo, que en la industria farmacéutica es importante garantizar la calidad del producto por lo que la neutralización del pH es de vital importancia.

Para un buen control de pH, en cualquier caso, se hace necesario contar con un modelo y/o técnicas de identificación que permitan diseñar el control.

2.1. Estado del arte de los simuladores de pH en la literatura

En esta sección se hace un breve resumen del estado de arte sobre el uso de simuladores o modelos de simulación de procesos de neutralización de pH que se han estudiado en aplicaciones para modelos y control no lineales. En (Fruzzetti, Palazoglu, & McDonald, 1997) se presenta una revisión de las diferentes técnicas de modelación usadas aplicadas a controles MPC de pH los cuales van desde un simple modelo lineal en una configuración adaptiva, múltiples modelos lineales como descripciones tipo *piecewise* para la descripción lineal y modelos tipo *fuzzy* así como modelos Wiener para la descripción no lineal y el uso de redes neuronales. En dicho documento se describen brevemente las propiedades generales de pH y los factores que originan su comportamiento no lineal y que más adelante describiremos para el caso concreto del modelo que se usará para el simulador que se ha usado para la validación de la técnica de control investigada.

Uno de los simuladores de procesos de neutralización de pH ampliamente utilizado por varios autores (p.e. Gómez et al, 2004, 2005), se muestra en la Figura 15 en el que se mezcla ácido nítrico (HNO_3) y soda caustica (NaOH) a volumen constante y como solución buffer se usa carbonato de sodio (NaHCO_3) que corresponde a una planta piloto de la Universidad de California en Santa Bárbara (UCSB). En dicho proceso por lo general se trata de mantener constante el valor de pH alrededor de la zona neutra (pH entre 6 y 8). Esta zona se caracteriza por ser altamente sensible y por ello es uno de los procesos más usados para identificar y modelar procesos no lineales.

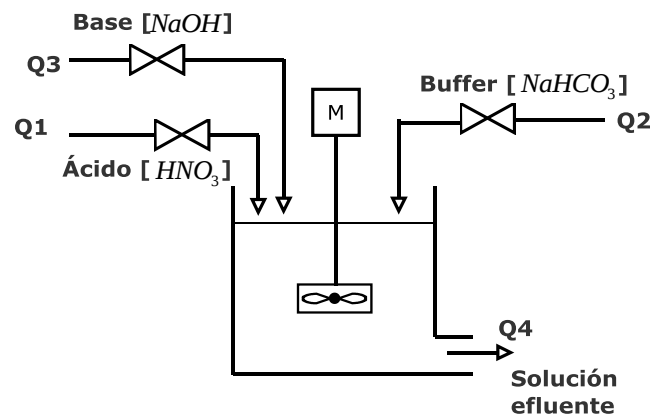


Figura 15. Proceso de neutralización de pH
Fuente: Elaboración propia

Esta planta incluye las características principales del proceso de naturaleza compleja tales como: comportamiento altamente no lineal, interacciones, retardos y comportamiento variante en el tiempo. En el caso de (Gómez y Baeyens, 2004 a) se considera dicho proceso en la configuración de un modelo Wiener que está compuesto de dos bloques, el bloque lineal que corresponde a la dinámica de mezclado del reactor (CSTR), mientras que la no linealidad estática es representada por la curva de titulación que da el pH de la solución de salida en función de los componentes químicos. Se usa el simulador de la UCSB para validar el algoritmo de identificación en subespacio de estado para la identificación simultánea de los bloques lineal y no lineal del modelo Wiener usando como bloque no lineal una aproximación polinómica. Los parámetros de los modelos se determinan por minimización de los errores de predicción. El mismo simulador es usado por (Arefi, Montazeri, Poshtan, & Jahed-Motlagh, 2006) en el que el bloque no lineal se obtiene usando redes neuronales (5 neuronas) y la parte lineal de la estructura Wiener corresponde a un modelo en espacio de estado de segundo orden.

El simulador de la planta en referencia cuyo diagrama se muestra en la Figura 16, incorpora las ecuaciones basadas en principios físicos de balances. El equilibrio químico puede modelarse introduciendo dos **invariantes de reacción** por cada flujo entrante. Para el caso en estudio y asumiendo que se toma como variable de interés el pH de la solución efluente se puede describir su dinámica, aplicando el balance másico al sistema. Las ecuaciones diferenciales que describen el comportamiento de las invariantes de reacción en la solución efluente (comportamiento dinámico del proceso de neutralización) son (Norquay, Palazoglu, & Romagnoli, 1998):

$$\frac{dW_{a4}}{dt} = \frac{1}{V}(W_{a1} - W_{a4})Q_1 + \frac{1}{V}(W_{a2} - W_{a4})Q_2 + \frac{1}{V}(W_{a3} - W_{a4})Q_3 \quad (2.1)$$

$$\frac{dW_{b4}}{dt} = \frac{1}{V}(W_{b1} - W_{b4})Q_1 + \frac{1}{V}(W_{b2} - W_{b4})Q_2 + \frac{1}{V}(W_{b3} - W_{b4})Q_3 \quad (2.2)$$

donde:

W_{a4} : invariante de reacción referida al balance del ión carbonato en la solución efluente

W_{b4} : invariante de reacción referida al balance de cargas en la reacción en la solución efluente

Q_1 : Flujo de la solución ácida

Q_2 : Flujo de la solución buffer

Q_3 : Flujo de la solución básica

V : Volumen del tanque (constante)

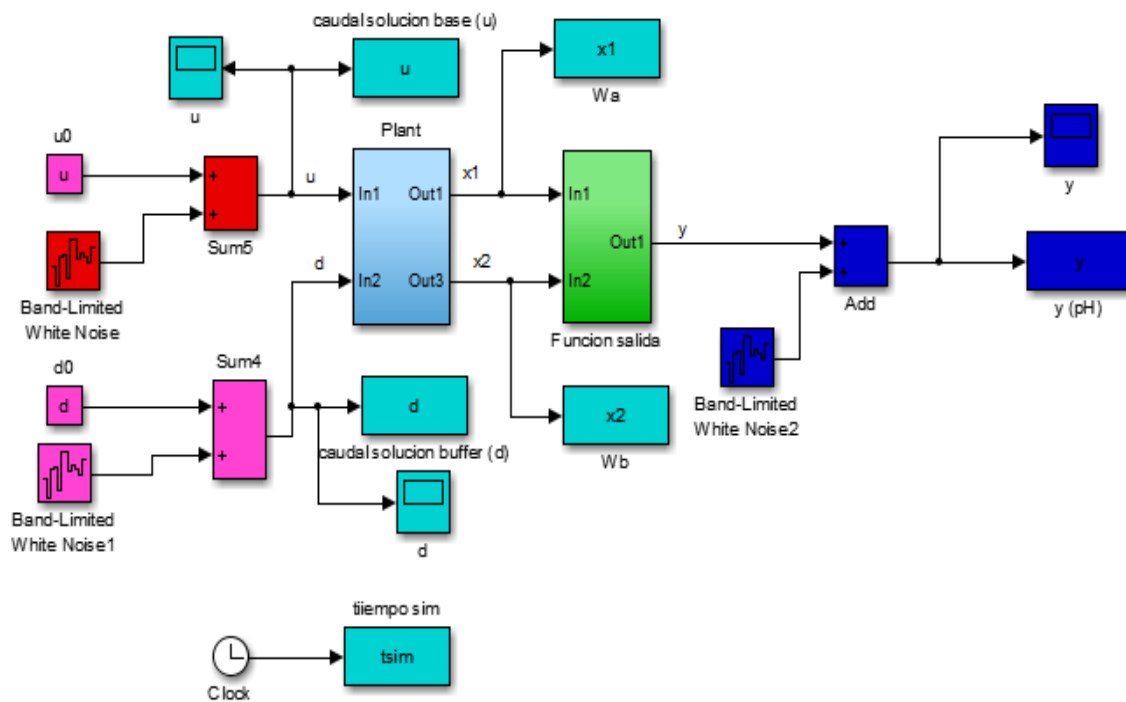


Figura 16. Esquema de la planta de neutralización de pH de la UCSB

Fuente: Elaboración propia

Las ecuaciones de equilibrio químico determinan la concentración del ion hidrógeno para las invariantes de reacción. Las constantes de equilibrio para este caso estarán dadas por las expresiones:

$$K_1 = \frac{\{HCO_3^-\}\{H^+\}}{\{H_2CO_3\}} \quad (2.3)$$

$$K_2 = \frac{\{CO_3^{-2}\}\{H^+\}}{\{HCO_3^-\}} \quad (2.4)$$

$$K_w = \{H^+\}\{OH^-\} \quad (2.5)$$

El pH de la solución efluente pH_4 se determina, a partir de las invariantes de reacción W_{a4} y W_{b4} , con definición de las constantes de disociación de H_2CO_3 :

$$pK_1 = -\log(K_1) \quad (2.6)$$

$$pK_2 = -\log(K_2) \quad (2.7)$$

se tiene así la ecuación implícita para pH_4 :

$$W_{a4} - 10^{-pH_4} + 10^{pH_4-14} + W_{b4} \frac{1 + 2 \cdot 10^{pH_4-pK_2}}{1 + 10^{pK_1-pH_4} + 10^{pH_4-pK_2}} = 0 \quad (2.8)$$

Definiendo el vector de estado:

$$x = [x_1 \quad x_2]^T = [W_{a4} \quad W_{b4}]^T \quad (2.9)$$

y las siguientes variables:

$u = Q_3$ variable manipulable (señal de control)
 $y = pH_4$ variable de salida (variable controlada)
 $d = Q_2$ perturbación (no medible)

El modelo expresado con las ecuaciones (2.1), (2.2) y (2.8) usando las variables definidas, se puede expresar como:

$$\dot{x} = f(x) + g(x)u + p(x)d \quad (2.10)$$

$$c(x, y) = 0 \quad (2.11)$$

donde:

$$f(x) = \left[\frac{q_1}{V} (W_{a1} - x_1) \quad \frac{q_1}{V} (W_{b1} - x_2) \right]^T \quad (2.12)$$

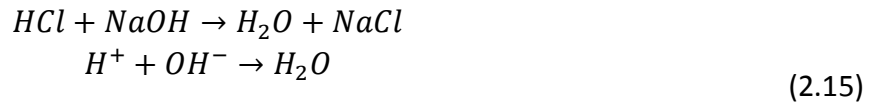
$$g(x) = \left[\frac{1}{V} (W_{a3} - x_1) \quad \frac{1}{V} (W_{b3} - x_2) \right]^T \quad (2.13)$$

$$p(x) = \left[\frac{1}{V} (W_{a2} - x_1) \quad \frac{1}{V} (W_{b2} - x_2) \right]^T \quad (2.14)$$

En (García & De Godoy, 2011) se ha elaborado en MATLAB/Simulink un modelo de planta de neutralización de pH que incluye la instrumentación además del modelo del reactor. En este caso el proceso de neutralización también contempla el uso del ácido (HCl) con soda caústica (NaOH); pero no considera la solución buffer como en el modelo de la UCSB. El esquema de la planta de neutralización usada se muestra en la Figura 17. Los programas y códigos son diseñados de manera que se facilita su uso para simulación y control bajo diferentes condiciones de operación. En el trabajo en referencia se diseña un controlador del tipo PID o PI para algunas variables.

Las reacciones implementadas en el proceso de neutralización cuyo diagrama se muestra en la Figura 18 se resumen a continuación:

- Reacciones químicas:



- Balance de masa en el CSTR:

$$V \frac{d}{dt} [Cl^-] = [Cl^-]_{in} \cdot Q_a - [Cl^-] \cdot Q_{out} \quad (2.16)$$

$$V \frac{d}{dt} [Na^+] = [Na^+]_{in} \cdot Q_b - [Na^+] \cdot Q_{out} \quad (2.17)$$

- Balance de neutralidad:

$$[Na^+] + [H^+] = [Cl^-] + [OH^-] \quad (2.18)$$

- Producto iónico del agua:

$$K_w = [H^+] \cdot [OH^-] = 10^{-14} \quad (2.19)$$

- Diferencia de la concentración iónica:

$$X = [OH^-] - [H^+] \quad (2.20)$$

La ecuación (2.20) junto a la ecuación (2.18) se puede escribir como:

$$X = [Na^+] - [Cl^-] \quad (2.21)$$

Y combinando (2.20) y (2.19) se puede despejar la concentración del ion hidrógeno:

$$\begin{cases} [H^+] = \frac{X}{2} \cdot \left(\sqrt{1 + \frac{4K_w}{X^2}} - 1 \right) & \text{si } X > 0 \\ [H^+] = -\frac{X}{2} \cdot \left(\sqrt{1 + \frac{4K_w}{X^2}} + 1 \right) & \text{si } X < 0 \\ [H^+] = \sqrt{K_w} & \text{si } X = 0 \end{cases} \quad (2.22)$$

Luego restando (2.16) de (2.17) se obtiene la ecuación diferencial que describe la dinámica del proceso:

$$V \frac{dX}{dt} = [Na_{in}^+] \cdot Q_b - [Cl_{in}^-] \cdot Q_a - X \cdot Q_{out} \quad (2.23)$$

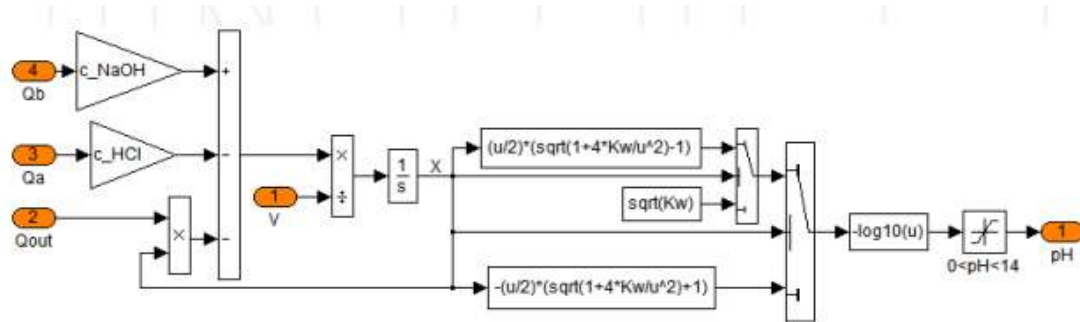


Figura 17. Esquema del proceso de neutralización de pH
Fuente: García, C., et al, 2011

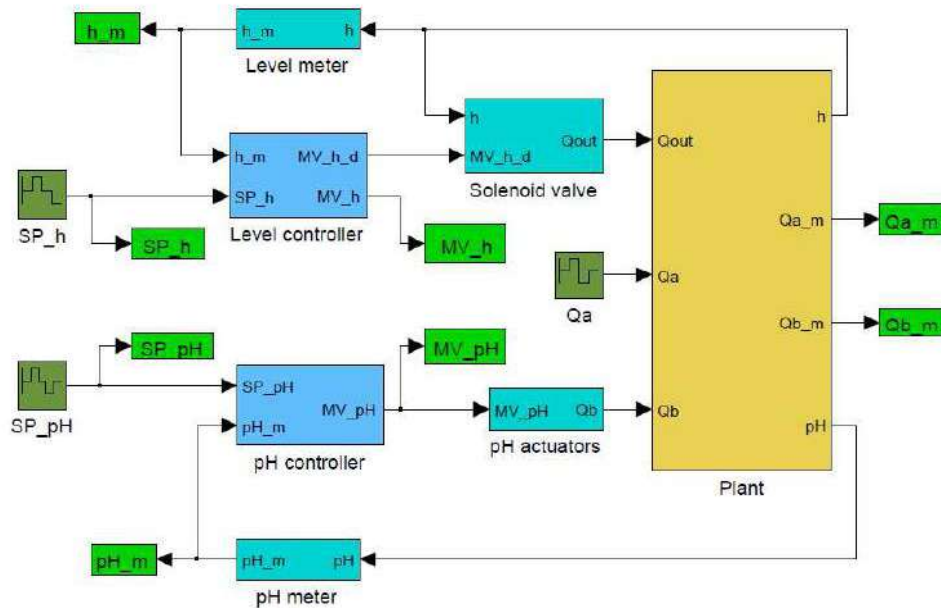


Figura 18. Representación de planta de neutralización de pH en Simulink
Fuente: García, C., et al, 2011

En (Norquay et al, 1998) también se usa un proceso de neutralización de pH de una solución de HCl con una solución de NaOH. Este proceso es usado para obtener un modelo Wiener que se usa para un control MPC. En (Fruzzetti, Palazoglu, & McDonald, 1997) se trabaja con el mismo proceso con otras concentraciones para obtener un modelo Hammerstein en el que se usa un modelo ARX para la parte lineal y una aproximación polinómica para la componente no lineal. Para encontrar los parámetros se usa un método de identificación ortogonal de acuerdo a un criterio de error estadístico. El modelo encontrado luego es usado para implementar un control MPC.

En (Franco, Mota, & Menezes, 2015) se presenta un modelo de neutralización de pH del mismo proceso que en los casos anteriores y se aplican modelos autorregresivos ARX, ARMAX y NARX que son usados como modelos de predicción. Los parámetros del modelo se toman de una planta piloto (ver Figura 19) y se evalúan los índices de desempeño alcanzándose valores del 98% en el mejor de los casos.

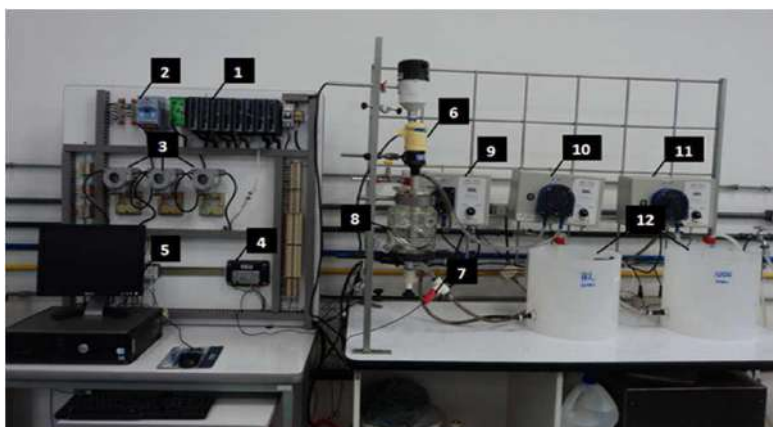


Figura 19. Planta piloto de neutralización de pH Centro Universitario da FEI
Fuente: (Franco, Mota, & Menezes, 2015)

Un modelo matemático de una planta piloto aplicando primeros principios del proceso de neutralización de ácido sulfúrico (H_2SO_4) y soda cáustica ($NaOH$) es implementado en Simulink, el cual se utiliza para implementar un controlador híbrido *Fuzzy Logic* y PID en (Naseer & Khan, 2013).

El diagrama de la planta piloto y el esquema del modelo implementado se muestran en la Figura 20 y en la Figura 21.

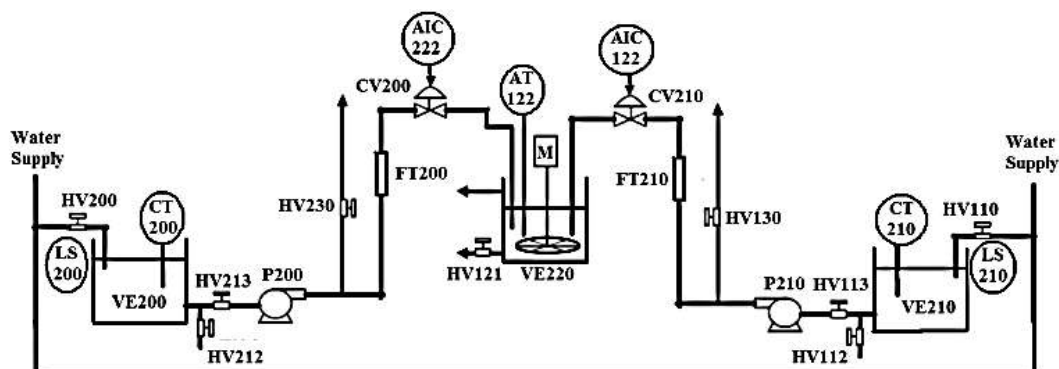


Figura 20. Planta piloto de School Engineering, University of Warwick.
Fuente: Naseer, O., et al, 2013

El modelo desarrollado se deduce a partir de las ecuaciones de balance de masa, balance de los componentes invariantes de reacción, las ecuaciones de neutralidad de las especies y combinando las constantes de equilibrio y producto iónico del agua, explicadas también ampliamente en (Ibrahim, 2008).

- Balance de masa:

$$V \frac{d\alpha}{dt} = F_1 C_1 - (F_1 + F_2) \alpha \quad (2.24)$$

$$V \frac{d\beta}{dt} = F_2 C_2 - (F_1 + F_2) \beta \quad (2.25)$$

siendo α y β los invariantes de reacción del ácido y la base del proceso.

Además se definen dichos invariantes haciendo un balance de carga, como sigue:

$$\alpha = [H_2SO_4] + [HSO_4^-] + [SO_4^{-2}] \quad (2.26)$$

$$\beta = [Na^+] \quad (2.27)$$

La condición de electroneutralidad viene dada por:

$$[Na^+] + [H^+] = [OH^-] + [HSO_4^-] + 2[SO_4^{-2}] \quad (2.28)$$

Las constantes de ionización y de producto iónico del agua en este caso son:

$$K_1 = \frac{[H^+] \cdot [HSO_4^-]}{[H_2SO_4]} \quad (2.29)$$

$$K_2 = \frac{[H^+] \cdot [SO_4^{-2}]}{[HSO_4^-]} \quad (2.30)$$

$$K_w = [H^+] \cdot [OH^-] \quad (2.31)$$

Las constantes de ionización del ácido sulfúrico son $K_1 = 10^3$ y $K_2 = 1.2 \cdot 10^{-2}$ y $K_w = 10^{-14}$.

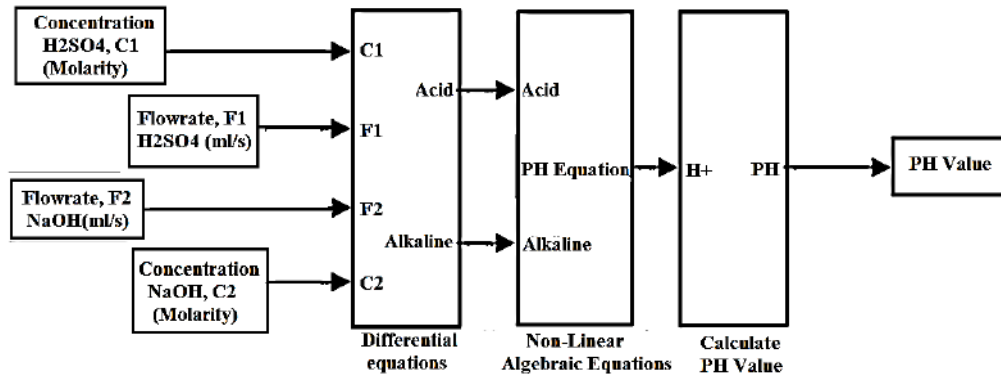


Figura 21. Modelo matemático del modelo de la planta de neutralización de pH.

Fuente: Naseer, O., et al (2013)

Combinando las ecuaciones de (2.26) a (2.31) se despeja la expresión que nos da el valor de $[H^+]$:

$$[H^+]^4 + a_1[H^+]^3 + a_2[H^+]^2 + a_3[H^+] + a_4 = 0 \quad (2.32)$$

siendo:

$$\begin{aligned} a_1 &= K_1 + \beta \\ a_2 &= K_1\beta + K_1K_2 - K_w - K_1\alpha \\ a_3 &= K_1K_2\beta - K_1K_w - 2K_1K_2\alpha \\ a_4 &= -K_1K_2K_w \end{aligned}$$

Se resuelve la ecuación (2.32) y luego se puede calcular $pH = -\log[H^+]$.

2.2. Estado del arte de control de plantas de neutralización de pH

Existen muchos trabajos desarrollados en control de los procesos de neutralización de pH para los que se necesita al menos una descripción, como los explicados en la sección anterior. Cabe mencionar que la mayoría de ellos se fundamentan en la metodología desarrollada por Thomas J. McAvoy presentada en el año 1972 desarrollada para el modelamiento de procesos de neutralización de pH en CSTRs en el que se aplican los principios de balance de masa en condiciones de equilibrio y los principios de neutralidad de las soluciones que se mezclan (ácidos- bases). Por otro lado, tal y como también se ha mencionado estos procesos son altamente no lineales con una alta sensibilidad alrededor del $pH=7$ o del punto de equilibrio en el caso de ácidos (base) fuerte con base (o ácido) débil, por lo que estos procesos son de especial interés en el estudio y aplicación de metodologías de control no lineal.

Haremos una breve descripción de alguna literatura desarrollada en la línea de control no lineal.

En (Henson & Seborg a), 1994) y (Henson & Seborg b), 1997 se desarrolla un algoritmo de control no adaptativo aplicado a la planta de neutralización de la Universidad de California antes descrita. El algoritmo usa el método de aproximación de linealización *input-output* tomando la derivada de la ecuación (2.11) y combinando con la (2.10) se

obtienen la derivada de la salida que se asigna como nueva variable de entrada al modelo, encontrándose un nuevo modelo en espacio de estado para el que se obtiene la ley de control. Será necesario diseñar un estimador usado para generar las estimas de los invariantes de reacción.

En (García & De Godoy, 2011) usando el modelo descrito en la sección precedente se desarrollan controladores PI tanto para el proceso como para los instrumentos de la planta, tal y como se muestra en la Figura 22.

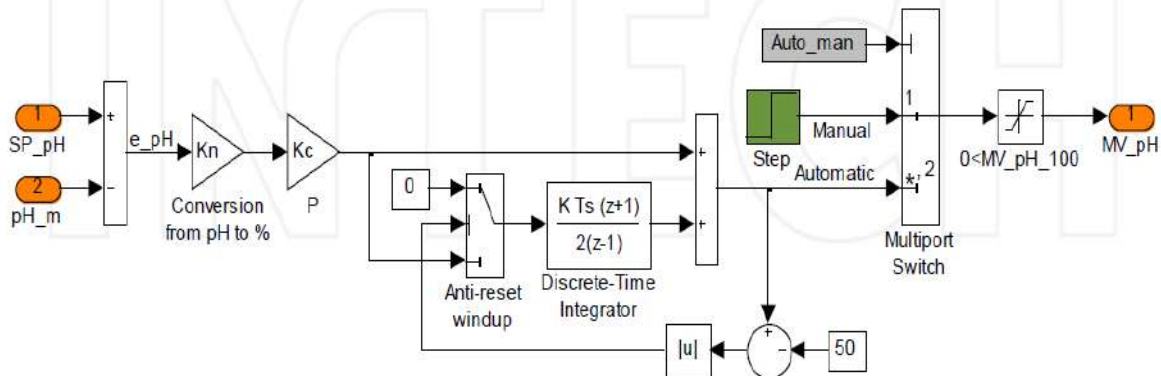


Figura 22. Controlador PI digital del proceso de pH

Fuente: García, C. et al, 2011

En (Prasad & Mathew, 2016) se diseña un control IMC- PI para un proceso de neutralización de pH producto de la reacción de ácido clorhídrico con soda cáustica. El modelo se obtiene aplicando primeros principios y se aproxima a una función de transferencia de primer orden con retardo. Contando con el modelo se aplica la metodología de diseño de controlador PI (método de Ziegler- Nichols) basado en la respuesta a escalón a lazo abierto. Una vez hecho esto, se diseña el controlador IMC. El controlador resultante se evalúa comparándose con un controlador simple PI usando los criterios ISE (*Integral Square Error*), IAE (*Integral Absolute Error*) y el criterio ITAE (*Integral Time Absolute Error*) resaltando un mejor comportamiento del algoritmo propuesto. Un método similar se aplica en (Sakthya, Dinesh, & Meenakshipriya, 2016) en el que se obtiene una aproximación del modelo de primer orden del proceso y se evalúan tres algoritmos PID usando los métodos Marlin, Smith *et al* y el método Branica *et al*, en los que la diferencia radica en la propuesta de cálculo de los parámetros del controlador. Del análisis de resultados se comprueba, usando los mismos criterios ISE, IAE e ITAE, que el controlador que mejor se comporta es el propuesto por Branica.

En (Chen, Peng, Han, & Guo, 2011) se aplica el modelo de lógica difusa con la técnica de control *sliding mode* para reducir el número de entradas que requeriría el controlador difuso (no requiere un modelo pero debe inferir las reglas en base a la información que se tenga del proceso y los estados). En la misma línea en (Naseer & Khan, 2013) se usa un controlador difuso pero en una configuración híbrida con un controlador PID de manera que el controlador difuso selecciona el *setpoint* para el controlador PID. Se concluye que esta propuesta es más estable y robusta que un controlador difuso simple.

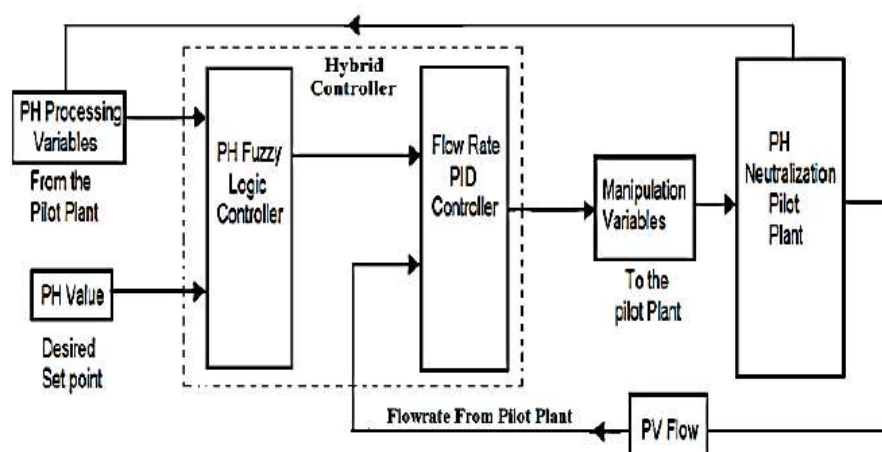


Figura 23. Diagrama controlador híbrido lógica difusa y PID
Fuente: Chen et al, 2013

Una de las alternativas del control no lineal es la de considerar los modelos orientados a bloques Wiener y Hammerstein vistos en el capítulo 1. Existe amplia literatura en la que se aplican estos modelos para el control de procesos no lineales como el proceso de neutralización.

En (Biagiola, Agamennoni y Figueroa, 2016) se obtiene un modelo Wiener de una planta de neutralización de pH entre un ácido fuerte (HA) y una base fuerte (BOH) en presencia de un agente buffer (BX). La neutralización se realiza en un tanque CSTR, como los vistos en otros casos con volumen constante. Se asume además que las velocidades de reacción son altas y que las condiciones de equilibrio químico se alcanzan instantáneamente además hay disociación total. Para la caracterización de la ganancia estática se hace una aproximación PWL asumiendo un rango de variables de entrada. El controlador se diseña en dos pasos: a) inversión de la ganancia no lineal y, b) el cálculo del controlador lineal (LTI) para compensar el comportamiento del modelo de bloque lineal del proceso. Para el diseño del controlador se usa la metodología H_∞ cuya robustez se analiza ante varias fuentes de incertidumbre: la variación de los parámetros del modelo lineal, el sector cónico del modelo de ganancia no lineal y el sector cónico del controlador. Un análisis más detallado se ha hecho también en (Biagiona y Figueroa, 2009) y (Biagiola, Agamennoni y Figueroa, 2004).

En (Hermansson & Syafiiy, 2015) se hace una revisión de los distintos métodos que se han usado para controlar procesos de neutralización especialmente enfocado a la aplicación de la metodología MPC. Se resumen algunos trabajos desarrollados usando modelos lineales, múltiples modelos lineales y los modelos no lineales. Un especial interés existe en estos últimos para los que se presentan alternativas como las basadas en modelos NARX y modelos Volterra. Se describe también brevemente las características encontradas de los controladores predictivos basados en modelos Wiener y Hammerstein y por último un análisis de los controles usando redes neuronales.

En (Babu & Swarnalatha, 2017) se hace un análisis comparativo de diferentes métodos de sintonización de controladores PID para la neutralización de pH en la industria textil. Estas son: Ziegler-Nichols, Tyreus Luyben, método C-H-R, criterio ITAE, control de modelo interno (IMC), método Cohen Coon.

En general, como ya se ha comentado antes, el proceso de neutralización de pH es uno de los ampliamente usados para validar métodos de identificación como algoritmos de control no lineal, algunos de los cuales se han descrito brevemente. Existen aún varias propuestas que no se han descrito en esta sección.

2.3. Descripción del proceso a modelar

En la industria existen diversas aplicaciones en las que se requiere control de pH, por ejemplo, en la industria alimentaria, plantas químicas, plantas de tratamiento de aguas residuales, aplicaciones de tratamiento o procesamiento de fuentes renovables y aprovechamiento de residuos agrícolas: fermentadores, biodigestores, en la industria petrolera, entre otros.

En la industria alimentaria se suele usar la medida del pH para determinar la madurez de una fruta, la evolución de una fermentación, la calidad de los alimentos y su grado de conservación.

Por otro lado, uno de los usos más frecuentes de la soda cáustica y ácido sulfúrico es en el tratamiento de aguas y la elaboración de alimentos. Las instalaciones de tratamiento de aguas efluentes usan dichas soluciones en el proceso de neutralización para el tratamiento de aguas residuales que pueden ser soluciones ácidas o básicas dependiendo de su origen y que pueden ser altamente contaminantes.

Algunas reacciones son exotérmicas por lo que en estos casos el proceso de neutralización es influenciado por la temperatura y se recomienda añadir un agente neutralizante a velocidad controlada.

Además de la necesidad del control de pH en las aplicaciones antes mencionadas, es importante recordar que por lo general la mayoría de procesos en los que se requiere dicho control, son no lineales.

En el caso del pH la curva de neutralización es altamente no lineal, especialmente en la zona cercana al valor neutro ($\text{pH}=7$), la cual dependerá a su vez de las sustancias químicas que se mezclan y la concentración de cada una de ellas.

2.3.1. Conceptos básicos en procesos de neutralización de pH

2.3.1.1. Definición ácido base

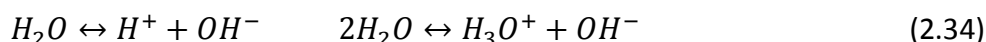
Existen diferentes formas de definir los ácidos y las bases. Según la teoría de Arrhenius (1884) un ácido es una sustancia que en una solución acuosa cede iones H^+ (o H_3O^+) y una base es una sustancia que cede iones hidroxilo OH^- . Sin embargo, esta teoría tiene dos limitaciones importantes: la primera que hay sustancias con propiedades básicas como el NH_3 que no tienen hidroxilos y la segunda que esta definición se limita solo a soluciones acuosas.

La teoría de Arrhenius fue mejorada por Brönsted y Lowry (1923) por la que un ácido es la especie capaz de ceder iones H^+ y la base es una especie capaz de aceptar iones H^+ . Con esta teoría se puede explicar el comportamiento básico del NH_3 .



2.3.1.2. Equilibrio iónico del agua

El agua aunque no es una solución iónica debido a su alto poder de polarización, se disocia en iones H^+ (proceso de autoionización) en una pequeña proporción y estos se unen en un enlace covalente con moléculas de agua formando el ion H_3O^+ .



En el caso del agua la constante de equilibrio teniendo en cuenta la ecuación (2.34), se define como:

$$K_c = \frac{[H^+][OH^-]}{[H_2O]} = \frac{[H_3O^+][OH^-]}{[H_2O]^2} \quad (2.35)$$

El producto iónico del agua también llamado de manera general constante de ionización es $K_w = K_c[H_2O]$ y se puede calcular como:

$$K_w = [H^+][OH^-] = [H_3O^+][OH^-] \quad (2.36)$$

A una temperatura de 25°C el valor de $K_w = 10^{-14}$. Dicho valor es válido tanto para agua pura como para especies disueltas en agua. En el caso del agua pura $[H_3O^+] = [OH^-] = 10^{-7}$, en disoluciones diluidas el producto iónico es constante por lo que un incremento del ion hidroxilo significará una disminución del ion hidrógeno y viceversa.

2.3.1.3. Definición de pH

Para medir el nivel de acidez de una solución se usa la escala de pH (potencial hidrogeniónico), que es la concentración de iones $[H^+]$ o de iones $[H_3O^+]$ a través de la siguiente función logarítmica:

$$pH = -\log_{10}[H^+] = -\log_{10}[H_3O^+] \quad (2.37)$$

Por razones de convención, según la variación de pH en la escala de 0 a 14 se clasifican como:

$pH < 7$	soluciones ácidas
$pH = 7$	soluciones neutras
$pH > 7$	soluciones alcalinas

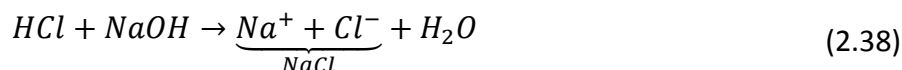
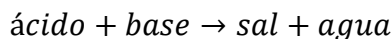
Otra clasificación de los líquidos según su acidez, alcalinidad y neutralidad es:

- Soluciones ácidas: $[H^+] > [OH^-]$
- Soluciones neutras: $[H^+] = [OH^-]$
- Soluciones básicas: $[H^+] < [OH^-]$

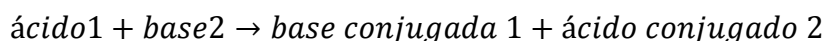
2.3.1.4. Reacciones de neutralización

La neutralización consiste en mezclar ácidos y bases de manera que se consiga una solución neutra, es decir, con un valor de pH igual a 7. Por lo general los productos de un proceso de neutralización son sales y agua.

Según Arrhenius las reacciones entre un ácido que aporta iones hidrógeno y una base que aporta iones hidroxilo es una reacción de neutralización, en la que se forma una mezcla de sal y agua.



La definición es ampliada por Brönsted en la que indica que una reacción ácido-base da lugar a una base y un ácido conjugado.



En los procesos de neutralización es importante tener en cuenta las características de las soluciones que se mezclan. Una de las características a considerar en el caso de los ácidos es el número de iones hidrógeno que tiene una molécula de ácido. En el caso de tener solo un átomo de hidrógeno la solución se denomina **monoprótica**, en el caso de tener dos se llama **diprótica** y para más de dos se denomina **poliprótica**. Una clasificación similar se puede hacer para el caso de las bases según el número de hidroxilos OH que contiene en su molécula se denominan **monohidroxilas** (un átomo), **dihidroxilas** (dos átomos) y **polihidroxilas** (más de dos átomos).

Los equilibrios de ionización de un ácido o base se caracterizan por una constante conocida como **constante de ionización** ácida (K_a) o básica (K_b). Para obtenerla se asume que $[H_2O]$ es constante, tal como se hizo en (2.35) y

(2.36) en las que a partir de la constante de equilibrio se obtiene la constante de ionización:

- Constante de acidez, de disociación o ionización



$$K_a = \frac{[A^-][H_3O^+]}{[HA]} \quad (2.41)$$

- Constante de basicidad, de disociación o ionización



$$K_b = \frac{[BH^+][OH^-]}{[B]} \quad (2.43)$$

La constante de ionización se considera como una medida cuantitativa de la fuerza del ácido o la base en la solución acuosa. Cuanto mayor sea K_a y/o K_b , mayor será la fuerza del ácido y/o de la base respectivamente, es decir, si $K_a \gg 1$ y si $K_b \gg 1$ se trata de ácidos o bases fuertes. Si por el contrario, la constante de equilibrio es menor que 1 se tienen ácidos y bases débiles.

Se puede definir las constantes de equilibrio o de ionización usando el mismo operador matemático $p = -\log_{10}$ como en el caso del pH, de modo que para un ácido se tiene $pK_a = -\log_{10}K_a$ y para bases $pK_b = -\log_{10}K_b$.

2.3.1.5. Valoración ácido-base

La valoración ácido-base es una técnica analítica que permite reconocer el punto de equivalencia de una reacción, es decir, el punto en el que se añade una cantidad equivalente de ácido y base, esto es, dadas una disolución de concentración de base (o ácido) conocida y una disolución de ácido (o base) de concentración desconocida se mide el volumen de la disolución de base (o ácido) necesario para que consuma todo el ácido (o base), este es el punto de equivalencia en la que se tiene la misma cantidad de ácido que de base.

