

**IDENTIFICACIÓN DE SISTEMAS DINÁMICOS
VARIANTES E INVARIANTES EN EL TIEMPO
BASADA EN LAS CARACTERÍSTICAS
CUALITATIVAS DE LA RESPUESTA
TRANSITORIA**

TESIS

Que para obtener el grado de
DOCTOR EN INGENIERÍA ELÉCTRICA

presenta

Nelio Pastor Gómez

Juan José Flores Romero

Director de Tesis

Universidad Michoacana de San Nicolás de Hidalgo

Agosto 2005

*“Dichoso aquel que puede penetrar las
causas secretas de las cosas”*

A Rosy, Kelsey Sabine y Aixy Montserrat.

Agradecimientos

*“Lo visible crea un trabajo en forma,
lo invisible le da su valor”*

Quiero manifestar mi más profundo agradecimiento a todas las personas que directa o indirectamente han contribuido para el desarrollo de mis estudios de doctorado y la conclusión de esta tesis.

A las autoridades universitarias, por las facilidades prestadas.

A Los Ingenieros Ernesto Nuñez Aguilar y Felipe Sanchez Ramos, por el respaldo que me dieron como Directores y Presidentes del Consejo Técnico de la Facultad de Ingeniería Civil. Al Dr. Fernando Velasco y al Ing. David Adame, por el apoyo incondicional para poner en marcha este proyecto.

A mi asesor el Dr. Juan Jose Flores Romero, por que mas allá de compartirme su conocimiento entendió mis inquietudes y necesidades como estudiante y padre de familia.

A los sinodales Dr. Edmundo Barrera Cardiel, Dr. Felix Calderón Solorio, Dr. Claudio R. Fuerte Esquivel y Dr. Juan Antonio Ortega Ramírez. Por sus valiosas críticas y aportaciones durante el proceso de revisión de este documento. Quiero manifestar, además, que para mi es invaluable la disposición que siempre tuvieron para la revisión de este documento.

A José Luis Guillen, por su disponibilidad y don de gentes, además del gran apoyo en bibliografía electrónica y de software. A Sergio Bravo y Mario Graff por el soporte con los formatos de Latex.

Al M.C. Alberto Avalos Jefe del Laboratorio de Sistemas Potencia por su apoyo en la etapa de pruebas. Al Dr. Gerardo Barrera Cardiel por facilitarme el equipo de cómputo para llevar a cabo la adquisición de datos.

Al Dr. Claudio Fuerte y Dr. Felix Calderón, por su valiosa colaboración en el desarrollo del tercer artículo derivado de esta tesis.

A mi esposa e hijas, por permitirme robales parte de su tiempo.

A todos aquellos que por un error de omisión, no han sido nombrados.

A Dios que me dio origen, me permite fijarme metas y junto a mi va en busca de ellas.

Resumen

El problema de la identificación de sistemas inicia con una serie de tiempo de datos observados y trata de determinar el modelo más simple capaz de exhibir el comportamiento observado. Este problema de optimización consiste en buscar el mejor modelo en un espacio de posibles modelos. En esta tesis se presenta la teoría y los algoritmos para llevar a cabo la identificación cualitativa y cuantitativa de sistemas dinámicos invariantes y variantes en el tiempo.

El método propuesto está basado en la eliminación sucesiva de las componentes de la respuesta transitoria del sistema. Partiendo de las componentes de más alta frecuencia, hasta llegar a las componentes no oscilatorias. El algoritmo determina en forma simultánea la estructura y los parámetros del modelo, representado por una ecuación diferencial ordinaria de coeficientes constantes.

Se propone, además, un método para la identificación de sistemas que varían lentamente en el tiempo. Para este análisis se usa el método de tiempo congelado; por medio de este método se obtienen varios modelos lineales invariantes en el tiempo (LIT) en forma consecutiva. El algoritmo propuesto modela el comportamiento de los coeficientes de estos modelos por medio de polinomios.

La tesis está conformada básicamente en cuatro partes: La revisión de algunas propiedades de la respuesta de los sistemas lineales invariantes en el tiempo, y la propuesta de un conjunto de filtros que permiten la implementación del algoritmo de identificación, la propuesta del método para identificar sistemas que varían lentamente en el tiempo y finalmente se presentan algunas pruebas de la metodología propuesta.

Abstract

The problem of system identification starts with a time-series of observed data and tries to determine the simplest model capable of exhibiting the observed behavior. This optimization problem searches the model from a space of possible models. In this thesis, it is presented the theory and algorithms to perform qualitative and quantitative system identification for linear time-invariant and time-varying dynamic systems.

The method described here are based on successive elimination of the components of the transient system's response. Sinusoidal of high frequencies are eliminated first, then their carrying waves. It is continued with the process until it obtain a non-oscillatory remainder. The algorithm determines simultaneously the structure and the parameters of the model, and represents the model by an ordinary differential equation of constant coefficients.

Also, It is proposed a method for slowly time-varying system identification. For this analysis it use frozen-time approach; by this method it obtain consecutive several linear time-invariant models. The algorithm proposed approaches the behavior of the coefficients of these models by means of polynomials.

The thesis is conformed basically in four parts: The revision of some properties of the response of the linear time-invariant systems, the proposal of a set of filters that allow us the implementation of the identification algorithm. the proposal of the method to identify slowly time-varying systems, and finally, some tests of the proposed methodology are presented.

Contenido

Dedicatoria	I
Agradecimientos	III
Resumen	V
Abstract	VII
Lista de Figuras	XIII
Lista de Tablas	XVII
Lista de Algoritmos	XIX
Lista de Publicaciones	XXI
Lista de Símbolos	XXIII
1. Introducción	1
1.1. Objetivo	1
1.2. Tesis	1
1.3. Justificación	2
1.4. Estado del Arte	3
1.5. Contribuciones	9
1.6. Organización de la Tesis	11
2. Razonamiento Cualitativo e Identificación de Sistemas	13
2.1. Identificación de Sistemas Dinámicos	13
2.2. Razonamiento Cualitativo	14
2.2.1. Representaciones de Cantidad	15
2.3. Relación entre Razonamiento Cualitativo e Identificación de Sistemas	18
3. Modelado Matemático de Sistemas Dinámicos	25
3.1. Representación de Modelos por medio Ecuaciones Diferenciales	25
3.1.1. Abstracción de los Valores Numéricos a los Valores Simbólicos	26
3.1.2. Modelado e Identificación de Sistemas	27
3.1.3. Elementos Básicos en la Identificación de Sistemas	32
3.2. Sistemas Lineales	34
3.3. Sistemas Lineales e Invariantes en el Tiempo	35

3.3.1.	Solución de Ecuaciones Diferenciales Ordinarias con Coeficientes Constantes	36
3.3.2.	Respuesta Natural de los Sistemas Dinámicos	37
3.3.3.	Solución General	39
3.3.4.	Componentes Exponenciales	41
3.3.5.	Componentes Senoidales	42
3.4.	Sistemas Lineales e Variantes en el Tiempo	43
3.5.	Conclusiones	45
4.	Diseño del Algoritmo de Identificación	47
4.1.	Identificación de Sistemas Dinámicos	47
4.2.	Algoritmo QSI	48
4.2.1.	Filtro de Componentes Senoidales TPAFilter	53
4.2.2.	Cálculo de Puntos Medios	58
4.2.3.	Reconstrucción de Residuos	58
4.2.4.	Filtro de Componentes Exponenciales EXPFilter	60
4.3.	Identificación Cuantitativa	63
4.3.1.	Componentes Exponenciales	63
4.3.2.	Componentes Senoidales	69
4.4.	Obtención del Modelo	71
4.4.1.	Raíces de la Ecuación Característica	71
4.4.2.	Función Pol_carac	73
4.5.	Adquisición de Datos y Pretratamiento	74
4.6.	Generador de Respuestas Transitorias	78
4.6.1.	Componentes Exponenciales.	79
4.6.2.	Componentes Senoidales	80
4.7.	Conclusiones	81
5.	Identificación de Sistemas Lentamente Variantes en el Tiempo.	83
5.1.	Formulación del Problema	84
5.2.	Algoritmo de Identificación de Sistemas Variantes en el Tiempo	88
5.3.	Conclusiones	91
6.	Análisis de Complejidad	93
6.1.	Introducción	93
6.2.	Complejidad Computacional	94
6.2.1.	Complejidad Temporal	94
6.2.2.	Notación Asintótica	95
6.2.3.	Complejidad de un Algoritmo	96
6.3.	Complejidad del Algoritmo Splines Cúbico	97
6.4.	Complejidad de la función TPAFilter.	99
6.5.	Complejidad de la función ExpFilter.	101

6.6.	Complejidad Computacional del Algoritmo QSI	102
6.7.	Complejidad de QSI vs. Mínimos Cuadrados	103
6.8.	Conclusiones	103
7.	Resultados	105
7.1.	Coeficiente de Correlación	106
7.1.1.	Índice Cualitativo de Similitud	107
7.2.	Sistemas Lineales Invariantes en el Tiempo	108
7.2.1.	Caso de Estudio 1: Señal de Prueba	109
7.2.2.	Caso de Estudio 2: Circuito RLC	113
7.2.3.	Caso de Estudio 3: línea de Transmisión	115
7.3.	Sistemas Variantes en el Tiempo	119
7.3.1.	Caso de Estudio 4: Sistema Masa-Resorte	119
7.3.2.	Caso de Estudio 5: Línea de Transmisión	124
7.4.	Conclusiones	130
8.	Conclusiones y Trabajo Futuro	131
8.1.	QSI vs. Identificación Clásica	133
8.2.	Trabajo Futuro	134
8.2.1.	Identificación de Sistemas no Lineales y Modelado de Señales Periódicas	134
8.2.2.	Modelado y Control de Estructuras Mecánicas	135
	Referencias	137
	Glosario	147
A.	Desarrollo del Software de Identificación	149
A.1.	Introducción	149
A.1.1.	Características	149
A.2.	Estructura General del Software de Identificación de Sistemas Dinámicos	150
A.3.	Modo de texto	151
A.3.1.	Identificación de Sistemas Variantes en el Tiempo	156
A.3.2.	Generador de Experimentos	157
A.4.	Interfaz Gráfica con el Usuario	157
A.4.1.	El botón Start QSI	158
A.4.2.	El botón Experiment Design	161
A.4.3.	El botón Exit	162
A.5.	Instalación del Software QSI	163
A.5.1.	Instalación	163
B.	Teorema de Nyquist	165
C.	Algoritmo Genético del Criador	167

Lista de Figuras

2.1.	Representación de la derivada de una función continua desde el enfoque del álgebra de signos	21
2.2.	Razonamiento mediante el cual se detectan los puntos de cambio en la serie de tiempo	21
2.3.	Clasificación de los puntos de cambio e identificación de la envolvente	22
2.4.	Forma cualitativa de las componentes senoidales	23
3.1.	Representación de los niveles de abstracción de un sistema físico . . .	27
3.2.	Modelado y caracterización. <i>a)</i> Tipo de modelado. <i>b)</i> Caracterización de los modelos	28
3.3.	Proceso típico para realizar la identificación de sistemas	30
3.4.	Suma de tres exponenciales	42
3.5.	Suma de dos senoidales de frecuencias diferentes	43
3.6.	Resultado de la suma de dos senoidales con frecuencias cercanas entre sí	44
4.1.	Diagrama de bloques del proceso de identificación usando QSI	49
4.2.	Filtrado de las componentes senoidales	52
4.3.	Filtrado de las componentes exponenciales de la respuesta	52
4.4.	Obtención de la envolvente de la señal de entrada	57
4.5.	Un paso en el proceso de filtrado de las componentes oscilatorias . . .	59
4.6.	Señal $X^*(t)$, que no presenta extremos	62
4.7.	Puntos importantes en la eliminación de los componentes exponenciales	67
4.8.	Parametrización de una componente senoidal	70
4.9.	Estructura general de un sistema de adquisición de datos	75
4.10.	Kernel de un filtro pasa banda ideal	78
5.1.	Aplicación del principio de tiempo congelado para modelar sistemas variantes en el tiempo usando QSI	84
5.2.	Matriz de coeficientes obtenida a partir de los modelos de tiempo congelado	85
5.3.	Matriz de coeficientes y su relación con los coeficientes de la EDO . .	86

5.4.	Diagrama de bloques del algoritmo genético usado para estimar los coeficientes del polinomio que caracteriza los coeficientes de la EDO	88
6.1.	Forma gráfica de las notaciones $O(g(N))$, $\Omega(g(N))$ y $\Theta(g(N))$	96
6.2.	Particionamiento de una señal y enlace de cada punto mediante un spline	98
6.3.	Complejidad de la función TPAFilter	100
6.4.	Análisis de complejidad de la función ExpFilter	101
6.5.	Análisis de complejidad del Algoritmo QSI	102
7.1.	Etiquetado de una serie de tiempo, de acuerdo con la magnitud y signo de la transición de sus valores	108
7.2.	Gráfica de dos series de tiempo cualitativamente similares	109
7.3.	Respuesta de un sistema LIT	110
7.4.	Eliminación de componentes por QSI	110
7.5.	Primera componente filtrada por QSI en el caso de estudio 1.	111
7.6.	Componentes de primer orden del residuo exponencial	111
7.7.	Comparación entre la respuesta del modelo y la respuesta observada.	113
7.8.	Circuito RLC para el caso de estudio 2	114
7.9.	Respuesta al escalón del circuito RLC	114
7.10.	Modelo de Simulink para una línea de transmisión monofásica	116
7.11.	Voltaje en la línea de transmisión del caso de estudio 3	116
7.12.	Respuesta transitoria de la línea de transmisión del caso de estudio 3	117
7.13.	Comparación entre la respuesta del modelo y la respuesta observada	118
7.14.	Comportamiento de los coeficientes estimado por medio de algoritmos genéticos	123
7.15.	Respuestas de los dos modelos estimados comparadas con la respuesta observada	124
7.16.	Línea de transmisión <i>a</i>). Diagrama monofásico <i>b</i>). Circuito equivalente	125
7.17.	Señal capturada y detalle del proceso de filtrado	126
7.18.	Tres ajustes diferentes para el coeficiente $C_1(t)$	128
7.19.	Comparación entre los valores esperados y los valores estimados para las funciones $R(t)$, $L(t)$ y $C(t)$	129
7.20.	Comparación entre la respuesta observada y la respuesta del modelo estimado	130
A.1.	Diagrama de flujo del software QSI	151
A.2.	Señal usada como parámetro de entrada para la función <i>qsi</i>	153
A.3.	Respuesta transitoria generada con el algoritmo de generación de respuestas	158
A.4.	Interfaz gráfica del software QSI	159
A.5.	Cuadro de diálogo para la entrada de datos	159
A.6.	Interfaz gráfica para desplegar el resultado de la identificación	160
A.7.	Cuadro de diálogo para configurar la respuesta simulada	161

A.8. Interfaz gráfica de la generación de respuestas transitorias	162
-----------------------------------------------------------------------------	-----

Lista de Tablas

4.1. Estructura de la matriz de parámetros	50
4.2. Estructura de la matriz de parámetros de las componentes senoidales	54
4.3. Matriz de parámetros de las componentes senoidales y exponenciales .	74
5.1. Datos de entrada para el algoritmo QSITimeVarying	90
7.1. Niveles de exactitud del coeficiente de correlación.	107
7.2. Definición de las etiquetas	107
7.3. Coeficientes del sistema masa-resorte	120
7.4. Coeficientes de las funciones $M_a(t)$, $Fr(t)$ y $K(t)$	121
7.5. Valores de R para el experimento de la línea de transmisión	126
7.6. Coeficientes del experimento de la línea de transmisión	127

Lista de Algoritmos

1.	Algoritmo QSI	49
2.	Filtro de componentes senoidales	55
3.	Filtro de componentes exponenciales	61
4.	Identificación de sistemas variantes en el tiempo	89

Lista de Publicaciones

Artículo 1

Juan J. Flores, Nelio Pastor y José J. Rincón “Qualitative System Identification for Linear Time-Invariant Dynamic Systems”. The Sixteenth International Workshop on Qualitative Reasoning QR 2002, p. 61-66.

Publicado también como capítulo de libro: MICAI’2002: Advances in Artificial Intelligence. Editors: Carlos A. Coello Coello, Alvaro de Albornoz, Enrique Sucar & Osvaldo Cairo. Springer Verlag 2002.

Artículo 2

Juan J. Flores, Nelio Pastor. “Time-Invariant Dynamic Systems Identification Based on the Qualitative Features of the Response”. publicado en : Journal of Engineering Applications of Artificial Intelligence, volume 18 issue 6 pp. 719-729.

Artículo 3

Nelio Pastor, Juan J. Flores, Felix Calderón, Claudio Fuerte. “Identification of Slowly Time Varying Systems Based on the Qualitative Features of Transient Response a Frozen Time Approach”. Enviado para revisión a IEEE Transactions on Systems, Man, and Cybernetics–Part C.

Lista de Símbolos

n	Orden del modelo.
n_1	Número de componentes exponenciales.
n_2	Número de componentes senoidales.
r	Factor exponencial de decaimiento.
ω	Frecuencia angular.
ω_0	Frecuencia central.
ω_{max}	Frecuencia de corte superior.
ω_{min}	Frecuencia de corte inferior.
ϕ	Angulo de defasamiento.
E_{n_1}	Etiqueta de componentes exponenciales.
ES_{n_2}	Etiqueta de componentes senoidales.
CCC	Coficiente de Correlación Cualitativo.
CCN	Coficiente de Correlación Numérico.
u	Señal de excitación del sistema.
$y(t)$	Respuesta del sistema.
$\hat{y}(t)$	Respuesta estimada.
$X(t)$	Datos de entrada para QSI.
$X^*(t)$	Residuo.
P	Matriz de parámetros.
k	Instante congelado.
K	Constante del resorte.
m	Autovalor de la ecuación característica.
D	Operador diferencial.
a	Coficiente de la expresión exponencial.
$x'(t)$	Derivada de $x(t)$.
T_s	Periodo de muestreo de la señal de entrada.
T	Periodo de la componente senoidal.
M_a	Masa.
F_r	Fuerza de fricción.

Capítulo 1

Introducción

1.1. Objetivo

El objetivo de esta tesis es desarrollar un método para obtener el modelo matemático de sistemas dinámicos, variantes e invariantes en el tiempo, partiendo del análisis cualitativo y cuantitativo de su respuesta transitoria. El método propuesto deberá ser capaz de determinar la estructura del modelo mediante el análisis cualitativo de la respuesta y los parámetros de esa estructura mediante un análisis cuantitativo.

1.2. Tesis

La respuesta transitoria de un sistema dinámico se puede expresar como la suma de componentes exponenciales, reales y/o complejos. Mediante un análisis cualitativo es posible descomponer esta suma en sus componentes más simples, determinar la estructura del modelo y expresarla mediante una ecuación diferencial ordinaria de coeficientes constantes. Una vez que se tienen por separado cada uno de los componentes de la respuesta se lleva a cabo un análisis cuantitativo para estimar los coeficientes de esa ecuación diferencial.

Los sistemas dinámicos que varían lentamente en el tiempo se pueden represen-

tar mediante una secuencia de modelos invariantes en el tiempo usando el principio de “tiempo congelado” [Desoer70, Kamen89, Freedman68 et al, LeYi95]. El comportamiento de los parámetros del conjunto de modelos obtenidos se puede aproximar mediante funciones polinomiales.

1.3. Justificación

En los métodos tradicionales, tales como Mínimos Cuadrados, Variable Instrumental, Máxima Verosimilitud, etc. el proceso de identificación de estructura recibe menos atención que la parametrización del modelo. En la mayoría de los casos la estimación estructural no se lleva a cabo en forma automatizada ya que el usuario selecciona un modelo de un conjunto de posibilidades. La selección de un determinado modelo es crucial para una buena aproximación, por lo general, esta selección se basa en el conocimiento del proceso de identificación y en el conocimiento previo de la dinámica del sistema que se quiere identificar. En esta tesis se presenta una forma alterna para inferir la estructura del modelo y para estimar los parámetros que mejor ajusten el comportamiento observado. El método propuesto no requiere información referente al orden del sistema, ya que mediante el análisis cualitativo de la respuesta transitoria del sistema se propone la estructura y el orden del modelo. Esto representa una ventaja con respecto a los métodos tradicionales mencionados anteriormente, ya que estos requieren que se les proporcione el orden del modelo que se desea identificar. Otra característica relevante, como se verá en el capítulo 7, se deriva del análisis de complejidad de donde se obtiene que la complejidad computacional de los algoritmos propuestos es $O(N)$, es decir su complejidad en tiempo es lineal con respecto al vector de muestras, mientras que el algoritmo de mínimos cuadrados clásico tiene una complejidad cuadrática y el mínimos cuadrados recursivo tiene una complejidad $O(N \log N)$ [An-Yeu94]. Esto significa que el conjunto de algoritmos propuestos en esta tesis mejoran el desempeño de uno de los métodos más usados para identificación de

sistemas.

Los sistemas reales son en forma natural variantes en el tiempo, debido a los cambios en las condiciones de operación y al envejecimiento de los componentes, entre otras causas. Uno de los sistemas más simples y fáciles de abordar son los sistemas que varían lentamente en el tiempo, cuyo comportamiento se puede considerar como el de un sistema Lineal e Invariante en el Tiempo (LIT) durante un pequeño período de tiempo [Freedman68 et al]. Los sistemas con variaciones lentas en el tiempo son de gran importancia tanto en aplicaciones prácticas como en estudios teóricos. Muchos sistemas reales varían lentamente en el tiempo. Por ejemplo, las variaciones en las condiciones ambientales son usualmente mucho más lentas que la dinámica de un sistema. Como resultado, los sistemas dinámicos con parámetros dependientes del ambiente (temperatura, presión, altitud, etc.) frecuentemente pueden ser modelados como sistemas que varían lentamente en el tiempo.

Con la finalidad de contribuir al desarrollo de teorías acerca de los sistemas variantes en el tiempo, se propone extender el método desarrollado para sistemas invariantes en el tiempo, aplicando el principio de tiempo congelado [Desoer70, LeYi95, Freedman68 et al], para analizar sistemas que varían lentamente con el tiempo. Este principio ha sido usando ampliamente para el diseño de sistemas de control [LeYi95, Freedman68 et al], pero no propiamente para la obtención del modelo de la planta.

1.4. Estado del Arte

El área del modelado e identificación de sistemas ha avanzado rápidamente desde los tiempos de Gauss. En 1795 Gauss fué pionero haciendo las primeras contribuciones a la estimación, con su formulación de la integral logarítmica para estimar la densidad de los números primos, y con su fórmula de interpolación por medio de diferencias finitas. En los últimos tiempos se ha visto desde las aplicaciones del filtro de Kalman (en los 60's) hasta las redes neuronales y algoritmos genéticos en los últimos días.

El área de identificación y modelado ocupa una posición muy importante dentro del avance tecnológico, por ejemplo, un área de particular importancia para la NASA son los modelos de la dinámica estructural y el análisis de los modos del sistema que representa a una nave espacial [Longman99]. Una nave espacial típica exhibe varios modos de vibración y requiere un buen modelo para asegurar el éxito de los actuadores del sistema de control. En una nave espacial grande este problema de identificación puede ser todo un desafío, por el número de modos de vibración y el tamaño del modelo requerido. El análisis de los modos se requiere para las nuevas naves espaciales antes de su primer vuelo para asegurar que los modos de vibración no serán excitados por la aerodinámica, produciendo oscilaciones aerolásticas y la falla estructural de las alas. Un algoritmo para la identificación de sistemas de este tipo fué desarrollado en las instalaciones de la NASA en Langley, Virginia [Longman99], este algoritmo está basado en una formulación de mínimos cuadrados lineales y resuelto por medio de descomposición de valores singulares. Un paquete para MatLab de este algoritmo es distribuido por la NASA a través de la empresa COSMIC, y es ampliamente usado por varias industrias del ramo.

En la práctica, la identificación de sistemas está caracterizada por muchas operaciones y cálculos para determinar el mejor modelo, lo que hace necesario un buen soporte de software. Existen diferentes paquetes comerciales disponibles para identificación, tales como los algoritmos para MatLab, contenidos en el “System Identification Toolbox” desarrollado por Lennart Ljung [Ljung05] y “Frequency Domain System Identification Toolbox” desarrollado por István Kollar [Kollar]. Como éstos, existen otros programas con módulos interactivos para la identificación de sistemas, por ejemplo, “MATRIXx” y “XMath” desarrollados por Peter Van Overschee [Overschee96] para Integrated Systems Inc. de Santa Clara, California.

La identificación de sistemas tiene sus orígenes en las técnicas estadísticas estándares, en particular, dos de estas rutinas básicas tienen una interpretación directa, y son conocidas como los métodos de los mínimos cuadrados y de máxima verosimilitud

[Alvarez83]. La comunidad del área de control ha tomado una parte muy activa dentro de la aplicación y desarrollo de esas técnicas básicas para los sistemas dinámicos, justamente después del nacimiento de la “Teoría de control moderna” en los años 60’s. La estimación por medio de máxima verosimilitud fue aplicada para ecuaciones de diferencias por Åström y Bholin en 1965. Posteriormente han florecido varias técnicas de estimación y parametrización de modelos.

Durante los últimos 20 años la identificación de sistemas ha sido un campo con bastantes aproximaciones “hechas a la medida”, cada una aplicable a una clase de problemas. Por ejemplo, Juang [Juang96], plantea un enfoque basado en las aplicaciones en el espacio de estado, orientándose a la identificación de modelos descritos en variables de estado; Overschee y De Moor [Overschee96], plantean métodos de subespacio para la identificación de modelos lineales en el espacio de estado; y Chen y Gu plantean la identificación de modelos lineales invariantes en el tiempo con un criterio H_∞ [Chen00].¹

Lo más común es encontrar bibliografía sobre identificación de sistemas enfocada en los métodos clásicos de estimación basados en probabilidad y estadística, incluyendo estimación espectral y modelado de series de tiempo: Box and Jenkins [BoxJenkins94], Eykhoff [Eykhoff74], Goodwing and Payne [Goodwin77], Davis and Vinter [Davis85], y Kashyap and Rao [Kashyap76], por citar algunos.

La información proporcionada por la bibliografía anterior es de gran importancia para los investigadores del área de identificación de sistemas. Sin embargo, Ljung en su libro “System Identification Theory for the User” [Ljung99] es más ambicioso que la bibliografía antes mencionada ya que enfoca la teoría de forma tal que permite comprender y aplicar directa y prácticamente las técnicas disponibles. Esencialmente Ljung pretendió incluir los métodos más modernos, así como la mayoría de las técnicas

¹ H_∞ Es un método de la teoría de control usada para el diseño de controladores. Esencialmente es un método de optimización que minimiza la norma de las matrices de transferencia de lazo cerrado; la norma de un sistema dinámico se define como la máxima amplificación que puede hacer el sistema a la energía de la señal de entrada.

clásicas de estimación disponibles. Quizá lo más significativo es que Ljung abarca todos los aspectos del proceso de la identificación de sistemas, desde el diseño de los experimentos hasta la selección de la estructura del modelo, desde la captura de datos hasta la estimación paramétrica y la validación. De esta forma el texto es, por un lado accesible y por otro lado lo suficientemente amplio como para ilustrar completamente el proceso de la identificación de sistemas. Quizá el libro que más se le empalma, en contenido, sea el de “System Identification” de Söderstrom y Stoica [Stoica89]. El cual contiene más detalle teórico, sin embargo la selección, organización y presentación de los tópicos, hace que el texto de Ljung sea más accesible.

En la actualidad esta área ha madurado con la amplia difusión de las técnicas básicas y el planteamiento de nuevos métodos basados en la programación evolutiva, tales como algoritmos genéticos, redes neuronales y lógica difusa [ChingTeng96, Nelles01]. Con el advenimiento de estos métodos y las técnicas actuales de optimización se puede hacer una estimación de la estructura más amplia de los modelos y de esta manera abarcar una clase más extensa de sistemas. Sin embargo, las técnicas de estimación paramétrica de más uso siguen siendo las basadas en los mínimos cuadrados y en las técnicas estadísticas.

El trabajo doctoral de Cem Say [CemSay92, CemSay96] presenta contribuciones acerca del modelado de sistemas físicos dentro del contexto de la inteligencia artificial, particularmente en el área de Razonamiento Cualitativo. El trabajo se basa en la representación propuesta por Benjamin Kuipers [Kuipers94] y usada en el programa de simulación cualitativa (QSIM por sus siglas en inglés de “Qualitative SIMulation”). Cem Say presenta el algoritmo de identificación cualitativa de sistemas (QSI por sus siglas en inglés de “Qualitative System Identification”). Este algoritmo propone modelos cualitativos descritos por ecuaciones diferenciales cualitativas (QDE por las siglas en inglés de Qualitative Differential Equation) Este algoritmo inicia con una etapa de verificación de restricciones. Es decir, dado un conjunto de restricciones y un conjunto de parámetros, el algoritmo verifica la combinación que se ajuste al

conjunto de comportamientos. La combinación que satisfaga la restricción se elige para que forme parte la QDE de salida, posteriormente esta QDE se simula usando QSIM y se valida si cumple con el conjunto de comportamientos observados, en caso afirmativo se imprime esta QDE como el modelo final, en otro caso se procede a extender el modelo agregando otras componentes que mejoren el ajuste al conjunto de comportamientos. Los componentes que forman a la QDE final son la integración de fragmentos de modelos tomados de una biblioteca de modelos parciales creados a partir de leyes físicas [DeKleer84, Trave-Massuyes91 et al].

Liliana Ironi también ha trabajado en el desarrollo de métodos para el modelado y simulación cualitativos, presentando métodos novedosos para la identificación de sistemas no lineales, combinando modelado cualitativo con lógica difusa [Bellazzi99], la identificación se lleva a cabo de la siguiente manera: partiendo del conocimiento a priori del sistema se plantea un modelo cualitativo representados por una ecuación diferencial cualitativa, se extraen los posibles comportamientos de este modelo usando QSIM y se mapean en el conjunto de reglas difusas. Con esta información se construye el sistema difuso y se lleva a cabo la identificación. En [GuglielmannIroni04] Guglielmann e Ironi demuestran como el Razonamiento Cualitativo juega un rol muy importante en la mejora de la robustez e interpretabilidad de los modelos obtenidos mediante esta técnica. Una versión ampliada hacia las redes neuronales se da en [Bellazzi00] donde el conjunto de reglas difusas sirven como reglas para el entrenamiento de la red neuronal. En particular su línea de investigación está orientada al trabajo para desarrollar métodos cualitativos partiendo de los métodos tradicionales para resolver problemas relacionados con la medicina y la ciencia de materiales [Liliana98, Capelo98].

Por su parte Elizabeth Bradley ha trabajado en el área de inteligencia artificial desarrollando herramientas computacionales para analizar sistemas e identificar sus modelos. Su proyecto llamado PRET [Bradley96] es un software que modela la dinámica interna de sistemas no lineales observando únicamente sus salidas. La parte

principal de este programa está basada en una biblioteca de métodos tradicionales de identificación de sistemas y aplica técnicas de inteligencia artificial para automatizar los procesos de identificación que en las técnicas tradicionales desarrolla el usuario. Estas técnicas de inteligencia artificial seleccionan y aplican el método apropiado seleccionandolo de una biblioteca y llevan a cabo un razonamiento cualitativo, simbólico y algebraico de las entradas proporcionadas por el usuario para realizar el proceso de identificación en forma totalmente automatizada.

Otra aplicación del razonamiento cualitativo al modelado de sistemas se encuentra en [Price03 et al] en este artículo Price y sus coautores modelan sistemas eléctricos que contienen sistemas mecánicos y electrónicos usados en automóviles. Describen un rango de modelos de simulación que pueden construirse a partir de la información que proporciona el circuito esquemático. Los resultados de la simulación se enlazan para establecer una noción de la funcionalidad común del sistema. Llevan a cabo dos niveles de razonamiento: razonamiento del comportamiento cualitativo del sistema y un razonamiento acerca del comportamiento de los elementos que forman el sistema. Estos dos niveles de razonamiento les ayuda a automatizar el proceso de diseño de los sistemas eléctricos mencionados. En la simulación de los modelos el sistema se representa como una red de resistencias, las subredes de resistencias se generan a partir de los modelos de cada componente. Las dependencias del diagrama esquemático describen de que forma se interconectan estas subredes de resistencias, del estado de los componentes durante la simulación se obtiene un conjunto detallado de datos que se usa para automatizar el análisis del diseño.

En su artículo [LiuCoghill04] Liu y Coghill proponen un método novedoso de modelado cualitativo, aplicado a robots cinemáticos, para ello utilizan principios de Cinemática Cualitativa² y Física cualitativa [DeKleer84]. Los autores presentan propiedades de mapeo para transferir la información cualitativa entre las entradas y

²La Cinemática Cualitativa es una rama de la mecánica cualitativa [Forbus87 et al, Oliver95 et al] que estudia el movimiento en espacios cualitativos, prescindiendo de las fuerzas que lo producen.

salidas del sistema en términos de valores característicos de los intervalos del círculo unitario. Las representaciones de velocidad y aceleración se hacen en terminos cualitativos basandose en principios de la Física cualitativa.

1.5. Contribuciones

De esta tesis se han derivado tres publicaciones una como capitulo del libro MICAI'2002: Advances in Artificial Intelligence [Flores02 et al]. El segundo artículo fue publicado en la revista "Journal of Engineering Applications of Artificial Intelligence" [Flores05]. y el tercer artículo fue Enviado para revisión a IEEE Transactions on Systems, Man, and Cybernetics–Part C [Pastor05 et al]. A continuación se presentan las contribuciones de los artículos, derivados de esta tesis.

El artículo "Qualitative System Identification for Linear Time-Invariant Dinamic Systems". publicado en The Sixteenth International Workshop on Qualitative Reasoning QR 2002. Presenta la teoría y algoritmos para llevar a cabo la identificación cualitativa de sistemas dinámicos lineales e invariantes en el tiempo. El método descrito se basa en la separación sucesiva de las componentes de la respuesta transitoria del sistema en estudio. Este procedimiento permite determinar la estructura del modelo en forma automatizada. El artículo está organizado en dos partes, el planteamiento de algunas propiedades matemáticas de la respuesta de los sistemas lineales e invariantes en el tiempo, y la propuesta de un conjunto de filtros que permiten la implementación del algoritmo de identificación. La contribución que aqui se presenta es la identificación automática de la estructura del modelo y el orden del mismo, que en el caso de los métodos clásicos, tales como los algoritmos para MatLab [Ljung05, Kollar], "MATRIX" y "XMath" [Overschee96], lo tiene que realizar el usuario.

El artículo "Systems Identification Based on the Qualitative Features of the Response", publicado en : Journal of Engineering Applications of Artificial Intelligence. Es una versión que corrige, complementa y amplía el artículo "Qualitative System

Identification for Linear Time-Invariant Dynamic Systems”. En este artículo se proponen y demuestran dos teoremas, el primer teorema se refiere al comportamiento cualitativo de la suma de exponenciales y con ello se determina la cantidad de componentes que la forman. El segundo teorema describe las características que tienen cada uno de los componentes exponenciales de la suma. Este teorema nos permite llevar a cabo la separación de las componentes exponenciales. Estos dos teoremas permiten proponer la metodología para llevar a cabo la identificación cuantitativa del modelo.

Este artículo presenta, además de la teoría para la identificación cualitativa, la teoría para identificación cuantitativa de sistemas dinámicos lineales e invariantes en el tiempo. Se presenta, además, la aplicación de la metodología propuesta en el análisis y modelado de circuitos eléctricos y sistemas de potencia.

En el artículo “Identification of Slowly Time Varying Systems Based on the Qualitative Features of Transient Response a Frozen Time Approach”. Enviado para revisión a IEEE Transactions on Systems, Man, and Cybernetics–Part C. Se propone un método para la identificación estructural y paramétrica de sistemas que varían lentamente en el tiempo. Por medio del método de tiempo congelado se obtiene un conjunto de modelos consecutivos invariantes en el tiempo. El algoritmo que se presenta modela la variación de los coeficientes de la ecuación característica del sistema por medio de polinomios. Se obtiene un modelo óptimo usando algoritmos genéticos. Se reportan los resultados obtenidos probando el método con un experimento controlado en un laboratorio de sistemas de potencia, el sistema modelado fue una línea de transmisión.

En esta tesis el análisis cualitativo de la respuesta permite hacer la estimación del modelo a partir de la respuesta del sistema, esto difiere con los desarrollos presentados por Cem Say [CemSay96], Bellazi [Bellazzi00], Guglielmann e Ironi [GuglielmannIroni04] y Bradley [Bradley96] Los cuales requieren como parte de entrada el árbol de comportamientos generado por QSIM [Kuipers94]. Esto representa una limitante, ya que generalmente en un experimento de identificación no está garantizada mucha infor-

mación apriori del sistema y solo se cuenta con las señales de entrada y salida del sistema. El desarrollo que se presenta aquí solo requiere como datos de entrada la respuesta transitoria del sistema y la señal de entrada.

Por otro lado el desarrollo de Cem Say produce un modelo cualitativo, es decir, su modelo esta descrito por ecuaciones diferenciales cualitativas, lo cual implica que el usuario tiene que hacer por su cuenta la transformación al modelo cuantitativo. La metodología que se presenta en esta tesis produce un modelo descrito por una ecuación diferencial ordinaria, es decir, este modelo ya incluye los valores numéricos de los coeficientes de la EDO.

Los desarrollos presentados por Guglielmann e Ironi, además de requerir la información proporcionada por QSIM requieren que se les pase el orden del modelo que se va a identificar. El algoritmo aquí presentado no requiere esta información ya que es parte de funciones determinar el orden y la estructura del modelo.

Comparado con el software de uso clásico la mejor aportación que podemos resaltar es la parte de la estimación del orden del modelo, es decir en paquetes de gran aplicación como Matlab, los algoritmos requieren que se indique cual es el orden del modelo a identificar.

Una vez realizado el análisis de complejidad del algoritmo se obtuvo una complejidad $O(N)$, es decir lineal en el tamaño de la entrada. Comparado con una de las técnicas de identificación más usadas, como lo es el método de los mínimos cuadrados recursivos que tiene una complejidad $O(N \log N)$, se observa que el algoritmo propuesto en esta tesis tiene un mejor desempeño.

1.6. Organización de la Tesis

El contenido de esta tesis se ha organizado en ocho capítulos y tres apéndices. El primer capítulo es la introducción y en él se hace la presentación de los objetivos, la tesis y su justificación. Se presenta además el estado del arte, los artículos deri-

vados de esta tesis y su contribución. El capítulo dos presenta algunas definiciones usadas en el razonamiento cualitativo y su relación con la identificación de sistemas. El capítulo tres se describen los elementos fundamentales del modelado matemático y las herramientas utilizadas en el desarrollo de la tesis. El capítulo cuatro presenta el diseño del algoritmo de identificación propuesto, un método para generar respuestas transitorias con comportamientos interesantes para nuestro estudio. El capítulo cinco presenta la metodología propuesta para llevar a cabo la identificación de sistemas lentamente variantes en el tiempo. En el capítulo seis se efectúa el análisis de complejidad de los algoritmos propuestos. En el capítulo siete se presentan los resultados obtenidos y en el capítulo ocho las conclusiones y las propuestas de trabajo futuro. El apéndice A presenta el desarrollo del software de identificación y manual de instalación y uso. El Apéndice B presenta el teorema de Nyquist y el Apéndice D La descripción del Algoritmo Genético del Criador.

