

Centro Nacional de Investigación y Desarrollo Tecnológico

Subdirección Académica

Departamento de Ingeniería Electrónica

TESIS DE MAESTRÍA EN CIENCIAS

**Identificación de Modelos Wiener-Hammerstein Utilizando Cálculo
Fraccionario**

presentada por
Ing. Patricia Vázquez Guerrero

como requisito para la obtención del grado de
Maestra en Ciencias en Ingeniería Electrónica

Director de tesis
Dr. Víctor Manuel Alvarado Martínez

Codirector de tesis
Dr. José Francisco Gómez Aguilar

Cuernavaca, Morelos, México. Julio de 2016.

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO
Centro Nacional de Investigación y Desarrollo Tecnológico

Cuernavaca, Mor., 2 de junio de 2016.

OFICIO No. DIE/120/2016
Asunto: Aceptación de documento de tesis

DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial de la C. **Patricia Vázquez Guerrero** con número de control **M14CE081** de la Maestría en Ciencias en Ingeniería Electrónica, le informamos que hemos revisado el trabajo de tesis profesional titulado **“IDENTIFICACIÓN DE MODELOS WIENER-HAMMERSTEIN UTILIZANDO CÁLCULO FRACCIONARIO”** y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTOR DE TESIS

Dr. Víctor Manuel Alvarado Martínez
Doctor en Ciencias en Ingeniería Electrónica
Cédula profesional 8031070

CODIRECTOR DE TESIS

Dr. José Francisco Gómez Aguilar
Doctor en Física
Cédula profesional 9124781

REVISOR 1

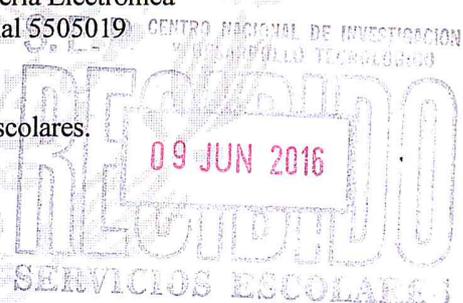
Dr. Manuel Adam Medina
Doctor en Ciencias en Ingeniería Electrónica
Cédula profesional 599203

REVISOR 2

Dr. Carlos Manuel Astorga Zaragoza
Doctor en Ingeniería Electrónica
Cédula profesional 5505019

C.p. Lic. Guadalupe Garrido Rivera.- Jefa del Departamento de Servicios Escolares.
Estudiante
Expediente

CMAZ/lrr



cenidet[®]
Centro Nacional de Investigación
y Desarrollo Tecnológico

Interior Internado Palmira S/N, Col. Palmira. C.P. 62490 Cuernavaca, Mor.
Tels. (01)777 362-77-70 Ext. 4106, e-mail: direccion@cenidet.edu.mx
www.cenidet.edu.mx





Cuernavaca, Mor., 9 de junio de 2016
OFICIO No. SAC/201/2016

Asunto: Autorización de impresión de tesis

ING. PATRICIA VÁZQUEZ GUERRERO
CANDIDATA AL GRADO DE MAESTRA EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA
PRESENTE

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **“Identificación de Modelos Wiener-Hammerstein Utilizando Cálculo Fraccionario”**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

“CONOCIMIENTO Y TECNOLOGIA AL SERVICIO DE MEXICO”

DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO



SEP TecNM
CENTRO NACIONAL
DE INVESTIGACIÓN
Y DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA

C.p. Lic. Guadalupe Garrido Rivera.- Jefa del Departamento de Servicios Escolares.
Expediente

GVGR/mcr



Dedicatoria

*A mis padres **Setelbait Guerrero y Miguel Vázquez**,
por su apoyo incondicional y ser mi ejemplo a seguir,
porque sin ellos no hubiera podido llegar a este momento,
¡LOS AMO!*

*A mi hermana **Martha Elvira Vázquez**,
por ser una inspiración para mi
una mujer admirable y gran apoyo en mi vida,
no tengo como agradecerte.*

*A mi sobrino **Fernandito**, por recordarme que la vida
es más fácil cuando sonreímos,
gracias por venir a iluminar mi vida*

*Y especialmente a ti **Dios**,
por darme la fuerza para seguir adelante cuando mas lo necesitaba
rodearme de personas valiosas, bendecirme y llenarme de amor.*

Agradecimientos

A Dios por cada una de sus bendiciones, por la fuerza de voluntad para superar las adversidades, por brindarme la salud y la felicidad que ahora tengo.

A mis padres y hermana, por ser mi apoyo incondicional, consentirme y cuidarme en todo momento, sin ustedes simplemente no sería posible. Gracias por creer en mi.

A mis asesores el Dr. Víctor Manuel Alvarado Martínez y al Dr. José Francisco Gómez Aguilar, por guiarme durante el desarrollo de este trabajo de investigación mediante sus valiosos consejos y observaciones.

A los miembros del comité revisor: Dr. Carlos Manuel Astorga Zaragoza y al Dr. Manuel Adam Medina por sus acertadas correcciones y comentarios que permitieron enriquecer este trabajo.

A mis profesores: Dr. Alejandro Rodríguez Palacios, Dr. Enrique Quintero-Mármol Márquez, Dr. Juan Reyes Reyes, Dr. Manuel Adam Medina, Dr. Carlos Manuel Astorga Zaragoza, Dr. Víctor Manuel Alvarado Martínez, Dra. Guadalupe López y Dr. Jaime Arau por todos los conocimientos brindados y ejemplo de profesionalismo.

A mis compañeros de la generación "Kharitonov" y a mis amigas: Lily, Elda, Lilo, Ivonne y Paulina gracias por esas porras cuando no encontraba inspiración, definitivamente fueron parte importante en la realización de esta tesis.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) y al Tecnológico Nacional de México (TecNM) por el apoyo económico brindado para poder realizar y culminar mis estudios de maestría.

Finalmente, agradezco de manera profunda al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) por brindarme los medios necesarios para mi formación académica, así como, por cada una de las amables atenciones que recibí durante mi estancia.

Resumen

En la presente tesis se aborda una nueva metodología para la identificación de modelos no lineales utilizando cálculo fraccionario, se lleva a cabo el desarrollo e implementación de un algoritmo para modelos Wiener-Hammerstein.

La identificación de modelos se realizó a partir de datos de entrada-salida de tres sistemas no lineales (Dos tanques acoplados, brazo robot y circuito electrónico), la estructura no lineal seleccionada en esta tesis es la Wiener-Hammerstein y por último se empleó el algoritmo de optimización de parámetros utilizado fue el conocido como Levenberg-Marquardt.

Se realizó la comparación de los modelos Wiener-Hammerstein fraccionarios obtenidos contra lo reportado en trabajos previos en el CENIDET ([Flores, 2011] y [Romero, 2013]) y en los siguientes artículos de investigación ([Vanbeylen, 2014], [Azmi et al., 2015] y [Romero et al., 2015]), la comparación se llevó a cabo efectuando diferentes pruebas de validación. Adicionalmente se comparó con los modelos Hammerstein-Wiener que se obtienen con las herramientas de identificación de Matlab.

También se programó una interfaz gráfica para llevar a cabo la identificación de modelos fraccionarios lineales, Hammerstein y Wiener-Hammerstein, dentro de esta tesis se incluye un manual de usuario y manual técnico para usuarios futuros.

Abstract

In this thesis a new methodology for identification of nonlinear models using fractional calculation is presented, the development and implementation of an algorithm for Wiener-Hammerstein models is implemented.

System identification was performed from input-output data of nonlinear systems (coupled tanks, robotic arm and electrical circuit) and then choosing the structure to be used, in this case, Wiener-Hammerstein. Lastly, the parameter optimization algorithm employed was Levenberg-Marquardt.

It was performed as a comparison of the models obtained with the reported in previous studies in CENIDET ([Flores, 2011] and [Romero, 2013]) and in research papers ([Vanbeylen, 2014], [Azmi et al., 2015] y [Romero et al., 2015]). This comparison was made performing various validation tests. Additionally, the models were compared with Hammerstein-Wiener model obtained via Matlab tools.

Lastly, a Graphic User Interface (GUI) for linear, Hammerstein and Wiener-Hammerstein fractional model algorithms, was programmed. A users manual and technical manual for future users is included.

Índice general

1. Introducción	1
1.1. Antecedentes	1
1.2. Planteamiento del problema	3
1.3. Objetivos	3
1.3.1. Objetivo General	3
1.3.2. Objetivos específicos	4
1.4. Justificación	4
1.5. Revisión del estado del arte	5
1.5.1. Identificación de Sistemas	7
1.5.2. Validación del modelo	8
1.6. Alcance de la tesis	15
1.7. Propuesta de solución	16
1.8. Organización del documento	17
2. Introducción al Cálculo fraccionario	18
2.1. Introducción	18
2.2. Integral fraccionaria	19
2.3. Derivada fraccionaria	21
2.3.1. Función Gamma	22
2.3.2. Función de Mittag-Leffler	22
2.4. Transformada de Laplace	23
3. Estructura Wiener-Hammerstein	25
3.1. Introducción	25
3.2. Antecedentes	26
3.2.1. Identificación de modelos lineales utilizando cálculo frac- cionario	26
3.2.1.1. Ejemplo: Circuito RC	30

3.2.2.	Identificación de modelos Hammerstein utilizando cálculo fraccionario	34
3.2.2.1.	Ejemplo: Secadora de cabello	37
3.3.	Representación de la estructura Wiener-Hammerstein	40
3.4.	Estructura Wiener-Hammerstein fraccionaria propuesta	40
3.4.1.	Diagrama a bloques	42
4.	Implementación del Algoritmo	44
4.1.	Sistemas dinámicos identificados	44
4.1.1.	Dos Tanques acoplados	44
4.1.2.	Brazo robot	45
4.1.3.	Circuito electrónico	45
4.2.	Comparación entre metodologías (Ventajas y desventajas)	48
4.2.1.	Dos Tanques acoplados	48
4.2.2.	Brazo robot	53
4.2.3.	Circuito electrónico	57
5.	Conclusiones	61
5.1.	Conclusiones generales	61
5.2.	Trabajos futuros	63
	Bibliografía	63
A.	Funciones de Matlab utilizadas y Manual de usuario de las funciones	68
A.1.	Funciones de Matlab utilizadas	68
A.2.	Funciones de CRONE utilizadas	68
B.	Interfaz gráfica	70
C.	Código para la Identificación de Modelos Wiener-Hammerstein de orden Fraccionario	78

Índice de tablas

3.1. Parámetros de validación de modelos en el sistema RC.	32
3.2. Parámetros de validación de modelos en el sistema de Secadora de Cabello.	39
4.1. Parámetros de validación de modelos en el sistema de dos tanques acoplados.	53
4.2. Parámetros de validación de modelos en el sistema de brazo robot.	56
4.3. Parámetros de validación de modelos en el sistema del circuito electrónico.	60

Índice de figuras

1.1.	Modelo de identificación no lineal Wiener-Hammerstein.	2
1.2.	Sistema dinámico, y es la salida, u es la entrada, w es la perturbación medida y v es la perturbación no medida	7
1.3.	El ciclo de Identificación de Sistemas	11
1.4.	Diagrama a bloques de la identificación de parámetros	13
3.1.	Modelos Hammerstein, Wiener, Wiener-Hammerstein y Hammerstein-Wiener	26
3.2.	Modelo Lineal.	27
3.3.	Circuito RC.	30
3.4.	Comparación de modelos y salida del Sistema del circuito RC. . .	33
3.5.	Acercamiento de la comparación de modelos y salida del sistema del circuito RC.	33
3.6.	Modelo Hammerstein.	34
3.7.	Modelo Hammerstein	39
3.8.	Modelo Wiener-Hammerstein.	40
3.9.	Diagrama a bloques de la obtención de modelos Wiener-Hammerstein fraccionarios.	43
4.1.	Sistema de Tanques.	45
4.2.	Entrada y salida del sistema de dos tanques acoplados	46
4.3.	Entrada y salida del sistema de Brazo robot	47
4.4.	Modelo Wiener-Hammerstein del Circuito electrónico no lineal. . .	47
4.5.	Circuito del Diodo, parte no lineal.	48
4.6.	Entrada y salida del sistema Circuito electrónico	49
4.7.	Modelo fraccionario y entero del sistema de dos tanques acoplados.	51
4.8.	Acercamiento del modelo fraccionario y entero del sistema de dos tanques acoplados	52
4.9.	Modelo fraccionario y entero del sistema de brazo robot	55

4.10. Zoom del modelo fraccionario y entero del sistema de brazo robot	55
4.11. Modelo fraccionario y entero del sistema circuito electrónico. . . .	58
4.12. Acercamiento del modelo fraccionario y entero del sistema Circuito electrónico.	59
B.1. Interfaz principal	70
B.2. Pantalla del modelo lineal	71
B.3. Diagrama de bloques del modelo lineal fraccionario	72
B.4. Pantalla para cargar datos	72
B.5. Pantalla del modelo lineal resultante	73
B.6. Pantalla del modelo Hammerstein	73
B.7. Diagrama de bloques del modelo Hammerstein fraccionario	74
B.8. Pantalla del modelo Hammerstein resultante	75
B.9. Pantalla del modelo Wiener-Hammerstein	75
B.10. Diagrama de bloques del modelo Wiener-Hammerstein fraccionario	76
B.11. Pantalla del modelo Wiener-Hammerstein resultante	77

Capítulo 1

Introducción

1.1. Antecedentes

Hoy en día, debido al desarrollo tecnológico, nos encontramos con sistemas cada vez más complejos en los que el control es necesario. Existen diferentes métodos para la realización del control de un sistema, sin embargo el más utilizado se basa en el modelo matemático de éste. Dicho modelo matemático es importante en la realización del análisis, simulación, optimización, diagnóstico de fallas y sobretodo el diseño de controladores.

El modelo matemático puede ser obtenido mediante el análisis de las leyes físicas que rigen al sistema. Aunque muchas veces resulta muy complicado, de tal manera que otra manera de obtener un modelo matemático de un sistema es utilizando la Identificación de Sistemas, cuyo objetivo es el construir y estimar modelos matemáticos con los datos de entrada y salida medidos directamente del sistema o proceso.

Por otra parte, en las últimas décadas el concepto de cálculo fraccionario ha evolucionado notablemente, no sólo en el ámbito matemático sino también en aplicaciones científicas y en el campo de la ingeniería, entre las áreas que se han visto más influenciadas destacan la teoría de las probabilidades, los procesos estocásticos, las ecuaciones integro-diferenciales, la teoría de las transformadas, las funciones especiales y el análisis numérico, a pesar de que es importante y necesaria la obtención de una teoría matemática rigurosa de los operadores fraccionarios, en el ámbito aplicado o en el área de la ingeniería esta técnica ha tomado auge principalmente durante los últimos 30 años. Algunas investigaciones las podemos encontrar en sistemas biofísicos [Gómez-Aguilar et al., 2013, Toledo-Hernandez et al., 2014], sistemas térmicos y de transferencia de calor [Gómez-Aguilar et al., 2015], sistemas mecánicos [Gómez-Aguilar et al., 2016b,

Atanackovic et al., 2014], eléctricos [Gómez-Aguilar et al., 2014, Ullah et al., 2014], por mencionar algunos ejemplos.

El cálculo fraccionario es la rama del análisis matemático que estudia a los operadores de integración y derivación de orden no entero (D^λ , donde λ es un número real) sobre dominios de funciones reales o complejas. Si bien, actualmente existen muchas definiciones de cálculo fraccionario el principal objetivo de cada una de ellas es generalizar el concepto de derivada de orden entero, de tal manera que para $\lambda=1$ se obtiene el operador derivativo conocido comúnmente. En este contexto, no solo se busca trabajar con órdenes de integración fraccionaria, sino también con reales y complejos y así poder obtener integrales y derivadas de orden arbitrario.

El uso de modelos con diferenciación fraccionaria en identificación de sistemas tomó auge a principios del presente siglo. Modelos fraccionarios son ahora lo suficientemente maduros y son utilizados en la representación de sistemas con frecuencia.

En esta tesis se trabajará con la identificación no lineal, específicamente con el modelo a bloques Wiener-Hammerstein (W-H); esta estructura consiste en una conexión en cascada de tres subsistemas o bloques, específicamente dos bloques lineales dinámicos con un bloque no lineal estático en el centro (Ver en la Figura 1.1) en los cuales se hará uso de las herramientas que nos proporciona el cálculo fraccionario, en particular derivadas fraccionarias y funciones de transferencia en frecuencia de orden no entero.



Figura 1.1: Modelo de identificación no lineal Wiener-Hammerstein.

1.2. Planteamiento del problema

Como se mencionó anteriormente, para obtener un modelo matemático de un sistema es necesario conocer a fondo el sistema a modelar y recurrir a las leyes físicas y químicas que describan el comportamiento dinámico del proceso, este análisis resulta complicado cuando se trabaja con sistemas no lineales complejos, es precisamente la identificación una opción para la estimación y construcción de modelos, ya que en muchas ocasiones no existe un modelo matemático del sistema que represente fielmente su comportamiento. El modelo matemático resultante es una buena aproximación del sistema real, sin embargo, existe la posibilidad de que se comporte de una manera inesperada sobre todo si se está intentado representar el modelo de un sistema no lineal.

Por tanto, el obtener un modelo matemático que represente la dinámica del sistema resultado de la identificación de sistemas es un procedimiento práctico en donde sólo se necesita la señal de entrada que excita al sistema y la señal de salida, es importante mencionar que las señales de entrada y salida se obtienen mediante métodos experimentales.

Para obtener un modelo no lineal que refleje la información del sistema se requiere utilizar algoritmos que utilicen estructuras más complejas; tal es el caso de la estructura Wiener-Hammerstein, en donde es posible incluir la no linealidad del sistema y aprovechar las herramientas de modelado que nos proporciona el cálculo fraccionario.

El problema surge de la necesidad de obtener modelos que se ajusten de una mejor manera al sistema real y de una manera más compacta, esto se realizará mediante la identificación de modelos Wiener-Hammerstein (W-H) utilizando operadores generalizados o de orden arbitrario..

1.3. Objetivos

1.3.1. Objetivo General

Asimilar e implementar métodos generalizados de identificación de sistemas para obtener modelos Wiener-Hammerstein de algunos sistemas utilizando cálculo fraccionario.

1.3.2. Objetivos específicos

1. Seleccionar la estructura a identificar con la que se va a trabajar.
2. Determinar los métodos de cálculo fraccionario susceptibles de ser probados en la identificación de modelos Wiener-Hammerstein.
3. Implementar los métodos elegidos en Matlab y en su caso proponer variantes en el cálculo fruto de los trabajos previos en identificación realizados en el CENIDET.
4. Comparar el desempeño de este trabajo contra trabajos previos en el CENIDET y contra lo reportado en la literatura con respecto a los métodos obtenidos.

1.4. Justificación

Es común que surjan muchas dudas de las aplicaciones del cálculo fraccionario (interpretación física, geométrica y la formulación del mismo) debido a su reciente auge en la comunidad científica, principalmente la utilidad y las ventajas que éste podría proporcionarnos. En las investigaciones de los últimos años se han demostrado ciertas ventajas en el uso del cálculo fraccionario en comparación con el cálculo ordinario; uno de los más importantes a considerar en esta tesis se debe a que el cálculo fraccionario puede representar sistemas dinámicos de orden superior y fenómenos complejos no lineales utilizando un menor número de coeficientes debido a que el orden arbitrario de las derivadas le da un grado de libertad adicional que permite ajustarse a un comportamiento específico.

Otra ventaja es que el cálculo fraccionario es no-local, esto quiere decir que las derivadas fraccionarias dependen del dominio y las condiciones de contorno, es decir que el dominio y las condiciones de contorno de las funciones deben ser elegidas y restringidas con el fin de encontrar las derivadas fraccionarias [Gómez-Aguilar, 2012], lo cual lleva a la necesidad de incluir información sobre la función más allá del punto a trabajar y no solo sobre una pequeña vecindad. Estas ventajas han permitido que el cálculo fraccionario tenga un papel importante en el modelado de sistemas complejos en muchas áreas de la ciencia y la tecnología.

1.5. Revisión del estado del arte

El uso de derivadas e integrales de orden no entero o fraccionario ha sido estudiado durante mucho años, sin embargo era utilizado como un concepto puramente matemático, en el trabajo de [Sánchez, 2011] encontramos una visión de cómo se ha dado la evolución histórica del cálculo fraccionario desde su nacimiento teórico e intuitivo desde finales del siglo XVII, su formalización en el siglo XIX, hasta la actualidad en donde se habla de las diferentes disciplinas en la que se ha logrado una aplicación gracias a los avances en *hardware* y *software* que permiten simular funciones matemáticas diferenciales de orden no entero o fraccionarias.

El uso de modelos con diferenciación fraccionaria en identificación de sistemas se inició a finales de los años noventa y principios del presente siglo. En los artículos de [Malti et al., 2007] y [Malti et al., 2006] se presenta una visión general sobre los métodos de identificación de sistemas lineales que utilizan modelos fraccionarios, y se realiza la identificación en el dominio del tiempo de sistemas fraccionarios utilizando el método de ecuación de error en la salida (OE- output error, por sus siglas en inglés).

En la vida real es común encontrarnos con sistemas dinámicos no lineales, una de las maneras de obtener el modelo matemático es mediante el método de identificación de sistemas orientado a bloques, entre los principales, encontramos a los modelos Hammerstein, Wiener y la combinación de los dos; Hammerstein-Wiener y Wiener-Hammerstein como lo menciona [Ljung, 2007] en su libro de identificación de sistemas no lineales.

En la actualidad ya se ha realizado la identificación de sistemas utilizando cálculo fraccionario, mediante las estructuras Hammerstein, Wiener o la combinación de ambas. En el artículo de [Aoun et al., 2002] se lleva a cabo la identificación de sistemas no lineales en tiempo continuo, esta identificación se caracteriza por tener dinámicas de orden fraccionarias utilizando un modelo Hammerstein que permite que su parte lineal pertenezca a la clase de modelos fraccionarios, se utiliza la definición diferenciación fraccionaria de Riemann-Liouville. Se concluye que los modelos fraccionarios son compactos y por lo tanto se utilizan para modelar la dinámica compleja del sistema con pocos parámetros. Utilizando la estructura Hammerstein, los autores de los artículos [Li et al., 2014] y [Zhao et al., 2014] presentan la identificación de parámetros de modelos tipo Hammerstein de orden fraccionario de la forma ARX (por sus siglas en inglés,

AutoRegressive with eXternal input) y OE, en los artículos mencionados se aplica la definición de derivada fraccionaria de Grünwald-Letnikov.

Otro artículo es el presentado por [Stanislawski et al., 2014], en donde se trabaja con la estructura orientada a bloques Wiener, se emplea el concepto de modelado de dinámicas no lineales por medio de funciones de base ortonormal para separar submodelos lineales y no lineales, se observa un muy buen rendimiento de identificación de un modelo basado en modelos Laguerre-Wiener de orden fraccionario, tanto en términos de los errores de predicción bajos y la reconstrucción precisa de las características reales del sistema.