El punto de neutralidad de una valoración es el punto en el que el pH es igual a 7, que es el punto de equivalencia en el que se ha alcanzado una neutralización completa, lo cual solo se da en el caso de ácidos y bases fuertes. Si se hace una valoración de ácido fuerte con base débil el pH es ácido (< 7) y si la valoración es de un ácido débil con base fuerte el pH es básico (> 7). Una gráfica de valoración usando una base fuerte para valorar soluciones de ácidos fuertes y débiles se muestra en la Figura 24 en la que si se valora un ácido fuerte con una base fuerte el pH=7 es el punto de equivalencia y es mayor en cuanto el ácido es más débil.

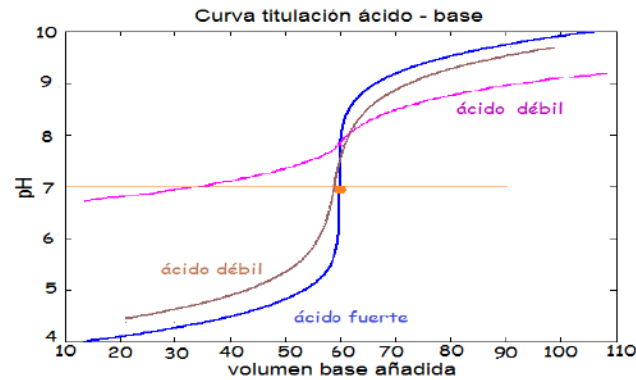


Figura 24. Curvas de titulación ácido con base fuerte
Fuente: Elaboración propia

2.3.2. Reacciones de neutralización del proceso a simular

En el caso de un proceso de neutralización de ácido sulfúrico (H_2SO_4) e hidróxido de sodio o soda cáustica (NaOH) se tienen las ecuaciones descritas a continuación.

– Balace de masa

Para el reactor de mezcla se hace el siguiente balance de masa asumiendo que el volumen V se mantiene constante

$$q_s = q_a + q_b \quad (2.44)$$

$$V \frac{dX_a}{dt} = q_a C_{a0} - q_s X_a \quad (2.45)$$

$$V \frac{dX_b}{dt} = q_b C_{b0} - q_s X_b \quad (2.46)$$

donde:

q_a	:	caudal de entrada de ácido sulfúrico (mL/s)
q_b	:	caudal de entrada de hidróxido de sodio (mL/s)
q_s	:	caudal de la solución efluente (mL/s)
X_a	:	concentración de ácido sulfúrico (mol/mL)
X_b	:	concentración de hidróxido de sodio (mol/mL)
C_{a0}	:	concentración inicial de ácido sulfúrico (mol/mL)
C_{b0}	:	concentración inicial de hidróxido de sodio (mol/mL)
V	:	volumen de tanque de mezclado (mL)

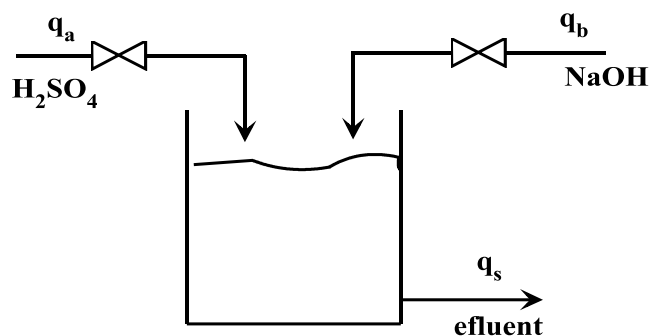
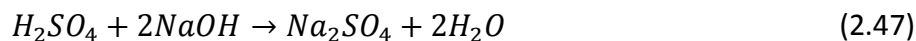


Figura 25. Esquema planta de neutralización de pH
Fuente: Elaboración propia

– Balance químico

Para hacer el balance químico se tiene en cuenta que la reacción en el proceso de mezcla del ácido y base tiene una curva de titulación diprótica, es decir, la disociación es completa prácticamente en la primera fase (la primera constante de disociación es muy alta) y aunque en realidad se requieren de dos etapas o reacciones de disociación parciales el planteamiento se puede simplificar. Las ecuaciones correspondientes, asumiendo que el proceso de neutralización no requiere de una solución buffer, se describen a continuación.

- *Reacción de neutralización*



- *Reacciones de ionización de ácido y base (asumiendo que la reacción es monoprótica y se desprecia una de las constantes de disociación):*



Por cada mol de ácido sulfúrico se requieren dos moles de hidrógeno, por lo que se puede escribir:

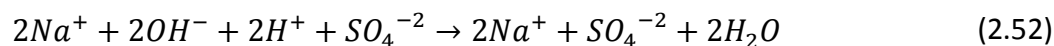
$$[H^+] = 2[H_2SO_4] = 2X_a \quad (2.49)$$



Por cada mol de soda cáustica se requiere solo un mol de ion hidroxilo

$$[OH^-] = [NaOH] = X_b \quad (2.51)$$

La disociación total de la mezcla ácido-base se puede escribir también de la siguiente forma:



Para el agua se tiene:



$$K_w = [H^+][OH^-] \quad (2.54)$$

donde K_w es la constante de equilibrio o producto iónico del agua

- Balance de cargas (electroneutralidad)

De la condición de neutralidad se puede deducir que:

$$[H^+] + [Na^+] = 2[SO_4^{-2}] + [OH^-] \quad (2.55)$$

Luego el invariante de reacción se escribe como:

$$L = [H^+] - [OH^-] = 2[SO_4^{-2}] - [Na^+] = 2X_a - X_b \quad (2.56)$$

Combinando las ecuaciones (2.51) a (2.56) se obtienen la ecuación diferencial de primer orden que definen

$$\frac{dL}{dt} = \frac{1}{V} (2q_a C_{a0} - q_b C_{b0} - (q_a + q_b)L) \quad (2.57)$$

$$L = 10^{-pH} - \frac{K_w}{10^{-pH}} \quad (2.58)$$

Despejamos pH y obtenemos

$$pH = -\log\left[\frac{1}{2}L + \sqrt{\frac{1}{4}L^2 + K_w}\right] \quad (2.59)$$

2.4. Diseño del simulador de pH

Para implementar el simulador tomamos las ecuaciones (2.57) a la (2.59) vistas en la sección anterior.

Para calcular los parámetros del simulador consideramos que se tienen las siguientes hipótesis:

- Mezcla homogénea
- Volumen constante
- No hay solución buffer

La primera constante de disociación del ácido sulfúrico es muy grande por lo que se considera disociación completa en la primera etapa.

Los datos iniciales del proceso, teniendo en cuenta que se usa como referencia un módulo de control de pH FESTO implementado en el laboratorio de Sistemas de Control de la Universidad de Piura, son:

- Soda cáustica: Concentración de ingreso de 0.003162 mol/L (calculado del valor de $\text{pH}=11.5$) para el ion hidroxilo (OH^-) que es igual a la de la soda caustica.
- Ácido sulfúrico: Concentración de ingreso de 0.003162 mol/L para el ion hidrógeno, por lo que teniendo en cuenta (2.49) la concentración de ácido sulfúrico es de 0.001581 mol/L
- Volumen $V=3\text{L}=3000\text{ mL}$

El diagrama del simulador se muestra en la Figura 26 que incorpora todas las ecuaciones descritas en 2.3.

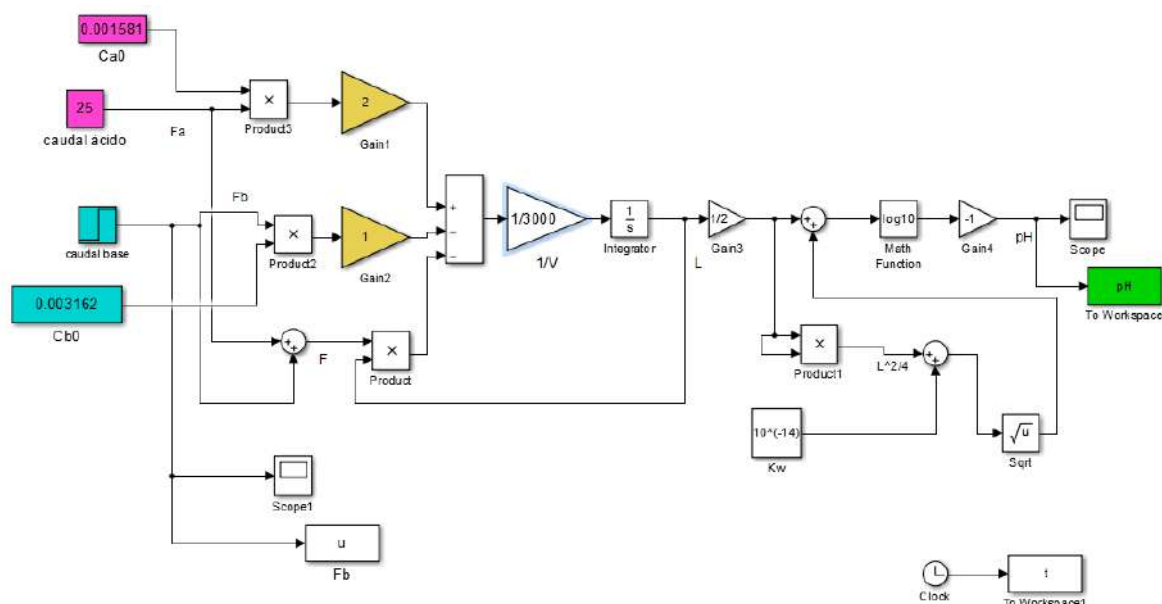


Figura 26. Modelo del proceso de neutralización en Simulink para variaciones de solución base
Fuente: Elaboración propia

Se ha considerado que las unidades de volumen son mL dada la sensibilidad del proceso a pequeñas variaciones de caudal, tal y como se muestra en la Figura 27 que corresponde a variaciones de caudal de la base (soda cáustica), manteniendo el caudal del ácido constante en las condiciones iniciales. Como se puede apreciar para pequeñas variaciones de caudal de solución base alrededor del punto neutro provoca grandes variaciones de pH. Se observan también las grandes variaciones en el tiempo de respuesta, característica propia de un sistema no lineal. Para el análisis realizado se ha considerado como condición inicial $\text{pH}=2.5$.

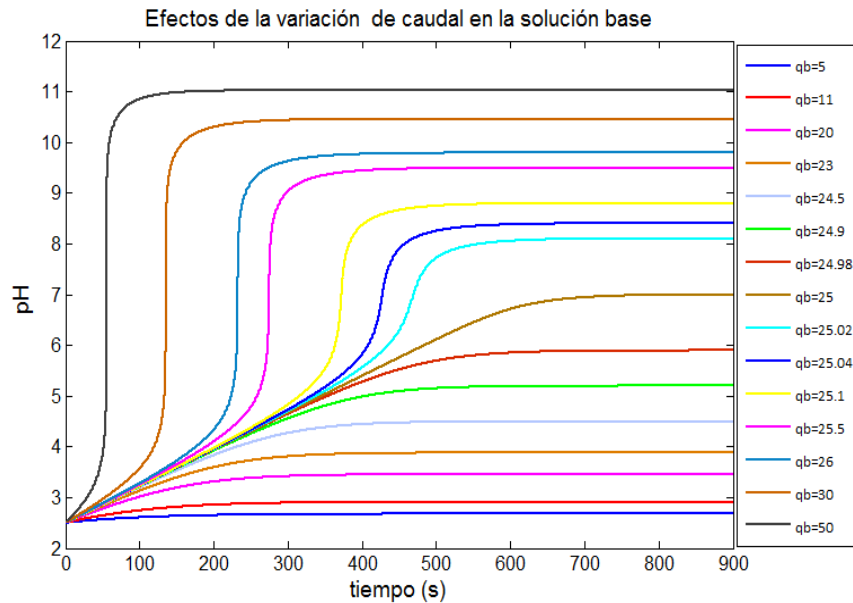


Figura 27. Variaciones de pH ante variaciones de caudal solución base (mL/s)
Fuente: Elaboración propia

A continuación se muestran los resultados de simulación de la variación de pH ante cambios en el caudal de solución del ácido sulfúrico, manteniendo el caudal de la solución básica constante y de acuerdo a las condiciones iniciales. El modelo usado en este caso se muestra en la Figura 28.

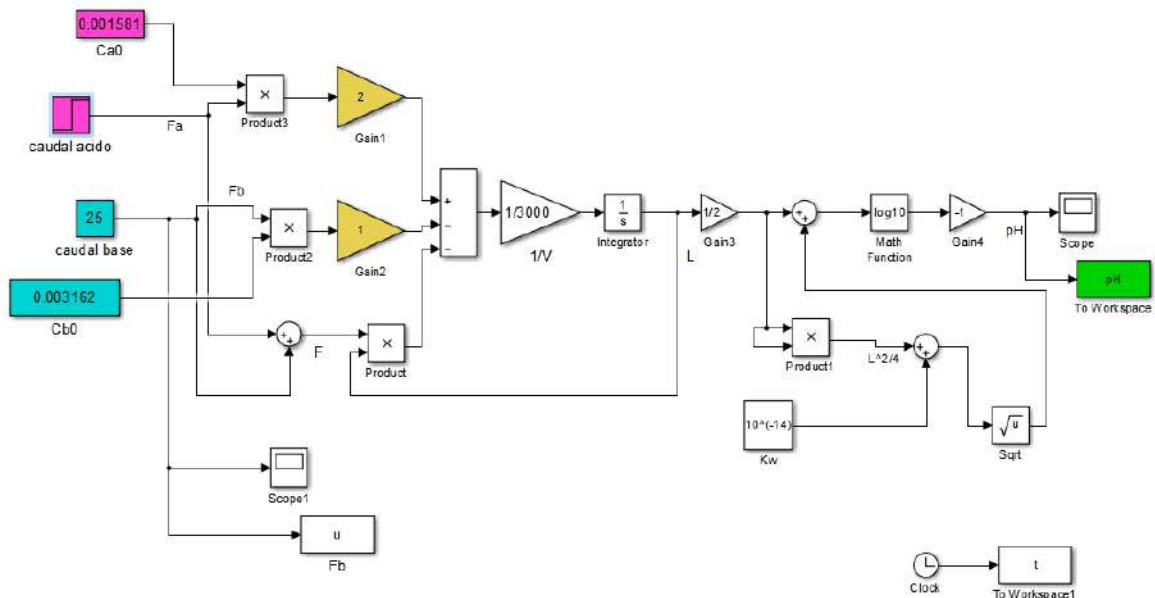


Figura 28. Modelo del proceso de neutralización en Simulink para variaciones de ácido
Fuente: Elaboración propia

A continuación se analiza el efecto en el valor de pH ante variaciones en el caudal de solución ácida de entrada, de la concentración de la solución base y solución ácida. Como se observa en las figuras 29 a 31, ante variaciones pequeñas de los caudales de entrada (así como de las concentraciones) cerca al punto de neutralización, se obtienen grandes variaciones de pH. Alejados de dicho punto se observa que es posible hacer grandes cambios de cualquiera de dichos valores y las variaciones de pH son menores.

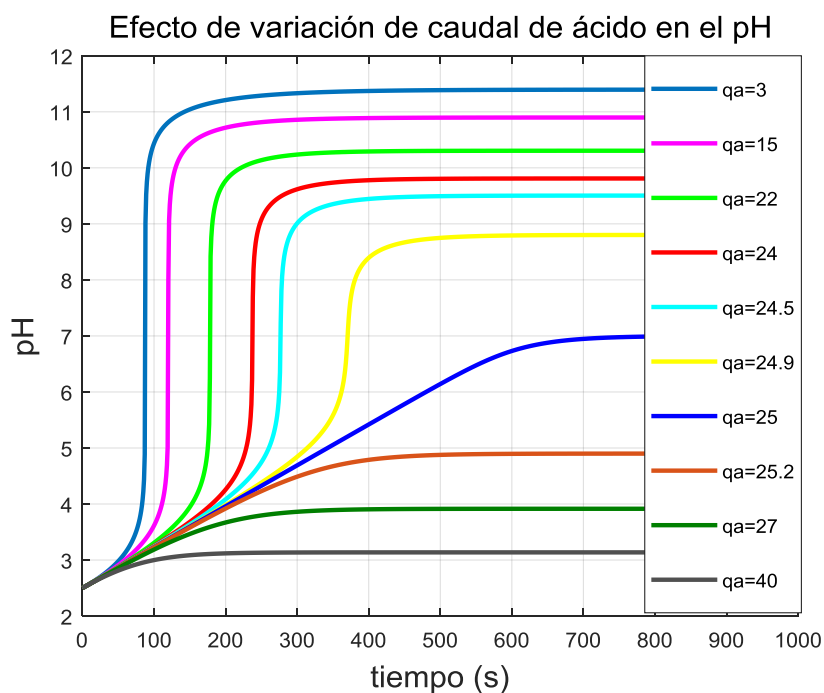


Figura 29. Variaciones de pH ante variaciones de caudal de solución ácida (mL/s)
Fuente: Elaboración propia

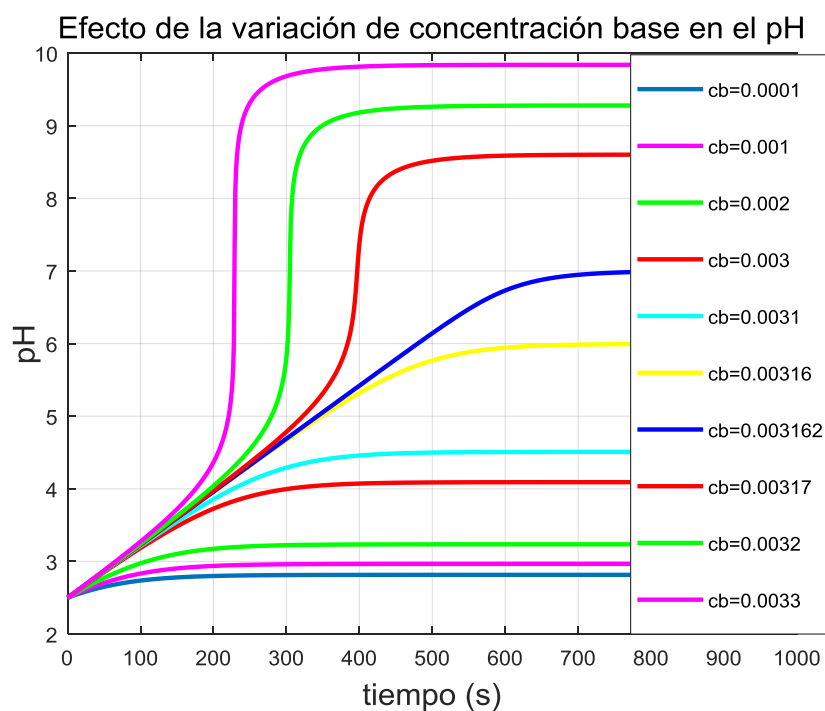


Figura 30. Variaciones de pH ante variaciones de concentración solución base (mol/L)
Fuente: Elaboración propia

Efecto de la variación de concentración del ácido en el pH

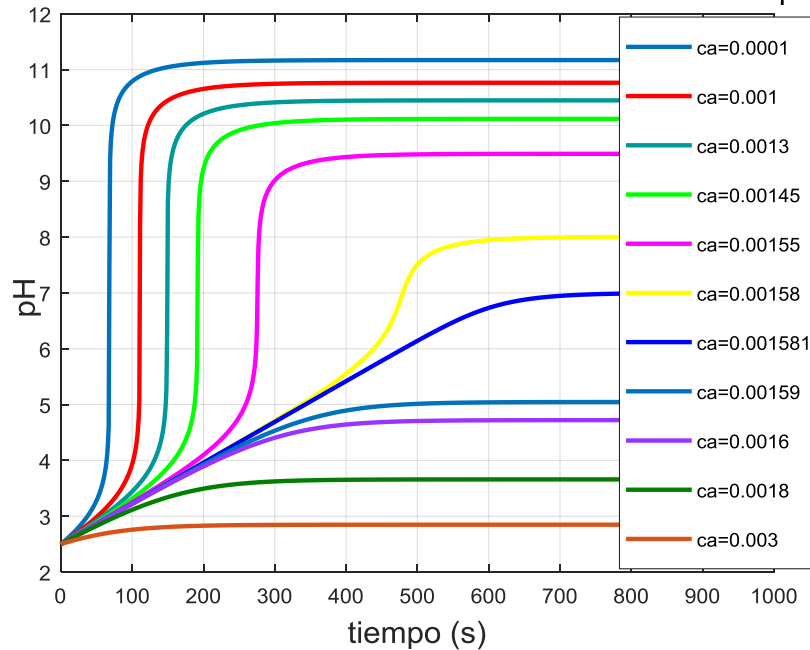


Figura 31. Variaciones de pH ante variación de concentración solución ácida (mol/L)
Fuente: Elaboración propia

Se ha implementado el modelo completo del proceso de neutralización considerando el ácido sulfúrico como ácido diprótico y usando las expresiones descritas en las ecuaciones desde la (2.26) a la (2.32). Las condiciones iniciales son las usadas en el modelo simplificado. El modelo implementado en Simulink se muestra a continuación.

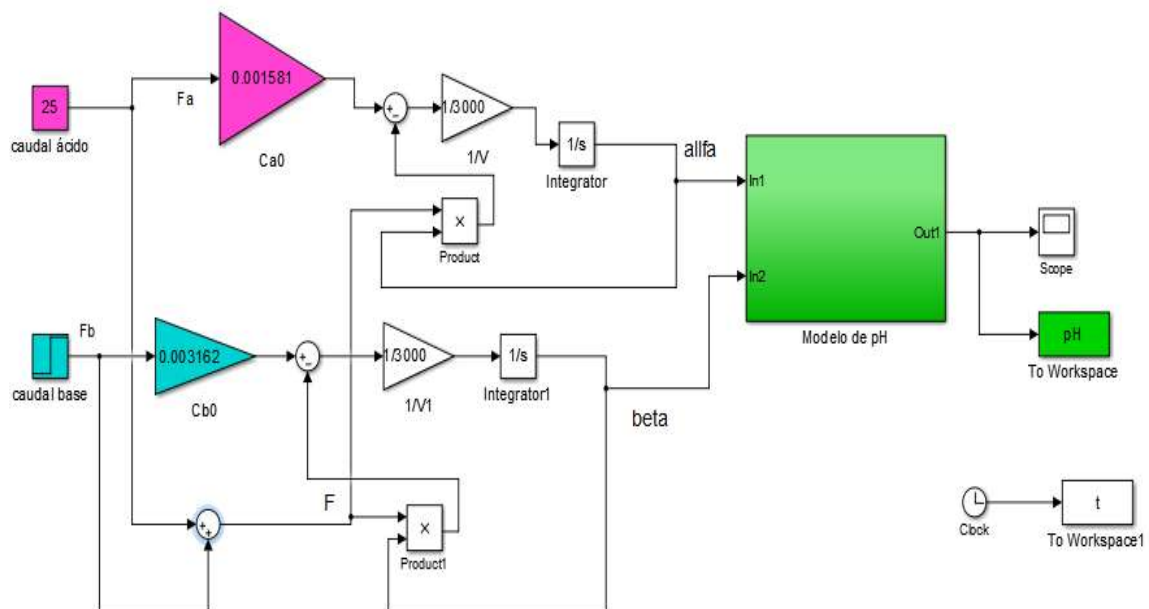


Figura 32. Modelo del proceso de neutralización para ácido sulfúrico diprótico
Fuente: Elaboración propia

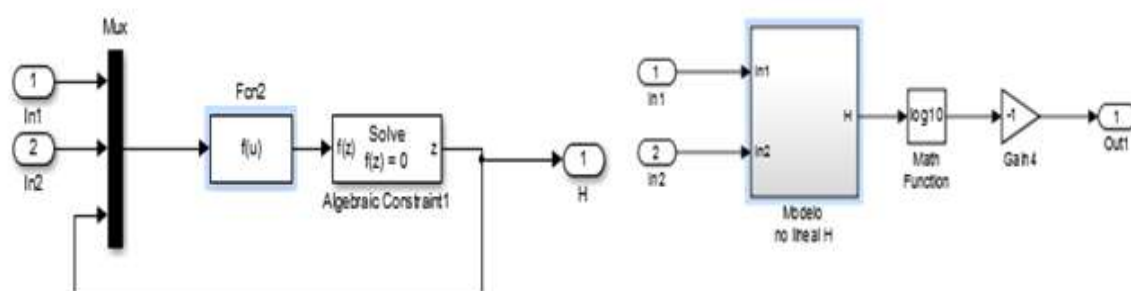


Figura 33. Bloque no lineal del modelo diprótico de pH
Fuente: Elaboración propia

Como se puede apreciar de la Figura 34 ambos modelos son válidos para los datos considerados en el proceso. Se ha graficado el comportamiento cuando el caudal de la solución base es una función escalón de 25 a 25.5 mL/s. Se observa la alta sensibilidad a pequeñas variaciones del caudal alrededor de dicho punto, como también se mostró para el modelo simplificado. Este comportamiento es evidenciado también en las gráficas anteriores ante variaciones de caudales (ácido y base) y ante variaciones de concentraciones. Para fines de control se trabajará alrededor de dicho punto de neutralización.

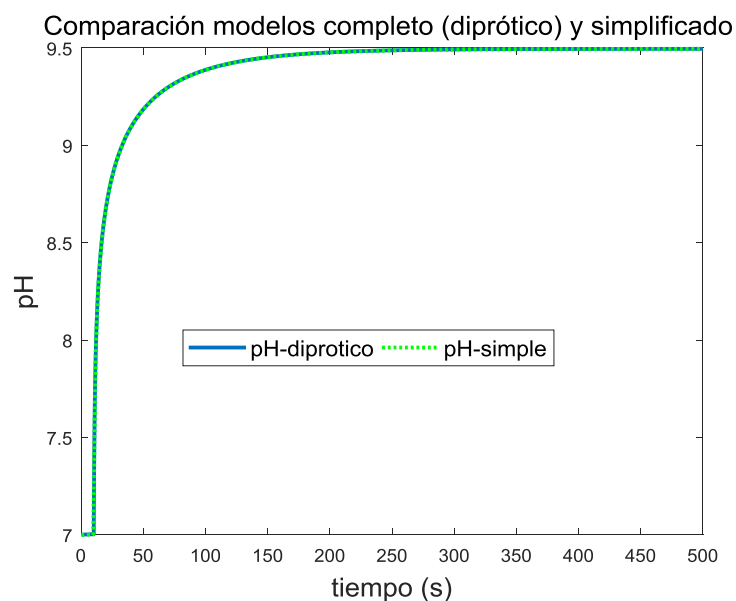


Figura 34. Comportamiento del pH en ambos modelos
Fuente: Elaboración propia

Capítulo 3

Obtención de los modelos Wiener y Hammerstein a partir de un simulador

En este capítulo se aplican los métodos de identificación vistos en el capítulo 1 usando como planta, el simulador del proceso de neutralización descrito en el capítulo precedente. Para encontrar el modelo usando las técnicas de identificación se deben analizar algunas características básicas del proceso a identificar, esto es, se deben hacer unos ensayos preliminares. Una vez hechos los primeros análisis, se diseñan las señales más adecuadas de entrada para aplicar al proceso. Esto se describe en la primera parte de este capítulo, luego se diseña el experimento, para finalmente aplicar los algoritmos de identificación. Se comparan los resultados haciendo ver que el modelo Wiener identificado con N4SID y descomposición SVD es el más oportuno.

3.1. Señales de prueba para identificación

La identificación de un proceso consiste en desarrollar un método experimental en el que usando medidas del proceso, se puede determinar un modelo según la estructura y el uso que se desea dar a dicho modelo.

Pueden encontrarse modelos para:

- a. Diseño: En el que por razones de seguridad y/o de ahorro se prefiere usar dicho modelo en lugar de experimentar en la planta real.
- b. Control: En el que el modelo permite predecir el comportamiento del proceso sin invertir mucho tiempo y teniendo en cuenta las dinámicas del proceso.
- c. Optimización: el modelo usado para optimizar algunos parámetros del proceso mismo o de todo el proceso de producción
- d. Detección de fallas

Existen técnicas de identificación *on line* y *off line*. En el primer caso, se parte de un modelo inicial y durante la operación se ajustan los parámetros del mismo, tomando los datos medidos en tiempo real. En el segundo caso se aplican una serie de señales y se almacenan los datos de entrada y salida del proceso que se desea identificar los cuales serán procesados posteriormente y usados para identificación según algunos de los métodos o estructuras descritas en el capítulo 1.

En este capítulo nos limitaremos a la aplicación de los métodos de identificación *off line* y se describirán algunas pruebas que requieren un mínimo conocimiento del proceso.

Para comenzar se debe tener en cuenta que existen algunas diferencias en las técnicas o métodos que se pueden aplicar si se requiere identificar un sistema lineal y un sistema no lineal (Mahmoodi, Poshtan, Jahed-Motlagh, & Montazeri, 2009). Se pueden aplicar sin mayor problema algunas pruebas o experimentos usados para identificación lineal, en algunos casos de no linealidad. Sin embargo se debe tener cuidado de utilizar la señal de excitación más adecuada. Una de las señales de excitación que se puede aplicar es la señal PRBS (*Pseudo Random Binary Sequency*), pero en el caso de un proceso no lineal, el modelo encontrado con esta señal estaría limitado pues al variar el rango de operación su comportamiento puede no ser representado adecuadamente. Para obtener un modelo no lineal resulta más conveniente aplicar una entrada GMN (*Generalized Multiple-level Noise*) que a diferencia de la PRBS además varía la amplitud de la señal en varios niveles limitados.

Para encontrar un modelo aplicando técnicas de identificación se deben seguir algunos pasos que son explicados brevemente en el diagrama de la Figura 35.

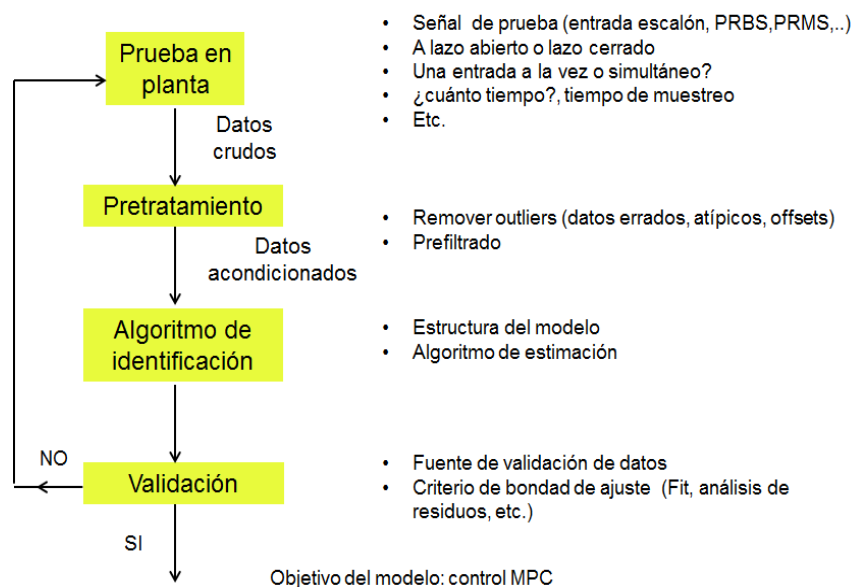


Figura 35. Diagrama del proceso de identificación
Fuente: Elaboración propia

3.1.1. Pruebas recomendadas

Algunas de las señales que pueden usarse para identificación no lineal (Zhu, 2001) son:

- a. **Prueba staircase:** Consiste en excitar al proceso con escalones tipo escalera ascendente y descendente. La ventaja es que con este tipo de señal se puede observar la dinámica del proceso en cada área (ganancias y constantes de tiempo). Se debe tener en cuenta que:
 - i. El número de escalones debe ser mayor o igual al grado del polinomio a ser identificado.

- ii. El ancho de los escalones (τ) debe ser: $\tau = T_s/4$, siendo $T_s = 98\%$ del *settling time*. Luego se debe usar escalones de ancho τ para $1/3$ el periodo de prueba, escalones de ancho 2τ para otro tercio del periodo de prueba y por último escalones de ancho 3τ para otro tercio del periodo de prueba. Se recomienda mezclar los escalones con anchos distintos cuando se genere la señal de prueba.

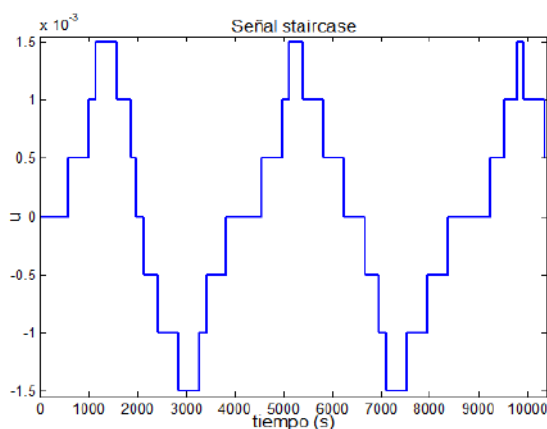


Figura 36. Señal de prueba escalera
Fuente: Elaboración propia

- b. **Prueba GMN:** Consiste en una señal de múltiples niveles al menos $m+1$ niveles siendo m el orden del polinomio que aproxima el comportamiento no lineal. Se prefiere una distribución uniforme en todo el rango de la señal. El tiempo de conmutación promedio, T_{sw} , entre los niveles es: $T_{sw} = T_s/3$, siendo $T_s = 0.98$ *settling time*.

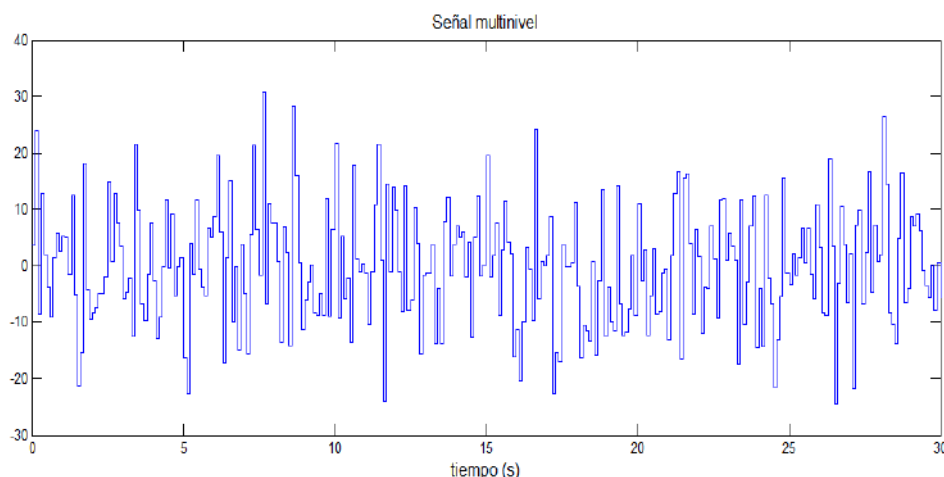


Figura 37. Señal multinivel
Fuente: Elaboración propia

- c. **Ruido blanco uniforme filtrado:** A menudo se usa un filtro pasabajo de primer orden. La ventaja frente a los otros dos métodos es que presenta una gran flexibilidad en la forma del espectro de la señal. Por lo general se usa la siguiente función:

$$v(t) = \frac{K}{1 - 0.6q^{-1}} e(t) \quad (3.1)$$

conocido como ruido blanco coloreado, siendo $e(t)$ el ruido blanco.

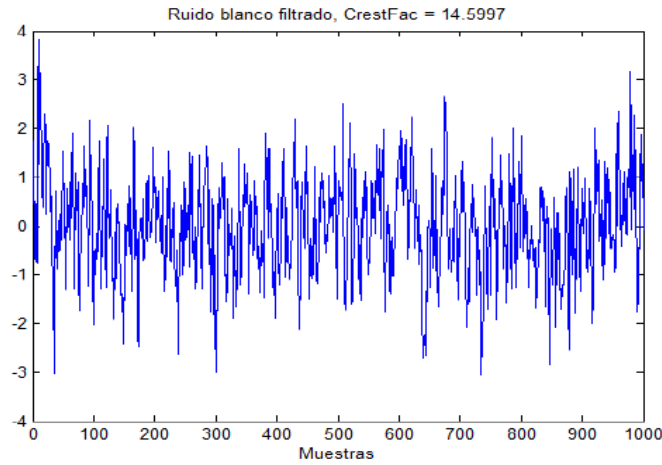


Figura 38. Ruido blanco coloreado
Fuente: elaboración propia

- d. **Señal multisenoidal:** Otra de las señales de prueba usadas consiste en la suma de un número finito de funciones cosenoidales. El desfase de las componentes de frecuencia influye en el factor de cresta de la señal y pueden escogerse de forma aleatoria con distribución uniforme dentro del intervalo: $[0, 2\pi]$. La señal de prueba tiene la forma: $y(t) = \sum_{k=0}^{n-1} A_k \text{sen}(2\pi f_k t + \phi_k)$, donde f_k es un múltiplo de $1/T$, siendo T el período de la señal multisenoidal.

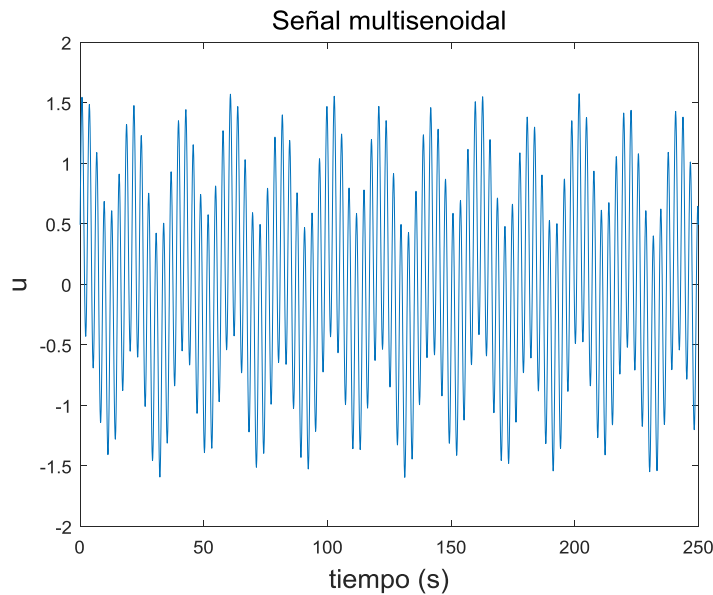


Figura 39. Señal de entrada multisenoidal con desfase
Fuente: Elaboración propia

3.1.1.1. Duración del experimento

Para tener medidas útiles en el proceso de adquisición de datos es necesario planificar la duración del experimento. Hay muchos criterios respecto a este parámetro. En (Zhu, 2001) se indica como criterio elegir entre 6 a 18 veces el *settling time* del proceso. La duración del experimento pudiera ser menor (entre 5 a 8 veces el *settling time*) si el número de entradas es pequeño y la relación señal/ruido es alta. Sin embargo debe ser lo suficientemente grande si hay muchas variables manipulables y la relación señal/ruido es baja (14 a 18 veces el *settling time*).

3.1.1.2. Selección tiempo de muestreo

Hay muchos criterios respecto a la selección del tiempo de muestreo, T_s . Por un lado la frecuencia de muestreo, según el teorema de Shanon, debe ser al menos el doble de las frecuencias máximas de una señal. Sin embargo, esto requiere el conocimiento de las armónicas más importantes del proceso. Una regla básica es que el periodo de muestreo sea el más pequeño posible, su valor dependerá entre otros también del sistema de cálculo. Como reglas empíricas se pueden elegir los siguientes criterios, los cuales dependerán del tipo de proceso (Ipanaqué, 2012):

- Por el tiempo de crecimiento, T_c : Sabiendo que T_c es el tiempo que tarda el proceso en alcanzar el 95% de su valor estacionario se puede elegir : $\frac{T_c}{15} < T_s < \frac{T_c}{5}$.
- Por la constante de tiempo mínimo, T_{min} : $T_s < \frac{T_{min}}{2}$, aunque en la práctica se recomienda $T_s < \frac{T_{min}}{5}$.
- Por el tiempo de asentamiento, t_s : Según (Zhu, 2001) el tiempo de muestreo se debe elegir entre $\frac{t_s}{100} \leq T_s \leq \frac{t_s}{20}$.