Capítulo 2

Razonamiento Cualitativo e Identificación de Sistemas

2.1. Identificación de Sistemas Dinámicos

La identificación de sistemas dinámicos es el proceso para determinar un modelo dinámico de un sistema desconocido. Se puede pensar en el sistema desconocido como una caja negra, teniendo disponible solamente las señales de entrada y salida. Así, podemos excitar sus entradas y observar las salidas. La identificación de sistemas deduce un modelo matemático de la dinámica interna del sistema partiendo de la observación de sus salidas. Este proceso se puede descomponer en dos partes: Identificación estructural, donde se trata de determinar la forma del modelo (típicamente una ecuación diferencial ordinaria), y la estimación paramétrica, donde se determinan los valores de los parámetros (coeficientes) del modelo. La primera parte se conoce también como identificación cualitativa, y la segunda como identificación cuantitativa, de Sistemas. Es pertinente observar que el término estimación paramétrica es usado en esta tesis para referir la estimación de los coeficientes de la ecuación característica de la ecuación diferencial ordinaria que describe al sistema.

La identificación de sistemas puede ser vista como un proceso de búsqueda en un

conjunto de posibles modelos, guiándose por los datos observados en un sistema físico. En este proceso de búsqueda, primeramente, se trata de determinar el modelo que mejor reproduzca las distinciones cualitativas del comportamiento observado, es decir, estamos determinando la estructura del modelo. El segundo paso consiste en estimar los parámetros del modelo propuesto en el paso anterior, y de esta manera tener una descripción completa del modelo que mejor reproduzca la dinámica observada.

Los métodos tradicionales para la identificación de sistemas continuos inician con una ecuación diferencial y el espacio de búsqueda se reduce a las combinaciones posibles de valores reales de los coeficientes de la Ecuación Diferencial Ordinaria (EDO) dada, en el caso de sistemas discretos el modelo se representa por una ecuación en diferencias [Ljung99]. En las técnicas tradicionales de ingeniería, el ingeniero es quien lleva a cabo la identificación estructural basándose en el conocimiento apriori del sistema en estudio, dejando a la computadora solamente el cálculo de los parámetros.

2.2. Razonamiento Cualitativo

Esta sección proporciona una descripción de diferentes conceptos que han sido desarrollados y presentados por Johan de Kleer en su trabajo “A Qualitative Physics Based on Confluences” [DeKleer84] y Kenneth D. Forbus en “Qualitative Reasoning” presentado como un capítulo del CRC Handbook of Computer Science and Engineering [Forbus96].

El razonamiento cualitativo es una área de la inteligencia artificial, mediante la cual se crean representaciones de aspectos contínuos del mundo real, tales como espacio, tiempo y cantidad, con los cuales se puede crear un razonamiento con poca información [Forbus96]. Típicamente se enfoca a dominios de la ciencia y la ingeniería, por lo que también se le conoce como física cualitativa. Y de igual manera que la física clásica es aplicable tanto a sistemas discretos como a sistemas continuos. Este campo del conocimiento tiene dos motivaciones: primero, la gente puede sacar conclusiones

muy útiles acerca del mundo físico sin usar ecuaciones diferenciales ni información cuantitativa. En la vida diaria podemos explicar lo que está pasando en nuestro alrededor y de qué manera nos puede afectar ésto, utilizando poca información y además no muy precisa. Segundo, los científicos e ingenieros utilizan un razonamiento cualitativo cuando inicialmente tratan de entender un problema, cuando proporcionan información para resolverlo mediante métodos más formales y cuando interpretan los resultados de una simulación cuantitativa. De esta forma los avances en razonamiento cualitativo permiten la creación de software que pueda ayudar a los científicos e ingenieros a tomar decisiones o producir análisis como los mencionados.

El razonamiento cualitativo tiene sus principios en 1977 en las investigaciones de De Kleer sobre cómo el conocimiento cualitativo y cuantitativo interactúan en la solución de un conjunto de problemas simples de mecánica clásica [DeKleer77].

La definición de Razonamiento Cualitativo se podría resumir de la siguiente manera: Razonamiento relacionando a una descripción no numérica de un sistema, conservando todas sus propiedades y distinciones importantes de comportamiento. Los Modelos cualitativos se orientan a capturar los aspectos fundamentales de un sistema o mecanismo, a un nivel de abstracción alto. Los Métodos tales como abstracción y aproximación son usados con frecuencia para construir modelos basados en aspectos cualitativos más que en los aspectos numéricos de un sistema.

Las representaciones fundamentales usadas en el razonamiento cualitativo se resumen en los siguientes grupos: cantidad, relaciones matemáticas, consideraciones de los modelos, causalidad, espacio, y tiempo [DeKleer84].

2.2.1. Representaciones de Cantidad

Mucho del esfuerzo en la investigación se hace el sentido de tratar de entender las propiedades de las representaciones a niveles altos de abstracción, ya que las propiedades de baja abstracción tienden a estar bien definidas por las propiedades matemáticas.

La representación de más alta abstracción para parámetros continuos es la abstracción de estado [DeKleer84], que es la que representa si una cantidad está dentro de un rango de operación normal o no. Esta es una representación útil para ciertas tareas por que es la representación más simple que puede expresar la diferencia entre lo que está funcionando bien o mal. El siguiente paso en la resolución es el álgebra de signos [DeKleer84, Forbus96], que representa los parámetros continuos como “-”, “+” o “0”, de acuerdo a si el signo del parámetro en consideración es negativo, positivo o cero. El álgebra de signos es muy poderosa: ya que las derivadas de los parámetros son variables cuyos valores pueden ser representados como signos; algunos de los principales resultados del cálculo diferencial pueden ser aplicados para razonar acerca del valor de los signos. Esto permite que el álgebra de signos sea usada para hacer razonamiento cualitativo acerca de la dinámica de un sistema, incluyendo las propiedades de las expresiones tales como oscilaciones y estabilidad.

La representación de valores continuos a través de conjuntos de relaciones ordinales (también conocidos como representación de espacio de cantidad) es el siguiente paso en la resolución. Por ejemplo, la temperatura de un fluido podría ser representada en términos de la relación entre el punto de congelación y el punto de ebullición del material que lo compone. Como en el álgebra de signos, los espacios de cantidad son expresiones suficientes para apoyar el razonamiento cualitativo acerca de la dinámica (el álgebra de signos puede ser modelada como un espacio de cantidad con un solo punto de comparación, el cero). Al contrario del álgebra de signos, que traza valores desde una estructura algebraica finita definida, los espacios de cantidad proporcionan una resolución variable porque los nuevos puntos de comparación se pueden agregar para refinar los valores. La temperatura del agua en un recipiente sobre una estufa, por ejemplo, probablemente se definiría en términos de su relación con la temperatura de la estufa, así como de sus puntos de congelación y ebullición.

Hay dos tipos de puntos de comparación usados en la definición de los espacios de cantidad, los puntos límite y los puntos de cambio (landmarks, término introducido

por Benjamin Kuipers en el desarrollo de QSIM [Kuipers94]), los puntos límite se derivan de las propiedades generales de un dominio aplicables a una situación específica, y los puntos de cambio son puntos con valores constantes de comparación introducidos durante el razonamiento para proporcionar una resolución adicional. Para determinar si un sistema oscilatorio de segundo orden es amortiguado, subamortiguado o críticamente amortiguado, por ejemplo, se requiere la comparación de los valores de los picos sucesivos. Denotando el valor del pico de un ciclo en particular como un landmark, y comparándolo con los landmarks generados en los ciclos sucesivos, proporcionando información para poder hacer la inferencia.

Los intervalos son bien conocidos como una representación de resolución de una variable para valores numéricos, y han sido muy usados en el razonamiento cualitativo [Gotz83, Moore66, Benhamou94]. Un espacio de cantidad puede ser considerado como información parcial de un conjunto de intervalos. Si tenemos información completa acerca de la relación ordinal entre los puntos límite y los valores de los landmarks, estos puntos de comparación definen un conjunto de intervalos que particionan el dominio de los parámetros. Este mapeo natural entre espacios de cantidad e intervalos ha sido explotado para una variedad de sistemas que usan intervalos cuyos puntos finales son valores numéricos conocidos para refinar las predicciones producidas por el razonamiento cualitativo.

La representación de orden de magnitud clasifica (estratifica) los valores de acuerdo con alguna noción de escala. Esta representación es importante para resolver ambigüedades y para simplificar modelos por que habilitan el razonamiento acerca de qué fenómenos y efectos pueden ser ignorados con seguridad, dado que no contribuyen en forma significativa en la dinámica del sistema. En la literatura se han usado varias de éstas técnicas de clasificación entre las que se incluyen márgenes numéricos y escalas logarítmicas.

2.3. Relación entre Razonamiento Cualitativo e Identificación de Sistemas

El razonamiento cualitativo ha sido un campo de investigación muy activo dentro de la inteligencia artificial durante mucho tiempo [Forbus96]. Motivada por el objetivo de construir una ingeniería artificial, el razonamiento cualitativo se ha propuesto automatizar los diferentes tipos de razonamientos que los ingenieros, técnicos y científicos hacen cuando analizan, explican, diseñan o modelan sistemas físicos.

Se puede observar, entonces, que el razonamiento cualitativo ha tenido un considerable impacto no solo en el área de inteligencia artificial, sino en general el campo de la ingeniería que se ha beneficiado de las novedosas soluciones surgidas de sus técnicas, y que han permitido tener un nuevo enfoque de como modelar los sistemas físicos [DeKleer84].

El análisis de sistemas dinámicos está basado en modelos matemáticos construidos a partir de leyes físicas o usando algún método específico basado en la observación de series de tiempo para identificar los parámetros del modelo. Mediante el razonamiento cualitativo se realiza la aproximación para obtener modelos cualitativos.

Casi nunca hay un modelo único que sea correcto para un sistema físico. La mayoría de los sistemas pueden ser modelados de varias formas, y las diferentes aplicaciones requieren diferentes tipos de modelos. El diseño de un modelo para un propósito específico aún es algo de arte. El razonamiento cualitativo ha desarrollado formalismos que combinan lógica y matemáticas con representaciones cualitativas para ayudar en la automatización del proceso de crear y refinar modelos. Una de las características relevantes del razonamiento cualitativo es la gran aportación de métodos para la creación automática de modelos para tareas específicas [Forbus96]. Estos métodos formalizan el conocimiento y las habilidades que, típicamente, están implícitas en el conocimiento de la mayoría de los matemáticos y los ingenieros. De esta manera las bibliotecas de modelos usados en el razonamiento cualitativo que fueron mayormente

te construidas “a mano”, usando leyes específicas del sistema en cuestión, podrán ampliarse mediante procedimientos automatizados y formales.

El algoritmo, de identificación de sistemas, presentado en esta tesis se basa en la interpretación cualitativa de la serie de tiempo producto de la respuesta transitoria del sistema y produce un modelo descrito mediante una ecuación diferencial ordinaria. Una de las principales diferencias con el algoritmo presentado por Cem Say [CemSay96] radica precisamente en el tipo de modelo producido, ya que el algoritmo de Cem Say produce un modelo descrito por una ecuación diferencial cualitativa.

De la teoría de la Ecuaciones Diferenciales Ordinarias, se desprende que la respuesta natural de un sistema de orden n es la suma de $n_1 + n_2$ términos exponenciales. Donde n_1 representa el número de raíces reales y n_2 el número de raíces complejas conjugadas de la ecuación característica de la Ecuación Diferencial Ordinaria.

Mediante un proceso de filtrado se separan las componentes de la respuesta transitoria. Se considera que las componentes pueden tener dos formas cualitativas: exponenciales reales de la forma ae^{-rt} y exponenciales complejas, es decir, senoidales con amortiguamiento exponencial de la forma $ae^{-rt} \text{sen}(\omega t + \phi)$.

Mediante un estudio de las funciones exponenciales se tienen las siguientes conclusiones:

$$E_{n_1}(t) = \sum_{i=1}^{n_1} a_i e^{-r_i t} \quad (2.1)$$

$$ES_{n_2}(t) = \sum_{j=1}^{n_2} a_j e^{-r_j t} \text{sen}(\omega_j t + \phi_j) \quad (2.2)$$

Donde:

$E_{n_1}(t)$ es la suma de n_1 componentes exponenciales y

$ES_{n_2}(t)$ es la suma de n_2 componentes senoidales con amortiguamiento exponencial.

Los parámetros que definen la forma cualitativa de $E_{n_1}(t)$ son los coeficientes y los

exponentes. Es decir, la forma cualitativa de $E_{n_1}(t)$ depende del orden de magnitud de las relaciones entre sus coeficientes y exponentes. Por otro lado, el parámetro más importante para determinar la forma cualitativa y la distinción de $ES_{n_2}(t)$ es la frecuencia, además del orden de magnitud de las relaciones de los coeficientes y los exponentes.

El primer paso en el proceso de identificación es detectar los extremos [Bronshtein77] en la respuesta transitoria del sistema que se desea identificar mediante el siguiente razonamiento :

- Una función cuya derivada cambia de signo entre dos puntos consecutivos pasa a través de un punto de cambio cuya derivada es cero.
- Los puntos en los cuales cambia el signo de la derivada se marcan como un máximo o un mínimo. Si la transición de signo fue de “-” a “+” el punto es un mínimo, si por el contrario la transición fue de “+” a “-” el punto es un máximo.
- Una función que cambia su signo entre dos puntos consecutivos pasa a través de un punto relevante cuya ordenada es cero.

Estos puntos se marcan como puntos importantes en la forma cualitativa de la señal, la Figura 2.1 muestra la representación cualitativa de la derivada usada en el algoritmo y la Figura 2.2 la detección de los puntos de cambio en una señal.

Se puede apreciar que la forma como se detectan estos puntos es mediante la información cualitativa que proporciona la derivada de la señal en cada instante de tiempo, es decir, no es importante saber cuanto vale, lo que es importante es conocer su signo. Esta aproximación submuestra la señal original y genera un nuevo conjunto de datos definido por los puntos marcados en la señal como se ilustra en la Figura 2.2.

Los puntos de cambio detectados mediante el razonamiento ilustrado en la Figura 2.2 se clasifican en dos conjuntos, uno para los puntos máximos y otro para los puntos

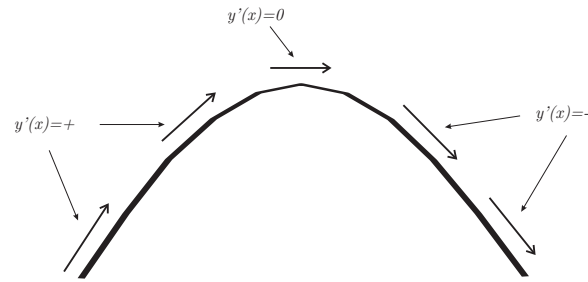


Figura 2.1: Representación de la derivada de una función continua desde el enfoque del álgebra de signos

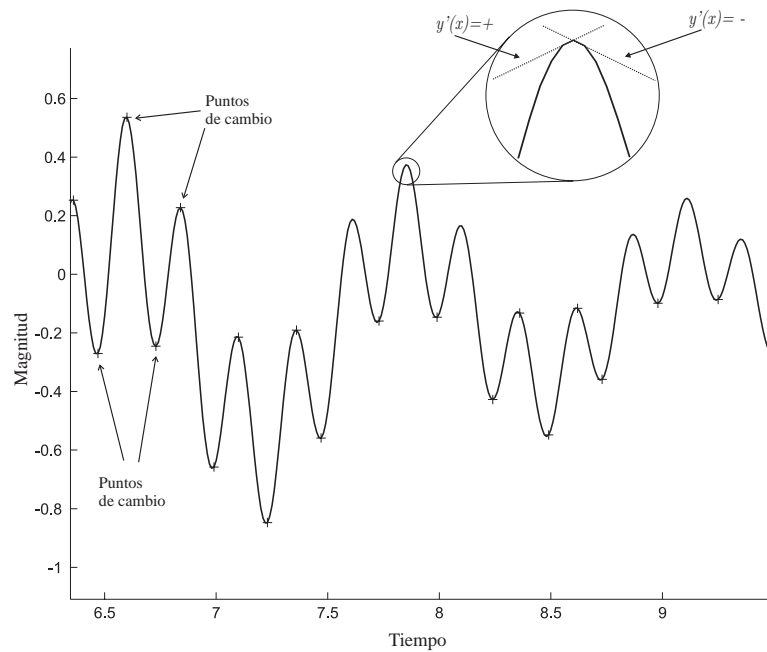


Figura 2.2: Razonamiento mediante el cual se detectan los puntos de cambio en la serie de tiempo

mínimos. Esta clasificación permite la identificación de una envolvente. La Figura 2.3 ilustra por medio de líneas discontinuas la envolvente de la señal que se desea filtrar. El valor promedio de la envolvente genera un vector de puntos medios mediante el cual se calcula el residuo de la señal.

La aplicación sucesiva de este razonamiento permite detectar las componentes oscilatorias presentes en la señal, e ir las sustrayendo. Cada componente oscilatoria se

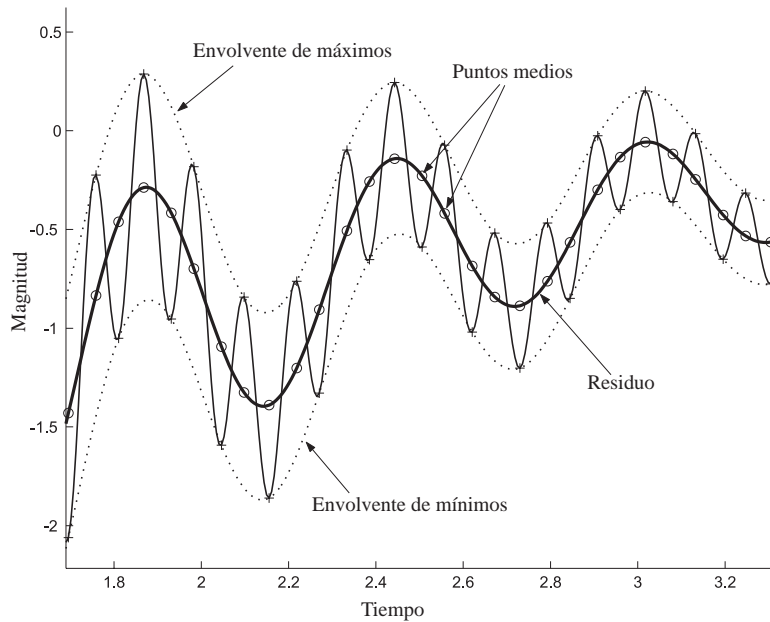


Figura 2.3: Clasificación de los puntos de cambio e identificación de la envolvente

expresa mediante la ecuación 2.3.

$$y = ae^{-rt} \text{sen}(\omega t + \phi) \quad (2.3)$$

donde:

a = Amplitud de la exponencial.

r = coeficiente de decaimiento.

ω = frecuencia angular

ϕ = angulo de defasamiento.

La forma cualitativa de cada una de las componentes oscilatorias, ya sustraídas de la respuesta transitoria, es como se presenta en la Figura 2.4. Cada una de estas componentes contribuye con un factor de segundo grado en la conformación de la ecuación característica de la ecuación diferencial que modela el sistema [Wylie95].

Este tipo de razonamiento es el que se usa en el diseño del algoritmo propuesto. Mediante estos sencillos razonamientos es posible determinar la estructura del modelo, incluyendo su orden.

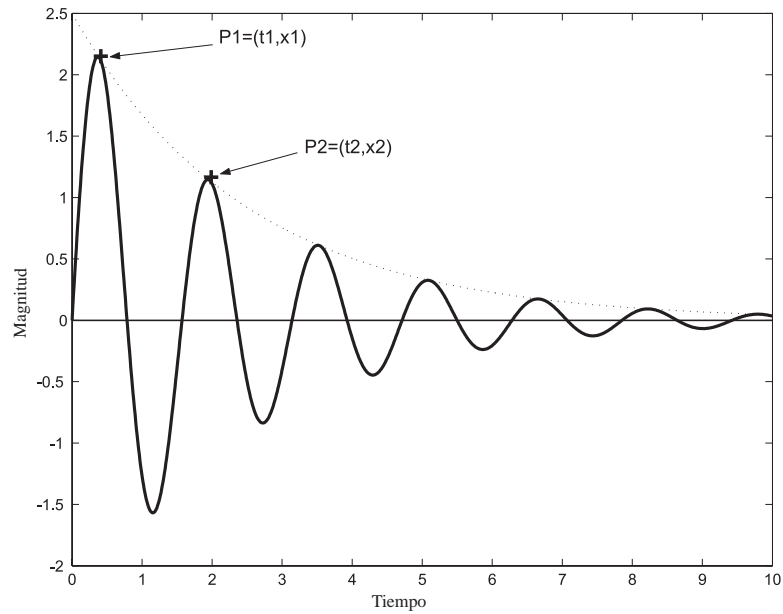


Figura 2.4: Forma cualitativa de las componentes senoidales

El refinamiento hacia los valores numéricos de los coeficientes de la EDO (identificación cualitativa) es posible una vez que conocemos el valor de los extremos y su distancia en el tiempo (Puntos P1 y P2 en la Figura 2.4).

Una vez que se han filtrado las componentes oscilatorias de la señal el residuo puede estar formado por una suma de exponenciales. El procedimiento para identificar y separar cada componente exponencial es muy similar al anterior, pero no se analiza la distancia en el tiempo de los máximos consecutivos por que no proporcionan información útil para este análisis. Es suficiente con la detección de los extremos para establecer por completo la estructura del modelo (ver Teorema 2).

El razonamiento que nos permite conocer la contribución de los componentes exponenciales reales a la estructura del modelo es el siguiente:

- El número de extremos (máximos y mínimos) de la señal determinan el número de términos de primer orden que se agregan a la EDO que describe al sistema.

Cada componente exponencial presente en la señal está determinada por la si-

guiente expresión

$$x_i(t) = a_i e^{-r_i t} \quad (2.4)$$

donde:

$$i = 1, 2, 3, \dots, n_1$$

Cada uno de estos componentes exponenciales contribuye con un factor de primer orden para la ecuación diferencial que modela al sistema.

En resumen se ha presentado la aplicación del razonamiento cualitativo en este trabajo. Se puede apreciar que mediante el razonamiento de las características cualitativas de la respuesta de un sistema se propone la estructura del modelo y el orden del mismo. Es decir, la identificación cualitativa. Los detalles para el desarrollo del algoritmo se presentan en los siguientes capítulos, de acuerdo con la organización presentada en la sección 1.6.

Capítulo 3

Modelado Matemático de Sistemas Dinámicos

Un modelo matemático de un sistema dinámico se puede definir como un conjunto de ecuaciones que representa la dinámica del sistema. Estos modelos no son únicos, es decir, un sistema puede representarse en muchas formas diferentes y por lo tanto tendrá muchos modelos matemáticos, esto dependerá de la perspectiva particular del enfoque.

3.1. Representación de Modelos por medio Ecuaciones Diferenciales

La dinámica de muchos sistemas continuos se describe en términos de ecuaciones diferenciales. Por medio de este tipo de ecuaciones podemos describirlos y hacer deducciones acerca de ellos. Estas ecuaciones se pueden obtener a partir de las leyes físicas que gobiernan al sistema. Podemos decir, entonces, que una ecuación diferencial representa la estructura de un sistema por medio de la selección de ciertas variables continuas que caracterizan el estado del sistema, y de ciertas restricciones

matemáticas sobre los valores que pueden tomar esas variables. Uno de los usos de la descripción por medio de ecuaciones diferenciales es para predecir el comportamiento de los sistemas en el tiempo, es decir, mediante un conjunto de funciones de tiempo se describe como evolucionan las variables del sistema a partir de un estado inicial. La Figura 3.1 [Kuipers94, p. 13] muestra los niveles de abstracción usados para el modelado cualitativo de sistemas propuesto por Kuipers; la parte superior representa la complejidad de los sistemas reales, la parte media representa la descripción del sistema y su comportamiento en términos de ecuaciones diferenciales y sus soluciones y, finalmente, la parte inferior representa una descripción más general del sistema mediante un modelo cualitativo y su correspondiente comportamiento cualitativo. En general se puede ver que una descripción mediante ecuaciones diferenciales no considera todos los aspectos que están presentes en el sistema real, y por lo tanto es una abstracción del sistema real, de igual manera se observa que un modelo cualitativo es la abstracción de la descripción dada por las Ecuaciones Diferenciales Ordinarias (EDO's).

Si el modelo en ecuación diferencial describe adecuadamente la estructura de un sistema físico, entonces, el comportamiento predicho por la solución de la ecuación diferencial describirá el comportamiento actual del sistema físico. Es decir, la parte superior y media de la Figura 3.1 conducen al mismo resultado. Un modelo con esta propiedad es de mucha utilidad porque se puede usar para predecir aspectos del comportamiento de un sistema físico, con la confianza de que el sistema real se comporta de la misma forma.

3.1.1. Abstracción de los Valores Numéricos a los Valores Simbólicos

Se pueden escribir modelos con ecuaciones diferenciales que describan sistemas físicos, incluyendo los valores numéricos de las constantes de la ecuación diferencial.

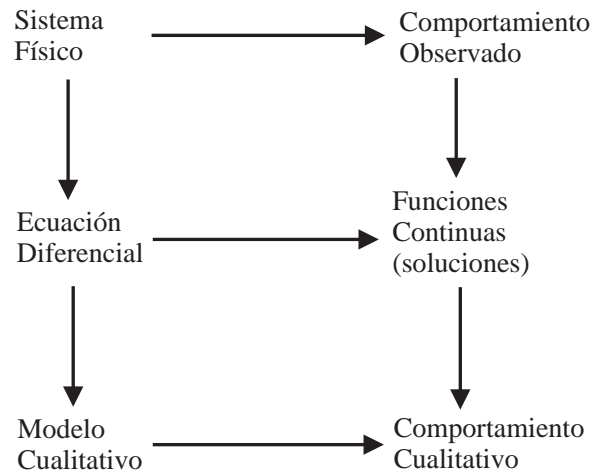


Figura 3.1: Representación de los niveles de abstracción de un sistema físico

Esta ecuación se puede usar para predecir el estado del sistema en instantes futuros o también para simularlo numéricamente, mediante su solución analítica. Sin embargo también se pueden obtener buenos resultados proporcionando menos información y usando un modelo más generalizado. Aplicando las leyes de la física cualitativa podemos reemplazar las constantes numéricas por los nombres simbólicos, entonces, la ecuación que se obtiene representa un estado relativo de conocimiento incompleto. Y como resultado, se tiene ahora un modelo más general, aplicable a un amplio rango de condiciones diferentes (parte inferior de la Figura 3.1).

3.1.2. Modelado e Identificación de Sistemas

El término modelado es manejado en forma muy general, como el método para obtener un modelo. La Figura 3.2 muestra las técnicas de modelado y la caracterización de los modelos respectivamente [Rodriguez94].

El modelado está dividido en modelado teórico y en modelado experimental (Figura 3.2 a). El primero se refiere al desarrollo de modelos a partir de principios y leyes físicas, por el contrario el modelado experimental se refiere al desarrollo de modelos a partir de resultados experimentales; es decir, se refiere a la identificación de sistemas

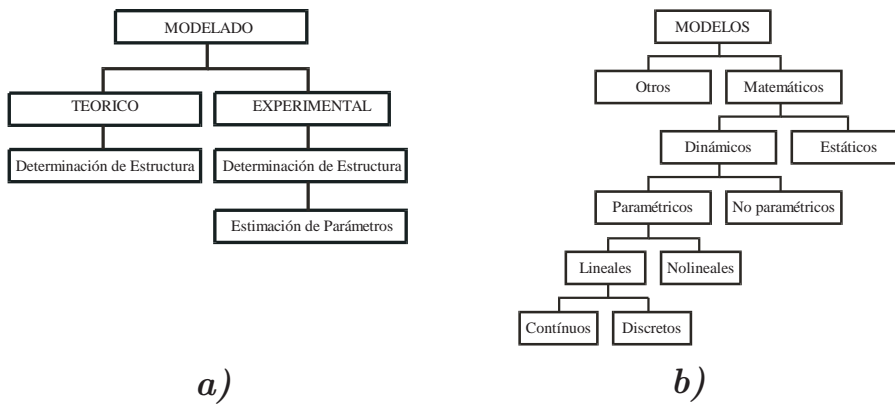


Figura 3.2: Modelado y caracterización. *a)* Tipo de modelado. *b)* Caracterización de los modelos

[Ljung99]. Por otro lado la caracterización de los modelos (Figura 3.2 *b)* se divide en dos partes: modelos matemáticos y otros tipos de modelos, tales como los modelos mentales, tablas, gráficas, etc.

Por su lado el modelado matemático se subdivide en modelos dinámicos y estáticos. Los modelos dinámicos son aquellos en los que la salida en un instante de tiempo depende de la entrada aplicada en ese mismo instante e instantes anteriores. Estos sistemas son conocidos también como sistemas con memoria y se representan por ecuaciones diferenciales o en diferencias. Los modelos estáticos son aquellos en los cuales la salida producida en un tiempo determinado depende en forma única de la entrada aplicada en ese mismo instante. Estos sistemas se conocen como sistemas algebraicos o sistemas sin memoria y se representan mediante ecuaciones algebraicas.

Los modelos lineales son aquellos que son representados por ecuaciones diferenciales ordinarias lineales y que cumplen además con el principio de superposición. Los sistemas lineales se pueden dividir considerando las características que poseen las variables que intervienen en los modelos matemáticos empleados para describir su comportamiento, de esta manera los podemos clasificar, además, en continuos y discretos. En los modelos continuos las variables que intervienen en sus modelos matemáticos son funciones continuas en el tiempo. Es decir, la variable tiempo puede

tomar todos los valores del conjunto de los números reales, y se representan mediante ecuaciones diferenciales ordinarias lineales. En los sistemas discretos las variables que intervienen son funciones de tiempo discreto. Es decir, la variable tiempo se puede especificar por una secuencia de muestras instantáneas espaciadas T_s segundos. T_s se conoce como el periodo de muestreo y la frecuencia de muestreo se define como se muestra en la siguiente ecuación:

$$\omega_s = \frac{2\pi}{T_s} \quad (3.1)$$

es importante considerar que para que esta secuencia de muestras describa completamente al sistema debe cumplir con el teorema de Nyquist, ver apéndice B.

La Figura 3.2 a) presenta las dos formas mediante las cuales se puede obtener un modelo. Cuando conocemos las características y propiedades del sistema es posible construir el modelo mediante el modelado teórico, pero cuando sólo se dispone de un conjunto de mediciones entonces la única forma de llevar a cabo el proceso de modelado es mediante la identificación de sistemas.

La identificación de sistemas puede entenderse como la construcción de modelos paramétricos a partir de datos experimentales, por medio del muestreo de secuencias de entrada-salida, conduciendo a la estimación de los parámetros que describan en una forma óptima al sistema. En la determinación del modelo matemático, es deseable obtener un modelo simple que satisfaga los objetivos específicos del análisis y del diseño. Es decir, dentro de una determinada clase de modelos, será seleccionado el modelo con menor número de parámetros que satisfaga la dinámica del sistema.

En la obtención de la descripción paramétrica de un sistema, se requiere un conocimiento razonable tanto del sistema como de las perturbaciones que la afectan. En la formulación matemática del problema de identificación, se debe introducir una función de costo o índice de desempeño, permitiendo con esto determinar la “aptitud” del modelo con respecto a los datos experimentales. El modelo óptimo es elegido como aquel que maximice el índice de desempeño o bien, que sea mínimo el criterio de error.

La función de costo, generalmente está determinada por la estructura propuesta para el modelo.

En términos generales, el proceso de identificación es realizado por medio de la excitación del sistema y observando sus entradas y salidas en un intervalo de tiempo. Para procesos de identificación fuera de línea es muy común almacenar la información en un archivo de computadora, para posteriormente procesarla. Se trata entonces de ajustar los datos almacenados a un modelo paramétrico del sistema.

El primer paso es determinar una forma apropiada para el modelo, típicamente una ecuación diferencial ordinaria. Como segundo paso se usa algún método para estimar los parámetros desconocidos del modelo (coeficientes en la ecuación diferencial). En la práctica la estimación de los parámetros se lleva a cabo en forma iterativa. Esto significa que para la estructura propuesta se estiman unos parámetros y el modelo obtenido se prueba para ver que tan adecuado es para la representación del sistema. Si no es el caso entonces se estima nuevamente la estructura y los parámetros hasta lograr una representación lo suficientemente buena. La Figura 3.3 muestra el diagrama de bloques del proceso clásico de identificación de sistemas dinámicos [Ljung99]. Un proceso de identificación típico cuantifica repetidamente la estructura de modelo, calculando el mejor modelo en la estructura propuesta, y evalúa las propiedades de este modelo para ver si son satisfactorias. Este ciclo puede ser ordenado de la siguiente manera:

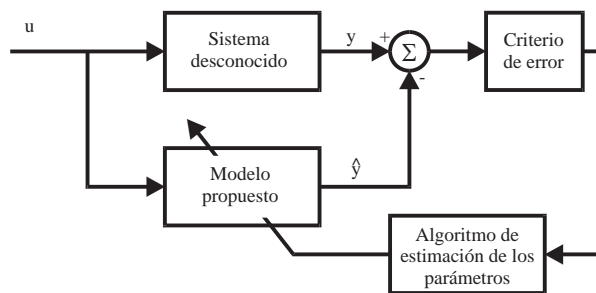


Figura 3.3: Proceso típico para realizar la identificación de sistemas

1. Diseñar el experimento y en su caso coleccionar los datos de entrada, u , y salida, y , (ver Figura 3.3) del sistema que se va a identificar.
2. Seleccionar y definir una estructura dentro de la cual se haga la búsqueda (modelo propuesto).
3. Calcular el mejor modelo dentro de la estructura propuesta de acuerdo a los datos de entrada-salida y a un criterio de error. Estimar valores para los parámetros de la estructura propuesta en el paso anterior.
4. Examinar las propiedades del modelo obtenido. El modelo propuesto se simula aplicando la entrada u y su salida \hat{y} se compara con la salida y del sistema que se está observando. Se mide la diferencia entre y y \hat{y} para ver si se alcanza el criterio de error.
5. Si el modelo es suficientemente bueno, entonces termina el proceso de identificación; en caso contrario regresa al paso 2 ó al paso 3, según se desee, para tratar con otra estructura, o bien, para recalcular el modelo utilizando los datos obtenidos en la iteración anterior.

La identificación de sistemas dinámicos es un proceso que permite construir modelos matemáticos de sistemas dinámicos basándose en mediciones de los datos de entrada y salida del sistema. Esencialmente, esto se lleva a cabo mediante el ajuste de los parámetros de un modelo dado hasta que la salida de este modelo ajusta, lo más cercanamente posible, a los datos de salida medidos. Una buena forma de validación es simular el modelo usando datos de entrada que no hayan sido usados en el proceso de identificación, es decir, probar la capacidad de predicción del modelo estimado, y compararlo con los datos observados.

3.1.3. Elementos Básicos en la Identificación de Sistemas

El procedimiento de identificación (ver Figura 3.3) involucra tres elementos básicos [Ljung99]:

1. El conjunto de datos de entrada-salida.
2. El conjunto de modelos candidatos (estructura propuesta del modelo).
3. Un criterio para seleccionar un modelo en particular (criterio de error).

Los datos

Los datos de entrada-salida generalmente son observados durante un experimento específico de identificación, donde el usuario puede determinar cuales señales serán medidas y cuando medirlas. El propósito del diseño del experimento es garantizar que los datos proporcionen la máxima información.

El conjunto de Modelos

La elección de la clase de modelo es fundamental y es a lo que se le llama caracterización [Alvarez83]. La caracterización es la etapa cualitativa, donde se decide qué forma dar al modelo. En esta etapa se formulan hipótesis sobre la naturaleza del sistema.

Un conjunto de modelos candidatos se obtienen de la especificación del tipo de estructura donde se realizará la búsqueda. Esto sin duda es lo más importante y, al mismo tiempo, el paso más difícil dentro del proceso de identificación. Es el paso más importante por que aquí se define la estructura del modelo y el más difícil por que para plantear la estructura se requiere una buena cantidad de conocimiento a priori de tal forma que esta sea suficiente para plantear una estructura adecuada.

Criterio de Error

Es la expresión matemática para evaluar el desempeño del modelo cuando se trata de reproducir los datos medidos y se expresa por medio de un índice de funcionamiento, ver Figura 3.3. El criterio de error es la expresión matemática de la diferencia entre el modelo propuesto y el sistema que se está identificando. Los tipos de criterios utilizados son los siguientes [Ljung99]:

- Distancia de estado.- Este error se define como la diferencia que existe entre la salida del sistema y la salida del modelo propuesto, cuando se aplica la misma entrada $u(t)$ a ambos.
- Error de predicción.- Está definido como la diferencia entre la salida estimada $\hat{y}(t)$ y la salida del sistema $y_s(t)$. Aquí la salida estimada \hat{y} depende de los parámetros estimados y de la salida y_s en el instante actual e instantes anteriores.

La diferencia entre los dos conceptos anteriores radica en el conjunto de datos que se usa para simular el modelo, mientras que en la distancia de estado se usa el mismo conjunto de datos en el error de predicción se espera que los datos correspondan a una ventana de tiempo diferente a la usada en el proceso de identificación.

Una vez que el tipo de modelo y el criterio de error son elegidos, queda el problema de minimizar el error. El problema de minimizar el error puede ser considerado como un problema de optimización paramétrica, o bien, como un problema estadístico, dado que las mediciones son perturbadas por ruido [Ljung99, Stoica89].

La optimización paramétrica consiste en encontrar un vector finito de parámetros $\theta = \{\theta_1, \theta_2, \dots, \theta_n\}$ que pueda, de alguna manera, ser definido como óptimo. En un caso simple es la maximización o minimización de alguna característica del sistema dependiente de θ , por ejemplo, la diferencia de la salida del modelo con respecto a la salida del sistema.

Por otro lado, los métodos estadísticos generalmente utilizan el error de predicción y se basan en la hipótesis de que las medidas y , del proceso, son funciones de los parámetros y de variables deterministas u (entradas), alteradas por un vector de ruido. Las mediciones se consideran como variables aleatorias y sus propiedades están determinadas por las propiedades estadísticas del ruido del sistema. El problema consiste en obtener una estimación de θ usando las medidas y y u .

Existen muchas aplicaciones para los modelos desarrollados. En algunas ocasiones se necesita un modelo típico de algún proceso del mundo real, la aplicación más común es para diseño de sistemas de control. Los modelos también son usados para hacer estimadores de estado o filtros de Kalman que pueden ser usados en el filtrado de datos y en el control retroalimentado ó también para diagnóstico y predicción de desempeño.

3.2. Sistemas Lineales

Un sistema lineal, en tiempo continuo o discreto, es aquel que cumple con dos propiedades matemáticas: homogeneidad y superposición. Si se puede mostrar que un sistema cumple con ambas propiedades, entonces se tiene la prueba de que es lineal [Rugh96].