También se han realizado investigaciones utilizando la combinación de Wiener y Hammerstein, en el artículo de [Vanbeylen, 2014] se emplea la definición de derivada fraccionaria de Grünwald-Letnikov se explica un modelo de parametrización fraccionaria basado en la mejor aproximación lineal (BLA por su siglas en inglés: Best Linear Approximation), se demuestra que no se necesitan más de dos optimizaciones iterativas y que se pueden minimizar modelos de grandes órdenes enteros.

Hacer uso de los algoritmos de identificación de sistemas fraccionarios ha tomado auge en el ámbito aplicado y actualmente se están realizando implementaciones en plantas reales, en el artículo de [Hernandez et al., 2014] se propone un modelo de orden fraccionario de un horno rotatorio de cemento. El modelo consta de dos partes: estática y dinámica, la parte estática depende del balance de energía de la manera de cómo se distribuye el calor alrededor de todo el proceso y la parte dinámica se obtiene a través del método de identificación fraccionaria. La entrada del modelo dinámico es la salida de la parte estática, finalmente se demuestra cómo el proceso responde a cambios en las variables de entrada.

Es importante mencionar que actualmente se cuenta con tres herramientas de programación en Matlab (Toolbox en inglés) que permiten el uso de los conceptos del cálculo fraccionario, estos son Ninteger, CRONE y Fomcon. En las referencias [Cois et al., 2002], [Duarte and Costa, 2004] y [Tepljakov et al., 2011], se explica de manera breve los principales módulos que contiene cada toolbox, en el caso de Fomcon y CRONE se explica un módulo para la identificación de sistemas lineales de orden fraccionario.

Los módulos de identificación de sistemas de las herramientas de programación mencionadas no toman en cuenta sistemas no lineales; por lo que la

contribución de esta tesis tiene un enfoque diferente ya que se modelaron sistemas no lineales los cuales requieren de mayor complejidad en la formulación del algoritmo de identificación, sin embargo se pretende aprovechar las herramientas que se encuentran desarrolladas en los toolboxes para la programación de la identificación de modelos no lineales Wiener-Hammerstein.

1.5.1. Identificación de Sistemas

En identificación de sistemas se busca construir modelos matemáticos de sistemas dinámicos basados en datos observados del sistema. Los sistemas dinámicos son abundantes en nuestro entorno y existe la necesidad de la obtención de un modelo matemático que se ajuste a nuestro sistema sin el uso de las leyes físicas que lo rigen, por tal motivo las técnicas de identificación de sistemas tienen una amplia área de aplicación entre las cuales destacamos los siguientes: sistemas electrónicos, sistemas biofísicos, procesos químicos, por mencionar algunos.

Sistemas dinámicos

Un sistema es aquel en el que variables de diferentes tipos interactúan y producen señales observables. Las señales observables que son de interés para identificar un sistema son llamadas salidas, mientras que las señales externas que pueden ser manipuladas por el operador son llamadas entradas. Otras señales son llamadas disturbios y éstas se dividen en las que se miden directamente y las que solo se observan a través de su influencia en la salida tal y como se muestra en la Figura 1.2.

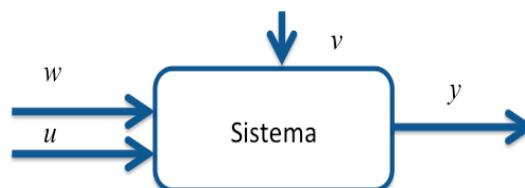


Figura 1.2: Sistema dinámico, y es la salida, u es la entrada, w es la perturbación medida y v es la perturbación no medida

Obtención de modelos

Existen dos maneras de obtener el modelo de un sistema:

1.- Modelo teórico: Se trata de un método analítico, en el que se recurre a leyes físicas que se encargan de describir el comportamiento dinámico del sistema.

2.- Identificación de sistemas: Se trata de un método experimental que permite obtener el modelo de un sistema a partir de la captura de datos de entrada al sistema y salida del mismo.

Procedimiento de identificación de sistemas

La construcción de un modelo a partir de datos involucra tres entidades básicas:

- El registro de datos de entrada y salida. El usuario determina que señales medir, cuando medirlas y selecciona también las señales de entrada y salida, los datos deben proporcionar la mayor cantidad de información.
- El conjunto de modelos. Se obtiene un conjunto de modelos candidatos, esta es la elección más importante y, al mismo tiempo lo más difícil del procedimiento de identificación de sistemas ya que es aquí donde el conocimiento previo y la intuición ingenieril se debe combinar con las propiedades formales de los modelos. Es llamada caja negra aquel modelo en donde se ajustan los parámetros a los datos experimentales pero dicho modelo no refleja las consideraciones físicas del sistema. Los conjuntos de modelos con parámetros ajustables con interpretación física pueden ser llamados cajas grises.
- Determinación del mejor modelo del conjunto, guiados por los datos. Se refiere a la evaluación de la calidad del modelo se basa típicamente en cómo se desempeñan los modelos cuando intentan reproducir los datos medidos.

1.5.2. Validación del modelo

Finalmente se debe probar si el modelo es adecuado y si es válido para su propósito. Estas pruebas son conocidas como validación del modelo. Estas involucran varios procedimientos para validar considerando como el modelo relaciona

los datos observados con la aplicación del modelo al sistema. Un comportamiento deficiente del modelo en estos términos nos hace rechazarlo, mientras que un buen desempeño nos hará desarrollar cierta confianza en el modelo. Un modelo no puede ser nunca aceptado como una descripción final y real del sistema, sin embargo, puede ser reconocido como una descripción lo suficientemente buena de ciertos aspectos que son de nuestro particular interés.

A continuación se muestran los cálculos estadísticos para poder realizar una validación y comparación de resultados:

FIT

Proviene del inglés FIT-Ajuste, se refiere a la aproximación de una señal estimada a la señal original y su medida es proporcionada en porcentaje.

$$FIT = 100 \left(1 - \frac{\|y - \hat{y}\|}{\|y - \text{promedio}(y)\|} \right), \quad (1.1)$$

donde:

- y es la salida del sistema.
- \hat{y} es la salida estimada del sistema.
- $\|x\|$ Se refiere a la norma cuadrada de un vector x .

El valor medio del error de simulación

$$\mu_t = \frac{1}{N} \sum_{t=1}^N e(t), \quad (1.2)$$

donde:

- μ es el valor medio del error de simulación.
- $e(t)$ es el error de estimación $e(t) = \hat{y}(t) - y(t)$.

- N es el número de datos.

La desviación estándar del error

Nos dice cuánto tienden a alejarse los valores concretos de la media en una distribución de datos. La desviación estándar de un conjunto de datos es una medida de cuánto se desvían los datos de su media.

$$S_t = \sqrt{\frac{\sum_{t=1}^N (e(t) - \mu)^2}{N}}, \quad (1.3)$$

donde:

- S es la desviación estándar del error.
- μ es el valor medio del error de simulación.
- $e(t)$ es el error de estimación $e(t) = \hat{y}(t) - y(t)$.
- N es el número de datos.

La raíz cuadrada del error medio al cuadrado

El error debido a la estimación de la media poblacional a partir de las medias muestrales.

$$e_{rms} = \sqrt{\frac{\sum_{t=1}^N (e(t))^2}{N}}, \quad (1.4)$$

donde:

- N es el número de datos.

- $e(t)$ es el error de estimación: $e(t) = \hat{y}(t) - y(t)$.

El ciclo de identificación del sistema

En el libro de [Ljung, 1998] se menciona el ciclo de identificación de sistemas; primero se colectan los datos, después se elige un conjunto de modelos, por último se selecciona el mejor modelo de este conjunto. Es muy probable que el primer modelo obtenido no pase las pruebas de validación, debemos entonces regresar y revisar los pasos del procedimiento como se muestra en la Figura 1.3.

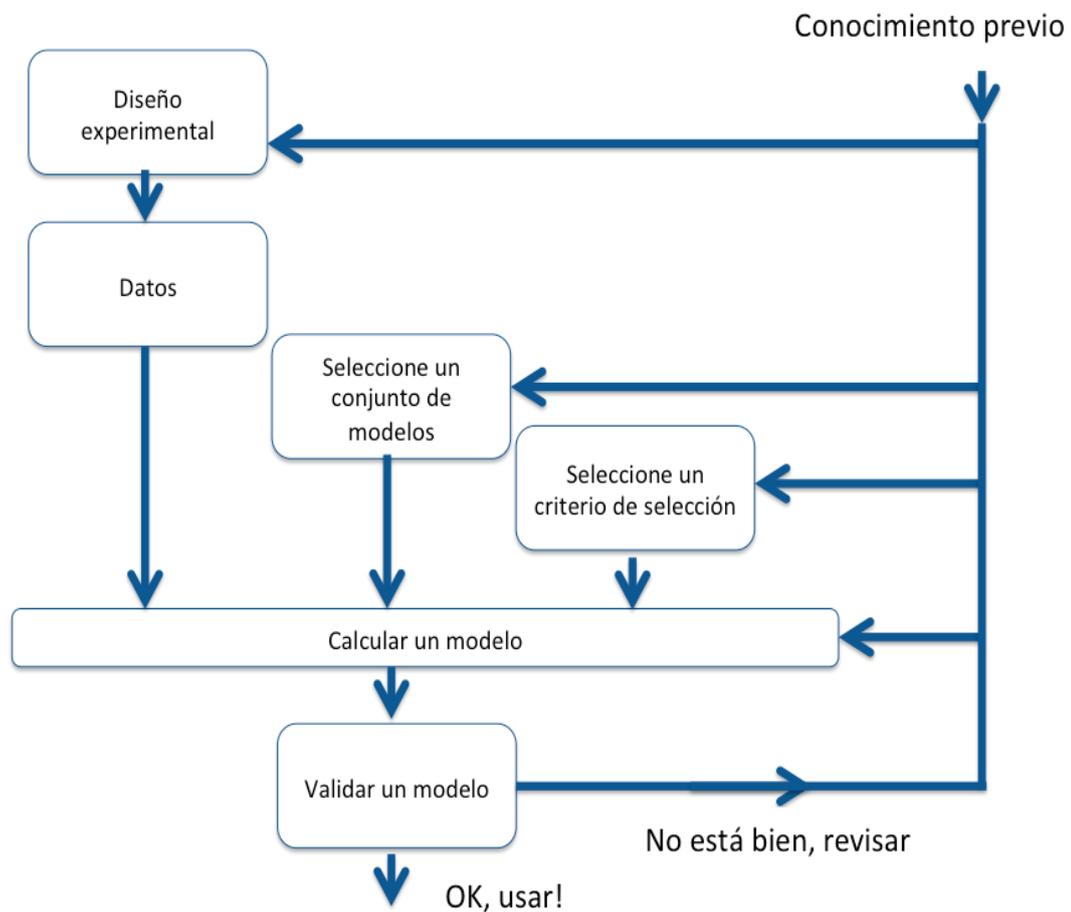


Figura 1.3: El ciclo de Identificación de Sistemas

El modelo puede ser deficiente por varias razones:

- El procedimiento numérico para encontrar el mejor modelo de acuerdo a nuestro criterio es deficiente.
- Los criterios no fueron bien elegidos.
- El conjunto de modelos no fue el apropiado, es decir, el conjunto no contiene ninguna descripción lo suficientemente buena del sistema.
- El conjunto de datos no proporciona suficiente información como para proveer de buenas bases en la selección de los modelos.

Métodos de optimización.

Cuando se conoce la estructura del modelo es necesario aplicar técnicas de optimización para determinar el vector de parámetros ($\theta = \hat{\theta}$) de tal manera que el modelo resultante $\hat{y} = f(u; \theta)$ pueda describir el sistema apropiadamente.

Antes de iniciar con cualquier algoritmo de optimización se necesita definir una función objetivo $E(\theta)$, que es la descripción matemática de lo que se desea optimizar o que tienda a cero, la elección más común es el error cuadrático medio, como se muestra en 1.5:

$$E(\theta) = \frac{1}{2}e(\theta)^2, \quad (1.5)$$

donde:

- e es la diferencia que existe entre salida del sistema y la salida estimada, como se representa en la ecuación 1.6.
- y es la salida del sistema.
- \hat{y} es la salida estimada.
- θ el vector de parámetros del modelo a identificar.

$$e(\theta) = y - \hat{y}. \quad (1.6)$$

En la Figura 1.4 se muestra el diagrama de la identificación paramétrica, donde la entrada u es aplicada al sistema y al modelo, la diferencia entre los dos

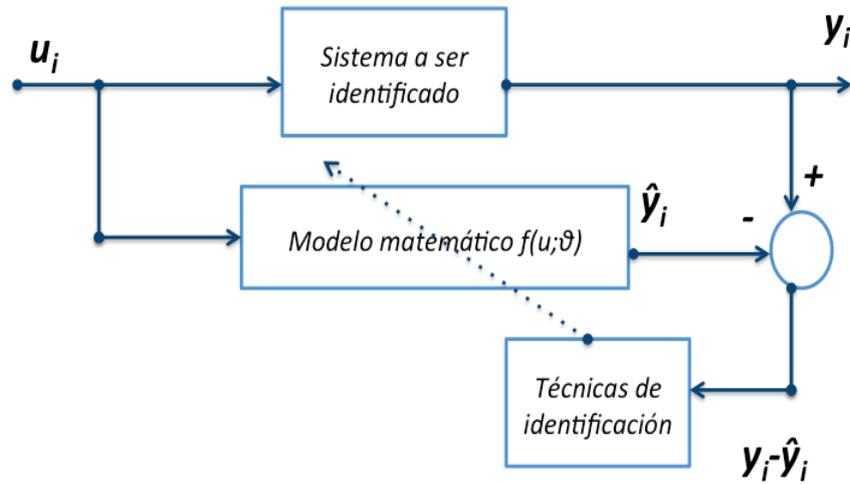


Figura 1.4: Diagrama a bloques de la identificación de parámetros

es utilizada de manera apropiada para actualizar el vector de parámetros θ y a su vez reducir la diferencia.

De acuerdo a los modelos utilizados en esta tesis se realizará una breve introducción de los métodos de optimización no lineales.

Gradiente descendente

Es una de las técnicas mas antiguas para la minimización de una función definida, este método es la base de muchas técnicas de optimización, y a pesar de su lenta convergencia, es utilizado frecuentemente para la optimización no lineal, gracias a su simplicidad esta descrito por la ecuación 1.7:

$$\theta_{k+1} = \theta_k - \eta g, \quad (1.7)$$

donde:

- $\eta > 0$ corresponde al coeficiente de aprendizaje.
- g es el gradiente de la función objetivo 1.5 como se muestra en 1.8

$$g(\theta_{k+1}) = \frac{\partial(E(\theta))}{\partial(\theta)} \Big|_{\theta=\theta_{k+1}}. \quad (1.8)$$

Idealmente se desea encontrar el valor de $\theta_{k+1} = \theta_k$ que satisfaga:

$$g(\theta_{k+1}) = \frac{\partial(E(\theta))}{\partial(\theta)} \Big|_{\theta=\theta_{k+1}} = 0. \quad (1.9)$$

Método de Newton

La dirección de descenso puede ser determinada utilizando segundas derivadas de la función objetivo $E(\theta)$, (si ésta puede ser determinada). Si la posición inicial está lo suficientemente cerca de un mínimo local, la función objetivo $E(\theta)$, puede ser aproximada por una función cuadrática mediante una expansión de series de Taylor:

$$E(\theta) = E(\theta_k) + g^T(\theta - \theta_k) + \frac{1}{2}((\theta - \theta_k)^T H((\theta - \theta_k))), \quad (1.10)$$

donde:

- H corresponde al Hessiano de $E(\theta)$, es decir la segunda derivada parcial de $E(\theta)$.

Considerando que se trata de una ecuación cuadrática, podemos encontrar su mínimo en el punto $\hat{\theta}$ diferenciando la ecuación 1.10 respecto a θ e igualando a cero.

$$0 = g + H(\hat{\theta} - \theta_k). \quad (1.11)$$

Asumiendo que la inversa de H existe, se despeja $\hat{\theta}$:

$$\hat{\theta} = \theta_k - H^{-1}g. \quad (1.12)$$

Levenberg-Marquardt

Debido a que H corresponde al Hessiano, es decir la segunda derivada parcial de $E(\theta)$, existe la posibilidad que la matriz H sea definida negativa o cero y por consecuencia envíe a un máximo local, para evitar dicho problema se debe asegurar que H es definida positiva, el Hessiano puede ser alterado adicionando una matriz λI y de esta manera hacer $(H + \lambda I)$ definida positiva asegurando que se llegue a un mínimo local. [Levenberg, 1944] y [Marquardt, 1963] introdujeron ese concepto representado en la ecuación 1.13

$$\theta_{k+1} = \theta_k - (H + \lambda I)^{-1}g, \quad (1.13)$$

donde:

- I es la matriz identidad.
- λ es el coeficiente de aprendizaje (un valor no negativo).

Se busca seleccionar una λ de manera que la matriz $H + \lambda I$ sea positiva definida. La ecuación anterior se aproxima al método de gradiente si $\lambda \rightarrow \infty$ y al método de Newton si $\lambda \rightarrow 0$.

1.6. Alcance de la tesis

El desarrollo de esta Tesis está acotado a realizar las siguientes actividades:

- Determinar los métodos del cálculo fraccionario susceptibles de ser probados en la identificación de modelos Wiener-Hammerstein.
- Desarrollar una metodología para la identificación de modelos Wiener-Hammerstein utilizando cálculo fraccionario, la cual proporcione modelos que representen de manera adecuada al sistema que se está identificando, buscando obtener buenos resultados durante las pruebas de validación
- Programar las funciones en Matlab necesarias para llevar a cabo la metodología planteada en la identificación de sistemas.
- Validar el modelo Wiener-Hammerstein entero y fraccionario en Matlab para verificar que la metodología desarrollada en esta tesis funciona correctamente.

- Realizar comparativo de desempeño de este trabajo contra trabajos previos en el CENIDET.
- Desarrollar una interfaz gráfica, que permita al usuario realizar la identificación de modelos Wiener-Hammerstein utilizando cálculo fraccionario de una manera sencilla.
- Obtener conclusiones de la comparación de desempeño de cada modelo.

1.7. Propuesta de solución

Para la realización de esta tesis se propuso seguir los siguientes pasos:

- Estudio de las técnicas clásicas de identificación.

Se llevará a cabo la revisión bibliográfica y un estudio de las técnicas clásicas de identificación lineales y no lineales, con el objetivo de desarrollar un metodología de identificación de sistemas utilizando cálculo fraccionario.

- Estudio sobre el cálculo fraccionario.

Se realizará una revisión bibliográfica sobre el cálculo fraccionario, se estudiaron diferentes definiciones y aplicaciones en identificación de sistemas. Esto se realizará para comprender las bases del cálculo fraccionario, y utilizar este conocimiento para poder seleccionar la definición y estructuras más adecuadas.

- Desarrollar de una metodología de identificación utilizando cálculo fraccionario.

Con base a los conocimientos adquiridos sobre el cálculo fraccionario y las técnicas de identificación no lineal, se logrará desarrollar una metodología de identificación de modelos Wiener-Hammerstein.

- Validar de la metodología.

Para poder corroborar la precisión del modelo resultante será necesario realizar el cálculo de parámetros de validación, la validación se realizará identificando distintos sistemas, verificando que los modelos obtenidos representen con cierta exactitud a los sistemas que se están identificando. Las

pruebas de validación propuestas en esta metodología son: el ajuste (o del inglés FIT), el valor medio del error, la desviación estándar del error y la raíz cuadrada del error medio al cuadrado.

- Programación del desarrollo.

Se desarrollaran las funciones que son necesarias para la identificación de modelos Winer-Hammerstein utilizando cálculo fraccionario, dichas funciones se encuentran programadas en Matlab.

- Comparación de Resultados

Finalmente para obtener conclusiones se realizará la comparación de desempeño de cada modelo con trabajos de identificación de sistemas realizados en el CENIDET y en la literatura.

1.8. Organización del documento

El documento de tesis está organizado de la siguiente manera:

En el **Capítulo 2** se hace una descripción de las definiciones y conceptos fundamentales del cálculo fraccionario, se mencionan la definición de integral fraccionaria, las dos definiciones más comunes de derivada fraccionaria: Riemann-Liouville y Grünwald-Letnikov, así como su relación con la función de Mittag-Leffler y la transformada de Laplace..

En el **Capítulo 3** se estudian ejemplos de identificación de modelos fraccionarios y se presenta el algoritmo de identificación de modelos Wiener-Hammerstein utilizando cálculo de orden generalizado.

En el **Capítulo 4** se realiza la implementación del algoritmo en tres sistemas diferentes, estos son: dos tanques acoplados, brazo-robot y un circuito electrónico y se presentan las tablas de las pruebas de validación realizadas.

En el **Capítulo 5** se presentan las conclusiones y las aportaciones de la tesis, y se plantean posibles investigaciones de trabajos futuros.

Capítulo 2

Introducción al Cálculo fraccionario

2.1. Introducción

El cálculo fraccionario es la rama del análisis de las matemáticas que estudia a los operadores de integración y derivación de orden no entero (D^n , donde n es un número real), sobre dominios de funciones reales o complejas, si bien existen muchas definiciones de cálculo fraccionario el principal objetivo de cada una de ellas es generalizar el concepto de derivada de orden entero de tal manera que para $n=1$ obtengamos el operador derivativo conocido comúnmente.

Los primeros escritos que se tienen del nacimiento del cálculo fraccionario datan de 1665, cuando L'Hopital le plantea por primera vez a Leibniz sobre la posibilidad de obtener la derivada de la función $d^n f(x)/dx^n$ cuando $n = 1/2$, posteriormente el cálculo fraccionario se siguió investigando por matemáticos reconocidos, entre los principales podemos mencionar a Euler, Laplace, Lacroix, Fourier, Riemann, Hardy y Littlewood, sin embargo las aún extensas investigaciones en cálculo ordinario no permitieron profundizar de manera esperada en el planteamiento de técnicas alternativas e innovadoras.

Entre los principales científicos investigadores del cálculo fraccionario destaca el trabajo de Liouville, el cual publica un gran número de artículos en donde se busca dar un primera definición formal de derivada e integral fraccionaria, por lo que se le considera el creador de las bases del cálculo fraccionario. Su primera investigación se basa en generalizar la derivada de orden entero de una función exponencial a derivadas de orden n arbitrarias:

$$D^n e^{ax} = a^n e^{ax}. \tag{2.1}$$

Posterior a los trabajos de Liouville surgieron las definiciones fraccionarias de Riemann-Liouville (1870-1884) y su discretización planteada por Grünwald-Letnikov (1867-1868), por mencionar algunas definiciones, ya que existen diversas propuestas de derivada e integral fraccionaria así como estudios en investigaciones en el tema. En el artículo de [De Oliveira and Machado, 2014] se presentan algunas de las definiciones alternativas, es importante mencionar que es un área en continua investigación, siendo recientemente publicadas dos nuevas definiciones de derivada fraccionaria [Caputo and Fabrizio, 2015, Atangana and Baleanu, 2016].