3.1.2. Diseño de los experimentos para la planta de pH

El diseño adecuado de los experimentos requiere tener algún conocimiento *a priori* del proceso: constantes de tiempo dominantes, ancho de banda de la dinámica del proceso, no linealidad y características de los disturbios. Una prueba inicial que siempre será de ayuda para una primera caracterización del proceso es la respuesta a escalón.

Para proceder a la identificación y evaluación de los modelos no lineales que se encuentren se tendrán en cuenta las condiciones iniciales de la planta dadas en la Tabla 2. El modelo en Simulink representado en la Figura 26 corresponde a una planta de neutralización de ácido sulfúrico y soda cáustica, basado en los primeros principios o principios físicos, sobre el cual aplicaremos los métodos de

identificación estudiados y bajo las condiciones nominales recomendadas por el fabricante.

Tabla 2. Condiciones de operación nominal

Parámetro	Símbolo	Valor	Unidades
Caudal de la solución ácida	q_a	25	mL/s
Caudal de la solución base	q_b	25	mL/s
Primera constante de disociación ácida	Ka_1	$3 \cdot 10^3$	mol
Segunda constante de disociación ácida	Ka_2	$1.02 \cdot 10^{-2}$	mol
Concentración de entrada del ácido	C_{a0}	0.001581	mol/L
Concentración de entrada de la base	C_{b0}	0.003162	mol/L
Volumen del reactor	V	3000	mL

Se ha realizado una serie de ensayos (simulaciones a lazo abierto) al modelo computacional de la planta de neutralización (previos a la identificación) a fin de estudiar la dinámica, determinar la no linealidad y con esta información preparar el experimento de identificación escogiendo la duración del mismo y el tiempo de muestreo, entre otros parámetros. Manipularemos u (caudal de la solución base), manteniendo en todo momento constante el flujo de la solución ácida en el valor nominal de 25 mL/s.

De las pruebas a lazo abierto vistas en el capítulo 2 realizadas en el simulador ante variaciones de caudal de ácido, base y concentraciones, se determina que el sistema es estable. Esto facilitará el análisis exhaustivo que se requerirá para la identificación así como el diseño del controlador del proceso.

Una vez diseñadas las entradas de excitación del proceso y que se han tomado datos entrada/salida del mismo, se debe realizar una identificación del proceso. Para ello es necesario seleccionar un rango para la estimación y otro rango que se usa para la validación cruzada.

El modelo identificado será válido si se cumplen unas condiciones o índices, algunos de los cuales usaremos en este capítulo y que se detallan a continuación.

MSE (Mean Square Error)

$$MSE = \frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2 \quad (3.2)$$

donde

y_k : salida real
 $\hat{y}_k$: salida del modelo
 N : datos de validación

FIT (Best Fit)

$$FIT = \left(1 - \frac{\|Y - Y_v\|}{\|Y - y_{mean}\|} \right) \times 100 \quad (3.3)$$

donde

Y : vector que contiene la salida del modelo
 Y_v : vector que contiene los datos de salida reales de validación
 y_{mean} : valor medio de Y

VAF (Variance Accounted For)

$$VAF = \max \left\{ 1 - \frac{Var(y - \hat{y})}{Var(y)}, 0 \right\} \times 100\% \quad (3.4)$$

donde

$y = \{y_k\}_{k=1}^N$: secuencia de la salida real

$\hat{y} = \{\hat{y}_k\}_{k=1}^N$: secuencia de la salida del modelo

$Var(\cdot)$: varianza de la señal cuasi-estacionaria

La señal que se use para la identificación debe excitar las frecuencias del proceso y además en el caso concreto de un proceso no lineal como el que se evalúa, en lugar de aplicar una PRBS se usará una señal del tipo GMN (*Gaussian Multilevel Noise*) o una PRS (*Pseudo Random Signal*) que nos permitan analizar las respuestas a cambios en rangos de amplitud y frecuencia diversos. La razón de ello es que, como ya se ha mencionado, una señal PRBS solo varía entre dos niveles, lo cual no permitirá evaluar la no linealidad del proceso.

En los métodos de identificación que usamos en este capítulo se ha asumido en primer lugar que la componente no lineal es de forma polinómica, pues de lo que se ha revisado en la literatura es una de las formas que se ajusta bastante bien para el tipo de no linealidad de procesos de neutralización de pH y por su simplicidad en la aplicación de algunos de los métodos estudiados.

3.2. Pruebas para la identificación

Para evaluar la no linealidad del proceso se ha considerado en esta segunda prueba preliminar un tiempo de simulación total de 16000 s. y se ha elaborado un experimento sencillo en el que a partir de las condiciones iniciales de operación dadas en la Tabla 2, se aplican escalones (entrada tipo escalera) de las condiciones estacionarias cada 800s (teniendo en cuenta que el tiempo de respuesta máximo a partir de las condiciones iniciales a pH=2.5 es alrededor de los 500 segundos) para garantizar que se llega al estacionario antes de aplicar otro cambio en la entrada.

En la Figura 40 y Figura 41 se puede apreciar el comportamiento dinámico del proceso y la alta sensibilidad de variaciones del caudal base alrededor de $\text{pH}=7$. Los valores estacionarios permiten también calcular una aproximación polinómica del comportamiento no lineal correspondiente a la curva de neutralización del proceso.

Es importante resaltar que las pruebas que se realizarán en este análisis preliminar ayudarán también para determinar los rangos de variación alrededor del valor estacionario o de operación que permita excitar las frecuencias (determinando el tiempo de respuesta y por tanto calculando el tiempo de muestreo de la señal de excitación) y las no linealidades del proceso.

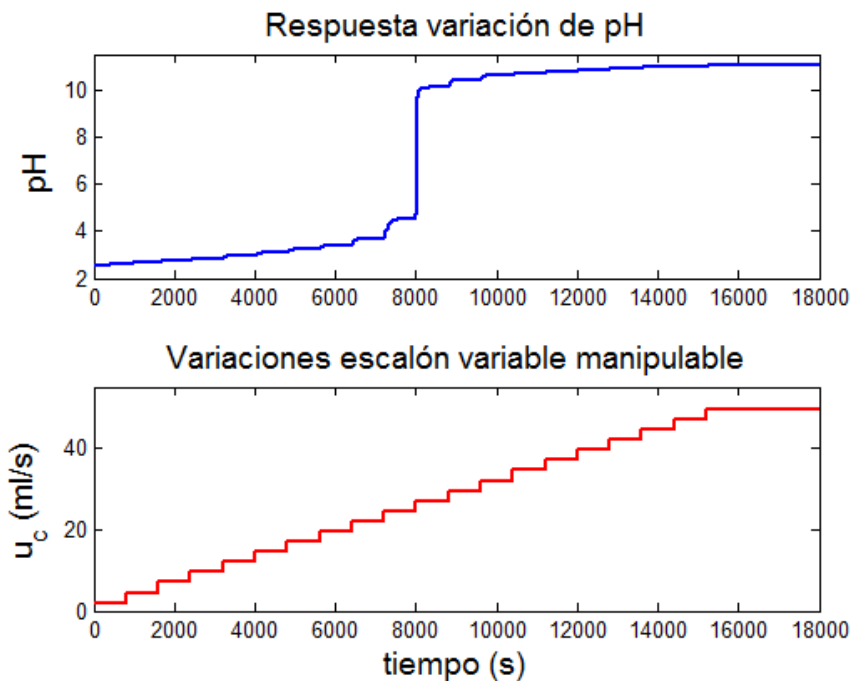


Figura 40. Respuesta ante variaciones escalón de la variable manipulable (caudal de la solución base)

Fuente: Elaboración propia

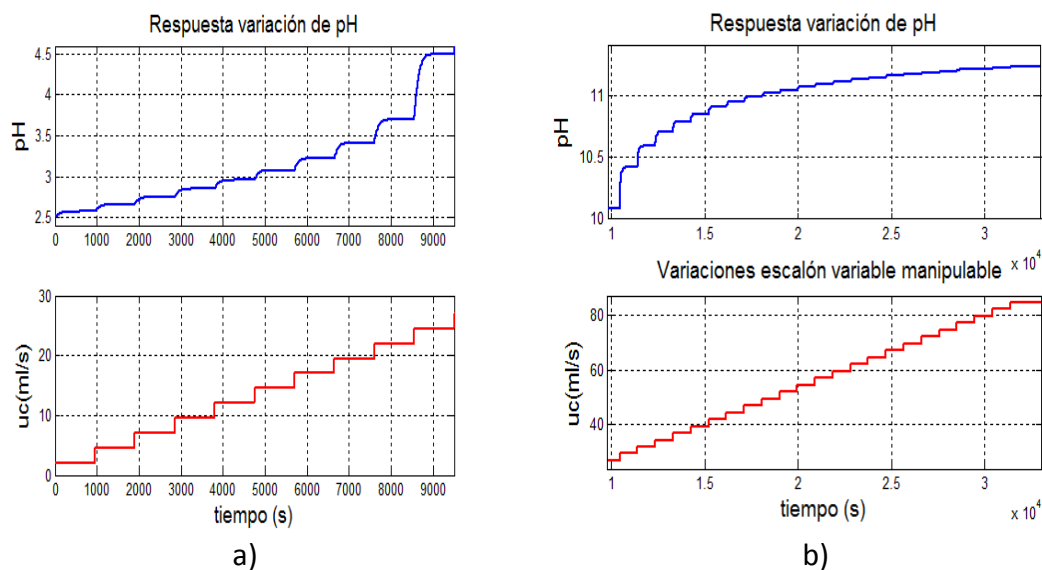


Figura 41. Ampliación respuesta variación de pH: a) $\text{pH} < 5$, b) $10 < \text{pH} < 11$

Fuente: Elaboración propia

El diagrama de simulación para las entradas a la planta se muestra en la Figura 42

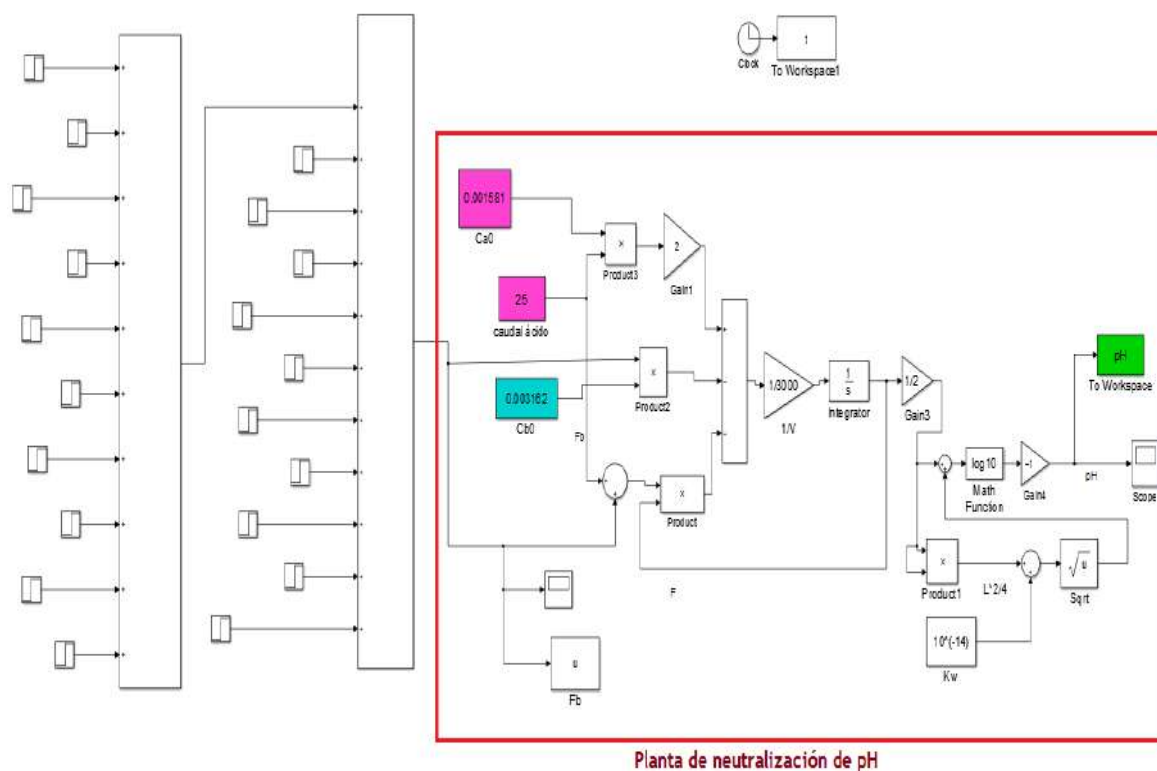


Figura 42. Simulador para entrada variaciones caudal base
Fuente: Elaboración propia

En la Figura 43 se muestra la mejor aproximación polinómica que ajusta los datos, en este caso se obtuvo con un polinomio de orden $n=19$. La característica estática corresponde a considerar los valores estacionarios medidos en las pruebas realizadas ante variaciones del caudal base.

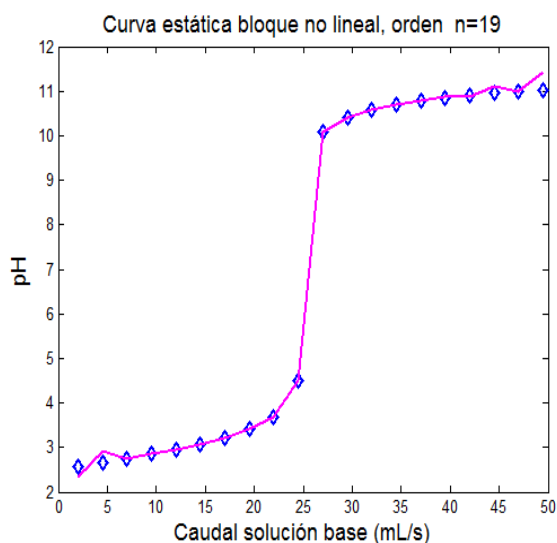


Figura 43. Característica estática de pH
Fuente: Elaboración propia

Como se pudo apreciar hay una alta sensibilidad alrededor del punto neutro, por lo que se aplicaron variaciones de 0.05 mL/s en el caudal de la solución base, a partir de 24.5 mL/s y a partir de pH=4.5 (valor inicial), para determinar los niveles de amplitud que podemos aplicar en la señal usada para identificación. En el caso de la prueba realizada el rango del caudal de base es: $24.5 \leq u_c \leq 25.5$ mL/s y el rango de la salida correspondiente es: $4.5 \leq pH \leq 9.5$.

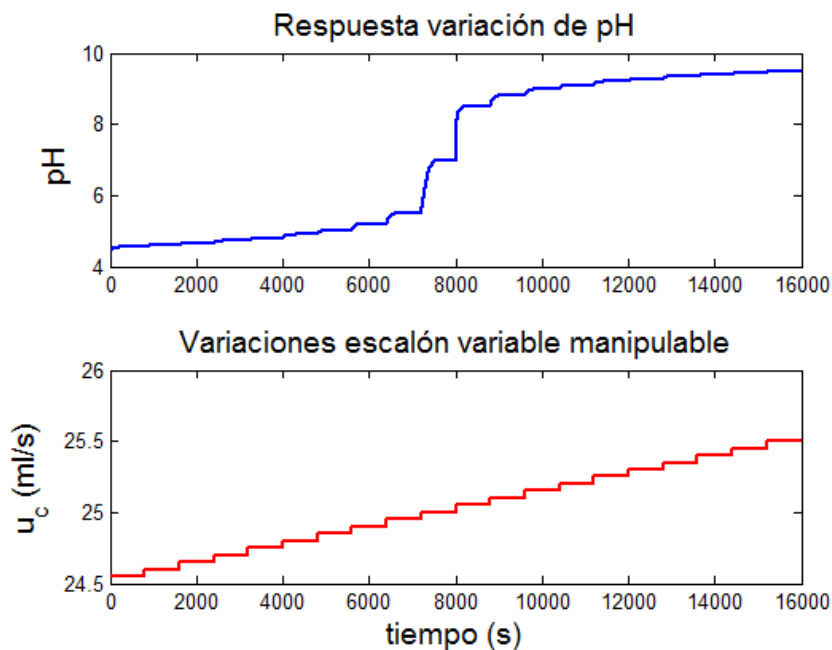


Figura 44. Rango reducido de variación caudal solución base
Fuente: Elaboración propia

El comportamiento estático en ese rango reducido se muestra en la Figura 45.

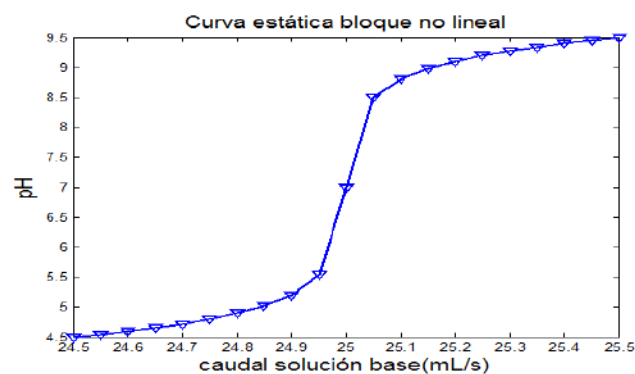


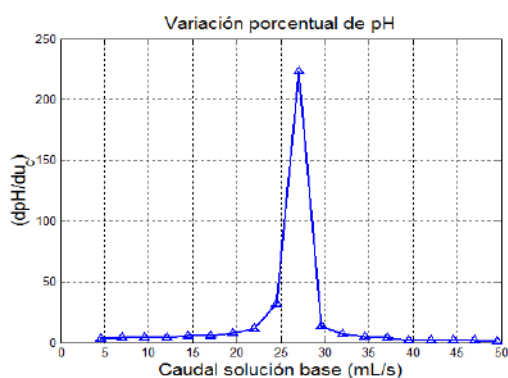
Figura 45. Curva de pH para rango reducido de variación de caudal base alrededor del valor neutro
Fuente: Elaboración propia

A partir de los datos estacionarios en cada rango de variación de caudales obtenidos en la Figura 40, se determinan las ganancias variables que, como se observa de la Tabla 3, evidencian la no linealidad del proceso. Se aprecia además la variación de la constante de tiempo que se reduce en zonas cercanas al valor neutro de pH. Esta característica se debe tener en cuenta al diseñar la señal de entrada para el proceso de identificación.

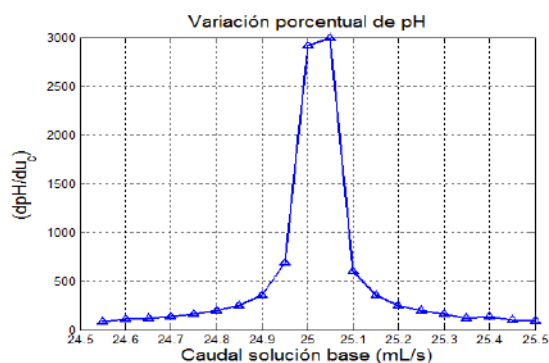
Tabla 3 Ganancia del proceso ante variaciones escalón a la entrada de la planta

Valor estacionario u_c (ml/s)	Valor estacionario pH solución efluente	Ganancia ($\Delta pH / \Delta u_c$) [pH/ml/s]	Constante de tiempo (s)
2	2.57		133.5
4.5	2.658	0.0352	108.7
7	2.75	0.0368	107
9.5	2.85	0.04	101
12	2.95	0.04	75
14.5	3.08	0.052	104
17	3.22	0.056	83
19.5	3.41	0.076	91
22	3.695	0.114	86
24.5	4.496	0.3204	105
27	10.08	2.2336	14
29.5	10.42	0.136	40
32	10.59	0.068	48
34.5	10.7	0.044	40
37	10.8	0.04	40
39.5	10.85	0.02	40
42	10.9	0.02	40
44.5	10.95	0.02	40
47	10.99	0.016	60
49.5	11.02	0.012	50

La variación porcentual tanto para el rango dado por la Tabla 4 como para el rango reducido se muestra en la Figura 46 a) y b) respectivamente, evidenciándose la alta ganancia alrededor del caudal correspondiente al pH neutro del proceso (el caudal del ácido en condiciones de operación es de 25 mL/s, por lo que el mismo caudal de base neutraliza la mezcla). Se aprecia una mayor variación porcentual en la Figura 46 b) que se explica porque las variaciones en el caudal de la solución base son menores que en el caso a) (variaciones de 0.05 mL/s).



a)



b)

Figura 46. Variación porcentual del pH ante variaciones de caudal base a) Rango $2.5 \leq pH \leq 11$, b) Rango $4.5 \leq pH \leq 9.5$

Fuente: Elaboración propia

3.2.1. Ensayos preliminares para la identificación

Para la identificación de los modelos Wiener y Hammerstein usando los métodos descritos en el capítulo 1 y el modelo simplificado de simulación de la planta de pH visto en el capítulo 2, se realizan las pruebas descritas a continuación.

3.2.1.1. Análisis de respuesta a variaciones escalón

Se aplican pequeñas variaciones en el caudal de base alrededor de las condiciones de operación, $u_c = 25 \text{ mL/s}$. con un tiempo de prueba de 500 seg. Los resultados se muestran en la Figura 47 y las características más importantes de la respuesta se analizan en la Tabla 4.

Claramente el proceso es estable y el comportamiento del pH de la solución efluente ante pequeñas variaciones de caudal base respecto al valor nominal (25 mL/s). Por los datos de la Tabla 5 y asumiendo que en la planta no se tendrán variaciones mayores del caudal base se elige el tiempo de muestreo $T_s = 1 \text{ s}$ teniendo en cuenta el criterio dado en 3.1.1.2 b) y que en el rango elegido del proceso la mínima constante de tiempo a lazo abierto es de 10s.

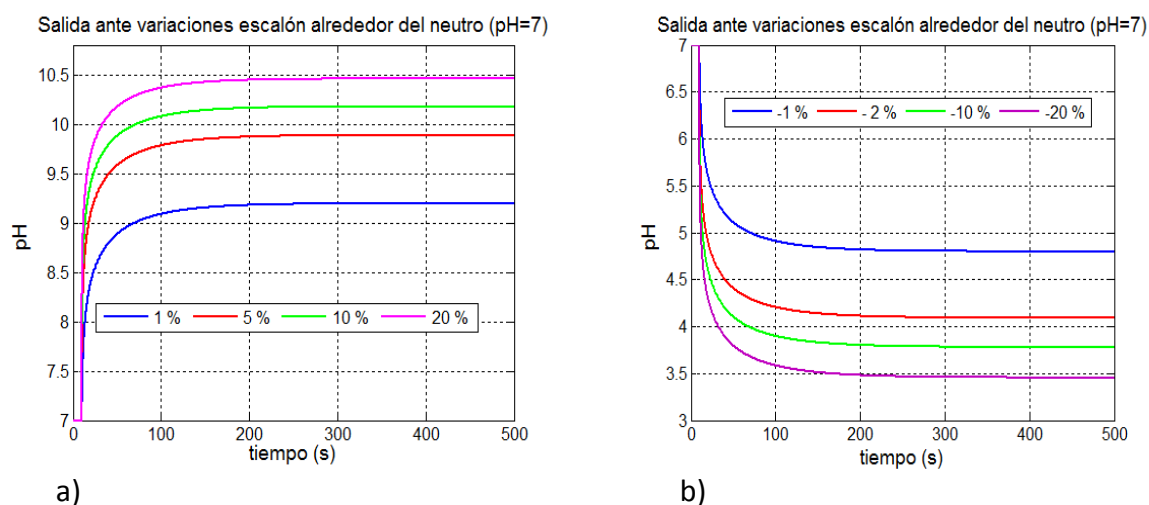


Figura 47. Respuesta ante variación en caudal de la solución base
Fuente: Elaboración propia

Tabla 4 Caracterización de la respuesta a escalón

Variación de caudal base		1%	-1%	5%	-5%	10%	-10%	20%	-20%
Tiempo de crecimiento (95%)	T_c	92	92	77	78.8	69.7	71	56.1	69
Constante de tiempo (s)	τ	10	11	5.21	5.97	3.97	4.07	3.09	3.5

3.2.2. Prueba PRS y obtención de los datos E/S

Una vez elegido el tiempo de muestreo, se aplica a la planta una señal PRS como señal de excitación de la entrada (Figura 48), es decir, se simula variación en la solución de la base en la planta entre dos niveles de entrada $u_{min} = 24.6 \text{ mL/s}$, $u_{max} = 25.4 \text{ mL/s}$, lo que corresponde a variaciones alrededor del 1% del valor nominal. Los datos E/S se muestran en la Figura 49.

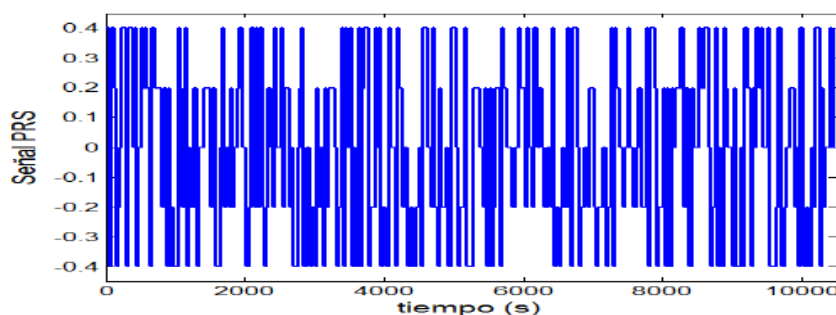


Figura 48. Señal de entrada para prueba de identificación
Fuente: Elaboración propia

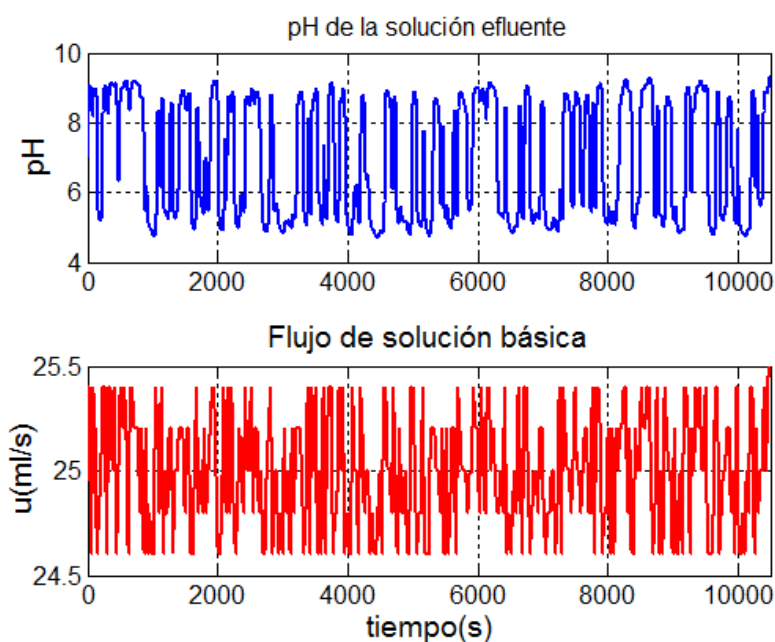


Figura 49. Datos entrada/salida
Fuente: Elaboración propia

La duración del experimento en esta prueba es de 10500s, y los datos E/S se usan para identificación de los modelos orientados a bloques desarrollados en el capítulo 1.

Se ha considerado como rango de estimación los datos desde 1:5000 y el resto de datos se toman para la validación cruzada.

3.3. Identificación lineal

Con los datos de estimación y validación se obtienen las respuestas de la Figura 50 y Figura 51, del modelo lineal encontrado usando el método N4SID del MATLAB. Los modelos encontrados se usarán para diseñar el control predictivo cuyas prestaciones se compararán con el control diseñado usando modelos orientados a bloques.

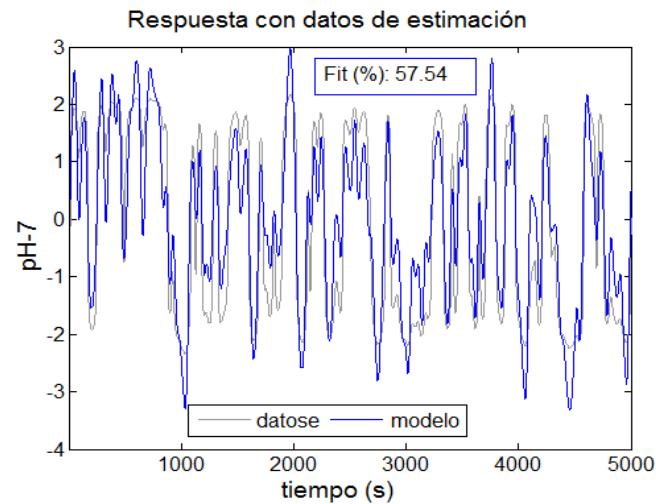


Figura 50. Respuesta a datos estimación del modelo lineal
Fuente: Elaboración propia

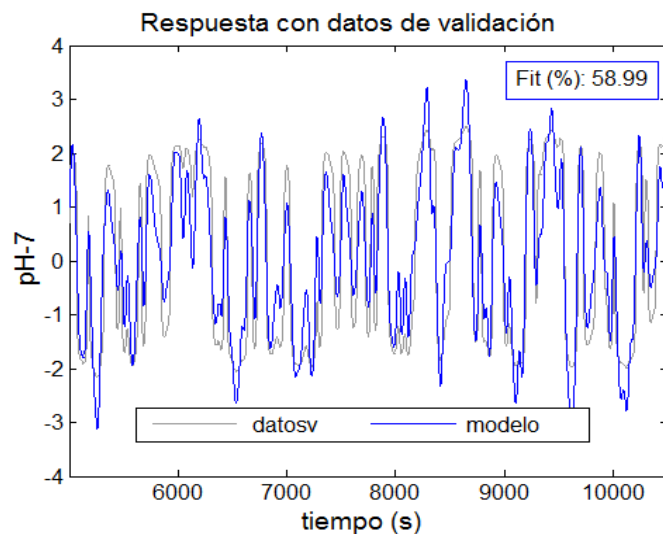


Figura 51. Respuesta a datos de validación del modelo lineal
Fuente: Elaboración propia

El modelo lineal en espacio de estado y en función de transferencias viene dado por:

Matrices en espacio de estado

$$A = \begin{bmatrix} 0.9825 & 1.0663 \\ -0.0019 & 0.7441 \end{bmatrix}, B = \begin{bmatrix} -0.1510 \\ -0.0649 \end{bmatrix}, C = [-0.5867 \quad 0.7327], D=0$$

Función de transferencia

$$\frac{y(z)}{u(z)} = \frac{0.04101z + 0.02163}{z^2 - 1.727z + 0.7331}$$

3.4. Identificación modelos Wiener y Hammerstein

3.4.1. Método N4SID con descomposición SVD

Con los métodos descritos en el capítulo 1 procedemos a buscar el modelo Wiener y Hammerstein para la planta de pH.

Empezando con el modelo Wiener se obtienen los resultados mostrados a continuación.

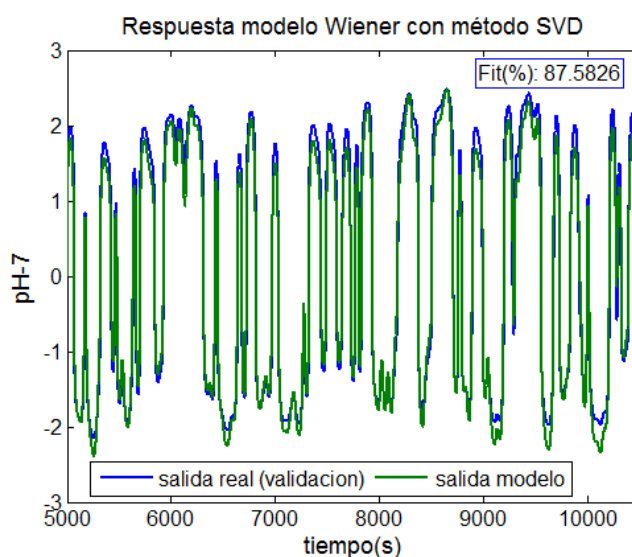


Figura 52. Respuesta modelo Wiener con datos de validación
Fuente: Elaboración propia

La Tabla 5 compara los parámetros de validación entre ambos modelos Wiener y lineal.

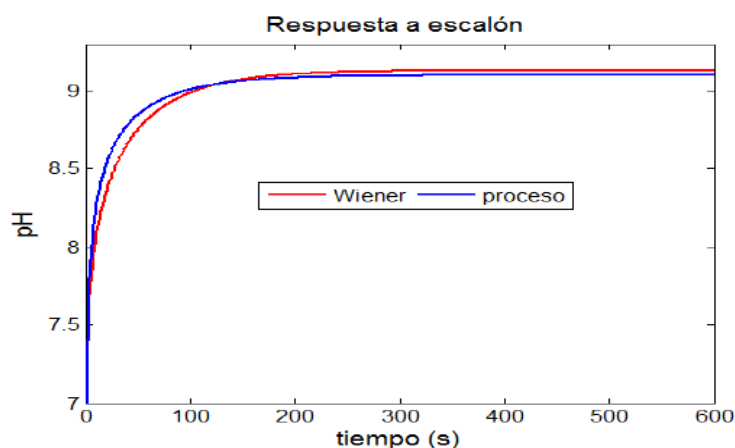
Tabla 5 Índices de *performance* de los modelos Wiener y lineal

	Wiener	Lineal
FIT (%)	87.58	58.99
MSE	0.0444	0.4897

Para analizar el comportamiento del modelo Wiener se aplica una entrada escalón y se compara con el proceso. Se hace notar que la respuesta presenta poco margen de error en el punto elegido y que corresponde a aplicar una entrada a escalón de 0.2 mL/s.

Si se aplican valores mayores o menores se tiene un mayor error estacionario de la salida del modelo comparado con el de la planta real, lo cual se explica porque el proceso es no lineal.

La Figura 54 a) muestra la característica inversa del bloque $\mathcal{N}^{-1}(\cdot)$, hallado directamente con el algoritmo. Para hallar $\mathcal{N}(\cdot)$ se usa otro algoritmo de interpolación (ver Figura 54 b), lográndose buenos resultados de aproximación.



Datos comparativos

Proceso

$$pH_{ss} = 9.1$$

$$\tau = 12 \text{ s}$$

Modelo Wiener

$$\hat{pH}_{ss} = 9.13$$

$$\hat{\tau} = 19 \text{ s}$$

Figura 53. Respuesta a escalón del modelo Wiener y proceso

Fuente: Elaboración propia

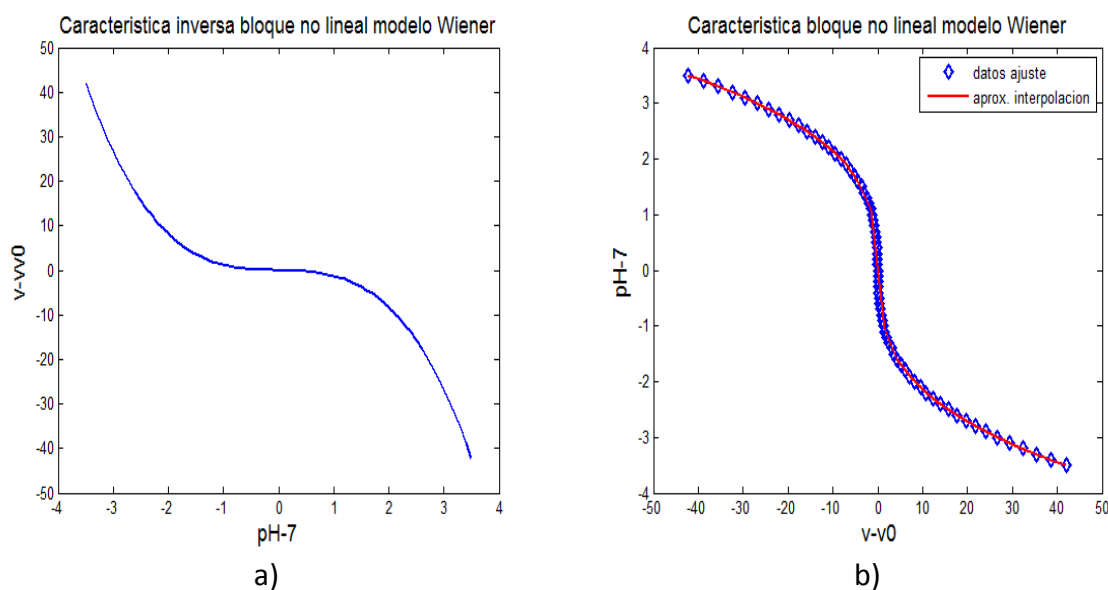


Figura 54. Característica del bloque no lineal del modelo Wiener : a) Inversa del bloque no lineal $\mathcal{N}^{-1}(\cdot)$, b) Bloque no lineal $\mathcal{N}(\cdot)$

Fuente: Elaboración propia

Aplicando los mismos datos E/S para estimación y validación usados para obtener el modelo Wiener se encuentra un modelo Hammerstein con un FIT relativamente bajo. Se puede deducir que el método usado para hallar el modelo Hammerstein para este proceso no es una buena alternativa de aproximación.

Se debe tener en cuenta que siguiendo el criterio dado en 3.1.1, los niveles de la señal de entrada aplicada para identificación deben ser mayores o iguales que el grado del polinomio de aproximación de la componente no lineal. Por los ensayos realizados en el caso del modelo Hammerstein, se ha debido recurrir a una aproximación de orden 7 del polinomio, lo que indica que se debe usar una señal de entrada con más niveles de los usados en nuestro caso (número de niveles= 5).

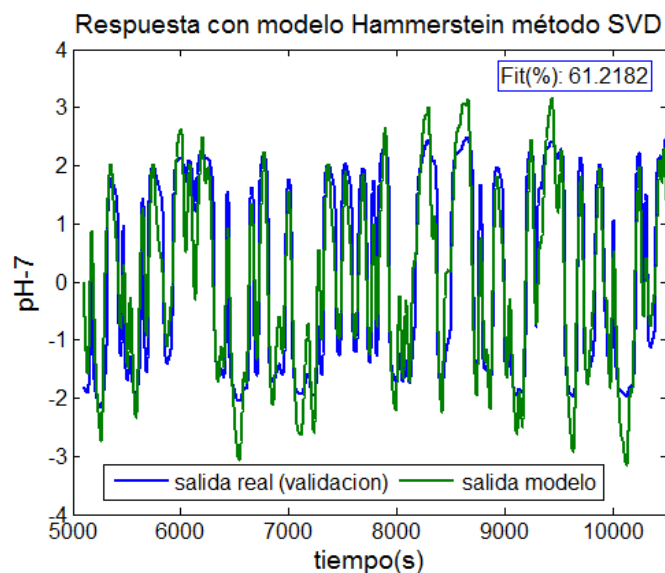


Figura 55. Respuesta modelo Hammerstein a datos de validación
Fuente: Elaboración propia

La característica del bloque estático no lineal y su inversa se presenta a continuación.

Una explicación que se pueda dar a estos resultados es que de partida el simulador de la planta de pH está estructurado más como un modelo Wiener, esto es, primero se incorporan los bloques de las dinámicas lineales y por último la componente no lineal de deducción/ cálculo de pH.

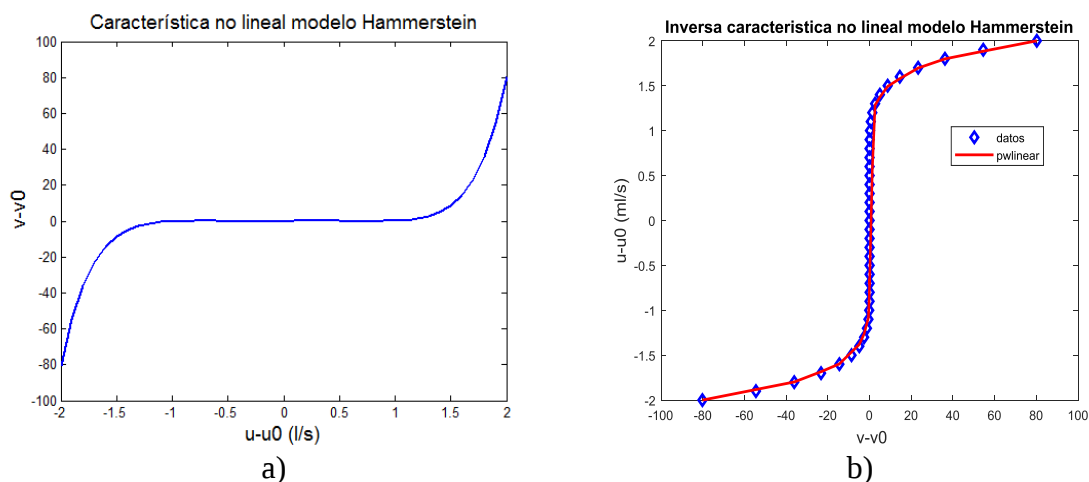


Figura 56. Características no lineal del modelo Hamerstein

De la Tabla 6 se observa que el modelo Hammerstein es solo ligeramente mejor que el modelo lineal.