La propiedad de homogeneidad o escalamiento significa que un cambio en la amplitud de la señal de entrada resultará en un cambio proporcional en la amplitud de la señal de salida. Por otro lado, la propiedad de superposición considera que si una entrada consiste de la suma ponderada de varias señales, entonces la salida es sólo la superposición, es decir, la suma ponderada de las respuestas del sistema a cada una de las señales. Matemáticamente, sea $y_1(t)$ la respuesta del sistema de tiempo continuo a la señal $x_1(t)$, y $y_2(t)$ sea la salida correspondiente a la entrada $x_2(t)$. Decimos entonces que un sistema es lineal si cumple con las propiedades de homogeneidad y superposición. Las dos propiedades que definen un sistema lineal pueden combinarse

en un sólo enunciado, el cual está descrito por la ecuación 3.2.

$$ax_1(t) + bx_2(t) = ay_1(t) + by_2(t) \quad (3.2)$$

donde a y b son constantes cualquiera.

Más aún, se puede mostrar que si $x_k(t)$, $k = 1, 2, 3, \dots$, son un conjunto de entradas a un sistema lineal con las correspondientes salidas $y_k(t)$, $k = 1, 2, 3, \dots$, entonces la respuesta a una combinación lineal de esas entradas dada por

$$x(t) = \sum_k a_k x_k(t) = a_1 x_1(t) + a_2 x_2(t) + \dots \quad (3.3)$$

es

$$y(t) = \sum_k a_k y_k(t) = a_1 y_1(t) + a_2 y_2(t) + \dots \quad (3.4)$$

3.3. Sistemas Lineales e Invariantes en el Tiempo

Una clase de sistemas dinámicos de gran importancia es aquella para la cual la entrada y la salida están relacionadas a través de una ecuación diferencial lineal de coeficientes constantes. Las ecuaciones de este tipo aparecen en una amplia variedad de sistemas y fenómenos físicos. La respuesta de un circuito RLC se puede describir en términos de una ecuación diferencial, así como la respuesta de un sistema mecánico que contiene fuerzas de restauración y de amortiguamiento. También la cinética de una reacción química y la cinemática del movimiento de un objeto o vehículo son descritas, por lo general, por ecuaciones diferenciales. Los sistemas que tienen características similares a los ejemplos anteriores se denominan sistemas lineales invariantes en el tiempo.

3.3.1. Solución de Ecuaciones Diferenciales Ordinarias con Coeficientes Constantes

Una ecuación diferencial lineal de orden n tiene la forma general:

$$c_n(t) \frac{d^n x(t)}{dt^n} + c_{n-1}(t) \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + c_1(t) \frac{dx(t)}{dt} + c_0(t) x(t) = f(t) \quad (3.5)$$

Si en la ecuación 3.5 $f(t) = 0$, entonces la ecuación diferencial es homogénea; en caso contrario, se llama no homogénea. Se le llama lineal porque es una combinación de la variable independiente y sus derivadas, la solución cumple con la ecuación 3.2. La solución de una ecuación diferencial ordinaria (EDO) es un conjunto de funciones $f(x)$ las cuales, cuando se sustituyen por la variable dependiente y y sus respectivas derivadas en la ecuación diferencial, reducen la ecuación diferencial a una identidad en función de la variable independiente. La solución exponencial de una ecuación diferencial es de la forma $y = Ce^{rt}$, donde C denota cualquier número real. Y e^{rt} representa la solución general de esa ecuación diferencial. Una solución particular es cualquier solución que se obtenga dando valores específicos a las constantes arbitrarias de la solución general.

Geoméricamente, la solución general de una ecuación diferencial representa una familia de curvas, llamadas curvas solución, una para cada valor asignado a la constante arbitraria. Las soluciones particulares de una ecuación diferencial se obtienen a partir de las condiciones iniciales que dan el valor de la variable dependiente o de alguna de sus derivadas para un valor particular de la variable independiente. El término “condiciones iniciales” proviene de que, con frecuencia, en problemas donde interviene el tiempo, se conoce el valor de la variable dependiente o de alguna de sus derivadas en el instante inicial $t = 0$.

Cuando los coeficientes c_i son constantes, es decir, no son funciones de t ; la ecuación se convierte en una ecuación diferencial ordinaria homogénea de coeficientes constantes, y una de sus aplicaciones es para representar sistemas lineales invariantes en el tiempo.

Si sustituimos el operador de derivación, $D = \frac{d}{dt}$, y consideramos el caso homogéneo en la ecuación 3.5 se obtiene la siguiente expresión:

$$(c_n D^n + c_{n-1} D^{n-1} + \dots + c_1 D + c_0) x(t) = 0 \quad (3.6)$$

Si los coeficientes c_i son constantes, la ecuación toma la forma de un polinomio de grado n con respecto al operador D .

$$c_n D^n + c_{n-1} D^{n-1} + \dots + c_1 D + c_0 = 0 \quad (3.7)$$

A la ecuación 3.7 se le conoce como la ecuación característica.

3.3.2. Respuesta Natural de los Sistemas Dinámicos

Si un sistema dinámico da lugar a una ecuación diferencial como la presentada en la ecuación 3.5 con $f(t) = 0$, la solución de esta ecuación representará la respuesta natural del sistema, llamada así por que está dada solamente por la dinámica de los componentes internos del sistema. Es decir, el sistema está libre de la influencia de fuentes o excitaciones externas. Si este sistema estuviera sujeto a la acción de alguna fuente externa su solución sería la respuesta forzada y estaría caracterizado por la ecuación no homogénea 3.5, es decir, $f(t) \neq 0$.

Supongamos que tenemos un sistema dinámico sin ninguna fuente activa para valores positivos del tiempo. Se desea obtener una expresión de las variables, como funciones del tiempo para todo $t > 0$. Con este fin escribiremos primero la ecuación de equilibrio correspondiente al sistema para $t > 0$. Estas ecuaciones son, en general, de la forma

$$c_n \frac{d^n x(t)}{dt^n} + c_{n-1} \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + c_1 \frac{dx(t)}{dt} + c_0 x(t) = 0 \quad (3.8)$$

Como se menciona con anterioridad, los sistemas dinámicos sin ninguna fuente ó función de excitación dan lugar a ecuaciones diferenciales homogéneas y su solución se conoce como respuesta libre o respuesta natural del sistema. Como ejemplo

analizaremos el caso más simple dado por la ecuación 3.9

$$\frac{dx(t)}{dt} + r x(t) = 0, \quad \text{para } t > 0 \quad (3.9)$$

que puede resolverse de la siguiente forma. Primero debemos buscar una función $x(t)$ tal que su derivada, sumada a ella misma (y multiplicada por una constante), sea nula para todo valor de t . Esto quiere decir que la función $x(t)$ y su derivada $\frac{dx(t)}{dt}$ deben tener la misma forma de onda. En tal caso se propone la función

$$x(t) = a e^{mt} \quad (3.10)$$

con a y m constantes, su derivada es

$$\frac{dx(t)}{dt} = m a e^{mt} \quad (3.11)$$

Sustituyendo las ecuaciones 3.10 y 3.11 en 3.9 se obtiene

$$mae^{mt} + rae^{mt} = 0 \quad (3.12)$$

$$m + r = 0 \quad (3.13)$$

$$m = -r \quad (3.14)$$

y la solución es

$$x(t) = ae^{-rt} \quad (3.15)$$

de esta manera conocemos la expresión matemática general de la respuesta; estando el valor de r determinado por los parámetros del sistema.

La solución a un sistema libre de fuentes se llama respuesta natural por que así es como se comporta el sistema naturalmente debido a su dinámica interna, sin estar forzado por alguna fuerza externa, exhibiendo un comportamiento debido solamente a sus condiciones iniciales. El sistema se comporta de esta forma debido a su propia estructura, puesto que no hay fuentes conectadas que lo exciten.

3.3.3. Solución General

Para hallar la solución general es necesario encontrar las raíces r_1, r_2, \dots, r_n de la ecuación 3.7. A cada raíz r_i corresponde una solución $e^{r_i t}$. Las n raíces de la ecuación 3.7 pueden estar dentro de tres posibilidades; reales distintas, reales iguales o complejas.

Si las raíces son reales y distintas, la solución de la ecuación 3.5 es de la forma:

$$y = a_1 e^{r_1 t} + a_2 e^{r_2 t} + \dots + a_n e^{r_n t} \quad (3.16)$$

Si las raíces son reales e iguales, la solución tiene la forma:

$$y = e^{rt} (a_1 + a_2 t + a_3 t^2 + \dots + a_n t^{n-1}) \quad (3.17)$$

Y si las raíces son reales y de ellas unas son iguales y otras, diferentes, se usan las dos leyes anteriores según el caso; así supongamos 6 raíces:

$$r_1 \neq r_2 = r_3 \neq r_4 = r_5 = r_6 \quad (3.18)$$

entonces la solución es:

$$y = a_1 e^{r_1 t} + a_2 e^{r_2 t} + a_3 t e^{r_3 t} + a_4 e^{r_4 t} + a_5 t e^{r_5 t} + a_6 t^2 e^{r_6 t} \quad (3.19)$$

Si las raíces son complejas, estarán dadas en pares conjugados, es decir, $m = r \pm j\omega$.

Y para cada par conjugado, la solución es:

$$y = e^{rt} (A \cos \omega t + B \operatorname{sen} \omega t) \quad (3.20)$$

Si hay otro par igual entonces:

$$y = e^{rt} t (A \cos \omega t + B \operatorname{sen} \omega t) \quad (3.21)$$

es solución, y así sucesivamente.

Las expresiones obtenidas de la forma $A \cos \omega t + B \operatorname{sen} \omega t$ también pueden representarse en la forma $a \operatorname{sen} (\omega t + \varphi)$ [Bronshtein77, p. 212,213] y la ecuación 3.20 toma la forma

$$y = a e^{rt} \operatorname{sen} (\omega t + \varphi) \quad (3.22)$$

donde

$$a = \sqrt{A^2 + B^2} \quad (3.23)$$

$$\varphi = \operatorname{arc\,tg} \left(\frac{B}{A} \right) \quad (3.24)$$

En la teoría de las oscilaciones y entre otras aplicaciones, frecuentemente, es importante establecer que si las partes reales de todas las raíces de la ecuación característica resultan negativas, cualquier solución de una ecuación diferencial lineal homogénea de coeficientes constantes tiende a cero cuando $t \rightarrow +\infty$. Desde el punto de vista de sistemas de control las raíces se deberán encontrar en el semiplano izquierdo del lugar de las raíces, es decir, se trata de sistemas estables.

Una vez que podemos representar el comportamiento de este tipo de sistemas en términos de componentes exponenciales y senoidales en su respuesta podemos hacer las siguientes definiciones:

$$E_{n_1}(t) = \sum_{1 \leq i \leq n_1} a_i e^{-r_i t} \quad (3.25)$$

$$ES_{n_2}(t) = \sum_{1 \leq i \leq n_2} a_i e^{-r_i t} \operatorname{sen}(\omega_i t + \varphi) \quad (3.26)$$

la ecuación 3.25 representa una suma con a lo más n_1 términos exponenciales, y la ecuación 3.26 representa una suma de n_2 funciones senoidales amortiguadas.

Las definiciones anteriores permiten dar una descripción cualitativa del comportamiento de las ecuaciones diferenciales partiendo de las componentes exponenciales y senoidales que la conforman, y nos llevan al considerar el teorema 1 planteado por Ray Wylie en [Wylie95].

Teorema 1 *Dado un sistema lineal, invariante en el tiempo, de orden n , la respuesta puede ser expresada como en la ecuación 3.27.*

$$X(t) = E_{n_1}(t) + ES_{n_2}(t) \quad (3.27)$$

donde $n_1 + 2n_2 = n$.

Este resultado es evidente de la definición de las ecuaciones 3.25 y 3.26, que se desprenden de la teoría de los sistemas lineales [Wylie95, Rugh96].

Se puede ver que si el segundo término de la ecuación 3.27 no existe, la respuesta será acíclica. En otro caso es una onda senoidal, donde $En_1(t)$ representa su eje o atractor, y $ESn_2(t)$ representa las componentes oscilatorias amortiguadas.

Si agregamos una función de excitación, la respuesta del sistema se puede descomponer en la respuesta natural, es decir, la solución de la ecuación homogénea, y la respuesta forzada. Si restringimos la función de excitación a la forma $e^{\alpha t} \sin \beta t$, la respuesta forzada siempre tendrá la misma forma cualitativa de la excitación. Esto garantiza la forma cualitativa de la respuesta y sólo le agrega términos exponenciales o senoidales a la misma.

A continuación se analiza la forma cualitativa de la respuesta expresada en la ecuación 3.27. Esta forma cualitativa puede derivarse de la forma cualitativa de los componentes exponenciales y senoidales.

3.3.4. Componentes Exponenciales

El comportamiento cualitativo de la parte exponencial de la respuesta está caracterizado por el teorema 2, propuesto y demostrado en [Flores02 et al].

Teorema 2 $X_1(t) = E_{n_1}(t) = \sum_{1 \leq i \leq n_1} a_i e^{-r_i t}$ exhibe a lo más n_1 extremos (máximos y mínimos), incluyendo uno cuando $t \rightarrow \infty$.

Supondremos, sin perder generalidad, que $r_1 > r_2 > \dots > r_n$. Este teorema es equivalente a decir que la derivada $X_1'(t)$ tiene a lo más n_1 ceros diferentes. Esto es,

$$X_1'(t) = -r_1 a_1 e^{-r_1 t} - \dots - r_n a_n e^{-r_n t} = 0 \quad (3.28)$$

Haciendo el cambio de variable $z = e^{-t}$, la ecuación 3.28 queda como

$$r_1 a_1 z^{r_1} + \dots + r_n a_n z^{r_n} = 0 \quad (3.29)$$

La ecuación 3.29 es un polinomio en z de grado n_1 . Basándonos en el teorema de Descartes [Rees&Sparks91, Kurosusch77], se puede decir que el número de raíces de un polinomio está determinado por los cambios de signo en los coeficientes del polinomio. Si tenemos n_1 términos en el polinomio dado en la ecuación 3.28, tendremos a lo más $n_1 - 1$ cambios de signo, y por lo tanto $n_1 - 1$ raíces. Se tiene que considerar también que hay una raíz más cuando la función se hace cero, y esto ocurre cuando $t \rightarrow \infty$.

Dado que tenemos a lo más n_1 raíces para la ecuación 3.28, podemos por consiguiente tener a lo más n_1 extremos para $E_{n_1}(t)$.

La Figura 3.4 muestra una suma de tres exponenciales, donde se hacen evidentes los tres extremos, un máximo y dos mínimos (uno en infinito).

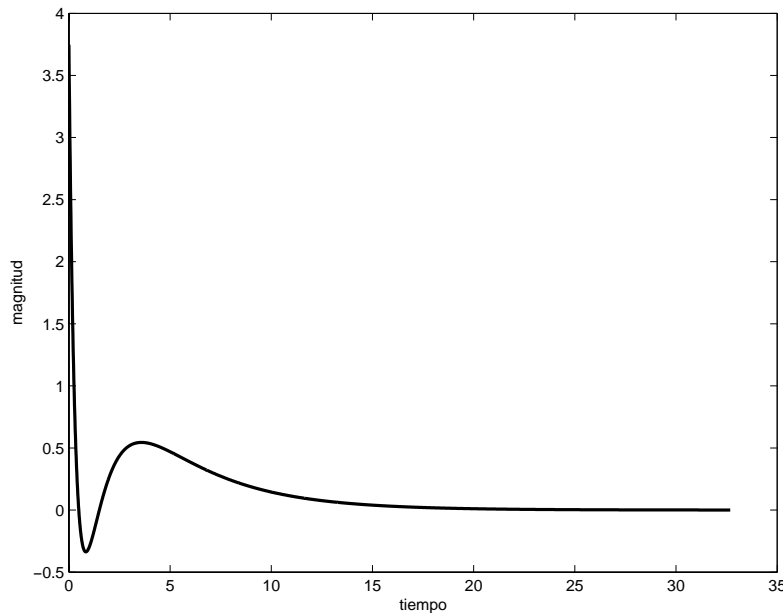


Figura 3.4: Suma de tres exponenciales

3.3.5. Componentes Senoidales

Si las frecuencias de los componentes senoidales de la ecuación 3.27 son iguales, sus formas se reducen a una sola. Si sus frecuencias son diferentes se podrán ver como

una senoidal rápida montada en una senoidal más lenta. La Figura 3.5 muestra una señal resultado de la suma de dos senoidales amortiguadas.

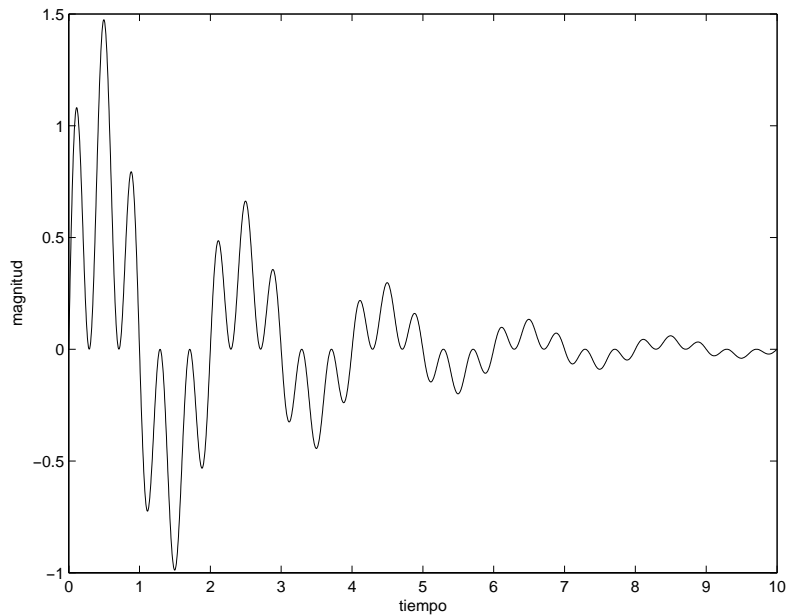


Figura 3.5: Suma de dos senoidales de frecuencias diferentes

Si las frecuencias de las senoidales son muy cercanas entre sí el resultado será una onda de resonancia. La Figura 3.6 muestra una señal de resonancia.

Estos resultados caracterizan completamente todas las posibles respuestas cualitativamente diferentes de un sistema dinámico lineal e invariante en el tiempo. En el siguiente capítulo se describe como usar esos resultados para producir un marco de trabajo para llevar a cabo la identificación de sistemas en los niveles tanto cualitativo como cuantitativo.

3.4. **Sistemas Lineales e Variantes en el Tiempo**

Los sistemas lineales variantes en el tiempo son aquellos sistemas que podemos modelar mediante EDO's, en los cuales los coeficientes de la EDO cambian con el

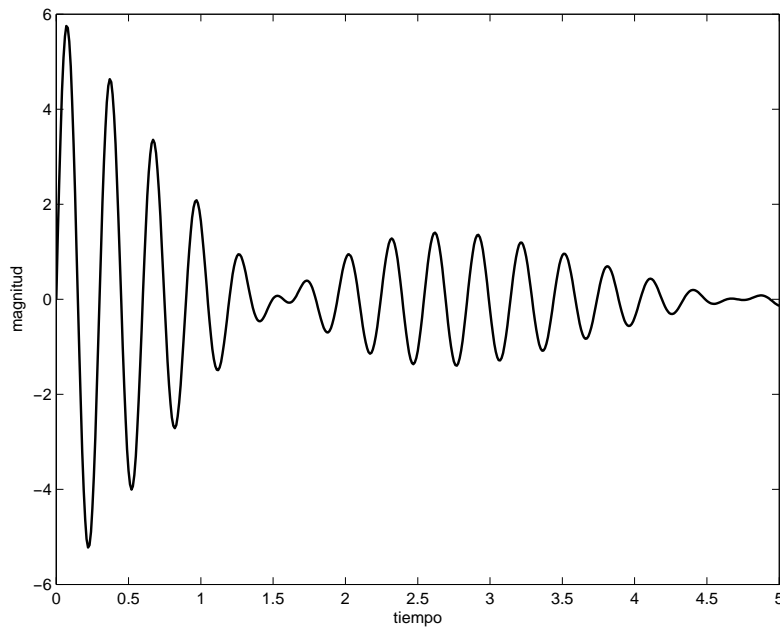


Figura 3.6: Resultado de la suma de dos senoidales con frecuencias cercanas entre sí

paso del tiempo. De esta forma los sistemas lineales variantes en el tiempo se pueden caracterizar por la ecuación 3.30.

$$c_n(t) \frac{d^n x(t)}{dt^n} + c_{n-1}(t) \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + c_1(t) \frac{dx(t)}{dt} + c_0(t) x(t) = f(t) \quad (3.30)$$

de igual manera que para el caso invariante si $f(t) = 0$, entonces la ecuación es homogénea en otro caso es no homogénea.

En la sección 3.3 se muestra que si $f(t)$ es continua y los coeficientes $c_i(t)$ son constantes, la ecuación 3.30 es tal que todas sus soluciones se pueden obtener a través de la asignación de valores a las constantes a_i y r_i en la fórmula de la solución general.

$$y = a_1 e^{r_1 t} + a_2 e^{r_2 t} + \dots + a_n e^{r_n t} + y_p(t) \quad (3.31)$$

donde

$$e^{r_1 t}, e^{r_2 t}, \dots, a_n e^{r_n t}$$

son las soluciones independientes asociadas a la ecuación diferencial homogénea y $y_p(t)$ es alguna solución particular de la ecuación no homogénea. En general se puede considerar que para encontrar las soluciones de los sistemas variantes en el tiempo, se pueden usar las estrategias y métodos de los sistemas lineales e invariantes en el tiempo siempre y cuando el sistema tenga una variación lenta en el tiempo, de tal forma que la respuesta transitoria sea mucho más rápida que esa variación. Es decir, consideramos a un sistema de variación lenta a aquel cuyos coeficientes se consideran prácticamente constantes durante el tiempo en que se observa la respuesta transitoria. Por ejemplo, consideremos al motor de una cámara fotográfica, donde los parámetros del modelo se deterioran con el tiempo y van cambiando, pero los podemos considerar como constantes mientras ocurre el transitorio producido por una excitación externa (por ejemplo, encendido o apagado del dispositivo). En el capítulo 5 se desarrolla un algoritmo de identificación de sistemas lineales lentamente variantes en el tiempo, en base a las propiedades, soluciones y métodos de identificación de sistemas lineales e invariantes en el tiempo.

3.5. Conclusiones

Es claro que con tales estrategias y métodos deben incluirse otras herramientas que lleven el análisis más allá de los fundamentos matemáticos para sistemas lineales invariantes en el tiempo.

Uno de los sistemas variantes en el tiempo más simple y más fácil de abordar es el de variación lenta cuyo comportamiento se puede ver como el de un sistema invariante durante un breve periodo de tiempo.

Los sistemas lentamente variantes en el tiempo son de gran importancia debido a que muchos de los sistemas prácticos son de variación lenta en el tiempo. Por ejemplo el efecto de las condiciones ambientales usualmente es más lento que la dinámica de los sistemas cuyos parámetros dependen del medio ambiente. Otro ejemplo es el efecto

de envejecimiento y el deterioro de componentes de un sistema dinámico.

En este capítulo se presentó la revisión de la representación de sistemas dinámicos por medio de ecuaciones diferenciales y los niveles de abstracción de los valores numéricos a los valores simbólicos. También se presentaron los tipos de modelos y de que manera se caracterizan. Se abordaron también las técnicas y métodos para el análisis de sistemas LIT y la solución general de las ecuaciones diferenciales. Finalmente, se presentaron los teoremas en los cuales se basa el desarrollo del algoritmo de identificación propuesto. El teorema 2 presentado en este capítulo es una aportación original y representan la base teórica de los algoritmos de identificación, tanto cualitativos como cuantitativos desarrollados en esta tesis.

Capítulo 4

Diseño del Algoritmo de Identificación

El algoritmo de identificación presentado, al cual denominaremos algoritmo de identificación cualitativa (*QSI*, por sus siglas en inglés de “Qualitative Systems Identification”), está basado en el hecho de que la respuesta de un sistema lineal e invariante en el tiempo se puede descomponer en una suma de términos exponenciales (reales y/o complejos). Si alguno de esos términos exponenciales son complejos, en cuyo caso serán pares complejos conjugados, cada uno de estos pares da lugar a una componente senoidal [Wylie95]. El algoritmo propuesto toma como entrada una serie de tiempo que representa la respuesta transitoria de un sistema dinámico lineal invariante en el tiempo.

4.1. Identificación de Sistemas Dinámicos

La identificación de sistemas permite construir modelos matemáticos de sistemas dinámicos basandose en la medición de datos y ajustando los parámetros de un modelo dado hasta que su respuesta coincide lo mejor posible con los datos medidos. El proceso de identificación se divide en dos partes: la primera parte se refiere a la

identificación estructural y es donde se determina la estructura del modelo. La segunda parte es la identificación paramétrica: en esta parte es donde se determinan los parámetros del modelo propuesto. Una característica importante en este trabajo es que el modelo resultante puede exhibir comportamientos de sistemas de orden mayor, esto es, el algoritmo identifica lo que observa y produce un modelo de menor orden capaz de reproducir la dinámica observada (ver los ejemplos de las secciones ??). Es decir, estamos haciendo modelado basado en la observación.

Dentro de las consideraciones que se suponen podemos mencionar las siguientes:

- El conjunto de datos observados que se adquiere cumple con el teorema de Nyquist, de tal forma que el número de datos es suficiente y contienen la información necesaria para representar la dinámica del sistema.
- El ruido inherente a los sistemas reales es tratado bajo dos consideraciones para eliminar las frecuencias que estén fuera del rango de interés: usando filtros analógicos o con filtros digitales. Estos filtros pueden ser diseñados, inclusive, como filtros pasa banda, de tal forma que también bloqueen las señales que estén por debajo de las frecuencias de interés, al mismo tiempo que se estén cancelando las frecuencias por encima de la banda de interés.

4.2. Algoritmo QSI

El algoritmo QSI estima la estructura del modelo en base a las propiedades de los sistemas lineales invariantes en el tiempo, analizando el comportamiento cualitativo del sistema que se va a identificar. Conforme se van filtrando las componentes de la respuesta se va haciendo también la estimación de los parámetros por medio de un análisis cuantitativo de cada una de esas componentes. La Figura 4.1 presenta el diagrama de bloques de este proceso de identificación.

El algoritmo es capaz de separar los términos de la ecuación 3.27 para determinar

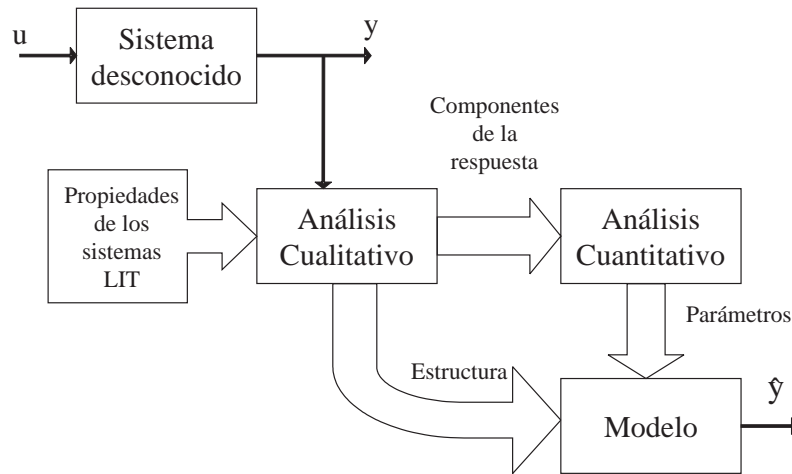


Figura 4.1: Diagrama de bloques del proceso de identificación usando QSI

la forma cualitativa y los coeficientes de la ecuación diferencial ordinaria que exhiba el comportamiento observado. Es decir, se lleva a cabo la identificación de sistemas.

La separación de los términos de la respuesta del sistema se lleva a cabo mediante un proceso de filtrado. El algoritmo 1 ilustra este proceso.

Algoritmo 1 Algoritmo QSI

```

QSI( $X$ )
1    $k \leftarrow 0$ 
2    $P \leftarrow 0$ 
3    $(X^*, nk, P) \leftarrow TPAFilter(X, nk, P)$ 
4    $(X, nk, P) \leftarrow ExpFilter(X^*, nk, P)$ 
5   regresar  $Modelo$ 

```

El algoritmo 1 requiere como entrada la respuesta transitoria del sistema que se desea identificar, analiza esta respuesta e identifica una componente a la vez. En la

linea 1 y 2 del algoritmo se inicializan a cero las variables que corresponden al orden del modelo y a la matriz de parámetros, respectivamente. Las dos funciones principales del algoritmo, *TPAFilter* y *ExpFilter*, son llamadas en las líneas 3 y 4. La función *TPAFilter* (ver la sección 4.2.1) identifica y filtra las componentes senoidales y la función *ExpFilter* identifica y filtra las componentes exponenciales. Ambas funciones requieren como entrada una serie de tiempo. Para el caso de la función *TPAFilter* la señal de entrada será la respuesta del sistema en estudio y en el caso de la función *ExpFilter* la señal de entrada será la señal ya procesada en *TPAFilter*. Es decir, el residuo una vez que se han filtrado las componentes senoidales. Las variables de salida de ambas funciones son: el orden del modelo, los parámetros de cada una de las componentes eliminadas y la señal residuo. Otras tres funciones que se incluyen en el algoritmo QSI son el cálculo de las condiciones iniciales del sistema y la validación del modelo por medio del cálculo del coeficiente de correlación numérico y el coeficiente cualitativo de correlación.

La determinación del orden del modelo se basa en sumar el orden de cada uno de los componentes que se van filtrando; las componentes senoidales contribuyen con dos unidades por cada componente filtrada y las componentes exponenciales contribuyen con una unidad por cada componente. La señal residuo es una serie de tiempo que contiene la respuesta del sistema menos la componente que se ha filtrado. Los parámetros de cada componente se ordenan en una matriz con la estructura que se presenta en la tabla 4.1:

Tabla 4.1: Estructura de la matriz de parámetros

Coefficiente a	Factor de decaimiento	frecuencia	angulo de defasamiento
a_1	r_1	ω_1	ϕ_1
\vdots	\vdots	\vdots	\vdots
a_{n_2}	r_{n_2}	ω_{n_2}	ϕ_{n_2}
a_{n_2+1}	r_{n_2+1}	0	0
\vdots	\vdots	\vdots	\vdots
$a_{n_2+n_1}$	$r_{n_2+n_1}$	0	0

Los renglones de la matriz de parámetros corresponden a cada componente identificada y las columnas a los parámetros de cada una de las componentes. Nótese que a partir del renglón $n_2 + 1$ los parámetros corresponden a las componentes exponenciales, y es esa la razón por la cual la columna de la frecuencia y el ángulo de defasamiento son igual a cero.

La función *TPAFilter* elimina una a una las componentes senoidales, iniciando por la de mayor frecuencia; el residuo $X^*(t)$ contendrá la suma de todos los componentes excepto el componente eliminado. Después de filtrar el j -ésimo componente senoidal, el residuo es

$$X_j^* = E_{n_1}(t) + ES_{n_2-j}(t) \quad (4.1)$$

La eliminación de componentes senoidales continúa hasta que la señal residuo es no oscilatoria. Una vez que sean extraído todos los componentes senoidales, el residuo es una suma de términos exponenciales y es filtrado por la función *ExpFilter* (ver sección 4.2.4). Este filtro lleva a cabo una eliminación sucesiva de los componentes exponenciales, iniciando con el componente con el más bajo factor de decaimiento.

La Figura 4.2 ilustra el proceso de filtrado de las componentes senoidales de la respuesta del sistema. La línea más delgada representa la componente con más alta frecuencia, es decir, la primer componente filtrada. Y así en forma progresiva de grosor de línea se ilustran las componentes filtradas, la línea más gruesa representa el residuo una vez que se han filtrado las componentes senoidales.

La Figura 4.3 ilustra el proceso de filtrado de las componentes exponenciales, la señal continua representa la señal de entrada y las líneas interrumpidas representan las componentes exponenciales filtradas.

Los filtros digitales han sido usados para dos propósitos generales [Smith99]: separación de señales que han sido combinadas y restauración de señales que han sido distorsionadas de alguna manera. El propósito en nuestra aplicación es separar las diferentes componentes que conforman la respuesta al impulso de un sistema LIT. Esta separación de componentes permite que cada componente sea analizada por se-

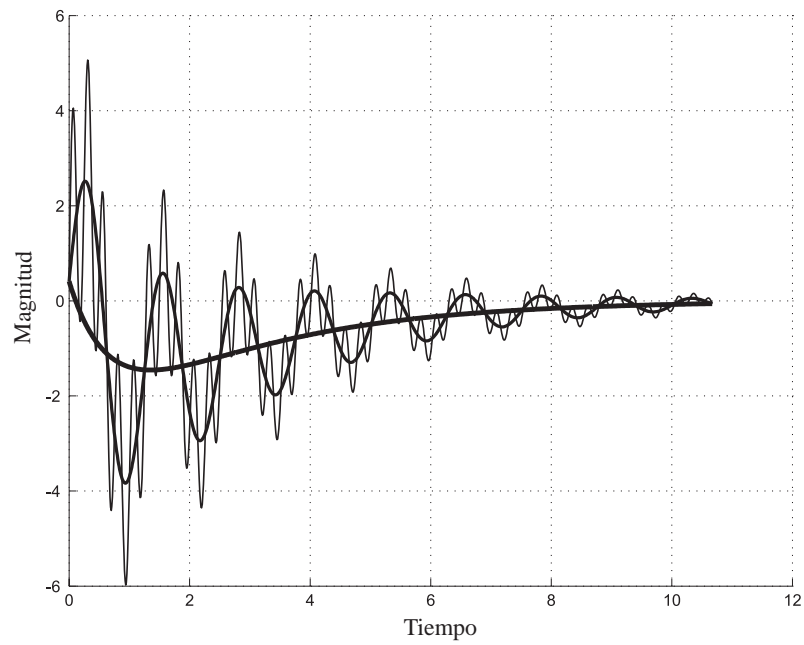


Figura 4.2: Filtrado de las componentes sinusoidales

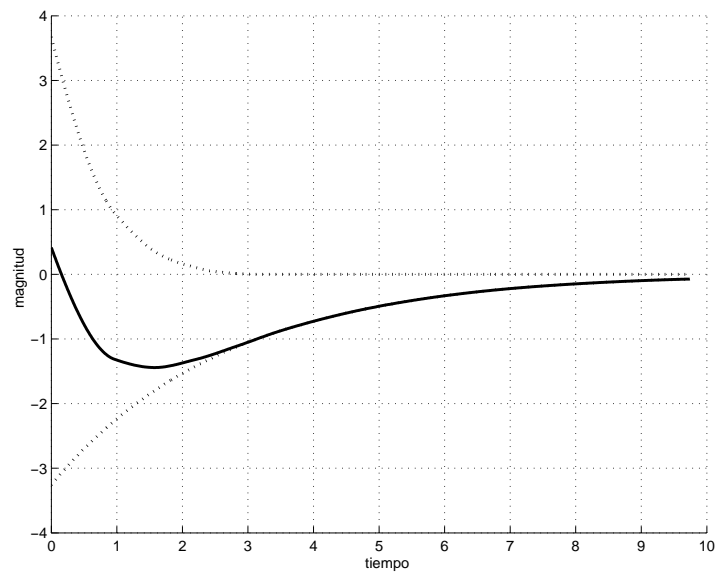


Figura 4.3: Filtrado de las componentes exponenciales de la respuesta

parado para determinar la estructura del modelo y los parámetros de esa estructura. Primero se separan las componentes senoidales, iniciando por la de mayor frecuencia, y posteriormente las componentes exponenciales, iniciando con el menor factor de decaimiento.

4.2.1. Filtro de Componentes Senoidales TPAFilter

Se llama filtro del promedio de dos puntos (TPA por sus siglas en inglés de “Two Point Average”) por que se basa en promediar un punto de la envolvente superior con un punto de la envolvente inferior para filtrar las componentes senoidales.

Este proceso de filtrado de componentes senoidales se lleva a cabo en tres etapas: se determina la envolvente de la señal oscilatoria, se determina el promedio de esa envolvente y se obtiene la señal residual a partir del promedio de la envolvente. Esta función recibe como parámetros la señal $X(t)$ que se desea filtrar, es decir, la respuesta del sistema, el orden del sistema k (con valor inicial 0) y la matriz de parámetros de los componentes P (vacía al inicio). La función se ejecuta en forma iterativa hasta que se hayan eliminado todas las componentes senoidales de $X(t)$ y regresa como resultados $X^*(t)$, k , P donde:

- $X^*(t)$ es la señal residuo una vez que se han eliminado las componentes senoidales.
- nk es el orden parcial del modelo. Hasta este punto, solo están consideradas las aportaciones de cada una de las componentes senoidales eliminadas y falta agregar las contribuciones del resto de las componentes.
- P es la matriz de parámetros de cada una de las componentes eliminadas, con la estructura mostrada en la Tabla 4.2. Los valores contenidos en cada columna de esta Tabla corresponden a las variables de la ecuación 4.2

Tabla 4.2: Estructura de la matriz de parámetros de las componentes senoidales

Coefficiente a	Factor de decaimiento	frecuencia	angulo de defasamiento
a_1	r_1	ω_1	ϕ_1
a_2	r_2	ω_2	ϕ_2
\vdots	\vdots	\vdots	\vdots
a_{n_2}	r_{n_2}	ω_{n_2}	ϕ_{n_2}

$$ES_j(t) = a_j e^{-r_j t} \sin(\omega_j t + \phi_j) \quad (4.2)$$

donde el renglón j define a la j -ésima componente senoidal de la respuesta del sistema.

El Algoritmo 2 corresponde al filtro de componentes senoidales. En esta función se lleva a cabo el filtrado y parametrización de las componentes senoidales. El filtro está compuesto por varias etapas, organizadas como se indica en el Algoritmo 2. A continuación se presenta la descripción de cada una de estas etapas.

El primer paso realizado por este algoritmo es analizar si la señal contiene oscilaciones periódicas (línea 1 del algoritmo), en cuyo caso es una componente senoidal. Si la señal es senoidal se inicia el ciclo de filtrado de las componentes, este ciclo inicia en la línea 2 y termina en la línea 11. En las líneas 3 y 4 se calculan los extremos de la señal, almacenándolos en dos vectores, uno para los máximos y otro para los mínimos. Con estos dos vectores se determina una envolvente para la componente senoidal (líneas 5 y 6) y a partir de ésta se calcula el vector de puntos promedio de la envolvente (línea 7). Los puntos promedio se interpolan para aproximar el residuo y filtrar la componente senoidal (línea 8). Para recuperar la serie de tiempo de la componente filtrada se le resta el residuo a la señal de entrada (línea 9); una vez que se ha recuperado esta componente se le calculan los parámetros y se agregan como un renglón en la matriz P (línea 10). En la línea 11 se incrementa el contador de orden del modelo en dos unidades. Este proceso se repite hasta que la señal deja de ser oscilatoria o si las oscilaciones dejan de ser periódicas, ya que el algoritmo verifica

Algoritmo 2 Filtro de componentes senoidales

```

TPAFILTER( $X, nk, P$ )
1  if oscillatory( $X$ )
2  do
3     $Xtr_{max} \leftarrow determine_{maxima}(X)$ 
4     $Xtr_{min} \leftarrow determine_{minima}(X)$ 
5     $Envelope_{max} \leftarrow CubicInterp(Xtr_{max})$ 
6     $Envelope_{min} \leftarrow CubicInterp(Xtr_{min})$ 
7     $m \leftarrow average(Envelope_{max}, Envelope_{min})$ 
8     $X^* \leftarrow CubicInterp(m)$ 
9     $S \leftarrow X - X^*$ 
10    $P \leftarrow [P; Param(S)]$ 
11    $nk \leftarrow nk + 2$ 
12   until  $X^*$  non - oscillatory
13 else
14    $X^* \leftarrow X$ 
15 regresar ( $X^*, k, P$ )

```

que las oscilaciones se repitan con periodicidad y en caso de no ser así, finaliza su ejecución. Esto último garantiza que las oscilaciones que se presentan debido a la suma de exponenciales reales, cuyos coeficientes tienen signos alternados, no se confunda con oscilaciones senoidales. La Figura 3.4 ilustra la suma de tres exponenciales, donde se puede apreciar como la señal presenta oscilaciones no periódicas.