Entre los principales objetivos se encuentra el dar una mayor consistencia, claridad y rigor a la teoría matemática sobre los operadores fraccionarios. Lo cual se logrará unificando los conceptos y estableciendo definiciones estándar.

2.2. Integral fraccionaria

Para poder abordar algunas definiciones de los operadores de integración y derivación fraccionaria se presentan algunos conceptos del cálculo elemental que sirven como elementos base para construir la teoría fraccionaria, en el libro de [Podlubny, 1998] podemos encontrar un acercamiento matemático más formal y riguroso.

La n -ésima derivada de una función f está definida recursivamente por:

$$D^n f(t) = D [D^{n-1} f(t)], \quad n \in N. \quad (2.2)$$

En el caso de que $n = 0$ se obtiene la función original $f(t)$, lo que garantiza que la función no esta siendo alterada.

De igual manera la n -ésima integral de una función f está definida recursivamente por:

$$I^n f(t) = \int_0^t I^{n-1} f(t) dt, \quad n \in N, \quad (2.3)$$

en donde:

$$D^0 f(t) = I^0 f(t) dt. \quad (2.4)$$

Cauchy planteó una manera de describir la 2.3 y demostró que al ser cierto 2.5, $I^n f(t)$ puede ser reducida a una integral de convolución.

$$I^n f(t) = \frac{1}{(n-1)!} \int_0^t (t-\tau)^{n-1} f(\tau) d\tau, \quad (2.5)$$

donde:

- t y 0 son los límites de integración, por lo tanto $t > a$.
- n es el orden de integración.

Dicho resultado es conocido como **Fórmula de Cauchy** y sirve como antecedente de la integral fraccionaria.

La forma clásica del cálculo fraccionario está dada por la integral de Riemann-Liouville, Joseph Liouville fué el primero en considerar la posibilidad de cálculo fraccionario en 1832.

Considerando la ecuación 2.5 e intercambiando a n por un número $\alpha < 0$ y reemplazando al factorial por la Función Gamma (2.3.1), la definición de integral por Riemann-Liouville está dada por:

$$I^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-\tau)^{\alpha-1} f(\tau) d\tau, \quad (2.6)$$

donde:

- t y a son los límites de integración, por lo tanto $t > a$.
- α es el orden de integración.
- $\Gamma(\cdot)$ es la función Gamma.

2.3. Derivada fraccionaria

La derivada fraccionaria de Riemann-Liouville de orden α de $f(t)$ se define como:

$$D^\alpha f(t) = \frac{d^n}{dt^n} (I^{n-\alpha} f(t)), \quad (2.7)$$

donde:

- n es un número natural que satisfaga que $n - 1 < \alpha \leq n$.
- α es el orden de derivación fraccionario.

Cuando $\alpha = n \in \mathbb{N}$ se recupera el resultado del cálculo ordinario clásico.

Una segunda definición para tiempo discreto fue introducida por Anton Karl Grünwald (1838-1920) en Praga en 1867, y por Aleksey Vasilievich Letnikov (1837-1888) en Moscú en 1868, esta segunda definición es conocida como derivada de **Grünwald-Letnikov**:

$$D^\alpha f(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{k=0}^{\frac{t-t_0}{h}} (-1)^k \binom{\alpha}{k} f(t - kh), \quad (2.8)$$

donde:

$$\binom{\alpha}{j} = \frac{\Gamma(\alpha + 1)}{\Gamma(j + 1)\Gamma(\alpha - j + 1)} \quad y \quad \binom{\alpha}{0} = 1, \quad (2.9)$$

- α es el orden de derivación.
- h es el cambio de derivación relativamente pequeño en t .
- k es el paso de derivación.

2.3.1. Función Gamma

La principal aportación de la función Gamma es la generalización de los factoriales $n!$ lo que permite que n se trate de un número real. Por lo tanto la función Gamma es una de las más importante y básicas dentro del cálculo fraccionario. En la ecuación 2.14 se muestra la definición matemática.

$$\Gamma(z) = \int_0^{\infty} e^{-t} t^{z-1} dt, \quad (2.10)$$

que converge en la mitad derecha del plano complejo $\Re(z) > 0$. Por lo tanto la función Gamma es continua para los números reales positivos.

La propiedad básica de la función Gamma es:

$$\Gamma(z + 1) = z\Gamma(z). \quad (2.11)$$

Si z es un número natural, entonces:

$$\Gamma(z + 1) = z!. \quad (2.12)$$

2.3.2. Función de Mittag-Leffler

Es una generalización de las funciones exponenciales que tiene importantes aplicaciones en el cálculo fraccionario. En el libro de [Podlubny, 1998] podemos encontrar la definición de la Función de Mittag-Leffler uniparamétrica la cual fue definida por Magnus Gustaf Mittag-Leffler como:

$$E_{\alpha}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + 1)}, \quad (2.13)$$

- α es un parámetro, donde $\alpha > 0$.

La función de Mittag-Leffler bi-paramétrica fue introducida por Agarwal y Erdelyi en 1953-1954.

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)}, \quad (2.14)$$

donde:

- α y β son parámetros, donde $\alpha > 0$ $\beta > 0$.

2.4. Transformada de Laplace

La transformada de Laplace de $D_0^\alpha f(t)$ para la definición de Riemann- Liouville se muestra en la ecuación 2.15:

$$\mathcal{L} \{D_0^n f(t)\} = s^n F(s). \quad (2.15)$$

La ecuación 2.15 se cumple si y solo si las definiciones 2.7 y 2.8 son equivalentes cuando se evalúan en sus condiciones iniciales como se muestra en la ecuación 2.16.

$$f(t_0) = D_{t_0}^1 f(t_0) = \dots = D_{t_0}^\infty f(t_0) = 0. \quad (2.16)$$

Este resultado es coherente con el caso clásico cuando n es un entero. De tal manera que es fácil definir una representación de un sistema dinámico como una representación en función de transferencia.

El modelo de un sistema fraccionario, considerado un sistema SISO lineal invariante en el tiempo, puede ser descrito mediante la siguiente ecuación diferencial:

$$a_0 y(t) + a_1 D^{n_{a1}} y(t) + \dots + a_L D^{n_{aL}} y(t) = b_1 D^{n_{b1}} u(t) + \dots b_J D^{n_{bJ}} u(t), \quad (2.17)$$

donde:

- $a_0 \dots a_L, b_1 \dots b_J$ son los parámetros y se trata números reales.
- $u(t)$ y $y(t)$ son las señales de entrada y salida, las cuales son diferenciables en órdenes reales (enteros o no enteros).
- $n_{a1} \dots n_{aL}, n_{b1} \dots n_{bJ}$ son los órdenes de la derivada y pertenecen a los números reales

La transformada de Laplace de $D^{n_a}y(t)$ y $D^{n_b}u(t)$ son $s^{n_a}Y(s)$ y $s^{n_b}U(s)$, por lo que aplicando la transformada de Laplace a la ecuación 2.17, se obtiene lo siguiente:

$$a_0Y(s) + a_1s^{n_{a1}}Y(s) + \dots + a_Ls^{n_{aL}}Y(s) = b_1s^{n_{b1}}U(s) + \dots + b_Js^{n_{bJ}}U(s). \quad (2.18)$$

Las definiciones mencionadas anteriormente nos serán de gran ayuda para la formulación del algoritmo para la Identificación de Modelos Wiener-Hammerstein fraccionarios, principalmente la definición descrita por **Grünwald-Letnikov** y las propiedades de la **Transformada de Laplace**.

Capítulo 3

Estructura Wiener-Hammerstein

3.1. Introducción

Modelar e identificar un sistema dinámico no lineal es una tarea retadora ya que los procesos no lineales son únicos en el sentido de que no comparten muchas propiedades. Un objetivo importante en el modelado e identificación de sistemas no lineales es que sea universal, es decir, que sean capaces de describir diversas clases de estructuras para diferentes sistemas.

Entre las diferentes maneras de obtener modelos no lineales de sistemas dinámicos se encuentran los orientados a bloques, entre estos, los principales son: los modelos Hammerstein, Wiener y la combinación de los dos; Hammerstein-Wiener (H-W) y Wiener-Hammerstein (W-H) como se muestra en la Figura 3.1:

En el libro de [Nelles, 2002] se explican las características de los modelos Hammerstein y Wiener:

- El modelo Hammerstein es el más conocido y aplicado para describir a las dinámicas no lineales, se asume una separación entre las no linealidades y la dinámica del proceso, consiste en un bloque estático no lineal seguido de un bloque dinámico lineal. La estructura describe a aquellos sistemas donde el actuador es no lineal, por ejemplo las características de una válvula, la saturación de un motor electromagnético, etc. Otra ventaja es que la estabilidad está determinada por la parte lineal del modelo.
- El modelo Wiener es la estructura contraria al modelo Hammerstein, consiste en un bloque dinámico lineal seguido de un bloque estático no lineal,

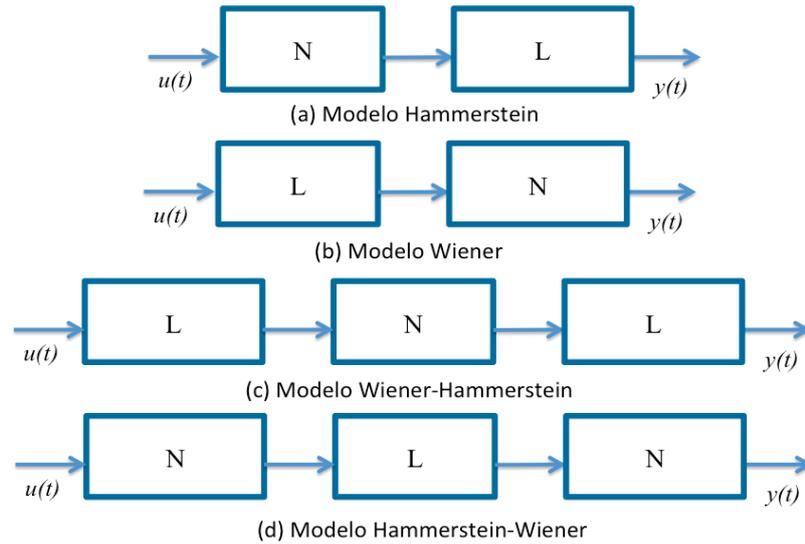


Figura 3.1: Modelos Hammerstein, Wiener, Wiener-Hammerstein y Hammerstein-Wiener

pocos procesos se ajustan a los supuestos de la estructura, entre los principales casos se encuentran aquellos en los que la no linealidad se encuentra en el sensor.

3.2. Antecedentes

En esta sección se plantean y ejemplifican dos diferentes algoritmos de identificación de sistemas fraccionarios basados en función de transferencia, aplicando la definición de **Grünwald-Letnikov** y la **Transformada de Laplace** a la derivada de orden fraccionario.

El primer caso se trata de un modelo lineal y el método de optimización utilizado es el de mínimos cuadrados, el segundo se trata de un modelo no lineal del tipo Hammerstein en el cual la identificación de parámetros se realiza con base en el algoritmo de Levenberg-Marquardt.

3.2.1. Identificación de modelos lineales utilizando cálculo fraccionario

Se tiene como objetivo obtener un modelo fraccionario que represente al sistema lineal de la Figura 3.2, la parte lineal H es del tipo de modelos lineales fraccionarios como se muestra en la ecuación 3.1

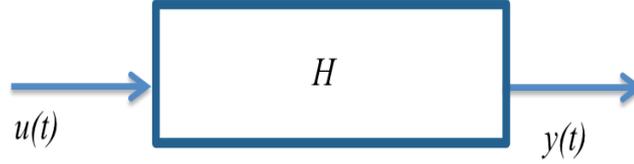


Figura 3.2: Modelo Lineal.

$$y(t) + a_1 D^{n_{a1}} y(t) + \dots + a_L D^{n_{aL}} y(t) = b_0 u(t) + b_1 D^{n_{b1}} u(t) + \dots + b_J D^{n_{bJ}} u(t), \quad (3.1)$$

donde:

- $a_1 \dots a_L, b_0 \dots b_J$ son números reales.
- $u(t)$ y $y(t)$ son las señales de entrada y salida, las cuales son diferenciables en órdenes reales (enteros o no enteros).
- $n_{a1} \dots n_{aL}, n_{b1} \dots n_{bJ}$ son los ordenes de la derivada reales (enteros o no enteros).

Si despejamos $y(t)$ como se muestra 3.2 para poder formular la salida estimada de manera vectorial.

$$\hat{y}(t) = - \sum_{l=1}^L a_l D^{n_{a_l}} y(t) + \sum_{j=0}^J b_j D^{n_{a_j}} u(t). \quad (3.2)$$

Para la obtención de una notación mas compacta, se introducen vectores que representen a la salida estimada como se muestra en la ecuación 3.3

$$\hat{y}(t) = [y(t), \dots, y(0)]^T = \Phi \theta, \quad (3.3)$$

donde:

- θ : corresponde al vector de parámetros a identificar mediante la *identificación paramétrica*.

$$\theta = [a_1, \dots, a_L, b_0, \dots, b_j]^T. \quad (3.4)$$

- Φ : se trata del vector de derivadas fraccionarias de los datos de entrada y salida, Φ_0 corresponde a la matriz de derivadas fraccionarias de los datos de salida $y(t)$ y Φ_1 a la matriz de derivadas fraccionarias de los datos de entrada $u(t)$.

$$\Phi = [\Phi_0, \Phi_1], \quad (3.5)$$

$$\Phi_0 = \begin{bmatrix} -D^{n_{a1}}y(0) & \dots & -D^{n_{aL}}y(0) \\ \vdots & \ddots & \vdots \\ -D^{n_{a1}}y(t) & \dots & -D^{n_{aL}}y(t) \end{bmatrix}, \quad (3.6)$$

$$\Phi_1 = \begin{bmatrix} -D^{n_{b1}}u(0) & \dots & -D^{n_{bJ}}u(0) \\ \vdots & \ddots & \vdots \\ -D^{n_{b1}}u(t) & \dots & -D^{n_{bJ}}u(t) \end{bmatrix}. \quad (3.7)$$

El error de predicci3n se deduce como:

$$e(t, \hat{\theta}) = y(t) - \hat{y}(t) = y(t) - \Phi \hat{\theta}, \quad (3.8)$$

donde:

- $y(t)$: es la salida del sistema.
- $\hat{y}(t)$: es la salida estimada.

Se tiene como objetivo encontrar la solución óptima del sistema, en donde se busca minimizar la función objetivo $J(\theta)$, es decir que tienda a cero, con ayuda del método de optimización de mínimos cuadrados.

$$J(\theta) = \int_0^N e^2(t, \hat{\theta}) dt. \quad (3.9)$$

Sustituyendo 3.8 en 3.9, se obtiene la ecuación 3.10

$$J(\theta) = (y(t) - \Phi(t)\hat{\theta})^T (y(t) - \Phi(t)\hat{\theta}). \quad (3.10)$$

El mínimo de la función objetivo será aquel valor de θ que hace que la derivada sea igual a cero:

$$\frac{dJ(\theta)}{d\theta} = 2(\Phi(t)\hat{\theta} - y(t))^T \Phi = 0. \quad (3.11)$$

De esa manera es posible despejar a $\hat{\theta}$ de la ecuación 3.11 y encontrar los parámetros del modelo fraccionario.

$$\hat{\theta} = [\Phi^T \Phi]^{-1} \Phi y(t). \quad (3.12)$$

Aplicando la *Transformada de Laplace* a la ecuación 3.1 se obtiene la ecuación 3.13, que permite obtener la función de transferencia $H(s)$ del modelo lineal:

$$a_0 Y(s) + a_1 s^{n_{a1}} Y(s) + \dots + a_L s^{n_{aL}} Y(s) = b_1 s^{n_{b1}} U(s) + \dots + b_J s^{n_{bJ}} U(s). \quad (3.13)$$

La función de transferencia $H(s)$ derivada de la *identificación de parámetros* es la siguiente:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_1 s^{n_{b1}} + \dots + b_J s^{n_{bJ}}}{a_1 s^{n_{a1}} + \dots + a_L s^{n_{aL}}}. \quad (3.14)$$

3.2.1.1. Ejemplo: Circuito RC

Para poder llevar a cabo un ejemplo de la identificación de sistemas lineales fraccionarios se diseñó un circuito RC que consta de una resistencia y un capacitor como se muestra en la Figura 3.3. El valor de la resistencia R1 es de $1k\Omega$ y del capacitor C1 de $100 \mu\text{F}$. Se decidió tomar como entrada $u(t)$ al voltaje aplicado en la fuente, siendo esta una secuencia binaria pseudoaleatoria (PRBS por sus siglas en inglés: Pseudorandom binary sequence), y la salida $y(t)$ es el voltaje en el capacitor C1.

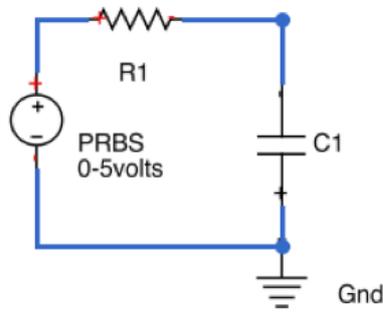


Figura 3.3: Circuito RC.

La solución en función de transferencia utilizando las Leyes de Kirchhoff es la representada en la ecuación 3.15, como se mencionó se toma en cuenta como entrada el voltaje aplicado en el circuito y como salida el voltaje en el capacitor.

$$H(s) = \frac{Y(s)}{U(s)} = \frac{1}{1 + RCs}. \quad (3.15)$$

Para llevar a cabo la identificación, se considera una función de transferencia final de la forma de un circuito RC como se muestra en [Calik, 2015]:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_1}{a_1 s^\gamma + 1}, \quad (3.16)$$

$$\begin{aligned} \theta &= [a_1, b_1]^T \\ \Phi &= [\Phi_0, \Phi_1]. \end{aligned} \quad (3.17)$$

Para el primer caso fraccionario las matrices correspondientes a Φ quedaron de la siguiente manera

$$\Phi_0 = \begin{bmatrix} -D^\gamma y(0) \\ \vdots \\ -D^\gamma y(t) \end{bmatrix}, \quad (3.18)$$

$$\Phi_1 = \begin{bmatrix} u(0) \\ \vdots \\ u(t) \end{bmatrix}. \quad (3.19)$$

Con la finalidad de tener una compatibilidad dimensional, en el segundo caso se decidió utilizar el tiempo de Planck, el cual es una unidad de tiempo, considerada como el intervalo temporal más pequeño que puede ser medido y es igual a $5,39 \times 10^{-44} \text{s}$, para el ajuste de unidades como se muestra en la ecuación 3.22 y de esta manera formar los vectores Φ_0 y Φ_1

$$\Phi_0 = \begin{bmatrix} \frac{-D^\gamma y(0)}{(t_{Planck})^{1-\gamma}} \\ \vdots \\ \frac{-D^\gamma y(t)}{(t_{Planck})^{1-\gamma}} \end{bmatrix}, \quad (3.20)$$

$$\Phi_1 = \begin{bmatrix} u(0) \\ \vdots \\ u(t) \end{bmatrix}. \quad (3.21)$$

En la siguiente ecuación se observa como se lleva a cabo la normalización de unidades, donde seg es la unidad de tiempo en segundos y t_{Planck} corresponde al tiempo de Planck igual a $5,39 \times 10^{-44} \text{seg}$

$$\frac{D^\gamma}{(t_{Planck})^{1-\gamma}} = \left[\frac{1}{\frac{seg^\gamma}{seg^{1-\gamma}}} = \frac{1}{seg} \right]. \quad (3.22)$$

La función de transferencia resultante en el primer caso es la siguiente:

$$H(s) = \frac{0,989}{1,483s^{0,99} + 1} \quad (3.23)$$

La función de transferencia resultante para el segundo caso utilizando el t_{Planck} es:

$$H(s) = \frac{0,9949}{1,788s^{1,05} + 1}. \quad (3.24)$$

Para el caso de orden entero se realizó la identificación utilizando el toolbox de MatLab, y se obtuvo un modelo de la ecuación 3.25

$$H(s) = \frac{0,743s + 2,587}{s^2 + 4,8s + 2,59}. \quad (3.25)$$

Las gráficas resultantes de los modelos se encuentran en las Figuras 3.4 y 3.5, en donde podemos interpretar visualmente que los tres modelos identificados cuentan con una buena aproximación al modelo original sin embargo, es difícil tomar una decisión acertada si no se hace un análisis mediante pruebas de validación, por ejemplo: como el cálculo del FIT (del inglés FIT- Ajuste, se refiere a la aproximación de una señal estimada a la señal original) o el número de parámetros identificados por mencionar algunos.

Tabla 3.1: Parámetros de validación de modelos en el sistema RC.

Ref.	No. de parámetros	<i>FIT</i> %
Modelo fraccionario	3	93.79
Modelo fraccionario(T_{planck})	3	96.68
Modelo entero	5	99.33

Los modelos fraccionarios son más compactos, por lo tanto, cuando el sistema pertenece a la clase de sistemas cuyas dinámicas tienen un orden fraccionario; el número de parámetros a optimizar se reduce considerablemente. De tal manera que este enfoque es especialmente útil cuando la dinámica del sistema revela

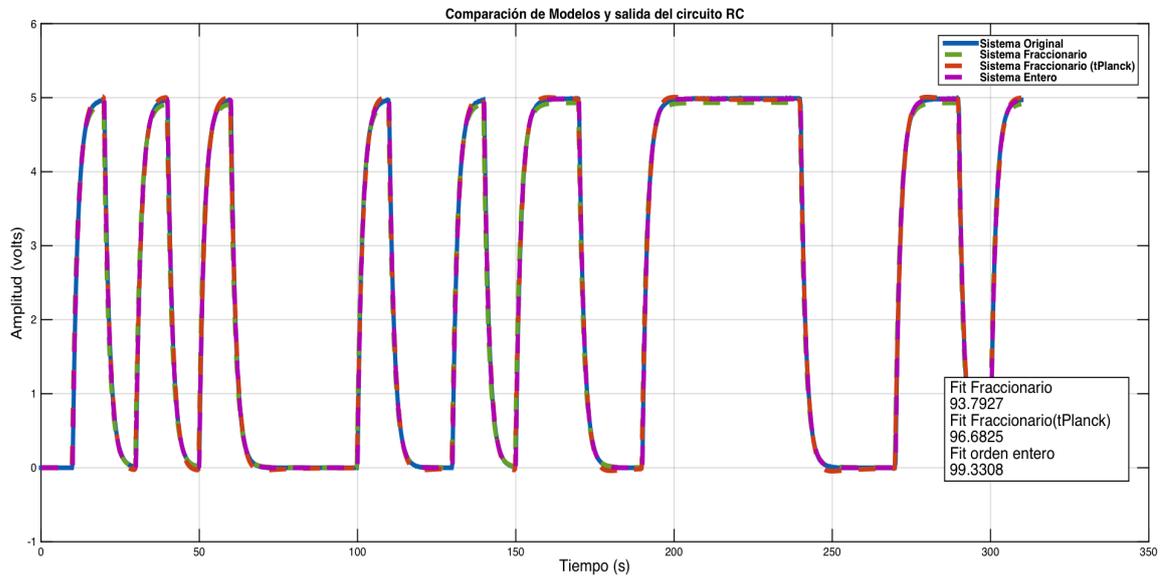


Figura 3.4: Comparación de modelos y salida del Sistema del circuito RC.