Tabla 6. Índice de performance de los modelos Hammerstein y lineal

	Hammerstein	Lineal
FIT (%)	61.22	58.99
MSE	0.4332	0.4897

La respuesta a escalón para una variación en el caudal base de 0.15 mL/s, muestra que hay error estacionario y conforme se aumenta la variación de la entrada, mayor es dicho error.

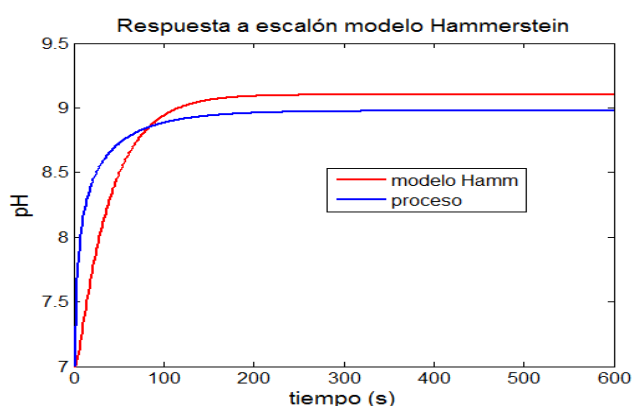


Figura 57.. Respuesta del modelo Hammerstein y del proceso

Fuente: Elaboración propia

Datos comparativos

Proceso

$$pH_{ss} = 8.976$$

$$\tau = 13.5 \text{ s}$$

Modelo Hammerstein

$$\hat{p}H_{ss} = 9.1$$

$$\hat{\tau} = 41 \text{ s}$$

En la Tabla 7 se muestran las características principales de los bloques lineal y no lineal de los modelos Wiener y Hammerstein encontrados con el método de descomposición SVD y N4SID antes descrito.

Tabla 7. Modelos Wiener y Hammerstein método N4SID con SVD

Modelo Wiener	Modelo Hammerstein
Bloque lineal Matrices en espacio de estado $A = \begin{bmatrix} 0.9831 & -0.0028 \\ -0.0021 & 0.9988 \end{bmatrix}$, $D=0$ $B = \begin{bmatrix} -51.4429 \\ -0.1951 \end{bmatrix}$, $C = [0.5806 \quad 0.0887]$ Función de transferencia $\frac{v(z)}{u(z)} = \frac{0.8551z + 0.8544}{z^2 - 1.982z + 0.9819}$	Bloque lineal Matrices en espacio de estado $A = \begin{bmatrix} 0.9837 & 0.0578 \\ -0.0434 & 0.7316 \end{bmatrix}$, $D=0$ $B = \begin{bmatrix} -0.0077 \\ 0.6538 \end{bmatrix}$, $C = [95.7933 \quad 2.6012]$, Función de transferencia $\frac{y(z)}{v(z)} = \frac{0.9586z + 2.489}{z^2 - 1.715z + 0.7222}$
Bloque no lineal $N^{-1}(y_k) = v_k = -0.2764y_k - 0.0055y_k^2 - 0.961y_k^3$	Bloque no lineal $N(u_k) = v = 0.0283u_k - 0.6228u_k^5 + 0.7819u_k^7$

3.4.2. Método con bases ortonormales y descomposición SVD

Para la identificación usando bases ortonormales se aplican los métodos descritos en 0.

Basándonos en el resultado de los modelos encontrados con el método anterior se considera una base ortonormal de segundo orden, tanto para el modelo Hammerstein como para el modelo Wiener, usando como polos de las bases los del sistema lineal encontrado usando el método N4SID. Los polos para la base ortonormal de segundo orden en este caso son: $[0.98 \ 0.9]$. Se han usado los mismos datos E/S y los rangos de estimación y validación usados para identificar los modelos en el apartado anterior.

Para el modelo Wiener se ha conseguido mejores resultados considerando un polinomio para la inversa del bloque no lineal con términos en y, y^2, y^5 . Los resultados se muestran a continuación. La característica del modelo lineal para los datos de validación son las mismas representadas en la Figura 51.

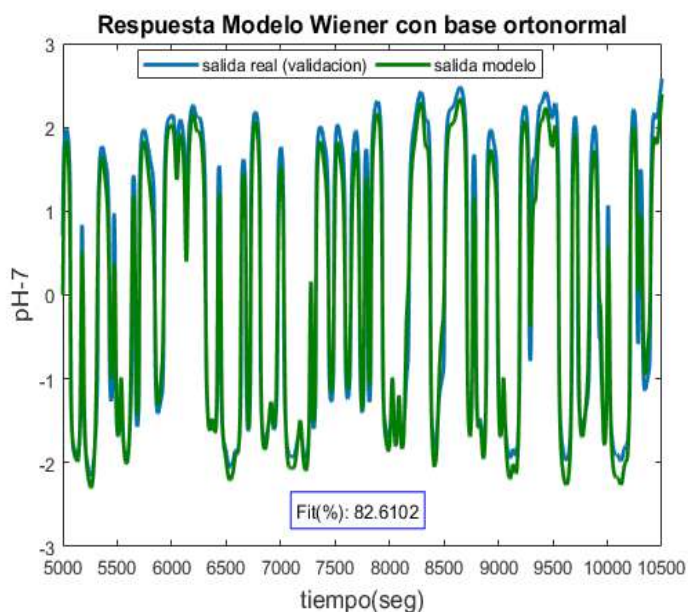


Figura 58. Respuesta modelo Wiener en bases ortonormales
Fuente: Elaboración propia

Para los mismos datos E/S, datos de estimación y validación usados para identificar el modelo Wiener se obtiene un modelo Hammerstein con no muy buenas prestaciones, como se observa de la Figura 59 .

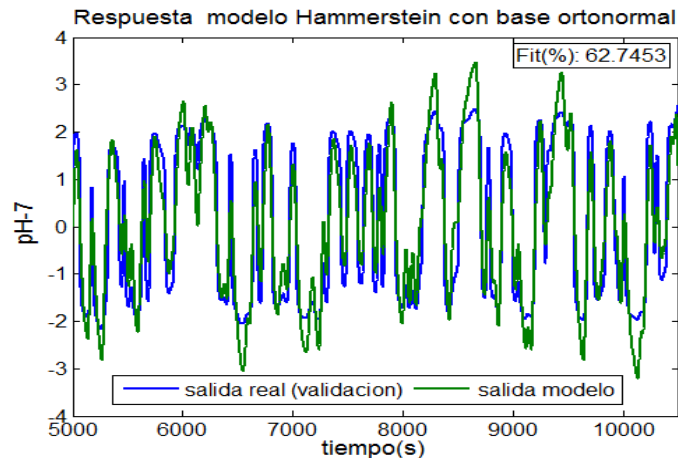


Figura 59. Respuesta modelo Hammerstein en bases ortonormales de Segundo orden
Fuente: Elaboración propia

Las características de validación del modelo Wiener comparado con el modelo lineal visto y con el modelo Hammerstein reportados en la Tabla 8 indican que es el mejor modelo del proceso.

Tabla 8. Índices de *performance* de modelos Wiener y Hammerstein con bases ortonormales

	Wiener	Hammerstein	Lineal
FIT(%)	82.61	62.745	58.97
MSE	0.0872	0.4	0.4904

En ambos casos, dado que se han elegido los mismos polos se obtienen las mismas bases ortonormales. Así mismo, la matriz A del modelo en espacio de estado del bloque lineal es la misma. Las entradas y salidas serán diferentes pues el bloque lineal en el caso del modelo Wiener antecede al bloque no lineal y en el modelo Hammerstein se invierte el orden.

A continuación se resumen los elementos: bloque lineal y no lineal de los modelos Wiener y Hammerstein. En la Figura 60 se observa el comportamiento de los elementos no lineales de ambos modelos.

Tabla 9. Modelos Wiener y Hammerstein con bases ortonormales

Modelo Wiener	Modelo Hammerstein
Bloque lineal Polos: 0.98, 0.81 Matrices en espacio de estado $A = \begin{bmatrix} 0.99 & 0 \\ 0.2062 & 0.81 \end{bmatrix},$ $B = [1 \ 0]', C = [0.3386 \ 0.1911], D = 0$ Bases ortonormales $B_o(z) = \frac{0.199}{z-0.98}, \quad B_1(z) = \frac{-0.5747z+0.5864}{z^2-1.79z+0.7938}$ Coeficientes $b = [2.6427 \ 0.3259]$	Bloque lineal Polos: 0.98, 0.81 Matrices en espacio de estado $A = \begin{bmatrix} 0.99 & 0 \\ 0.2062 & 0.81 \end{bmatrix},$ $B = [1 \ 0]', C = [-0.1936 \ -0.8412],$ $D = 0$ Bases ortonormales $B_o(z) = \frac{0.199}{z-0.98}, \quad B_1(z) = \frac{-0.5747z+0.5864}{z^2-1.79z+0.7938}$ Coeficientes $b = [-5.1154 \ -1.4345]$
No lineal $N^{-1}(y) = y_k + 0.0302y_k^2 + 0.0894y_k^5$	No lineal $N(u) = -0.3163u_k - 0.0066u_k^2 + 0.9486u_k^3$

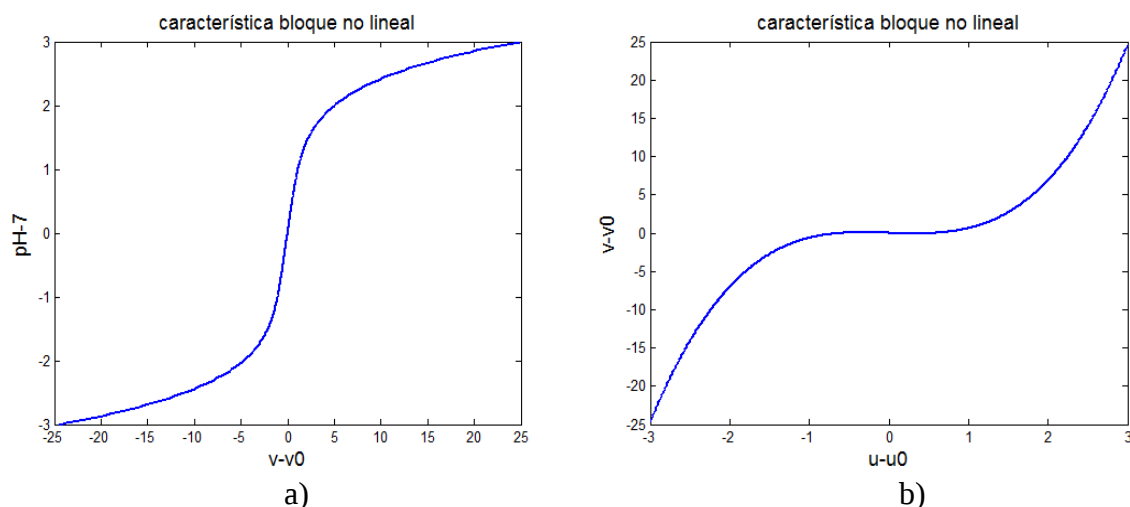


Figura 60. Característica no lineal: a) modelo Wiener, b) Modelo Hammerstein
Fuente: Elaboración propia

3.4.3. Modelos Wiener y Hammerstein usando instrucción nlhw de MATLAB

Es posible encontrar los modelos Wiener y Hammerstein usando la instrucción *nlhw* para identificar los modelos orientados a bloques. La instrucción permite identificar los modelos Wiener, Hammerstein y Hammerstein Wiener, cuya sintaxis general es:

`model = nlhw(Data, Orders, InputNonlinearity, OutputNonlinearity)`

donde:

- *Orders*: especifica un vector $[n_a \ n_b \ n_k]$ que contiene el orden de la salida, de la entrada y los retardos de la función de transferencia lineal
- *InputNonlinearity* y *OutputNonlinearity*: Especifican el tipo de no linealidad de los bloques a la entrada y la salida del modelo.

Si se desea especificar un modelo Hammerstein el bloque de salida no lineal se especifica como `[]` y viceversa en el caso del bloque de entrada no lineal si se desea un modelo Wiener.

Los elementos que se pueden especificar en el argumento *InputNonlinearity* o *OutputNonlinearity*, son

<i>pwlinear</i>	función <i>piecewise linear</i>
<i>poly1d</i>	polinomio unidimensional
<i>deadzone</i>	bloque zona muerta
<i>saturation</i>	función de saturación
<i>sigmoidnet</i>	función <i>sigmoide</i>
<i>wavenet</i>	red <i>wavelet</i>
<i>unitgain</i>	bloque ganancia unitaria
<i>customnet</i>	definida por el usuario

A continuación se muestran los resultados comparando con distintos bloques no lineales (los que dieron mejores resultados para los datos E/S) aplicados a la estructura de un modelo Hammerstein y Wiener. Las características para cada uno de los modelos que se identifican y que arrojaron mejores resultados son:

- **Modelo m1:** Hammerstein con orden bloque lineal: [2 3 1] y no lineal del tipo pwlinear con diez puntos de quiebre
- **Modelo m2:** Hammerstein con orden bloque lineal [3 2 0] y bloque no lineal del tipo sigmoide.
- **Modelo m3:** Wiener con bloque lineal y no lineal como en m1.
- **Modelo m4:** Wiener con orden bloque lineal [2 3 1] y no lineal del tipo sigmoide.

Los elementos que pueden usarse para el bloque no lineal son: *sigmoidnet*, *wavenet*, *treepartition*, *customnet*, *neuralnet*, *linear*.

Comparando todos los modelos encontrados se observa de la Figura 61 y Figura 62 que el modelo con mejor *FIT* es el modelo Wiener.

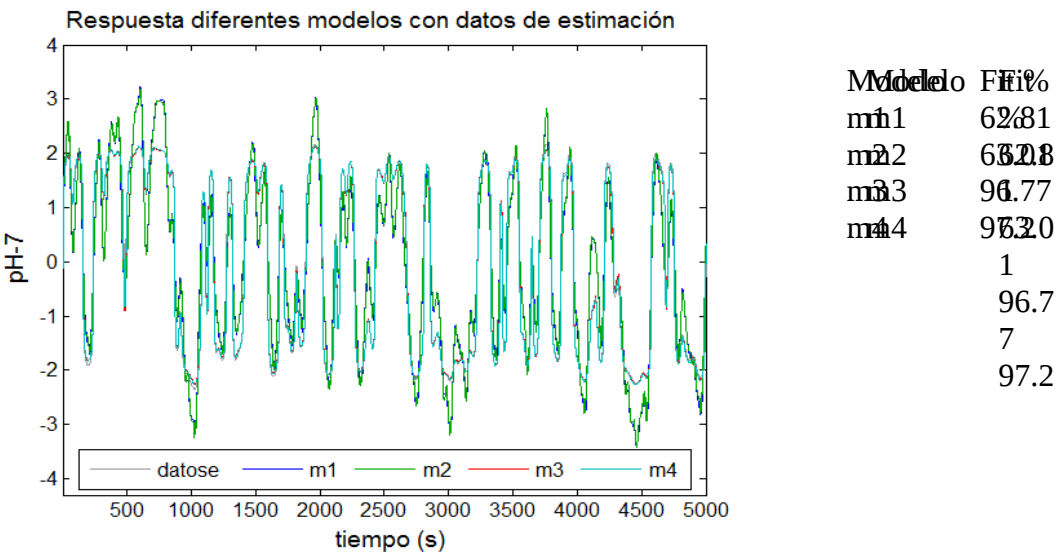


Figura 61. Comparación modelos con datos de estimación
Fuente: Elaboración propia

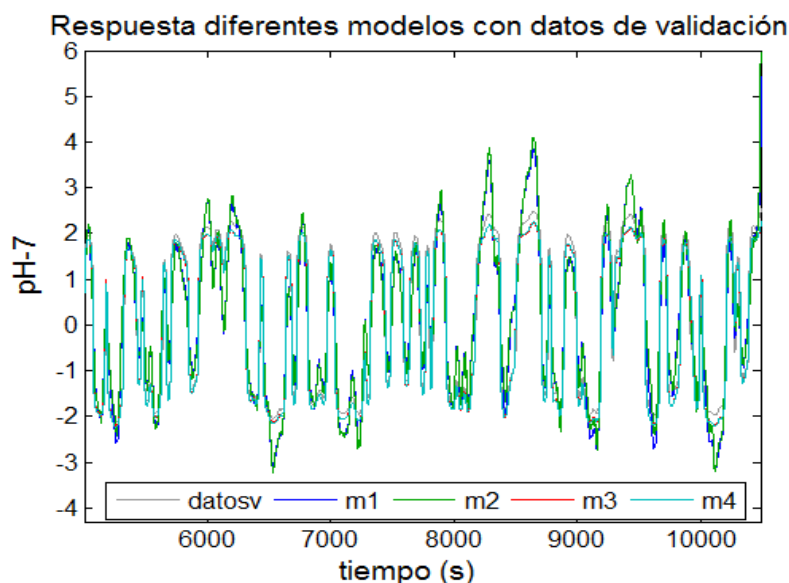


Figura 62. Comparación modelos con datos de validación
Fuente: Elaboración propia

De los modelos Wiener encontrados se elige el modelo con bloque no lineal de tipo *pwlinear* para comparar la respuesta a una entrada escalón de dicho modelo con el modelo encontrado con el método SVD, que es el que usaremos para el control predictivo. La respuesta a escalón del modelo **m3** con función no lineal del tipo *pwlinear* se analiza en la Figura 63 que se muestra a continuación.

De los datos de la tabla se puede ver que el ajuste encontrado es muy bueno pues el error estacionario y la constante de tiempo son prácticamente iguales, para el análisis realizado aplicando la misma variación de caudal base a la entrada que se usó para analizar la respuesta a escalón de los modelos previamente encontrados.

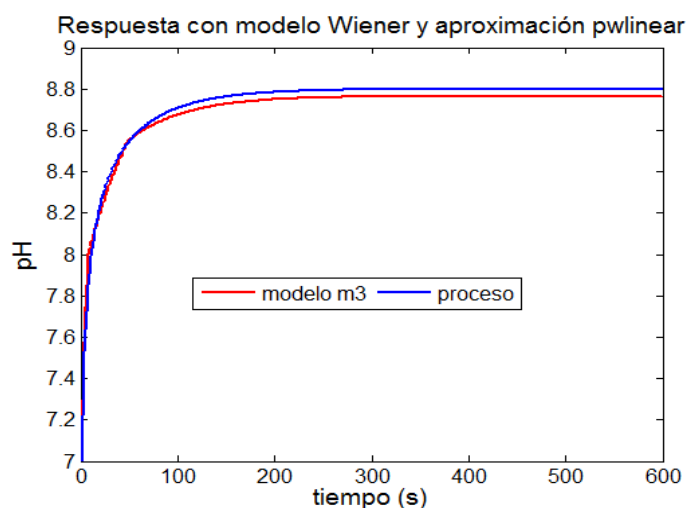


Figura 63. Respuesta a escalón para una variación de 0.1 mL/s del caudal base
Fuente: Elaboración propia

Datos comparativos

Proceso

$$pH_{ss} = 8.8$$

$$\tau = 15 \text{ s}$$

Modelo Hammerstein

$$\widehat{pH}_{ss} = 8.766$$

$$\hat{\tau} = 15 \text{ s}$$

En la ampliación que se muestra de los primeros 100s. se puede observar que el transitorio no es muy bueno; pero se puede mejorar considerando más puntos de

quiebre en la función. No obstante por los datos antes vistos, se puede usar el modelo encontrado por ser más simple (cuanto más puntos de quiebre se usen mayor tiempo de cálculo se requiere) y porque da un comportamiento similar tanto en transitorio como en estacionario como ya se ha mencionado (tiene la misma constante de tiempo que la del proceso y el error estacionario es de -0.39%).

Un mejor comportamiento se encuentra con el modelo Wiener **m₄** que usa como bloque no lineal una función sigmoide, de la forma: $f(x) = \frac{1}{e^{-x}+1}$. Sin embargo, este modelo resulta complejo para la aplicación prevista en esta tesis. Como se observa de la Figura 64 b) el comportamiento estacionario del modelo es mejor que en el caso del modelo Wiener con *pwlinear*, ya que presenta la misma constante de tiempo que el proceso y error estacionario nulo.

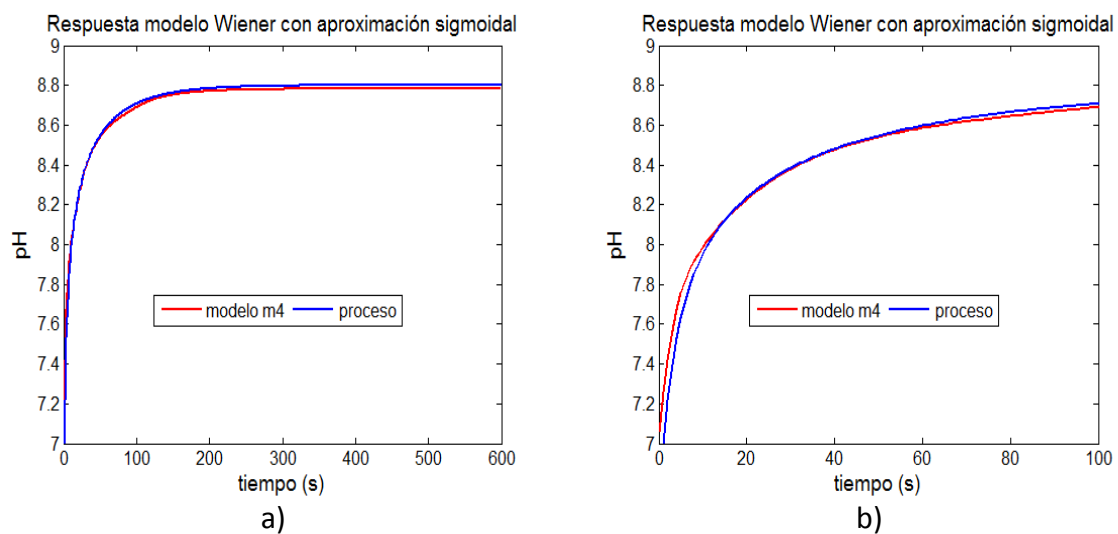


Figura 64. Respuesta del modelo **m₄** con variación de 0.1 mL/s en el caudal base
Fuente: Elaboración propia

En la Figura 65 se aprecia el comportamiento del modelo Wiener encontrado con bloque no lineal *pwlinear* para los datos de estimación y validación. Los resultados muestran un ajuste del 89% , decir muy aceptable el modelo obtenido.

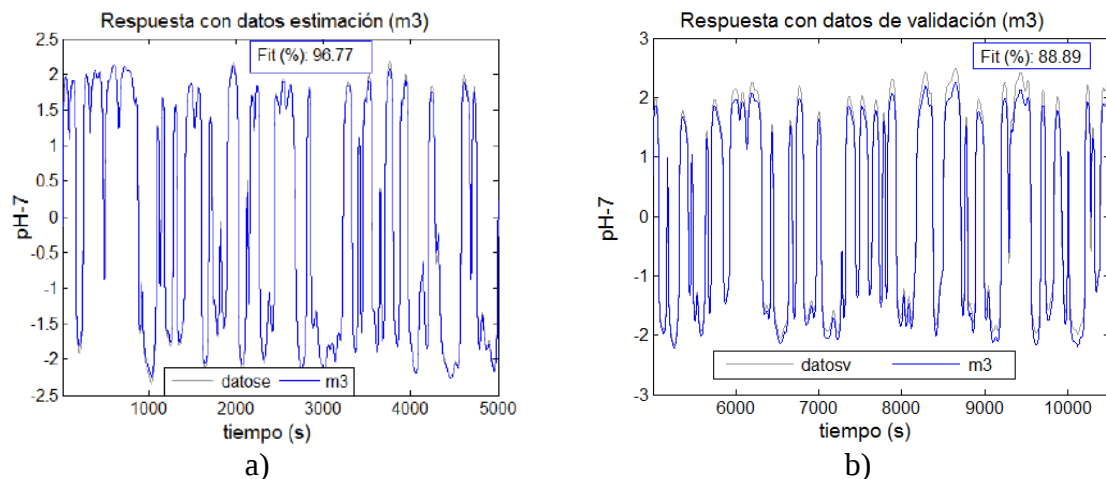


Figura 65. Respuesta del modelo **m₃** con a) datos de estimación, b) datos de validación
Fuente: Elaboración propia

Evaluable la practicidad para la implementación el modelo obtenido con N4SID con descomposición SVD es más adecuado, por la simplicidad del bloque no lineal (aproximación polinómica), mientras que el algoritmo de Matlab *nlhw* proporciona un bloque no lineal *pwlinear* que para el ajuste encontrado tiene un alto número de puntos de quiebre y por tanto varios segmentos, lo que dificulta su implementación e incrementa el esfuerzo en el cálculo de la ganancia. Además los ajustes son similares, pues como se pudo observar de la figura 52, la aproximación con el método propuesto da un FIT de 87.58% frente a 88.89 % del método de aproximación con el algoritmo *nlhw*, por lo que no hay excesiva diferencia, que amerite el uso de los resultados encontrados con este último modelo.

En este capítulo se ha evidenciado que la identificación no lineal requiere un esfuerzo adicional desde el punto de vista teórico y práctico, no obstante los resultados comparados con la identificación lineal muestran que vale la pena su implementación. Además como se verá en el capítulo siguiente, es importante un adecuado modelo que represente de la mejor manera el comportamiento dinámico del proceso para un control más eficiente.

Capítulo 4

Controlador PID predictivo para pH

Los algoritmos más ampliamente usados en la industria son los de control PID. Los operadores de procesos industriales tienen muy buena experiencia en sintonizar este tipo de controladores. Se pueden usar diferentes reglas de sintonización muchas de ellas basadas en la aproximación del sistema con un modelo de primer orden con tiempo muerto. Aunque las reglas no siempre son la mejor forma de lograr una buena sintonización, logran resultados empíricos, en buena parte de los casos aceptables.

Hay algunas desventajas de un control PID enfocadas principalmente a ciertos procesos, tal es el caso de sistemas con tiempo muerto especialmente grandes en el que el control es muy lento y el caso de sistemas de fase no mínima su sintonización resulta muy compleja. Hay alternativas en las que se puede superar esta limitación, tales como los predictores de Smith y el control *deadbeat*; pero su uso industrial no es muy aceptado. El control predictivo, basado en un modelo, supera varios de estos inconvenientes (Silva, Henrique, & Oliveira-Lopes, 2012), (Norquay, Palazoglu, & Romagnoli, 1998), (Long, y otros, 2015), (Clarke, Mothadi, & Tuffs, 1987).

La funcionalidad integral de un controlador PID se puede complementar con oportunas adaptaciones, a un esquema de control predictivo basado en modelos (MPC), con la enorme ventaja de poder seguir usando las plataformas hardware ampliamente usadas en la industria. La idea de diseñar un control PID usando un algoritmo MPC que permita usar la estructura sencilla de un PID combinada con las características de comportamiento de un controlador predictivo ha sido estudiada en la literatura (Kokate & Waghmare, 2011), (Miller, Shah, Wood, & Kwok, 1999), (Uren & Van Schoor, 2011) (Saeed, Uddin, & Katebi, 2013), (Kawabe, 2012), (Qin, Hu, & Song, 2012), (Wu, 2015), (Zhang, Wu, Lu, & Gao, 2014) e implementada en algunos sectores industriales, obteniéndose muy buenos resultados.

Las desventajas de un PID pueden ahora superarse aprovechando las características de los controladores predictivos, en los que el control para sistemas con tiempo muerto es más rápido y resulta más simple la sintonización de controles para sistemas de fase no mínima.

En este capítulo se presenta el desarrollo de un control PID predictivo, para el cual se ha usado un modelo Wiener. También se ha implementado una estructura Harmmeststein evidenciando que la estructura del proceso a modelar deber ser coherente con el modelo de predicción usado. En este caso, la mejor alternativa ha sido la estructura usando el modelo Wiener.

Se muestran también los resultados comparados con regulador predictivo lineal. La conclusión principal es la bondad conseguida con el controlador predictivo PID usando un modelo Wiener.

4.1. Controlador MPC predictivo con estructura en espacio de estado

El objetivo de diseño de un control predictivo basado en modelos (MPC) es calcular la trayectoria de una variable manipulada futura u para optimizar el comportamiento futuro en base a una función de costo. La optimización se realiza en una ventana de tiempo limitada, denominado horizontes de predicción y/o de control (ver Figura 66), cuya solución puede ser determinística o iterativa, según las restricciones que se establezcan o no.

El principio del control predictivo requiere de los siguientes términos (Wang, 2009):

- **Ventana de horizonte móvil:** la ventana dependiente del tiempo desde un instante arbitrario t_i a $t_i + N$. La longitud de la ventana N permanece constante.
- **Horizonte de predicción (N_p):** indica que tan lejos en el futuro se desea predecir. Este parámetro es igual a la longitud de la ventana de horizonte móvil, $N_p = N$.
- **Horizonte de control recesivo (N_c):** aunque la trayectoria óptima de la señal de control futuro está completamente descrita dentro de la ventana de horizonte móvil, la entrada de control actual a la planta sólo toma la primera muestra de la señal de control, mientras desprecia el resto de la trayectoria.
- **Función objetivo:** para tomar una buena decisión, se necesita de un criterio para reflejar el objetivo. El objetivo está relacionado con una función de error basada en la diferencia entre las respuestas actual y deseada. Esta función objetivo es a menudo llamada función de costo J y la acción de control se encuentra minimizando esta función de costo dentro de la ventana de optimización.

Como se describe en (Camacho & Bordons, 1999), la metodología de los controles predictivos consiste básicamente en los siguientes pasos:

1. Las salidas futuras para el horizonte N son predichas en cada instante t . Estas salidas $y(t + k|t)$ para $k = 1 \dots N$ dependen de los valores conocidos en t (esto es, entradas y salidas pasadas) y de las señales de control futuras $u(t + k|t)$ en $k = 1 \dots N - 1$ que se envían al sistema para ser calculadas.
2. El conjunto de señales de control futuras se calculan optimizando un criterio J , función de costo o índice de prestaciones, que permita mantener el proceso lo más cerca posible de la trayectoria de referencia $w(t + k)$ (puede ser la entrada de referencia o una aproximación). El criterio usado por lo general es una función cuadrática de los errores entre la salida predicha y la trayectoria de referencia. Además en muchos casos se incluye un término en función del esfuerzo de control. Si no hay restricciones se determina la ley de control resolviendo el criterio de minimización; pero si hay restricciones se debe usar un método de optimización iterativo.
3. Por último, la señal de control $u(t|t)$ hallada es enviada al proceso. Los demás valores son descartados pues en el próximo instante de muestreo se repite el proceso y se obtiene un nuevo valor de la señal de control para ese instante.

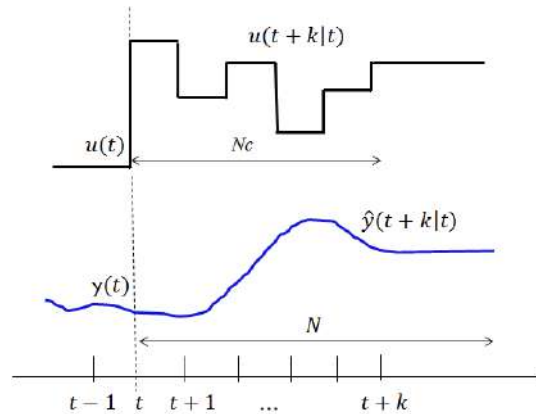


Figura 66. Estrategia de control predictivo basado en modelo
Fuente: Elaboración propia

Como es de suponer esta estrategia requiere de un modelo que sea capaz de reproducir la dinámica del proceso. En las formulaciones iniciales del control predictivo basado en modelos, los modelos de respuesta impulsiva finita (FIR) y respuesta a escalón (FSR) fueron ampliamente usados. Entre los algoritmos basados en dichos modelos se encuentran el control de matriz dinámica (DMC) y la formulación DMC cuadrática. Los modelos tipo FIR son especialmente atractivos para los ingenieros de procesos pues la estructura del modelo da una descripción transparente del retardo del proceso, tiempo de respuesta y la ganancia. Sin embargo, están limitados a plantas estables y a menudo requieren de una gran cantidad de parámetros. Los modelos de función de transferencia dan una descripción simple de la dinámica del proceso y son aplicables tanto a sistemas estables como inestables. Un ejemplo de esta estructura es la de los controles predictivos generalizados GPC (Generalized Predictive Control), sin embargo, esta estructura es considerada menos efectiva para procesos multivariables. Otra estructura utiliza la formulación del modelo MPC en espacio de estado, que facilita la aplicación a sistemas multivariables. Esta es la que se usará en la implementación del algoritmo PID de características predictivas.

4.1.1. MPC con modelo en espacio de estado.

El control MPC es uno de los controladores más exitosos en la industria de procesos y describe una clase de algoritmos de control que controla el comportamiento futuro de una planta a través del uso de un modelo del proceso explícito. Hasta hace poco las aplicaciones se orientaban sólo a modelos lineales pero sus características son aceptables cuando el proceso opera alrededor de unas condiciones de trabajo.

En el caso del modelo MPC en espacio de estado que se implementaría se involucra la acción integral en el modelo a espacio de estado, como se describe en (Uren & Van Schoor, 2011) y (Wang, 2009) partiendo de la descripción de la planta en espacio de estado:

$$\begin{aligned} \mathbf{X}_m(k+1) &= \mathbf{A}_m \mathbf{X}_m(k) + \mathbf{B}_m u(k) \\ y(k) &= \mathbf{C}_m \mathbf{X}_m(k) + D_m u(k) \end{aligned} \quad (4.1)$$

donde: $u(k)$ es la variable de entrada, $y(k)$ es la variable de salida y $\mathbf{X}_m(k)$ es el vector de la variable de estado de dimensión n . Dado que se debe diseñar un control predictivo, que permita incluir la acción integradora, se modifica la expresión de manera que se incluya el incremento $\Delta u(k)$. En una formulación con el principio del horizonte recesivo se asume que la entrada $u(k)$ no puede afectar la salida $y(k)$ al mismo tiempo, por lo que se asume $D_m = 0$.

Haciendo la operación diferencia en ambos miembros de la expresión (4.1) con esta simplificación, se tiene:

$$\underbrace{\mathbf{X}_m(k+1) - \mathbf{X}_m(k)}_{\Delta \mathbf{X}_m(k+1)} = \mathbf{A}_m \underbrace{(\mathbf{X}_m(k) - \mathbf{X}_m(k-1))}_{\Delta \mathbf{X}_m(k)} + \mathbf{B}_m \underbrace{(u(k) - u(k-1))}_{\Delta u(k)} \quad (4.2)$$

de esta manera se consigue el efecto integrador conectando $\Delta \mathbf{X}_m(k)$ a la salida $y(k)$

El nuevo vector aumentado quedaría como sigue:

$$\mathbf{X}(k) = [\Delta \mathbf{X}_m(k)^T \ y(k)^T]^T \quad (4.3)$$

Haciendo la misma operación diferencia en la ecuación de salida, se puede escribir:

$$\begin{aligned} (y(k+1) - y(k)) &= \mathbf{C}_m \underbrace{(\mathbf{X}_m(k+1) - \mathbf{X}_m(k))}_{\Delta \mathbf{X}_m(k+1)} \\ &= \mathbf{C}_m \mathbf{A}_m \Delta \mathbf{X}_m(k) + \mathbf{C}_m \mathbf{B}_m \Delta u(k) \end{aligned} \quad (4.4)$$

Con (4.2) y (4.4) se plantea la nueva forma de espacio de estado:

$$\underbrace{\begin{bmatrix} \Delta \mathbf{X}_m(k+1) \\ y(k+1) \end{bmatrix}}_{\mathbf{X}(k+1)} = \underbrace{\begin{bmatrix} \mathbf{A}_m & \mathbf{0}_m^T \\ \mathbf{C}_m & 1 \end{bmatrix}}_A \underbrace{\begin{bmatrix} \Delta \mathbf{X}_m(k) \\ y(k) \end{bmatrix}}_{\mathbf{X}(k)} + \underbrace{\begin{bmatrix} \mathbf{B}_m \\ \mathbf{C}_m \mathbf{B}_m \end{bmatrix}}_B \Delta u(k) \quad (4.5)$$

$$y(k) = \underbrace{[\mathbf{0}_m \ 1]}_C \begin{bmatrix} \Delta \mathbf{X}_m(k) \\ y(k) \end{bmatrix} \quad (4.6)$$

donde $\mathbf{0}_m = [0 \ 0 \ \dots \ 0]$ es un vector $1 \times n$. La (4.5) se puede escribir en forma compacta como:

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{A}\mathbf{X}(k) + \mathbf{B}\Delta u(k) \\ y(k) &= \mathbf{C}\mathbf{X}(k) \end{aligned} \quad (4.7)$$

Este modelo es el que se usará para el diseño del control predictivo.