Una vez que se han filtrado todas las componentes senoidales la función termina enviando como salida: la señal residuo, el orden del modelo y los parámetros de las componentes filtradas.

Una modificación a un filtro de diferencias de dos puntos se plantea como la alternativa para realizar el primer paso en el proceso de filtrado y le llamaremos filtro diferencial de dos puntos. La forma como funciona es muy simple; se detectan y capturan los puntos donde existen cambios en el signo de la derivada de la serie de tiempo que se está analizando. El cálculo de la derivada en cada punto de la señal está descrito por la ecuación 4.3.

$$x'(t) = \frac{x(t + \Delta t) - x(t - \Delta t)}{2\Delta t} \quad (4.3)$$

donde:

$x'(t)$ derivada de x en el instante t .

Δt periodo de muestreo.

$x(t + \Delta t)$ es la señal de entrada en el instante $t + \Delta t$.

$x(t - \Delta t)$ es la señal de entrada en el instante $t - \Delta t$

Existen diversas formas de calcular la derivada [Smith88, Mathews00], se ha escogido esta formulación por que ofrece una mejor aproximación hacia el valor de la derivada, es decir, la secante formada entre los puntos anterior y siguiente da una mejor aproximación del lugar del máximo, que usando diferencias hacia adelante o hacia atrás.

Mediante la aproximación de la ecuación 4.3 se capturan los puntos en los cuales existe un cambio en el signo de la derivada, es decir, estamos detectando los máximos

y mínimos de x . Con la información anterior se forman dos vectores, uno para los máximos y otro para los mínimos. Estos vectores permiten modelar una envolvente de la señal de entrada. Con la finalidad de poder calcular los puntos medios se interpola para cada instante de tiempo del vector de máximos un valor en el vector de mínimos y de igual manera para cada instante de tiempo del vector de mínimos un valor en el vector de máximos. Para llevar a cabo esto se usa la función de interpolación cúbica *splines* [Lou94]. Este procedimiento nos permite interpolar los valores entre dos extremos consecutivos (detalle “Puntos Interpolados” en la Figura 4.4), una vez que se han agregado los puntos interpolados a los vectores de máximos y mínimos los vectores resultantes forman la envolvente de la señal y se usan para calcular los puntos medios.

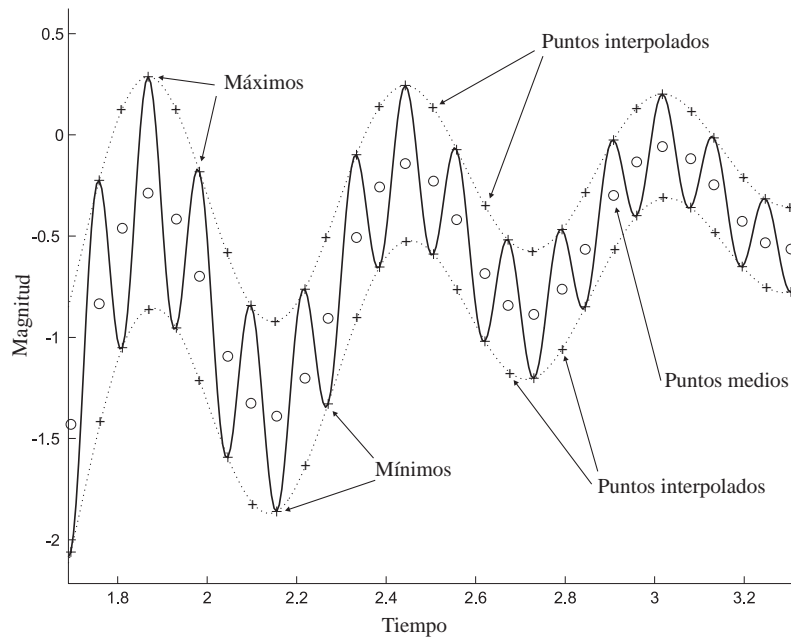


Figura 4.4: Obtención de la envolvente de la señal de entrada

4.2.2. Cálculo de Puntos Medios

Una vez que se tiene la envolvente se procede a calcular la portadora usando el vector de puntos medios (indicados en la Figura 4.4). Para cada instante de la envolvente se calcula su valor medio, aplicando la fórmula de la ecuación siguiente:

$$m(k) = \frac{Envelope_{\text{máx}}(k) + Envelope_{\text{mín}}(k)}{2} \quad (4.4)$$

donde:

$m(k)$ es el valor del punto medio en el k –ésimo instante, $Envelope_{\text{máx}}(k)$ es el valor de la envolvente formada por los máximos en el instante k y $Envelope_{\text{mín}}(k)$ es el valor de la envolvente formada por los mínimos en el instante k .

La fórmula de la ecuación 4.4 es una implementación muy simple de un filtro pasabajas, que permite eliminar la componente de más alta frecuencia en forma cualitativa, es decir, no requerimos conocer los parámetros de la componente para eliminarla. El cálculo de los parámetros de cada uno de los componentes se verá más adelante, en la sección 4.3.2.

4.2.3. Reconstrucción de Residuos

El vector m representa a la señal residuo $X^*(t)$ definida en la ecuación 4.1, pero discretizada en los valores de la envolvente. Para hacerla continua en los valores del vector de tiempo, t , hacemos uso nuevamente de la función de interpolación cúbica splines. La Figura 4.5 muestra el proceso del filtrado cualitativo de la senoidal y la obtención de $X^*(t)$ a partir de $m(k)$.

En este punto se ha filtrado la componente senoidal más rápida contenida en $X(t)$. Con la finalidad de obtener los parámetros de la componente eliminada $S(t)$, se procede a recuperarla mediante la operación indicada en la ecuación 4.5.

$$S(t) = X(t) - X^*(t) \quad (4.5)$$

El resultado será una señal senoidal montada en el eje de las abscisas. Para garantizar lo anterior se aplica un filtro que elimine las perturbaciones de baja frecuencia, desviaciones y variaciones periódicas derivadas del error numérico al calcular $X^*(t)$. Este filtro consiste en restarle su promedio a $S(t)$, es decir, $S(t) = S(t) - \overline{S(t)}$ [Ljung99, p. 387]. Una vez que se tiene $S(t)$ se procede a calcular sus parámetros, los cuales se almacenan en P . Este proceso se repite hasta que se hayan eliminado todas las componentes senoidales de la respuesta del sistema. Es decir, hasta que $X^*(t)$ sea no oscilatoria o no periódica. En tal caso se retornan los valores del residuo, el orden y la matriz de parámetros.

La aplicación de este filtro se hace en forma iterativa parando una vez que la señal resultante sea no oscilatoria.

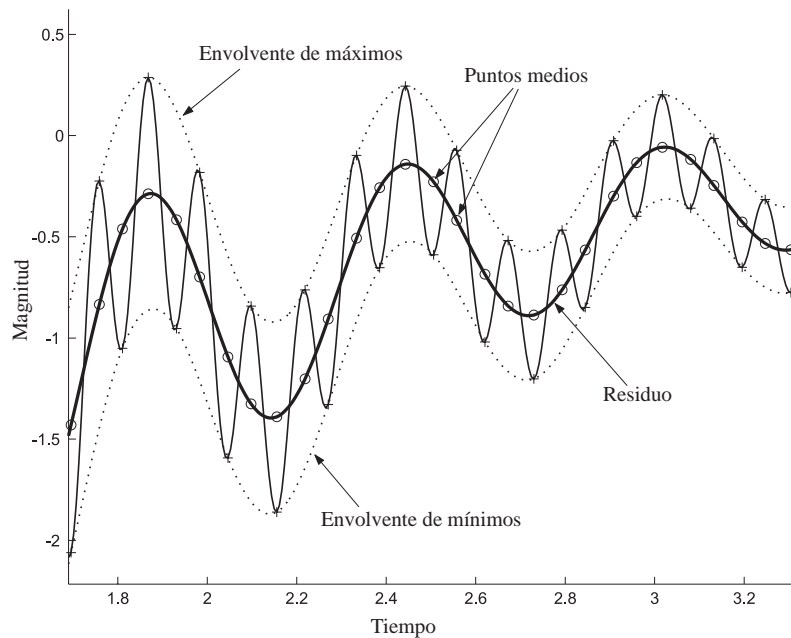


Figura 4.5: Un paso en el proceso de filtrado de las componentes oscilatorias

Al terminar la aplicación del filtro, se conoce la cantidad de componentes senoidales que conforman la respuesta del sistema, es decir se sabe cuanto vale n_2 en la

expresión del orden del sistema:

$$n = n_1 + 2n_2 \quad (4.6)$$

Una vez conocido el número de componentes senoidales, n_2 , podemos derivar una terminología para etiquetar la respuesta del sistema, en función de estos componentes. Denotaremos a la suma de n_2 componentes senoidales como ESn_2 .

4.2.4. Filtro de Componentes Exponenciales **EXPFilter**

Una vez que la señal ha sido procesada por el filtro de componentes senoidales la salida de este filtro es la señal residuo $X^*(t)$, la cual puede contener componentes exponenciales. La función *EXPFilter* identifica la cantidad de componentes exponenciales presentes en $X^*(t)$ y los filtra. El Algoritmo 3 presenta la operación del filtro de componentes exponenciales, *EXPFilter*.

La entrada de la función *EXPFilter* es la señal residuo $X^*(t)$ y la salida es el modelo del sistema, en el caso en el que todas las componentes filtradas hayan sido exponenciales el residuo deberá ser cero. Si la forma de $X^*(t)$ no presenta extremos (ver Figura 4.6) el algoritmo verifica si se trata de una señal con forma exponencial, en caso afirmativo se incluyen sus parámetros en el modelo y en otro caso se ignoran y termina el proceso.

El primer paso de este filtro es identificar el número de componentes exponenciales presentes en la señal, mediante la determinación de los extremos presentes (línea 1). Como se establece en el teorema 2, el número de extremos más uno es igual al número de componentes exponenciales que conforman la señal.

Una vez que se conoce el número de extremos se puede proponer la estructura de la ecuación diferencial que pueda reproducir el comportamiento exhibido por la señal de entrada al filtro de componentes exponenciales. Es decir, en términos prácticos en este punto ha terminado la identificación cualitativa.

Algoritmo 3 Filtro de componentes exponenciales

```

EXPFILTER( $X^*$ ,  $nk$ ,  $P$ )
1    $Xtr \leftarrow numextremos(X^*)$ 
2   if  $Xtr > 0$ 
3     do
4        $nk \leftarrow nk + 1$ 
5        $P_i \leftarrow PuntoInflexion(X^*)$ 
6        $P_{exp} \leftarrow ParamExp(X^*, P_i)$ 
7        $P = [P; P_{exp}]$ 
8        $X^* \leftarrow X^* - Exponencial(P_{exp})$ 
9        $Xtr \leftarrow Xtr - 1$ 
10    while  $Xtr > 0$ 
11  else
12     $P_{exp} \leftarrow ParamExp(X^*, 0)$ 
13     $X^* \leftarrow X^* - Exponencial(P_{exp})$ 
14    if  $X^* = 0$ 
15       $nk \leftarrow nk + 1$ 
16       $P \leftarrow [P; P_{exp}]$ 
17  regresar ( $X^*$ ,  $k$ ,  $P$ )
  
```

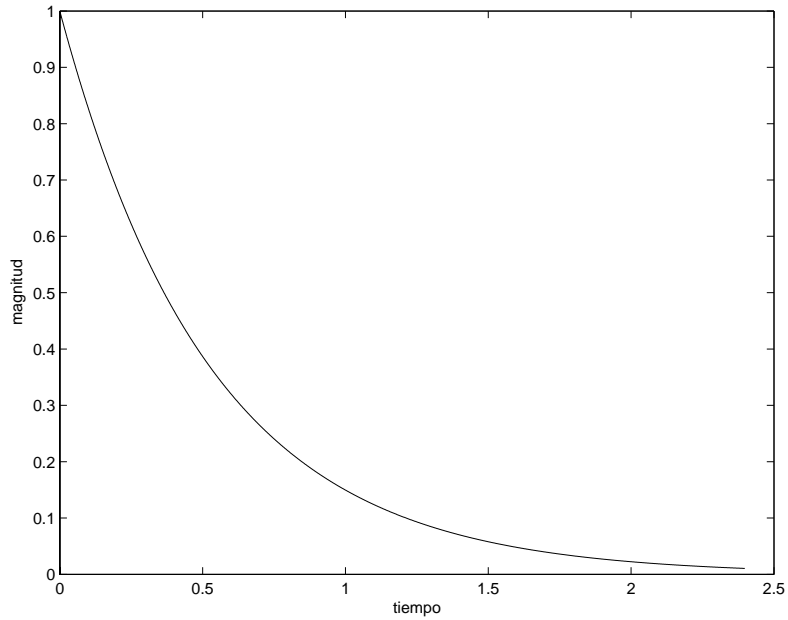


Figura 4.6: Señal $X^*(t)$, que no presenta extremos

Si el número de extremos es mayor que 0 inicia el proceso de filtrado agregando una unidad al contador del orden del modelo por cada componente exponencial filtrada (línea 4). Las componentes exponenciales se eliminan una a la vez, iniciando con la del factor de decaimiento más lento. Para este efecto se usa la información del {ultimo punto de inflexión hallado en la señal (línea 5)}. Truncando X^* se usan los datos entre $X^*(P_i)$ y $X^*(N)$ para identificar la componente con factor de decaimiento más lento (línea 6). Los parámetros obtenidos en la línea anterior se agregan como un renglón de la matriz P (línea 7). se actualiza la señal residuo restandole la componente exponencial identificada en la línea 6 (línea 8). El ciclo termina decrementando el contador del número de extremos (línea 9). Este proceso se realiza tantas veces como extremos se hayan detectado. En caso de no existir ningún extremo el algoritmo verifica si la señal tiene forma exponencial y la filtra (líneas 12 y 13), si el resultado de la operación de la línea 13 es cero, se incrementa el contador de orden del modelo (línea 15) y los parámetros identificados en la línea 12 se agragan a la matriz de

parametros (línea 16). En otro caso se ignoran y termina la ejecución de la función. Los valores de salida de la función son la señal residuo, el orden del modelo y los parámetros del modelo.

Como se menciona anteriormente, el número de componentes exponenciales, n_1 , nos permite complementar la ecuación 4.6 y de esta manera conocer completamente la estructura del modelo que describe el comportamiento del sistema. Hasta este punto se ha completado la parte cualitativa del proceso de identificación. A continuación se presenta el procedimiento propuesto para llevar a cabo el filtrado y parametrización de las componentes exponenciales.

De igual manera que en el filtro de componentes senoidales, podemos derivar una terminología para etiquetar la señal, denotaremos a la suma de n_1 exponenciales como En_1 . Esta etiqueta complementa la etiqueta definida para los componentes senoidales y describe completamente la respuesta del sistema, por ejemplo, la etiqueta: *Componentes* = (E_3, ES_2) describe una respuesta que contiene 3 componentes exponenciales y 2 componentes senoidales.

4.3. Identificación Cuantitativa

Una de las características que extiende las capacidades del algoritmo *QSI* es la determinación de los coeficientes de la ecuación diferencial, es decir, la parametrización del modelo propuesto. El método propuesto considera parametrizar cada una de las componentes de la respuesta del sistema al mismo tiempo que se van filtrando.

4.3.1. Componentes Exponenciales

Esta sección presenta el método de parametrización de los componentes exponenciales. Como se observa anteriormente para eliminar las componentes exponenciales es necesario conocer sus parámetros. Para tal efecto se asume que ya han sido filtradas las componentes senoidales, y que el residuo es una suma de exponenciales.

La base para separar los componentes exponenciales es el teorema 3, propuesto y demostrado en [Flores02 et al].

Teorema 3 *Si $x_1, x_2, x_3, \dots, x_k$ son componentes de una suma de k exponenciales $E_k(t)$ con factores de decaimiento $r_1, r_2, r_3, \dots, r_k$ respectivamente, y $r_1 > r_2 > \dots > r_k$, entonces $x_1(t^*) \ll x_2(t^*) \ll \dots \ll x_k(t^*)$, donde t^* es el instante del punto de inflexión.*

Demostración: Para una suma con solo dos componentes exponenciales se supone que

$$x_1(t) = a_1 e^{-r_1 t} \quad (4.7)$$

$$x_2(t) = a_2 e^{-r_2 t} \quad (4.8)$$

y que

$$\frac{r_1}{r_2} = \alpha > 1 \quad (4.9)$$

Es evidente que

$$\left(\frac{1}{\alpha}\right)^2 \ll 1 \quad (4.10)$$

la hipótesis es:

$$x_1(t^*) \ll x_2(t^*) \quad (4.11)$$

$$\frac{x_1(t^*)}{x_2(t^*)} \ll 1 \quad (4.12)$$

si hacemos

$$\beta = \frac{x_1(t^*)}{x_2(t^*)} \ll 1 \quad (4.13)$$

entonces

$$E_2(t) = -a_1 e^{-r_1 t} + a_2 e^{-r_2 t} \quad (4.14)$$

es la suma de esos exponenciales. De acuerdo a la definición de la segunda derivada, hay un punto de inflexión cuando

$$E''(t) = 0 \quad (4.15)$$

$$E_2''(t) = a_1 r_1^2 e^{-r_1 t} - a_2 r_2^2 e^{-r_2 t} = 0 \quad (4.16)$$

resolviendo para t^*

$$t^* = \frac{\ln\left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)}{(r_2 - r_1)} \quad (4.17)$$

entonces sustituimos t^* en la ecuación 4.13

$$\beta = \frac{a_1 e^{-r_1 \frac{\ln\left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)}{(r_2 - r_1)}}}{a_2 e^{-r_2 \frac{\ln\left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)}{(r_2 - r_1)}}} \quad (4.18)$$

$$= \frac{a_1 e^{\ln\left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right) \left(-\frac{-r_1}{(r_2 - r_1)}\right)}}{a_2 e^{\ln\left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right) \left(-\frac{-r_2}{(r_2 - r_1)}\right)}} \quad (4.19)$$

$$= \frac{a_1 \left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)^{\left(-\frac{-r_1}{(r_2 - r_1)}\right)}}{a_2 \left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)^{\left(-\frac{-r_2}{(r_2 - r_1)}\right)}} \quad (4.20)$$

$$= \frac{a_1}{a_2} \left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right)^{\left(\frac{r_2 - r_1}{(r_2 - r_1)}\right)} \quad (4.21)$$

$$= \frac{a_1}{a_2} \left(\frac{a_2}{a_1} \left(\frac{r_2}{r_1}\right)^2\right) \quad (4.22)$$

finalmente

$$\beta = \frac{a_1 a_2}{a_2 a_1} \left(\frac{r_2}{r_1}\right)^2 \quad (4.23)$$

$$\beta = \left(\frac{r_2}{r_1}\right)^2 \quad (4.24)$$

comparando la ecuación 4.24 con la ecuación 4.9, tenemos que

$$\beta = \left(\frac{1}{\alpha}\right)^2 \quad (4.25)$$

como

$$\left(\frac{1}{\alpha}\right)^2 \ll 1 \quad (4.26)$$

por lo tanto

$$\beta \ll 1 \quad (4.27)$$

En general, una vez que $r_1 > r_2 > r_3 > \dots > r_k$

$$x_1(t^*) \ll x_2(t^*) \ll x_3(t^*) \ll \dots \ll x_k(t^*) \quad \square \quad (4.28)$$

El teorema 3 implica que a partir del instante t^* (punto P_i en la Figura 4.7), las componentes, x_1, \dots, x_{k-1} son despreciables con respecto a x_k . De esta forma podemos usar el segmento final de $E_2(t)$, a partir de t^* , para aproximar y parametrizar la componente con factor de decaimiento más lento.

Con éste resultado se plantea el filtrado de las componentes exponenciales por medio del siguiente proceso: primero se detecta el último punto de inflexión (P_i) y se eligen los puntos P_1 y P_2 (ver Figura 4.7). Se toman todos los puntos del segmento delimitado entre P_1 y P_2 , estos puntos se seleccionan de acuerdo con las siguientes fórmulas:

$$P_1 = E_2(t^* + \Delta t) \quad (4.29)$$

$$P_2 = E_2(t(N)) \quad (4.30)$$

donde: N es la longitud de $E(t)$ y t^* es el instante del último punto de inflexión.

Una vez que se sabe que la componente que será eliminada es exponencial, se utiliza el método de linealización de los datos de la siguiente forma:

Sea

$$x = ae^{-rt} \quad (4.31)$$

se aplican logaritmos a la ecuación 4.31

$$\log(x) = \log(a) - rt \quad (4.32)$$

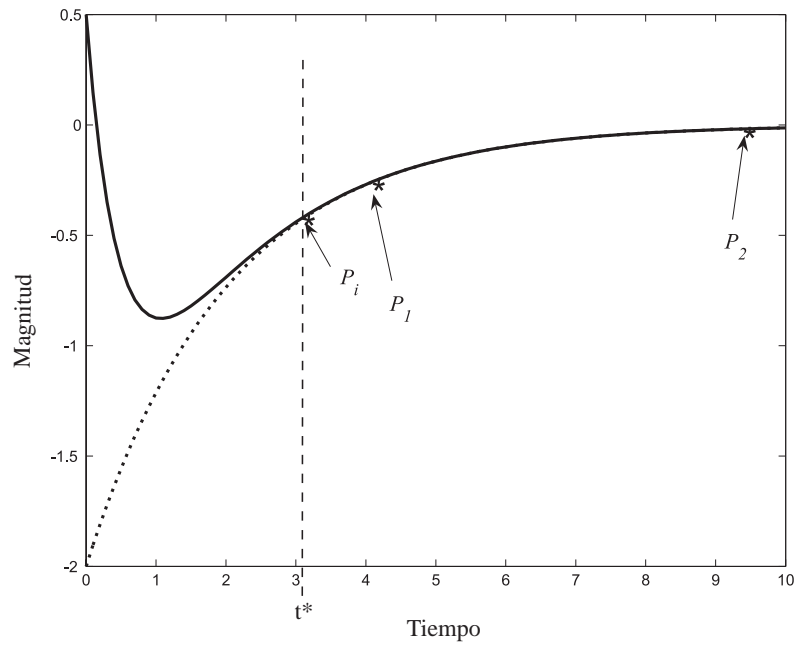


Figura 4.7: Puntos importantes en la eliminación de los componentes exponenciales

hacemos los siguientes cambios de variable

$$\log(x) = y \quad (4.33)$$

$$\log(a) = b \quad (4.34)$$

$$r = m \quad (4.35)$$

y se obtiene una relación lineal entre las variables y y t

$$y = -mt + b \quad (4.36)$$

Los datos originales (t, x) se han transformado en $(t, \log(x))$; este proceso es conocido como el método de linealización de los datos [Mathews00]. Posteriormente se procede a calcular la recta de regresión dada en la ecuación 4.36, para el conjunto de datos $(t, \log(x))$.

Una vez calculadas las variables m y b hacemos el cambio de variable para m y

hallamos el valor de a_n mediante la relación

$$a = e^b \quad (4.37)$$

Con los valores de a y r , se calcula la componente que se está filtrando, por ejemplo, si se trata del k -ésimo componente que se sustrae de la señal $E_{n_1}(t)$, para obtener el residuo $X^*(t)$:

$$E_k(t) = a_k e^{-r_k t} \quad (4.38)$$

$$X^*(t) = E_{n_1}(t) - E_k(t) \quad (4.39)$$

El residuo $X^*(t)$ contiene la suma de todos los exponenciales, excepto el que ha sido filtrado. En este caso, después de eliminar el k -ésimo componente exponencial, el residuo es

$$X_k^*(t) = E_{n_1-k}(t) \quad (4.40)$$

Este procedimiento se repite n_1 veces, es decir, hasta que el residuo no contenga más componentes exponenciales.

En el caso de que el número de extremos del residuo $X^*(t)$ sea cero, aún queda la posibilidad de que $X^*(t)$ tenga forma exponencial, en este caso la función *ExpFilter* verifica si la forma de $X^*(t)$ es exponencial o no. En caso afirmativo se aplica el proceso de parametrización de la curva, para filtrar también esta componente.

Como se puede ver en esta sección, el cálculo de los parámetros de la componente es necesario para poder filtrarla. El método para determinar los parámetros de la exponencial es de suma importancia en la estimación de los parámetros de las componentes senoidales ya que la forma de la señal es $ae^{-rt} \text{sen}(\omega t)$ donde las curvas exponenciales $\pm ae^{-rt}$ la envuelven. A continuación se presenta el método propuesto para la parametrización de las componentes senoidales.

4.3.2. Componentes Senoidales

De igual manera que en el caso de los componentes exponenciales, al mismo tiempo que se van filtrando los componentes senoidales se estiman sus parámetros. Como se vió en la sección 4.2.3 el residuo es la señal resultante una vez que se ha filtrado la componente más rápida de la señal. Partiendo de este resultado se recupera la componente filtrada mediante la siguiente formulación:

$$ES_j(t) = X(t) - X^*(t) \quad (4.41)$$

donde:

$X(t)$ es la señal de entrada.

$X^*(t)$ es el residuo.

$ES_j(t)$ es la componente que se esta filtrando.

Despues de aplicar esta operación se obtiene una senoidal amortiguada definida por:

$$ES_j(t) = a_j e^{-r_j t} \sin(\omega_j t + \phi_j) \quad (4.42)$$

Esta operación equivale a desmontar la senoidal de la señal y montarla sobre el eje de las abscisas. Es decir, la senoidal sustraída oscila alrededor del eje de las abscisas, aproximándose asintóticamente a él. Además las dos curvas exponenciales $y_e = \pm a_j e^{-r_j t}$ la envuelven, tocándola en sus extremos (ver Figura 4.8).

Para calcular las variables que definen esta curva seleccionamos todos los puntos donde existan los máximos y son marcados como $P_1, P_2 \dots P_m$ (ver Figura 4.8).

Usando el vector de puntos formado por estos máximos se determina la envolvente superior, la cual representa la parte exponencial del decaimiento de la senoidal. Las variables a y r , se calculan utilizando las ecuaciones 4.37 y 4.35 deducidas en la subsección anterior.

Para determinar ω_j usamos la ecuación 4.43

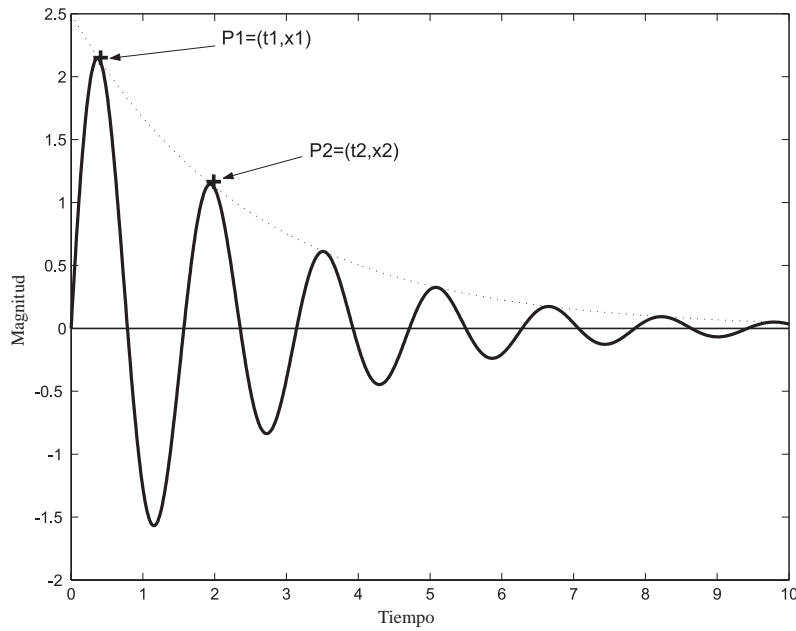


Figura 4.8: Parametrización de una componente senoidal

$$\omega_j = \frac{\sum_{i=2}^m \frac{2\pi}{T_{i,i-1}}}{m-1} \quad (4.43)$$

donde $T_{i,i-1}$ es el período entre los instantes t_i y t_{i-1} , correspondientes a los puntos P_i y P_{i-1} ($i=2, 3, \dots, m$) respectivamente. En general el periodo, $T_{i+1,i}$, se calcula por medio de la siguiente ecuación

$$T_{i,i-1} = t_i - t_{i-1} \quad (4.44)$$

En otras palabras lo que se está haciendo es calcular un valor promedio para ω_j considerando las posibles variaciones, sobre todo, en mediciones obtenidas de sistemas reales.

Finalmente se calcula el ángulo de defasamiento ϕ de la siguiente manera:

Se sustituye la relación trigonométrica

$$\sin(\omega_j t + \phi_j) = \sin(\omega_j t) \cos(\phi_j) + \cos(\omega_j t) \sin(\phi_j) \quad (4.45)$$

en la ecuación 4.42.

$$ES_j(t) = a_j e^{-r_j t} [\sin(\omega_j t) \cos(\phi_j) + \cos(\omega_j t) \sin(\phi_j)] \quad (4.46)$$

Hacemos $\omega_j t = k2\pi$, donde $k = 0, 1, 2, 3, \dots$ y lo sustituimos en la ecuación 4.46.

$$ES_j(\hat{t}) = a_j e^{-r_j \hat{t}} \sin(\phi_j) \quad (4.47)$$

donde

$$\hat{t} = kT \quad (4.48)$$

Y resolvemos para ϕ_j

$$\phi_j = \arcsin\left(\frac{ES_j(kT_j)}{a_j e^{-r_j kT_j}}\right) \quad (4.49)$$

Cabe hacer la observación que los parámetros que se calculan en esta sección son las raíces de la ecuación característica de la ecuación diferencial mediante la cual se aproxima al sistema en estudio, es decir, mediante estos parámetros podemos obtener los coeficientes de la ecuación diferencial ordinaria que describe al sistema.

4.4. Obtención del Modelo

Los resultados obtenidos mediante el método propuesto corresponden a las raíces (también llamados eigenvalores ó autovalores) de la ecuación característica. Una vez que se obtienen los parámetros que definen a cada una de las componentes de la respuesta es posible calcular la ecuación característica, y a partir de ésta construir la EDO de la siguiente forma.

4.4.1. Raíces de la Ecuación Característica

Las variables r y ω de cada una de las componentes identificadas en la respuesta del sistema representan las raíces de la ecuación característica, estas raíces generan

los autovalores de la forma siguiente:

$$m = -r \pm j\omega \quad (4.50)$$

La ecuación 4.50 define el eigenvalor para el caso de las componentes senoidales. Para el caso de las componentes exponenciales los eigenvalores están dados por:

$$m = -r \quad (4.51)$$

y este último resulta ser el caso más simple, en cuyo caso la ecuación característica está dada por

$$m + r = 0 \quad (4.52)$$

sustituyendo m por el operador diferencial $\frac{dx(t)}{dt}$, resulta la ecuación diferencial de primer orden siguiente

$$\frac{dx(t)}{dt} + rx(t) = 0 \quad (4.53)$$

En el caso de raíces complejas la ecuación característica resultante es de segundo orden y está definida a partir de la ecuación 4.50

$$(m + r + j\omega)(m + r - j\omega) = 0 \quad (4.54)$$

$$m^2 + 2rm + (r^2 + \omega^2) = 0 \quad (4.55)$$

y sustituyendo m obtenemos la ecuación diferencial ordinaria

$$\frac{d^2x(t)}{dt^2} + 2r\frac{dx(t)}{dt} + (r^2 + \omega^2)x(t) = 0 \quad (4.56)$$

Estos resultados ilustran las dos opciones más simples en la obtención del modelo de un sistema a partir de las características de su respuesta. Para respuestas más complejas, es decir, respuestas que contengan varias componentes senoidales y exponenciales la metodología es la misma y la ecuación diferencial se obtiene mediante la multiplicación de todas las raíces obtenidas. En general, la ecuación característica

de $n - \text{ésimo}$ orden que nos permita generar la EDO que describa la dinámica del sistema en estudio se obtiene a partir de las siguientes expresiones.

$$\begin{aligned} \text{Polinomio}_1 &= (m + r_1 + j\omega_1)(m + r_1 - j\omega_1) \cdots \\ &\cdots (m + r_2 + j\omega_2)(m + r_2 - j\omega_2) \cdots \\ &\cdots (m + r_{n_2} + j\omega_{n_2})(m + r_{n_2} - j\omega_{n_2}) \end{aligned} \quad (4.57)$$

$$\text{Polinomio}_2 = (m + r_1)(m + r_2) \cdots (m + r_{n_1}) \quad (4.58)$$

donde Polinomio_1 se refiere a las componentes oscilatorias y Polinomio_2 a las componentes exponenciales. Multiplicando estos dos polinomios se obtiene una ecuación de $n - \text{ésimo}$ orden con la siguiente forma

$$m^n + a_{n-1}m^{n-1} + \cdots + a_1m + a_0 = 0 \quad (4.59)$$

finalmente, sustituimos m para obtener la ecuación diferencial ordinaria

$$\frac{d^n x(t)}{dt^n} + a_{n-1} \frac{d^{n-1} x(t)}{dt^{n-1}} + \cdots + a_1 \frac{dx(t)}{dt} + a_0 x(t) = 0 \quad (4.60)$$

El método descrito propone a un mismo tiempo la estructura del modelo y estima los coeficientes para esta estructura, mediante operaciones simples. Esta descripción será la base para el desarrollo del algoritmo que nos permita automatizar todo este proceso. La función que automatiza este proceso se llama *Pol_carac*

4.4.2. Función Pol_carac

La función *Pol_carac* calcula el polinomio característico utilizando la información que proporciona la matriz final de parametros (Tabla 4.3).

Las variables de la segunda y tercera columna representan las raíces de la ecuación característica; la segunda columna representa la parte real de la raíz y la tercera columna la parte imaginaria. Si la parte imaginaria es cero entonces la raíz representa una componente exponencial en otro caso es la raíz de una componente senoidal.

Tabla 4.3: Matriz de parámetros de las componentes senoidales y exponenciales

Coeficiente a	Factor de decaimiento	frecuencia	angulo de defasamiento
a_1	r_1	ω_1	ϕ_1
\vdots	\vdots	\vdots	\vdots
a_{n_2}	r_{n_2}	ω_{n_2}	ϕ_{n_2}
a_{n_2+1}	r_{n_2+1}	0	0
\vdots	\vdots	\vdots	\vdots
$a_{n_2+n_1}$	$r_{n_2+n_1}$	0	0

El desarrollo de este software pasó por varias migraciones hacia las versiones más actuales del MATLAB. Sus inicios los tuvo en la versión 5.2, siguiendo con la version 6.0 para finalmente concluirse en la versión 6.5 para windows. Fué probado en las plataformas de Matlab para Windows y Linux, comprobando su buen funcionamiento en las versiones 5.2, 6.0 6.5 y 7.01 de Windows y en la versión 7.01 para Linux. El sistema tuvo un buen desempeño en ambas plataformas.

En el apéndice A.5 se presenta el manual de instalación y uso, el cual contiene un par de ejemplos.

4.5. Adquisición de Datos y Pretratamiento

En la actualidad dado el vertiginoso avance de la tecnología, existe un sinfín de dispositivos y equipos de medición y adquisición de datos. En particular para cada variable que se quiere medir y/o adquirir se han desarrollado diferentes métodos y aparatos para capturarlas. A pesar de esta gran variedad de instrumentos es posible representar a los equipos de adquisición y medición por una estructura generalizada, ver Figura 4.9.

La operación de cada uno de los elementos que conforman un sistema de adquisición de datos está bien definida.

- Sensor de entrada.- Es el elemento encargado de hacer el contacto con la variable que se va a medir. El tipo de sensor depende de la variable física a medir

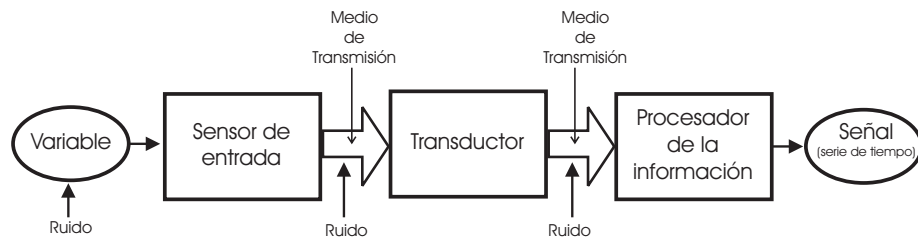


Figura 4.9: Estructura general de un sistema de adquisición de datos

(temperatura, voltaje, presión, flujo, densidad, etc.)

- **Transductor.-** Este elemento se encarga de acondicionar la señal de forma tal que sea un valor de voltaje o un determinado número de impulsos en el tiempo.
- **Medio de transmisión.-** Es el elemento mediante el cual se envía la información entre elementos, esta parte del sistema de adquisición -al igual que la variable- se ve afectada por factores externos que influyen sobre la naturaleza de la variable. Estas variaciones para los fines de la adquisición son ruido que en un momento dado pueden introducir un error apreciable.
- **Procesador de la información.-** Este elemento es el que se encarga de realizar los procesos de muestreo, codificación, representación y almacenamiento. Esta etapa deja lista la información mediante una serie de tiempo o una matriz de datos para posteriormente ser usada en alguna aplicación particular. En la actualidad, por lo general, en esta etapa se usan tarjetas de adquisición de datos diseñadas para velocidades de muestreo muy altas.

Una situación típica, cuando se trabaja con series de tiempo, es que la señal esté conformada de dos partes; una parte que contiene las frecuencias de interés y otra que está conformada por ruido, que sea de un ancho de banda mayor que la señal de interés. Generalmente cuando la señal ha sido capturada de un proceso de simulación en computadora las alteraciones no están presentes, gracias al control que se puede tener en el proceso de la simulación y a la ausencia de factores externos que

afectan a las mediciones. Por el contrario, cuando los datos han sido adquiridos en un experimento de identificación de un sistema físico existen varios factores que pueden influir en la adquisición de los datos. En estos casos se hace necesario que la señal reciba un pretratamiento antes de ser procesado por el algoritmo de identificación.