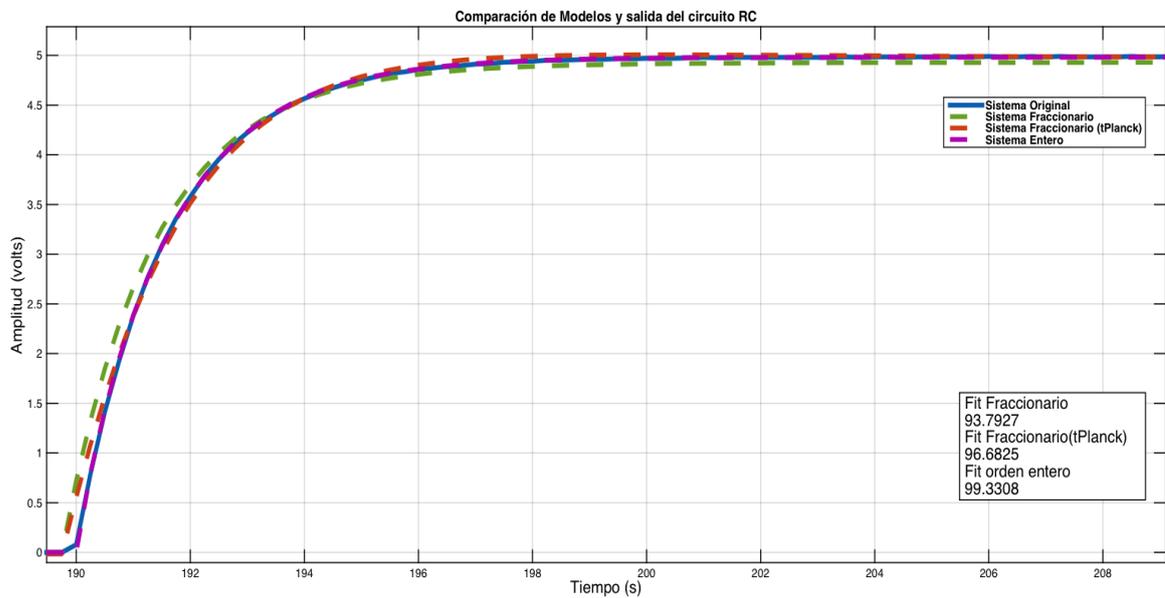


Figura 3.5: Acercamiento de la comparación de modelos y salida del sistema del circuito RC.

comportamientos fraccionarios, abriendo la posibilidad de futuras investigaciones de la interpretación física del modelo obtenido.

Es importante notar que el modelo entero tiene un FIT mas alto, sin embargo la función de transferencia obtenida no representa al modelo esperado de una función de transferencia para un circuito RC de acuerdo a las leyes físicas que rigen al sistema. Las funciones de transferencia obtenidas en 3.23 y 3.24 representan una función de transferencia de un circuito RC donde el capacitor es un elemento generalizado llamado elemento de fase constante (o capacitor imperfecto) [Gómez-Aguilar et al., 2016a] y [Gómez-Aguilar et al., 2016c]. Debido a la introducción del tiempo de Planck, la función de transferencia representada en 3.24 es consistente con las unidades permitiendo la reproducibilidad en el laboratorio. Las funciones de transferencia 3.23 y 3.24 no solamente corresponden a las funciones de transferencia esperadas en un modelo RC sino que las generalizan, donde en el caso particular $\gamma = 1$ se obtiene la función de transferencia clásica de un circuito RC.

3.2.2. Identificación de modelos Hammerstein utilizando cálculo fraccionario

El modelo Hammerstein es el más conocido y aplicado para describir dinámicas no lineales, se asume una separación entre las no linealidades y la dinámica del proceso, consiste en un bloque estático no lineal seguido de un bloque dinámico lineal. La estructura describe a aquellos sistemas donde el actuador es no lineal, por ejemplo: las características de una válvula, la saturación de un motor electromagnético, etc.

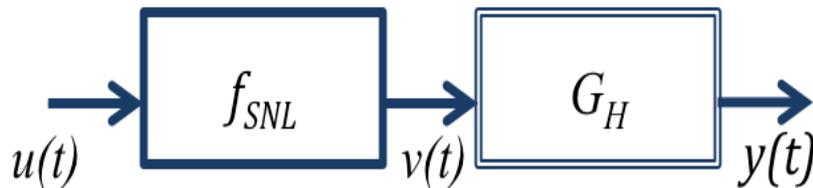


Figura 3.6: Modelo Hammerstein.

donde:

- $v(t) = f_{SNL}(u(t))$, es una función no lineal de transformación.
- $y(t) = G_H(q)v(t)$, es una función de transferencia lineal.

Dónde se asume que la parte No Lineal (f_{SNL}) se puede aproximar por una expansión de polinomios como se muestra en la ecuación 3.26 :

$$v(t) = f_{SNL}(u(t)) = \sum_{k=0}^M \alpha_k u^k(t), \quad (3.26)$$

donde:

- α : son los parámetros del polinomio.
- k : es el orden del polinomio.
- $u(t)$ es la entrada del sistema y del polinomio.

La parte lineal G_H es del tipo de modelos lineales fraccionarios.

$$\begin{aligned} a_{H0}y(t) + a_{H1}D^{n_{Ha1}}y(t) + \dots + a_{HL}D^{n_{HaL}}y(t) \dots \\ \dots = b_{H0}v(t) + b_{H1}D^{n_{Hb1}}v(t) + \dots + b_{HJ}D^{n_{HbJ}}v(t), \end{aligned} \quad (3.27)$$

donde:

- $a_{H0}, \dots, a_{HL}, b_{H0}, b_{H1}, \dots, b_{HJ}$ son números reales.
- $v(t)$ y $y(t)$ son las señales de entrada y salida al modelo G_H , respectivamente, las cuales son diferenciables en órdenes reales (enteros o no enteros).
- $n_{Ha1}, \dots, n_{HaL}, n_{Hb1}, \dots, n_{HbJ}$ son los órdenes de derivación reales (enteros o no enteros).

Por lo tanto la función de transferencia generalizada $G_H(s)$ es:

$$G_H(s) = \frac{Y(s)}{V(s)} = \frac{b_{H1}s^{n_{Hb1}} + \dots + b_{HJ}s^{n_{HbJ}}}{a_{H1}s^{n_{Ha1}} + \dots + a_{HL}s^{n_{HaL}}}. \quad (3.28)$$

El vector de parámetros a identificar $\theta = [\theta_{NL}, \theta_H]^T$ está conformado como se muestra a continuación:

$$\begin{aligned}\theta_{NL} &= [\alpha_0, \dots, \alpha_k]^T, \\ \theta_H &= [a_{HL}, \dots, a_{H0}, b_{HJ}, \dots, b_{H0}, n_{HaL}, \dots, n_{Ha1}, n_{HbJ}, \dots, n_{Hb1}]^T.\end{aligned}\quad (3.29)$$

La representación de la salida estimada se plantea de la siguiente manera:

$$\hat{y}(t) = \mathcal{L}^{-1}(G_H) * \sum_{k=0}^M \hat{\alpha}_k [u(t)]^k. \quad (3.30)$$

Un método de optimización no lineal debe ser utilizado, cuyo objetivo es minimizar la función objetivo $J(\theta)$ que en este caso es la integral del error cuadrático como se muestra en la ecuación 3.31

$$J(\theta) = \int_0^k (e(t))^2 dt = \int_0^k (y(t) - \hat{y}(t))^2 dt, \quad (3.31)$$

donde:

- $J(\theta)$ es la función objetivo.
- e es el error de predicción.
- $y(t)$ es la salida del sistema.
- $\hat{y}(t)$ es la salida estimada que se encuentra en la ecuación 3.30.

El valor óptimo de θ se obtiene entonces iterativamente utilizando una técnica de optimización no lineal, por lo que se utilizó el algoritmo de Levenberg-Marquardt de la ecuación 3.32.

$$\theta_{k+1} = \theta_k - (H + \lambda I)^{-1} g, \quad (3.32)$$

donde:

$$\begin{aligned}
g &= -2 \int_0^t e(t)S(t, \hat{\theta})dt & : & \text{ gradiente,} \\
H &\approx 2 \int_0^t S(t, \hat{\theta})e(t)S^T(t, \hat{\theta})dt & : & \text{ Hessiano,} \\
S(t, \hat{\theta}) &= \frac{\partial \hat{y}(t, \hat{\theta})}{\partial \theta} & : & \text{ Funcion de sensibilidad,} \\
\lambda &: & \text{ Coeficiente de aprendizaje.}
\end{aligned} \tag{3.33}$$

3.2.2.1. Ejemplo: Secadora de cabello

Se llevó a cabo la implementación del algoritmo de identificación de modelos Hammerstein utilizando cálculo fraccionario, los datos utilizados para la realización de la identificación fueron obtenidos de la base de datos de identificación de sistemas DaiSy (Database for the Identification of Systems, por sus siglas en inglés).

El sistema consiste en una configuración de laboratorio que actúa como una secadora de cabello, el aire es desplegado a través de un tubo y es calentado en la entrada. La temperatura del aire se mide mediante un termopar en la salida y la entrada es la tensión aplicada en el dispositivo de calefacción (una malla de alambres de resistencias)

Se decidió utilizar un función de transferencia con 2 parámetros en el numerador y 3 parámetros en el denominador lo que corresponde a 1 derivada fraccionaria para la parte del numerador y 2 derivadas fraccionarias para el denominador. Para la parte no lineal se estableció un polinomio de segundo grado, la representación la podemos encontrar en la ecuación 3.35

$$v(t) = f_{SNL}(u(t)) = \alpha_2 u^2(t) + \alpha_1 u(t) + \alpha_0, \tag{3.34}$$

$$G_H(s) = \frac{Y(s)}{V(s)} = \frac{b_{H1}s^{n_{Hb1}} + b_{H0}}{a_{H2}s^{n_{Ha2}} + a_{H1}s^{n_{Ha1}} + a_{H0}}. \tag{3.35}$$

El vector de parámetros a identificar $\theta = [\theta_{NL}, \theta_H]^T$ está conformado como se muestra a continuación:

$$\begin{aligned}\theta_{NL} &= [\alpha_2, \alpha_1, \alpha_0]^T, \\ \theta_H &= [a_{H2}, a_{H1}, a_{H0}, b_{H1}, b_{H0}, n_{Ha2}, n_{Ha1}, n_{Hb1}]^T.\end{aligned}\quad (3.36)$$

La función de transferencia resultante para el caso del modelo Hammerstein fraccionario es la siguiente:

$$v(t) = f_{SNL}(u(t)) = 0,40139u^2(t) - 2,126u(t) - 0,28587, \quad (3.37)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{-0,086s^{0,367} + 0,075}{1,348s^{3,164} + 0,834s^{1,380} + 0,115}. \quad (3.38)$$

Para el caso de orden entero se realizó la identificación utilizando el toolbox de MatLab, y se obtuvo un modelo de la ecuación 3.40.

$$v(t) = f_{SNL}(u(t)) = 0,0087693u^2(t) - 0,043728u(t) - 0,020047, \quad (3.39)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{0,5579s + 1,173}{s^2 + 0,3193s + 0,05308}. \quad (3.40)$$

La gráfica resultante de los modelos se encuentran en la Figura 3.7, en donde podemos interpretar visualmente que los dos modelos identificados cuentan con una buena aproximación al modelo original sin embargo es necesario hacer un análisis mediante pruebas de validación como el cálculo del FIT o el número de parámetros identificados para poder tomar una decisión acertada.

Como se muestra en la tabla 3.2, el modelo fraccionario cuenta con un mejor FIT, esto se debe gracias al grado de libertad adicional que le da el orden de la derivada, sin embargo se deben estimar más parámetros durante la identificación del modelo. De tal manera que este enfoque resulta de gran importancia cuando la dinámica del sistema presenta comportamientos fraccionarios.

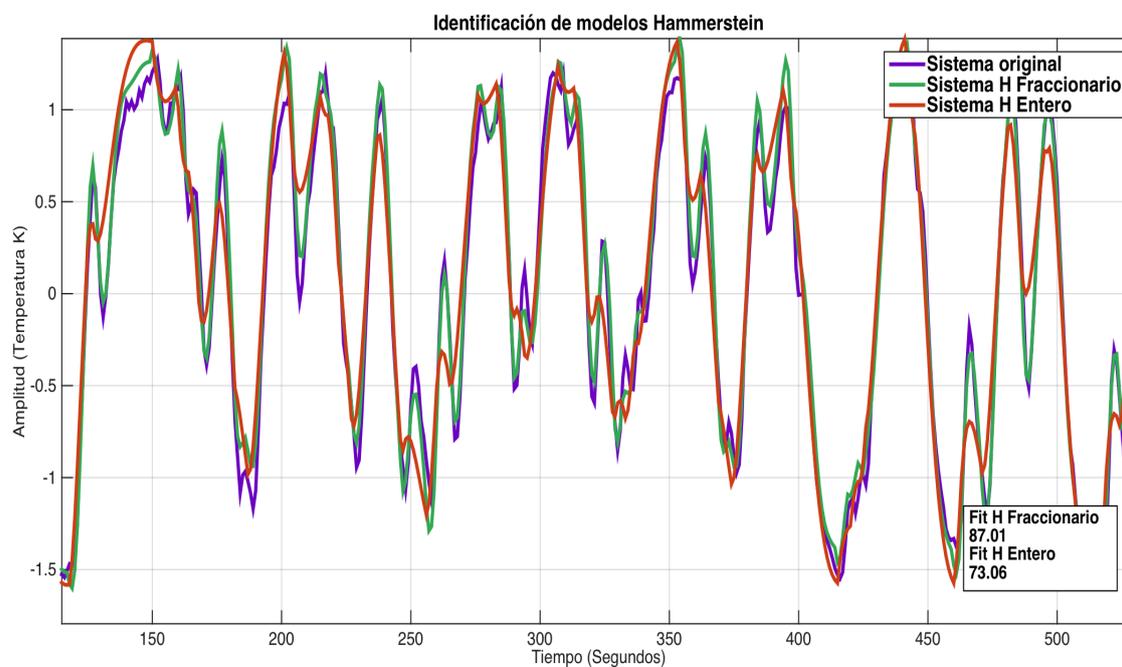


Figura 3.7: Modelo Hammerstein

Tabla 3.2: Parámetros de validación de modelos en el sistema de Secadora de Cabello.

Ref.	No. de parámetros	<i>FIT</i> %
Modelo fraccionario	11	87.01
Modelo entero	7	73.06

3.3. Representación de la estructura Wiener-Hammerstein

En un sistema W-H, un sistema no lineal está situado entre dos partes lineales invariantes en el tiempo G_W y G_H . La señal $u(t)$ representa la entrada y la señal $y(t)$ representa la salida. G_W y G_H son bloques lineales dinámicos como se muestra en la Figura 3.8, mientras que f_{SNL} se trata de un bloque estático.

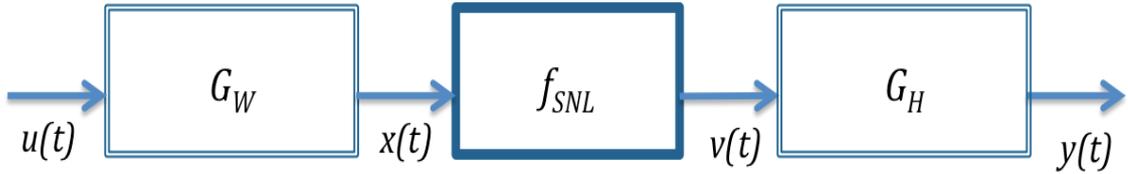


Figura 3.8: Modelo Wiener-Hammerstein.

donde:

- $x(t) = G_W(q)u(t)$, es una función de transferencia lineal.
- $v(t) = f_{SNL}(x(t))$, es una función de transformación no lineal.
- $y(t) = G_H(q)v(t)$, es una función de transferencia lineal.

3.4. Estructura Wiener-Hammerstein fraccionaria propuesta

Los bloques lineales G_H y G_W son del tipo de modelos lineales fraccionarios como de muestra en las Ecuaciones 3.41 y 3.43 y la parte No Lineal (f_{SNL}) se puede aproximar por una expansión de polinomios como se muestra en la ecuación 3.42

$$G_W(s) = \frac{X(s)}{U(s)} = \frac{b_{W1}s^{n_{wb1}} + \dots + b_{WJ}s^{n_{wbJ}}}{a_{W1}s^{n_{wa1}} + \dots + a_{WL}s^{n_{waL}}}, \quad (3.41)$$

$$v(t) = N(x(t)) = \sum_{k=0}^M \alpha_k x^k(t), \quad (3.42)$$

$$G_H(s) = \frac{Y(s)}{V(s)} = \frac{b_{H1}s^{n_{Hb1}} + \dots + b_{HJ}s^{n_{HbJ}}}{a_{H1}s^{n_{Ha1}} + \dots + a_{HL}s^{n_{HaL}}}, \quad (3.43)$$

donde:

- $a_{W1}, \dots, a_{WL}, b_{W1}, \dots, b_{WJ}$ son números reales.
- $a_{H1}, \dots, a_{HL}, b_{H1}, \dots, b_{HJ}$ son números reales.
- α : son los parámetros del polinomio.
- k : es el orden del polinomio.
- $n_{Wa1}, \dots, n_{WaL}, n_{Wb1}, \dots, n_{Wbj}$ son los órdenes de derivación reales (enteros o no enteros).
- $n_{Ha1}, \dots, n_{HaL}, n_{Hb1}, \dots, n_{Hbj}$ son los órdenes de derivación reales (enteros o no enteros).

El vector de parámetros a identificar $\theta = [\theta_W, \theta_{NL}, \theta_H]^T$ está conformado como se muestra a continuación:

$$\begin{aligned} \theta_W &= [a_{WL}, \dots, a_{W1}, b_{WJ}, \dots, b_{W1}, n_{WaL}, \dots, n_{Wa1}, n_{WbJ}, \dots, n_{Wb1}]^T, \\ \theta_{NL} &= [\alpha_0, \dots, \alpha_k]^T, \\ \theta_H &= [a_{HL}, \dots, a_{H0}, b_{HJ}, \dots, b_{H1}, n_{HaL}, \dots, n_{Ha1}, n_{HbJ}, \dots, n_{Hb1}]^T. \end{aligned} \quad (3.44)$$

La representación de la salida estimada se plantea de la siguiente manera:

$$\hat{y}(t) = \mathcal{L}^{-1}(G_H) * \sum_{k=0}^M \hat{\alpha}_k [\mathcal{L}^{-1}(G_W) * u(t)]^k. \quad (3.45)$$

Un método de optimización no lineal debe ser utilizado, cuyo objetivo es minimizar la función objetivo $J(\theta)$ que en este caso es la integral del error cuadrático como se muestra en la ecuación 3.46

$$J(\theta) = \int_0^k (e(t))^2 dt = \int_0^k (y(t) - \hat{y}(t))^2 dt, \quad (3.46)$$

donde:

- $J(\theta)$ es la función objetivo.
- e es el error de predicción.
- $y(t)$ es la salida del sistema.
- $\hat{y}(t)$ es la salida estimada que se encuentra en la ecuación 3.30.

El valor óptimo de θ se obtiene de manera iterativa utilizando una técnica de optimización no lineal, por lo que se utilizó el algoritmo de Levenberg-Marquardt de la ecuación 3.47.

$$\theta_{k+1} = \theta_k - (H + \lambda I)^{-1}g, \quad (3.47)$$

donde:

$$\begin{aligned} g &= -2 \int_0^t e(t)S(t, \hat{\theta})dt && : \text{ gradiente,} \\ H &\approx 2 \int_0^t S(t, \hat{\theta})e(t)S^T(t, \hat{\theta})dt && : \text{ Hessiano,} \\ S(t, \hat{\theta}) &= \frac{\partial \hat{y}(t, \hat{\theta})}{\partial \theta} && : \text{ Funcion de sensibilidad,} \\ \lambda &: \text{ Coeficiente de aprendizaje.} \end{aligned} \quad (3.48)$$

3.4.1. Diagrama a bloques

En la Figura 3.9 se presenta de manera reducida y en forma de diagrama a bloques el algoritmo de identificación de modelos Wiener-Hammerstein fraccionario propuesto en esta tesis.

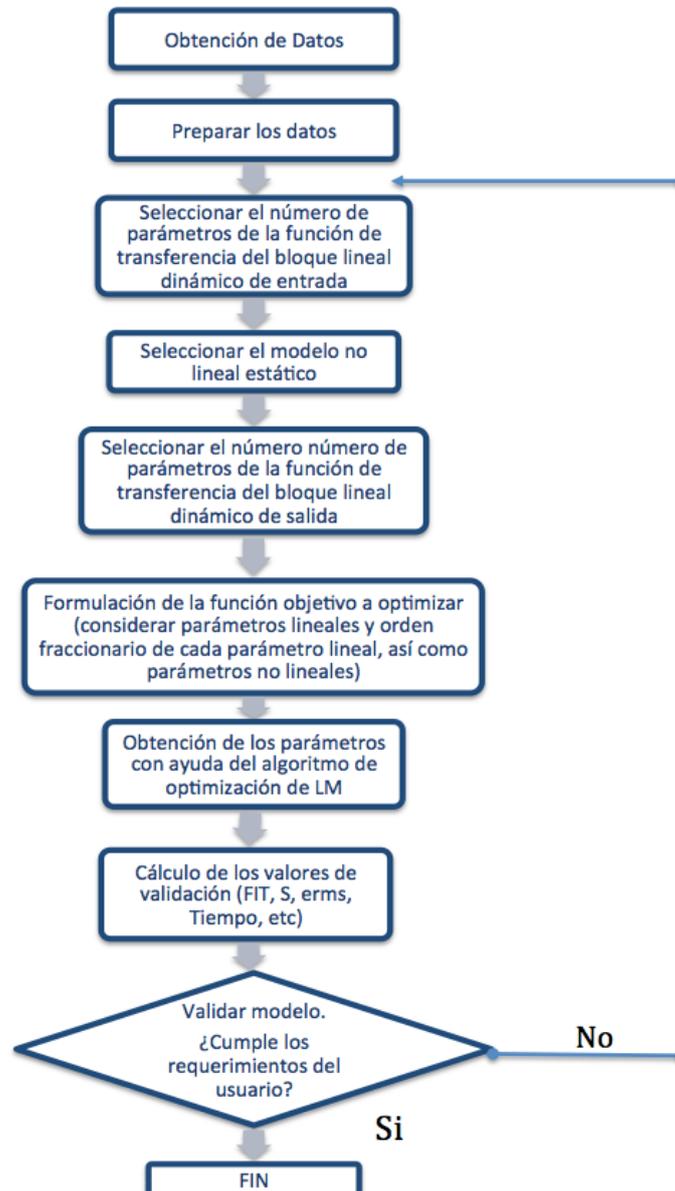


Figura 3.9: Diagrama a bloques de la obtención de modelos Wiener-Hammerstein fraccionarios.