Luego de la formulación matemática, el siguiente paso en el diseño del sistema de control predictivo es calcular la salida predicha de la planta con la señal de control futura como variable manipulable. En primer lugar, se asume que el estado $\mathbf{X}(k)$ es disponible y proporciona la información actual del proceso, por lo general no todas las variables de estado son medibles o accesibles y será necesario diseñar un estimador de estado como se verá más adelante. La trayectoria de control futura dentro del horizonte de control N_c (se recuerda que se debe cumplir que $N_c \leq N_p$.) se designa como:

$$\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_c-1) \quad (4.8)$$

Si el horizonte de predicción es N_p que además es la longitud de la ventana de optimización, la información de las variables de estado futuras se puede predecir desde el instante k :

$$\mathbf{X}(k+1|k), \mathbf{X}(k+2|k), \dots, \mathbf{X}(k+m|k), \dots, \mathbf{X}(k+N_p|k) \quad (4.9)$$

Aplicando recursivamente la (4.7) hasta N_p , se pueden calcular los estados futuros de la planta definidos en (4.9) como se detalla:

$$\begin{aligned} \mathbf{X}(k+1|k) &= \mathbf{A}\mathbf{X}(k) + \mathbf{B}\Delta u(k) \\ \mathbf{X}(k+2|k) &= \mathbf{A}\mathbf{X}(k+1) + \mathbf{B}\Delta u(k+1) = \mathbf{A}^2\mathbf{X}(k) + \mathbf{A}\mathbf{B}\Delta u(k) + \mathbf{B}\Delta u(k+1) \\ \mathbf{X}(k+3|k) &= \mathbf{A}\mathbf{X}(k+2) + \mathbf{B}\Delta u(k+2) = \mathbf{A}^3\mathbf{X}(k) + \mathbf{A}^2\mathbf{B}\Delta u(k) + \mathbf{A}\mathbf{B}\Delta u(k+1) + \mathbf{B}\Delta u(k+2) \\ &\vdots \\ \mathbf{X}(k+N_c|k) &= \mathbf{A}\mathbf{X}(k+N_c-1) + \mathbf{B}\Delta u(k+N_c-1) \\ &= \mathbf{A}^{N_c}\mathbf{X}(k) + \mathbf{A}^{N_c-1}\mathbf{B}\Delta u(k) + \dots + \mathbf{A}\mathbf{B}\Delta u(k+N_c-2) + \mathbf{B}\Delta u(k+N_c-1) \\ &\vdots \\ \mathbf{X}(k+N_p|k) &= \mathbf{A}\mathbf{X}(k+N_p-1) + \mathbf{B}\Delta u(k+N_p-1) \\ &= \mathbf{A}^{N_p}\mathbf{X}(k) + \mathbf{A}^{N_p-1}\mathbf{B}\Delta u(k) + \mathbf{A}^{N_p-2}\mathbf{B}\Delta u(k+1) \\ &\quad + \mathbf{A}^{N_p-3}\mathbf{B}\Delta u(k+2) \dots + \mathbf{A}\mathbf{B}\Delta u(k+N_p-2) + \mathbf{B}\Delta u(k+N_p-1) \end{aligned} \quad (4.10)$$

Se asume que por el principio de horizonte de control $\Delta u(k+N_c) = \Delta u(k+N_c+1) = \dots = \Delta u(k+N_p-1) = 0$, por lo que se puede simplificar (4.10) y escribir $\mathbf{X}(k+N_p|k)$, como sigue:

$$\begin{aligned} \mathbf{X}(k+N_p|k) &= \mathbf{A}^{N_p}\mathbf{X}(k) + \mathbf{A}^{N_p-1}\mathbf{B}\Delta u(k) + \mathbf{A}^{N_p-2}\mathbf{B}\Delta u(k+1) \\ &\quad + \mathbf{A}^{N_p-3}\mathbf{B}\Delta u(k+2) \dots + \mathbf{A}^{N_p-N_c+1}\mathbf{B}\Delta u(k+N_c-2) \\ &\quad + \mathbf{A}^{N_p-N_c}\mathbf{B}\Delta u(k+N_c-1) \end{aligned} \quad (4.11)$$

Haciendo lo mismo con las variables de salida predichas, usando (4.11), se obtiene:

$$y(k + N_p | k) = CA^{N_p}X(k) + CA^{N_p-1}B\Delta u(k) + \dots + A^{N_p-N_c}B\Delta u(k + N_c - 1) \quad (4.12)$$

Estas ecuaciones puestas en forma matricial, se escriben como:

$$Y = F X(k) + \emptyset \Delta U \quad (4.13)$$

Siendo

$$Y = [y(k + 1|k) \ y(k + 2|k) \ \dots \ y(k + N_p|k)]^T, \quad (4.14)$$

$$\Delta U = [\Delta u(k) \ \Delta u(k + 1) \ \dots \ \Delta u(k + N_c - 1)]^T \quad (4.15)$$

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix} \quad (4.16)$$

$$\emptyset = \begin{bmatrix} CB & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ CAB & CB & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \dots & \mathbf{0} \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \dots & CA^{N_p-N_c}B \end{bmatrix} \quad (4.17)$$

Para simplificar el algoritmo se asume una entrada de referencia $r(k)$ constante durante la ventana de optimización N_p y la siguiente función de costo:

$$J = (r - Y)^T (r - Y) + \Delta U^T \bar{R} \Delta U \quad (4.18)$$

donde la información de referencia viene dada por:

$$r^T = [1 \ 1 \ \dots \ 1] \cdot r(k) \quad (4.19)$$

siendo la dimensión de r , $N_p \times 1$. La función de costo (4.17) tiene dos partes, la primera minimiza los errores entre la referencia y la salida; la segunda parte se enfoca a minimizar los esfuerzos de control. $\bar{R}$ es una matriz diagonal de pesos que viene dada por: $\bar{R} = \lambda I$, donde I es una matriz identidad de dimensión $N_c \times N_c$ y el peso $\lambda \geq 0$ se usa para sintonizar la respuesta a lazo cerrado. El problema de optimización consiste en encontrar ΔU minimizando la función de costo J .

Reemplazando (4.13) en (4.18) se tiene:

$$J = (r - F X(k))^T (r - F X(k)) - 2\Delta U^T \emptyset^T (r - F X(k)) + \Delta U^T (\emptyset^T \emptyset + \bar{R}) \Delta U \quad (4.20)$$

Derivando se obtiene:

$$\frac{\partial J}{\partial \Delta U} = -2\phi^T(r - F X(k)) + 2(\phi^T \phi + \bar{R})\Delta U \quad (4.21)$$

Se aplica la condición de minimización $\frac{\partial J}{\partial U} = 0$, y se obtiene la ley de control óptima:

$$\Delta U = (\phi^T \phi + \bar{R})^{-1} \phi^T (r - F X(k)) \quad (4.22)$$

donde el término $(\phi^T \phi + \bar{R})^{-1} \phi^T r$ corresponde al cambio en la referencia, mientras que $-(\phi^T \phi + \bar{R})^{-1} \phi^T F$ corresponde al control de realimentación de estado dentro del marco del control predictivo. Ambos dependen de los parámetros del sistema, por tanto son matrices constantes para un sistema invariante en el tiempo. Por el principio del control recesivo sólo se toma el primer elemento de ΔU en el instante k como el control incremental, por tanto:

$$\Delta u(k) = [1 \ 0 \ \dots \ 0](\phi^T \phi + \bar{R})^{-1}(\phi^T R_s r(k) - \phi^T F X(k)) \quad (4.23)$$

siendo $R_s^T = [1 \ 1 \ \dots \ 1]$, de dimensión $1 \times N_p$. La ley de control se puede escribir como:

$$\Delta u(k) = K_y r(k) - K_{GPC} X(k) \quad (4.24)$$

donde K_y es el primer elemento de $(\phi^T \phi + \bar{R})^{-1} \phi^T R_s$ y K_{GPC} es la primera fila de $(\phi^T \phi + \bar{R})^{-1} \phi^T F$.

Considerando la estructura de las matrices A y C , la última columna de F es idéntica a R_s , por tanto, K_y es idéntico al último elemento de K_{GPC} , pudiéndose escribir:

$$K_{GPC} = [K_x \ K_y] \quad (4.25)$$

donde K_x corresponde al vector de realimentación de $\Delta X_m(k)$ y K_y corresponde a la ganancia de realimentación de $y(k)$.

Con las expresiones de la ley de control y teniendo en cuenta lo explicado anteriormente se implementa en el entorno de simulación del Simulink, considerando que $X(k) = [\Delta X_m(k)^T \ y(k)]^T$. Como se muestra en la Figura 67 se requiere la medida del vector $X(k)$, por lo que en aplicaciones prácticas de este control se debe diseñar un estimador de estado.

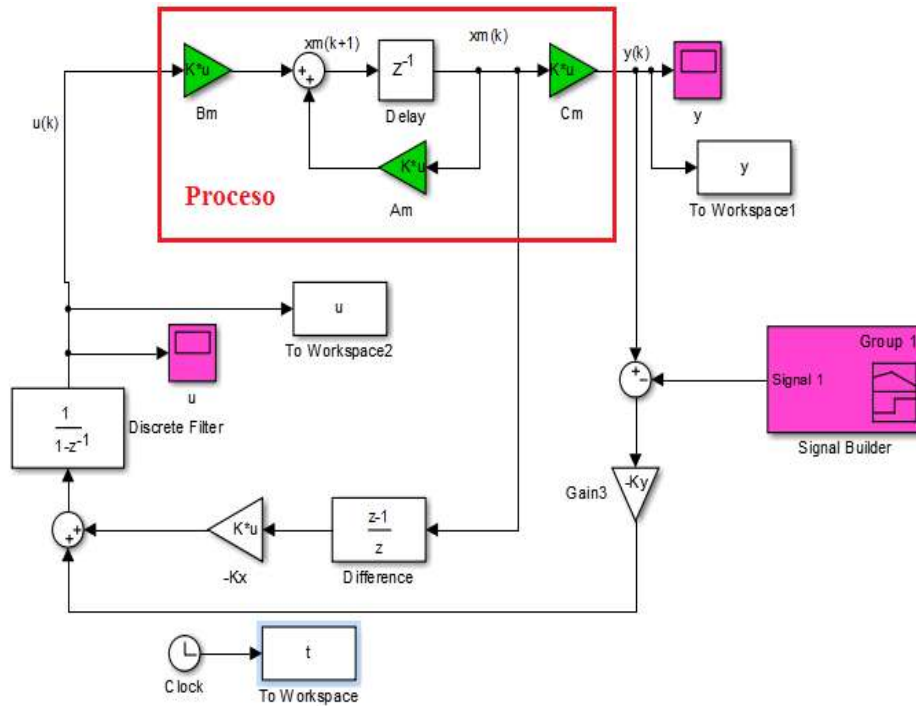


Figura 67. Control MPC con espacio de estado aumentado
Fuente: Elaboración propia

4.2. Control PID predictivo

Para encontrar la ley de control predictivo de un controlador PID se utiliza la formulación del PID discreto no predictivo que tiene la forma:

$$u(k) = k_p e(k) + T k_i \sum_{j=1}^k e(j) + \frac{k_D}{T} [e(k) - e(k-1)] \quad (4.2)$$

donde k_p , k_i y k_D son las ganancias proporcional, integral y derivativo del PID continuo, T el tiempo de muestreo y $e(k)$ el error tomado en el instante k . Tomando las diferencias de ambos lados en el paso k y $k+1$ se obtiene la llamada ley de control PID en su forma velocidad:

$$\Delta u(k) = u(k) - u(k-1) = k_p [e(k) - e(k-1)] + T k_i e(k) + \frac{k_D}{T} [e(k) - 2e(k-1) + e(k-2)] \quad (4.27)$$

$$\Delta u(k) = \left[k_p + T k_i + \frac{k_D}{T} \right] e(k) + \left[-k_p - 2 \frac{k_D}{T} \right] e(k-1) + \frac{k_D}{T} e(k-2) \quad (4.28)$$

Transformando en el dominio z se obtiene:

$$U(z) = \frac{[p_0 + p_1 z^{-1} + p_2 z^{-2}]}{1 - z^{-1}} E(z) \quad (4.29)$$

Donde:

$$p_0 = \left(k_p + T k_I + \frac{k_D}{T}\right), \quad p_1 = -\left(k_p + 2 \frac{k_D}{T}\right), \quad p_2 = \frac{k_D}{T} \quad (4.30)$$

O lo que es lo mismo:

$$k_p = -p_1 - 2p_2 = K_p \quad (4.31)$$

$$T k_I = p_0 + p_1 + p_2 = K_I \quad (4.32)$$

$$\frac{k_D}{T} = p_2 = K_D \quad (4.33)$$

Para encontrar el algoritmo en su forma predictiva (Uren & Van Schoor, 2011), primero se hace el desarrollo de la ley de control PID dada por (4.28) en términos de la entrada de referencia $r(k)$, el vector de estado $\mathbf{X}_m(k)$ y la entrada $u(k)$. Por simplicidad, consideramos que los parámetros encontrados corresponden al PID discreto: K_p , K_I y K_D .

Se tiene entonces:

$$\begin{aligned} \Delta u(k) &= [K_p + K_I + K_D]e(k) + [-K_p - 2K_D]e(k-1) + K_D e(k-2) = \\ &= p_0 e(k) + p_1 e(k-1) + p_2 e(k-2) \end{aligned} \quad (4.34)$$

Considerando que $e(k) = r(k) - y(k)$ y asumiendo que $\dots = r(k-1) = r(k) = r(k+1) = \dots$, se puede escribir la (4.34) como:

$$\Delta u(k) = (p_0 + p_1 + p_2)r(k) - p_0 y(k) - p_1 y(k-1) - p_2 y(k-2) \quad (4.35)$$

Desarrollando la salida en términos de las variables de estado y la variable manipulable, usando la expresión de la salida del proceso dada por (4.1), se tiene:

$$\begin{aligned} y(k) &= C_m \mathbf{X}_m(k) = C_m (A_m \mathbf{X}_m(k-1) + B_m u(k-1)) \\ &= C_m A_m^2 \mathbf{X}_m(k-2) + C_m A_m B_m u(k-2) + C_m B_m u(k-1) \end{aligned} \quad (4.36)$$

Usando el mismo método para $y(k-1)$ y $y(k-2)$, se tiene:

$$y(k-1) = C_m A_m \mathbf{X}_m(k-2) + C_m B_m u(k-2) \quad (4.37)$$

$$y(k-2) = C_m \mathbf{X}_m(k-2) \quad (4.38)$$

Reemplazando las ecuaciones desde la (4.36) a la (4.38) en la (4.35) se obtiene:

$$\begin{aligned} \Delta u(k) = & (p_0 + p_1 + p_2)r(k) - [p_0 C_m A_m^2 + p_1 C_m A_m + p_2 C_m] \mathbf{X}_m(k-2) \\ & - [p_0 C_m A_m B_m + p_1 C_m B_m] u(k-2) - p_0 C_m B_m u(k-1) \end{aligned} \quad (4.39)$$

Si el orden del proceso es n entonces las dimensiones de las matrices del proceso SISO serán: $A_m \in R^{n \times n}$, $B_m \in R^{n \times 1}$, $C_m \in R^{1 \times n}$. La entrada $u(k)$ y salida $y(k)$ son unidimensionales.

Para que el algoritmo desarrollado en (4.39) tenga carácter predictivo debe ser equivalente a la ley de control MPC óptimo dada por (4.24).

De la (4.24) y reemplazando $\mathbf{X}(k)$ por su equivalente en términos de las variables originales del modelo del proceso en espacio de estado se puede escribir:

$$\Delta u(k) = K_y r(k) - [K_x \quad K_y] \begin{bmatrix} \Delta \mathbf{X}_m(k) \\ y(k) \end{bmatrix} \quad (4.40)$$

Desarrollando se tiene:

$$\Delta u(k) = K_y r(k) - [K_x \quad K_y] \begin{bmatrix} \mathbf{X}_m(k) - \mathbf{X}_m(k-1) \\ y(k) \end{bmatrix} \quad (4.41)$$

Para igualar esta expresión con la encontrada del PID en (4.39) se opera reemplazando las variables de estado $\mathbf{X}_m(k)$, $\mathbf{X}_m(k-1)$ y la salida $y(k)$ en términos del estado $\mathbf{X}_m(k-2)$ y en términos de las entradas pasadas $u(k-1)$ y $u(k-2)$, usando las ecuaciones de salida y estado del modelo del proceso. De esta forma se tiene:

$$\begin{aligned} \Delta u(k) = & K_y r(k) - [(K_x + K_y C_m) A_m^2 - K_x A_m] \mathbf{X}_m(k-2) \\ & - [(K_x + K_y C_m) A_m B_m - K_x B_m] u(k-2) \\ & - (K_x + K_y C_m) B_m u(k-1) \end{aligned} \quad (4.42)$$

Ahora solo se deben igualar los términos de la ecuación (4.39) con la (4.42) y se pueden determinar los valores de los parámetros p_0 , p_1 y p_2 y por ende la de los parámetros del controlador PID discreto: K_p , K_I , K_D , que para el caso de la aproximación planteada se derivan a partir de la (4.31) a la (4.33). El algoritmo desarrollado se implementa en Simulink (Figura 68).

Se ha usado como proceso un sistema masa resorte cuya representación en espacio de estado discretizado con $T_s = 0.01s$ es el siguiente:

$$A_{m2} = \begin{bmatrix} 0.9623 & 0.0901 \\ -0.3604 & 1.0423 \end{bmatrix}, B_{m2} = \begin{bmatrix} 0.018 \\ 0.3604 \end{bmatrix}, C_{m2} = [0.991 \quad 0.045]$$

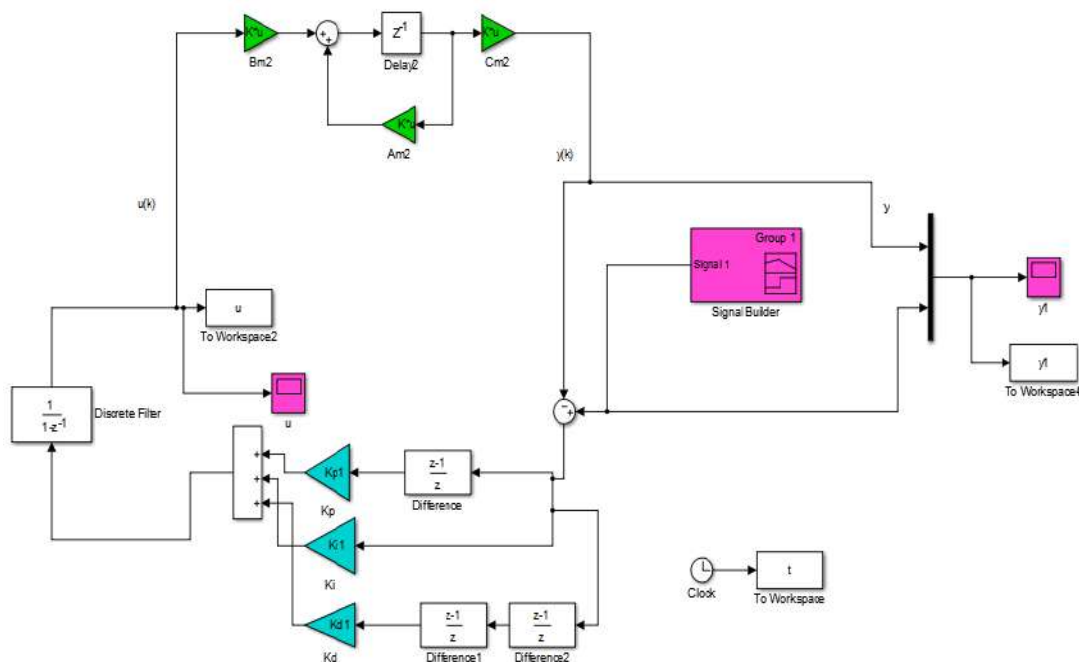


Figura 68. Control PID predictivo
Fuente: Elaboración propia

Comparando los resultados del algoritmo PID predictivo con el control MPC (Figura 69), se puede deducir que el control PID predictivo responde más rápido que el control MPC encontrado, usando los mismos parámetros de sintonización.

No obstante los valores máximos y mínimos de la variable manipulable en breves instantes son mucho mayores para el control PID predictivo (en cada cambio de la referencia la variable manipulable da un salto impulsivo en breves instantes pero luego se comporta como la variable manipulable del control MPC). Como era de esperarse se evidencia una reducción apreciable de este valor conforme aumenta el valor de λ , esto es debido a que en la función objetivo dicho factor pondera el esfuerzo de control.

Algunas características importantes de ambos controladores se muestran en la Tabla 10.

Tabla 10. Parámetros característicos de controladores predictivos PIDp y MPC para sistema masa-resorte

Parámetro	PIDp			MPC		
	$\lambda = 0.05$	$\lambda = 1$	$\lambda = 5$	$\lambda = 0.05$	$\lambda = 1$	$\lambda = 5$
u_{c_max}	100	90.1	53.42	16	6.2	6
u_{c_min}	-122.24	-109.72	-60.9767	-5.77	0	0
$t_{est}(5\%)$ en seg	1.2	1.2	1.2	1	2.5	3
K_p	9.9039	2.6762	1.6762			
K_i	2.9387	0.6357	0.3664			
K_d	19.545	13.7883	10.46			
K_{mx}^T				$\begin{bmatrix} 9.1079 \\ 2.343 \end{bmatrix}$	$\begin{bmatrix} 2.1535 \\ 1.4525 \end{bmatrix}$	$\begin{bmatrix} 1.2826 \\ 1.0868 \end{bmatrix}$
K_y				2.977	0.6629	0.387

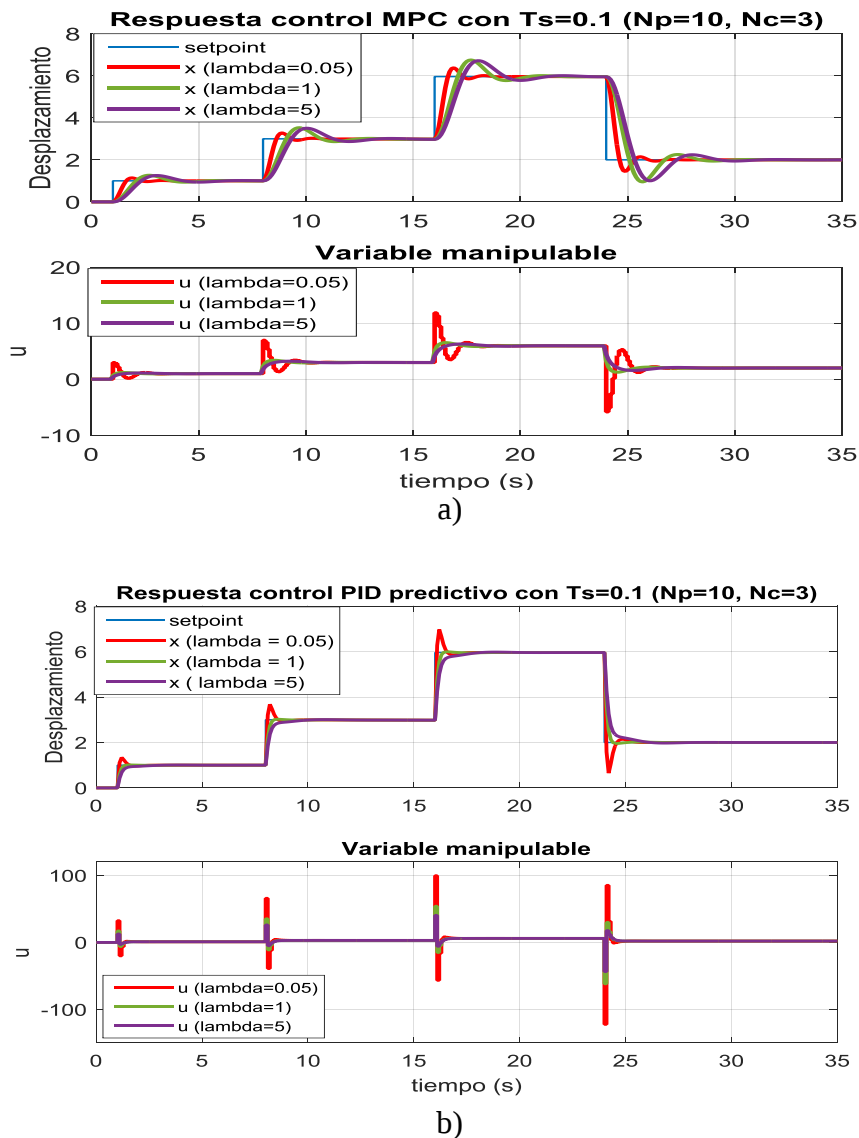


Figura 69. a) Respuesta MPC, b) Respuesta PID predictivo
Fuente: Elaboración propia

4.3. Control PID predictivo con modelo Wiener (WPIDp)

Visto el algoritmo PID predictivo desarrollado en el apartado anterior se aplica a la planta de pH modelada con una estructura Wiener. Para ello se usa el modelo identificado con el método N4SID con descomposición SVD descrito en el capítulo 3.

Es importante conocer la expresión de la función inversa del bloque no lineal pues el fundamento del PID predictivo es que se aplica a un modelo lineal del proceso. De allí la importancia de desacoplar los dos bloques.

Para la implementación del control predictivo usando modelo Wiener, tanto para el control WPIDp - desarrollado en esta tesis - como para el control WMPC, que usaremos para comparar los resultados, se ha estructurado el esquema de control como se indica en la Figura 70 a). Este esquema obedece al hecho que el proceso no lineal se descompone en un

bloque lineal, que en nuestro caso se representa en espacio de estado y un bloque no lineal (aproximación polinómica), como se muestra en la Figura 70 b).

Cabe mencionar que en la estructura con modelo Wiener la inversa de la no linealidad se aplica tanto a la entrada de referencia como a la salida del proceso.

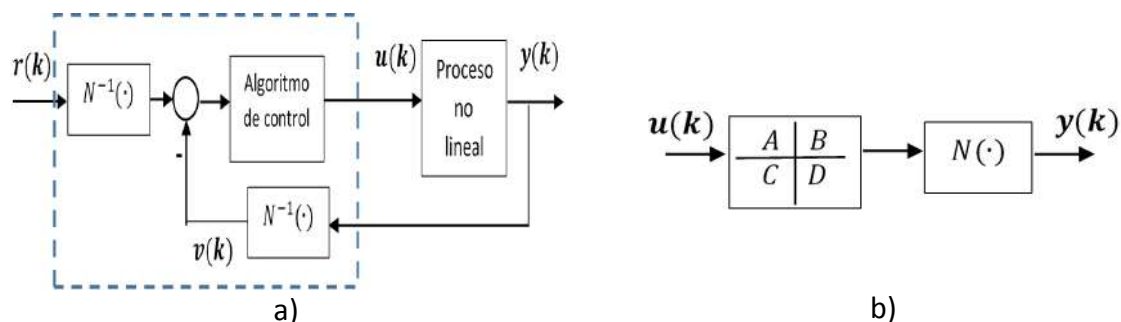


Figura 70. a) Esquema control con modelo Wiener del proceso, b) Modelo Wiener

Los métodos de identificación usados en el caso de la estructura Wiener han permitido obtener directamente la inversa del bloque no lineal por lo que no será necesario calcularlo. La implementación del control predictivo con estructura Wiener se muestra en la Figura 71.

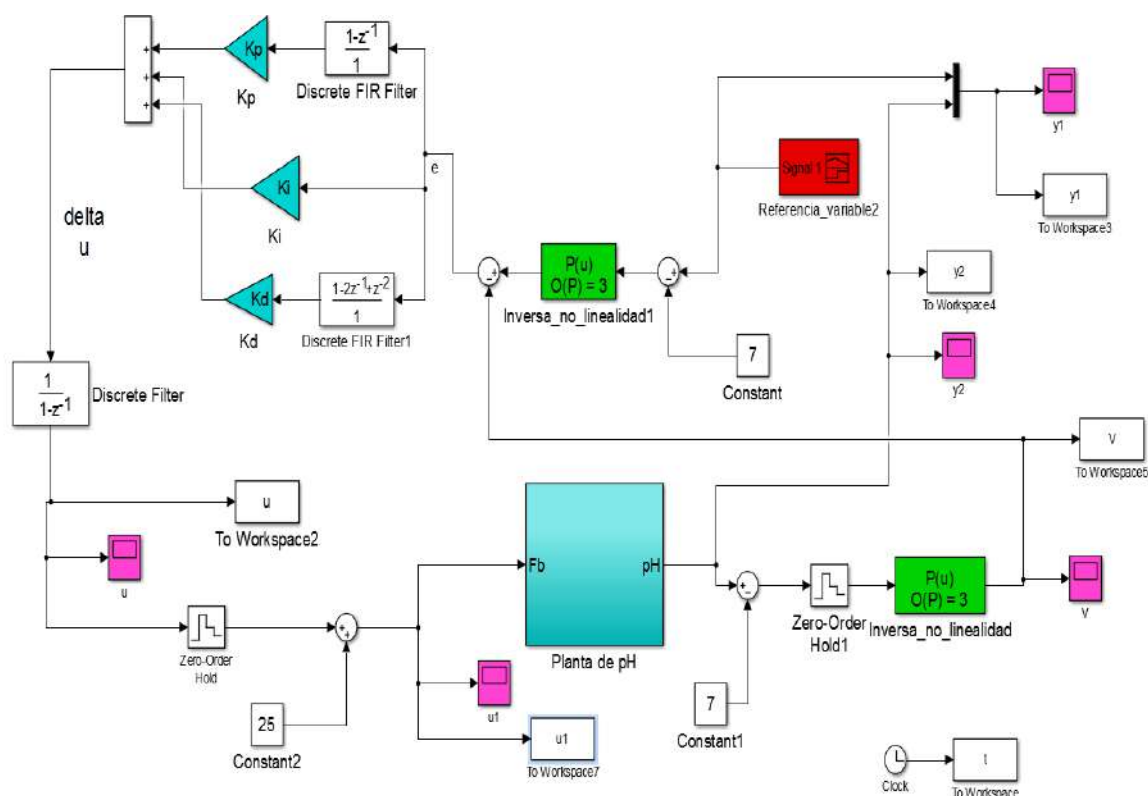


Figura 71. Diagrama del control WPID predictivo de la planta de pH
Fuente: Elaboración propia

La respuesta del control Wiener PID predictivo, llamado WPIDp, para $N_c = 3$, $\lambda = 100$ ante dos valores del horizonte de predicción, N_p (Figura 72) muestra que el control

WPIDp responde mejor para $N_p = 10$, sin embargo, el esfuerzo de control es mayor para un horizonte mayor, en especial en los instantes de mayor variación en la referencia.

Se debe tener en cuenta que en una aplicación real es importante evaluar la sintonización teniendo en cuenta posibles restricciones en cuyo caso será necesario implementar un optimizador que contemple dichas restricciones pues los valores de la variable de control no deben salir de los límites.

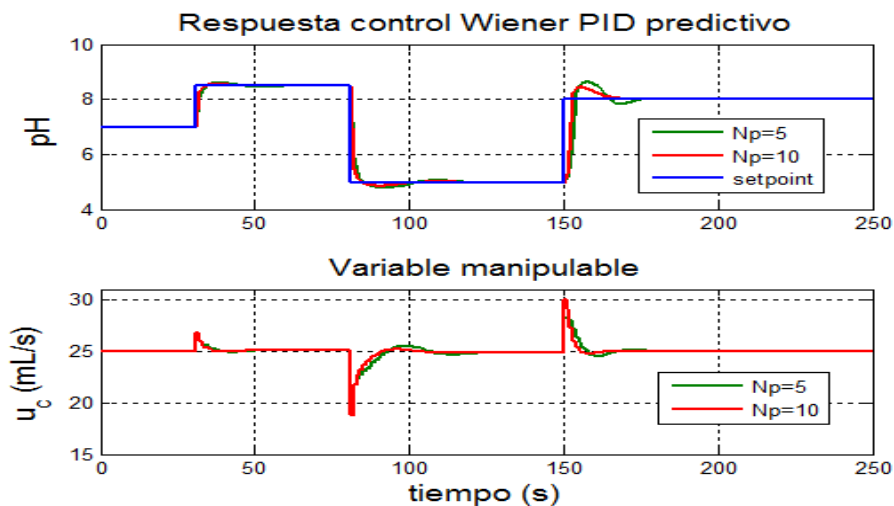


Figura 72. Salida control WPID predictivo para modelo SVD ante variaciones del horizonte de predicción.

Fuente: Elaboración propia

4.3.1. Control PID predictivo con modelo Wiener (WPIDp) y control MPC con modelo Wiener (WMPC)

Se diseña el control WMPC con el objetivo de comparar ambos controladores, para una comparación adecuada se han usado los mismos horizontes del WPIDp. Como se observa de la Figura 73, la respuesta del WMPC presenta menos sobreoscilaciones a un horizonte de predicción mayor, aunque también aumenta el tiempo de respuesta. Al contrario del control WPIDp para el que a un mayor horizonte de predicción se presenta un menor esfuerzo de control.

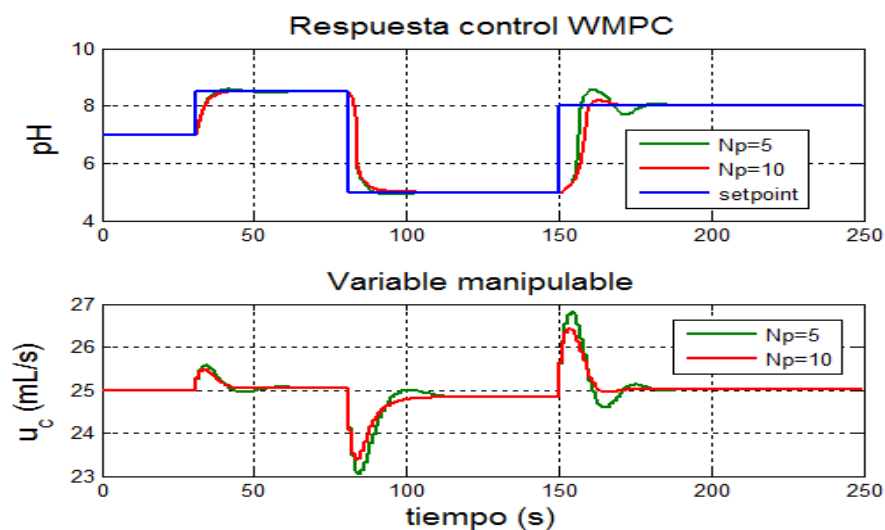


Figura 73. Respuesta WMPC ante variaciones de N_p
Fuente: Elaboración propia

Comparando las respuestas de ambos controladores para $N_p = 10$ se puede apreciar que efectivamente el control WMPC responde un poco más lento que el WPIDp, como se muestra en la Figura 74 aunque presenta menor sobreoscilación que el control WPIDp

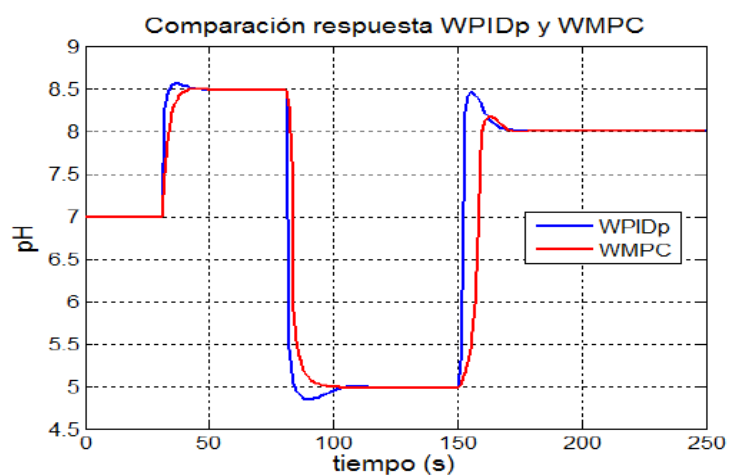


Figura 74. Respuesta de control WPIDp y WMPC para horizonte de predicción $N_p = 10$
Fuente. Elaboración propia

El control WMPC implementado en Simulink se muestra a continuación.

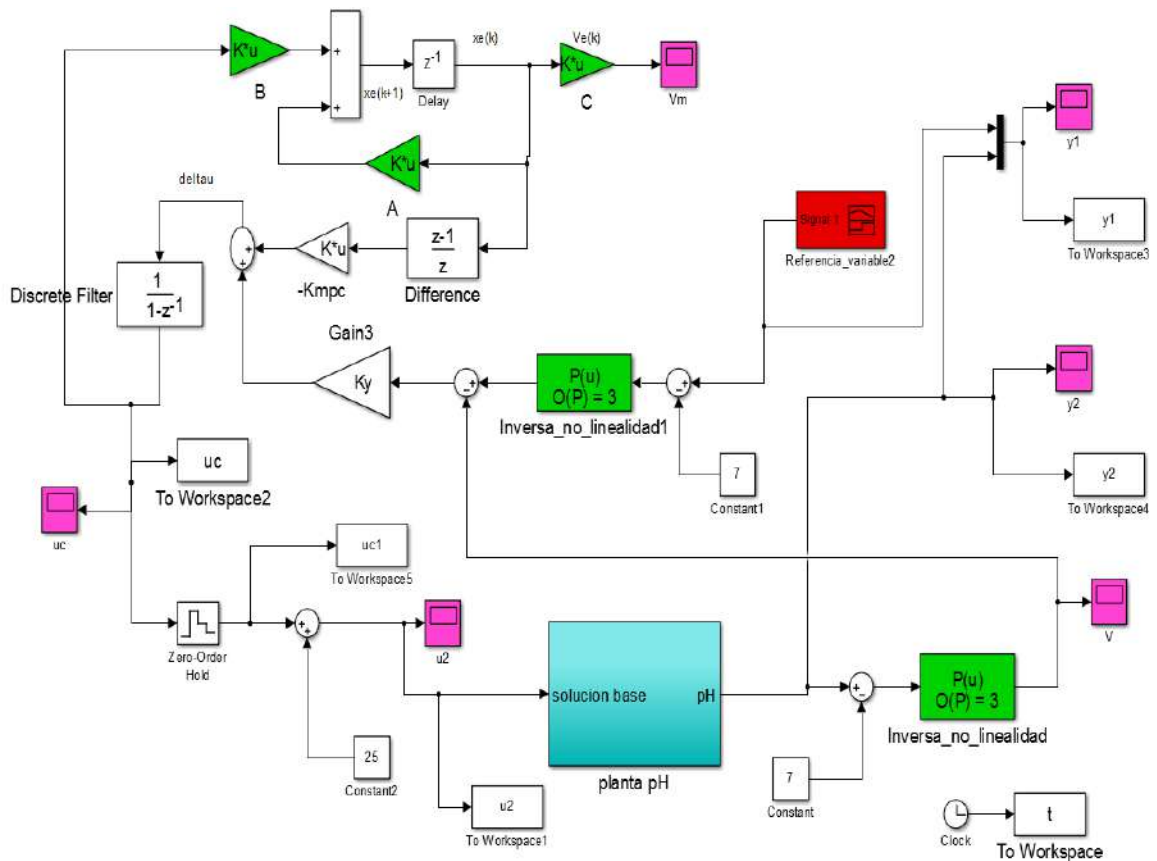


Figura 75. Modelo control Wiener MPC para planta de neutralización de pH
Fuente: Elaboración propia

En la Figura 76 se muestran las respuestas del control WPIDp y el control WMPC para distintos valores de λ . Ambos controladores responden satisfactoriamente a grandes variaciones del *setpoint*.

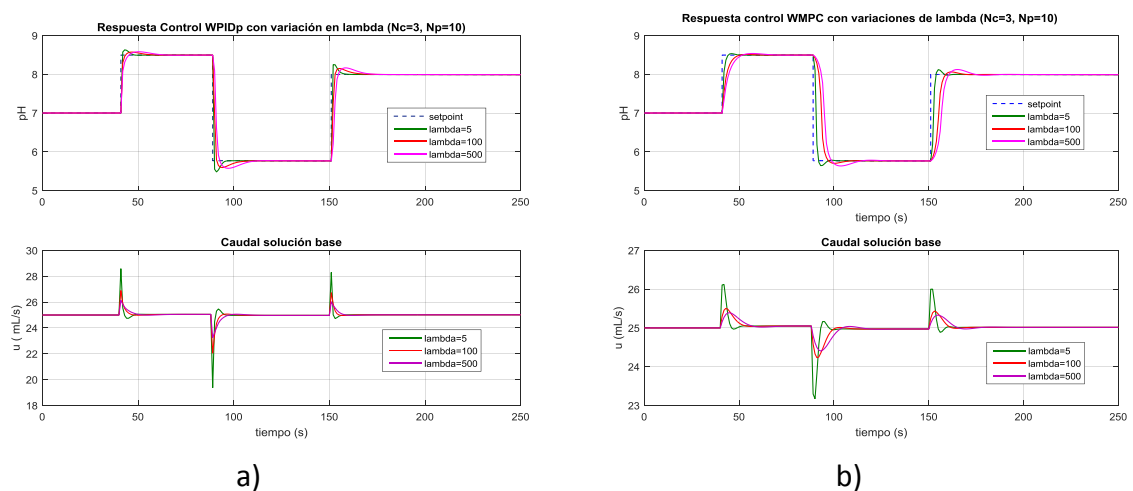


Figura 76. Comparación respuestas a variaciones de lambda: a) Control WPIDp, b) Control WMPC.
Fuente: Elaboración propia

La energía que se requiere para el control (variación de la variable manipulable) es ligeramente mayor para el caso del WPIDp (Figura 77) aunque

también se evidencia una mayor rapidez en la respuesta, como se observa en la Figura 76.

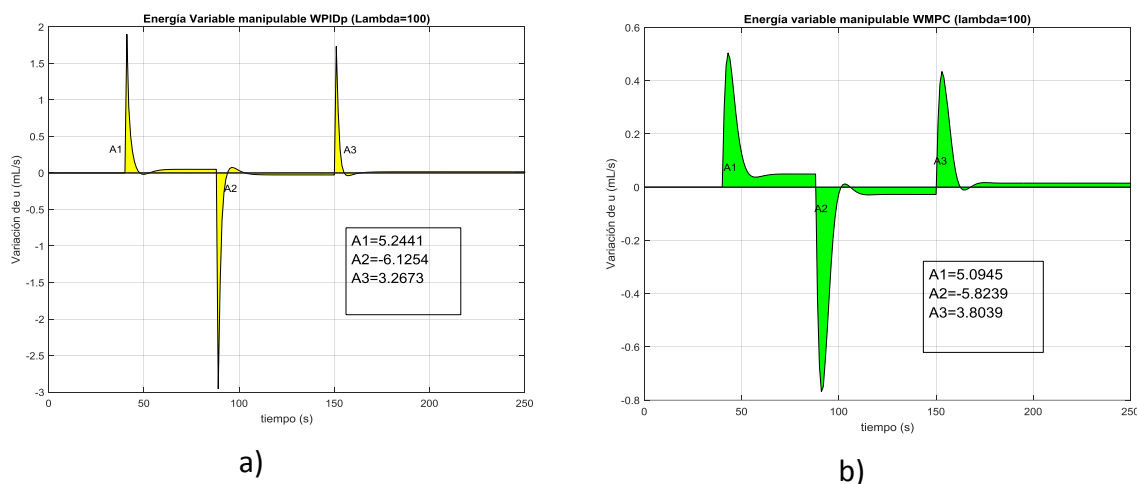


Figura 77. Energía variable manipulable para $\lambda=100$: a) Control WPIDp, b) Control WMPC
Fuente: Elaboración propia

A continuación se detallan los datos más importantes de cada controlador para los distintos valores de λ . Se ha reportado en dicha tabla los valores de las variables manipulables máximos y mínimos y tal como también se ha analizado de la Figura 77 son bastante cercanos tanto para el control WPIDp propuesto como para el control WMPC.