Hay varios problemas potenciales en los datos que deben tomarse en cuenta, entre los que se encuentran:

1. Perturbaciones de alta frecuencia durante el almacenamiento de los datos. Estas perturbaciones tienen una frecuencia por encima de las frecuencias de interés de la dinámica del sistema.
2. Alteraciones eventuales y datos fuera de rango.
3. Desviación y pérdidas, perturbaciones de baja frecuencia, posiblemente de carácter periódico.

Esas deficiencias son, generalmente, producto de algunas variaciones del medio ambiente y otro tipo de fuentes externas que afectan los valores que se están midiendo.

Cuando hay una mala elección del período de muestreo ó, simplemente se tiene la pérdida de información y lo que consecuentemente se podría reflejar como un problema de traslape.

En aplicaciones fuera de línea, se recomienda siempre graficar primero los datos para inspeccionarlos y detectar cualquier deficiencia que pudieran tener.

Esas deficiencias pueden generar señales que esten más allá de nuestro control y que afectan el sistema. Considerando el dominio de los sistemas lineales, se puede suponer que esos efectos pueden estar contenidos en un término $\epsilon(t)$ sumado a la salida del sistema, al cual usualmente se le llama ruido.

Para eliminar el efecto de traslape se recomienda el diseño de un filtro analógico sintonizado a partir de la frecuencia de Nyquist. [Nyquist46]. Este tipo de filtro remueve la parte no deseada de la señal, de tal forma que esta garantizadas las frecuencias de interés.

Una vez que la señal ha pasado por la etapa del prefiltrado se puede optar por dos alternativas, pretratamiento antes de adquirirla, es decir, procesarla mediante otro filtro analógico para eliminar el ruido o capturarla para darle posteriormente el pretratamiento por medio de un filtro digital, el cual se implementa mediante un algoritmo en la computadora.

En este trabajo se diseñó un filtro digital mediante el cual se le da el pretratamiento a la señal. Los filtros digitales se usan principalmente para dos propósitos generales: separación de señales, para señales que han sido mezcladas previamente, y restauración de señales, para señales que han sido distorsionadas. El filtro diseñado para esta aplicación es un filtro pasa bajas del tipo “Convolución de la Transformada Rápida de Fourier” (FFT Convolution) [Smith99]; este filtro opera de la siguiente forma: la señal de entrada se transforma al dominio de la frecuencia usando la transformada rápida de Fourier y se multiplica por el kernel del filtro, la señal resultante se transforma al dominio del tiempo usando la transformada inversa de Fourier. La salida de este filtro es una serie de tiempo, la cual sólo contiene la banda de frecuencias permitida por el filtro.

El filtro usado opera en el dominio de la frecuencia y es de la clase de filtros selectivos en frecuencia. Un filtro selectivo en frecuencia es aquel que deja pasar con exactitud exponenciales complejas (señales oscilatorias) de un conjunto de frecuencias y elimina por completo el resto. La respuesta a la frecuencia, $H(\omega)$, de esta clase de filtros para valores de ω en el rango de $\omega_{min} \leq \omega \leq \omega_{max}$ está dada por la siguiente expresión

$$H(\omega) = \begin{cases} 1, & \text{si } \omega_{min} \leq |\omega| \leq \omega_{max} \\ 0, & \text{si No} \end{cases}$$

donde

$$|\omega| \leq \frac{2\pi}{N}k \quad (4.61)$$

y

ω_{max} = Frecuencia de corte superior.

ω_{min} = Frecuencia de corte inferior.

La Figura 4.10 muestra el kernel de este filtro.

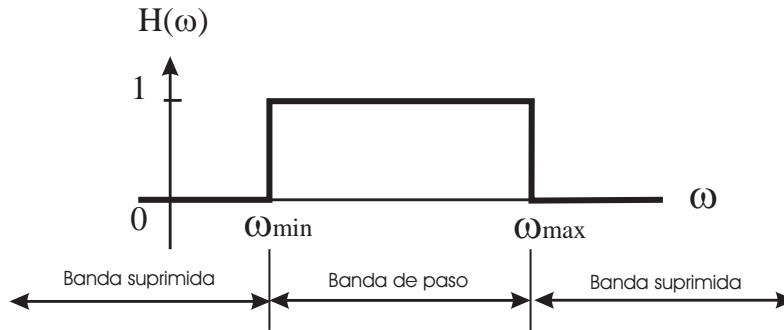


Figura 4.10: Kernel de un filtro pasa banda ideal

Un filtro con la respuesta a la frecuencia mostrada en la Figura 4.10 es conocido como un filtro pasa banda ideal ya que la banda de frecuencias que deja pasar está delimitada por las frecuencias de corte ω_{min} y ω_{max} . La banda de frecuencias que deja pasar el filtro es la banda de paso y la banda de frecuencias eliminadas es la banda suprimida.

La aplicación de este filtro a las señales adquiridas en algún experimento de identificación permite eliminar las componentes del ruido de alta y baja frecuencia, proporcionando, de esta forma, una señal que represente solamente la dinámica del sistema.

4.6. Generador de Respuestas Transitorias

Se diseñó un algoritmo capaz de generar respuestas de sistemas que exhiban un comportamiento cualitativo deseable. Al decir deseable nos referimos a que podemos controlar el número de componentes senoidales y exponenciales presentes en la respuesta pero no así su magnitud y frecuencia, lo cual se genera en forma aleatoria basándose en ciertas reglas que garantizan que cada componente de la respuesta sea

distinguible. Es decir, el problema es determinar los coeficientes y el orden adecuado para la ecuación diferencial de tal forma que se garantice que las características cualitativas deseadas estén presentes en la respuesta. Ya que la respuesta se compone de exponenciales, $E_n(t)$, y senoidales, $ES_n(t)$, se puede diseñar cada una de estas componentes por separado.

4.6.1. Componentes Exponenciales.

Se inicia el diseño con las componentes exponenciales con factores de decaimiento r_1 y r_2 , es decir,

$$E_n(t) = a_1 e^{-r_1 t} + a_2 e^{-r_2 t} \quad (4.62)$$

$$E'_n(t) = -r_1 a_1 e^{-r_1 t} - r_2 a_2 e^{-r_2 t} = 0 \quad (4.63)$$

se resuelve para t

$$t = \frac{\ln\left(-\frac{r_2 a_2}{r_1 a_1}\right)}{-(r_1 - r_2)} \quad (4.64)$$

Es evidente que en la ecuación (4.64) el argumento del logaritmo natural debe ser positivo. Es decir:

$$\frac{r_2 a_2}{r_1 a_1} < 0 \quad (4.65)$$

dado que los valores de los factores de decaimiento r_1 y r_2 son positivos, entonces el signo de la amplitud debe alternarse:

$$\text{signo}(a_1) \neq \text{signo}(a_2) \quad (4.66)$$

Para asegurar que el valor de t sea positivo se debe cumplir que

$$(r_2 | a_2 |) < (r_1 | a_1 |) \quad (4.67)$$

entonces

$$\frac{r_1}{r_2} > \frac{a_2}{a_1} \quad (4.68)$$

se define α como la relación de los factores de decaimiento

$$\alpha = \frac{r_1}{r_2} \quad (4.69)$$

$$a_2 < \alpha a_1 \quad (4.70)$$

En general para n componentes exponenciales con coeficientes de decaimiento $r_1 > r_2 > r_3 > \dots > r_n$, se debe cumplir que

$$\alpha_i a_i > a_{i+1}; \quad i = 1, 2, 3, \dots, n-1 \quad (4.71)$$

donde:

$$\alpha_i = \frac{r_i}{r_{i+1}} \quad (4.72)$$

lo anterior proporciona las condiciones necesarias de las r 's y las a 's que proporcionen la forma cualitativa deseada.

4.6.2. Componentes Senoidales

Para diseñar la componente senoidal, las condiciones para los parámetros de $ES_m(t)$ son las siguientes:

Se parte de la frecuencia más lenta y la designaremos por ω_1 . Se produce el resto de la componente de acuerdo con las siguientes reglas:

$$\omega_1 < \omega_2 < \dots < \omega_i < \dots < \omega_n \quad (4.73)$$

es conveniente tener al menos cuatro períodos completos de la senoidal rápida montada sobre la más lenta. Esto garantiza que sean más distinguibles las características cualitativas de la respuesta diseñada. Es decir:

$$\frac{\omega_2}{\omega_1} \geq 4 \quad (4.74)$$

$$\omega_2 \geq 4\omega_1 \quad (4.75)$$

Estas características permiten generar una serie de tiempo la cual exhiba un comportamiento deseado.

4.7. Conclusiones

En este capítulo se ha presentado el diseño del algoritmo de identificación, dividido en dos partes: identificación cualitativa (determinación de la estructura) e identificación cuantitativa (estimación paramétrica). La primera parte se lleva a cabo mediante el análisis cualitativo de la serie de tiempo que describe el comportamiento transitorio del sistema en estudio. Se presentó el diseño de los filtros mediante los cuales se lleva a cabo la separación de las componentes del sistema. Esta separación se lleva a cabo en dos partes, primero se separan las componentes senoidales iniciando con la componente de mayor frecuencia. Ya que han sido separadas todas las componentes senoidales se procede a separar las componentes exponenciales. Aprovechando que se dispone de cada una de las componentes que se están filtrando, se calculan los parámetros de cada una de estas componentes, llevándose a cabo en forma consecutiva los dos procesos de identificación. Se presentó también la metodología para obtener el modelo una vez que se han estimado los parámetros de cada componente. El diseño contempla hacer la identificación tanto de sistemas reales como de sistemas simulados y para tal efecto se diseñó una función que procesa las señales que se han adquirido de un sistema real, para eliminar el ruido y los datos fuera de rango que afecta y alteran el comportamiento del sistema que se está observando. Finalmente se presenta una función para generar, en forma aleatoria, comportamientos que simulan la respuesta transitoria de sistemas dinámicos.

Capítulo 5

Identificación de Sistemas Lentamente Variantes en el Tiempo.

Los sistemas dinámicos en la vida práctica son de manera inherente variantes en el tiempo. Estas variaciones se deben a los cambios en las condiciones de operación, fluctuaciones y sus efectos en los componentes del sistema. Uno de los sistemas variantes más fáciles de abordar son los de variación lenta en el tiempo, cuyo comportamiento se parece al de los sistemas invariantes durante un corto lapso de tiempo. Las variaciones en las condiciones ambientales son usualmente mucho más lentas que la dinámica de los sistemas. Por lo cual, los sistemas con parámetros que dependen de las condiciones ambientales (temperatura, presión, altitud, etc.) a menudo pueden ser modelados como sistemas lentamente variantes en el tiempo. El envejecimiento y deterioro son otros ejemplos de variaciones lentas de sistemas dinámicos en operación.

La representación matemática de los modelos variantes en el tiempo se puede dar mediante una ecuación diferencial lineal de coeficientes variables:

$$a_n(t) \frac{d^n x(t)}{dt^n} + a_{n-1}(t) \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + a_0(t) x(t) = 0 \quad (5.1)$$

Una de las aproximaciones usadas para el análisis de sistemas lentamente variables es el método de tiempo-congelado introducido en los 60's [Freedman68 et al, Desoer70] para el análisis de estabilidad de sistemas con parámetros lentamente variantes en el tiempo y usado también para identificación y control [LeYi95].

5.1. Formulación del Problema

La idea principal de la aproximación de tiempo-congelado se puede resumir como sigue: Un sistema variante en el tiempo se puede modelar como una secuencia de sistemas invariantes en el tiempo, llamándolos sistemas de tiempo-congelado. El sistema de tiempo-congelado en cada instante representa el comportamiento dinámico de la planta en ese momento; la Figura 5.1 ilustra este principio.

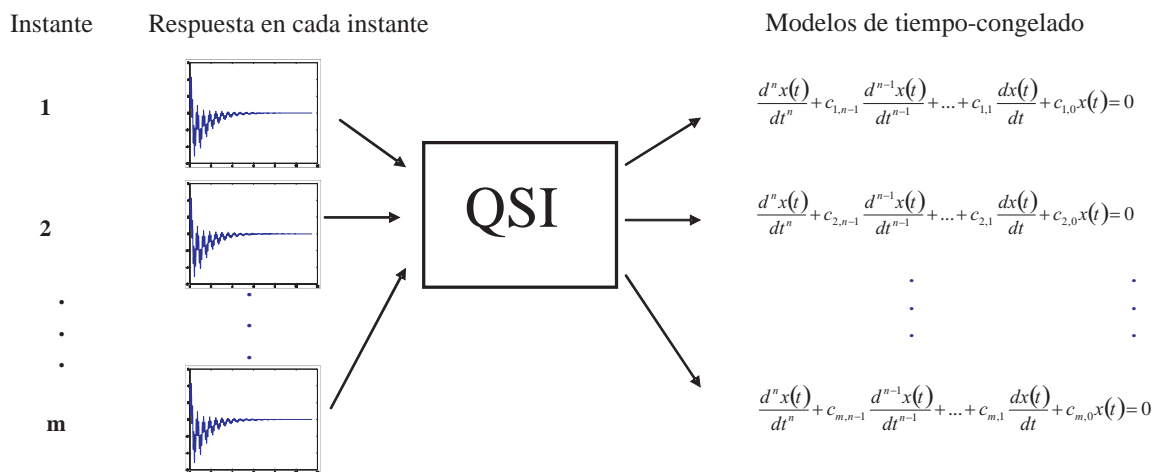


Figura 5.1: Aplicación del principio de tiempo congelado para modelar sistemas variantes en el tiempo usando QSI

El término tiempo-congelado o instante-congelado es usado para indicar que durante un lapso de tiempo se considera que la estructura de un sistema dado no cambia. Esto permite que su respuesta se pueda considerar como la respuesta de un sistema lineal e invariante en el tiempo durante ese lapso de tiempo. Se espera que durante este lapso de tiempo la dinámica transitoria sea mucho más rápida que la variación de

los componentes que definen la estructura del sistema que se está observando. En cada instante congelado, el proceso de identificación se lleva a cabo usando QSI. Los modelos resultantes de la aplicación de QSI, y de la aproximación de tiempo congelado, se organizan en forma consecutiva formando una matriz que describe el comportamiento del sistema en el tiempo, la Figura 5.2 presenta los modelos de tiempo congelado y su correspondiente matriz de coeficientes.

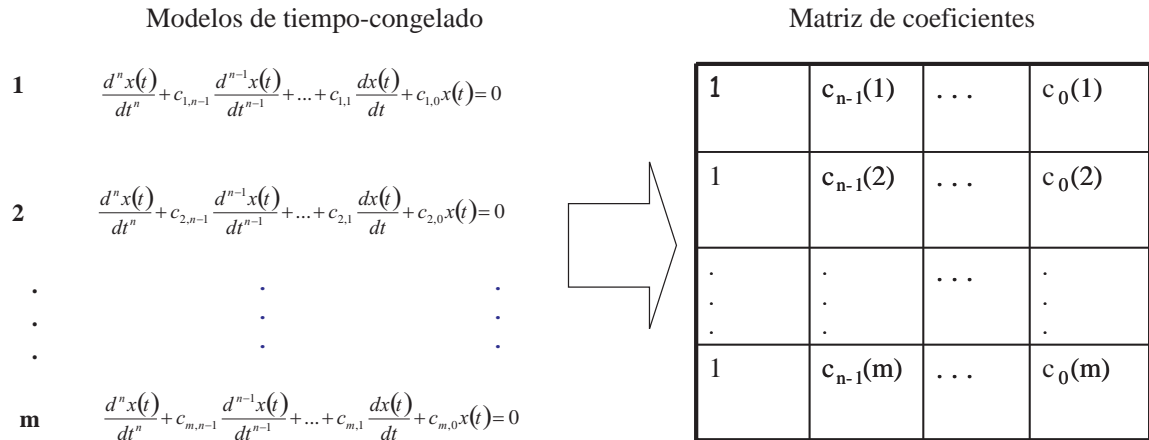


Figura 5.2: Matriz de coeficientes obtenida a partir de los modelos de tiempo congelado

Cada columna de la matriz representa el comportamiento de cada uno de los coeficientes de la EDO que describe al sistema. Es decir, para un sistema de orden n , la primera columna representa el comportamiento del coeficiente del $n - \text{ésimo}$ término, la segunda al del $(n - 1) - \text{ésimo}$ término y así sucesivamente. La Figura 5.3 muestra la relación entre cada columna y los coeficientes de la ecuación diferencial. Los coeficientes (no constantes) son modelados a partir de la matriz de tiempo congelado, aproximando su comportamiento por medio funciones polinomiales.

Existen dos formas de las ecuaciones diferenciales mediante las cuales podemos representar un modelo variante en el tiempo: mediante la forma canónica y la forma normalizada [Wylie95]. La forma canónica es la más utilizada y es como se ilustra en la ecuación 5.1. La forma normalizada se deriva de la forma canónica, una vez que

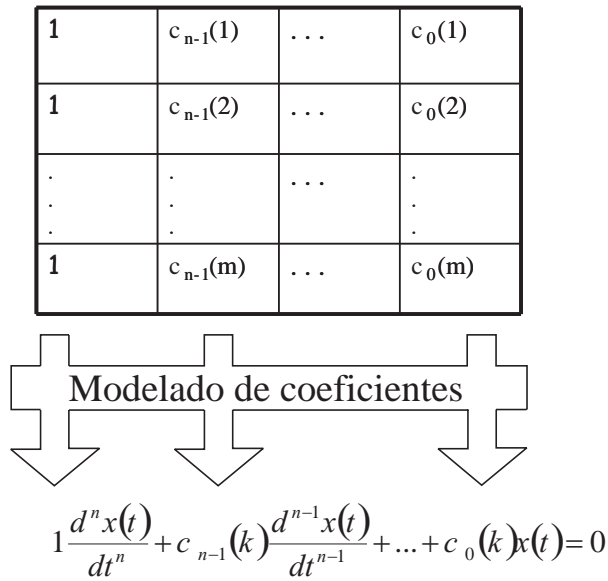


Figura 5.3: Matriz de coeficientes y su relación con los coeficientes de la EDO

se divide toda la ecuación entre el coeficiente del término de mayor orden, como se indica en la ecuación 5.2.

$$\frac{a_n(t)}{a_n(t)} \frac{d^n x(t)}{dt^n} + \frac{a_{n-1}(t)}{a_n(t)} \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + \frac{a_0(t)}{a_n(t)} x(t) = 0 \quad (5.2)$$

Definiendo

$$c_n(k) = \frac{a_n(t)}{a_n(t)} = 1 \quad (5.3)$$

$$c_{n-1}(k) = \frac{a_{n-1}(t)}{a_n(t)} \quad (5.4)$$

\vdots

$$c_0(k) = \frac{a_0(t)}{a_n(t)} \quad (5.5)$$

y sustituyendo en la ecuación 5.2 obtenemos la forma normalizada.

$$1 \frac{d^n x(t)}{dt^n} + c_{n-1}(k) \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + c_0(k) x(t) = 0 \quad (5.6)$$

De esta forma los resultados contenidos en la matriz de coeficientes de la Figura 5.3 nos permiten obtener en modelo descrito por la ecuación 5.1 usando las definiciones dadas en las ecuaciones 5.3, 5.4, y 5.5, donde

$$a_i(t) = \sum_{j=0}^{D_i} p_j t^j \quad (5.7)$$

con $i = 0, 1, \dots, n$. En esta expresión p_j representa los coeficientes del polinomio que aproxima a la función $a_i(t)$ y D_i es el grado más alto de ese polinomio, modelando de esta forma las variaciones de los coeficientes. Es decir, $a_i(t)$ es un polinomio de grado D_i .

Para estimar las funciones que aproximan las variaciones de los coeficientes se usa un algoritmo genético [Goldberg98, Pastor00], de tal forma que el proceso de optimización haga mínimo el valor de la ecuación 5.8.

$$\begin{aligned} \min Error(dif) &= \sqrt{\frac{1}{N} \sum_{k=1}^N [dif]^2} \quad (5.8) \\ dif &= \left[c_n(k) - \frac{a_n(t)}{a_n(t)} \right] + \left[c_{n-1}(k) - \frac{a_{n-1}(t)}{a_n(t)} \right] + \dots \\ &\dots + \left[c_0(k) - \frac{a_0(t)}{a_n(t)} \right] \quad (5.9) \end{aligned}$$

El algoritmo genético lleva a cabo la optimización del grado de los polinomio y sus coeficientes de tal forma que se obtiene la mejor aproximación. La Figura 5.4 presenta el diagrama de bloques del algoritmo genético usado. En el Apéndice C se presenta la descripción del Algoritmo Genético usado en esta tesis.

El formato del cromosoma que define cada individuo en el algoritmo genético es el siguiente:

$$individuo = [\{D_n \ p_{n,D_n} \ \dots \ p_{n,0}\} \ \{D_{n-1} \ p_{n-1,D_{n-1}} \ \dots \ p_{n-1,0}\} \ \dots \ \{D_0 \ p_{0,D_0} \ \dots \ p_{0,0}\}] \quad (5.10)$$

donde

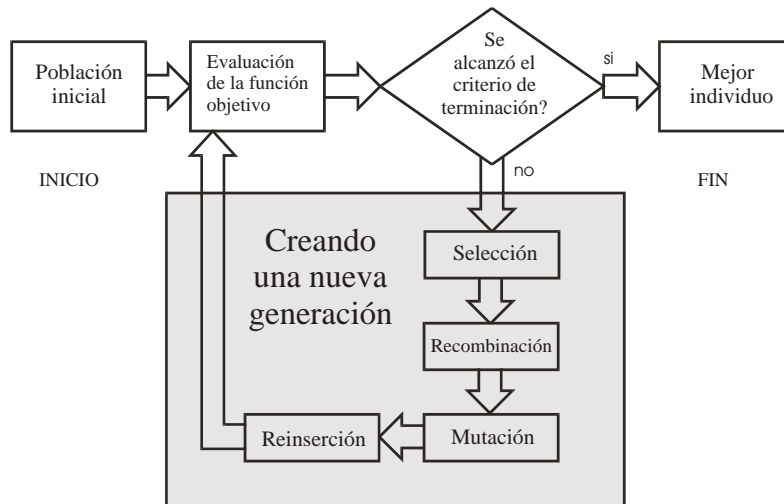


Figura 5.4: Diagrama de bloques del algoritmo genético usado para estimar los coeficientes del polinomio que caracteriza los coeficientes de la EDO

$D_n, D_{n-1}, \dots, D_0 =$ Grados respectivos de los polinomios que definen la variación de los coeficientes $a_n(t), a_{n-1}(t), \dots, a_0(t)$

$p_{n,D_n}, p_{n,D_n-1}, \dots, p_{n,0} =$ Coeficientes del polinomio $a_n(t)$

$p_{n-1,D_{n-1}}, p_{n-1,D_{n-1}-1}, \dots, p_{n-1,0} =$ Coeficientes del polinomio $a_{n-1}(t)$

⋮

$p_{0,D_0}, p_{0,D_0-1}, \dots, p_{0,0} =$ Coeficientes del polinomio $a_0(t)$

5.2. Algoritmo de Identificación de Sistemas Variantes en el Tiempo

Identificar un modelo usando la aproximación de tiempo-congelado y QSI involucra tres elementos fundamentales: los datos, el conjunto de EDO's y las funciones que aproximan el modelo variante en el tiempo. El conjunto de datos, $Data$, es una matriz de tamaño $k \times N$ donde los renglones corresponden a las series de tiempo capturadas de las respuestas transitorias observadas en cada instante congelado. El conjunto de

modelos se obtiene del procesamiento de cada serie de tiempo a través de QSI, y se almacenan en la variable M . Las aproximaciones de los polinomios (ver ecuación 5.7) se determinan de acuerdo con el algoritmo 4.

Algoritmo 4 Identificación de sistemas variantes en el tiempo

```

QSITIMEVARYING(Data)
1    $[k, N] \leftarrow \text{size}(\textit{Data})$ 
2   do
3      $M[i] \leftarrow \text{QSI}(\textit{Data}[i])$ 
4   until  $i = k$ 
5   do
6      $\textit{model} - \textit{TV} \leftarrow \text{GeneticAlg}(M)$ 
7      $\textit{CCN} \leftarrow \text{valida}(\textit{model} - \textit{TV})$ 
8   until  $\textit{CCN} \geq 0.9$ 
9   regresar  $\textit{model} - \textit{TV}$ 

```

Este algoritmo trabaja con los datos organizados en la matriz $Data$. Primero determina el tamaño de esta matriz para establecer el número de veces que se ejecutará el algoritmo QSI (línea 1), de acuerdo con el valor de k , y la cantidad N de datos almacenados en cada evento. Una vez que se procesa cada renglón de la variable $Data$, a través de QSI, se obtienen k modelos descritos por EDO's cuyos coeficientes se almacenan en la matriz M (línea 3). Las columnas de M describen el comportamiento en el tiempo de cada coeficiente de la ecuación característica. Una vez que se tiene el conjunto de modelos, organizados en M , éstos son procesados por medio del algoritmo genético para estimar las funciones que mejor describan el comportamiento de los coeficientes (línea 6). Es decir, estamos identificando las funciones $a_n(t)$,

$a_{n-1}(t), \dots, a_0(t)$ de la ecuación 5.1. y consecuentemente el modelo variante en el tiempo. El modelo variante en el tiempo obtenido se valida (línea 7), si el coeficiente de correlación es mayor que 0.90 se acepta el modelo, en otro caso se repite la estimación de los polinomios hasta alcanzar el criterio de terminación.

La Tabla 5.1 muestra la organización de la matriz *Data*. Y la ecuación 5.11 muestra la estructura de la matriz de coeficientes *M*.

Tabla 5.1: Datos de entrada para el algoritmo QSITimeVarying

Instante	Respuesta
1	$x_1 \dots x_N$
\vdots	$\vdots \quad \vdots \quad \vdots$
k	$x_1 \dots x_N$

$$M = \begin{bmatrix} C_{1,n} & C_{1,n-1} & \cdots & C_{1,0} \\ \vdots & \vdots & \vdots & \vdots \\ C_{k,n} & C_{k,n-1} & \cdots & C_{k,0} \end{bmatrix} \quad (5.11)$$

Para validar las aproximaciones se hace lo siguiente: se evalúan las funciones $a_n(t)$, $a_{n-1}(t), \dots, a_0(t)$ para $t = \{1, 2, \dots, k\}$. Se calcula el coeficiente de correlación entre cada vector resultante y su correspondiente columna en la matriz *M*. Finalmente obtenemos un valor total de correlación promediando las correlaciones obtenidos para cada coeficiente. La salida del algoritmo son los polinomios $a_i(t)$, que más aproximan los datos observados.

El algoritmo se programó en matlab y su ejecución es en el espacio de trabajo en modo de texto, es decir, no tiene una interfaz gráfica. Para ejecutar el programa es necesario disponer del conjunto de datos ordenados como se muestra en la Tabla 5.1 y basta con teclear el siguiente comando en matlab:

```
>> modelovt=qsitimevarying(data);
```

El modelo resultante se almacena en la variable *modelovt*. La aplicación de este algoritmo nos permite obtener los coeficientes de la ecuación 5.1. La sección de re-

sultados presenta dos casos de aplicación de esta metodología. El primer problema es una sistema simulado en simulink, y el segundo problema es un experimento realizado en el laboratorio de sistemas de potencia.

5.3. Conclusiones

Este capítulo presentó la metodología para identificar sistemas lentamente variantes en el tiempo, se propone un algoritmo que permite realizar la identificación usando el principio de tiempo congelado en combinación con el algoritmo QSI y un algoritmo genético, este último se usa para aproximar las funciones que describen el comportamiento de los coeficientes variantes en el tiempo, de la EDO. Al igual que en el caso de sistemas invariantes en el tiempo este algoritmo se programó en matlab, con la restricción de que no tiene una interfaz gráfica con el usuario.

Capítulo 6

Análisis de Complejidad

6.1. Introducción

La solución práctica de un problema requiere por un lado un algoritmo o método de solución y por otro lado un programa o codificación de ese algoritmo en una computadora. Ambos elementos tienen su importancia, pero la del algoritmo es fundamental. Para efectos prácticos, son de suma importancia los recursos físicos necesarios para que un programa se ejecute. Aunque puede haber varios aspectos, los más usuales son el tiempo de ejecución y la cantidad de memoria (espacio).

Para cada problema se determina una medida N de su tamaño (número de datos) y se busca la respuesta en función de dicho valor N . El concepto exacto que mide N depende de la naturaleza del problema. Así, para un vector se suele utilizar como N su tamaño; para una matriz, el número de elementos que la componen; para un grafo, puede ser el número de nodos; en un archivo se suele usar el número de registros, etc.

En este capítulo se efectúa un análisis de complejidad del algoritmo de identificación QSI y del algoritmo de interpolación splines cúbico. La razón por la cual se analiza el algoritmo splines cúbico es por que este algoritmo no fue desarrollado como parte del código de QSI, y sin embargo, se usa para llevar a cabo la interpolación y reconstrucción de la envolvente de la respuesta transitoria y de su atractor.

6.2. Complejidad Computacional

La complejidad computacional estudia los costos de cómputo necesarios para resolver un problema; entendiéndose por costos los recursos de espacio de almacenamiento y de tiempo de cómputo. Debido a que es necesario un estudio que no tenga alguna dependencia con arquitecturas de cómputo, el tiempo se mide en función al tamaño del problema que un algoritmo ha de resolver.

Al hablar de complejidad, no se está refiriendo a la dificultad que se tendría para diseñar un programa, o a lo rebuscado de un algoritmo. La teoría de complejidad tiene que ver con dos medidas de desempeño: tiempo y espacio.

La complejidad temporal tiene que ver con el tiempo que tarda un programa para ejecutarse. La complejidad espacial estudia la cantidad de espacio de almacenamiento que es necesaria para ejecutar el programa. La mayoría de estudios de complejidad están orientados hacia el desempeño de los algoritmos en función del tiempo. En el análisis que desarrollaremos acerca de QSI enfatizaremos en la complejidad temporal.

Al analizar la complejidad de un algoritmo, el tiempo está expresado en término de pasos de computación elementales (asignaciones, comparaciones, multiplicaciones, etc.). Por ejemplo, una operación de asignación ocupa una unidad de tiempo para ejecutarse, un ciclo ocupa el número de iteraciones en que está definido, etc.

6.2.1. Complejidad Temporal

El tamaño de los datos de entrada, N , es el parámetro que define al tiempo de ejecución de un programa, es posible describir este tiempo como una función de la entrada $T(N)$.

Cuando se define la complejidad en función al tamaño de la entrada, no se está considerando a los datos que forman esa entrada, sino al tamaño de los mismos. En la práctica se observa que un algoritmo ocupa diferente tiempo de ejecución para entradas del mismo tamaño, pero con diferentes datos. Por ejemplo, un método de

ordenamiento puede tardar menos tiempo si su entrada está ordenada, o tardar mucho mayor tiempo incluso con los mismos valores, pero presentados en desorden (es decir, con el mismo tamaño de entrada tiene diferente comportamiento).

Por lo anterior, se adopta el criterio de tomar siempre, como base de análisis de la complejidad de un algoritmo, el caso en el cual consume el mayor tiempo de ejecución, es decir, el peor caso.

6.2.2. Notación Asintótica

Esta notación define un conjunto de funciones tales que satisfacen ciertos criterios y se usa para caracterizar la complejidad de tiempo y espacio de un algoritmo. Está definida en términos cuyo dominio es el conjunto de los números naturales.

Sea $g(N)$ una función dada, que va de los enteros positivos a los reales positivos [Cormen01 et al].

$$g : \mathbb{N} \rightarrow \mathbb{R} \quad (6.1)$$

Definimos las funciones $O(g(N))$, $\Omega(g(N))$ y $\Theta(g(N))$, de la siguiente manera:

1. $T(N) \in O(g(N))$ si y sólo si existen dos enteros positivos $N_0 \in \mathbb{N}$ y $c > 0$, tal que para todo $N \geq N_0$

$$0 \leq T(N) \leq cg(N) \quad (6.2)$$

2. $T(N) \in \Omega(g(N))$ si y sólo si existen dos enteros positivos $N_0 \in \mathbb{N}$ y $c > 0$, tal que para todo $N \geq N_0$

$$0 \leq cg(N) \leq T(N) \quad (6.3)$$

3. $T(N) \in \Theta(g(N))$ si y sólo si existen los enteros positivos $N_0 \in \mathbb{N}$, $c_1 > 0$ y $c_2 > 0$, tal que para todo $N \geq N_0$

$$0 \leq c_1g(N) \leq T(N) \leq c_2g(N) \quad (6.4)$$

La Figura 6.1 muestra en forma gráfica las funciones $O(g(N))$, $\Omega(g(N))$ y $\Theta(g(N))$. En cada gráfica, el valor de N_0 delimita el mínimo valor posible para el cual las funciones cumplen con las condiciones dadas. La Figura 6.1 a) muestra la notación $O(g(N))$ la cual expresa una cota superior para $T(N)$. La Figura 6.1 b) muestra la notación $\Omega(g(N))$, la cual proporciona una cota inferior para $T(N)$. La Figura 6.1 c) muestra la notación $\Theta(g(N))$, la cual limita el crecimiento de la función tanto por arriba como por debajo, por la función $g(N)$.

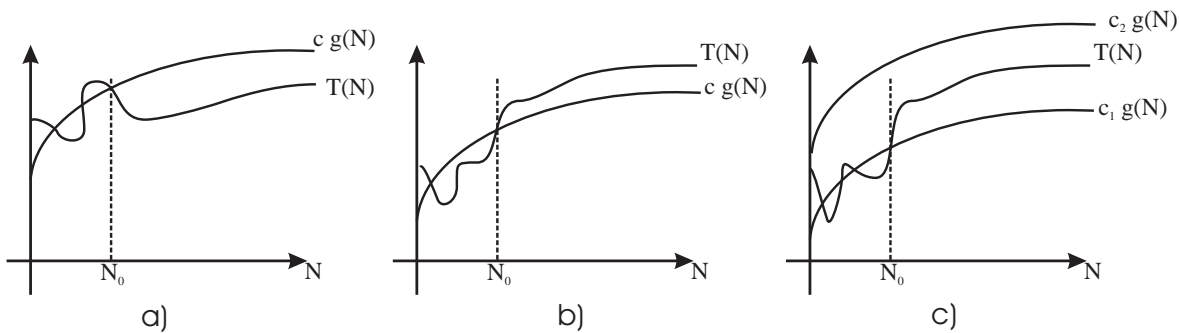


Figura 6.1: Forma gráfica de las notaciones $O(g(N))$, $\Omega(g(N))$ y $\Theta(g(N))$

6.2.3. Complejidad de un Algoritmo

La función que define el tiempo de ejecución de un programa proporciona información interesante para clasificar los diferentes algoritmos que existen para resolver problemas. Con esta función es posible comparar el desempeño de diferentes algoritmos desarrollados para un problema en particular.

Para simplificar el estudio de la complejidad, se han adoptado ciertas convenciones en su notación, una de ellas es la del concepto de orden, que indica, de forma simple, el grado de complejidad de un algoritmo, sin considerar por completo la función de tiempo. Es decir, para un algoritmo que tiene una función de tiempo $T\left(\frac{N^2-N}{2}\right)$ se dice que es de orden N^2 o simplemente $O(N^2)$. A continuación se enlistan las propiedades más usuales de la notación O : [Cormen01 et al]

- Si $|f(N)| \leq |g(N)|$, entonces $f(N) = O(g(N))$. En particular, $f(N) = O(f(N))$.
- Si $\lim \frac{f(N)}{g(N)} < \infty$, $f(N) = O(g(N))$.
- $N^a = O(N^b)$ si y sólo si $a \leq b$.
- $a^N = O(b^N)$ si y sólo si $a \leq b$.
- $\log(N) = O(N^e)$ para todo $e > 0$.
- $O(f(N)) + O(g(N)) = O(\max\{|f(N)|, |g(N)|\})$.
- $O(f(N))O(g(N)) = O(f(N)g(N))$.
- $O(cN) = O(N)$.

De lo anterior se puede deducir que, si $P(N) = z_0N^k + z_1N^{k-1} + z_2N^{k-2} + \dots + z_k$ es un polinomio en N , el término dominante (de mayor grado) es el que define su crecimiento asintótico. Es decir, $P(N) = O(N^k)$.

A continuación analizaremos la complejidad computacional de QSI por medio de cada una de las funciones que la componen.

6.3. Complejidad del Algoritmo Splines Cúbico

El algoritmo splines cúbico se usa para estimar valores de la envolvente por medio de un proceso de interpolación (ver Figura 4.4 en la sección 4.2.1). A partir de esta envolvente es posible calcular el vector de puntos medios, $m(t)$, el cual representa el comportamiento submuestreado del atractor de la envolvente, es decir, el residuo $X^*(t)$. Para reconstruir $X^*(t)$ usamos el algoritmo splines.

La forma como funciona el algoritmo splines es relativamente simple [Junheng92]: Dada una serie de tiempo de L puntos (p_0, p_1, \dots, p_L) . El intervalo $[p_0, p_L]$ se particiona

en secciones $[x_0, x_1], \dots, [x_{L-1}, x_L]$ donde x_0 es la coordenada en x de p_0 y x_L es la coordenada en x de p_L . Cada sección es enlazada por un polinomio cúbico $f_i(x)$, (ver Figura 6.2). De esta forma el spline total consiste de la unión de los diferentes polinomios cúbicos $f_i(x)$.

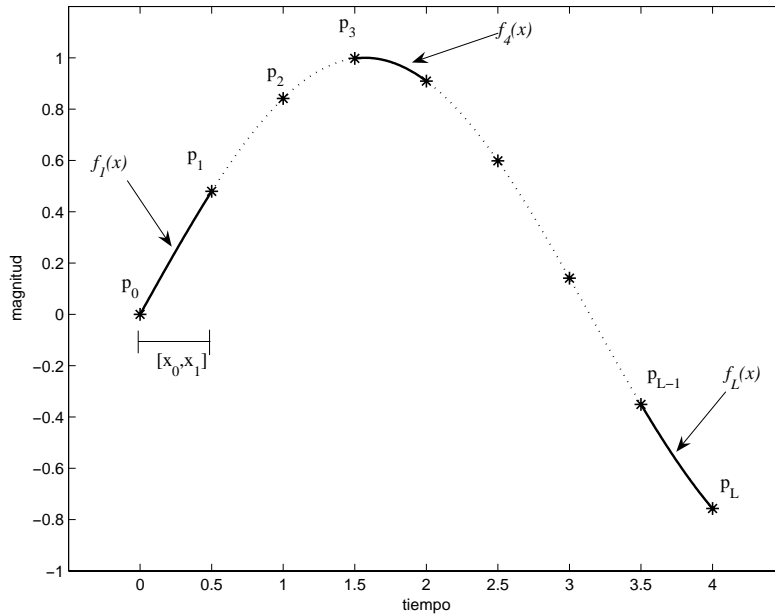


Figura 6.2: Particionamiento de una señal y enlace de cada punto mediante un spline

La Figura 6.2 muestra un ejemplo de como se lleva a cabo la interpolación de una señal con 9 puntos ($L = 8$), mediante el algoritmo splines. Se ilustra como se hace la partición, donde cada segmento está delimitado por dos puntos adyacentes de la señal. Se ilustran los enlaces generados para los segmentos $[x_0, x_1]$, $[x_3, x_4]$ y $[x_{L-1}, x_L]$, en línea continua, definidos por las funciones $f_1(x)$, $f_4(x)$ y $f_L(x)$. La línea discontinua ilustra el resto de los segmentos de la interpolación.