Capítulo 4

Implementación del Algoritmo

4.1. Sistemas dinámicos identificados

Se prosiguió con la realización de la programación de los algoritmos de identificación mencionados para las estructuras Hammerstein y Wiener-Hammerstein, se decidió aplicar y obtener un modelo para tres casos de estudio y de esta manera verificar la adecuada representación de estos modelos mediante la validación numérica. Los sistemas no lineales que se identificaron son:

- Dos tanques acoplados.
- Brazo robot.
- Circuito electrónico no lineal.

4.1.1. Dos Tanques acoplados

El primer sistema a identificar consiste en un par de tanques, en el cual el agua fluye a través de una tubería al tanque 1 y desemboca en el tanque 2, el nivel de agua del sistema es ajustado a través de un pequeño orificio en el fondo del tanque 2; la entrada al sistema es la alimentación aplicada a la bomba que alimenta al tanque 1 (en volts) y la salida es la altura del agua en el tanque 2 (en metros), como se muestra en la Figura 4.1. Este sistema fue tomado del manual de identificación de Matlab, con el objetivo de poner a prueba a las estructuras no lineales desarrolladas, el conjunto de datos utilizado de entrada-salida, es de 3000 muestras.

En la Figura 4.2, se muestra la entrada y la salida que se obtuvieron resultado de la simulación.

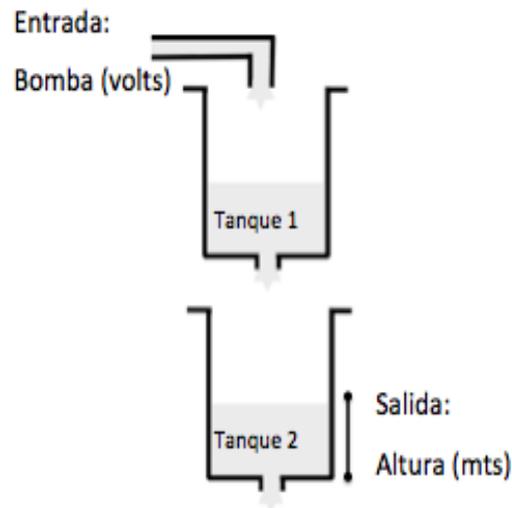


Figura 4.1: Sistema de Tanques.

4.1.2. Brazo robot

El siguiente sistema se obtuvo de la base de datos de identificación de sistemas DaISy [leuven, 2008], los cuales corresponden a un brazo robot. El brazo está instalado en un motor eléctrico. La salida corresponde a la aceleración angular del brazo flexible y es adquirida midiendo desde el par de reacción del brazo y la entrada aplicada es un barrido sinusoidal periódico, tanto la salida como la entrada se pueden apreciar en la Figura 4.3.

4.1.3. Circuito electrónico

Finalmente se utilizó un sistema propuesto en el artículo Wiener-Hammerstein benchmark [Schoukens and Ljung, 2009]. Se trata de un circuito electrónico no lineal, con la estructura de tipo Wiener Hammerstein como se muestra en la Figura 4.4.

El primer bloque lineal $G1(s)$ es un filtro Chebyshev de tercer orden, el segundo bloque lineal $G2(s)$ es un filtro Chebyshev inverso de tercer orden, el bloque no lineal estático está construido utilizando el circuito de un diodo (Figura 4.5), el conjunto de datos de entrada-salida del sistema es de 188000 datos muestreados y la frecuencia de muestreo es de 51200Hz.

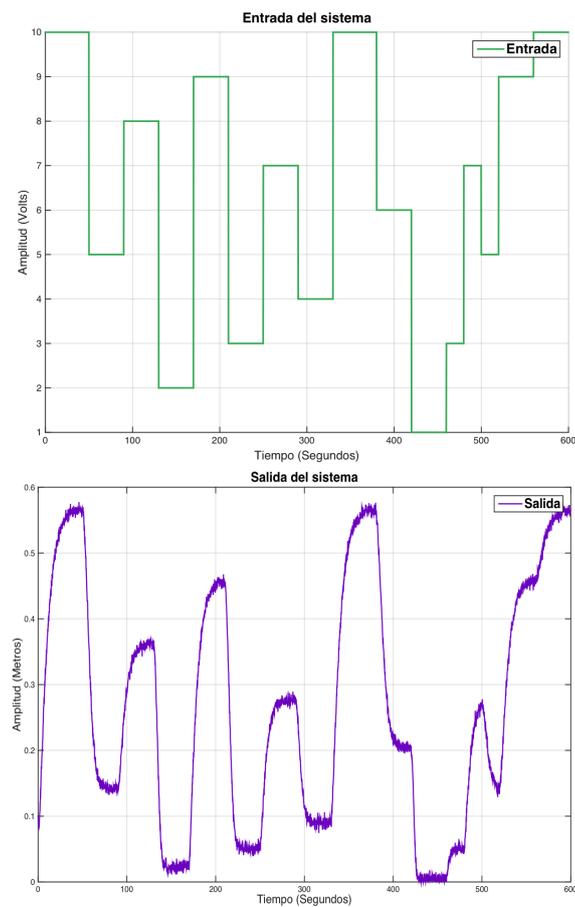


Figura 4.2: Entrada y salida del sistema de dos tanques acoplados

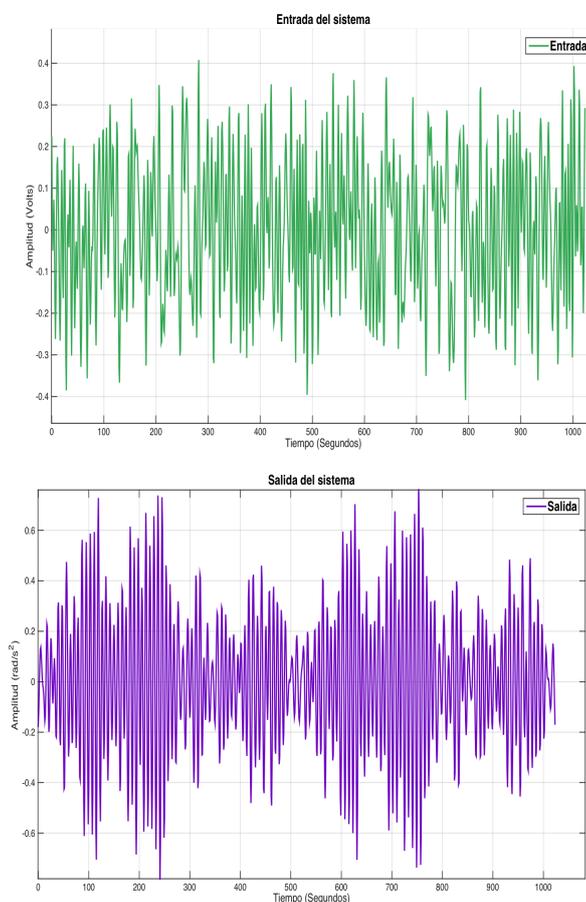


Figura 4.3: Entrada y salida del sistema de Brazo robot

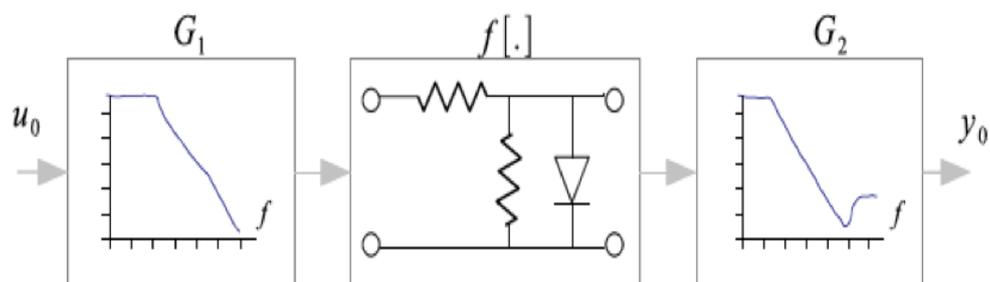


Figura 4.4: Modelo Wiener-Hammerstein del Circuito electrónico no lineal.

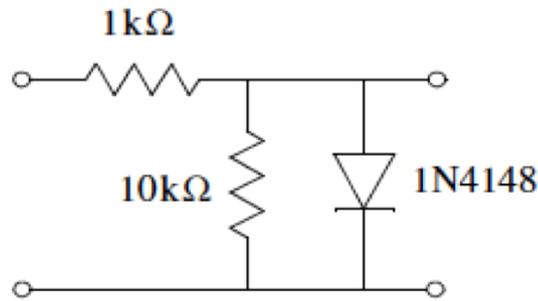


Figura 4.5: Circuito del Diodo, parte no lineal.

El conjunto de datos se dividió en dos partes: la parte de estimación y la parte de prueba. En este trabajo se utilizó el primer conjunto de datos para la estimación y el segundo conjunto para la validación.

Los datos de entrada y de salida que se se muestran en la Figura 4.6 se obtuvieron del artículo [Schoukens and Ljung, 2009].

4.2. Comparación entre metodologías (Ventajas y desventajas)

Para la pruebas en el modelo Wiener-Hammerstein se consideraron los datos de los taques acoplados, el brazo robot, la secadora de cabello y el circuito electrónico, como se muestra a continuación.

4.2.1. Dos Tanques acoplados

Para la obtención del modelo Wiener-Hammerstein se consideró para la parte no lineal un polinomio de grado dos, para los bloques lineales de entrada y salida se consideraron dos funciones de transferencia, representado de la siguiente manera:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{b_{W1}s^{n_{b1}} + b_{W0}}{a_{W2}s^{n_{a2}} + a_{W1}s^{n_{a1}} + a_{W0}}, \quad (4.1)$$

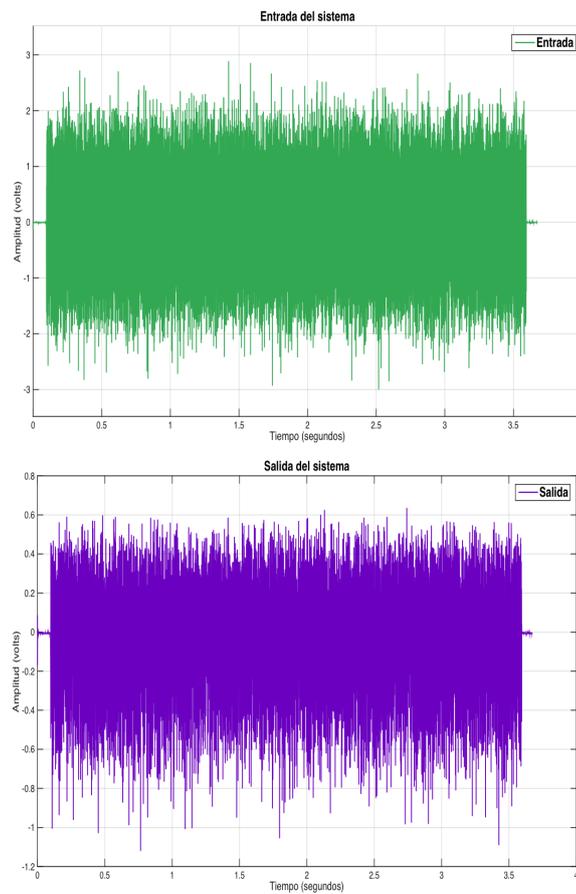


Figura 4.6: Entrada y salida del sistema Circuito electrónico

$$v(t) = \alpha_0 + \alpha_1 u(t) + \alpha_2 u^2(t), \quad (4.2)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{b_{H1}s^{n_{b1}} + b_{H0}}{a_{H2}s^{n_{a2}} + a_{H1}s^{n_{a1}} + a_{H0}}, \quad (4.3)$$

donde:

$$\begin{aligned} \theta_W &= [a_{W2}, a_{W1}, a_{W0}, b_{W1}, b_{W0}, n_{Wa2}, n_{Wa1}, n_{Wb1}]^T, \\ \theta_{NL} &= [\alpha_0, \alpha_1, \alpha_2]^T, \\ \theta_H &= [a_{H2}, a_{H1}, a_{H0}, b_{H1}, b_{H0}, n_{Ha2}, n_{Ha1}, n_{Hb1}]^T. \end{aligned} \quad (4.4)$$

Con ayuda del algoritmo de Levenberg-Marquardt proporcionado por Matlab se obtuvo el siguiente modelo No Lineal a bloques Wiener-Hammerstein de tipo fraccionario:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{0,2529s^{2,5673} - 1,016}{0,009337s^{4,0329} + 1,587s^{0,95668} + 0,2041}, \quad (4.5)$$

$$v(t) = 1,5847 - 0,53779u(t) + 0,50313u^2(t), \quad (4.6)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{4,288e - 05s^{0,26359} + 0,0001349}{1,119s^{1,4163} - 1,661s^{0,0005} + 1,986}. \quad (4.7)$$

Para poder realizar un comparativo entre la metodología fraccionaria y entera, se programó el modelo de tipo Wiener-Hammerstein de orden entero, de igual manera los parámetros del modelo fueron encontrados utilizando el algoritmo de Levenberg-Marquardt que ofrece Matlab, como se muestra a continuación:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{0,03758s + 1,776}{-0,09868s^2 + 0,9309s + 0,7992}, \quad (4.8)$$

$$v(t) = 1,2798 - 0,32324u(t) + 0,70239u^2(t), \quad (4.9)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{-2,429e - 05s + 0,0003066}{0,8705s^2 + 1,279s + 0,1873}. \quad (4.10)$$

Adicionalmente se decidió realizar una comparación con un modelo Hammerstein-Wiener resultado de la identificación obtenida con el toolbox de Matlab de orden entero.

En las Figuras 4.7 y 4.8 se puede hacer una comparación visual del modelo Wiener-Hammerstein fraccionario, Wiener-Hammerstein entero y el modelo Hammerstein-Wiener entero, visualmente el modelo fraccionario tiene una mejor aproximación que los modelos de orden entero ya que este sigue de una mejor manera a la señal original, sin embargo es necesario realizar un análisis mas profundo con los resultados obtenidos con los índices de validación.

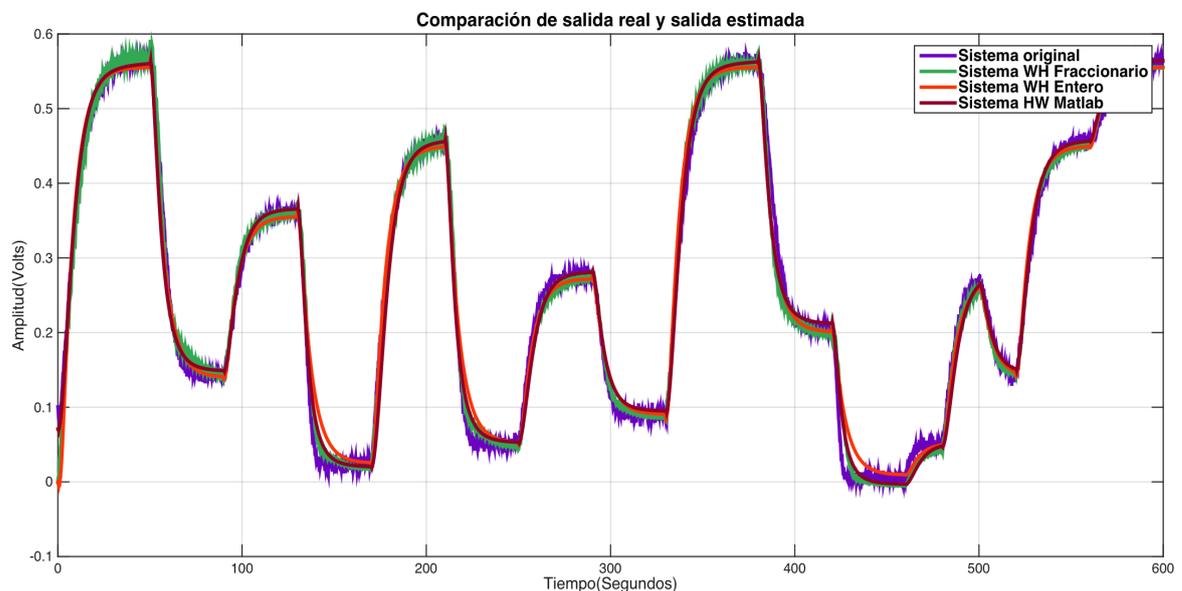


Figura 4.7: Modelo fraccionario y entero del sistema de dos tanques acoplados.

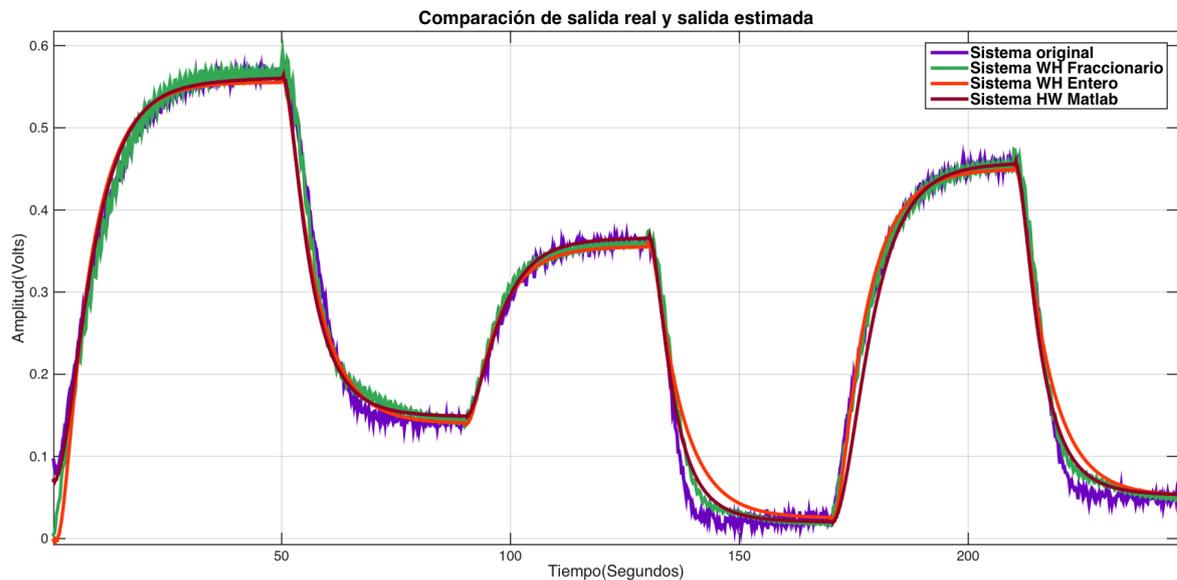


Figura 4.8: Acercamiento del modelo fraccionario y entero del sistema de dos tanques acoplados

En la tabla 4.1 podemos encontrar los resultados obtenidos por el modelo Wiener-Hammerstein fraccionario. El modelo obtenido tiene un mejor desempeño en términos del FIT y e_{rms} que las metodologías de estructuras no lineales NARX y Hammerstein-Wiener basado en redes neuronales, realizadas por [Flores, 2011] en el CENIDET, sin embargo requiere de el cálculo de mayor número de parámetros.

Donde:

- n_p es el número de parámetros.
- $Iter$ es el número de iteraciones.
- t_{mint} es el tiempo de identificación en minutos.
- μ_t es el valor medio del error de simulación.
- S_t es la desviación estándar del error.
- e_{rmst} es la raíz cuadrada del error medio al cuadrado.
- $FIT\%$ o ajuste de la señal estimada a la señal original.

Tabla 4.1: Parámetros de validación de modelos en el sistema de dos tanques acoplados.

Ref.	n_p	Iter	t_{mint}	μ_t	S_t	e_{rmst}	FIT %
W-H Fracc	19	141	11.68	0.00043	0.016	0.016	91.32
W-H Entero	13	101	5.53	-0.0009	0.020	0.020	89.08
NARX [Flores, 2011]	13	100	15	-	-	0.0293	90.72
H-W [Flores, 2011]	17	100	25	-	-	0.0361	88.55
H-W Matlab	21	20		-0.0003	0.017	0.017	90.45

4.2.2. Brazo robot

Para la obtención del modelo Wiener-Hammerstein se consideró para la parte no lineal un polinomio de grado dos, para los bloques lineales de entrada y salida se consideraron dos funciones de transferencia, representado de la siguiente manera:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{b_{W4}s^{n_{b4}} + b_{W3}s^{n_{b3}} + b_{W2}s^{n_{b2}} + b_{W1}s^{n_{b1}} + b_{W0}}{a_{W5}s^{n_{a5}} + a_{W4}s^{n_{a4}} + a_{W3}s^{n_{a3}} + a_{W2}s^{n_{a2}} + a_{W1}s^{n_{a1}} + a_{W0}}, \quad (4.11)$$

$$v(t) = \alpha_0 + \alpha_1 u(t) + \alpha_2 u^2(t), \quad (4.12)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{b_{H4}s^{n_{b4}} + b_{H3}s^{n_{b3}} + b_{H2}s^{n_{b2}} + b_{H1}s^{n_{b1}} + b_{H0}}{a_{H5}s^{n_{a5}} + a_{H4}s^{n_{a4}} + a_{H3}s^{n_{a3}} + a_{H2}s^{n_{a2}} + a_{H1}s^{n_{a1}} + a_{H0}}, \quad (4.13)$$

donde:

$$\begin{aligned} \theta_W &= [a_{W5}, a_{W4}, a_{W3}, a_{W2}, a_{W1}, a_{W0}, b_{W4}, b_{W3}, b_{W2}, b_{W1}, b_{W0} \dots \\ &\quad \dots n_{Wa5}, n_{Wa4}, n_{Wa3}, n_{Wa2}, n_{Wa1}, n_{Wb4}, n_{Wb3}, n_{Wb2}, n_{Wb1}]^T, \\ \theta_{NL} &= [\alpha_0, \alpha_1, \alpha_2]^T, \\ \theta_H &= [a_{H5}, a_{H4}, a_{H3}, a_{H2}, a_{H1}, a_{H0}, b_{H4}, b_{H3}, b_{H2}, b_{H1}, b_{H0} \dots \\ &\quad \dots n_{Ha5}, n_{Ha4}, n_{Ha3}, n_{Ha2}, n_{Ha1}, n_{Hb4}, n_{Hb3}, n_{Hb2}, n_{Hb1}]^T. \end{aligned} \quad (4.14)$$

Con ayuda del algoritmo de Levenberg-Marquardt proporcionado por Matlab mediante la introducción de la función objetivo e inicialización aleatoria de los parámetros se obtuvo el siguiente modelo no lineal a bloques Wiener-Hammerstein de tipo fraccionario:

$$G_W(s) = \frac{0,3828s^{4,98} - 0,402s^{4,80} + 0,115s^{3,80} + 0,073s^{1,89} + 0,0033}{-0,52s^{4,13} + 0,64s^{2,88} + 0,77s^{2,88} + 0,673s^{2,88} + 0,29s^{0,68} + 0,0011}, \quad (4.15)$$

$$v(t) = -0,021876 + 0,85186u^1(t) + 0,5221u^2(t), \quad (4.16)$$

$$G_H(s) = \frac{1,086s^{4,02} - 1,05s^{2,94} - 0,241s^{2,75} - 0,256s^{0,77} + 0,00066}{1,013s^{6,28} + 0,769s^{4,05} + 0,031s^{3,00} + 0,015s^{2,82} + 0,248s^{2,05} + 0,022}. \quad (4.17)$$

Para poder realizar un comparativo entre la metodología fraccionaria y entera, se programó el modelo de tipo Wiener-Hammerstein de orden entero, de igual manera los parámetros del modelo fueron encontrados utilizando el algoritmo de Levenberg-Marquardt que ofrece Matlab:

$$G_W(s) = \frac{0,2487s^4 - 0,2139s^3 + 0,3466s^2 + 0,1603s + 0,001803}{-0,9303s^5 + 1,916s^4 + 0,08211s^3 - 0,03078s^2 + 0,8702s + 0,007968}. \quad (4.18)$$

$$v(t) = -0,0047119 + 0,857694u(t) + 0,024615u^2(t), \quad (4.19)$$

$$G_H(s) = \frac{-0,199s^4 + 1,073s^3 - 1,4s^2 + 0,02678s - 0,05864}{-0,1849s^5 + 0,3973s^4 + 1,92s^3 + 0,3703s^2 + 0,1937s + 0,07631}. \quad (4.20)$$

Adicionalmente se decidió realizar una comparación con un modelo Hammerstein -Wiener resultado de la identificación obtenida con el toolbox de Matlab de orden entero.