Tabla 11. Parámetros característicos de controladores PID y MPC predictivos con modelo Wiener para planta de pH

Parámetro	WPIDp			WMPC		
	$\lambda = 5$	$\lambda = 100$	$\lambda = 500$	$\lambda = 5$	$\lambda = 100$	$\lambda = 500$
u_{c_max}	27.8	26.5	25.5	26.12	25.5	25.39
u_{c_min}	21	23	24	23.17	24.23	24.41
K_p	0.6826	0.4408	0.1837			
K_i	0.3049	0.0814	0.0291			
K_d	0	0	0			
K_{mx}^T				$\begin{bmatrix} -0.3964 \\ -0.0604 \end{bmatrix}$	$\begin{bmatrix} -0.256 \\ -0.0387 \end{bmatrix}$	$\begin{bmatrix} -0.1067 \\ -0.0161 \end{bmatrix}$
K_y				-0.3049	-0.0814	-0.0291

4.3.2. Control PID predictivo con modelo Wiener (WPIDp) y control MPC para modelo lineal

Para validar las prestaciones del control WPIDp comparado con un control MPC, se ha diseñado este último tomando como modelo lineal el que se encontró en el capítulo 3. El esquema de implementación en Simulink se muestra en la Figura 78.

De la Figura 79 se observa que con los siguientes parámetros de sintonización: horizonte de control $N_c = 3$, horizonte de predicción $N_p = 10$ y parámetro de ponderación $\lambda = 10$; y, ante variaciones grandes del *setpoint* de pH respecto al valor neutro, el MPC no se comporta adecuadamente. En la simulación se ha aplicado una variación de 0.3 en el caudal nominal del ácido a los 400s y se puede observar que el control WPIDp responde mucho mejor que el control MPC. Esto se explica porque este último usa un modelo lineal que está limitado a un rango de operación más reducido comparado con el modelo usado para el control WPIDp.

Cabe mencionar que para el control MPC se encontró el modelo lineal aplicando una señal PRBS con variaciones de $\pm 0.1 \text{ mL/s}$ alrededor del valor nominal de 25 mL/s . Esto correspondía a variaciones en la salida de $6 \leq \text{pH} \leq 8$. El modelo lineal que se obtuvo en espacio de estado viene dado por:

$$A_m = \begin{bmatrix} 0.9885 & 0.06233 \\ 0.001635 & 0.9362 \end{bmatrix}; B_m = \begin{bmatrix} 0.02745 \\ 0.001823 \end{bmatrix}; C_m = [36.09 \quad -0.04912]; D_m = 0$$

Además fue necesario diseñar un estimador pues se deben compensar los errores del modelo encontrado (*FIT*: 70.9%).

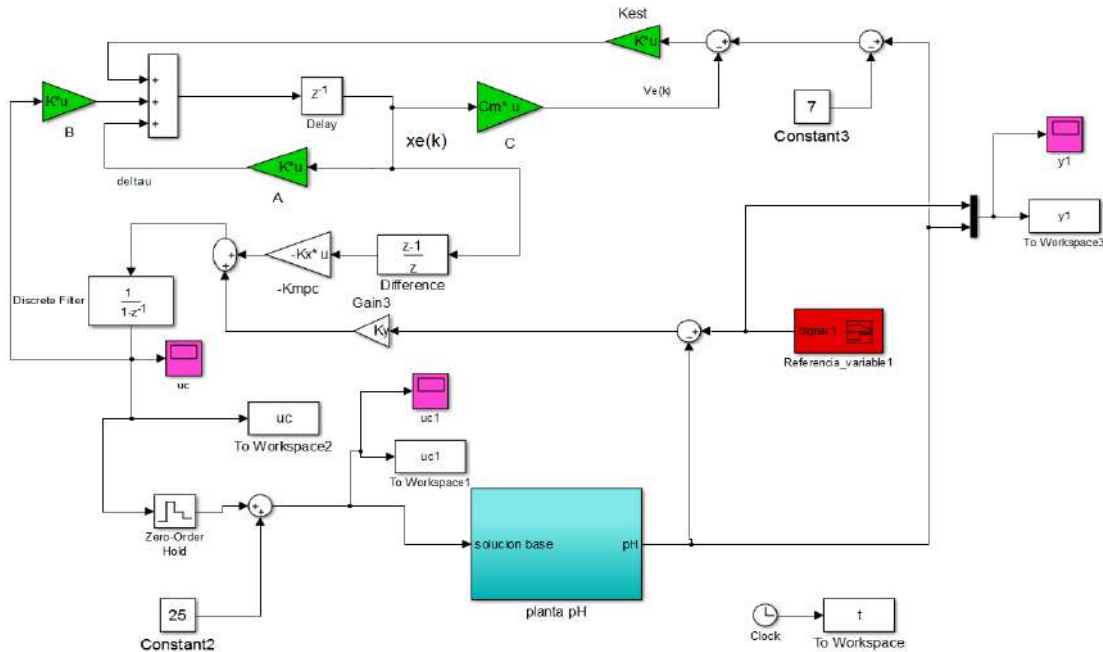


Figura 78. Esquema en Simulink del control MPC
Fuente: Elaboración propia

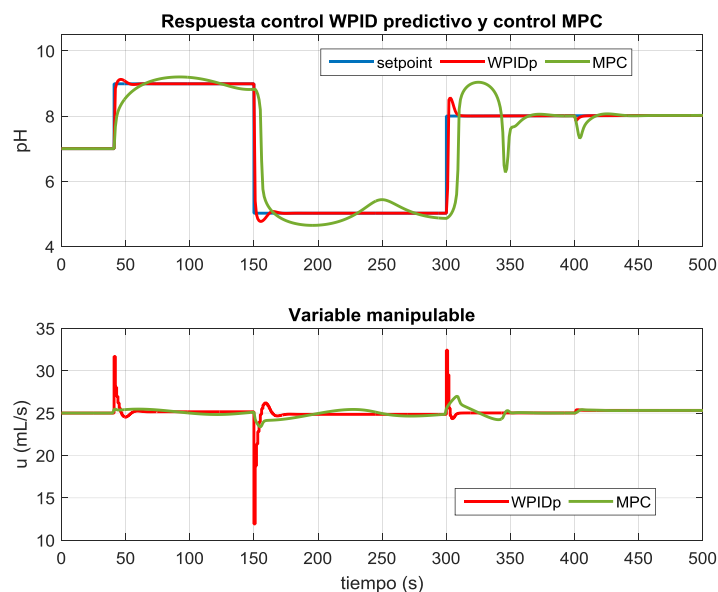


Figura 79. Comparación respuesta control WPIDp y MPC a grandes variaciones respecto al valor nominal del caudal de solución base

Fuente: Elaboración propia

Si se reduce el rango de variación en el *setpoint* hay mejoras en la respuesta del control MPC como *se* puede observar en la Figura 80 en donde se comparan las respuestas de ambos controladores con variación en el parámetro λ de 100 a 10, notándose una notable mejora para $\lambda = 10$.

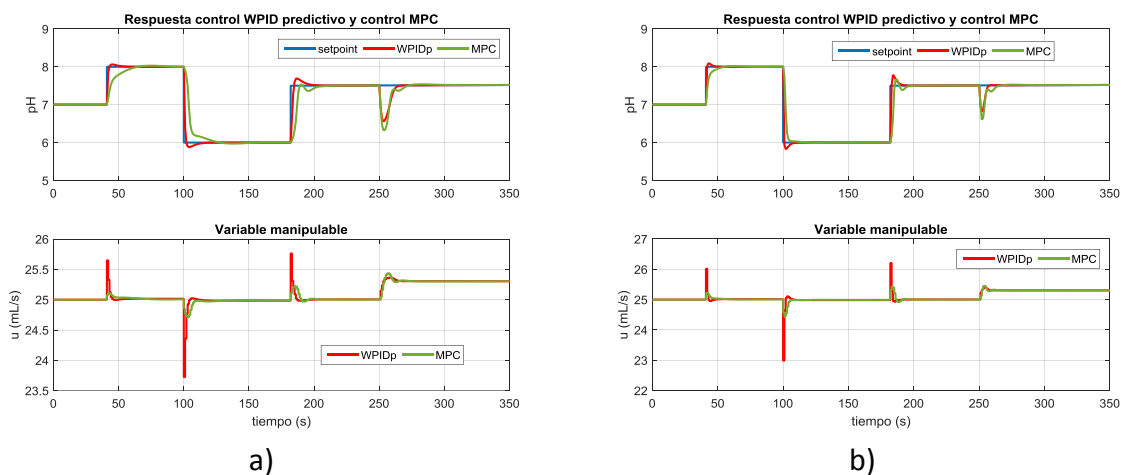


Figura 80. Respuesta WPIDp y MPC pequeñas variaciones respecto al valor nominal del caudal de solución base con: a) $\lambda=100$, b) $\lambda=10$

Fuente: Elaboración propia

4.3.3. Comparación control WPIDp y control PI clásico

En este apartado analizaremos el comportamiento del control PID predictivo encontrado usando el modelo Wiener comparado con el control PI clásico diseñado.

Introduciremos variaciones en el caudal de la solución ácida, para analizar la respuesta de los controladores ante dicha perturbación.

Será necesario diseñar un control PID clásico, que se usará para comparar tanto con el control WPIDp como con el control HPIDp que se explicará más adelante. Dado que nos interesa evaluar el control alrededor de pH=7 se aplica una variación a la entrada de la planta que va desde 25mL/s a 25.08 mL/s. La sintonización la hacemos usando el método de asignación de polos, considerando que la respuesta a escalón se aproxima a la de un sistema de primer orden sin retardo (Figura 81). En este caso será suficiente sintonizar un control PI.

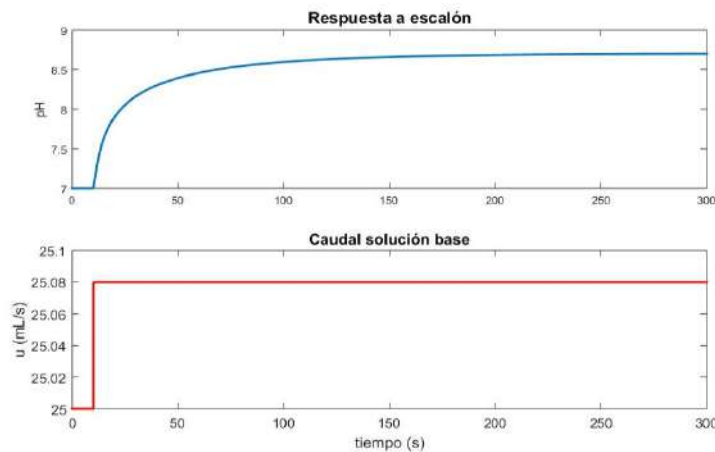


Figura 81. Respuesta a escalón a lazo abierto del proceso
Fuente: Elaboración propia

Datos de la respuesta a escalón: $\tau = 14$, $K = \frac{y_s}{u_s} = \frac{1.686}{0.08} = 21.075$

Modelo primer orden: $P(s) = \frac{K}{\tau s + 1} = \frac{21.075}{14s + 1}$

En la Figura 82 se compara la respuesta del proceso real y del modelo de primer orden, ym.

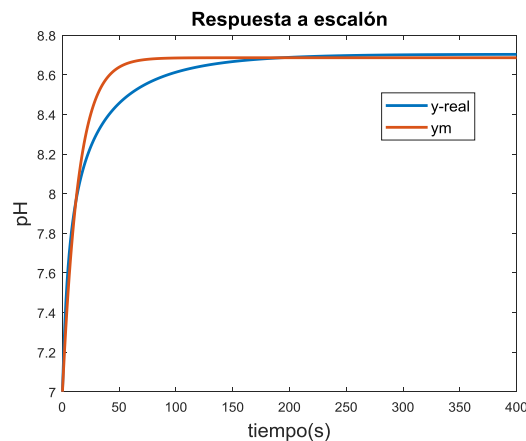


Figura 82. Comparación respuesta a escalón del proceso y modelo
Fuente: Elaboración propia

Asumiendo que el sistema a controlar es continuo y la función de transferencia del controlador PI es: $C(s) = K_p + \frac{K_I}{s}$, la función de transferencia a lazo cerrado:

$$T_{cl}(s) = \frac{C(s)P(s)}{1 + C(s)P(s)} = \frac{21.07(K_p s + K_I)}{14s^2 + (1 + 21.07K_p)s + 21.07K_I} \quad (4.43)$$

Asignando los polos: $-1/7$ y $-1/2$ (7 y 2 veces más rápido que el del modelo encontrado) se obtienen las siguientes ecuaciones para el polinomio característico y las que por reducción permitirán hallar los parámetros del controlador:

$$(7s + 1)(2s + 1) = 14s^2 + 9s + 1 \quad (4.44)$$

$$(1 + 21.07K_p) = 9, \quad 21.07K_I = 1 \quad (4.45)$$

De donde se obtiene: $K_p = \frac{8}{21.07} \approx 0.38$, $K_I = 1/21.07 \approx 0.047$

En la Figura 83 se muestra la respuesta del control PI ante variaciones del *setpoint*. Se observa que el comportamiento no es adecuado para las variaciones esperadas en el *setpoint*, cuando estas son en un rango amplio, sin embargo, si se reducen las amplitudes de las variaciones, el control PI da una mejor respuesta (Figura 84). Se puede también notar el comportamiento de la variable de control en ambos casos. Es claro que la variable de control es mejor en el último caso pues el diseño responde a los criterios de un control PI, el cual trabaja bien para un proceso lineal o con variaciones muy pequeñas alrededor del punto de equilibrio.

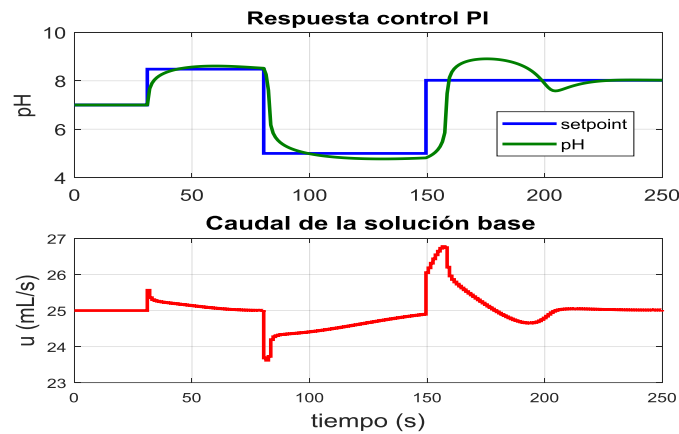


Figura 83. Respuesta del control PI ante variaciones del caudal base fuera del rango lineal
Fuente: Elaboración propia

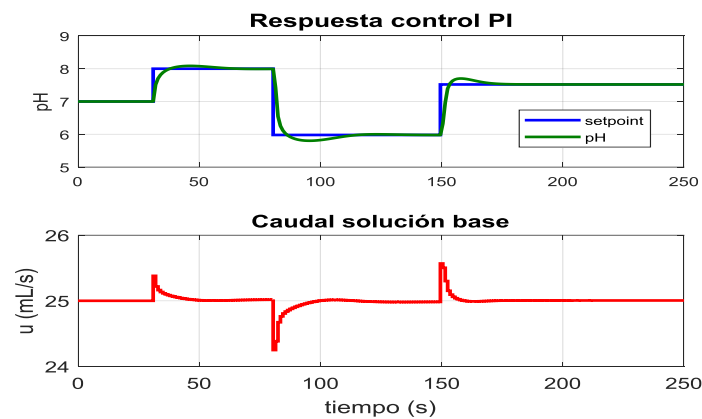


Figura 84. Control PI ante pequeñas variaciones en el *setpoint*
 Fuente: Elaboración propia

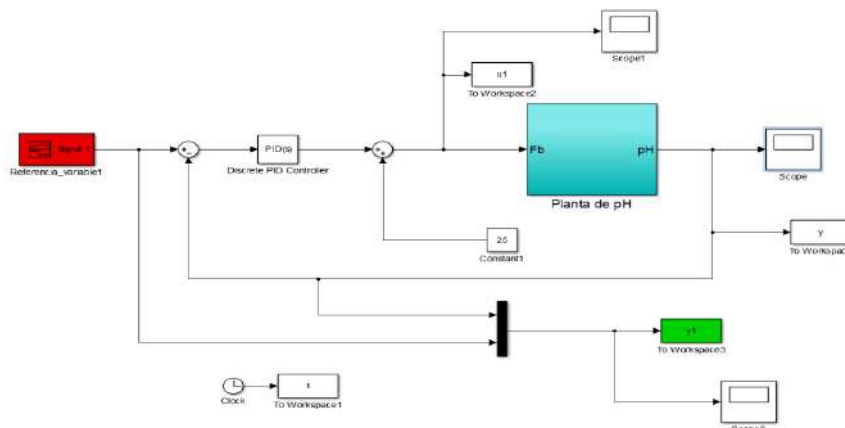


Figura 85. Control PID en Simulink
 Fuente: Elaboración propia

En la Figura 86 se puede ver una mayor robustez del control WPIDp frente al control PI. El control WPIDp responde satisfactoriamente para un amplio rango de variación en el *setpoint*. El comportamiento de ambos controladores es muy bueno para rangos pequeños de variación en el *setpoint* y tienen un buen rechazo a las perturbaciones (se simula una variación de caudal de solución ácida aplicada a los 400 s. para dar tiempo de alcanzar el estacionario). En el caso del control WPIDp se han aplicado los horizontes: $N_c = 3$, $N_p = 10$, $\lambda = 100$.

Los parámetros de los controladores PI implementados se detallan en la tabla siguiente y son los que se han encontrado en apartados anteriores.

Tabla 12. Parámetros del control PID y WPIDp para planta de pH

Control	K_P	K_I	K_D
PI	0.38	0.047	0
WPIDp	0.4408	0.0814	0

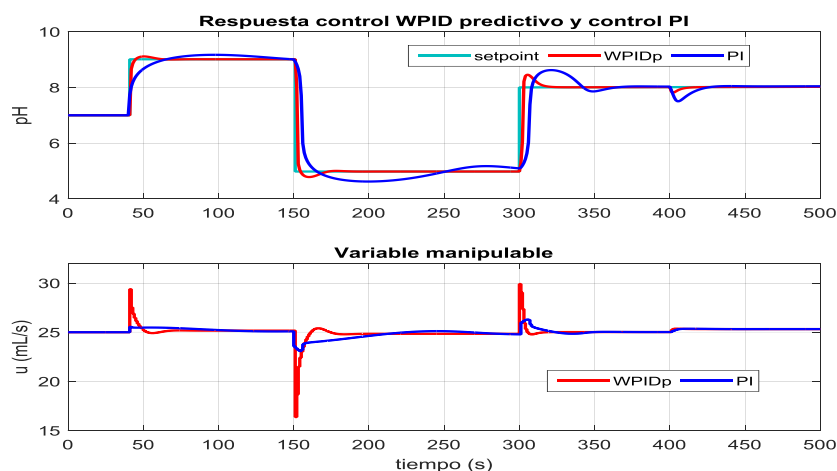


Figura 86. Comparación control WPIDp vs PI con grandes variaciones del *setpoint*
Fuente: Elaboración propia

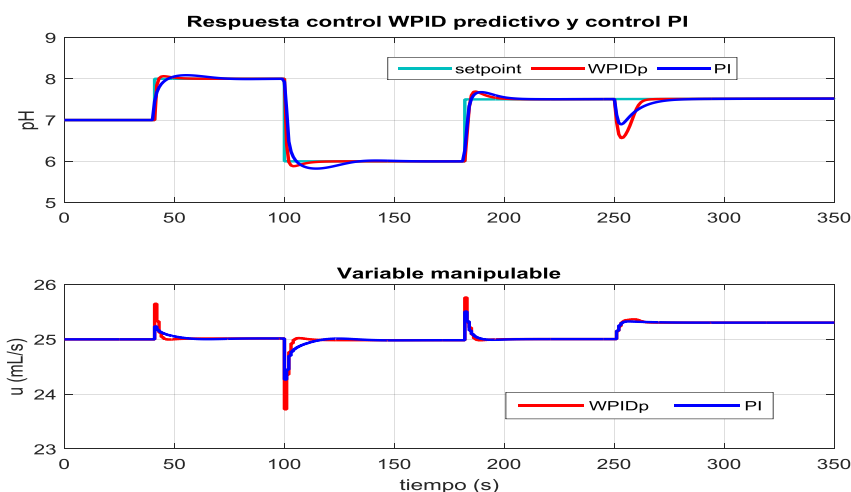


Figura 87. Comparación control WPIDp vs PI con pequeñas variaciones del *setpoint*
Fuente: Elaboración propia

4.4. Control PID predictivo con modelo Hammerstein (HPIDp)

Para el diseño del controlador predictivo Hammerstein es necesario calcular la inversa de la no linealidad hallada en el capítulo 3, dado que el algoritmo usado arroja el polinomio que caracteriza el bloque no lineal. Se ha aplicado la instrucción *interp1* del MATLAB con el método *pchip* y se ha implementado una función para la aproximación lineal por el método *pwlinear*. En ambos casos se obtiene una buena aproximación de la función inversa (Figura 88) pero cara a la implementación en el control PID predictivo se ha preferido usar la caracterización usando *pwlinear*.

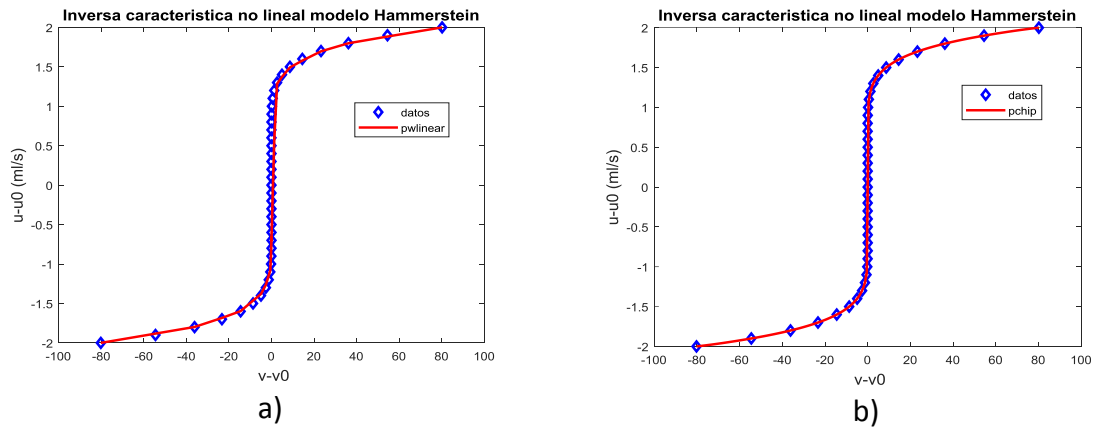


Figura 88. Característica de la inversa del bloque no lineal modelo Hammerstein: a) Aproximación pchip, b) Aproximación pwnear
Fuente: Elaboración propia

Con la caracterización de la inversa del bloque no lineal y tomando el modelo Hammerstein hallado en el capítulo 3 se implementa el control Hammerstein PID predictivo, HPIDp, como se muestra en la Figura 89.

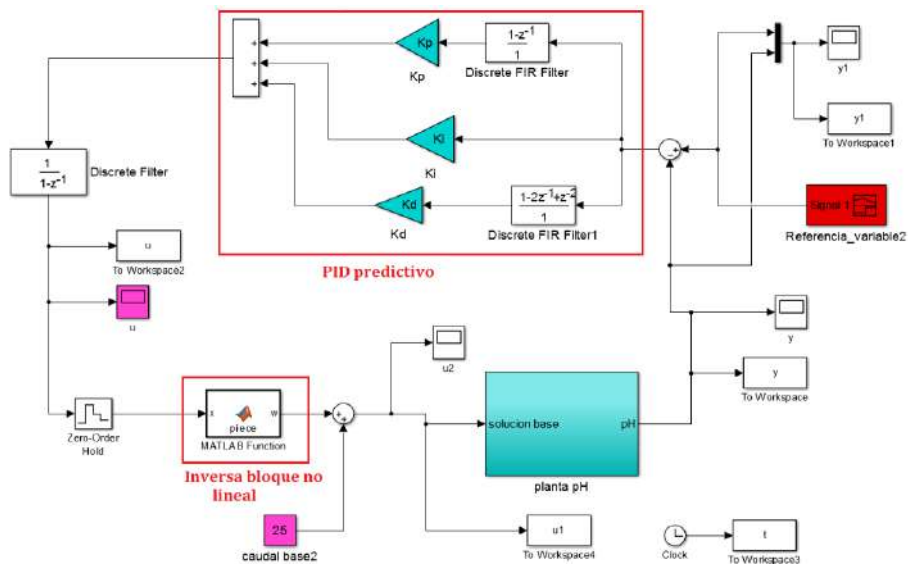


Figura 89. Implementación del control Hammerstein PID predictivo o HPIDp
Fuente: Elaboración propia

El control HPIDp usando los mismos horizontes del WPIDp, esto es: $N_c = 3$, $N_p = 10$, $\lambda = 100$, no responde adecuadamente, lo cual se debe a que el modelo de Hammerstein encontrado para grandes variaciones del *setpoint* no es muy bueno (FIT de 61.2% para el método SVD y FIT de 62.75% para modelo con bases ortonormales). El modelo Hammerstein en el caso de la aplicación por la misma estructura del proceso no dará buenos resultados pues basado en los principios del proceso de neutralización, como se ha explicado en el capítulo 2, primero se da la dinámica lineal y luego se calcula con una función no lineal la concentración y por ende el valor de pH, lo que corresponde a una estructura Wiener. Para validar los algoritmos de control predictivo con modelo Hammerstein se hace a continuación el análisis del control Hamerstein PID predictivo, aplicando variaciones en el flujo ácido a los 400s., tal y como se hizo en el control WPIDp, para $\lambda = 10$.

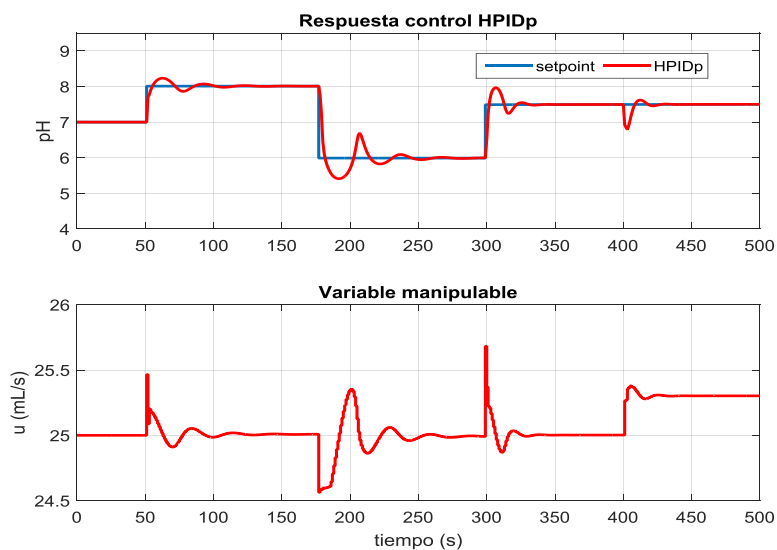


Figura 90. Control HPID predictivo para variaciones pequeñas respecto al valor nominal del caudal de solución base y con perturbación en el flujo ácido

Fuente: Elaboración propia

4.4.1. Control PID predictivo con modelo Hammerstein y control MPC para modelo lineal

El algoritmo HPIDp se ha comparado con el MPC con los horizontes $N_c = 3$, $N_p = 10$, $\lambda = 10$ para el mismo rango de variación del *setpoint* del apartado anterior. Se observa (Figura 91) que para este rango de operación, el control HPIDp se comporta peor que el MPC.

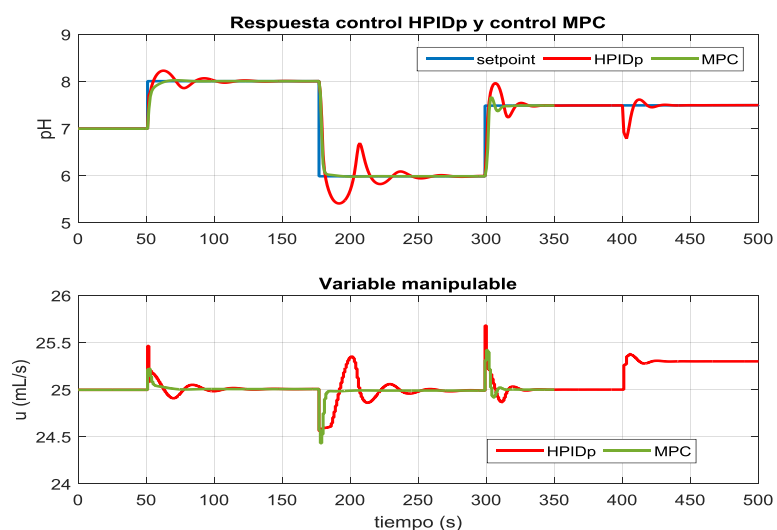


Figura 91. Respuesta de control HPIDp y MPC con perturbación en el flujo ácido

Fuente: Elaboración propia

4.4.2. Comparación control HPIDp y control PI clásico

En el caso del control HPIDp usando los mismos parámetros de sintonización de la parte predictiva usados para el control WPIDp se observa que no responde adecuadamente para variaciones pequeñas en el *setpoint*. No se hace la prueba para variaciones grandes pues como se observó en la sección 4.4 el control HPIDp no responde satisfactoriamente, dado que el modelo usado no tenía un buen ajuste. Se observa de la Figura 92 que presenta más sobreoscilaciones que el control PI, además en los instantes iniciales se aleja de la referencia, esto debido a que se ha usado una aproximación no lineal tipo *pwlinear*. Se ha usado como parámetro de ponderación $\lambda=10$.

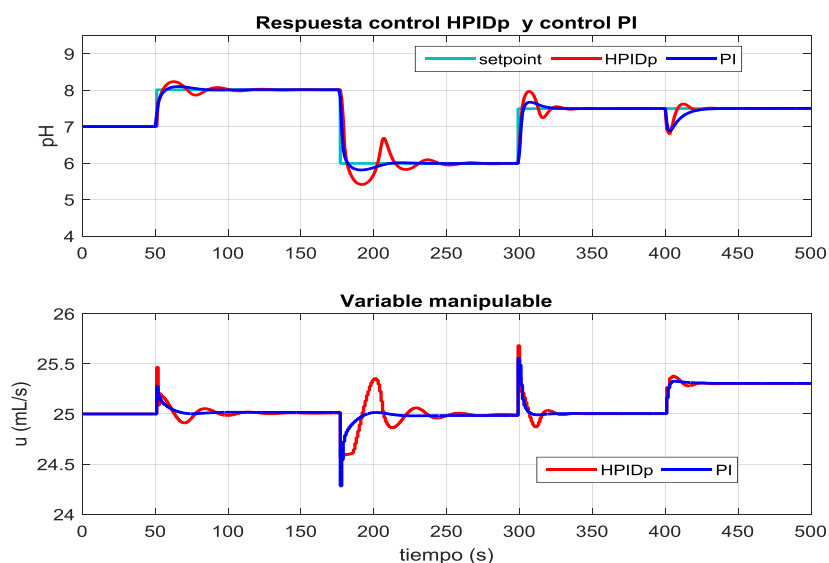


Figura 92. Comparación control HPIDp y control PI ante pequeñas variaciones del *setpoint*
Fuente: Elaboración propia

Para tener un mejor criterio de evaluación se ha reducido el parámetro λ asignándose $\lambda = 0.5$. Aunque se observan mejoras en la respuesta del control HPIDp no resultan satisfactorias ni mejor que el control PI (Figura 93).

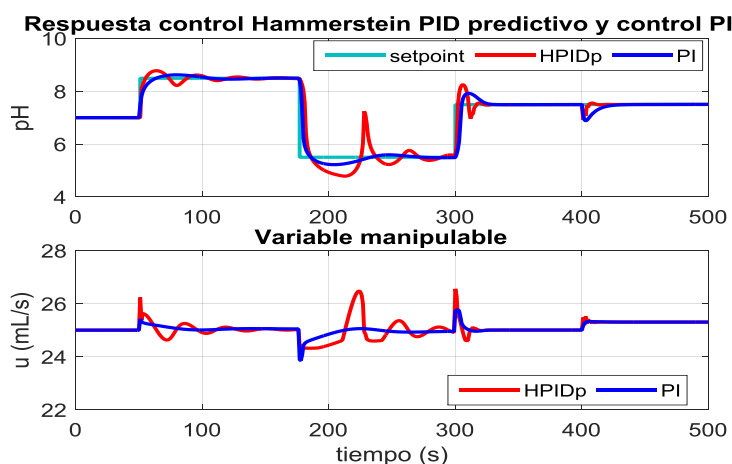


Figura 93. Comparación HPIDp con ajuste en λ y PI
Fuente: Elaboración propia

Conclusiones y recomendaciones

El proceso de neutralización de pH estudiado es un proceso no lineal y con alta sensibilidad alrededor del pH neutro, en nuestro caso, $pH=7$, pues se mezclan un ácido y una base fuerte. La alta sensibilidad en el rango $4 \leq pH \leq 9$ mostrada en la Figura 40 y Figura 43 nos ha permitido validar la robustez y factibilidad del control predictivo presentado en este trabajo, rango en el cual en otras estructuras clásicas de PID no se tienen resultados aceptables, como se ha evidenciado en esta tesis.

Dos métodos de identificación usados para encontrar un modelo que tenga en cuenta la no linealidad del proceso han sido evaluados, estos son: método de identificación N4SID con descomposición SVD (valores singulares) y métodos de descomposición en bases ortonormales, ambos propuestos por Gómez et al, (2004, 2005), Giri y Bai (2010), tanto para modelos Wiener como Hammerstein. De las alternativas estudiadas la mejor aproximación se consigue con los modelos Wiener. Como se observa en el capítulo 2 para los mismos rangos E/S de estimación y validación, el modelo Wiener tanto para los métodos SVD y bases ortonormales usados en este trabajo como para los métodos de identificación y modelos previstos en el Toolbox de Identificación del MATLAB, es el que ha satisfecho los requerimientos de identificación con un buen ajuste (FIT de la literatura inglesa) y mínimo error cuadrático medio (MLS), reportados en la Tabla 5.

Con el objetivo de evaluar una alternativa de control predictivo con estructura similar a la planteada en la tesis, se compara la respuesta del control WPIDp con el control WMPC (Wiener MPC) observándose un comportamiento ligeramente mejor en el algoritmo propuesto. En el capítulo 4 se implementó el algoritmo de control PID predictivo usando los modelos Wiener y Hammerstein. En el capítulo 4 se comparan las respuestas del control WPIDp y sus bondades frente al control MPC, sintonizado para el modelo lineal correspondiente. Una vez más se pudo comprobar que el control predictivo usando modelo Wiener o simplemente llamado WPIDp tiene muy buenas características (bajas sobreoscilaciones, respuesta rápida, rechazo a perturbaciones).

A partir de los resultados de esta tesis, se puede concluir que la propuesta planteada es una buena alternativa para incorporar características predictivas a una estructura PID, con la ventaja que esta última es ampliamente conocida en el ámbito por los operarios en una planta, además la infraestructura de equipamiento es la que usualmente se encuentra en una industria, no requiriendo de inversiones fuertes. La Figura 91 muestra el comportamiento del control con el algoritmo propuesto (WPIDp) confrontada a un control PI clásico sintonizado para un rango lineal y ante grandes variaciones alrededor del punto

de referencia, evidenciándose las bondades de una estructura que contempla la característica no lineal del proceso.

De la literatura revisada se puede decir que existen algoritmos PID predictivos que permiten ajustar los parámetros del controlador de forma iterativa, partiendo de una sintonización de un PID simple; pero esta alternativa puede implicar un mayor esfuerzo computacional.

Aunque el modelo Hammerstein no ha sido el adecuado para implementar el control HPIDp (Hammerstein PID predictivo), no se invalida la posibilidad de aplicarse si se tiene un buen modelo Hammerstein. Es ya sabido que las prestaciones de un algoritmo MPC es altamente dependiente de la exactitud del modelo, por ello es muy importante contar con el mejor modelo de la planta, que en el caso del modelo Hammerstein usando los métodos presentados en este trabajo no ha sido adecuado pues el FIT ha sido menor que 80%. Se hace notar que en la metodología de identificación usada se ha usado el proceso de identificación en una etapa (el bloque lineal no lineal calculado en un solo proceso). Cabe la posibilidad, no explorada en este trabajo, que se pueda conseguir un mejor modelo Hammerstein aplicando la metodología de identificación en dos etapas, esto es hallar primero el bloque lineal o no lineal y luego se determina el bloque no lineal o lineal según sea el caso.

Aunque la estrategia propuesta es una alternativa que se abre al campo de controles predictivos aplicados a procesos no lineales; por los requerimientos de predicción, se hace necesario encontrar un buen modelo y usar mayor tiempo en el proceso de obtención de dicho modelo que si se sintonizara un simple PID. Por otro lado, no siempre será posible aplicar las señales de prueba (PRS o GMN) a cualquier proceso, lo que podría significar una limitación en la implementación del algoritmo, aún por explorar.

Recomendaciones finales

Como futuras líneas de investigación o mejoras del proyecto realizado es recomendable explorar los métodos de identificación de modelos Wiener y Hammerstein, tal como se ha mencionado, así como la aplicación de las técnicas PID predictivas en plantas reales. Claramente el objetivo de la tesis es dar un punto de partida que facilite su uso en procesos agroindustriales en donde es de interés el control del pH, sin embargo, las metodologías aplicadas pueden usarse para otro tipo de proceso.

Será importante determinar con algunas pruebas empíricas y con conocimiento a priori de algunas características del proceso que evidencien el tipo de no linealidad para así seleccionar la posible estructura del modelo: Wiener, Hammerstein o una combinación de ambos.

Bibliografía

- Arefi, M., Montazeri, A., Poshtan, J., & Jahed-Motlagh, M. (2006). Nonlinear model predictive control of chemical processes with a Wiener identification approach. *In Industrial Technology, 2006. ICIT 2006. IEEE International Conference on*, 1735-1740.
- Arousi, F. (2009). Predictive algorithms for linear and nonlinear processes. *Disertación doctoral. Budapest University of Technology and Economics*.
- Babu, R., & Swarnalatha, R. (2017). Comparison of Different Tuning Methods for pH Neutralization in Textile Industry. *Journal of Applied Sciences*, 142-147.
- Bars, R., Haber, R., & Schmitz, U. (2011). *Predictive Control in Process Engineering*. Alemania: WILEY-VCH.
- Bars, R., Haber, R., Schmitz, U., & Arousi, F. (2008). Predictive PI(D) Control. Workshop on Process Automation.
- Biagiola, S. I., & Figueroa, J. L. (2009). Identificación robusta de filtros Wiener y Hammerstein. *Revista Iberoamericana de Automática e Informática Industrial*, 6, 98-107.
- Biagiola, S. I., Agamennoni, O. E., & Figueroa, J. L. (2004). Hinfinito control of a Wiener-type system. *International Journal of Control*, 77(6), 572-583.
- Biagiola, S. I., Agamennoni, O. E., & Figueroa, J. L. (2016). Robust control of Wiener systems: Application to a pH neutralization process. *Brazilian Journal of Chemical Engineering*, 33(1), 145-153.
- Bordons, C. (2000). Control Predictivo: metodología, tecnología y nuevas perspectivas. *Recuperado de <https://control-ps2316-sept2009.wikispaces.com/file/view/CONTROL+PREDICTIVO.pdf>*.
- Brouri, A., Giri, F., Mkhida, A., Chaoui, F. Z., Elkarkri, A., & Chhibaat. (2014). Identification of Nonlinear Systems Structured by Hammerstein- Wiener model. *International Journal of Electrical, Computer, Electronics and Communication Engineering*, 8(5), 726-730.