Durante la interpolación de cada punto de la serie de tiempo, cada punto se procesa en forma individual y para cada uno se genera un spline. El spline, total, asociado con una curva de L puntos contiene $L \cdot S$ puntos donde S es un multiplicador que determina el número total de puntos en cada segmento interpolado (spline). Es obvio

que la suavidad del spline depende directamente del multiplicador S . Es decir, entre más grande sea al valor de S más suave será la curva resultante. En el caso de la interpolación usada en el proceso de identificación presentado en esta tesis el producto $(L \cdot S) \in O$ donde N es el tamaño de la respuesta del sistema que se está identificando.

Cada uno de los pasos del algoritmo splines (particionamiento y unión) tienen una complejidad $O(L)$, (ver [Lou94]), donde L es el número de puntos de los datos en la serie de tiempo original. El cálculo de todos los segmentos del algoritmo splines requiere en total $O(k \cdot S \cdot L)$ pasos, donde k es el grado del spline más uno (en el caso cúbico $k = 4$). El proceso de unión de cada spline es secuencial en el total de puntos a unir, y su complejidad es $O(S \cdot L)$. La complejidad global del algoritmo depende únicamente de la unión de los segmentos, ya que k es una constante relacionada únicamente por la generación de polinomios de interpolación y no depende de L , y además, $L \cdot S \in O N$; el algoritmo es esencialmente lineal en el número de datos de la serie de tiempo, es decir, $T(N) \in O(N)$.

6.4. Complejidad de la función TPAFilter.

La Figura 6.3 presenta el pseudocódigo de la función TPAFilter y su correspondiente análisis de complejidad. Esta función recibe como entrada la señal X , de tamaño N , el orden del modelo, k y la matriz de parámetros P . Estos dos últimos serán cero al inicio de la ejecución del algoritmo. Las variables máx y mín son el número de extremos, máximos y mínimos respectivamente, de X , de forma tal que máx + mín < N . El número de veces que se ejecuta el ciclo del filtro es función directa de la cantidad de componentes senoidales y está dado por la variable n_2 .

El orden de TPAFilter resulta $O(N \cdot n_2)$, donde n_2 representa el número de componentes senoidales. De la ecuación 4.6 tenemos que

$$n = n_1 + 2n_2 \tag{6.5}$$

donde n es el orden del sistema, n_1 el número de componentes exponenciales y n_2 el

Algoritmo	Complejidad
TPAFilter(X,k,P)	
if oscillatory(X)	O(1)
do	O(n ₂)
Xtr _{max} =determine _{maxima} (X)	O(N)
Xtr _{min} =determine _{minima} (X)	O(N)
k=k+2	O(1)
Envelope _{max} =CubicSpline(Xtr _{max})	O(N)
Envelope _{min} =CubicSpline(Xtr _{min})	O(N)
m=average(Envelope _{max} ,Envelope _{min})	O(N)
S=X-X*	O(N)
P=[P;SinParam(S)]	O(N)
until X* no oscillatory	O(1)
else	O(1)
X*=X	O(1)
return(X*,k,P)	O(1)
Complejidad total	O(N*n ₂)

Figura 6.3: Complejidad de la función TPAFilter

número de componentes senoidales, por lo tanto

$$O(n_2) \in O(n) \quad (6.6)$$

Como $n \ll N$, y además el número de veces, n_2 , que se ejecuta el ciclo no depende del número de puntos entonces podemos considerar como si el contador estuviera dado en forma explícita y por lo tanto n_2 se puede considerar como una constante. Es decir,

$$O(N \cdot n_2) \in O(N) \quad (6.7)$$

El peor caso para la función TPAFilter sería cuando el orden, n resultara del tamaño de la entrada, lo cual es poco probable ya que en términos prácticos se espera que el orden de un sistema sea $n \leq 10$.

6.5. Complejidad de la función *ExpFilter*.

La función *ExpFilter* recibe como entrada el residuo $X^*(t)$ proveniente de la función *TPAFilter*, cuyo tamaño es N . La variable n_1 se refiere al número de componentes exponenciales de la señal y siempre se cumple que $n_1 \ll N$. La Figura 6.4 presenta el pseudocódigo de la función *ExpFilter* y su análisis de complejidad.

Algoritmo	Complejidad
<pre> ExpFilter(X,k,P) Xtr=cout extrema of X if Xtr>0 do k=k+1 P_i=InflectionPoint(X) P=ExpParam(X,P_i) X=X-Exponential(P) Xtr=Xtr-1 while Xtr>0 else k=k+1 P=ExpParam(X) X=X-Exponential(P) return (X,k,P) </pre>	<pre> O(N) O(1) O(n₁) O(1) O(N) O(N) O(N) O(1) O(1) O(1) O(1) O(1) O(N) O(N) O(1) </pre>
Complejidad total	$O(n_1 \cdot N)$

Figura 6.4: Analisis de complejidad de la función *ExpFilter*

En los ciclos con contador explícito, se pueden distinguir dos casos, que el tamaño N forme parte de los límites o que no. Si el ciclo se realiza un número fijo de veces, independiente de N , entonces la repetición sólo introduce una constante multiplicativa que puede absorberse en el análisis de complejidad. En este caso la función *ExpFilter* ejecuta $n_1 - 1$ veces la eliminación de las componentes exponenciales en cuyo caso el orden del ciclo se ve afectada por la constante $(n_1 - 1)$ que como se menciona anteriormente se puede absorber.

$$O(n_1 \cdot N) \in O(N) \quad (6.8)$$

De esta manera la complejidad del algoritmo es $O(N)$.

6.6. Complejidad Computacional del Algoritmo QSI

La complejidad de llamar a un procedimiento viene dada por la complejidad del contenido del procedimiento en sí. El costo de llamar no es sino una constante que se puede obviar inmediatamente dentro de nuestros análisis. El algoritmo QSI opera fundamentalmente haciendo la llamada de las dos funciones anteriores para llevar a cabo la obtención del modelo, por lo que cada uno de estos procedimientos contribuyen con un orden $O(N)$. La Figura 6.5 presenta este algoritmo y el análisis de complejidad correspondiente.

Algoritmo	Complejidad
QSI(X)	-
degree=0	$O(1)$
P=0; Parameter Matrix	$O(1)$
(X,k,P)=TPAFilter(X,degree,P)	$O(N)$
(X,k,P)=ExpFilter(X,degree,P)	$O(N)$
return Model(deegree,X,P)	$O(1)$
Complejidad total	$O(N)$

Figura 6.5: Análisis de complejidad del Algoritmo QSI

Se ha determinado la complejidad computacional del algoritmo QSI, por medio del análisis de las dos funciones principales que la componen, *TPAFilter* y *ExpFilter*. El análisis realizado reporta una complejidad lineal para cada uno de estos procedimientos y consecuentemente para el algoritmo QSI.

Una buena observación de estos resultados es que en cada una de las instrucciones analizadas siempre se consideró el peor caso. En términos prácticos la consecuencia de estas consideraciones es que es posible conseguir un mejor desempeño si ocurre el

caso promedio. Mejor aún, si ocurre el mejor caso.

6.7. Complejidad de QSI vs. Mínimos Cuadrados

Si comparamos la medida de complejidad para QSI con el método más usado en la identificación de sistemas, el método de los mínimos cuadrados, tenemos que mientras el algoritmo QSI tiene una complejidad lineal, el algoritmo de los mínimos cuadrados recursivos (RLS por sus siglas en inglés de “Recursive Least Squares”) tiene una complejidad cuadrática [An-Yeu94]. Posteriormente se han desarrollado algoritmos que mejoran el desempeño de los mínimos cuadrados, tal es el caso de los algoritmos “Fast RLS”, que logra una complejidad $O(N \log N)$, y “Split RLS”, con una complejidad $O(N)$ [An-Yeu94].

El algoritmo “Split RLS” es un algoritmo que usa el principio “divide y vencerás”. Por medio de la aplicación de particionamientos múltiples este algoritmo segmenta el conjunto de datos y hace las correspondientes optimizaciones para cada partición. En el caso de sistemas consistentes de ecuaciones lineales el método hace el particionamiento por renglón, y resuelve aplicando algún método iterativo. Si el sistema no es consistente el particionamiento se hace formando submatrices y la solución se obtiene por medio de la aplicación de un método iterativo basado en el gradiente.

6.8. Conclusiones

En este capítulo se hace un análisis de complejidad de las funciones usadas en proceso de identificación mediante QSI. Se incluye una discusión acerca de las propiedades del análisis de algoritmos y como aplican en el análisis de cada una de las funciones usadas. De este análisis se puede concluir que el algoritmo desarrollado es eficiente y comparable con una de las técnicas más usadas en el área de la identificación de

Capítulo 7

Resultados

En este capítulo, se presentan tres ejemplos de aplicación para el caso de sistemas LIT. Se presentan además dos casos de estudio para sistemas variantes en el tiempo. En el primer ejemplo se identifica la estructura y los parámetros de un sistema cuya respuesta ha sido generada por medio del módulo de diseño de experimentos. El segundo caso presenta la identificación de un circuito RLC. El tercer caso de estudio es la identificación de una línea de transmisión de un sistema eléctrico de potencia. En el segundo y tercer caso las respuestas analizadas no fueron adquiridas desde un sistema físico, sino más bien son resultado de simulaciones llevadas a cabo en simulink, de Matlab. El cuarto y quinto casos de estudio se refieren a sistemas con variaciones lentas en el tiempo; el cuarto caso es la representación matemática de un sistema masa-resorte simulado en Matlab; por medio de esta simulación se genera la respuesta transitoria y se aplica el algoritmo de identificación. El quinto caso es un experimento de laboratorio de una línea de transmisión. En este experimento simulamos el envejecimiento de la línea por medio de el incremento de la resistencia en cada uno de los instantes congelados que se observan.

7.1. Coeficiente de Correlación

La medida usada para validar los modelos obtenidos es el coeficiente de correlación [Walpole89]. Para la aplicación de este método de validación es necesario llevar a cabo la simulación del modelo obtenido para comparar su respuesta con la señal observada. La ecuación 7.1 muestra la expresión para calcular el coeficiente de correlación CCN

$$CCN^2 = \frac{\sum (\hat{y} - \bar{y})^2}{\sum (y - \bar{y})^2} \quad (7.1)$$

$$CCN = \pm \sqrt{\frac{\sum (\hat{y} - \bar{y})^2}{\sum (y - \bar{y})^2}} \quad (7.2)$$

donde \hat{y} es la señal estimada, y es la señal observada y \bar{y} es el promedio de la señal observada.

El coeficiente de correlación CCN compara la varianza de \hat{y} con la de y . Este valor está confinado dentro del intervalo $[-1, 1]$. Cuando $CCN = 1$ representa una perfecta correlación positiva entre los datos. Cuando $CCN = -1$ representa una perfecta correlación negativa entre los datos, es decir, los datos varían en direcciones opuestas. Cuando $CCN = 0$, no hay correlación entre los datos. Los valores intermedios describen correlaciones parciales. Por ejemplo sí $CCN = 0.88$ significa que el ajuste entre la salida del modelo y los datos es razonablemente buena. La Tabla 7.1 muestra los rangos de los posibles valores de CCN y sus respectivos niveles de aceptación.

Con la misma finalidad de validar el modelo obtenido se implementó el Índice Cualitativo de Similitud [Cuberos02 et al], como una aplicación independiente de QSI.

¹La notación usada en estadística para especificar el coeficiente de correlación es r . Pero como en esta tesis se está usando r para definir el factor de decaimiento de las exponenciales, usaremos CCN para denotar el coeficiente de correlación numérica y CCC para denotar al coeficiente de correlación cualitativa.

Tabla 7.1: Niveles de exactitud del coeficiente de correlación.

Coeficiente de Correlación.	Exactitud
1.0	Excelente
de 0.90 a 0.99	Muy buena
de 0.80 a 0.89	Buena
≤ 0.79	Regular

7.1.1. Índice Cualitativo de Similitud

Este índice es una medida que permite comparar entre las formas cualitativas de dos series de tiempo, tomando en cuenta la evolución de los valores de las series. Esta aplicación usa índices de similitud definidos por etiquetas que representan un comportamiento cualitativo. Cada etiqueta representa un rango de valores, los cuales pueden considerarse similares desde el punto de vista cualitativo. Cada etiqueta se obtiene a partir de la transición entre dos valores adyacentes, de la serie de tiempo. Esta etiqueta depende de la magnitud y signo de la transición. Cada etiqueta está representada por medio de un caracter simple, por lo tanto la evolución de la serie se traduce a una cadena de caracteres. Finalmente se define un índice de similitud de las series de tiempo de acuerdo a la similitud de las cadenas de caracteres obtenidas, la Tabla 7.2 presenta la manera como se definen las etiquetas, de acuerdo con la razón de cambio entre dos puntos adyacentes. La Figura 7.1 muestra una serie de tiempo y los símbolos que identifican sus etiquetas.

Tabla 7.2: Definición de las etiquetas

Etiqueta	Rango	Símbolo
Incremento Grande	$\frac{1}{5}, \infty$	H
Incremento medio	$\frac{1}{25}, \frac{1}{5}$	M
Incremento pequeño	$0, \frac{1}{25}$	L
Sin variación	0	0
Decremento pequeño	$0, -\frac{1}{25}$	l
Decremento medio	$-\frac{1}{25}, -\frac{1}{5}$	m
Decremento grande	$-\frac{1}{5}, -\infty$	h

Ya que el índice cualitativo de similitud está definido dentro del rango de [0,1]

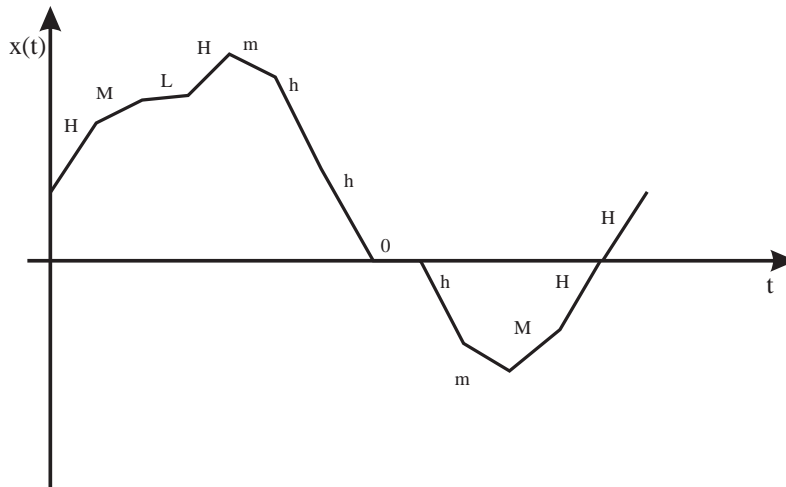


Figura 7.1: Etiquetado de una serie de tiempo, de acuerdo con la magnitud y signo de la transición de sus valores

y fundamentalmente expresa de que manera están correlacionadas cualitativamente dos series de tiempo, lo definiremos como el Coeficiente Cualitativo de Correlación y lo denotaremos por CCC .

Este coeficiente expresa la medida de que tan bien se ajustan las formas cualitativas de dos series de tiempo, la Figura 7.2 muestra dos series de tiempo diferentes en magnitud pero cualitativamente similares. Esta medida de correlación se usa para comparar cualitativamente la señal observada con la señal obtenida mediante la simulación del modelo.

En los casos de estudio presentados en este capítulo se ha usado esta aplicación para medir que tan bien se ajustan las respuestas de los modelos identificados con los comportamientos observados, desde el punto de vista cualitativo.

7.2. Sistemas Lineales Invariantes en el Tiempo

En esta sección presentamos los diferentes casos de estudio correspondientes a los sistemas Lineales Invariantes en el Tiempo (LIT). Estos tres casos son simulaciones realizadas en Matlab y Simulink.

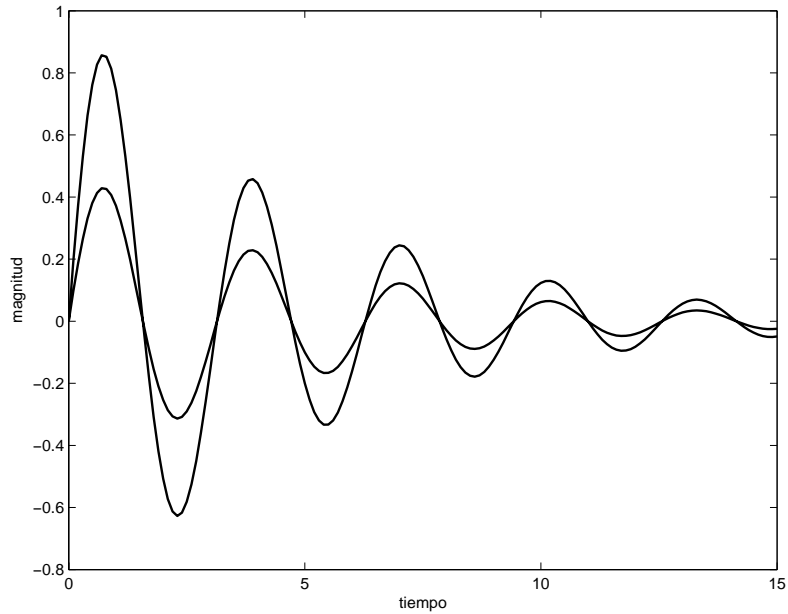


Figura 7.2: Gráfica de dos series de tiempo cualitativamente similares

7.2.1. Caso de Estudio 1: Señal de Prueba

Este ejemplo es una serie de tiempo generada por el módulo llamado diseño de experimentos (ver sección 4.6). La Figura 7.3 presenta la señal que se está analizando. Este ejemplo ilustra de una manera muy simple el proceso que sigue el algoritmo QSI para obtener el modelo.

La Figura 7.4 presenta el proceso de eliminación de las componentes oscilatorias. La línea más delgada corresponde a la señal original. La línea de grosor medio corresponde al residuo una vez que se ha filtrado la componente de más alta frecuencia (ver Figura 7.5). Finalmente la línea más gruesa representa el residuo exponencial, una vez que se han extraído todas las componentes oscilatorias.

La Figura 7.6 presenta el residuo exponencial y sus componentes, las líneas punteadas representan los componentes exponenciales de primer orden que la conforman.

El algoritmo QSI agrega dos unidades al orden del modelo por cada componente oscilatoria eliminada y una unidad por cada componente exponencial. El algoritmo

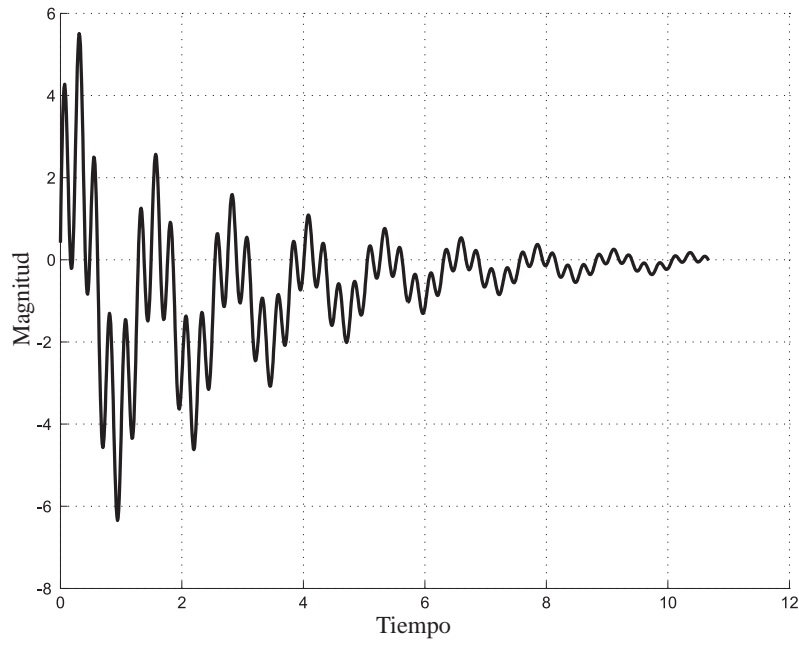


Figura 7.3: Respuesta de un sistema LIT

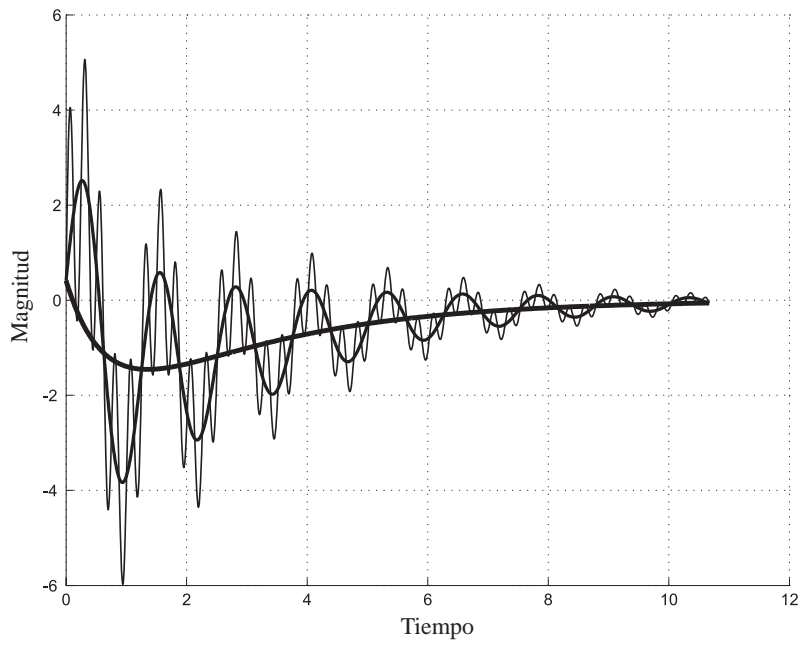


Figura 7.4: Eliminación de componentes por QSI

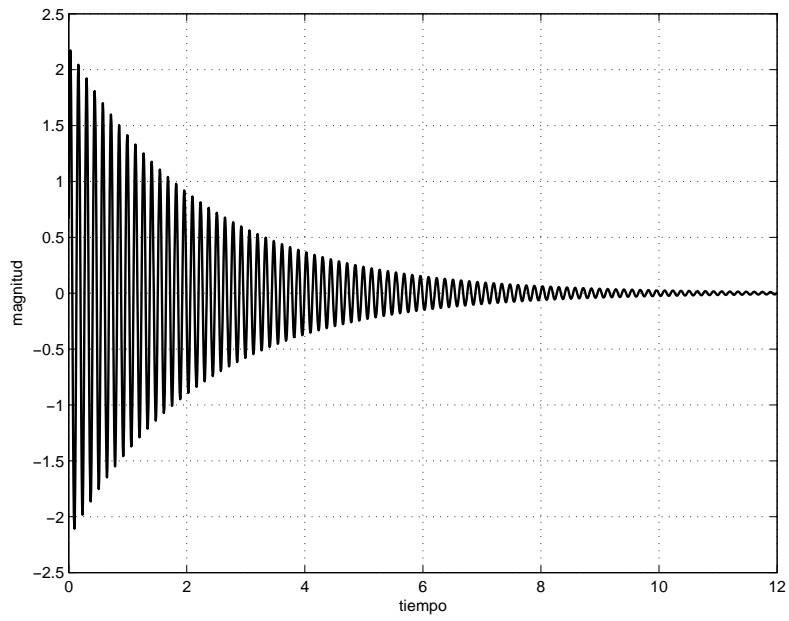


Figura 7.5: Primera componente filtrada por QSI en el caso de estudio 1.

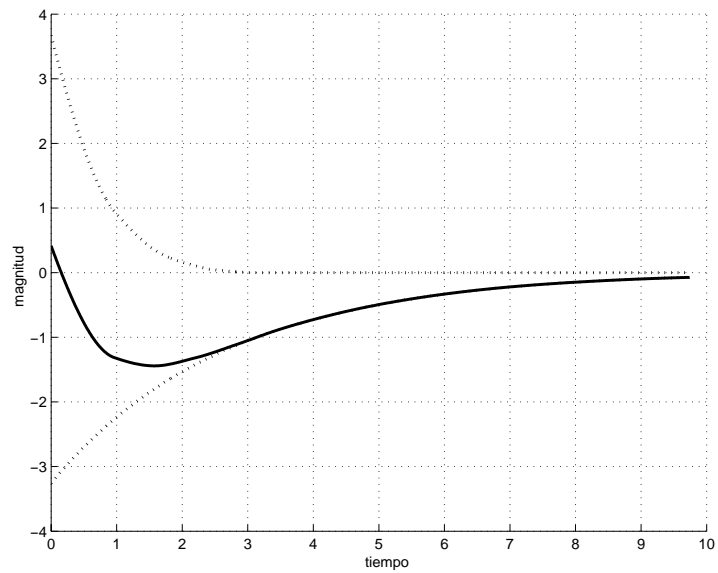


Figura 7.6: Componentes de primer orden del residuo exponencial

QSI ha determinado que el sistema LIT más simple capaz de exhibir el comportamiento observado es de orden 6.

$$c_6 \frac{d^6 x(t)}{dt^6} + c_5 \frac{d^5 x(t)}{dt^5} + \dots + c_1 \frac{dx(t)}{dt} + c_0 x(t) = 0 \quad (7.1)$$

La solución para la ecuación (7.1) tiene la forma mostrada en la ecuación 7.2, que es la forma extendida de la ecuación

$$X(t) = \sum_{i=1}^2 a_i e^{-r_j t} \text{sen}(\omega_i t + \phi) + \sum_{i=3}^4 a_i e^{-r_i t} \quad (7.2)$$

Cuyos parámetros se dan a continuación:

a	r	ω	ϕ
2.2021	-0.3210	± 25.0107	0
3.3375	-0.3211	± 5.0141	0
-3.2579	-0.3766	0	0
3.6731	-1.4711	0	0

El modelo resultante tiene la forma mostrada en la ecuación 7.1. La identificación paramétrica nos proporciona los siguientes coeficientes para la ecuación diferencial.

La Figura 7.7 muestra la respuesta del modelo y la respuesta observada.

$$\begin{aligned} \frac{d^6 x(t)}{dt^6} + 3.1 \frac{d^5 x(t)}{dt^5} + 654.2 \frac{d^4 x(t)}{dt^4} + 1622 \frac{d^3 x(t)}{dt^3} + \\ + 16926.7 \frac{d^2 x(t)}{dt^2} + 29413.4 \frac{dx(t)}{dt} + 8570.6 x(t) = 0 \end{aligned} \quad (7.3)$$

El valor del CCC para este caso fué:

$$CCC = 1.0 \quad (7.4)$$

y el CCN fué:

$$CCN = 0.99 \quad (7.5)$$

Los resultados de este experimento fueron satisfactorios, ya que se obtuvo un valor del coeficiente de correlación numérico muy cercano a la unidad.

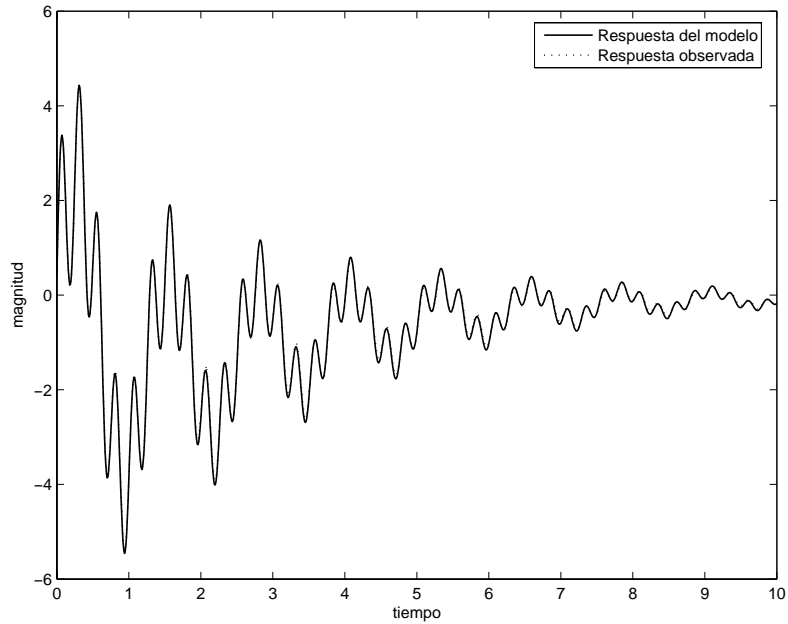


Figura 7.7: Comparación entre la respuesta del modelo y la respuesta observada.

7.2.2. Caso de Estudio 2: Circuito RLC

En este caso se determinó el modelo del circuito RLC mostrado en la Figura 7.8. La señal usada para excitar el circuito y observar su respuesta fue un escalón (ver Figura 7.9).

Los valores usados en el experimento, correspondientes a la Figura 7.8 son: $R_s = 50\Omega$. $L = 0.1\text{Henry}$ $C = 1000\mu\text{F}$ $V_s = 100\text{V}$.

Con esos valores se obtiene la ecuación diferencial que describe este sistema (ver ecuación 7.6). Es decir, el modelo esperado es

$$\frac{d^2i(t)}{dt^2} + 20\frac{di(t)}{dt} + 10000i(t) = 0 \quad (7.6)$$

La Figura 7.9 muestra la respuesta del sistema, la señal observada es la corriente en la bobina L .

Después de ejecutar QSI, los parámetros obtenidos son:

a	r	ω	ϕ
20.0092	-9.9932	± 99.5074	0

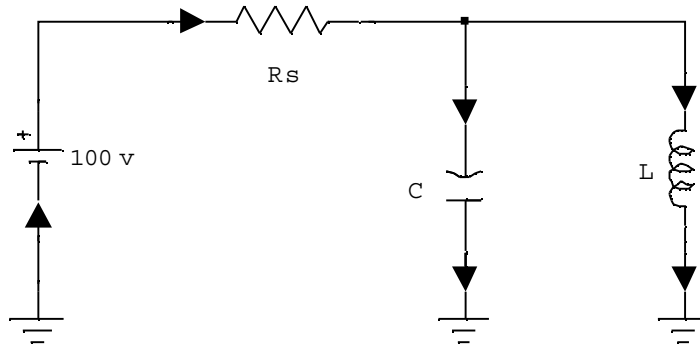


Figura 7.8: Circuito RLC para el caso de estudio 2

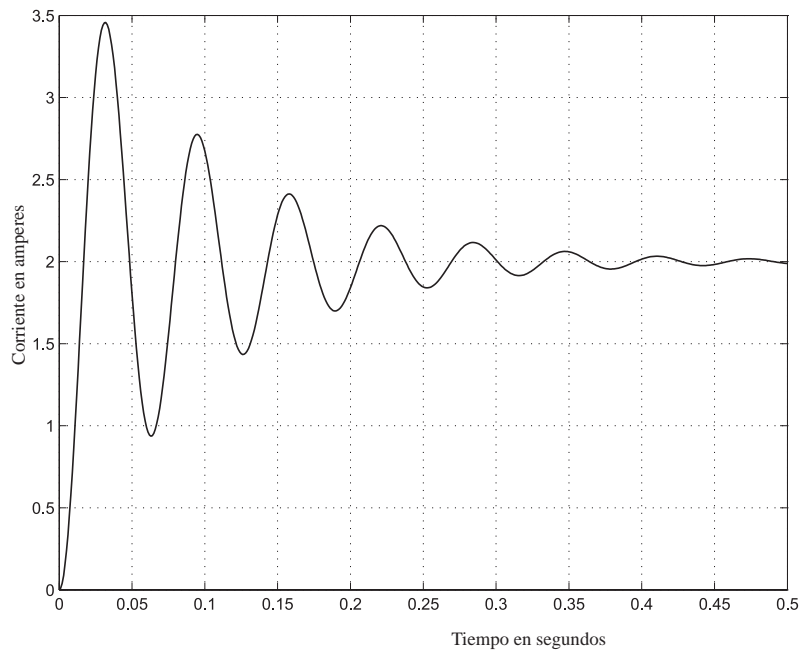


Figura 7.9: Respuesta al escalón del circuito RLC

Es decir se esta identificando una senoidal amortiguada dada por:

$$X(t) = 20e^{-9.9} \sin(99.54t) \quad (7.7)$$

El modelo resultante se muestra en la ecuación 7.8

$$\frac{d^2i(t)}{dt^2} + 19.9864\frac{di(t)}{dt} + 10001i(t) = 0 \quad (7.8)$$

El coeficiente cualitativo de correlación para este caso de estudio es:

$$CCC = 1.0 \quad (7.9)$$

Y el coeficiente de correlación numérico:

$$CCN = 0.99 \quad (7.10)$$

Si se compara la ecuación 7.6 con la ecuación 7.8 se puede ver que los coeficientes del modelo obtenido son muy cercanos a los coeficientes del modelo esperado. Adicionalmente, tenemos dos medidas de calidad que permiten evaluar que tan bien reproduce el modelo la respuesta observada. El valor del CCC indica que la forma de la respuesta del modelo estimado y la forma de la respuesta observada están perfectamente correlacionadas, es decir, sus formas cualitativas son iguales. El valor del CCN permite concluir que el modelo obtenido es muy bueno, de acuerdo con la aproximación de la tabla 7.1.

7.2.3. Caso de Estudio 3: línea de Transmisión

El siguiente caso de estudio se refiere a un modelo lineal de una línea de transmisión de potencia. La Figura 7.10 muestra el modelo de simulink usado en este caso de estudio. En este caso se analiza el transitorio debido a la desconexión de la carga.

La señal capturada es el voltaje en las terminales del capacitor C_2 (Ver Figura 7.11). El transitorio inicia en $t = 0.039$ segundos y termina en $t = 0.1166$ segundos, la ventana de tiempo usada para el proceso de identificación esta comprendida dentro de este intervalo.

La respuesta transitoria, de la linea de transmisión, una vez que se ha desmontado del voltaje de la fuente de alimentación, se muestra en la Figura 7.12.

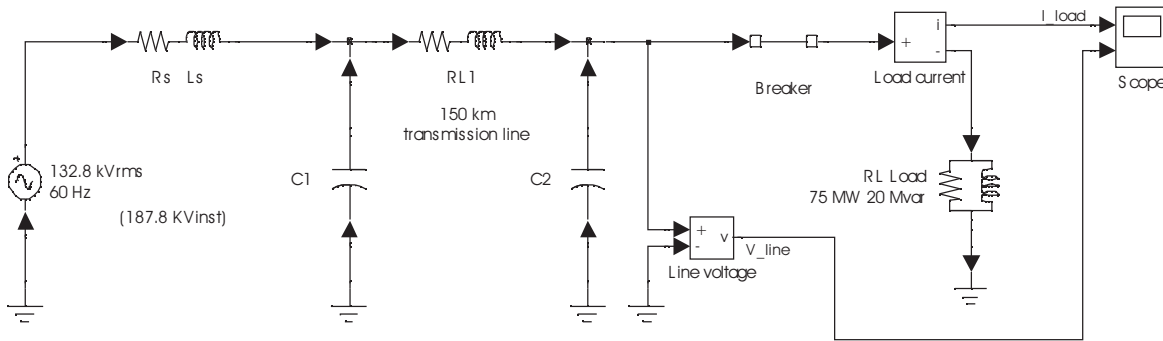


Figura 7.10: Modelo de Simulink para una línea de transmisión monofásica

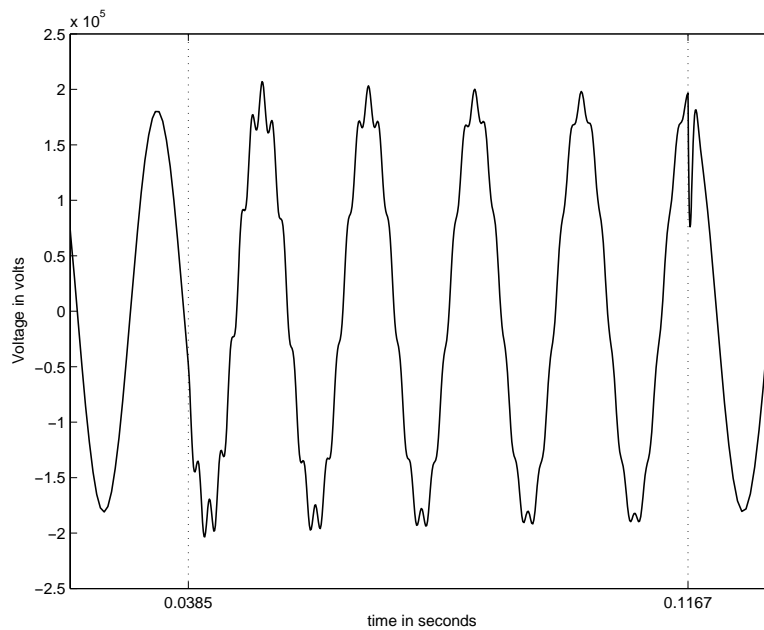


Figura 7.11: Voltaje en la línea de transmisión del caso de estudio 3

Los valores de los elementos del circuito son: $R_s = 2.64 \Omega$. $L_s = 70mH$. $R_1 = 5.2 \Omega$. $L_1 = 138mH$. $C_1 = C_2 = 0.967 \mu F$.

Estos valores nos llevan a la siguiente ecuación diferencial (modelo esperado):

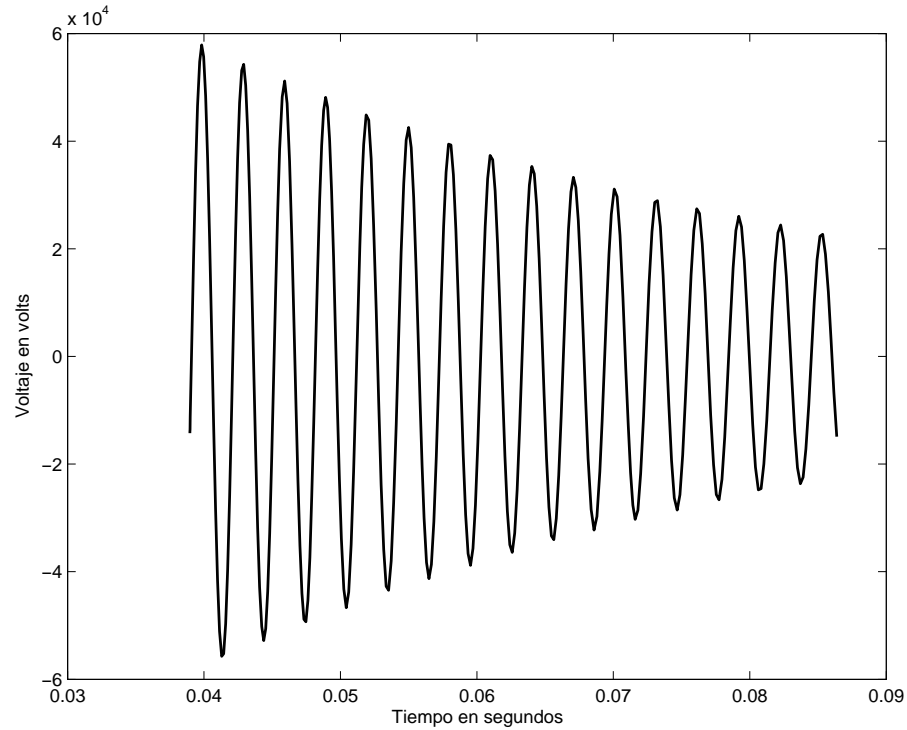


Figura 7.12: Respuesta transitoria de la línea de transmisión del caso de estudio 3

$$\begin{aligned}
 & 6.76E^{-8} \frac{d^4 v_{C2}(t)}{dt^4} + 5.1E^{-6} \frac{d^3 v_{C2}(t)}{dt^3} + \\
 & + \frac{d^2 v_{C2}(t)}{dt^2} + 76.94 \frac{dv_{C2}(t)}{dt} + 74.9v(t) = 0
 \end{aligned} \tag{7.11}$$

Los parámetros obtenidos por QSI son:

$$\begin{array}{cccc}
 a & r & \omega & \phi \\
 1.3E5 & -20.2953 & \pm 2074.86 & 0.5836
 \end{array}$$

El modelo resultante se muestra en la ecuación 7.12

$$\frac{d^2}{dt^2} v_{C2}(t) + 40.5907 \frac{d}{dt} v_{C2}(t) + 43.05E5 v_{C2}(t) = 0 \tag{7.12}$$

El coeficiente cualitativo de correlación es:

$$CCC = 0.99 \tag{7.13}$$

Y el coeficiente de correlación numérico es:

$$CCN = 0.97 \quad (7.14)$$

La Figura 7.13 muestra la comparación entre la respuesta del modelo y la respuesta observada en la línea de transmisión.