En las Figuras 4.9 y 4.10 se puede hacer una comparación visual entre los Wiener-Hammerstein fraccionario, Wiener-Hammerstein entero y el modelo Hammerstein -Wiener entero, visualmente el modelo fraccionario tiene una mejor

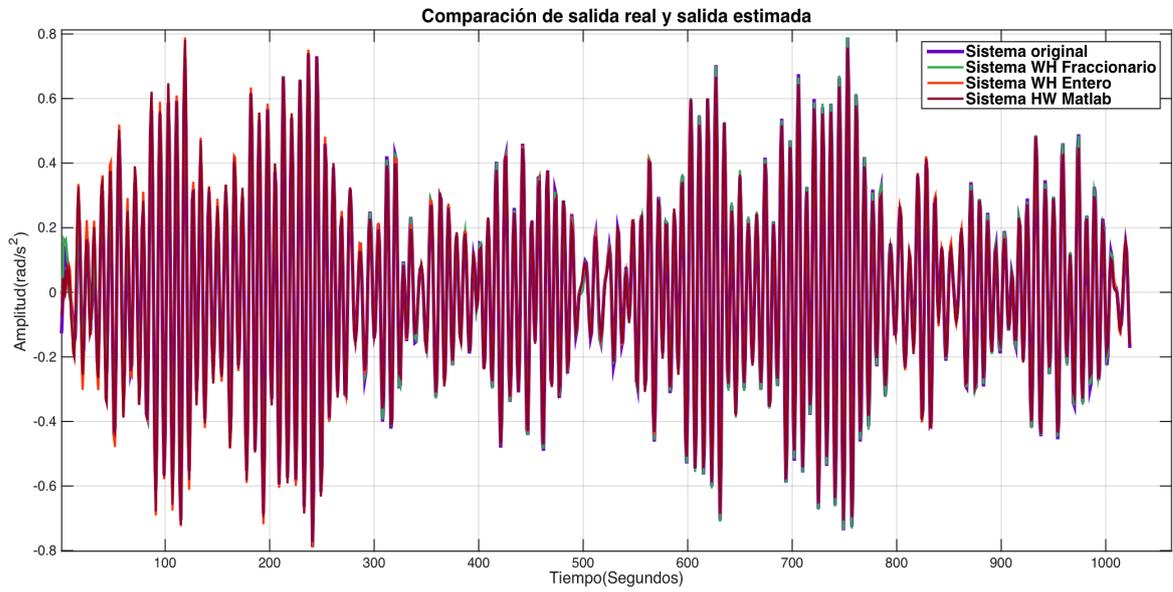


Figura 4.9: Modelo fraccionario y entero del sistema de brazo robot

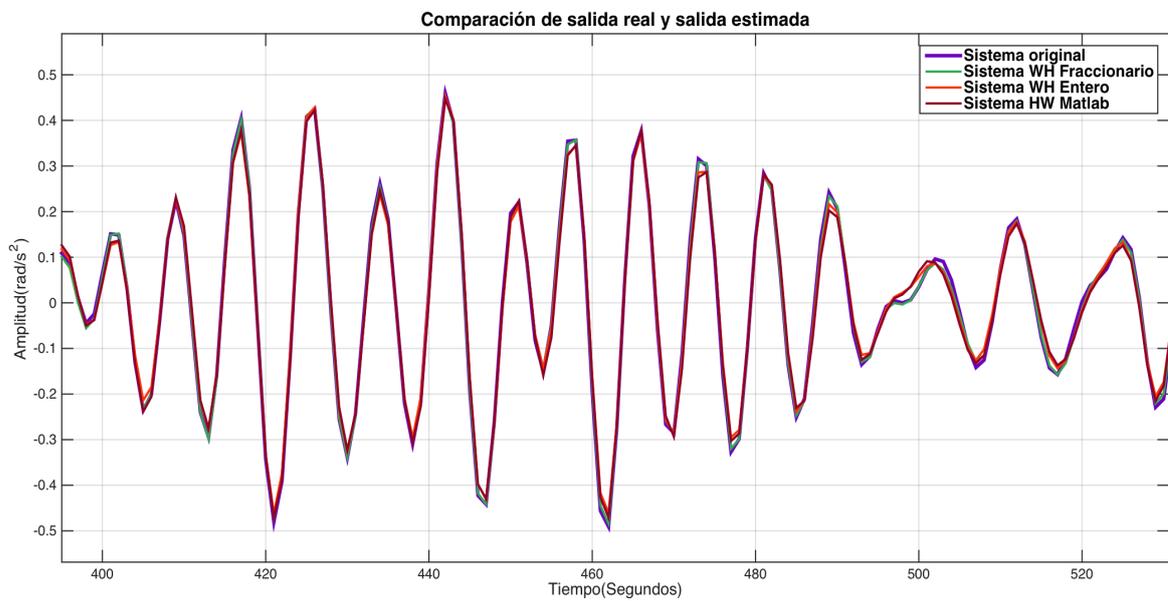


Figura 4.10: Zoom del modelo fraccionario y entero del sistema de brazo robot

aproximación que los modelos de orden entero, sin embargo es necesario realizar un análisis más profundo con los resultados obtenidos con los índices de validación.

En la Tabla 4.2 podemos encontrar los resultados obtenidos por el modelo Wiener-Hammerstein fraccionario. El modelo obtenido tiene un mejor desempeño que la metodología basada en redes neuronales en específico una arquitectura perceptrón multicapa recurrente de dos capas realizada por [Romero et al., 2015], la reportada en el artículo de [Azmi et al., 2015] y el modelo Hammerstein-Wiener de Matlab en terminos del FIT y e_{rms} , sin embargo requiere de el cálculo de un número mucho mayor de parametros, así como número de iteraciones y por lo tanto tiempo de ejecución.

Tabla 4.2: Parámetros de validación de modelos en el sistema de brazo robot.

Ref.	n_p	$Iter$	t_{mint}	μ_t 10^{-3}	S_t	e_{rmst}	FIT %
W-H Fracc	43	552	27	-0.56	0.010	0.010	96.08
W-H Entero	25	56	0.16	-2.5	0.029	0.029	89.28
NARX [Romero et al., 2015]	16	-	-	0.043	0.159	0.159	-
Algoritmo de búsqueda gravitacional [Azmi et al., 2015]	6	2000	-	-	-	-	92.90
H-W Matlab	19	20	-	-	0.024	0.024	91

Donde:

- n_p es el número de parámetros.
- $Iter$ es el número de iteraciones.
- t_{mint} es el tiempo de identificación en minutos.
- μ_t es el valor medio del error de simulación.
- S_t es la desviación estándar del error.

- e_{rmst} es la raíz cuadrada del error medio al cuadrado.
- $FIT\%$ o ajuste de la señal estimada a la señal original.

4.2.3. Circuito electrónico

Para el caso del circuito electrónico para la obtención del modelo Wiener-Hammerstein se consideró para la parte no lineal un polinomio de grado dos, para los bloques lineales de entrada y salida se consideraron dos funciones de transferencia, representado de la siguiente manera:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{b_{W2}s^{n_{b2}} + b_{W1}s^{n_{b1}} + b_{W0}}{a_{W3}s^{n_{a3}} + a_{W2}s^{n_{a2}} + a_{W1}s^{n_{a1}} + a_{W0}}, \quad (4.21)$$

$$v(t) = \alpha_0 + \alpha_1 u(t) + \alpha_2 u^2(t), \quad (4.22)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{b_{H2}s^{n_{b2}} + b_{H1}s^{n_{b1}} + b_{H0}}{a_{H3}s^{n_{a3}} + a_{H2}s^{n_{a2}} + a_{H1}s^{n_{a1}} + a_{H0}}, \quad (4.23)$$

donde:

$$\begin{aligned} \theta_W &= [a_{W3}, a_{W2}, a_{W1}, a_{W0}, b_{W2}, b_{W1}, b_{W0}, n_{Wa3}, n_{Wa2}, n_{Wa1}, n_{Wb2}, n_{Wb1}]^T, \\ \theta_{NL} &= [\alpha_0, \alpha_1, \alpha_2]^T, \\ \theta_H &= [a_{H3}, a_{H2}, a_{H1}, a_{H0}, b_{H2}, b_{H1}, b_{H0}, n_{Ha3}, n_{Ha2}, n_{Ha1}, n_{Hb2}, n_{Hb1}]^T. \end{aligned} \quad (4.24)$$

Con ayuda del algoritmo de Levenberg-Marquardt proporcionado por Matlab se obtuvo el siguiente modelo No Lineal a bloques Wiener-Hammerstein de tipo fraccionario:

$$G_W(s) = \frac{Y(s)}{U(s)} = \frac{1,615s^{0,0090587} + 1,157s^{0,0081748} + 1,7236}{1,245e - 08s^{2,153} + 0,01156s^{0,67374} + 6,357s^{4,869e-15}}, \quad (4.25)$$

$$v(t) = -0,026542 + 2,4352u(t) - 0,75657u^2(t), \quad (4.26)$$

$$G_H(s) = \frac{Y(s)}{U(s)} = \frac{1,325s^{0,047672} + 1,963s^{0,043091} + 0,3022}{2,857e - 08s^{2,2276} + 0,003015s^{0,94704} + 9,09s^{4,0018e-14}}. \quad (4.27)$$

Adicionalmente se decidió realizar una comparación con un modelo Hammersstein -Wiener resultado de la identificación obtenida con el toolbox de Matlab de orden entero.

En las Figuras 4.11 y 4.12 se puede apreciar como visualmente el modelo Wiener-Hammerstein fraccionario tiene una buena aproximación a la salida original del sistema. Sin embargo se requiere hacer un análisis en terminos de los índices de validación es importante mencionar que no se requirieron muchas épocas para poder obtener un FIT de 84% sin embargo el tiempo de ejecución fue grande ya que se procesaron 10,000 datos.

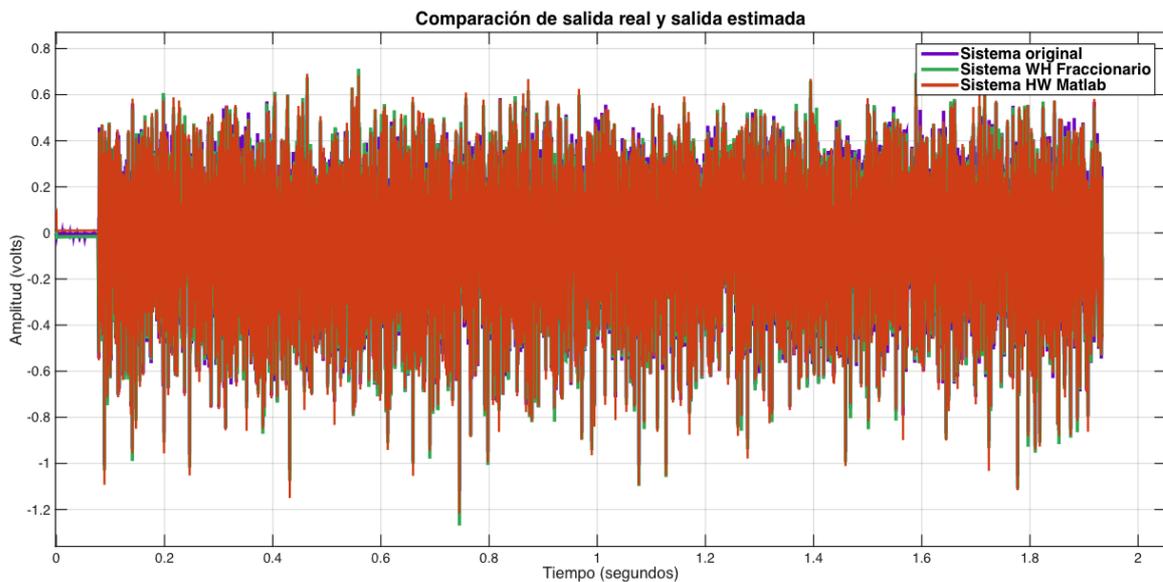


Figura 4.11: Modelo fraccionario y entero del sistema circuito electrónico.

En la Tabla 4.3 se encuentran los parámetros de validación en donde podemos observar que el modelo fraccionario tiene un mejor desempeño de acuerdo al valor de e_{rms} comparando con el modelo obtenido en la metodología basada en redes neuronales en específico una arquitectura perceptrón multicapa recurrente de

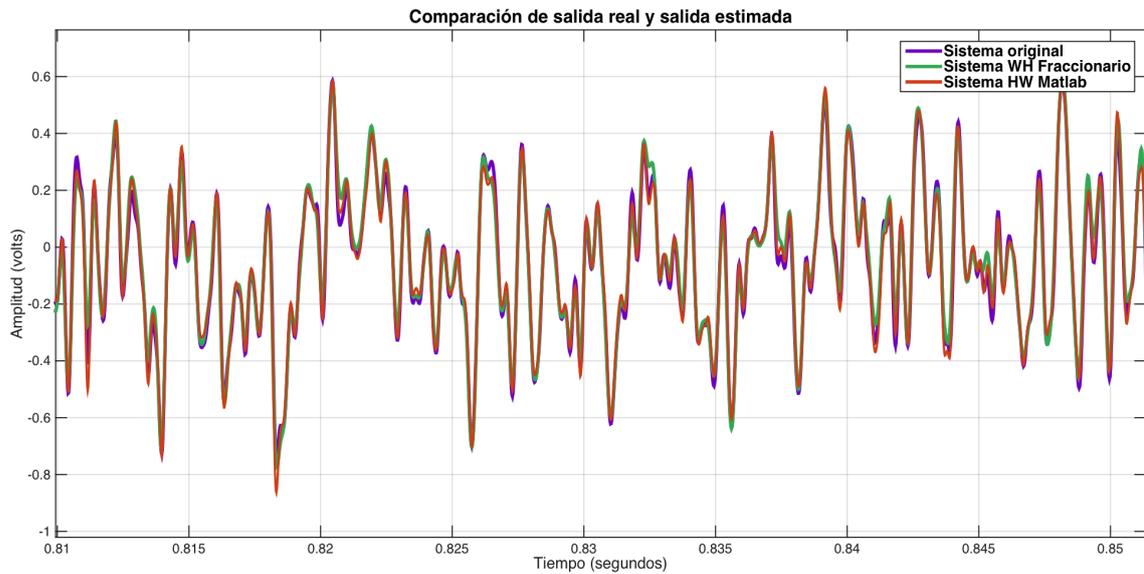


Figura 4.12: Acercamiento del modelo fraccionario y entero del sistema Circuito electrónico.

dos capas realizada por [Romero, 2013] en el CENIDET, sin embargo tiene un desempeño deficiente comparado con lo reportado por [Flores, 2011] en términos del FIT reportado, adicionalmente el modelo fraccionario necesita un número mayor de parámetros a ser calculados.

Donde:

- n_p es el número de parámetros.
- $Iter$ es el número de iteraciones.
- t_{mint} es el tiempo de identificación en minutos.
- μ_t es el valor medio del error de simulación.
- S_t es la desviación estándar del error.
- e_{rmst} es la raíz cuadrada del error medio al cuadrado.
- $FIT\%$ o ajuste de la señal estimada a la señal original.

Tabla 4.3: Parámetros de validación de modelos en el sistema del circuito electrónico.

Ref.	n_p	$Iter$	t_{mint}	μ_t	S_t	e_{rmst}	$FIT\%$
W-H Fracc	27	150	4.6	0.0031	0.036	0.037	84.68
NARX [Flores, 2011]	18	50	4.5	-	-	0.0017	99.29
Red Neuronal [Romero, 2013]	14	-	0.064	0.0047	0.0567	0.0560	
W-H Fraccio- nario Artículo [Vanbeylen, 2014]	26	-	-	-	-	0.0002	
H-W Matlab	27	20	-	-0.0021	0.027	0.027	88.3

Capítulo 5

Conclusiones

5.1. Conclusiones generales

A lo largo de este capítulo se presentaran las conclusiones generales del trabajo de tesis así como algunos trabajos que se pudieran derivar producto de la mejora del algoritmo propuesto en el capítulo 4.

En el desarrollo de la tesis se llevo a cabo un análisis de las diferentes investigaciones realizadas dentro de la identificación de modelos fraccionarios no lineales para lo cual fue necesaria la revisión y estudio del cálculo fraccionario, principalmente en el uso de funciones de transferencia fraccionarias.

Una vez que se comprendió y asimiló la teoría reciente de identificación de modelos fraccionarios se obtuvo el desarrollo matemático de las estructuras para su implementación. La implementación de los algoritmos fue realizada utilizando la derivada fraccionaria de Gröndwald-Letnikov y su Transformada de Laplace. Para la programación de esta se utilizaron las funciones en Matlab del grupo de investigación CRONE, también se realizó una interfaz gráfica de usuario (GUI por las siglas en inglés de Graphical User Interface) en donde el usuario puede realizar identificación de modelos fraccionarios de tipo: lineal, Hammerstein y Wiener-Hammerstein.

Durante este trabajo de tesis se realizó la identificación de sistemas dinámicos no lineales basados en datos de entrada-salida de un sistema real cuyo objetivo es generar un modelo matemático para que describa las propiedades del sistema, el algoritmo propuesto fue probado con los siguientes sistemas: sistema de dos tanques acoplados, un sistema de brazo robot y un circuito electrónico, de los cuales los datos de entrada-salida fueron proporcionados.

De acuerdo a lo reportado en 4.1, 4.2 y 4.3, la metodología que se llevó a cabo para la identificación de modelos Wiener-Hammerstein fraccionarios presentó resultados competitivos comparado con los resultados reportados previos en el CENIDET o en la literatura en términos del FIT o e_{rms} , sin embargo el tiempo de simulación fué mayor y es necesario un mayor número de parámetros a identificar.

El número de parámetros a identificar aumenta ya que se incluye el orden de las derivadas dentro del algoritmo de optimización, a pesar de eso los modelos fraccionarios son mas compactos que los modelos de orden entero obteniendo buenos resultados en las pruebas de validación, esto se debe en muchas ocasiones la naturaleza del sistema pertenece a la clase de modelos con dinámica fraccionaria, abriendo la posibilidad de futuras investigaciones de la interpretación física del modelo obtenido como lo explica [Podlubny, 2002]. Adicionalmente el modelo fraccionario permite un mejor ajuste gracias a los grados de libertad que se tiene al poder modificar el orden de la derivada.

Las aportaciones que se dejan con la realización de esta tesis son:

- Un algoritmo alterno a lo reportado en la literatura que propociona una fiel representación matemática de un sistema físico de acuerdo a los índices de validación, mediante el uso de datos experimentales.
- Un toolbox con interfaz gráfica de usuario (GUI), accesible para alumnos y catedráticos que estén interesados en trabajar con modelos fraccionarios, la interfaz permite obtener modelos fraccionarios del tipo lineales, Hammerstein y Wiener Hammerstein a partir de datos de entrada y salida.
- Introducción del orden de la derivada en la optimización en el algoritmo de Levenberg-Maquardt, lo cual permite la identificación del orden fraccionario de la derivada dando mayor grado de libertad y por lo tanto un mejor ajuste en el modelo resultante
- La introducción en el CENIDET de una herramienta matemática como lo es el cálculo fraccionario aplicado en identificación de sistemas.

5.2. Trabajos futuros

El cálculo fraccionario es una herramienta matemática que se encuentra en constante investigación e innovación por lo que durante la realización de esta tesis se encontraron temas relacionados bastante interesantes que se pueden abordar dándole un enfoque diferente al algoritmo propuesto. Tomando como antecedente este trabajo de tesis se proponen los siguientes temas:

- Desarrollar o proponer un método para la inicialización de los parámetros θ en el algoritmo de Levenberg-Marquardt.
- Desarrollar un algoritmo de optimización fraccionario e implementarlo en el algoritmo desarrollado en esta tesis.
- Implementar diferentes funciones en el bloque No Lineal por ejemplo: saturación, lineal a bloques, función sigmoide, entre otras.
- Contemplar sistemas fraccionarios que cuenten con múltiples entradas y una salida (MISO), sistemas fraccionarios de una entrada y múltiples salidas (SIMO) y a sistemas fraccionarios de múltiples entradas y salidas (MIMO.)
- Diseño de controladores fraccionarios y controladores para modelos fraccionarios.
- Realizar la identificación de modelos no lineales utilizando cálculo fraccionario en tiempo discreto.