- Camacho, E. F., & Bordons, C. (1999). *Model Predictive Control*. Londres: Ed. Springer-Verlag.
- Candia, A. (n.d.). *Cap. IV Valorizaciones de neutralización: Aplicaciones*. Retrieved 2016, from https://www.academia.edu/24332392/Cap_4_Vol._neutralizacion_aplicaciones
- Castaño, S. (2015). *Restricciones en el MPC*. Retrieved 2016, from <http://controlautomaticoeducacion.com/restricciones-en-el-mpc/>
- Cerquena, Y. (n.d.). *Interpolación con trazadores o splines*. Retrieved from <http://www.ilustrados.com/documentos/eb-splines%20cubicos.pdf>
- Cervantes, A., Agamennoni, O. E., & Figueroa, J. L. (2002). A nonlinear model predictive control system based on Wiener piecewise linear models. *Journal of Process Control*, 13, 656-666.
- Chen, J., Peng, Y., Han, W., & Guo, M. (2011). Adaptive fuzzy sliding mode control in pH neutralization process. *Procedia Engineering*, 15, 954-958.
- Clarke, D. W., Mothadi, C., & Tuffs, P. S. (1987). Generalized Predictive control- Part II. Extensions and interpretation. *Automatica*, 23(2), 149-160.
- Clarke, D., Mothadi, C., & Tuffs, P. (1987). Generalized predictive control - Part II. Extensions and interpretation. *Automatica*. 149-160.
- Cortés, C. A. (2013). Identificación de sistemas lineales y no lineales. *Tesis de maestría en Ingeniería Eléctrica. Universidad Tecnológica de Pereira*.
- Da Silva, D. (2009). Controle preditivo nao linear baseado no modelo de Hammerstein com prova de estabilidade. *Disertación doctoral. Universidade Federal do Rio Grande do Norte. Brasil*.
- Franco, I. C., Mota, A. S., & Menezes, M. R. (2015). Identificacao de sistemas nao-lineares dinámicos utilizando modelos autoregressivos aplicado em um proceso de neutralizacao de pH. *Blucher Chemical Engineering Proceedings*, 1(ISSN 2359-1757, <http://dx.doi.org/10.1016/chemeng-cobeq2014-0692-24404-147656>), 116338-645.
- Fruzzetti, K. P., Palazoglu, A., & McDonald, K. A. (1997). Nonlinear model predictive control using Hammerstein models. *Journal of process control*, 7(1), 31-41.
- García, C., & De Godoy, R. J. (2011). Modelling and simulation of pH neutralization plant including the process instrumentation. In T. Michalowski (Ed.), *Applications of MATLAB in Science and Engineering* (Claudio Garcia and Rodrigo Juliani Correa De Godoy (2011). Modelling and Simulation of pH Neutralization Plant Including the Process Instrumentation, Applications of DOI: 10.5772 ed., pp. 485-510). INTECH Open Acces Publisher.

- García, R. A., Rubio, F. R., & Ortega, M. G. (2012). Robust PID Control of the Quadrotor Helicopter. *IFAC Conference on Advances in PID Control*, (pp. 229-334).
- Giri, F., & Bai, E.-W. (2010). *Block-oriented Nonlinear System Identification*. Er-Wei Eds.
- Golub, G., & Van Loan, C. (1996). *Matrix computations* (Third ed.). Baltimore and London: The Johns Hopkins University Press.
- Gómez, J. C. (2005a). *Subspace Identification of Hammerstein and Wiener models*. Retrieved marzo 15, 2016, from http://www.fceia.unr.edu.ar/~jcgomez/isis/Seminar_LSD_2005.pdf
- Gómez, J. C., & Baeyens, E. (2004a). Identification of Block Oriented Nonlinear Systems using Orthonormal Bases. *Journal of Processs Control*, 14(6), 685-697.
- Gómez, J. C., & Baeyens, E. (2004b). Wiener model identification and predictive control of a pH neutralisation process. *IEE proceedings on Control Theory and Applications*, 151(3), 329-338.
- Gómez, J. C., & Baeyens, E. (2005b). Subspace Identification of multivariable Hammerstein and Wiener models. *European Journal of Control*, 11(2), 127-136.
- Gómez, J. C., Jutan, A., & Baeyens, E. (2004). Wiener model identification and predictive control of a pH neutralization process. *IEE Proceedings of the Control Theory Application*.
- Gruber, J. K., & Bordons, C. (2007). Control predictivo no lineal basado en modelos Volterra. Aplicación a una planta piloto. *RIAI*, 4(3), 34-45.
- Haber, R., & Keviczky, L. (1999). *Nonlinear system identification input-output model approach* (Vol. 1). Kluwer Acadmic Publishers.
- Hagenblad, A. (1999). Aspects of identification of Wiener models. *Thesis on Science Technology.Linkopings Universitet*. Suecia.
- Harris, D. (2007). Valoraciones ácido base. In *Análisis químico cuantitativo*. Editorial Reverté.
- Henson, M., & Seborg, D. a. (1994). Adaptive nonlinear control of a pH neutralization process. *IEEE Transactions on Control Systems of Technology*, 2(3), 169-182.
- Henson, M., & Seborg, D. b. (1997). *Nonlinear process control*. NJ: Prentice Hall.
- Hermansson, A. W., & Syafiiey, S. (2015). Model predictive control of pH processes. *Control Engineering Practice*, 45, 98-109.
- Hermansson, A. W., Syafiiey, S., & Noorz, S. M. (2010b). Multiple model predictive control of nonlinear pH neutralization system. *2010 IEEE International Conference* (pp. 301-304). Industrial Engineering and Engineering Management (IEEM).

- Hernández, A., De Keyser, R., Manrique, J., & Ipanaqué, W. (2014). Modelling and Nonlinear Model Predictive control of a rotary disc driver for fishmeal production. *European Control Conference*, 1819-1824.
- Hernandez, A., Keyser, R. D., Manrique, J., Oliden, J., & Ipanaqué, W. (2014). Modeling and Nonlinear Model Predictive Control. *European Control Conference (ECC)*. Strasbourg, France.
- Ibrahim, R. (2008). Practical modeling and control implementation studies on a pH neutralization process pilot plant. *Disertación doctoral. University of Glasgow. Escocia*.
- Ipanaqué Alama, W. (2005). *Control Predictivo: Una Técnica Para El Futuro Del A Industria*. Piura: Universidad de Piura.
- Ipanaque Alama, W., & Manrique, J. (2011). Identification and Control of pH using Optimal Piecewise Linear Wiener. *Proceeding 18th World Congress of the International Federation of Automatic Control (IFAC)*, (pp. 12301-12306). Milan, Italy.
- Ipanaqué, W. (2012). *Control automático de procesos. Innovando los procesos productivos*. Ed. Consejo Nacional de Ciencia y Tecnología. Lima: Ed. Consejo Nacional de Ciencia y Tecnología.
- Jalaleddini, K., & Kearney, R. (2011). An identification Algorithm for Hammerstein systems using Subspace Method. *American Control Conference*, 4793-4797.
- Jebarani, I. M., & Rammohan, T. (2014). Fuzzy Logic based PID controller for pH Neutralization Process. *International Journal of Computer Applications*, 95(16), 36-40.
- Julian, P. (1999). A high level canonical piecewise linear representation: Theory and applications. *Disertación doctoral. Universidad Nacional del Sur*. Bahía Blanca, Argentina.
- Julian, P., Agamennoni, O., & Desagues, A. (1999). High level canonical piecewise linear representation using a simplicial partition. *IEEE Transactions on Circuits Systems I*, 46, 466-480.
- Kambale, S., George, S., & Zope, R. G. (2015). Controllers used in pH Neutralization Process: A Review. *International Research Journal of Engineering and Technology (IRJET)*, 2(3), 354-361.
- Kawabe, T. (2012). Model Predictive PID Traction Control Systems for Electric Vehicles. *2012 IEEE International Conference and Control Applications*, 112-117.
- Khaled, E., & Abdessattar, C. (2010). Modelilng and estimation of Hammerstein System with Preload Nonlinearity. *Leonard Journal of Sciences*, 16(9), 13-20.

- Kokate, R. D., & Waghmare, L. M. (2011). Review of tuning methods of DMC and Performance Evaluation with PID Algorithms on a FOPDT Model. *International Journal of Control and Automation*, 4(2), 95-110.
- Kouvaritatis, B., & Cannon, M. (2001). *Nonlinear predictive control: Theory and Practice*. IEE Control Engineering Series 61.
- Kozek, M., & Jovanovic, N. (2008). Efficient modelling of Propofol pharmacokinetics: a novel 2-compartment model. *Proceedings of the BioMedSim03*, 113–118.
- Kozek, M., & Sinanovic, S. (2008). Identification of Wiener models using optimal local linear models. *Simulation Modelling Practice and Theory*, 16(8), 1055-1066.
- Ljung, L. (1999). *System Identification: Theory for the User*. 2nd. Ed. Prentice Hall.
- Long, R., Quan, S., Zhang, L., Chen, Q., Zeng, C., & Ma, L. (2015). Current sharing in parallel fuel cell generation system based on model predictive control. *International Journal of Hydrogen Energy*, 40(35), 11587-11594.
- Mahmoodi, S., Montazeri, A., Poshtan, J., & Jahed-Motlagh, M. R. (2007). Volterra-Laguerre modeling for NMPC. *Signal Processing and Its Applications* (pp. 1318-1321). New York: 9th International Symposium.
- Mahmoodi, S., Poshtan, J., Jahed-Motlagh, M., & Montazeri, A. (2009). Nonlinear model predictive control of a pH neutralization process based on Wiener--Laguerre model. *Chemical Engineering Journal*, 46, 328-337.
- Martin, A. M. (n.d.). Introducción al equilibrio ácido base. <http://materias.fi.uba.ar/6305/Acido-Base.pdf>.
- Medina, C. (2011). Identificación sistemas no lineales usando series de Volterra-Laguerre. *Tesis Maestría en Ciencias con mención en Automática e Instrumentación*. Universidad Nacional de Ingeniería. Lima, Perú.
- Mendoza, R., & Zambrano, G. (2012). Identificación y diseño del controlador para un sistema de control de pH en un proceso químico. *Tesis para título de Ingeniero Eléctrico*. Escuela Superior Politécnica del Litoral. Guayaquil, Ecuador.
- Miller, R. M., Shah, S. L., Wood, R. K., & Kwok, E. K. (1999). Predictive PID. *ISA Transactions*, 38(1), 11-23.
- Morari, M., & Lee, J. (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23, 667-682.
- Naseer, O., & Khan, A. A. (2013). Hybrid Fuzzy Logic and PID Controller for pH Neutralization Pilot Plant. *International Journal of Fuzzy Logic Systems*, 3(2).

- Neyra, J. (2009). Control predictivo no lineal basado en modelación Hammerstein polinomial aplicado a un módulo de ph. *Tesis para optar título ingeniero mecánico eléctrico*. Piura, Perú: Universidad de Piura.
- Norquay, S. J., Palazoglu, S. J., & Romagnoli, J. (1998). Model predictive control based on Wiener models. *Chemical Engineering Science*, 53(1), 75-84.
- Novak, J., Chalupa, P., & Bobal, V. (2011). Multiple model modeling and predictive control of the pH neutralization process. *International Journal of Mathematical Models and Methods in Applied Sciences*, 5(7), 1170-1179.
- Prasad, P., & Mathew, J. (2016). Performance analysis of pH neutralization process for conventional PI controller and IMC based PI controller. *IJIRST, International Journal for Innovative Research in Science & Technology*, 3(1), 262-267.
- Prautzsch, H., Boehm, W., & Paluszny, M. (2005). *Métodos de Bézier y B-Splines*. KIT Scientific Publishing (ISBN 3-937300-47-3).
- Priyanka, U., George, S., & Mobeen, S. A. (2015). Control of pH Neutralization Process using Fuzzy Logic: A Review. *International Research Journal of Engineering and Technology (IRJET)*, 2(3), 1081-1084.
- Purizaga, Z. (2007). Control predictivo no lineal basado en modelos Hammerstein y Wiener para pH. *Tesis para título Ingeniero Mecánico Eléctrico*. Universidad de Piura. Piur, Perú.
- Qin, G., Hu, L., & Song, L. (2012). The simulation and Analysisi of PID Structure Generalizaed Predictive Control Algorithm. *Journal of Convergence Information Technology*, 7(23), 700-707.
- Ramdani, A., & Grouni, S. (2017). Dynamic matrix control and generalized predictive control, comparison study with IMC-PID. *Interrnational Journal off Hyddrogen Energy*, 42(28), 17561-17570.
- Rocha, C., & Escorcía, J. (2010). Neutralización de los niveles de pH por medio de lógica difusa. *Eighth LACCEI Latin American And Caribbean Conference for Engineering and Technology*.
- Saeed, Q., Uddin, V., & Katebi, R. (2013). MIMO predictive PID control: a practical approach for quadruple tank. *Journal of Circuits, Systems and Computers*, 22(3), 11350009.
- Sahar, P., Krishman, S. H., Rao, V. S., & Patwardhan, S. C. (2004). Modelling and Predictive control of MIMO nonlinear systems using Wiener-Laguerre models. *Chemical Engineering Communications*, 191(8), 1083-1119.
- Sakthlya, S., Dinesh, D., & Meenakshipriya, B. (2016). Designing of PID Controllers for pH neutralization process. *Indian Journal of Science and Technology*, 3(12), 164-169.

- Silva, C. H., Henrique, H. M., & Oliveira-Lopes, L. C. (2012). Experimental Application of Predictive Controllers. *Journal of Control Science and Engineering*, 18.
- Sung, S. W., & Lee, J. (2004). *Modeling and control of Wiener-type processes*. *Chemical Engineering Science*.
- Uren, K., & Van Schoor, G. (2011). *Predictive PID Control of Non- Minimum Phase Systems*. (D. V. (Ed.), Editor) Retrieved febrero 2015, from <https://www.intechopen.com/books/advances-in-pid-control/predictive-pid-control-of-non-minimun-phase-systems>
- Wang, L. (2009). *Model Predictive Control System design and implementation using MATLAB*. Ed. Springer-Verlag.
- Wu, S. (2015). State Space predictive functional control optimization based new PID design for multivariable processes. *Chemometrics and Intellignce Laboratory Systems*, 143, 116-27.
- Ylén, J. P. (2001). Measuring, modelling and controlling the pH value and the dynamic chemical state. *Disertación doctoral en Science in Technology*. Helsinki University of Technology.
- Zhang, H., Wang, Y., Xiang, M., & Li, M. (2006). SVD-based Identification algorithm for Hammerstein- typed Nonlinear Systems. *Conference Intelligence Control and Automation*. VVCICA, 1.
- Zhang, R., Wu, S., Lu, R., & Gao, F. (2014). Predictive control optimization based PID control for temperature in an industrial surfactant reactor. *Chemometrics and Intelligent Laboratory Systems*, 135, 48-62.
- Zhu, Y. (2001). *Multivariable system identification for process control*. UK: Elsevier Science LTD.

Anexos

Anexo A. Artículo presentado en el 2017 CHILEAN Conference on Electrical, Electronics, Engineering, Information and Communication Technologies, 2017, Chile

Wiener Predictive Control for a pH neutralization plant

William Ipanaque Alama, *Member IEEE*
Departamento de Ingeniería Mecánica Eléctrica
Universidad de Piura
Piura, Perú
william.ipanaque@udep.pe

Irene Alvarado Tabacchi, *Member IEEE*
Departamento de Ingeniería Mecánica Eléctrica
Universidad de Piura
Piura, Perú
irene.alvarado@udep.pe

Abstract— This paper contains a new alternative to control a pH neutralization process using a PID control with predictive characteristics. The structure uses a Wiener model of a pH simulator that corresponds to a bench-scale plant at University of California, Santa Barbara (UCSB). The study concluded in a new alternative, named WPIDp (Wiener PID predictive) with good performance. We compare the new algorithm with a MPC linear predictive control evidencing the notorious advantages.

Keywords—pH, predictive control, PID, Wiener

I. INTRODUCCIÓN

El control PID es aún el más difundido en aplicaciones industriales. Sin embargo este tipo de regulador tiene inconvenientes en procesos con fuerte retardo, ceros a fase no mínima y procesos no lineales, entre otros. El control predictivo, supera varios de estos inconvenientes ([1] - [6]). Hay diversas estructuras de control predictivo que se distinguen por el modelo que usan y por la función de costo que optimizan ([7] - [12]).

Una de las ventajas del control predictivo son las buenas prestaciones que se obtienen y la capacidad de adaptarse también a procesos no lineales como es el proceso de neutralización donde la variable pH es relevante ([4], [9]-[11], [13]-[18]). Sin embargo, en la mayor parte de los casos el control predictivo requiere de una plataforma adecuada para su implementación, sobre todo cuando los modelos son más complejos ([6], [11]).

En este artículo se busca deducir un diseño de un PID incorporando un modelo del proceso, basándose en una simplificación para emular un regulador predictivo. En la literatura hay algunas alternativas presentadas ([12], [19] - [26]). La propuesta aquí presentada se basa en un modelo Wiener ([4], [9]-[11], [13], [15], [27], [28]) para emular el

comportamiento no lineal, partiendo de un control predictivo con formulación en variables de estado ([21], [23]) del cual se deduce un PID con la ventaja de poder usarse en una plataforma hardware tradicional pero con características de un regulador predictivo.

II. FUNDAMENTOS DEL CONTROL PID PREDICTIVO

La funcionalidad integral es importante en lazo de control también aparece en sistemas de control predictivo avanzados tales como el control predictivo basado en modelos (MPC) siendo uno de los más usados a nivel industrial el Control Predictivo Generalizado (GPC) ([7] - [10]) y el control de matriz dinámica (DMC) [12], con la ventaja que la acción integral elimina el offset o error con datos de predicción. La diferencia principal en cada uno de estos algoritmos es el modelo usado para hacer la predicción.

La idea de diseñar un control PID usando un algoritmo MPC que permita usar la estructura sencilla de un PID combinada con las características de comportamiento de un controlador predictivo ha sido estudiada en diversa literatura e implementada en algunos sectores industriales, obteniéndose muy buenos resultados. Algunos de los algoritmos predictivos utilizan la formulación del modelo con la estructura GPC, es decir, entrada y salida en discreto y otros usan la estructura de un MPC con modelo en espacio de estado.

Este trabajo presenta una nueva estrategia para el control PID predictivo usando la estructura de origen MPC y un modelo no lineal tipo Wiener, que está configurado con dos bloques en cascada. El primer bloque es lineal y a continuación el bloque no lineal que caracteriza la ganancia estática del proceso. En la literatura revisada se ha implementado el control PID predictivo para sistemas lineales usando estructura GPC, ([7], [8], [19], [20] [22], [24]), MPC en espacio de estado ([21], [23]), DMC [12] y PFC ([25], [26]).

El planteamiento que se hace y que se presenta en este artículo es la de simplificar la formulación de un modelo no lineal concentrando la no linealidad en la ganancia y la dinámica en un bloque lineal, cuyo modelo se usa para el diseño de un MPC. A su vez a partir de este control MPC inducir a una estructura PID para el cual se sintonizan adecuadamente los parámetros de sintonización. En consecuencia se tiene una

Esta investigación se ha desarrollado gracias al apoyo de la Universidad de Piura y a INNOVATE Perú, quien gracias al financiamiento del Proyecto: "Estudio y Desarrollo teórico experimental para obtención de planta piloto de biodigestor con parámetros controlados mediante aplicación de sistemas embebidos, para aprovechar el potencial energético y fertilizador de los residuos de la cáscara de cacao", CONVENIO 145-pnicp-piap-2015, ha motivado el estudio reportado en el presente artículo para resolver uno de los objetivos de medición y control presentados.

plataforma PID pero en su sintonización se ha usado el concepto de predicción con el modelo Wiener.

A. Formulación del MPC en espacio de estado

En el caso del MPC en espacio de estado se incluye la acción integral en el modelo a espacio de estado, como se desarrolla en [21], en donde se describe la planta en espacio de estado, como

$$\begin{aligned} X_m(k+1) &= A_m X_m(k) + B_m u(k) \\ y(k) &= C_m X_m(k) + D_m u(k) \end{aligned} \quad (1)$$

Siendo: $u(k)$ la variable de entrada, $y(k)$ la variable de salida y $X_m(k)$ el vector de la variable de estado de dimensión n .

Dado que se debe diseñar un control predictivo, que permita incluir la acción integradora, se modifica la expresión de manera que se incluya el incremento $\Delta u(k)$. Así mismo, en [21] se plantean las siguientes ecuaciones considerando una formulación con el principio del horizonte recesivo en el que se asume que la entrada $u(k)$ no puede afectar la salida $y(k)$ al mismo tiempo, por lo que se asume $D_m = 0$.

Haciendo la operación diferencia en ambos miembros de la ecuación de estado en la expresión (1) con esta simplificación, se tiene:

$$\begin{aligned} \frac{X_m(k+1) - X_m(k)}{\Delta X_m(k)} &= \frac{A_m (X_m(k) - X_m(k-1)) + B_m (u(k) - u(k-1))}{\Delta X_m(k)} \\ &= A_m + \frac{B_m}{\Delta u(k)} \end{aligned} \quad (2)$$

El vector de estado aumentado que incluye la acción integradora viene dado por:

$$X(k) = [\Delta X_m(k)^T \ y(k)^T]^T \quad (3)$$

Continuando con el análisis hecho en [21], la ecuación de salida también se despeja de:

$$\begin{aligned} (y(k+1) - y(k)) &= C_m (X_m(k+1) - X_m(k)) \\ &= C_m A_m \Delta X_m(k) + \tilde{C}_m B_m \Delta u(k) \end{aligned} \quad (4)$$

De esta manera se consigue el efecto integrador conectando $\Delta X_m(k)$ a la salida $y(k)$.

Con (2) y (4) se plantea la nueva forma de espacio de estado:

$$\begin{aligned} \begin{bmatrix} \Delta X_m(k+1) \\ y(k+1) \end{bmatrix}_{X(k+1)} &= \begin{bmatrix} A_m & 0 \\ C_m & 1 \end{bmatrix} \begin{bmatrix} \Delta X_m(k) \\ y(k) \end{bmatrix}_{X(k)} \\ &+ \begin{bmatrix} B_m \\ \tilde{C}_m B_m \end{bmatrix} \Delta u(k) \end{aligned} \quad (5)$$

$$y(k) = \frac{[0_m \ 1]}{c} \begin{bmatrix} \Delta X_m(k) \\ y(k) \end{bmatrix} \quad (6)$$

Donde $0_m = [0 \ 0 \ \dots \ 0]$ es un vector $1 \times n$. La tripleta (A, B, C) del modelo aumentado se usará para el diseño del control predictivo.

A partir de la formulación matemática, se calcula la salida predicha de la planta con la señal de control futura como variable manipulable. Se sabe que todas las variables de estado no son medibles o accesibles y será necesario diseñar un estimador de estado; pero en este artículo se asume que el estado $X(k)$ es disponible y proporciona la información total y actual del proceso.

Según [21] la trayectoria de control futura dentro del horizonte de control N_c ($N_c \leq N_p$) se designa como:

$$\Delta U = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_c-1)]^T \quad (7)$$

Si el horizonte de predicción es N_p , la información de las variables de estado futuras se puede predecir desde el instante k :

$$X(k+1|k), X(k+2|k), \dots, X(k+m|k), \dots, X(k+N_p|k) \quad (8)$$

De (5) se deduce que:

$$X(k+1|k) = AX(k) + B\Delta u(k) \quad (9)$$

Aplicando recursivamente la (9) hasta N_p y asumiendo que: $\Delta u(k+N_c) = \Delta u(k+N_c+1) = \dots = \Delta u(k+N_p-1) = 0$, se pueden calcular los estados futuros de la planta definidos en (8), por lo que se puede escribir $X(k+N_p|k)$, como sigue:

$$\begin{aligned} X(k+N_p|k) &= A^{N_p} X(k) + A^{N_p-1} B \Delta u(k) \\ &+ A^{N_p-2} B \Delta u(k+1) + A^{N_p-3} B \Delta u(k+2) \\ &+ A^{N_p-N_c+1} B \Delta u(k+N_c-2) \\ &+ A^{N_p-N_c} B \Delta u(k+N_c-1) \end{aligned} \quad (10)$$

Haciendo lo mismo con las variables de salida predichas, usando (10) se puede escribir en forma matricial, para todo el rango de predicción:

$$Y = F X(k) + \Phi \Delta U \quad (11)$$

Donde:

$$Y = [y(k+1|k) \ y(k+2|k) \ \dots \ y(k+N_p|k)]^T$$

$$F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_p} \end{bmatrix}$$

$$\Phi = \begin{bmatrix} CB & 0 & \dots & 0 \\ CAB & CB & \dots & 0 \\ \vdots & \vdots & \dots & 0 \\ CA^{N_p-1}B & CA^{N_p-2}B & \dots & CA^{N_p-N_c}B \end{bmatrix} \quad (12)$$

Luego se asume una entrada de referencia $r(k)$ constante durante la ventana de optimización N_p y la siguiente función de costo:

$$J = (r - Y)^T(r - Y) + \Delta U^T \bar{R} \Delta U \quad (13)$$

En el que r viene dada por:

$$r^T = [1 \quad 1 \quad \dots \quad 1] \cdot r(k) \quad (14)$$

Y la dimensión de r es $N_p \times 1$. La función de costo (13) tiene dos partes, la primera minimiza los errores entre la referencia y la salida; la segunda parte se enfoca a minimizar los esfuerzos de control. $\bar{R}$ es una matriz diagonal de pesos que viene dada por: $\bar{R} = \lambda I$, donde I es una matriz identidad de dimensión $N_c \times N_c$ y el peso $\lambda \geq 0$ se usa para sintonizar la respuesta a lazo cerrado.

Para el algoritmo PID predictivo se dará también un peso al error. El problema de optimización consiste en encontrar ΔU minimizando la función de costo J , de modo que se obtiene:

$$\frac{\partial J}{\partial \Delta U} = -2\Phi^T(r - F X(k)) + 2(\Phi^T \Phi + \bar{R}) \Delta U \quad (15)$$

Se aplica la condición de minimización $\frac{\partial J}{\partial u} = 0$, y la ley de control óptima:

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (r - F X(k)) \quad (16)$$

En (16) el término $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T r$ corresponde al cambio en la referencia, mientras que $-(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F$ corresponde al control de realimentación de estado dentro del marco del control predictivo. Ambos dependen de los parámetros del sistema, por tanto son matrices constantes para un sistema invariante en el tiempo. Por el principio del control recesivo sólo se toma el primer elemento de ΔU en el instante k como el control incremental, por tanto:

$$\Delta u(k) = [1 \quad 0 \quad \dots \quad 0](\Phi^T \Phi + \bar{R})^{-1} \Phi^T R_s r(k) - \Phi^T F X(k) \quad (17)$$

Donde $R_s^T = [1 \quad 1 \quad \dots \quad 1]$, de dimensión $1 \times N_p$. La ley de control se puede escribir como:

$$\Delta u(k) = K_y r(k) - K_{GPC} X(k) \quad (18)$$

En (18), K_y es el primer elemento de $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T R_s$ y K_{GPC} es la primera fila de $(\Phi^T \Phi + \bar{R})^{-1} \Phi^T F$.

$$K_{GPC} = [K_x \quad K_y] \quad (19)$$

Ahora K_x corresponde al vector de realimentación de $\Delta x_m(k)$ y K_y corresponde a la ganancia de realimentación de $y(k)$.

B. Formulación del PID predictivo

Basándonos en las matrices encontradas para el MPC, se plantea el control PID predictivo usando la ley:

$$u(k) = k_p e(k) + T k_i \sum_{j=1}^k e(j) + \frac{k_D}{T} [e(k) - e(k-1)] \quad (20)$$

Donde k_p , k_i y k_D son las ganancias proporcional, integral y derivativa, T el tiempo de muestreo y $e(k)$ el error en el instante k . Tomando las diferencias de ambos lados en el paso k y $k+1$ se obtiene la llamada ley de control PID en la forma velocidad:

$$\Delta u(k) = u(k) - u(k-1) = k_p [e(k) - e(k-1)] + T k_i e(k) + \frac{k_D}{T} [e(k) - 2e(k-1) + e(k-2)] \quad (21)$$

$$\Delta u(k) = \left[k_p + T k_i + \frac{k_D}{T} \right] e(k) - \left[k_p + 2 \frac{k_D}{T} \right] e(k-1) + \frac{k_D}{T} e(k-2) \quad (22)$$

Transformando en el dominio z se obtiene:

$$U(z) = \frac{[p_0 + p_1 z^{-1} + p_2 z^{-2}] E(z)}{1 - z^{-1}} \quad (23)$$

Con:

$$p_0 = \left(k_p + T k_i + \frac{k_D}{T} \right), p_1 = - \left(k_p + 2 \frac{k_D}{T} \right), p_2 = \frac{k_D}{T} \quad (24)$$

O lo que es lo mismo:

$$k_p = -p_1 - 2p_2, \quad T k_i = p_0 + p_1 + p_2, \quad \frac{k_D}{T} = p_2 \quad (25)$$

Para encontrar el algoritmo en su forma predictiva, primero se hace el desarrollo de la ley de control PID dada por (22) en términos de la entrada de referencia $r(k)$ y el vector de estado $x_m(k)$ y la entrada $u(k)$. Por simplicidad, asumiremos $T = 1$, dado que es el tiempo de muestreo usado para la identificación del modelo Wiener de la planta a controlar.

Se tiene entonces:

$$\begin{aligned}\Delta u(k) &= [k_p + k_i + k_d]e(k) \\ &- [k_p + 2k_d]e(k-1) + k_d e(k-2) \\ &= p_0 e(k) + p_1 e(k-1) + p_2 e(k-2)\end{aligned}\quad (26)$$

Considerando que $e(k) = r(k) - y(k)$ y asumiendo que $\dots = r(k-1) = r(k) = r(k+1) = \dots$, se puede escribir la (26) como:

$$\Delta u(k) = (p_0 + p_1 + p_2)r(k) - p_0 y(k) - p_1 y(k-1) - p_2 y(k-2)\quad (27)$$

Desarrollando la salida en términos de las variables de estado y la variable manipulable, y usando (1), se tiene:

$$\begin{aligned}y(k) &= C_m X_m(k) = C_m A_m^2 X_m(k-2) \\ &+ C_m A_m B_m u(k-2) + C_m B_m u(k-1)\end{aligned}\quad (28)$$

Usando el mismo método para $y(k-1)$ y $y(k-2)$, se tiene:

$$y(k-1) = C_m A_m X_m(k-2) + C_m B_m u(k-2)\quad (29)$$

$$y(k-2) = C_m X_m(k-2)\quad (30)$$

Reemplazando las ecuaciones desde la (28) a la (30) en la (27) se obtiene:

$$\Delta u(k) = a_0 r(k) - a_1 X_m(k-2) - a_2 u(k-2) - p_0 C_m B_m u(k-1)\quad (31)$$

Donde $a_0 = (p_0 + p_1 + p_2)$, $a_1 = [p_0 C_m A_m^2 + p_1 C_m A_m + p_2 C_m]$, $a_2 = [p_0 C_m A_m B_m + p_1 C_m B_m]$.

Para que el algoritmo desarrollado en (31) tenga carácter predictivo debe ser equivalente a la ley de control MPC óptimo dada por (18). Por ello desarrollando dicha expresión se obtiene:

$$\Delta u(k) = K_y r(k) - [K_x \quad K_y] \begin{bmatrix} \Delta X_m(k) \\ y(k) \end{bmatrix}\quad (32)$$

Y en términos de las variables de estado originales del proceso:

$$\Delta u(k) = K_y r(k) - K_x (X_m(k) - X_m(k-1)) - K_y y(k)\quad (33)$$

Para igualar esta expresión con la encontrada del PID en (31) se opera reemplazando las variables de estado $x_m(k)$, $x_m(k-1)$ y la salida $y(k)$ en términos del estado $x_m(k-2)$ y las entradas pasadas $u(k-1)$ y $u(k-2)$, usando las ecuaciones de salida y estado del modelo del proceso. De esta forma se tiene:

$$u(k) = K_y r(k) - S_1 X_m(k-2) - S_2 u(k-2) - S_3 B_m u(k-1)\quad (34)$$

Donde $S_1 = [(K_x + K_y C_m) A_m^2 + K_x A_m]$, $S_2 = [(K_x + K_y C_m) A_m B_m + K_x B_m]$ y $S_3 = (K_x + K_y C_m)$.

Los parámetros de (34) se igualan con los de la ecuación (31) y se obtienen los valores de los parámetros p_0 , p_1 y p_2 y por ende la de los parámetros k_p , k_i , k_d del controlador PID.

III. SIMULACIONES

El algoritmo Wiener PID predictivo se implementa usando el modelo Wiener de una planta de neutralización de pH de la Universidad de California (UCSB) encontrada en [9], con las siguientes características:

Bloque lineal:

$$A_m = \begin{bmatrix} 2.921 & -2.845 & 0.9226 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B_m = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\quad (35)$$

$$C_m = [0.0251 \quad -0.0488 \quad 0.0237], D_m = 0\quad (36)$$

Bloque no lineal:

$$N^{-1}(y_k) = v_k = 0.1988 y_k + 0.4065 y_k^2 + 0.8918 y_k^3\quad (37)$$

El modelo Wiener (Fig. 1) representa a la planta con dos bloques. El primer bloque representa la dinámica de la planta que se asume como la componente lineal y el bloque siguiente corresponde al comportamiento no lineal, que caracteriza las condiciones estáticas. La representación en el caso del modelo identificado en [9] considera el bloque lineal en espacio de estado y se estima el bloque no lineal como una aproximación polinómica. El método usado para la identificación consiste en estimar simultáneamente los parámetros de los dos bloques (lineal y no lineal) con un algoritmo que usa la identificación en subespacio de estado N4SID para estimar las matrices de un sistema en espacio de estado equivalente que incluyan los parámetros del bloque no lineal también a estimar:

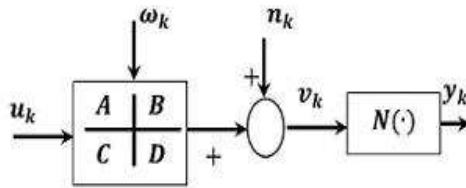


Fig. 1 Modelo Wiener en espacio de estado

Diversos autores han usado una planta de neutralización de pH para identificación y control del procesos, por las características de no linealidad [9], [10], [11], [13], [14], [15], [16], [17]). El diagrama esquemático de la planta de neutralización de pH de la UCSB se muestra en la Fig. 2.

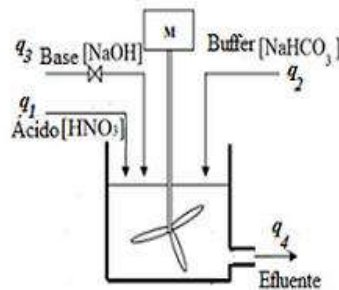


Fig. 2. Esquema de una planta de neutralización de pH

El objetivo de control es regular el pH de la solución neutra de salida, q_4 a través de la manipulación del flujo de la solución base, q_3 (NaOH), que se mezcla con una solución ácida, q_1 (HNO₃), compensando los posibles cambios en el flujo del buffer, q_2 (NaHCO₃) que se considera como una perturbación no medible, que entran a un tanque agitado. El flujo de la solución ácida, q_1 y el volumen del tanque (V) se asumen constantes. Se consideran las ecuaciones de balance de carga y la del ión carbonato como los momentos invariantes de reacción tanto para los flujos de entrada como de salida. Dichos invariantes son denotados como:

W_{a1}, W_{b1} : invariantes de la solución ácida

W_{a2}, W_{b2} : invariantes de la solución buffer

W_{a3}, W_{b3} : invariantes de la solución base

W_{a4}, W_{b4} : invariantes de la solución efuyente

Las ecuaciones de balance (descritas en [9], [11] y [13]) se resumen en:

$$\frac{dW_{a4}}{dt} = \frac{1}{V}(W_{a1} - W_{a4})q_1 + \frac{1}{V}(W_{a2} - W_{a4})q_2 + \frac{1}{V}(W_{a3} - W_{a4})q_3 \quad (38)$$

$$\frac{dW_{b4}}{dt} = \frac{1}{V}(W_{b1} - W_{b4})q_1 + \frac{1}{V}(W_{b2} - W_{b4})q_2 + \frac{1}{V}(W_{b3} - W_{b4})q_3 \quad (39)$$

$$W_{a4} - \frac{10^{-pH_4} + 10^{pH_4-14}}{1 + 2 \cdot 10^{pH_4-pK_2}} + W_{b4} \frac{1 + 10^{pK_1-pH_4} + 10^{pH_4-pK_2}}{1 + 10^{pK_1-pH_4} + 10^{pH_4-pK_2}} = 0 \quad (40)$$

Donde:

q_1 : Caudal de la solución ácida

q_2 : Caudal de la solución buffer

q_3 : Caudal de la solución básica

V : Volumen del tanque

El pH de la solución efuyente pH_4 se determina, a partir de las invariantes de reacción W_{a4} y W_{b4} . Las constantes de disociación del ácido débil H_2CO_3 , vienen dadas por pK_1 y pK_2 .

Los parámetros de la planta de neutralización de pH en condiciones de operación se muestran a continuación.

TABLA I CONDICIONES DE OPERACIÓN NOMINAL DE LA PLANTA

$q_1=16.6$ ml/s	$W_{b1}=0$ mol
$q_2=0.55$ ml/s	$W_{b2}=3 \times 10^{-2}$ mol
$q_3=15.55$ ml/s	$W_{b3}=5 \times 10^{-5}$ mol
$W_{a1}=3 \times 10^{-3}$ mol	$W_{b4}=5.28 \times 10^{-4}$ mol
$W_{a2}=3 \times 10^{-2}$ mol	$V=2900$ ml
$W_{a3}=3.05 \times 10^{-3}$ mol	$pK_1=6.35$
$W_{a4}=4.32 \times 10^{-4}$ mol	$pK_2=10.25$

La Fig. 3 representa la implementación en Simulink.

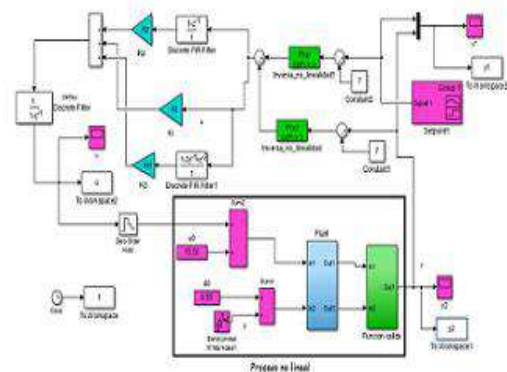


Fig. 3. Modelo de simulación control WPIID predictivo

Con las ecuaciones descritas desde la (38) a la (40) y teniendo como datos las condiciones de operación especificados en la Tabla I, se implementa el control PID predictivo con modelo Wiener, (WPIDp), como se muestra en la Fig. 4. Dado que el PID predictivo se ha diseñado para un proceso lineal, se aplica la inversa de la no linealidad tanto a la entrada de referencia ref como a la salida y .

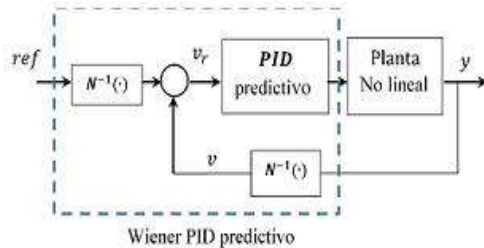


Fig. 4. Esquema control WPID predictivo

En la Fig. 5 se observa la respuesta de la planta ante variaciones en el caudal de la solución base con el algoritmo de control PID predictivo usando el modelo Wiener descrito y la respuesta del control MPC basado en un modelo lineal encontrado en [9]. Los parámetros del control predictivo son los siguientes: horizonte de control $N_c = 3$, horizonte de predicción, $N_p = 15$ y el factor de peso $\lambda = 0.5$ con los que se hallan las ganancias MPC.