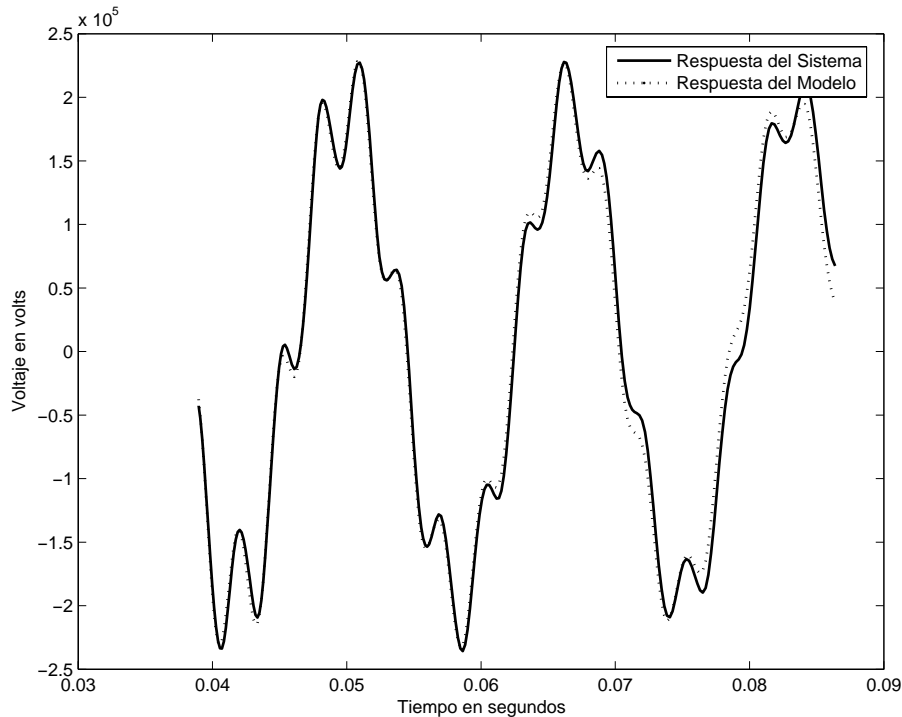


Figura 7.13: Comparación entre la respuesta del modelo y la respuesta observada

El modelo obtenido en este ejemplo es interesante, ya que hay diferencias entre el orden del modelo esperado y el orden del modelo obtenido, sin embargo el CCC y el CCN indican que este modelo es adecuado ya que las dos medidas de correlación son mayores que 0.90 (ver la tabla 7.1). Es decir estamos obteniendo un modelo de segundo orden que reproduce la dinámica de la planta en forma aceptable. A continuación se presenta un discusión acerca de la diferencia entre los valores esperados y los obtenidos; el método presentado está basado en la observación de la respuesta transitoria, el algoritmo va descomponiendo la respuesta en sus componentes más

simples conforme los “ve” en la señal. Si algún elemento del sistema tiene una contribución pequeña de tal forma que no sea distinguible en la salida, el algoritmo de identificación no podrá “ver” esa contribución y consecuentemente no será modelada. Otro de las posibles razones es el error implícito en los cálculos numéricos que se realizan durante el filtrado de la señal, ya que la detección de los extremos se hace mediante una aproximación de la derivada. y los valores de la frecuencia y el factor de decaimiento de cada componente se hacen en función de estos valores. Finalmente se puede decir que el algoritmo propuesto produce un modelo capaz de reproducir la dinámica observada. Es decir, es un modelo que no necesariamente modela los parámetros de la planta sino únicamente su comportamiento.

7.3. Sistemas Variantes en el Tiempo

En esta sección se analizan los casos correspondientes a sistemas con variaciones lentas en el tiempo. El 4° caso es una simulación llevada a cabo en simulink y el 5° caso es un experimento llevado a cabo en un laboratorio de sistemas de potencia.

7.3.1. Caso de Estudio 4: Sistema Masa-Resorte

Considere el sistema masa resorte con parámetros que varían lentamente en el tiempo, y cuyo modelo se describe por la ecuación diferencial 7.15.

$$M_a(t) \frac{d^2x(t)}{dt^2} + Fr(t) \frac{dx(t)}{dt} + K(t)x(t) = 0 \quad (7.15)$$

donde $M_a(t)$ es la masa, $Fr(t)$ es la fricción y $K(t)$ es la constante del resorte. Aunque este es un caso de estudio simulado podríamos pensar que propiedades del sistema cambian con el tiempo debido al envejecimiento y desgaste de los componentes.

Aplicamos el algoritmo 4. La matriz de coeficientes obtenida se presenta en la Tabla 7.3. Nótese que se usa la letra mayúscula K para denotar la constante del resorte y la letra minúscula k para denotar el instante congelado.

Tabla 7.3: Coeficientes del sistema masa-resorte

t_k	$C_2(t_k)$	$C_1(t_k)$	$C_0(t_k)$
1	1	0.1642	1.9264
2	1	0.1846	2.1597
3	1	0.2086	2.5798
4	1	0.2121	2.9247
5	1	0.2469	3.3513
6	1	0.2826	3.744
7	1	0.3302	4.3944
8	1	0.398	5.4008
9	1	0.4755	6.4748
10	1	0.5843	7.9399
11	1	0.7543	9.9699

El conjunto de modelos que se obtiene aplicando QSI tiene la forma de polinomios mónicos; es decir, el coeficiente del término de mayor orden es unitario. Considérese el modelo ilustrado en la ecuación 7.15, de acuerdo con las definiciones dadas en las ecuaciones 5.3, 5.4 y 5.5, los coeficientes de la ecuación característica, obtenida por QSI, están dados por:

$$C_2(t_k) = \frac{M_a(t_k)}{M_a(t_k)} = 1 \quad (7.16)$$

$$C_1(t_k) = \frac{Fr(t_k)}{M_a(t_k)} \quad (7.17)$$

$$C_0(t_k) = \frac{K(t_k)}{M_a(t_k)} \quad (7.18)$$

Una vez que se obtiene el modelo para cada instante k , el algoritmo estima las funciones que describen el comportamiento de cada uno de los coeficientes. En este caso, las funciones son descritas por un cociente de polinomios, tal como se describe en las ecuaciones 7.17 y 7.18.

Como se describió en la sección 5.1, se propone un polinomio para modelar el comportamiento de los coeficientes. Es decir, las funciones que describen esos comportamientos estarán dados por:

$$M_a(t_k) = A_{M_a} t_k^{D_{M_a}} + \dots + B_{M_a} t_k + C_m \quad (7.19)$$

$$Fr(t_k) = A_{Fr} t_k^{D_{Fr}} + \dots + B_{Fr} t_k + C_{Fr} \quad (7.20)$$

$$K(t_k) = A_K t_k^{D_K} + \dots + B_K t_k + C_K \quad (7.21)$$

donde D_i con $i = m, Fr, K$ es el orden del polinomio.

El algoritmo genético debe determinar el orden y los coeficientes de las funciones que corresponden a $M_a(t_k)$, $Fr(t_k)$ y $K(t_k)$, de tal forma que 7.23 sea minimizada.

$$dif(k) = \left(\frac{A_{Fr} t_k^D + \dots + B_{Fr} t_k + C_{Fr}}{A_{M_a} k^D + \dots + B_{M_a} k + C_{M_a}} - C_1(t_k) \right) \quad (7.22)$$

$$+ \left(\frac{A_K t_k^D + \dots + B_K t_k + C_K}{A_{M_a} t_k^D + \dots + B_{M_a} t_k + C_{M_a}} - C_0(t_k) \right) \\ \sum dif^2(t_k) \quad (7.23)$$

Con la finalidad de comparar el desempeño del algoritmo genético [Pastor00] el ajuste se llevó a cabo usando, además, el método de los mínimos cuadrados no lineales [Dennis00]. Los resultados obtenidos se presentan en la Tabla 7.4. En este caso las funciones polinomiales dadas en las ecuaciones 7.19, 7.20 y 7.21 resultaron de segundo orden y fueron suficientes para modelar el comportamiento de los coeficientes.

Tabla 7.4: Coeficientes de las funciones $M_a(t)$, $Fr(t)$ y $K(t)$

Mínimos Cuadrados No Lineales			
	A	B	C
$M_a(t)$	-0.003136	-0.003849	0.744887
$Fr(t)$	0.001056	0.006308	0.126297
$K(t)$	0.007828	0.169841	1.444667
Algoritmos Genéticos			
	A	B	C
$M_a(t)$	-0.003062	-0.004699	0.744448
$Fr(t)$	0.001099	0.005	0.124295
$K(t)$	0.007999	0.166196	1.45

Sustituyendo los coeficientes del polinomio obtenido por medio del método de mínimos cuadrados en las ecuaciones 7.19, 7.20 y 7.21, se obtienen las expresiones dadas en las ecuaciones 7.24, 7.25 y 7.26.

$$M_a(t) = -0.003136t^2 - 0.003849t + 0.744 \quad (7.24)$$

$$Fr(t) = 0.001056t^2 + 0.006308t + 0.1262 \quad (7.25)$$

$$K(t) = 0.007828t^2 + 0.169841t + 1.44 \quad (7.26)$$

Y sustituyendo los resultados obtenidos por medio de los algoritmos genéticos en las ecuación 7.19, 7.20 y 7.21, se obtienen las expresiones dadas en las ecuaciones 7.27, 7.28 y 7.29.

$$M_a(t) = -0.003062t^2 - 0.004699t + 0.744 \quad (7.27)$$

$$Fr(t) = 0.001099t^2 + 0.005t + 0.1242 \quad (7.28)$$

$$K(t) = 0.007999t^2 + 0.166196t + 1.45 \quad (7.29)$$

El coeficiente de correlación entre la respuesta del sistema original y el modelo estimado fue $CCN = 0.99$ para el modelo obtenido con algoritmos genéticos y $CCN = 0.98$ con el modelo obtenido con mínimos cuadrados.

La Figura 7.14 presenta el comportamiento en el tiempo de los coeficientes del modelo. Por simplicidad solo se graficó el comportamiento de los coeficientes estimado por los algoritmos genéticos. Lo anterior se justifica en base al análisis de las ecuaciones para $m(t)$, $Fr(t)$ y $K(t)$, donde se puede observar que la diferencias entre la función obtenida por mínimos cuadrados y la obtenida por algoritmos genéticos es pequeña de forma tal que al graficarlas no se observan diferencias distinguibles.

En la Figura 7.14 podemos observar que los tres parámetros del modelo son variantes en el tiempo, mientras que las variaciones para $m(t)$ y $Fr(t)$ son variaciones muy lentas, casi constante en el caso de $Fr(t)$, $K(t)$ varía en una proporción mayor

en forma casi lineal. Nótese en las ecuaciones 7.26 y 7.29 como las contribuciones del término cuadrático es pequeña comparada con la contribución del término lineal.

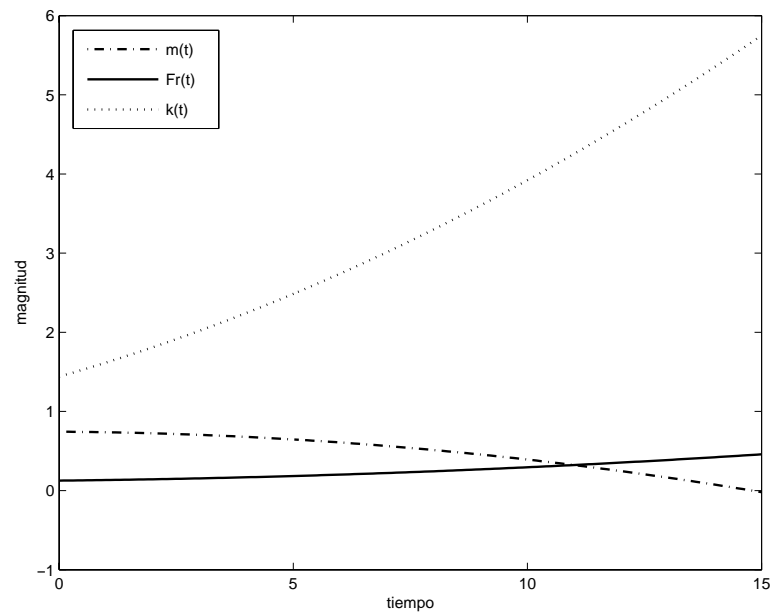


Figura 7.14: Comportamiento de los coeficientes estimado por medio de algoritmos genéticos

Para llevar a cabo la validación del modelo estimado se sustituyen las funciones dadas en las ecuaciones 7.24, 7.25 y 7.26 y se simula el modelo para obtener la respuesta del modelo estimado por medio de mínimos cuadrados. De igual manera se sustituyen las funciones dadas en las ecuaciones 7.27, 7.28 y 7.29 y se hace la simulación para obtener, ahora, la respuesta correspondiente al modelo identificado por medio de algoritmos genéticos. Las respuestas obtenidas en cada caso se grafican y comparan con la respuesta observada. La Figura 7.15 muestra las respuestas de los modelos identificados y la respuesta exhibida por el sistema observado.

Los resultados obtenidos son satisfactorios, ya que las pruebas de validación dan resultados muy cercanos a la unidad. La idea de llevar a cabo la optimización usando dos métodos es para tener una comparación del desempeño de los algoritmos genéticos contra un método clásico. La Figura 7.15 ilustra la similitud de las tres señales.

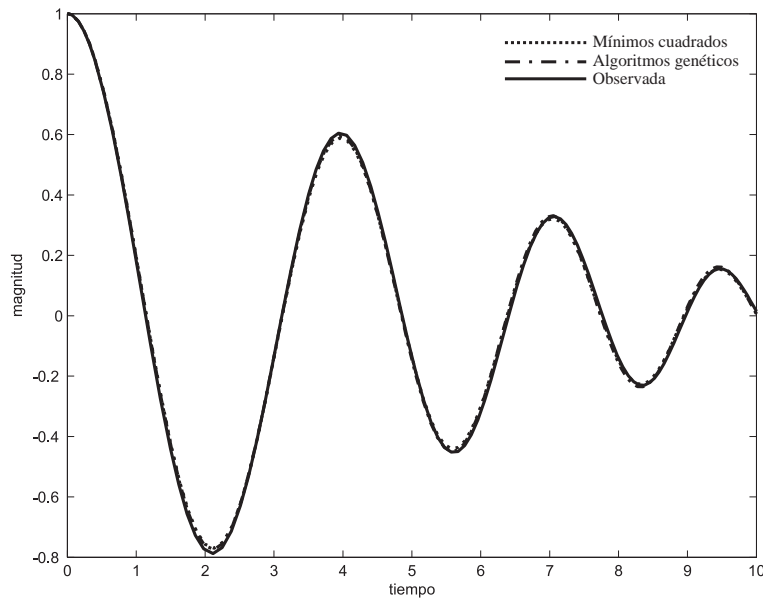


Figura 7.15: Respuestas de los dos modelos estimados comparadas con la respuesta observada

7.3.2. Caso de Estudio 5: Línea de Transmisión

Este experimento se desarrolló en el laboratorio de sistemas de potencia de la División de Estudios de Postgrado de la FIE. El experimento consiste en la captura del efecto transitorio en una línea de transmisión durante la desconexión de la carga. La Figura 7.16 muestra en a) el digrama monofásico y en b) el circuito equivalente. El equipo usado fué una consola experimental marca LabVolt con una fuente de CA ajustada a 20 volts; para capturar los datos del transitorio se usó la tarjeta de adquisición NI PCI5112 de National Instruments, (100 MHz, 100 MS/s 8-bit Digitizer). El modelo usado para la línea de transmisión fue el modelo π mostrado en la Figura 7.16 b). Los valores de los elementos de este modelo fueron; $V_s = 20v$, $C_1 = 1.017\mu F$, $C_2 = 0.967\mu F$, $L = 29.65mH$, y R varía como se muestra en la Tabla 7.5.

El efecto de envejecimiento en una línea de transmisión fué simulado por medio de una resistencia variable R . Ajustamos la resistencia a un valor dado (ver la Tabla 7.5), energizamos la línea de transmisión y entonces desconectamos la carga. Una vez que

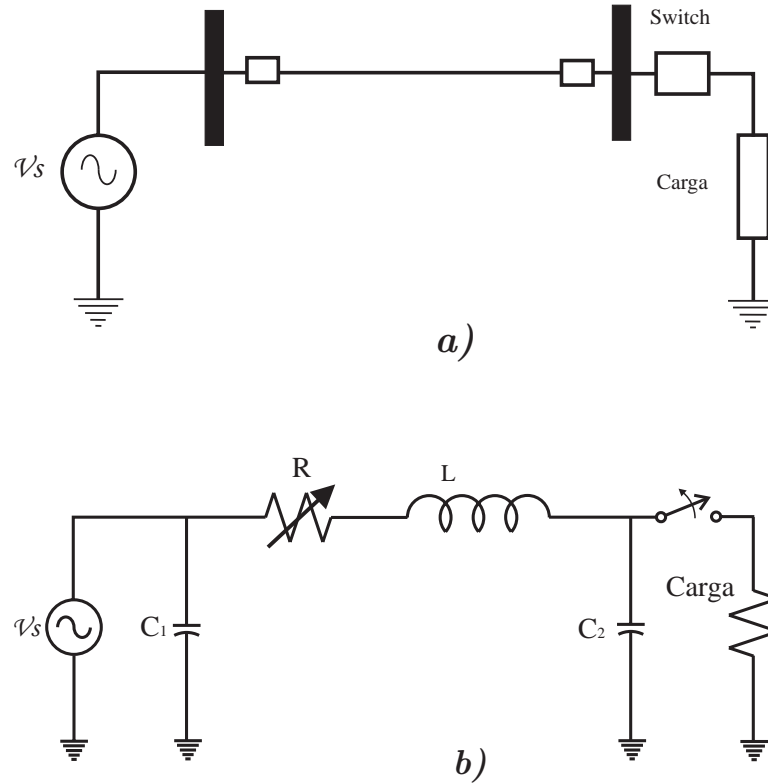


Figura 7.16: Línea de transmisión a). Diagrama monofásico b). Circuito equivalente

desconectamos la carga iniciamos la captura del efecto transitorio. Este experimento se repitió para los valores de R mostrados en la Tabla 7.5. Durante el experimento la variable observada fue el voltaje en C_2 .

Como se mencionó en la sección 4.5, durante la adquisición de los datos en un experimento, los datos generalmente no están en buena forma para ser procesados, y por lo tanto se hace necesario aplicar un pretratamiento para eliminar el ruido y otros componentes que puedan afectar el proceso de identificación. La Figura 7.17 muestra la señal adquirida y el detalle muestra el resultado del proceso del filtrado.

Una vez que la señal capturada es filtrada, usamos el algoritmo mostrado en la Figura 4 para procesar esta señal. La Tabla 7.6 muestra la matriz de coeficientes generada por el algoritmo QSI.

El modelo π de la línea de transmisión expresada como EDO se muestra en la

Tabla 7.5: Valores de R para el experimento de la línea de transmisión

t	R
t_{k_1}	0.386
t_{k_2}	0.396
t_{k_3}	0.406
t_{k_4}	0.426
t_{k_5}	0.436
t_{k_6}	0.476
t_{k_7}	0.516
t_{k_8}	0.596
t_{k_9}	0.776
$t_{k_{10}}$	1.216

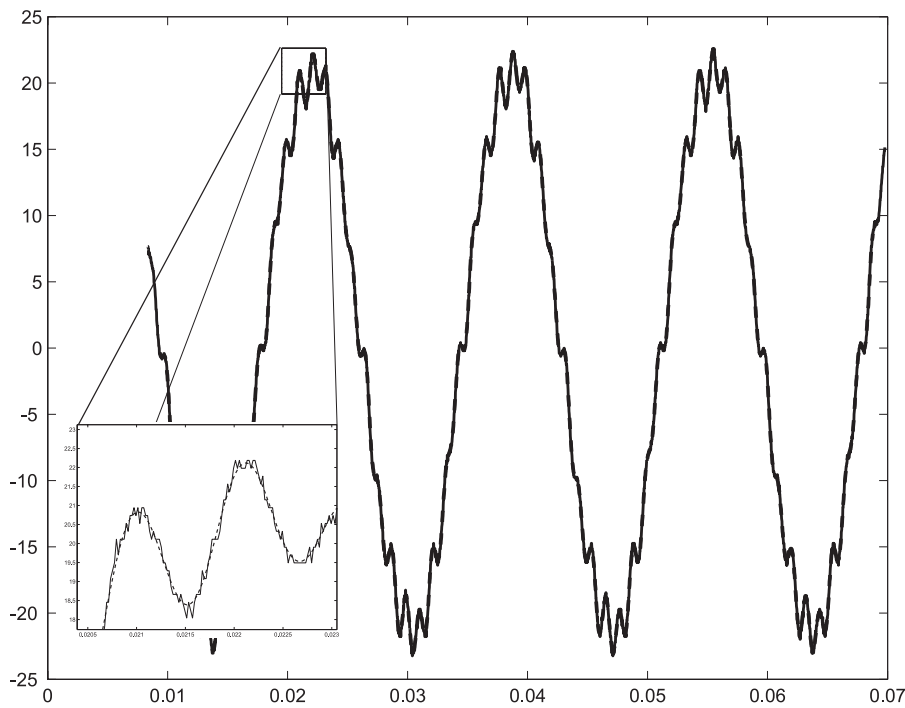


Figura 7.17: Señal capturada y detalle del proceso de filtrado

ecuación 7.30

$$L(t) C(t) \frac{d^2 v_{c_2}}{dt^2} + R(t) C(t) \frac{dv_{c_2}}{dt} + v_{c_2} = v_s \quad (7.30)$$

de acuerdo a lo expuesto en la sección 5.1 los coeficientes $C_i(t)$ estimados por QSI, están dados por las definiciones de las ecuaciones 5.3, 5.4 y 5.5, por lo tanto

Tabla 7.6: Coeficientes del experimento de la línea de transmisión

k	$C_2(k)$	$C_1(k)$	$C_0(k)$
1	1	11.33780349	32147873.02
2	1	11.63152897	31834739.03
3	1	11.92525445	32076947.89
4	1	12.5127054	31982739.73
5	1	12.80643088	32087805.38
6	1	13.9813328	31982739.78
7	1	15.15623471	31955757.71
8	1	17.50603855	32018490.09
9	1	22.79309717	32087809.44
10	1	35.71701824	32018492.74

$$C_2(k) = \frac{L(k) C(k)}{L(k) C(k)} = 1 \quad (7.31)$$

$$C_1(k) = \frac{R(k) C(k)}{L(k) C(k)} = \frac{R(k)}{L(k)} \quad (7.32)$$

$$C_0(k) = \frac{1}{L(k) C(k)} \quad (7.33)$$

El algoritmo genético deberá determinar las funciones para $R(t)$, $L(t)$ and $C(t)$ de tal forma que la ecuación 7.34 sea minimizada.

$$\left(\frac{R(k)}{L(k)} - C_1(k) \right) + \left(\frac{1}{L(k) C(k)} - C_0(k) \right) = dif(k) \quad (7.34)$$

Siguiendo el procedimiento descrito en la sección 5.1, el algoritmo genético determina el orden y los coeficientes de las funciones que determinan a los coeficientes. Finalmente el mejor ajuste se logra mediante polinomios de cuarto grado; la Figura 7.18 presenta las aproximaciones proporcionadas por los polinomios probados para el coeficiente $C_1(t)$.

Los polinomios que mejor describen el comportamiento de los datos de la tabla 7.6 son los siguientes:

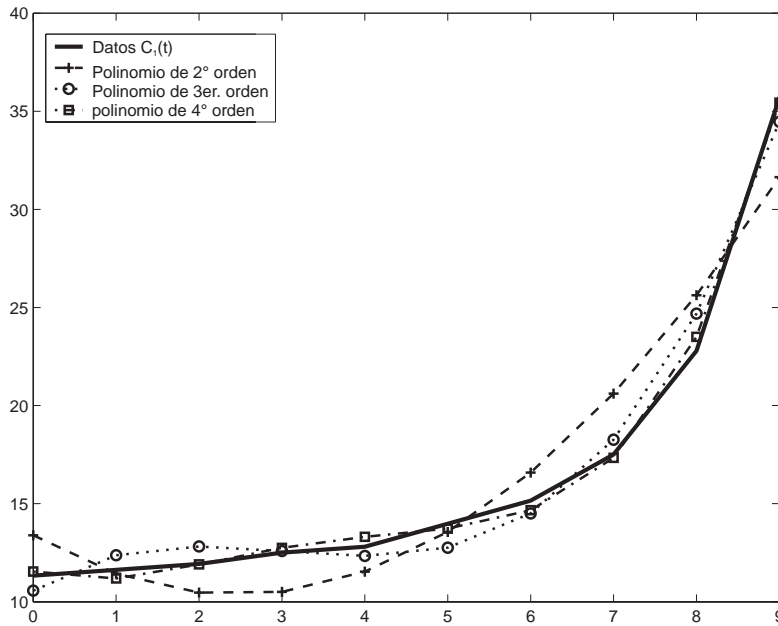


Figura 7.18: Tres ajustes diferentes para el coeficiente $C_1(t)$

$$R(t) = 0.000474t^4 - 0.006177t^3 + \dots \quad (7.35)$$

$$+ 0.026557t^2 - 0.02843t + 0.268339$$

$$L(t) = -0.0000001t^4 - 0.0000003t^3 + \dots \quad (7.36)$$

$$+ 0.0000002t^2 - 0.0000574t + 0.0232831$$

$$C(t) = -0.000015E^{-6}t^4 + 0.0004E^{-6}t^3 - \dots \quad (7.37)$$

$$- 0.003E^{-6}t^2 + 0.01E^{-6}t + 1.33E^{-6}$$

Como se puede observar en las funciones dadas por las ecuaciones 7.36 y 7.37, los coeficientes de los términos del primero al cuarto orden son pequeños comparados con el término independiente. Es decir, esos términos no contribuyen significativamente

en la evaluación de sus respectivas funciones. En forma práctica podemos suponer que esas funciones son una constante. La Figura 7.19 presenta las gráficas correspondientes a $R(t)$, $L(t)$ y $C(t)$. Las líneas continuas representan los valores estimados y las líneas discontinuas representan los valores esperados.

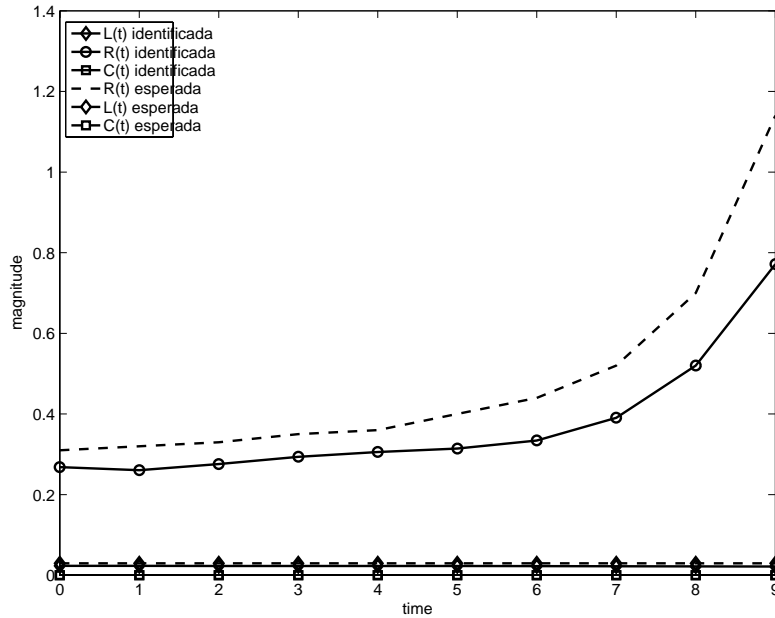


Figura 7.19: Comparación entre los valores esperados y los valores estimados para las funciones $R(t)$, $L(t)$ y $C(t)$

Se puede observar que las líneas correspondientes a $L(t)$ y $C(t)$ son prácticamente horizontales, comparadas con la variación de $R(t)$.

La Figura 7.20 muestra la señal observada en el instante $k = 7$, y su señal correspondiente una vez realizado el proceso de identificación. Los coeficientes se calcularon usando las ecuaciones 7.35, 7.36 y 7.37 evaluadas en el instante $k = 7$.

El coeficiente de correlación entre las dos señales fue: $CCN = 0.9988$ y $CCC = 1.0$. Es decir, el modelo estimado reproduce la dinámica de la planta en forma muy adecuada.

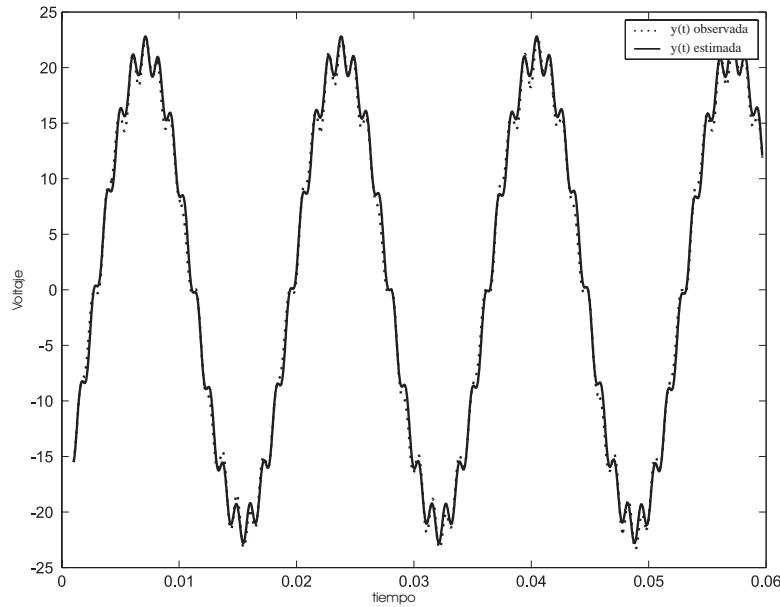


Figura 7.20: Comparación entre la respuesta observada y la respuesta del modelo estimado

7.4. Conclusiones

En este capítulo se presentan 5 casos de estudio los tres primeros se refieren a sistemas lineales e invariantes en el tiempo, mientras que los dos restantes se refieren a sistemas que varían lentamente con el tiempo. Los cuatro primeros casos son sistemas que fueron simulados en simulink, el quinto caso fue experimento realizado en el laboratorio de sistemas de potencia de la División de Postgrado de la FIE. En todos ellos, el nivel de similitud en los comportamientos exhibidos por los modelos obtenidos y las respuestas observadas, nos indican que el software diseñado tiene buen desempeño ya que las dinámicas de los modelos observados se pudieron reproducir con gran exactitud.

Capítulo 8

Conclusiones y Trabajo Futuro

Se presentó el análisis, los algoritmos y la implementación de un marco de trabajo para llevar a cabo la identificación cualitativa y cuantitativa de sistemas lineales invariantes y variantes en el tiempo.

Al mismo tiempo que se identifica la estructura del sistema se envía cada una de las componentes que forman la respuesta a una función de estimación cuantitativa que permite calcular los coeficientes de la ecuación característica de la EDO que describe la dinámica del sistema y de esta manera completar el proceso de la identificación. La parte de la estimación cuantitativa se implementó usando las propiedades fundamentales de las funciones exponenciales y de la respuesta de los sistemas lineales e invariantes en el tiempo.

Se plantean, y demuestran, tres teoremas que sirven como base para diseñar los algoritmos principales del software desarrollado. Los teoremas 2 y 3 son aportaciones originales derivadas de esta tesis. Mediante el teorema 1 se hace la descripción cualitativa de las ecuaciones diferenciales ordinarias de coeficientes constantes en base a las componentes de la respuesta transitoria de los sistemas dinámicos. El teorema 2 permite caracterizar la señal exponencial en base al número de extremos que exhibe. Y el teorema 3 Describe el comportamiento cualitativo de las componentes exponenciales de la respuesta.

El algoritmo ha tenido un buen comportamiento en la determinación de las componentes exponenciales y senoidales de diferentes sistemas probados.

El algoritmo de identificación, incluyendo el filtro de ruido se implementaron usando MATLAB. En esta implementación, la entrada es la señal de respuesta del sistema que se desea identificar, y la salida es la estructura del modelo cualitativo y cuantitativo de un sistema de caja negra. Las limitaciones de esta técnica de identificación de sistemas están relacionadas con la disposición de un conjunto de datos adecuado. Sin una buena adquisición de datos no hay mucho que se pueda hacer y, ciertamente, hay muchas razones por las que no se pueden lograr adquisición o mediciones de datos en ciertas aplicaciones.

Se obtuvo una expresión de complejidad del algoritmo, que nos permitió compararlo con una de las técnicas tradicionales más usadas, la cual resultó mejor que la técnica de los mínimos cuadrados recursivos (RLS por sus siglas en inglés) y comparable con la complejidad de la técnica “Split RLS” que es una técnica que mejora el desempeño de los mínimos cuadrados recursivos.

Se presentó un nuevo algoritmo de identificación para sistemas lentamente variantes en el tiempo. Este algoritmo está basado en QSI y la aproximación de tiempo congelado. Para eliminar el ruido implícito en las mediciones de laboratorio se diseñó un filtro digital pasa-bajas basado en la transformada rápida de Fourier. Este tipo de filtros opera en el dominio de la frecuencia y son conocidos también como selectivos en frecuencia por que permiten el paso de las señales que están dentro de la frecuencia seleccionada, las señales que no están dentro de la frecuencia seleccionada son atenuadas. Esto se lleva a cabo por medio de la transformación de la serie de tiempo al dominio de la frecuencia por medio de la transformada rápida de Fourier y se convoluciona con el kernel del filtro, la señal resultante se transforma al dominio del tiempo usando la transformada inversa de Fourier.

Se presentaron cinco casos de estudio con resultados que cubren las expectativas, ya que los modelos obtenidos pueden reproducir la dinámica de los sistemas con gran

exactitud. Las pruebas de validación de los modelos estimados son aceptables, con coeficientes de correlación muy cercanos a la unidad. El algoritmo fue probado a nivel de simulación usando MATLAB y Simulink, también se probó usando equipo de laboratorio haciendo mediciones en un sistema físico.

El algoritmo fue implementado en sus inicios en matlab 5.3 para windows y se concluyó en la versión 7.0 para linux, pasando por las versiones 6.0 y 6.5 para windows. Se probó su funcionamiento en las plataformas de linux, mac y windows todas en la versión 7.1. Los resultados obtenidos fueron satisfactorios ya que se verificó la funcionalidad del software.

8.1. QSI vs. Identificación Clásica

Como es bien sabido en el campo de la identificación de sistemas existen muchos métodos que han sido probados y validados. Estos métodos por lo general requieren además del conocimiento a priori, que el usuario especifique el orden del sistema, entre otras cosas. El algoritmo desarrollado presenta algunas propuestas diferentes para llevar a cabo la tarea de identificación. Quizá una de las contribuciones más importantes es el hecho de que basados en el comportamiento cualitativo de la respuesta del sistema el algoritmo puede proponer el orden del sistema, situación que en algunos de los métodos clásicos no se ha podido automatizar, es decir, es el usuario el que especifica el orden del modelo. Otra de las contribuciones es el bajo costo computacional que representa QSI, en donde el peor caso es de orden $O(N)$. Si bien es cierto que si consideramos el pretratamiento de los datos la complejidad aumenta a $O(N \log N)$ debido al uso de la transformada de Fourier. Pero esto se puede salvar si se opta por una mejor alternativa, y es la de utilizar un filtro analógico en el proceso de la captura de los datos. El hecho de que el filtro digital diseñado no forme parte integral del software se convierte entonces en una gran ventaja.

8.2. Trabajo Futuro

La investigación reportada en esta tesis tiene algunas sugerencias para trabajo futuro, dentro de las cuales podemos mencionar extender la metodología para el modelado de líneas de transmisión trifásicas y el modelado de la dependencia a la frecuencia de los componentes de una red eléctrica. Otra sugerencia es ampliar las capacidades del modelado de forma tal que la representación de los modelos pueda llevarse a cabo por medio de un sistema de ecuaciones diferenciales, y no solamente por una sola ecuaciones diferencial. Lo anterior con la finalidad de ampliar las capacidades hacia el modelado de máquinas eléctricas (transformador y motores).

Una característica que se puede agregar es la capacidad de combinar hipótesis basadas en leyes físicas dependiendo del dominio, por ejemplo, leyes de newton, leyes de kirchoff, ley de bernoulli, etc.

8.2.1. Identificación de Sistemas no Lineales y Modelado de Señales Periódicas

El modelado de señales periódicas juega un importante papel en el procesamiento de señales por las siguientes razones: en muchas aplicaciones de procesamiento de señales, es deseable eliminar las oscilaciones senoidales de los datos observados o estimar las frecuencias desconocidas de las oscilaciones.

La estimación de los parámetros de las senoides en el ruido agregado a la salida de los sistemas dinámicos es de gran importancia desde el punto de vista teórico y práctico. Esto se presenta en muchas aplicaciones de la ingeniería tales como los radares, las comunicaciones, el control, ingeniería biomédica y otras.

El análisis en el dominio de la frecuencia y diseño de sistemas de potencia es complicado en presencia de armónicos. La presencia de armónicos puede llevar a una interacción impredecible entre los componentes en las redes de potencia, los cuales en el peor de los casos pueden conducir a inestabilidades.

La idea será usar técnicas de modelado no lineal para modelar señales periódicas. La primera técnica no lineal que se propone estudiar y desarrollar se refiere al hecho de que una onda senoidal que se pasa a través de una función no lineal produce un espectro de armónicos. En consecuencia, la señal puede ser parametrizada como función periódica conocida (con frecuencia desconocida) en serie con una no linealidad estática desconocida. La frecuencia desconocida y los parámetros de la no linealidad serán estimados usando QSI.