Bibliografía

- [Aoun et al., 2002] Aoun, M., Malti, R., Cois, O., and Oustaloup, A. (2002). System identification using fractional hammerstein models. In *Proc. of the 15th IFAC World Congress*.
- [Atanackovic et al., 2014] Atanackovic, T. M., Pilipovic, S., Stankovic, B., and Zorica, D. (2014). *Fractional calculus with applications in mechanics: wave propagation, impact and variational principles*. John Wiley & Sons.
- [Atangana and Baleanu, 2016] Atangana, A. and Baleanu, D. (2016). New fractional derivatives with nonlocal and non-singular kernel: Theory and application to heat transfer model. *Thermal Science*, 20(2):763–769.
- [Azmi et al., 2015] Azmi, K. Z. M., Pebrianti, D., Ibrahim, Z., Sudin, S., and Nawawi, S. W. (2015). Simultaneous computation of model order and parameter estimation for system identification based on gravitational search algorithm. In *Intelligent Systems, Modelling and Simulation (ISMS), 2015 6th International Conference on*, pages 135–140. IEEE.
- [Calik, 2015] Calik, A. (2015). Investigation of electrical RC circuit within the framework of fractional calculus. *Rev. Mex. Fis*, 61:58–63.
- [Caputo and Fabrizio, 2015] Caputo, M. and Fabrizio, M. (2015). A new definition of fractional derivative without singular kernel. *Progr. Fract. Differ. Appl*, 1(2):1–13.
- [Cois et al., 2002] Cois, O., Lanusse, P., Melchior, P., Dancla, F., and Oustaloup, A. (2002). Fractional systems toolbox for matlab: applications in system identification and crone csd. In *41st IEEE conference on Decision and Control*.
- [De Oliveira and Machado, 2014] De Oliveira, E. C. and Machado, J. A. T. (2014). A review of definitions for fractional derivatives and integral. *Mathematical Problems in Engineering*.

- [Duarte and Costa, 2004] Duarte, V. and Costa, J. (2004). Ninteger: a non-integer control toolbox for matlab. *Fractional derivatives and applications. Bordeaux: IFAC*.
- [Flores, 2011] Flores, U. (2011). Identificación de sistemas no lineales mediante las estructuras narx y hammerstein-wiener”. Master’s thesis, CENIDET.
- [Gómez-Aguilar, 2012] Gómez-Aguilar, J. F. (2012). *Aplicación del Cálculo Fraccionario al Modelado de Sistemas Biofísicos, Bioeléctricos y Mecánicos*. PhD thesis, Universidad de Guanajuato.
- [Gómez-Aguilar et al., 2016a] Gómez-Aguilar, J. F., Escalante-Martínez, J., Calderón-Ramón, C., Morales-Mendoza, L., Benavidez-Cruz, M., and Gonzalez-Lee, M. (2016a). Equivalent circuits applied in electrochemical impedance spectroscopy and fractional derivatives with and without singular kernel. *Advances in Mathematical Physics*, 2016.
- [Gómez-Aguilar et al., 2016b] Gómez-Aguilar, J. F., López-López, M. G., Alvarado-Martínez, V. M., Reyes-Reyes, J., and Adam-Medina, M. (2016b). Modeling diffusive transport with a fractional derivative without singular kernel. *Physica A: Statistical Mechanics and its Applications*, 447:467–481.
- [Gómez-Aguilar et al., 2016c] Gómez-Aguilar, J. F., Miranda-Hernández, M., López-López, M. G., Alvarado-Martínez, V. M., and Baleanu, D. (2016c). Modeling and simulation of the fractional space-time diffusion equation. *Communications in Nonlinear Science and Numerical Simulation*, 30(1):115–127.
- [Gómez-Aguilar et al., 2015] Gómez-Aguilar, J. F., Morales, L., González, M., Alvarado-Martínez, V. M., and López-López, M. G. (2015). Fractional thermal diffusion and the heat equation. *Open Physics*, 13(1).
- [Gómez-Aguilar et al., 2014] Gómez-Aguilar, J. F., Razo-Hernández, J. R., Rosales-García, J., and Guía-Calderón, M. (2014). Circuitos eléctricos RC y LC fraccionarios. *Ingeniería, investigación y tecnología*, 15(2):311–319.
- [Gómez-Aguilar et al., 2013] Gómez-Aguilar, J. F., Rosales, J., and Guía, M. (2013). RLC electrical circuit of non-integer order. *Central European Journal of Physics*, 11(10):1361–1365.
- [Hernandez et al., 2014] Hernandez, O., Ortiz, P., and Herrera, J. (2014). Cement rotary kiln model using fractional identification. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, 12(2):87–92.

- [leuven, 2008] leuven, K. (2008). Stadius center for dynamical systems, signal processing and data analytics.
- [Levenberg, 1944] Levenberg, K. (1944). A method for the solution of certain nonlinear problems in least squares.
- [Li et al., 2014] Li, Y., Zhai, L., Chen, Y., and Ahn, H.-S. (2014). Fractional-order iterative learning control and identification for fractional-order hammerstein system. pages 840–845.
- [Ljung, 1998] Ljung, L. (1998). *System identification*. Springer.
- [Ljung, 2007] Ljung, L. (2007). Identification of nonlinear systems. *9th International Conference on Control, Automation, Robotics and Vision, ICARCV*.
- [Malti et al., 2006] Malti, R., Aoun, M., Sabatier, J., and Oustaloup, A. (2006). Tutorial on system identification using fractional differentiation models. In *14th IFAC symposium on system identification*, pages 606–611.
- [Malti et al., 2007] Malti, R., Victor, S., Nicolas, O., and Oustaloup, A. (2007). System identification using fractional models: state of the art. In *ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 295–304. American Society of Mechanical Engineers.
- [Marquardt, 1963] Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441.
- [Nelles, 2002] Nelles, O. (2002). *Nonlinear system identification*. IOP Publishing.
- [Podlubny, 1998] Podlubny, I. (1998). *Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications*. Academic press.
- [Podlubny, 2002] Podlubny, I. (2002). Geometric and physical interpretation of fractional integration and fractional differentiation. *Applied Analysis*, 5(4):367–386.
- [Romero, 2013] Romero, H. (2013). *Identificación de Sistemas Utilizando Redes Neuronales: Un Enfoque Basado en el Balance Entre Sencillez, Precisión y Costo Computacional*. PhD thesis, CENIDET.

- [Romero et al., 2015] Romero, H. M., Carmona, J.-C., Reyes-Reyes, J., Alvarado-Martínez, V. M., and Mantilla, J. (2015). Computational cost improvement of neural network models in black box nonlinear system identification. *Neurocomputing*, 166:96–108.
- [Sánchez, 2011] Sánchez, J. M. (2011). Génesis y desarrollo del cálculo fraccional. *Pensamiento Matemático*, (1):4.
- [Schoukens and Ljung, 2009] Schoukens, J. and Ljung, L. (2009). Wiener-hammerstein benchmark. *15th IFAC Symposium on System Identification*.
- [Stanislawski et al., 2014] Stanislawski, R., Latawiec, K. J., Galek, M., and Lukaszyn, M. (2014). Modeling and identification of a fractional-order discrete-time siso laguerre-wiener system. In *Methods and Models in Automation and Robotics (MMAR), 19th International Conference*, pages 165–168. IEEE.
- [Tepljakov et al., 2011] Tepljakov, A., Petlenkov, E., and Belikov, J. (2011). Fomcon: Fractional-order modeling and control toolbox for matlab. In *Proc. 18th Int Mixed Design of Integrated Circuits and Systems (MIXDES) Conference*, pages 684–689.
- [Toledo-Hernandez et al., 2014] Toledo-Hernandez, R., Rico-Ramirez, V., Iglesias-Silva, G. A., and Diwekar, U. M. (2014). A fractional calculus approach to the dynamic optimization of biological reactive systems. part I: Fractional models for biological reactions. *Chemical Engineering Science*, 117:217–228.
- [Ullah et al., 2014] Ullah, N., Khan, W., and Wang, S. (2014). High performance direct torque control of electrical aerodynamics load simulator using fractional calculus. *Acta Polytechnica Hungarica*, 11(10):59–78.
- [Vanbeylen, 2014] Vanbeylen, L. (2014). A fractional approach to identify wiener-hammerstein systems. *Automatica*, 50(3):903–909.
- [Zhao et al., 2014] Zhao, Y., Li, Y., and Chen, Y. (2014). Complete parametric identification of fractional order hammerstein systems. In *Fractional Differentiation and Its Applications (ICFDA), International Conference on*, pages 1–6. IEEE.

Apéndice A

Funciones de Matlab utilizadas y Manual de usuario de las funciones

A.1. Funciones de Matlab utilizadas

lsim(sys, u)

Simula la respuesta de un sistema sys ante una entrada u.

lsqnonlin(fun,x0)

Resuelve problemas de ajuste de curva mediante mínimos cuadrados no lineales, utilizando el algoritmo de Levenberg-Marquardt, comienza en el punto x_0 y encuentra un mínimo de la suma de los cuadrados de las funciones descritas en fun.

nlhw([nb n f nk])

Estima el modelo Hammerstein-Wiener del sistema.

A.2. Funciones de CRONE utilizadas

frac_poly_exp(coef, order)

Del inglés *explicitfractionalpolynomials*, (polinomios fraccionarios explícitos), donde coef se refiere a los coeficientes de los polinomios y order al orden de cada coeficiente. Por ejemplo:

```
frac_poly_exp([2, 1, 3], [1.2, 0.5, 0]),
```

Da como resultado:

$$2s^{1,2} + s^{0,5} + 3.$$

frac_tf(num,den)

Proporciona una función de transferencia fraccionaria, es una clase que permite la representación de sistemas fraccionarios.

Se construye con dos argumentos, numerador y denominador de la siguiente manera:

- `frac_tf (num:frac_poly_exp, den:frac_poly_exp)`.

Apéndice B

Interfaz gráfica

Interfaz gráfica permite al usuario realizar la identificación de un modelo fraccionario, lineal y no lineal (Hammerstein y Wiener-Hammerstein). Primero se debe ejecutar el programa de nombre Interfaz, con esta operación se abre la ventana que se muestra en la Figura B.1, en estas pantalla el usuario tiene la oportunidad de seleccionar el tipo de modelo que desea trabajar.

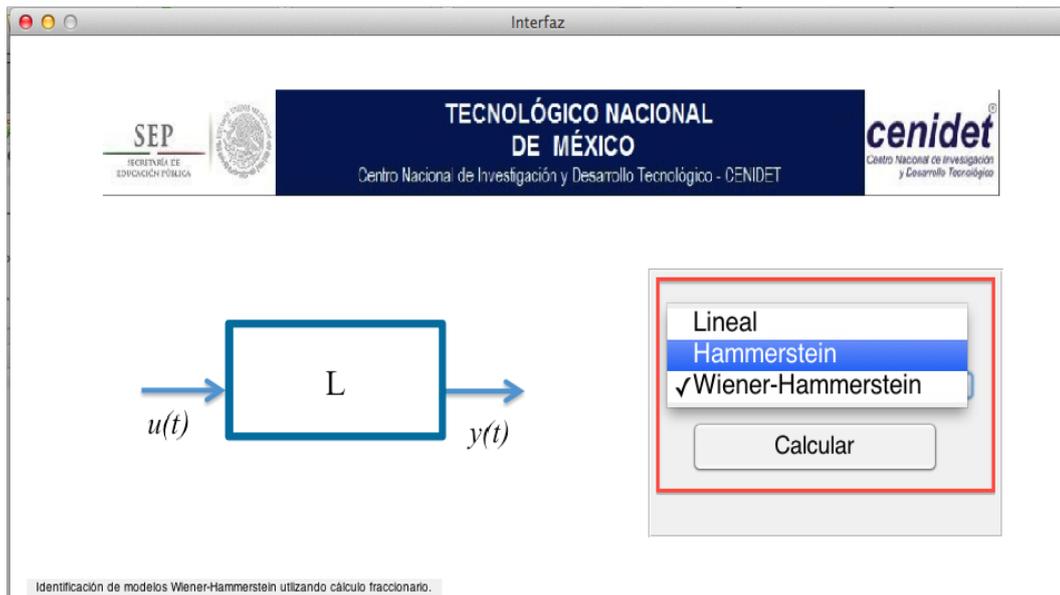


Figura B.1: Interfaz principal

Modelo lineal fraccionario

Al oprimir el botón **Modelo**, se desplegará una imagen del diagrama de bloques del modelo lineal con la estructura identificada resultante, como se muestra

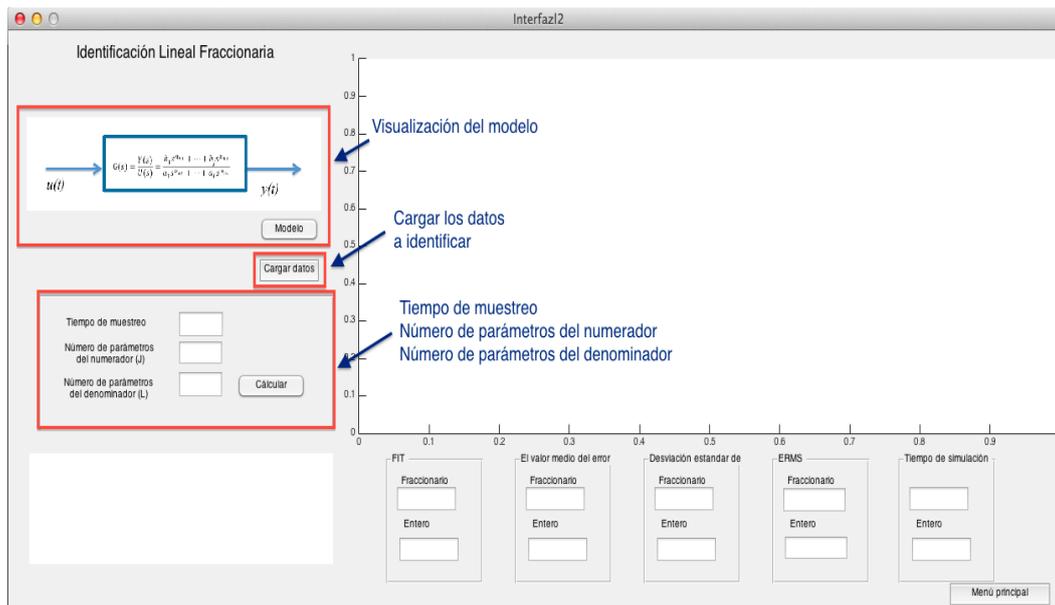


Figura B.2: Pantalla del modelo lineal

en la Figura B.3.

En todos los casos antes de comenzar la identificación el usuario debe de cargar los datos, dando click en el botón **Cargar datos** como se muestra en la Figura B.2.

Posteriormente se mostrará la pantalla de la Figura B.4 en donde el usuario podrá seleccionar los datos que desea identificar, los datos deben de ser un archivo .mat y deben de contener el vector de entrada con el nombre de u y salida y .

Finalmente el usuario podrá ver el modelo lineal fraccionario, ver Figura B.5, así como la gráfica comparativa y los índices de desempeño del modelo obtenido.

Modelo Hammerstein fraccionario

El usuario debe de especificar el tiempo de muestreo y el número de parámetros del numerador y denominador del modelo lineal. Para el caso del modelo no lineal (NL) se debe introducir número 1 si se desea trabajar con polinomios y el número 2 para una función de tipo seno, ver Figura B.6. El campo "grado del polinomio" se utiliza para especificar el grado del polinomio.

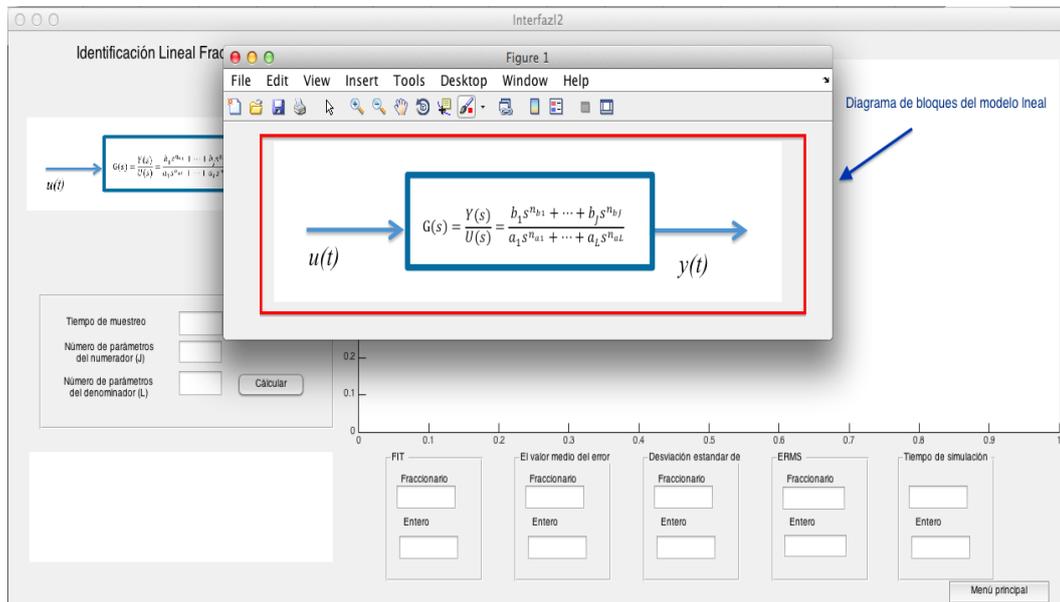


Figura B.3: Diagrama de bloques del modelo lineal fraccionario

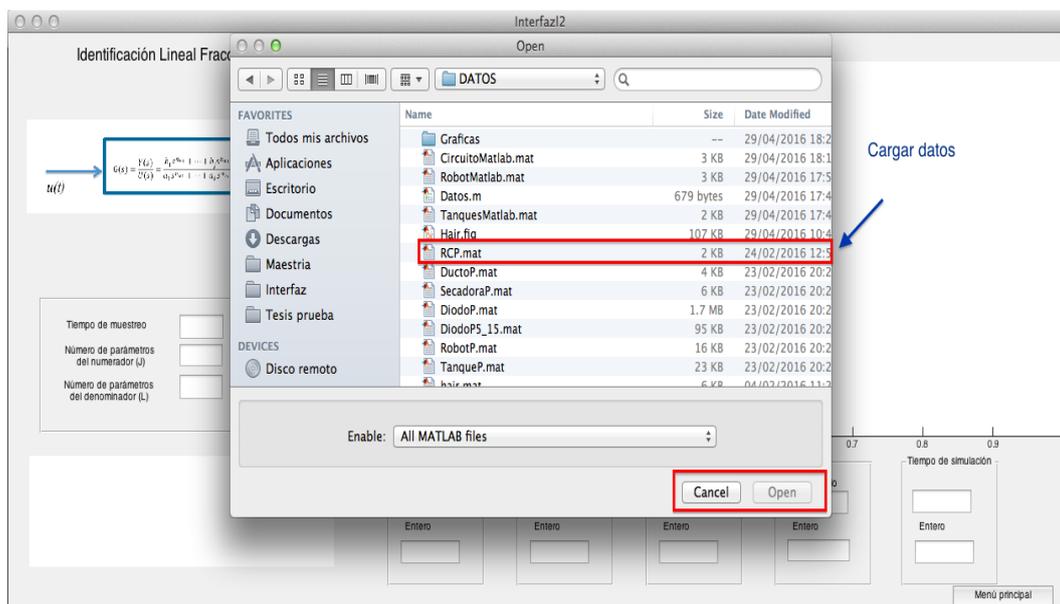


Figura B.4: Pantalla para cargar datos

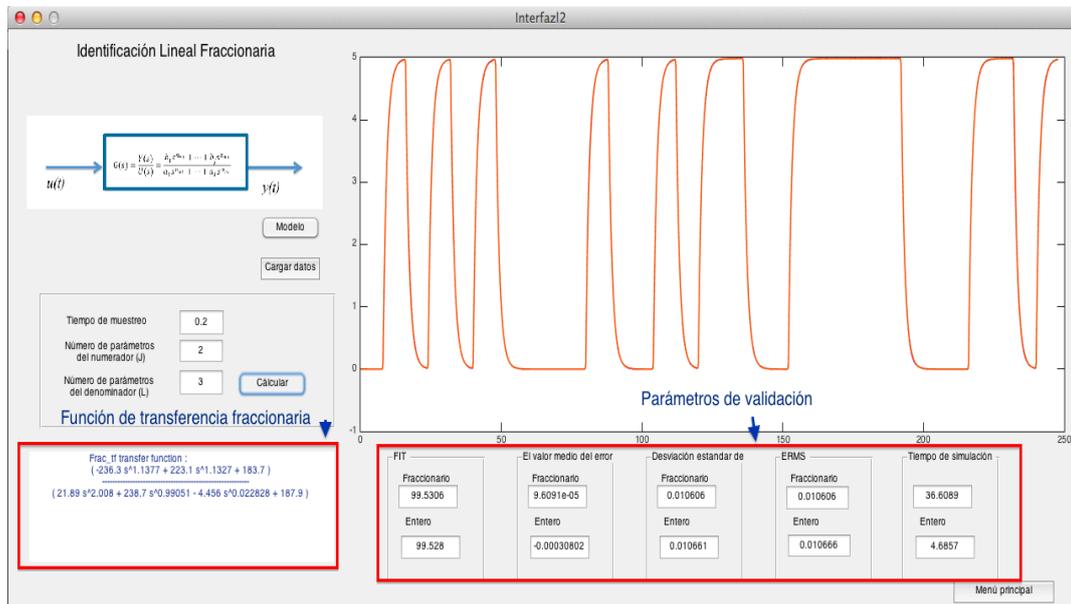


Figura B.5: Pantalla del modelo lineal resultante

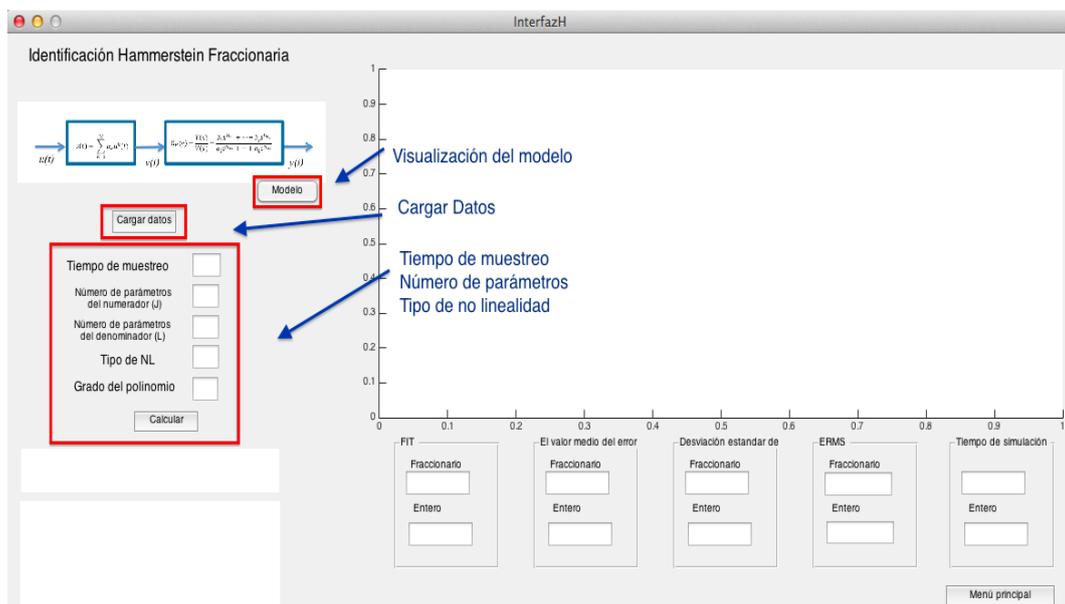


Figura B.6: Pantalla del modelo Hammerstein

Al oprimir el botón **Modelo**, se desplegará una imagen del diagrama de bloques del modelo Hammerstein con la estructura identificada resultante, como se muestra en la Figura B.7.