En el caso del PID predictivo se usa el bloque lineal del modelo Wiener y a partir de allí se encuentran los parámetros del PID: $K_p = 11.8186$, $k_i = 1.4634$, $k_d = 0.0036$. Se puede notar un mejor comportamiento del control WPIDp, que responde más rápido que el MPC aunque con una sobreoscilación ligeramente mayor. Se puede observar que la variable manipulable u_c o esfuerzo de control es también ligeramente mayor. En el caso del control WPIDp se tiene: $u_{c,max} = 26.01$, $u_{c,min} = 11.24$, mientras que para el control MPC se tiene: $u_{c,max} = 22.91$, $u_{c,min} = 11.53$.

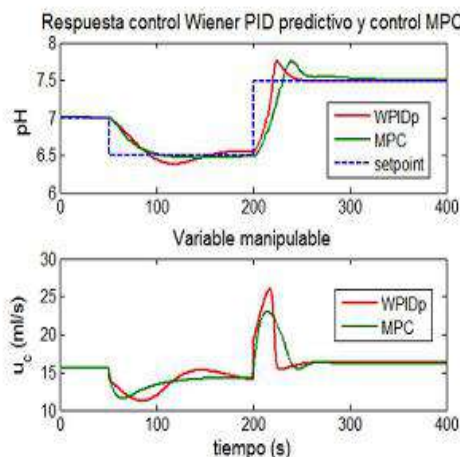


Fig. 5. Respuesta del control WPID predictivo y el control MPC ante pequeñas variaciones alrededor del punto nominal

En la Fig. 6 se observa que ante variaciones mayores alrededor de las condiciones nominales, el comportamiento del control MPC se degrada y no responde adecuadamente. Esto se explica porque la sintonización del control MPC se realiza basándose en un modelo lineal encontrado para variaciones relativamente pequeñas alrededor del punto nominal. Los parámetros usados en este caso para sintonizar tanto el WPIDp como el control MPC son: $N_c = 3$, $N_p = 10$ y $\lambda = 1.5$. Para el control WPIDp se obtiene: $K_p = 4.088$, $k_i = 0.6627$ y $k_d = 0.001$. Se ha logrado un control PID cuya sintonización ha requerido los criterios de selección de un MPC variando los horizontes de control, horizontes de predicción y el factor de peso λ . Es claro que el esfuerzo de control para conseguir mejores prestaciones en el caso del control WPIDp es mayor que en el caso del control MPC. Para analizar las características del control implementado se han aplicado las siguientes perturbaciones: una variación negativa del 30% sobre el valor nominal del caudal de la solución buffer a los 150s, y variación positiva del 50% sobre el caudal de la solución ácida a los 250s. Se observa de la Fig. 7, que el control WPIDp responde rápidamente retornando la salida al valor de referencia.

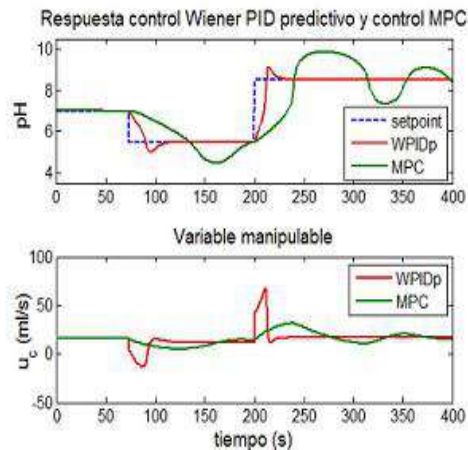


Fig. 6. Respuesta del control WPID predictivo y control MPC para grandes variaciones de pH

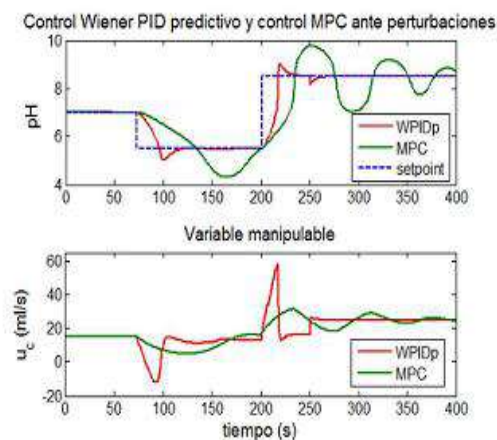


Fig. 7. Respuesta del control WPID predictivo y del control MPC ante perturbaciones en el caudal de la solución buffer y el caudal del ácido.

De las respuestas mostradas se evidencia que la variable manipulable, u_c toma valores negativos. Se debe prestar atención pues es normal que en el proceso real existan ciertas restricciones originadas por ejemplo por los límites de actuación de válvulas, motores o bombas, las cuales deben tomarse en consideración en la implementación del control.

IV. CONCLUSIONES

Se presenta el diseño de un algoritmo PID cuyos parámetros de diseño se han calculado usando la estrategia de control predictivo MPC, con una sintonización sencilla. Las prestaciones del control PID predictivo aplicado a un modelo no lineal se han conseguido usando un modelo Wiener con una estructura que ha facilitado la implementación en una planta de neutralización de pH y que se comporta mucho mejor que un control con características predictivas sintonizado para un modelo lineal como es el caso del control MPC. La estructura planteada facilita la implementación en procesos no lineales con las ventajas predictivas de un control MPC por su simplicidad de sintonización como se ha mostrado en este trabajo.

REFERENCIAS

- [1] A. Ramdani y S. Grouni, «Dynamic matrix control and generalized predictive control, comparison study with IMC-PID», *International Journal of Hydrogen Energy*, vol. 42, nº 28, pp. 17561-17570, 2017.
- [2] R. Long, S. Qian, L. Zhang, Q. Chen, C. Zeng y L. Ma, «Current sharing in parallel fuel cell generation system base on model predictive control», *International Journal of Hydrogen Energy*, vol. 40, nº 35, pp. 11587-11594, 2015.
- [3] C. Silva, H. Henrique y L. Oliveira, «Experimental Application of Predictive Controllers», *Journal of Control Science and Engineering*, nº doi:10.1155/2012/159072, 2012.
- [4] S. J. Norquay, A. Palazoglu y J. Romagnoli, «Model predictive control based on Wiener models», *Chemical Engineering Science*, vol. 53, nº 1, pp. 75-84, 1998.
- [5] P. Parra, W. Ipanaque y J. Manrique, «Design and implementation of a Temperature Predictive Controller for a Dryer Plant of Cocon Beans», *2015 Asia-Pacific Conference on Computer Aided System Engineering*, pp. 202-207, 2015.
- [6] A. Hernández, R. De Keyser, J. Manrique, J. Oliden y W. Ipanaque, «Modeling and Nonlinear Model Predictive Control of a rotary disc dryer for fishmeal production», *European Control Conference*, pp. 1819-1824, 2014.
- [7] S. Boujnah, N. Khraief, E. Maherzi, E. Besbes y S. Belghith, «Predictive PID Control Based on GPC Control of Inverted Pendulum», *Research Journal of Applied Sciences, Engineering and Technology*, vol. 7, nº 20, pp. 4319-4326, 2014.
- [8] D. W. Clarke, C. Mothadi y P. S. Tuffs, «Generalized Predictive control-Part II. Extensions and interpretation», *Automatica*, vol. 23, nº 2, pp. 149-160, 1987.
- [9] J. C. Gómez, A. Jutan y E. Baeyens, «Wiener model identification and predictive control of a pH neutralization process», *IEE Proceedings-Control Theory Applications*, vol. 151, nº 3, pp. 329-338, 2004.
- [10] A. W. Hemmanson y S. Syafisey, «Model predictive control of pH processes: A review», *Control Engineering Practice*, vol. 45, pp. 98-109, 2015.
- [11] S. Mahmoodi, J. Poshtan, M. Jafar-Motlagh y A. Montazeri, «Nonlinear model predictive control of a pH neutralization process based on Wiener-Laguerre models», *Chemical Engineering Journal*, vol. 146, nº 3, pp. 328-337, 2009.
- [12] R. D. Kokate y L. M. Waghmare, «Review of tuning Methods of DMC and Performance Evaluation with PID Algorithms on a FOPDT Model», *International Journal of Control and Automation*, vol. 4, nº 2, pp. 95-110, 2011.
- [13] S. I. Biagiola, O. E. Agamennoni y J. L. Figueroa, «Robust control of Wiener systems: application to a pH neutralization process», *Brazilian Journal of Chemical Engineering*, vol. 33, nº 1, pp. 145-153, 2016.
- [14] C. García y R. De Godoy, «Modelling and simulation of pH neutralization plant including the process instrumentation», *Applications of MATLAB in Science and Engineering*, pp. 485-510, 2011.
- [15] W. Ipanaque y J. Manrique, «Identification and control of pH using optimal piecewise linear Wiener models», *IFAC Proceedings Volumes*, vol. 18, nº Part 1, pp. 1819-1824, 2011.
- [16] P. Prasad y J. Mathew, «Performance analysis of pH neutralization process for conventional PI controller and IMC based PI controllers», *International Journal for Innovative Research in Science & Technology*, vol. 3, nº 1, pp. 262-267, 2016.
- [17] S. Sakthiya, D. Dinesh y B. Meenakshipriya, «Designing of Controllers for pH Neutralization Process», *International Journal for Research in Applied Science and Engineering Technology*, vol. 3, nº 12, pp. 164-169, 2015.
- [18] O. Naseer y A. Khan, «Hybrid Fuzzy Logic and PID controller for pH Neutralization Pilot Plant», *International Journal of Fuzzy logic systems*, vol. 3, nº 2, pp. 1-14, 2013.
- [19] M. R. Katebi y M. H. Moradi, «Predictive PID Controllers», *IEE Proceedings-Control Theory Applications*, vol. 148, nº 6, pp. 478-487, 2001.
- [20] R. Miller, S. Shah, R. Wood y E. Kwok, «Predictive PID», *ISA Transactions*, vol. 38, nº 1, pp. 11-23, 1999.
- [21] K. Uren y G. V. Schoor, «Predictive PID Control of Non-minimum Phase Systems», *InTech*, 2011.
- [22] Q. Saeed, V. Uddin y R. Katebi, «MIMO predictive PID control: a practical approach for quadruple tank», *Journal of Circuits, Systems and Computers*, vol. 22, nº 3, 2013.
- [23] T. Kawabe, «Model Predictive PID Traction Control Systems for Electric Vehicles», *2012 IEEE International Conference on Control Applications*, pp. 112-117, 2012.
- [24] G. Qin, L. Hu y L. Song, «The simulation and Analysis of PID Structure Generalized Predictive Control Algorithms», *Journal of Convergence Information Technology*, vol. 7, nº 23, pp. 700-707, 2012.
- [25] S. Wu, «State space predictive functional control optimization based new PID design for multivariable processes», *Chemosensors and Intelligence Laboratory Systems*, vol. 143, pp. 16-27, 2015.
- [26] R. Zhang, S. Wu, R. Lu y F. Gao, «Predictive control optimization based PID control for temperature in an industrial surfactant reactor», *Chemosensors and Intelligent Laboratory Systems*, vol. 135, pp. 48-62, 2014.
- [27] M. Kozek y S. Simanovic, «Identification of Wiener models using optimal local linear models», *Simulation Modeling Practice and Theory*, vol. 16, nº 8, pp. 1055-1066, 2008.
- [28] S. Sung y J. Lee, «Modeling and control of Wiener-type processes», *Chemical Engineering Science*, vol. 59, nº 7, pp. 1515-1521, 2004.

Anexo B. Programas de identificación y control

Anexo B1

Programa de identificación Wiener método N4SID con descomposición SVD

```
clc
disp('carga datos');

load('E:\MAESTRIA IME\datos\u_pH22setd'); % Datos entrada
load('E:\MAESTRIA IME\datos\y_pH22setd'); % datos salida
%% tiempo de muestreo usado 1s
Ts=input('tiempo de muestreo de los datos ');
% genera datos para identificacion y validacion de los modelos
datos1=iddata(pH,u,Ts);
%% longitud de datos usados 10500
disp('longitud de los datos ');
N=length(pH)
datos=datos1(1:N-2);
plot(datos)
%elimina el offset de los datos
datosd=detrend(datos,1);

%selecciona datos estimación y validación
%=====
nmin_e=input('ingrese el minimo valor del rango de estimacion: ');
nmax_e=input('ingrese el valor maximo del rango de estimacion ');
datose=datosd(nmin_e:nmax_e); %datos para identificacion
nmin_v=input('ingrese el valor minimo del rango de validacion ');
datosv=datosd(nmin_v:length(datosd.u)); % datos para validacion

% Modelo Lineal con N4SID
%=====

%Ajusta el mejor modelo lineal estimación
modelo=n4sid(datose,2,'focus','prediction','n4weight','MOESP');
[A,B,C,D]=ssdata(modelo);
[num,den]=ss2tf(A,B,C,D);

model0=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy0, fit0]=compare(datose,model0);
compare(datose,model0)

%Grafica datos del modelo lineal para validación
yhat=dlsim(A,B,C,D,datosv.u);
modell=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy1, fit1]=compare(datosv,model1);% validacion cruzada
compare(datosv,model1)
merror1=norm(datosv.y-yhat)^2/length(yhat);

%MODELO WIENER

% Ajuste de datos para identificación no lineal
```

```

% Aproximación polinómica (tercer orden)
%=====
g1=(datose.y);
g2=(datose.y).^2;
g3=(datose.y).^3;

% Calculo de bloque no lineal con SVD modelo Wiener
%=====
Y=[g1 g2 g3];
datos2=iddata(Y,datose.u,Ts);
modelo2=n4sid(datos2,2,'focus','prediction','n4weight','MOESP');
[A2,B2,C2,D2]=ssdata(modelo2);
thetcd=[C2 D2];

%Descomposición SVD minima
[U1,S1,V1] = svd(thetcd,'econ');
% m=1 orden del vector de salida
% n : orden del modelo lineal
% p: dimension vector de entrada
% L orden del polinomio bloque no lineal

U1r=U1(1:3,1); % U1r dimension Lmxm,
ALFA=pinv(U1r) % dimension Lm*m
V1T=V1(:,1); % dim (n+p)xm
cd=S1(1,1)*V1T' % vector [C D]=S1rXV1T' S1 dim mxm
% matrices bloque lineal
nr=length(A2);
C2M=cd(1,1:nr);D2M=cd(1,nr+1);
% Funcion transferencia modelo bloque lineal
[num1,den1]=ss2tf(A2,B2,C2M,D2M)

%Identificacion bloque no lineal
vvu=dlsim(A2,B2,C2M,D2M,datosv.u); %
yy=[-3.5:0.1:3.5]'; % rango variación de la salida
vvy=ALFA(1,1)*yy+ALFA(1,2)*yy.^2+ALFA(1,3)*yy.^3;
alfa=fliplr(ALFA);
l=[alfa 0];
% grafica bloque no lineal PARA TODO EL RANGO DE pH
figure
plot(yy,vvy,'LineWidth',2)
title('Característica inversa bloque no lineal modelo Wiener','FontSize',12.5)
xlabel('pH-7','FontSize',12.5)
ylabel('v-vv0','FontSize',12.5)
yhat4=interp1(vvy,yy,vvu,'pchip');

% Compara salida del modelo con salida real
figure
Fit=goodnessOfFit(yhat4,datosv.y,'NRMSE')*100;
merror2=norm(datosv.y-yhat4)^2/length(yhat4);
tv=length(datosv.y);
time=nmin_v*Ts:Ts:nmin_v*Ts+Ts*tv-1;
plot(time,[datosv.y yhat4],'LineWidth',2)
title('Respuesta modelo Wiener con método SVD','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
xlabel('tiempo(s)','FontSize',12.5)
text(6000,-0.1,['Fit(%) :',num2str(Fit)],'EdgeColor','blue','LineStyle','-')

```

```

legend('salida real (validacion)', 'salida modelo')
%% calculo bloque no lineal aproximacion polinomial
p=polyfit(vvy,yy,15);
xx=linspace(min(vvy),max(vvy),100);
pp2=polyval(p,xx);
figure, plot(vvy,yy,'bd',xx,pp2,'r','LineWidth',2)
    title('Caracteristica bloque no lineal modelo Wiener','FontSize',12.5)
    legend('datos ajuste','aprox. polinomial')
xlabel(' v-v0','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
%% calculo bloque no lineal interpolacion
xx=linspace(min(vvy),max(vvy),30);
ys=interp1(vvy,yy,xx,'pchip');

figure, plot(vvy,yy,'bd',xx,ys,'r','LineWidth',2)
    title('Caracteristica bloque no lineal modelo Wiener','FontSize',12.5)
    legend('datos ajuste','aprox. interpolacion')
xlabel(' v-v0','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)

```

Anexo B2

Programa de identificación Hammerstein método N4SID con descomposición SVD

```

clc
disp('carga datos');

load('E:\MAESTRIA IME\datos\u_pH22setd'); % Datos de entrada
load('E:\MAESTRIA IME\datos\y_pH22setd'); % Datos de salida
%% Tiempo de muestreo seleccionado 1s
Ts=input('tiempo de muestreo de los datos ');
% genera datos para identificacion y validacion de los modelos
datos1=iddata(pH,u,Ts);

% Longitud de datos 10500
disp('longitud de los datos ');
N=length(pH)
datos=datos1(1:N-5);
plot(datos)
%elimina el offset de los datos
datosd=detrend(datos,1);

%selecciona datos estimación y validación
=====
nmin_e=input('ingrese el minimo valor del rango de estimacion: ');
nmax_e=input('ingrese el valor maximo del rango de estimacion ');
datose=datosd(nmin_e:nmax_e); %datos para identificacion
nmin_v=input('ingrese el valor minimo del rango de validacion ');
nmax_v=input('ingrese el valor maximo del rango de validacion ');
datosv=datosd(nmin_v:nmax_v); % datos para validacion

% genera componente no lineal
%Aproximación polinómica orden 7)
g1=(datose.u).^1;
g2=(datose.u).^5;
g3=(datose.u).^7;

%
%parametros auxiliares de bloque no lineal
% Modelo Lineal con N4SID
%
modelo=n4sid(datose,2,'focus','prediction','n4weight','CVA');
[A,B,C,D]=ssdata(modelo);
[num,den]=ss2tf(A,B,C,D);
yhat=dlsim(A,B,C,D,datose.u);
model=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
[yy0, fit0]=compare(datose,model);
figure
compare(datose,model)

% validacion del modelo
[yy2, fit2]=compare(datosv,model);
figure
compare(datosv,model) % validacion cruzada
yhat1=dlsim(A,B,C,D,datosv.u);
Fit=goodnessOfFit(yhat1,datosv.y,'NRMSE')*100;
merror1=norm(datosv.y-yhat1)^2/length(yhat1);

% Calculo de bloque no lineal con SVD modelo Hammerstein

```

```

U=[g1 g2 g3];
datos1=iddata(datosv.y,U,1);
modelo2=n4sid(datos1,2,'focus','prediction','n4weight','CVA');
[A2,B2,C2,D2]=ssdata(modelo2);
thetbd=[B2' D2']';

%Descomposición SVD minima
[U1,S1,V1] = svd(thetbd,'econ');
U1r=U1(:,1);S1r=S1(1,1);
Bd=U1r*S1r;nr=length(A2);
B3=Bd(1:nr,:);D3=Bd(nr+1,:);
alfa=V1(:,1)% dimension rpxp,

% Funcion transferencia modelo bloque lineal
[num1,den1]=ss2tf(A2,B3,C2,D3);
g1v=(datosv.u).^1;
g2v=(datosv.u).^5;
g3v=(datosv.u).^7;

vv=alfa(1,1)*g1v+alfa(2,1)*g2v+alfa(3,1)*g3v;%+alfa(4,1)*g4v;
yhat3=dlsim(A2,B3,C2,D3,vv);
%% gráfica característica bloque no lineal
Ur=[-2:0.1:2]; % entrada solución base
v=alfa(1,1)*(Ur).^1+alfa(2,1)*(Ur).^5+alfa(3,1)*(Ur).^7;%salida no lineal
figure
plot(Ur,v,'LineWidth',2)
title('característica no lineal modelo Hammerstein','FontSize',14)
ylabel(' v-vv0','FontSize',14)
xlabel('u-u0 (mL/s)','FontSize',14)

% comparacion modelo Hammerstein

Fit4=goodnessOfFit(yhat3,datosv.y,'NRMSE')*100;
mserror2=norm(datosv.y-yhat3)^2/length(yhat3);
figure
tv=length(datosv.y);
time=(nmin_v)*Ts:Ts:(nmin_v+tv-1)*Ts;
plot(time,[datosv.y yhat3],'LineWidth',2)
title('Respuesta con modelo Hammerstein método SVD','FontSize',14)
ylabel('pH-7','FontSize',14)
xlabel('tiempo(s)','FontSize',14)
text(5000,-3,['Fit(%)':
',num2str(Fit4)],'EdgeColor','black','LineStyle','-')
legend('salida real (validacion)','salida modelo')

%=====
%calcula inversa
% =====
xx=linspace(min(v),max(v),200);

pp2=interp1(v,Ur,xx,'pchip');
figure, plot(v,Ur,'bd',xx,pp2,'r','LineWidth',2)
title('Inversa característica no lineal modelo
Hammerstein','FontSize',12.5)
xlabel(' v-vv0','FontSize',12.5)
ylabel('u-u0 (ml/s)','FontSize',12.5)

```

Anexo B3

Programa de identificación Wiener con bases ortonormales

```

clc
load('E:\MAESTRIA IME\datos\u_pH22setd');
load('E:\MAESTRIA IME\datos\y_pH22setd');

Ts=input('tiempo de muestreo de los datos ');
% genera datos para identificacion y validacion de los modelos
datos1=iddata(pH,u,Ts);
disp('longitud de los datos ');
N=length(pH)
datos=datos1(1:N-2);
plot(datos)
%elimina el offset de los datos
datosd=detrend(datos,1);

%selecciona datos estimación y validación
%=====
nmin_e=input('ingrese el minimo valor del rango de estimacion: ');
nmax_e=input('ingrese el valor maximo del rango de estimacion ');
datose=datosd(nmin_e:nmax_e); %datos para
identificacion
nmin_v=input('ingrese el valor minimo del rango de validacion ');
datosv=datosd(nmin_v:length(datosd.u)); % datos para validacion

%
% Modelo Lineal con N4SID
%
%Ajusta el mejor modelo lineal estimación
modelo=n4sid(datose,2,'focus','prediction','n4weight','MOESP');
[A,B,C,D]=ssdata(modelo);
[num,den]=ss2tf(A,B,C,D);

model0=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy0, fit0]=compare(datose,model0);
compare(datose,model0)

%Grafica de datos del modelo lineal para validacion
yhat=dlsim(A,B,C,D,datosv.u);
modell=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy1, fit1]=compare(datosv,model1);
compare(datosv,model1)
mserror1=norm(datosv.y-yhat)^2/length(yhat);

%=====
%Cálculo base ortonormal modelo Wiener
%Bloque no lineal modelo de 5to. orden
%=====
u=datose.u;
y=datose.y;
g1=y;
g2=y.^2;
g3=y.^5;

poles=[0.98 0.81]; % polos bases lineales

```

```

p1=length(poles); % numero de polos

%=====
% calcula base ortonormal
%=====

num0=sqrt(1 - poles(1)^2);
den0=[1 -poles(1)];
num1=sqrt(1 - poles(2)^2)*[-poles(1) 1];
den1=conv(den0,[1 -poles(2)]);

u0=dlsim(num0,den0,u);
u1=dlsim(num1,den1,u);

Phi=[-g2,-g3,u0,u1];
%=====
% Calcula los coeficientes
%=====
theta=pinv(Phi)*y;
a=[theta(1,1) theta(2,1)];
b=[theta(3,1) theta(4,1)];
uv=datosv.u;
v0=dlsim(num0,den0,uv);
v1=dlsim(num1,den1,uv);

%=====
% Representacion bloque no lineal
%=====
yy=[-3:0.02:3]';
vvv=yy+a(1)*yy.^2+a(2)*yy.^5;

figure
plot(vvv,yy,'LineWidth',2)
title('característica bloque no lineal','FontSize',12.5);
xlabel('v-v0','FontSize',12.5);
ylabel('pH-7','FontSize',12.5);

%=====
% salida del bloque no lineal por interpolacion
%=====

yhat2=interp1(vvv,yy,vhat,'pchip');

yv=datosv.y;
tv=length(yv);
time=nmin_v*Ts:Ts:nmin_v*Ts+Ts*tv-1;
figure
plot(time,[yv yhat2],'LineWidth',2)
FIT=goodnessOfFit(yhat2,yv,'NRMSE')*100;
mserror2=norm(datosv.y-yhat2)^2/length(yhat2);
title('Respuesta Modelo Wiener con base ortonormal','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
xlabel('tiempo(seg)','FontSize',12.5)
text(5000,-1,['Fit(%) : ',num2str(FIT)],'EdgeColor','blue','LineStyle','-')
legend('salida real (validacion)','salida modelo')

%=====

```

```

% Representacion bloque lineal en espacio de estado
%=====
A1=diag(poles);
B1=[1;zeros(p1-1,1)];
Ctilde=eye(p1,p1);
pp=-poles;
unos=ones(size(poles));
lambda=sqrt(diag(ones-abs(poles).^2));

%=====
% construccion de las matrices de orden reducido
%=====

for m=2:p1
    for n=1:m-1
        A1(m,n)=1-poles(m)*poles(m-1);
        Ctilde(m,n)=prod(pp(n:m-1));
    end;
end;

%=====
% Ajuste de la matriz A1
%=====

for m=3:p1
    for n=1:m-1
        A1(m,n)=A1(m,n)*prod(pp(n:m-2));
    end;
end;

%=====
% cálculo de la matriz C1
%=====
C1=b*lambda*Ctilde;
D1=0;

vhat3=dlsim(A1,B1,C1,D1,datosv.u);
%yhat3=vhat3+a(1)*vhat3.^2+a(2)*vhat3.^3;
yhat3=interp1(vvv,yy,vhat3,'pchip');
mserror2=norm(datosv.y-yhat3)^2/length(yhat3);
Fit3=goodnessOfFit(yhat3,yv,'NRMSE')*100;
figure
plot(time,[datosv.y yhat3],'LineWidth',2)
title('Salida Datos validacion(azul) y modelo en ss base
ortonormal','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
xlabel('tiempo(seg)','FontSize',12.5)
text(5000,-1,['Fit%]: ',num2str(Fit3)],'EdgeColor','black','LineStyle','-')
legend('salida real (validacion)','salida modelo')

```

Anexo B4

Programa de identificación Hammerstein con bases ortonormales

```

clc
load('E:\MAESTRIA IME\datos\u_pH22setd');
load('E:\MAESTRIA IME\datos\y_pH22setd');

Ts=input('tiempo de muestreo de los datos ');
% genera datos para identificacion y validacion de los modelos
datos1=iddata(pH,u,Ts);
disp('longitud de los datos ');
N=length(pH)
datos=datos1(1:N-2);
plot(datos)
%elimina el offset de los datos
datosd=detrend(datos,1);

%=====
%selecciona datos estimación y validación
%=====
nmin_e=input('ingrese el minimo valor del rango de estimacion: ');
nmax_e=input('ingrese el valor maximo del rango de estimacion ');
datose=datosd(nmin_e:nmax_e); %datos para
identificacion
nmin_v=input('ingrese el valor minimo del rango de validacion ');
datosv=datosd(nmin_v:length(datosd.u)); % datos para validacion

%=====
% Modelo Lineal con N4SID
%
%Ajusta el mejor modelo lineal estimación
%=====

modelo=n4sid(datose,2,'focus','prediction','n4weight','MOESP');
[A,B,C,D]=ssdata(modelo);
[num,den]=ss2tf(A,B,C,D);

model0=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy0, fit0]=compare(datose,model0);
compare(datose,model0)

%Grafica de datos del modelo lineal para validacion
yhat=dlsim(A,B,C,D,datosv.u);
modell=idss(A,B,C,D,zeros(size(A,1),1),zeros(size(A,1),1),Ts);
figure
[yy1, fit1]=compare(datosv,model1);
compare(datosv,model1)
mserror1=norm(datosv.y-yhat)^2/length(yhat);

%calcula base ortonormal modelo Hammerstein
u=datose.u;
g1=u;
g2=u.^2;
g3=u.^3;
g=[g1 g2 g3];
polos=[0.98 0.81];
p1=length(polos);

```

```

num0=sqrt(1 - polos(1)^2);
den0=[1 -polos(1)];
num1=sqrt(1 - polos(2)^2)*[-polos(1) 1];
den1=conv(den0,[1 -polos(2)]);

u01=dlsim(num0,den0,g1);
u02=dlsim(num0,den0,g2);
u03=dlsim(num0,den0,g3);
u11=dlsim(num1,den1,g1);
u12=dlsim(num1,den1,g2);
u13=dlsim(num1,den1,g3);

Phi=[u01 u02 u03 u11 u12 u13];
y=datose.y;
theta=pinv(Phi)*y;

%function M = col_orden( theta,d,p )
%theta: vector columna dim [d,p,1]
%Desapila p columnas igual al orden de la base ortonormal
% d: número de filas, equivalente al orden del polinomio
%p: orden de la base ortonormal
theta_ab=col_orden(theta,3,2);

[U,S,V]=svd(theta_ab,0);
a=U(:,1);
b=V(:,1) '*S(1,1);

uv=datosv.u;
yv=datosv.y;
uu=[-3:0.01:3]';
vvv=a(1)*uu+a(2)*uu.^2+a(3)*uu.^3;
figure
plot(uu,vvv,'LineWidth',2)
title('característica bloque no lineal','FontSize',12.5);
xlabel('u-u0','FontSize',12.5);
ylabel('v-v0','FontSize',12.5);
vhat=a(1)*uv+a(2)*uv.^2+a(3)*uv.^3;
u1v=dlsim(num0,den0,vhat);
u2v=dlsim(num1,den1,vhat);

yhat1=b(1)*u1v+b(2)*u2v;
Fit=goodnessOfFit(yhat1,yv,'NRMSE')*100;
%
%Compara bloques no lineales

figure
tv=length(datosv.y);
time=nmin_v*Ts:Ts:nmin_v*Ts+Ts*tv-1;
plot(time,[yv yhat1],'LineWidth',2)
title('Respuesta con modelo Hammerstein con base
ortonormal','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
xlabel('tiempo(seg)','FontSize',12.5)
text(5000,-1,['Fit(%) : ',num2str(Fit)],'EdgeColor','black','LineStyle','-
')
legend('salida real (validacion)','salida modelo')

```

```

%=====
% Representacion bloque lineal en espacio de estado
%=====
A1=diag(polos);
B1=[1;zeros(p1-1,1)];
Ctilde=eye(p1,p1);
pp=-polos;
unos=ones(size(polos));
lambda=sqrt(diag(ones-abs(polos).^2));

%=====
% construccion de las matrices de orden reducido
%=====

for m=2:p1
    for n=1:m-1
        A1(m,n)=1-polos(m)*polos(m-1);
        Ctilde(m,n)=prod(pp(n:m-1));
    end;
end;

%=====
% Ajuste de la matriz A1
%=====

for m=3:p1
    for n=1:m-1
        A1(m,n)=A1(m,n)*prod(pp(n:m-2));
    end;
end;

%=====
% calculo de la matriz C1
%=====
C1=b*lambda*Ctilde;
D1=0;

yhat3=dlsim(A1,B1,C1,D1,vhat);
Fit3=goodnessOfFit(yhat3,datosv.y,'NRMSE')*100;
mserror2=norm(datosv.y-yhat3)^2/length(yhat3);
figure
plot(time,[datosv.y yhat3],'LineWidth',2)
title('Salida Datos validacion(azul) y modelo en ss base
ortonormal','FontSize',12.5)
ylabel('pH-7','FontSize',12.5)
xlabel('tiempo(seg)','FontSize',12.5)
text('5000,-1,['Fit%): ',num2str(Fit3)],'EdgeColor','black','LineStyle','-
legend('salida real (validacion)','salida modelo')

```

Anexo B5

Programa desapila columnas de base ortonormal

```
function M = col_orden( theta,d,p )
%theta: vector columna dim [d x p,1]
%Desapila p columnas igual al orden de la base ortonormal
%  d: número de filas, equivalente al orden del polinomio
%p: orden de la base ortonormal
M=theta(1:d);
s=0;
t=d;
for i=2:p
    s=s+d;
    t=t+d;
    m=theta(s+1:t);
    M=[M m];
end

end
```

Anexo B6

Programa identificación modelos no lineales Wiener y Hammerstein usando instrucción nlhw de MATLAB

```

clc
disp('carga datos');

load('G:\MAESTRIA IME\datos\u_pH22setd');
load('G:\MAESTRIA IME\datos\y_pH22setd');
% =====
% Tiempo de muestreo 1s
% =====
Ts=input('tiempo de muestreo de los datos ');
% =====
% genera datos para identificacion y validacion de los modelos
% =====
datos1=iddata(pH,u,Ts);
disp('longitud de los datos ');
N=length(pH)
datos=datos1(1:N-2);
plot(datos)
%elimina el offset de los datos
datosd=detrend(datos,1);

%selecciona datos estimación y validación
%=====
nmin_e=input('ingrese el minimo valor del rango de estimacion: ');
nmax_e=input('ingrese el valor maximo del rango de estimacion ');
datose=datosd(nmin_e:nmax_e); %datos para
identificacion
nmin_v=input('ingrese el valor minimo del rango de validacion ');
datosv=datosd(nmin_v:length(datosd.u)); % datos para validacion
%=====
% Modelo Hammerstein con nlhw
%=====
% aproximacion piecewise linear
m1=nlhw(datose,[2 3 1],pwlinear('num',10),[]);
figure
compare(datosv,m1)
figure
compare(datose,m1)
pause

figure
% aproximacion sigmoide
m2=nlhw(datose,[3 2 0],'sigmoidnet',[]);
compare(datosv,m2)
figure
compare(datose,m2)
pause

%=====
% Modelo Wiener con nlhw
%=====

% aproximacion piecewise linear
m3=nlhw(datose,[2 3 1],[],pwlinear('num',10));
figure

```

```
compare(datosv,m3)
figure
compare(datose,m3)
pause
```

```
figure
% aproximacion sigmoide
m4=nlhw(datose,[2 3 1],[],'sigmoidnet');
compare(datosv,m4)
figure
compare(datose,m4)
pause
```

```
%=====
% Compara todos los modelos
%=====
```

```
figure
compare(datosv,m1,m2,m3,m4)
figure
compare(datose,m1,m2,m3,m4)
```

Anexo B7

Programa control Wiener PID predictivo

```

clear all
load('E:\MAESTRIA IME\datos\identset25dNL2.mat');
load('E:\MAESTRIA IME\datos\identset25d.mat');
% Modelo discreto
Ts=1; %tiempo de muestreo
Am=A2; Bm=B2; Cm=C2M;
%
Na=size(Am);
Om=zeros(1,Na(1)); % ceros para completar la matriz aumentada

%Espacio de estados
A=[Am Om'; Cm*Am 1];
B=[Bm; Cm*Bm];
C=[Om 1];

x0=[0 0]';
[e,f]=size(Am);

Nu=3;
N2=10;
lam=100;
delta=1;
F=C*A;
Gaux(1)=C*B;
for i=2:N2
F=[F; C*A^i];
Gaux(i)=C*A^(i-1)*B;
end
Gaux1=Gaux';
Gm=zeros(N2,Nu);
for i=1:Nu
Gm(i:end,i)=Gaux1(1:end-i+1);
end
di=lam*ones(Nu,1);
R=diag(di);

deltai=delta*ones(N2,1);
DELTA=diag(deltai);
K=inv((Gm'*DELTA*Gm+R))*Gm'*DELTA;

CalPID=[1 1 1;-1 0 -2;0 0 1];
%
%
r=ones(N2,1);
L=[1;zeros(Nu-1,1)]';
Ky=L*K*r
Kmpc=L*K*F
Kx=Kmpc(1,1:e);

%=====

Tx1=Ky*Cm;
P=(Kx+Tx1)*Am-Kx)*Am;
Qu=(Kx+Tx1)*Am-Kx)*Bm;

```

```

Ru=(Kx+Tx1)*Bm;
%=====
% calculo de matrices para hallar parámetros PID
%=====

Ap=[(Cm*Am^2)' (Cm*Am)' Cm';Cm*Am*Bm Cm*Bm 0;Cm*Bm 0 0];
Bp=[P'; Qu; Ru];

%Calculo q0, q1 y q2
q= Ap\Bp;
Kpid=inv(CalPID)*q
q0=q(1,1);q1=q(2,1);q2=q(3,1);
Kp=Kpid(1,1);Ki=Kpid(2,1);Kd=Kpid(3,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%
% Característica polinómica inversa de bloque no lineal
% ALFA=a0*y +a1*y^2+a2* y^3
alfa=flipplr(ALFA);

l=[alfa 0];

```

Anexo B8

Programa control Hammerstein PID predictivo

```

clear all

load('E:\MAESTRIA IME\datos\identset23NL2.mat');
load('E:\MAESTRIA IME\datos\identset23.mat');
Am=A2;Bm=B3;Cm=C2;
Ts=1; %tiempo de muestreo

Na=size(Am);
Om=zeros(1,Na(1));% ceros para completar la matriz aumentada

%Espacio de estados
A=[Am Om';Cm*Am 1];
B=[Bm;Cm*Bm];
C=[Om 1];

x0=[0 0]';
[e,f]=size(Am);

Nu=3;
N2=10;
lam=0.5;
delta=1;
F=C*A;
Gaux(1)=C*B;
for i=2:N2
F=[F;C*A^i];
Gaux(i)=C*A^(i-1)*B;
end
Gaux1=Gaux';
Gm=zeros(N2,Nu);
for i=1:Nu
Gm(i:end,i)=Gaux1(1:end-i+1);
end
di=lam*ones(Nu,1);
R=diag(di);

deltai=delta*ones(N2,1);
DELTA=diag(deltai);
K=inv((Gm'*DELTA*Gm+R))*Gm'*DELTA;

CalPID=[1 1 1;-1 0 -2;0 0 1];
%

r=ones(N2,1);
L=[1;zeros(Nu-1,1)]';
Ky=L*K*r
Kmpc=L*K*F
Kx=Kmpc(1,1:e);

%=====

Tx1=Ky*Cm;
P=(Kx+Tx1)*Am-Kx)*Am;
Qu=(Kx+Tx1)*Am-Kx)*Bm;
Ru=(Kx+Tx1)*Bm;
%=====

```

```

% calculo de matrices para hallar parametros PID
%=====

Ap=[ (Cm*Am^2)' (Cm*Am)' Cm'; Cm*Am*Bm Cm*Bm 0; Cm*Bm 0 0];
Bp=[P'; Qu; Ru];

%Calculo q0, q1 y q2
q= Ap\Bp;
Kpid=inv(CalPID)*q
q0=q(1,1);q1=q(2,1);q2=q(3,1);
Kp=Kpid(1,1);Ki=Kpid(2,1);Kd=Kpid(3,1);

```

Anexo B9

Programa control MPC con modelo lineal y estimador

```

clear all

% Modelo discreto lineal
Ts=1; %tiempo de muestreo
Am=[0.9825 1.0663;-0.0019 0.7441];Bm=[-0.151 -0.0649]';
Cm=[-0.5867 0.7327];
%
Na=size(Am);
Om=zeros(1,Na(1));% ceros para completar la matriz aumentada

%Espacio de estados
A=[Am Om';Cm*Am 1];
B=[Bm;Cm*Bm];
C=[Om 1];

x0=[0 0]';
[e,f]=size(Am);

Nu=3;
N2=8;
lam=100;
delta=1;
F=C*A;
Gaux(1)=C*B;
for i=2:N2
F=[F;C*A^i];
Gaux(i)=C*A^(i-1)*B;
end
Gaux1=Gaux';
Gm=zeros(N2,Nu);
for i=1:Nu
Gm(i:end,i)=Gaux1(1:end-i+1);
end
di=lam*ones(Nu,1);
R=diag(di);

deltai=delta*ones(N2,1);
DELTA=diag(deltai);
K=inv((Gm'*DELTA*Gm+R))*Gm'*DELTA;

r=ones(N2,1);
L=[1;zeros(Nu-1,1)]';
Ky=L*K*r
Kmpc=L*K*F
Kx=Kmpc(1,1:e);

```