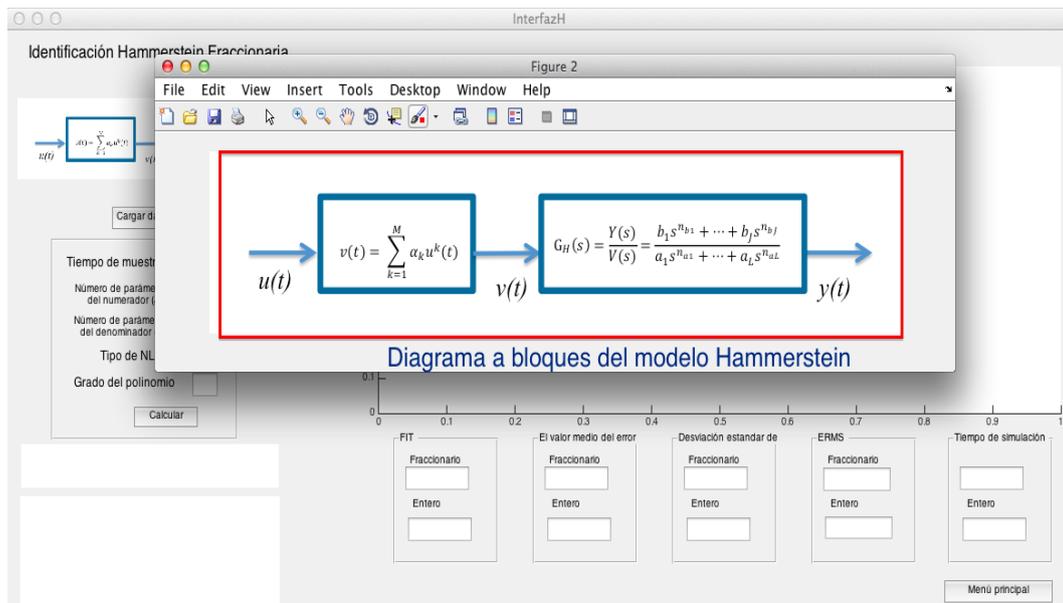


Figura B.7: Diagrama de bloques del modelo Hammerstein fraccionario

Finalmente el usuario podrá ver el modelo Hammerstein fraccionario, ver Figura B.8, así como la gráfica comparativa y los índices de desempeño del modelo obtenido.

Modelo Wiener-Hammerstein fraccionario

El usuario debe de especificar el tiempo de muestreo y el número de parámetros del numerador y denominador para los dos modelos lineales, es decir el bloque de entrada y de salida del sistema Wiener-Hammerstein, ver Figura B.9. Para el caso del modelo no lineal (NL) se debe introducir número 1 si se desea trabajar con polinomios y el número 2 para una función de tipo seno. El campo grado del polinomio se utiliza para especificar el grado del polinomio.

Al oprimir el botón **Modelo**, se desplegará una imagen del diagrama de bloques del modelo Wiener-Hammerstein con la estructura identificada resultante, como se muestra en la Figura B.10.

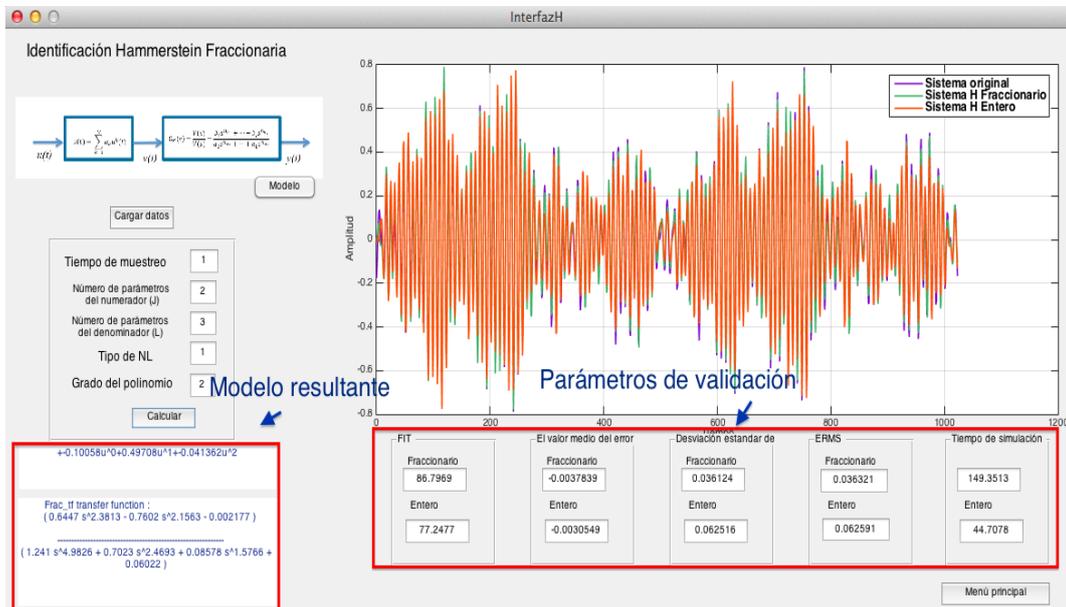


Figura B.8: Pantalla del modelo Hammerstein resultante

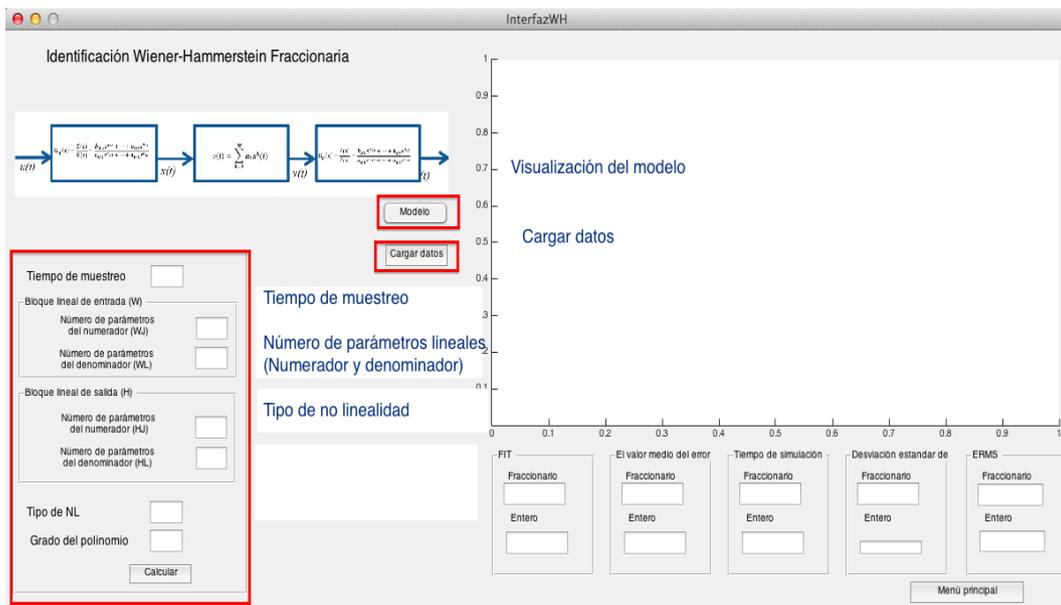


Figura B.9: Pantalla del modelo Wiener-Hammerstein

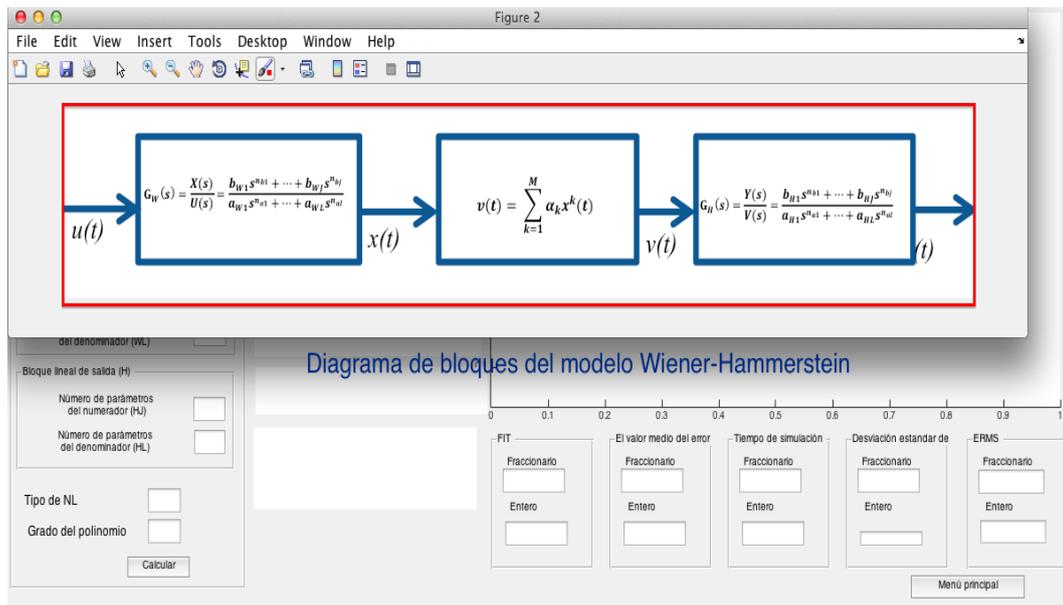


Figura B.10: Diagrama de bloques del modelo Wiener-Hammerstein fraccionario

Finalmente el usuario podrá ver el modelo Wiener-Hammerstein fraccionario, ver Figura B.11, así como la gráfica comparativa y los índices de desempeño del modelo obtenido.

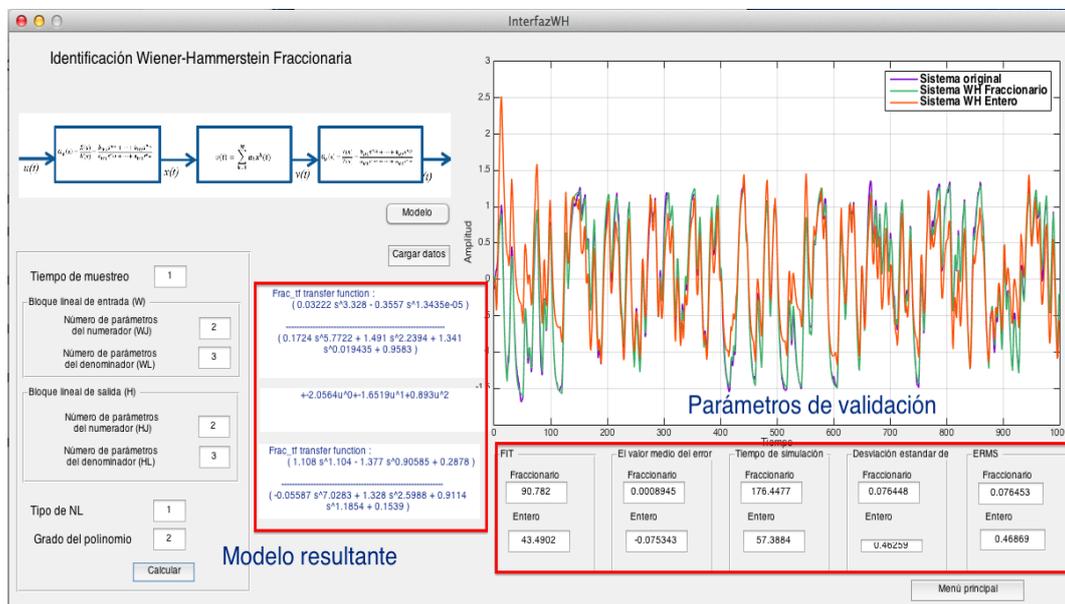


Figura B.11: Pantalla del modelo Wiener-Hammerstein resultante

Apéndice C

Código para la Identificación de Modelos

Wiener-Hammerstein de orden

Fraccionario

```

%% Wiener-Hammerstein con FT fraccionaria

function [Gw,Gh,NoLineal,yWH,x,Gw2,Gh2,NoLinealEntero,yWH2,x2,x0,t1,t2] =
WHInterfaz(u,y,ts,ncw,npw,nch,nph,NL,gp)

% u: Entrada
% y: Salida
% ts: Tiempo de muestreo
%
% ncw: Número de ceros Wiener
% npw: Número de polos Wiener
% nch: Número de ceros Hammerstein
% nph: Número de polos Hammerstein
%
% NL: No lineal
% 1: polinomio
% gp: Grado del polinomio
% 2: Seno

%% %%%%%%%%%%% Wiener-Hammerstein Fraccionario%%%%%%%%%%
format shortg
t=0:ts:(length(u))*ts-ts;

switch NL
    case 1
        gp=gp;
    case 2
        gp=2;
end

x0= rand(1,2*npw+2*ncw+2+gp+1+2*nph+2*nch+2); %Condiciones iniciales

%%Inicialización de exponentes
if ncw ~=0
k=ncw;
for i=1:ncw
    xnbw(i)=k;
    k=k-1;
end

k=npw;
for i=1:npw
    xnaw(i)=k;
    k=k-1;
end

x0(ncw+npw+3:2*ncw+npw+2)= sqrt(xnbw);
x0(2*ncw+npw+3:2*ncw+2*npw+2)= sqrt(xnaw);

else
    k=npw;
    for i=1:npw
        xnaw(i)=k;
    end
end

```

```
        k=k-1;
    end

    x0(2*ncw+npw+3:2*ncw+2*npw+2)= sqrt(xnaw);

    end

    if nch ~=0
        k=nch;
        for i=1:nch
            xnbh(i)=k;
            k=k-1;
        end

        k=nph;
        for i=1:nph
            xnah(i)=k;
            k=k-1;
        end

        x0((2*ncw+2*npw+5)+nch+nph:(2*ncw+2*npw+5)+2*nch+nph-1)= sqrt(xnbh);
        x0((2*ncw+2*npw+5)+2*nch+nph:(2*ncw+2*npw+5)+2*nch+2*nph-1)= sqrt(xnah);

    else

        k=nph;
        for i=1:nph
            xnah(i)=k;
            k=k-1;
        end

        x0((2*ncw+2*npw+5)+2*nch+nph:(2*ncw+2*npw+5)+2*nch+2*nph-1)= sqrt(xnah);

    end
```

```

%Levenberg-Marquardt WH Fraccionario
tic
Empieza = strcat('Comienza identificación Fraccionaria')

options = optimoptions('lsqnonlin','Display','iter');
options.Algorithm = 'levenberg-marquardt';
options.MaxIter = 100;% 594; %Number of iterations exceeded
options.MaxIter
options.MaxFunEvals=50000; % or number of function evaluations exceeded
options.MaxFunEvals.
[x,resnorm,residual,exitflag,output] = lsqnonlin(@(x)
funcioncostoWHInterfaz(x,y,u,t,npw,ncw,nph,nch,NL,gp),x0,[],[],options);
t1=toc;

%%% Modelo Final
%%% Modelo Wiener Lineal
bw = x(1:ncw+1);
nbw = [x(ncw+npw+3:ncw+npw+ncw+2).^2 0];
aw = x(ncw+2:ncw+npw+2);
naw = [x(ncw+npw+3+ncw:ncw+npw+ncw+npw+2).^2 0];

Gw=frac_tf(frac_poly_exp(bw,nbw),frac_poly_exp(aw,naw));
y_W = lsim(Gw,u,t);

%%%Modelo No lineal
v=0;
NoLineal='';
switch NL
    case 1

        for i=0:1:gp
            v=v+x((2*ncw+2*npw)+4+(2*nch+2*nph)+1+i).*y_W.^i;
            NoLineal = strcat(NoLineal
, '+' ,num2str(x((2*ncw+2*npw)+4+(2*nch+2*nph)+1+i)), 'u^' , num2str(i));
        end
    case 2
        v=
x((2*ncw+2*npw+5)+nch+nph+nch+nph).*sin(x((2*ncw+2*npw+5)+nch+nph+nch+nph
+1).*y_W)+x((2*ncw+2*npw+5)+nch+nph+nch+nph+2);

        a0=num2str(x((2*ncw+2*npw+5)+nch+nph+nch+nph));
        a1=num2str(x((2*ncw+2*npw+5)+nch+nph+nch+nph+1));
        a2=num2str(x((2*ncw+2*npw+5)+nch+nph+nch+nph+2));
        NoLineal = strcat('v=',a0,'*sin(', a1,'u)+',a2); %Resultado No Lineal
end

%%% Modelo Hammerstein Lineal
bh = x((2*ncw+2*npw+5)+1-3:(2*ncw+2*npw+5)+nch-2);
nbh = [x((2*ncw+2*npw+5)+nch+nph:(2*ncw+2*npw+5)+nch+nph+nch-1).^2 0];
ah = x((2*ncw+2*npw+5)+nch-1:(2*ncw+2*npw+5)+nch+nph-1);
nah = [x((2*ncw+2*npw+5)+nch+nph+nch:(2*ncw+2*npw+5)+nch+nph+nch+nph-
1).^2 0];

Gh=frac_tf(frac_poly_exp(bh,nbh),frac_poly_exp(ah,nah));
yWH = lsim(Gh,v,t);

```

```

%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Wiener-Hammerstein Entero%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Empieza = strcat('Comienza identificación de orden entero')
tic

x02= [x0(1:ncw+1),x0(ncw+2:ncw+npw+2),...
      x0((2*ncw+2*npw+5)+1-3:(2*ncw+2*npw+5)+nch-
2),x0((2*ncw+2*npw+5)+nch-1:(2*ncw+2*npw+5)+nch+npw-1)...

x0((2*ncw+2*npw)+4+(2*nch+2*nph)+1:(2*ncw+2*npw)+4+(2*nch+2*nph)+1+gp)];

%%Inicialización
k=ncw;
for i=1:ncw+1
    nbw(i)=k;
    k=k-1;
end

k=npw;
for i=1:npw+1
    naw(i)=k;
    k=k-1;
end

k=nch;
for i=1:nch+1
    nbh(i)=k;
    k=k-1;
end

k=nph;
for i=1:nph+1
    nah(i)=k;
    k=k-1;
end

%Levenberg-Marquardt Entero
tic
options = optimoptions('lsqnonlin','Display','iter');
options.Algorithm = 'levenberg-marquardt';
options.TolFun=1.5e-10000; %Magnitude of search direction was smaller
than the specified tolerance.
options.TolX=1e-10000; % Change in x was less than the specified
tolerance.
options.MaxIter =50; %Number of iterations exceeded options.MaxIter
options.MaxFunEvals=1500; % or number of function evaluations exceeded
options.MaxFunEvals.
[x2,resnorm2,residual2,exitflag2,output2] = lsqnonlin(@(x2)
funcioncostoWHEnteroInterfaz(x2,y,u,t,npw,ncw,nph,nch,nbw,naw,nbh,nah,NL,
gp),x02,[],[],options);
t2=toc;

%% Modelo

```

```

%%% Modelo Wiener Lineal
bw = x2(1:ncw+1);
aw = x2(ncw+2:ncw+npw+2);
Gw2=frac_tf(frac_poly_exp(bw,nbw),frac_poly_exp(aw,naw));
y_We = lsim(Gw2,u,t);

%%%Modelo No lineal
v2=0;
NoLinealEntero='';
switch NL

    case 1
        for i=0:1:gp
            v2=v2+x2((ncw+npw)+4+nch+nph+1+i).*y_We.^i;
            NoLinealEntero = strcat(NoLinealEntero
, '+' ,num2str(x2((ncw+npw)+4+(nch+nph)+1+i)), 'u^', num2str(i));

        end

    case 2
        v2= x2((ncw+npw+5)+nch+nph+2-3+1).*sin(x2((ncw+npw+5)+nch+nph+2-
3+2)).*y_We)+x2((ncw+npw+5)+nch+nph+2-3+3);
        a0=num2str(x2((ncw+npw+5)+nch+nph+2-3+1));
        a1=num2str(x2((ncw+npw+5)+nch+nph+2-3+2));
        a2=num2str(x2((ncw+npw+5)+nch+nph+2-3+3));
        NoLinealEntero = strcat('v=',a0,'*sin(', a1,'u)+' ,a2);
%Resultado No Lineal

end

%%% Modelo Hammerstein Lineal
bh = x2((ncw+npw+5)+1-3:(ncw+npw+5)+nch+1-3);
ah = x2((ncw+npw+5)+nch+2-3:(ncw+npw+5)+nch+nph+2-3);

Gh2=frac_tf(frac_poly_exp(bh,nbh),frac_poly_exp(ah,nah));
yWH2 = lsim(Gh2,v2,t);

```

```

function z = funcioncostoWHInterfaz(x,y,u,t,npw,ncw,nph,nch,NL,gp)

bw = x(1:ncw+1);
nbw = [x(ncw+npw+3:ncw+npw+3+ncw-1).^2 0];
aw = x(ncw+2:ncw+npw+2);
naw = [x(ncw+npw+3+ncw:ncw+npw+3+ncw+npw-1).^2 0];

Gw=frac_tf(frac_poly_exp(bw,nbw),frac_poly_exp(aw,naw));

y_W = lsim(Gw,u,t);
v=0;
switch NL
    case 1
        for i=0:1:gp
            v=v+x((2*ncw+2*npw)+4+(2*nch+2*nph)+1+i).*y_W.^i;
        end
    case 2
v=
x((2*ncw+2*npw+5)+2*nch+2*nph).*sin(x((2*ncw+2*npw+5)+2*nch+2*nph+1).*y_W
)+x((2*ncw+2*npw+5)+2*nch+2*nph+2);
end

bh = x((2*ncw+2*npw+5)+1-3:(2*ncw+2*npw+5)+nch+1-3);
nbh = [x((2*ncw+2*npw+5)+nch+nph+3-3:(2*ncw+2*npw+5)+nch+nph+3+nch-1-
3).^2 0];
ah = x((2*ncw+2*npw+5)+nch+2-3:(2*ncw+2*npw+5)+nch+nph+2-3);
nah = [x((2*ncw+2*npw+5)+nch+nph+3+nch-1+1-
3:(2*ncw+2*npw+5)+nch+nph+3+nch+nph-1+1-1-3).^2 0];
Gh=frac_tf(frac_poly_exp(bh,nbh),frac_poly_exp(ah,nah));

y_id = lsim(Gh,v,t);
z = y - y_id;

```