El problema de las oscilaciones del ciclo límite se encuentra en muchas aplicaciones. Por lo tanto, el modelado matemático de los ciclos límites se convierte en un tópico esencial que ayuda para entender mejor las oscilaciones del ciclo límite en varios campos del conocimiento. Otra técnica no lineal es usar una EDO no lineal de segundo orden para modelar la señal periódica como una oscilación del ciclo límite. Los coeficientes de la EDO sugerida se modelaran como funciones polinomiales mediante QSI.

8.2.2. Modelado y Control de Estructuras Mecánicas

Las vibraciones son un problema común en estructuras mecánicas, particularmente en las partes flexibles. Esto puede evitarse si se reduce la flexibilidad en tales partes haciéndolas más fuertes y pesadas. Para muchas aplicaciones, es deseable tener estructuras ligeras, lo cual hace la solución anterior poco factible. En cambio, sería mejor tener un dispositivo que pudiera realizar una absorción activa de las vibraciones sin un aumento sustancial en la masa. En el caso en el que tales dispositivos son incluidos en el material de la estructura, las estructuras son conocidas como estructuras inteligentes.

Una forma para diseñar este tipo de estructuras es usando elementos piezoeléctricos que se anexan a la estructura. Los elementos piezoeléctricos exhiben una deformación significativa cuando se les aplica un voltaje, y por el contrario, produce un voltaje cuando son deformados. Esta característica permite usar estos elementos

como actuadores o como sensores en una estructura de este tipo. La idea es modelar el comportamiento de una estructura, sujeta a vibraciones y oscilaciones, que permita diseñar las estrategias de control para generar las señales de entrada para los actuadores. No solo para que amortiguen las vibraciones sino también para que prevengan la propagación de oscilaciones a lo largo de la estructura.

Referencias

- [Alvarez83] Alvarez Jaime, Bonilla Moisés, Introducción a la identificación de procesos, CINVESTAV-IPN 1983.
- [An-Yeu94] An-Yeu Wu & K.J. Ray Liu, Split recursive Least Squares: Algorithms, Architecture and Applications. url = [http:// citeseer.ist.psu.edu/226011.html](http://citeseer.ist.psu.edu/226011.html), 1994.
- [Benhamou94] Benhamou F., Mc Allester D. and Van Hentenryck P. CLP Intervals Revisited. Logic Programming Proceedings of the 1994 International Symposium, 124-138 Ithaca New York 1994.
- [Bellazzi99] Bellazzi, R.; Guglielmann, R.; and Ironi, L. 1999. A qualitative-fuzzy framework for nonlinear black-box system identification. In IJCAI99, 1041–1046.
- [Bellazzi00] R. Bellazzi, R. Guglielmann, L. Ironi: How to improve fuzzy-neural system modeling by means of qualitative simulation, IEEE Trans. on Neural Networks, 11, 1, (2000), 249-253.
- [BoxJenkins94] Box G. E. P. and Jenkins G. M. Time Series Analysis, Forecasting and Control. Holden Day, 1994.
- [Boyce69] Boyce, W.E., and DiPrima, R.C. 1969. Elementary Differential Equations. New York: John Wiley, second edition.

- [Bradley96] Bradley, E., and Stolle, R. 1996. Automatic construction of accurate models of physical systems. *Annals of Mathematics of Artificial Intelligence* 17:1–28.
- [Bradley01] Bradley, E.; Easley, M.; and Stolle, R. 2001. Reasoning about nonlinear system identification. *Artificial Intelligence* 133:139–188.
- [Bronshtein77] Bronshtein I. Semendiaev K. *Manual de Matemáticas para Ingenieros y Estudiantes*. 3a edición editorial MIR, Moscu 1977.
- [Capelo98] A.C. Capelo, L. Ironi, S. Tentoni: Automated mathematical modeling from experimental data: an application to material science, *IEEE Transactions on Systems, Man and Cybernetics - Part C.*, 28, 3 (1998), 356-370.
- [CemSay92] Cem Say A. C. *Qualitative System Identification*. Doctoral dissertation. Bogazici University 1992, Istambul, Turkey.
- [CemSay96] Cem Say A. C. and Selahattin Kuru. *Qualitative System Identification: Deriving Structure From Behavior*. *Artificial Intelligence*, Vol 83 pp. 75-141 1996.
- [Chen00] Chen J. and Gu G. *Control-Oriented System Identification: An H_∞ Approach*. John Wiley and sons, New York, New York, 2000.
- [ChingTeng96] Ching-Teng Lin and C. S. George Lee. *Neural Fuzzy Systems A Neuro-Fuzzy Synergism to Intelligent Systems*, 1996 Prentice Hall P T R, Upper Saddle River, New Jersey.
- [Cormen01 et al] Cormen T. H. Leiserson C. E. Rivest R. L. Stein C. *Introduction to Algorithms*, Second Edition, 2001 Mc Graw Hill Book Company Cambridge, Massachusetts.
- [Cuberos02 et al] Cuberos F.J.; Ortega, J.A.; Gasca, R.M.; and Toro, M. 2002. Qsi - qualitative similarity index. *Proc. 16th Int. Workshop on Qualitative Reasoning*.

- [Davis85] Davis M. H. A. and R. Vinter. Stochastic Modelling and Control. Chapman and Hall, London, 1985.
- [DeKleer77] De Kleer J. 1977. Multiple Representations of Knowledge in a Mechanics Problem Solvers, Poceedings of IJCAI-77 pag. 299-304.
- [DeKleer84] De Kleer J. and Brown J, 1984 “A Qualitative Physics based on confluences”, Artificial Intelligence 24 7-83.
- [Desoer70] C. A. Desoer. Slowly varying discrete systems $x_{i+1} = A_i x_i$, Electronics Letters 6, pp. 339-340, 1970.
- [Dennis00] J.E. Dennis, Jr. Schnabel Robert B. ”Numerical Methods for Unconstrained Optimization and Nonlinear Equations”. Classics in Applied Mathematics. SIAM. 2000.
- [Downing96] Downing, C.J.; Byrne, B.; Coveney, K.; and Marnane, W. 1996. Controller optimisation and system identification using genetic algorithms. Technical report, Dept. of Electrical Engineering University College Cork.
- [Eykhoff74] System Identification. Wiley, New York, New York, 1974.
- [Flores95 et al] Flores, J. and Farley, A. M. 1995. Reasoning about constant coefficient dynamic systems. In Proc. 9th Int. Workshop on Qualitative Reasoning About Physical Systems.
- [Flores02 et al] Flores Juan and Pastor Nelio. Qualitative and Quantitative Systems Identification for Linear Time Invariant Dynamic Systems, Workshop on Qualitative Reasoning 2002 Sitges Barcelona España. Junio 2002.
- [Flores05] Juan J. Flores, Nelio Pastor. “Time-Invariant Dynamic Systems Identification Based on the Qualitative Features of the Response”. publicado en : Journal of Engineering Applications of Artificial Intelligence, volume 18 issue 6 pp. 719-729.

- [Forbus96] Forbus Kenneth D. Qualitative Reasoning Draft of chapter for the CRC Handbook of Computer Science. The Institute for Learning Sciences Northwestern University, Evanston Illinois USA 1996.
- [Forbus87 et al] K. D. Forbus, P. Nielsen and B. Faltings. Qualitative Kinematics: a Frame, Proceedings of IJCAI-87, pp 430-436, 1987.
- [Freedman68 et al] Freedman M. and G. Zames. Logarithmic variation criteria for stability of systems with time-varying gains, SIAM J. Control vol. 6,N° 3. pp. 487- 507. 1968.
- [Goodwin77] Goodwin G. C. and Payne R. L. Dynamic System Identification: Experiment Design and Data Analysis. Academic Press, New York, New York, 1977.
- [Goldberg98] Goldberg, D.E. 1998. Genetic Algorithms in Search, Optimization, and Machine Learning. Boston: Addison Wesley.
- [GeATbx01] Pohlheim Hartmut. “Evolutionary Algorithms: Principles, Methods and Algorithms”, document part of the Genetic and Evolutionary Algorithm Toolbox for use with MatLab. <http://www.geatbx.com>. November 2001.
- [Gotz83] Gotz Alefeld and Jurgen Herzberger, Introduction to Interval Computation. Academic Press, New York 1983.
- [GuglielmannIroni04] Guglielmann Raffaella and Ironi Liliana. The need for qualitative reasoning in fuzzy modeling: robustness and interpretability issues. 18 th. International Qualitative Reasoning Workshop, North Western University, Urbana Evanston Illinois, August 2-4 2004.
- [Haupt98] Haupt, R.L., and Haupt, S.E. 1998. Practical Genetic Algorithms. New York: Wiley Interscience.
- [Hunt93] Hunt, K.J. 1993. Black-box and partially known system identification with genetic algorithms. In European Control Conference.

- [Juang96] Juang J-N Applied System Identification. Prentice Hall, Englewood Cliffs, New York, 1996.
- [Junheng92] Junheng Luo and Dominique Thiébaud. 1992. Smoothing Strange Attractors with Splines. Department of Computer Science Smith College Northampton, MA 01063
- [Kailath80] Kailath T. Linear Systems. Prentice Hall, Englewood Cliffs, New Jersey, 1980.
- [Kamen89] Kamen E. W., Khargonekar P. P. and Tannenbaum A. Control of Slowly-Varying Linear Systems. IEEE Transactions on Automatic Control, vol 34 No. 12 December 1989. Cambridge, Massachusetts.
- [Kashyap76] R. L. and Rao A. R. Dynamic Stochastic Models From Empirical Data. Academic Press, New York, New York, 1976.
- [Kay00] Kay, H.; Rinner, B.; and Kuipers, B. 2000. Semi-quantitative system identification. Artificial Intelligence 119:103–140.
- [Kollar] Kollar István. 2004. Frequency Domain System Identification Toolbox for Use with Matlab, User's Guide. Version 3.3d May 11,2005. Gamax Ltd. URL: <http://elecwww.vub.ac.be/fdident/>
- [Kristinsson92] Kristinsson, K., and Dumont, G.~A. 1992. System identification and control using genetic algorithms. IEEE Trans. Syst. Man Cybern. 22(5):1033–1046.
- [Kuipers94] Kuipers, Benjamin. 1994. Qualitative reasoning: modeling and simulation with incomplete knowledge. The MIT Press. Cambridge, Massachusetts.
- [Kurosch77] Kurosch, A.~G. 1977. Curso de Algebra Superior. México: Editorial MIR. Translated from Russian to Spanish by Emiliano Aparicio Bernardo.

- [LeYi95] Le Yi Wang. Identification and Control of Slowly Varying Systems: Recent Advances; Proceedings of the 34th Conference on decision & control, New Orleans, LA December 1995.
- [Liliana98] L. Ironi, S. Tentoni: An integrated quantitative-qualitative approach to automated modeling of visco-elastic materials from experimental data, in: R. Teti (ed.), Proc. ICME 98 - CIRP International Seminar on Intelligent Computation in Manufacturing Engineering, Capri, 1-3 July 1998, CUES-Salerno & RES Communication-Naples, 381-388.
- [LiuCoghill04] Liu Honghai and Coghill George M. Qualitative Modelling of Kinematic Robots. 18th. International Qualitative Reasoning Workshop, North Western University, Evanston Urbana Illinois August 2-4 2004.
- [Ljung99] Ljung, L. 1999. System Identification: Theory for the user 2nd Edition. Upper Saddle River, NJ: Prentice-Hall.
- [Ljung05] Ljung, L. System Identification Toolbox for Use with Matlab, User Guide. Revised for version 6.1.1 March 2005. The MathWorks Inc, Natick MA: Prentice-Hall.
- [LjungGlad94] Ljung L. and Glad T. Modeling of Dynamic Systems. Prentice Hall, Upper Saddle River, New Jersey, 1994.
- [Longman99] Longman Richard, Jer-Nan Juang. Optimized Systems Identification, technical report, Columbia University, Structural Dynamics Branch at NASA Langley Virginia, Marzo 1999.
- [Lou94] Luo Junheng and Thiébaud Dominique. Smoothing Strange Attractors with Splines. Department of Computer Science. Smith College. Northampton, Massachusetts. 1994.

- [Mahmood95] Mahmood Samavat and Ali Jabar Rashidie. A new Algorithm for Analysis and Identification of Time Varying Systems, Proceedings of the American Control Conference, Seattle Washington, June 1995
- [Mathews00] Mathews John, Fink Kurtis 2000. Métodos Numéricos con MATLAB 3a Ed. Prentice Hall, Madrid España.
- [Matlab87] MatLab, Tutorial de, 1987. Series de trabajo de matlab.
- [Moore66] Ramon E. Moore, Interval Analysis. Prentice Hall Inc. Englewood Cliffs, NJ, 1966.
- [MuhDirk93] Mühlenbein H. Schlierkamp-Voosen D., “Predictive Models for the Breeder Genetic Algorithm: I. Continuous Parameter Optimization”, *Evolutionary Computation* 1(1):25-49, 1993.
- [MuhDirk93b] Mühlenbein H. Schlierkamp-Voosen, “Optimal Interaction Of Mutation And Crossover in the Breeder Genetic Algorithm”.
- [MuhDirk95] Muhlenbein H. Schlierkamp-Voosen D., “Analysis of Selection, Mutation and Recombination in Genetic Algorithms”. In: Wolfgang & Eeckman, Eds., *Evolution as a Computational Process*, Berlin 1995.
- [Nelles01] Nelles Oliver. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Springer-Verlag Berlin Heidelberg, 2001.
- [Nyquist46] H. Nyquist, “Certain Topics in Telegraph Transmission Theory”, *AIEE Transactions*, 1946, p.617
- [Oliver95 et al] P. Oliver, A. Orrnsby and K. Nakata. Occupancy array based kinematic reasoning, *Proceedings of 9th International Workshop on Qualitative Reasoning*, University of Amsterdam, The Netherlands. pp 148-155, 1995

- [Oppenheim94] Oppenheim Alan V., Señales y Sistemas (traducción de la primera ed. en ingles), Prentice Hall 1994.
- [Overschee96] P. van Overschee and B. de Moor. Subspace Identification for Linear Systems: Theory, Implementation, Applications. Kluwer Academic Publishers, Boston, Massachussetts, 1996.
- [Pastor00] Pastor, N. 2000. Identificación de Sistemas Dinámicos Utilizando Algoritmos Genéticos. Tesis de Maestría, Facultad de Ingeniería Eléctrica, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, México.
- [Pastor05 et al] Nelio Pastor, Juan J. Flores, Felix Calderón, Claudio Fuerte. “Identification of Slowly Time Varying Systems Based on the Qualitative Features of Transient Response a Frozen Time Approach”. Enviado para revisión a IEEE Transactions on Systems, Man, and Cybernetics–Part C.
- [Price03 et al] Price C. J., Snooke N. A., Lewis S. D. Adaptable Modeling of Electrical Systems. 17th International Qualitative Reasoning Workshop. University of Brasília, Brasil 20-22 August 2003.
- [Rodriguez94] Rodríguez Ramírez, Francisco J. Dinámica de Sistemas. México: Trillas: UNAM Facultad de Ingeniería, 1989 (reimpresión 1994).
- [Rugh96] Rugh Wilson J. Linear System Theory 2nd Edition, Prentice Hall Inc. Upper Saddle River NJ 1996.
- [Rees&Sparks91] Rees, P.K., Sparks F. and Reeds, C.S. 1991. Algebra. USA: McGraw-Hill.
- [Smith88] Smith W. Allen, Análisis Numérico. Prentice Hall Hispanoamericana, Mexico 1988.
- [Smith99] Smith, S.W. 1999. The Scientists and Engineers Guide to Digital Signal Processing. California Technical Publishing.

-
- [Stoica89] Söderström T. and Stoica P. 1989. System Identification. Prentice Hall International, London.
- [Trave-Massuyes91 et al] Trave-Massuyes L., Borseau P., Dague P., Dubois D., Missier A., Piera N., Prade H. Mathematical Aspects, LAAS Report 91050 Centre National de la Recherche Scientifique, Toulouse Cedex France 1991.
- [Walpole89] Walpole R.E. Myers R.H. Probabilidad y Estadística para Ingenieros. 3a Edición, Ed. McGraw Hill, 1989 México D.F.
- [Wylie95] Wylie C. Ray; Barret, Louis C. 1995. Advanced Engineering Mathematics, McGraw-Hill. Sixth Edition.
- [ZhangZibo87] Zhang and Zibo, F. N. Application of Genetic Algorithms to System Identification. Australia: Department of Electrical and Computer Engineering University of Wollongon 1987.

Glosario

<i>EDO</i>	Ecuación Diferencial Ordinaria.
<i>QDE</i>	Ecuación Diferencial Cualitativa.
<i>QSI</i>	Algoritmo de Identificación Cualitativa de Sistemas.
<i>QSIM</i>	Qualitative Simulation.
<i>LIT</i>	Lineal e Invariante en el Tiempo.
<i>RLC</i>	Resistivo Inductivo Capacitivo.
<i>TPAFilter</i>	Filtro de Componentes Senoidales.
<i>EXPFilter</i>	Filtro de Componentes Exponenciales.
<i>FFT</i>	Transformada Rapida de Fourier.

Apéndice A

Desarrollo del Software de Identificación

A.1. Introducción

Para llevar a cabo la identificación se ha desarrollado un software que, entre otras características, incluye una interfaz gráfica y la interacción por medio de comandos en modo de texto. Estas características permiten que el usuario pueda elegir la forma de ejecución del software. Este capítulo presenta el desarrollo del software para identificación de sistemas dinámicos aplicando los principios presentados en el capítulo 4.

Para la programación de las diferentes rutinas y funciones se utilizó como herramienta de desarrollo el programa MATLAB.

A.1.1. Características

- Proporciona una herramienta para parametrizar y estimar la EDO por medio de algoritmos sencillos. Este software permite simular la respuesta de sistemas con comportamientos diseñados por el usuario.

- Diseñado para trabajar en dos modos, de esta manera se puede usar en modo de texto o por medio de una interfaz gráfica, con un desempeño similar.
- Se requiere unicamente una matriz de datos, que contenga el vector de tiempo t y la respuesta del sistema $y(t)$, para llevar a cabo la identificación.
- Puede ejecutarse apartir de la versión 5.3 de MATLAB para Windows. Aunque también se probó en la versión 7.0 para Unix/Linux y Mac (se recomienda la versión 6.5 para Windows).

A.2. Estructura General del Software de Identificación de Sistemas Dinámicos

El proceso de identificación de sistemas dinámicos consiste en una búsqueda en el espacio de modelos, la cual requiere realizar cálculos para obtener un modelo óptimo. Para simplificar y acelerar los cálculos se propone desarrollar un software que incluya, entre otras características, una interfaz gráfica de usuario mediante la cual se pueda interactuar con el proceso de identificación. El software considera las posibilidades de su ejecución mediante instrucciones directas en la ventana de comandos de matlab ó mediante la interfaz gráfica con el usuario. La ejecución por medio de comandos permite al usuario aplicar QSI en sus desarrollos, sin tener que hacer uso de la interfaz gráfica.

La estructura general del software se presenta en la Figura A.1.

Las funciones *qsi* y *gen_exp* se pueden ejecutar desde la ventana de comandos de matlab. La función *qsi* hace la llamada al algoritmo de identificación, mientras que *gen_exp* llama al algoritmo para diseñar respuestas transitorias.

A continuación se presenta el modo de uso por medio de instrucciones en la ventana de comandos.

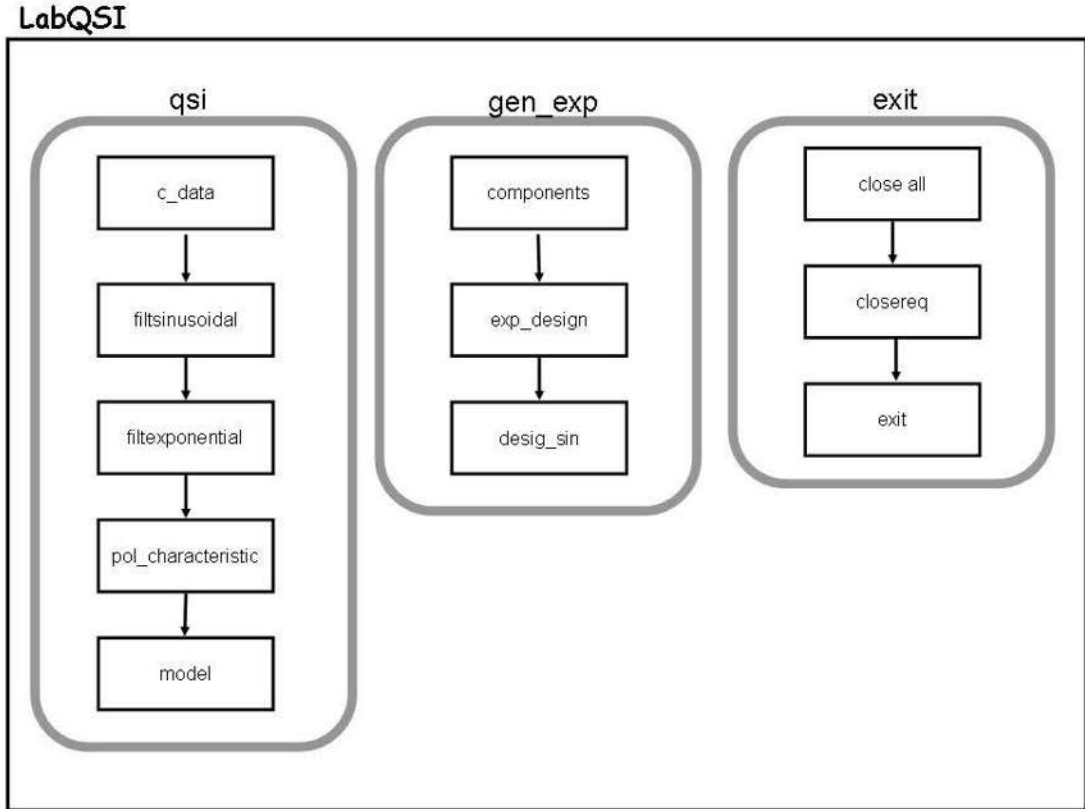


Figura A.1: Diagrama de flujo del software QSI

A.3. Modo de texto

QSI trabaja con dos variables, una de entrada y otra de salida. La variable de entrada es la respuesta transitoria del sistema que se desea identificar y la variable de salida es el modelo identificado. La variable de entrada se forma con las series de tiempo correspondientes al intervalo de tiempo t y la respuesta $y(t)$ del sistema, dando como resultado una variable de tipo matriz de $2 \times N$. Donde N es el número de puntos, de las series de tiempo capturadas. El primer renglón corresponde al intervalo de tiempo, t , dentro del cual se observó al sistema y el segundo renglón a la respuesta, $y(t)$, del sistema.

El comando en Matlab que permite agrupar las dos variables (t y $y(t)$) se escribe

como sigue:

```
>>response=[t;y(t)];
```

Dentro del ejemplo presentado la variable descrita anteriormente la llamaremos *response* ó *respuesta* indistintamente.

Generalmente los datos que forman la variable de entrada son datos que han sido almacenados en disco o que están disponibles en la memoria de la computadora por que han sido adquiridos recientemente o han sido generados mediante alguna simulación.

Para llevar a cabo la identificación podemos teclear en la ventana de comandos de matlab cualquiera de las siguientes instrucciones:

```
>> modelo=qsi(respuesta)
```

La forma anterior se usa si la variable *respuesta* ha sido formada previamente o si estaba almacenada en algún archivo. Si no se dispone de la variable *respuesta*, pero se tiene en la memoria del espacio de trabajo de Matlab las variables *t* y *y(t)*, se puede ejecutar la identificación de la manera siguiente:

```
>>modelo=qsi([t;y(t)]);
```

donde *respuesta* es la matriz de datos de entrada (de tamaño $2 \times N$) y *modelo* es la variable de salida donde se almacenará la estructura que contiene la descripción del modelo. La Figura A.2 presenta la respuesta del sistema que se usará como ejemplo en esta sección.

A continuación se muestra un ejemplo usando los datos contenidos en la variable *response* (Figura A.2).

```
>> model=qsi(response)
model =
  components: [1x1 struct]
  label: 'E1,ES2'
  initial_cond: [2x1 double]
  data: [1x1 struct]
```

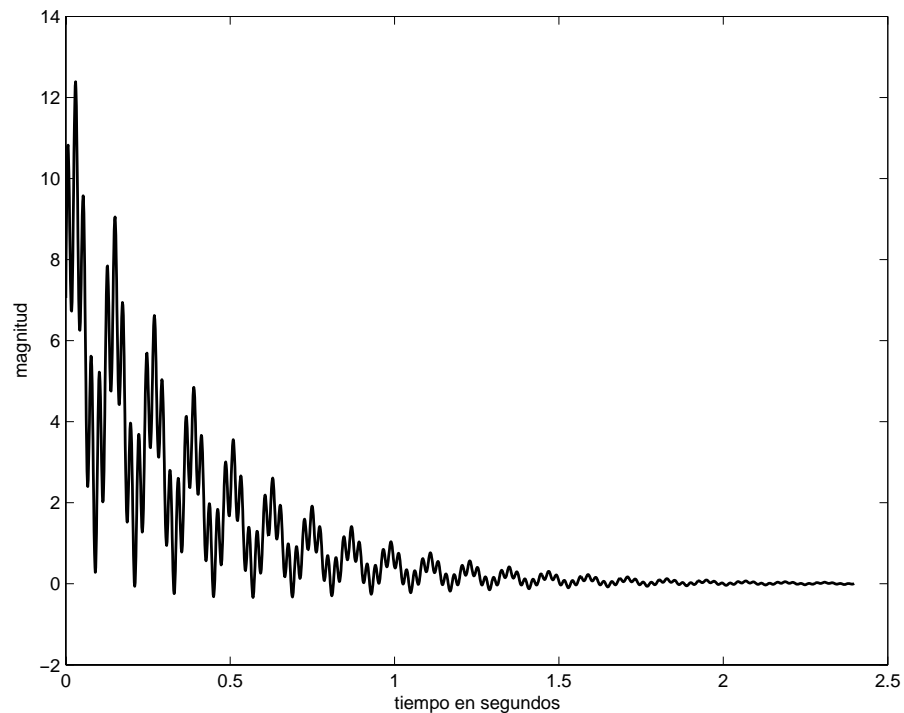


Figura A.2: Señal usada como parámetro de entrada para la función *qsi*

```

parameters: [3x4 double]
source: 0
coefficients: [1 12.2297 7.1463e+004 5.4220e+005 ... \
               ... 1.9068e+008 5.7966e+008]
order: 5
correlation: 0.9909
estimated_y: [1x2001 double]
roots: [5x1 double]

```

La salida de la función *qsi* es la descripción del modelo y se almacena en una estructura que contiene los siguientes elementos:

- *components* .- Contiene el número de componentes de cada tipo que integran la señal.
- *label* .- Contiene la forma cualitativa del modelo indentificado.

- *initial_cond* .- Son las condiciones iniciales estimadas de la respuesta del sistema donde el primer valor corresponde a $y(0)$ y el segundo valor a $y'(0)$.
- *data* .- Aquí se almacenan los datos de entrada.
- *parameters* .- Corresponden a los parámetros que definen a cada una de las componentes que conforman la respuesta del sistema, de aquí se calculan las raíces de la ecuación característica y consecuentemente los coeficientes de la ecuación diferencial que describe al modelo.
- *source* .- Es la señal correspondiente a la fuente de alimentación del sistema; en caso de ser un impulso su valor será 0 y en el caso de un escalón se especifica el valor del escalón. En otro caso se especifica la forma de onda. Por ejemplo, $A \sin(n\omega t + \varphi)$ donde A, n son valores positivos, ω es la frecuencia angular, t es la variable de tiempo y φ es el ángulo de defasamiento.
- *coefficients* .- Es un vector donde se almacenan los coeficientes estimados de la ecuación diferencial ordenados desde el coeficiente de mayor orden al de menor orden.
- *order* .- Es el orden del modelo estimado.
- *correlation* .- Es el coeficiente de correlación entre la salida del modelo y la señal observada.
- *estimated_y* .- Es la salida del modelo estimado, usado para el cálculo del coeficiente de correlación.
- *roots* .- Son las raíces de la ecuación característica de la ecuación diferencial.

Para recuperar estos valores es necesario teclear en la ventana de comandos el nombre de la variable de salida (model, en este ejemplo) seguido de un punto y del valor que se desea recuperar, a continuación se presenta el ejemplo correspondiente a la señal de la Figura A.2.

```
>> model.components
```

```
ans =
```

```
ES: 2
E: 1
```

Para este ejemplo los componentes son: dos senoidales y una exponencial.

La sentencia “model.parameters” proporciona la solución para la ecuación (3.5) en la forma de las ecuaciones (3.25 y 3.26)

```
>> model.parameters
```

```
ans =
```

```
2.8280   -2.3213  262.0023  -0.0125
3.2563   -2.2657  52.5172   0.0443
10.0379  -3.0558    0         0
```

donde los parámetros de cada una de los componentes son ordenados de la siguiente forma: la primera columna corresponde a los coeficientes a , la segunda al factor de decaimiento r , la tercera a la frecuencia angular ω y la cuarta al ángulo de defasamiento ϕ . Para el ejemplo que se ilustra los dos primeros renglones definen una componente senoidal y el tercer renglón define una componente exponencial, como se ilustra en las ecuaciones A.1, A.2 y A.3.

$$\text{Senoidal}_1 = 2.828 e^{-2.3213t} \text{sen}(262t - 0.0125) \quad (\text{A.1})$$

$$\text{Senoidal}_2 = 3.2563 e^{-2.2657t} \text{sen}(52.51t + 0.0443) \quad (\text{A.2})$$

$$\text{Exponencial}_1 = 10.0379 e^{-3.0558t} \quad (\text{A.3})$$

“model.coefficients”, presenta los coeficientes c_i de la EDO que describe al sistema, ecuación (3.5), ordenados de c_{n-1} a c_0 .

```
>> model.coefficients
```

```
ans =
```

```
1.0e+008 *
0.0000 0.0000 0.0007 0.0054 1.9068 5.7966
```

Con la instrucción “model.roots” se obtienen las raíces de la ecuación característica de la EDO.

```
>> model.roots
```

```
ans =
```

```
1.0e+002 *
-0.0232 + 2.6200i
-0.0232 - 2.6200i
-0.0227 + 0.5252i
-0.0227 - 0.5252i
-0.0306
```

Para recuperar el valor de las condiciones iniciales estimadas se teclea

```
>> model.initial_cond
```

```
ans =
```

```
0.1443
0.2043
```

A.3.1. Identificación de Sistemas Variantes en el Tiempo

Una de las características que amplía el funcionamiento del algoritmo QSI es la identificación de sistemas lentamente variantes en el tiempo. Para esta característica no se dispone interfaz gráfica, solo se puede ejecutar desde el modo de texto.

Una vez que se tienen capturada la respuesta del sistema en cada instante congelado esta se procesa con QSI y se obtiene un conjunto de modelos, cada uno correspondiendo a un instante congelado. Este conjunto de modelos se organiza en una matriz

de tal forma que cada renglón corresponde a un instante congelado, por ejemplo: si $m_1, m_2 \dots m_k$ son los modelos identificados, entonces la matriz de coeficientes de los modelos se conformará de la siguiente forma:

```
>>M=[m1;m2;...;mk];
```

y la identificación del modelo variante en el tiempo se lleva a cabo mediante los comandos siguientes:

```
>>modelovt=qsitimevarying(M);
```

El modelo resultante se almacena en la variable de salida *modelovt*

A.3.2. Generador de Experimentos

Para ejecutar el generador de experimentos a través de la ventana de comandos (modo de texto) la sintaxis es:

```
>> response=gen_exp(E,ES,archivo);
```

donde *response* es la variable de salida, *E* y *ES* son el número de componentes exponenciales y senoidales, respectivamente, requeridos en la respuesta, *archivo* es el nombre para el archivo con extensión *.mat* donde será almacenada la variable *response*, por ejemplo

```
>> response=gen_exp(2,2,'example');
```

Este comando genera una respuesta con dos exponencial y dos senoidales, “example” es el nombre del archivo donde se guardar la respuesta generada. La Figura A.3 presenta la señal generada con el ejemplo anterior.

A.4. Interfaz Gráfica con el Usuario

La interfaz gráfica con el usuario (GUI por sus siglas en inglés de “Graphic User Interface”) se ejecuta cuando se teclea la siguiente instrucción desde la venta de comandos de matlab:

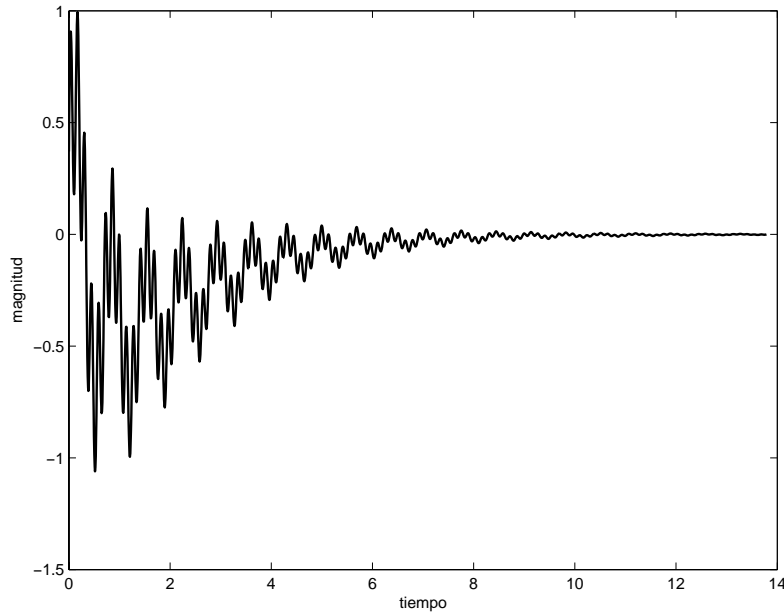


Figura A.3: Respuesta transitoria generada con el algoritmo de generación de respuestas

```
>>labqsi
```

La Figura A.4 muestra la pantalla desplegada por la GUI. En ella se pueden observar los elementos que la componen, se destacan en la parte inferior los tres botones que controlan la operación del software.

Como se puede ver en la Figura A.1 hay tres botones que permiten el uso del software, el primer botón (de izquierda a derecha) *Start QSI*, llama a la función *qsi()*; el segundo botón *Experiment Design*, llama a la función *gen_exp()*; y el tercer botón *Exit*, llama a la función *closereq*.

A continuación se hará una descripción del funcionamiento de cada uno de los botones de la interfaz gráfica.

A.4.1. El botón Start QSI

Cuando se da un click sobre el botón *Start QSI* se cargan los datos correspondientes a la variable *response* por medio del cuadro de diálogo presentado en la Figura

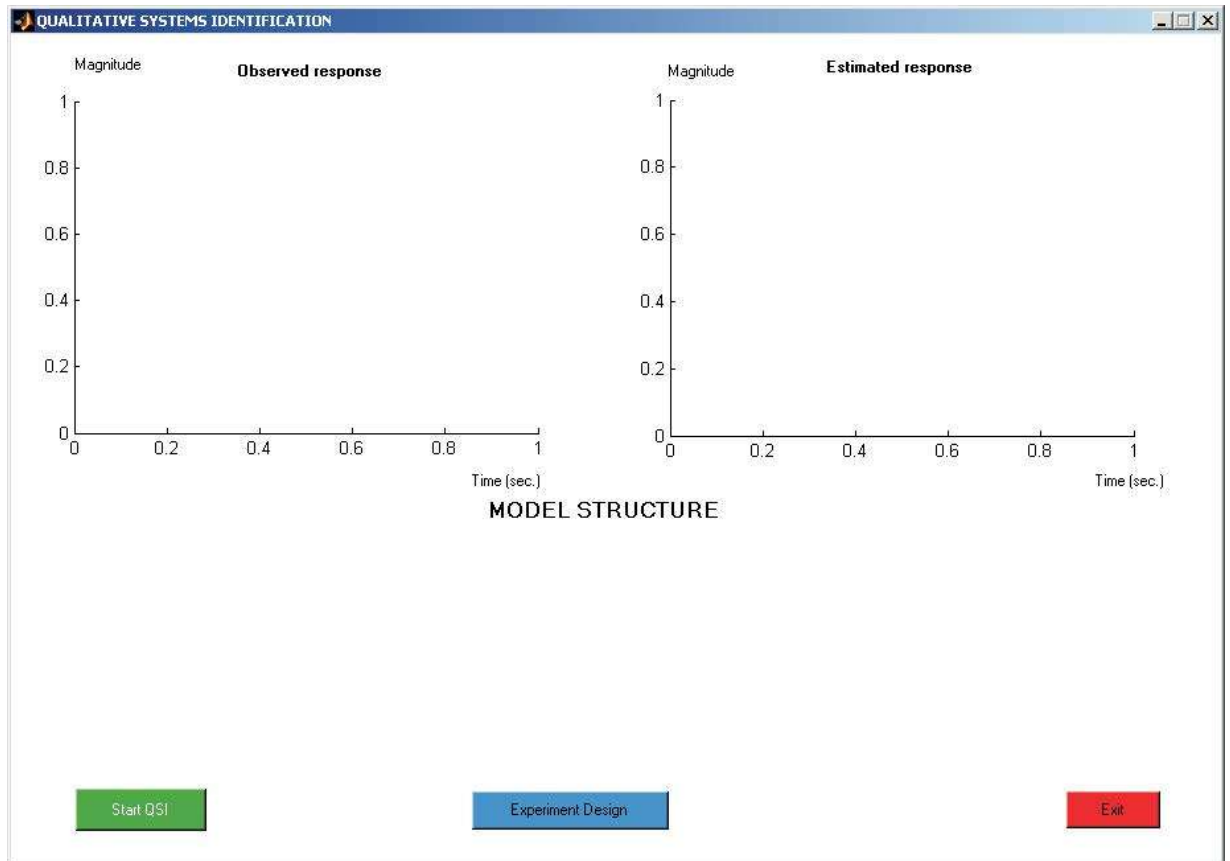


Figura A.4: Interfaz gráfica del software QSI

A.5.

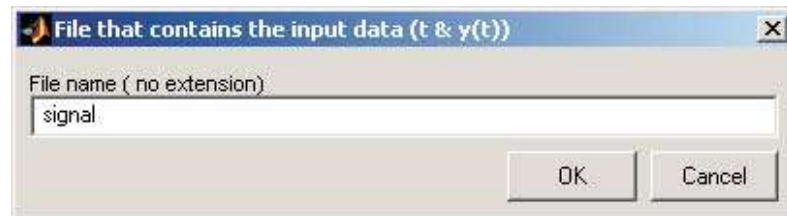


Figura A.5: Cuadro de diálogo para la entrada de datos

El cuadro de diálogo de la Figura A.5 solicita el nombre de un archivo tipo *mat*. Los archivos *mat* son archivos usados por matlab para almacenar variables del espacio de trabajo en disco.

Una vez que se ha pasado el nombre del archivo donde se almacena la variable *response* la interfaz hace la llamada a la función *qsi(response)*. y le pasa el control del programa. Una vez que *qsi()* lleva a cabo el proceso de la identificación regresa el control del programa a la interfaz gráfica la cual despliega los resultados de la identificación como se ilustra en la Figura A.6.

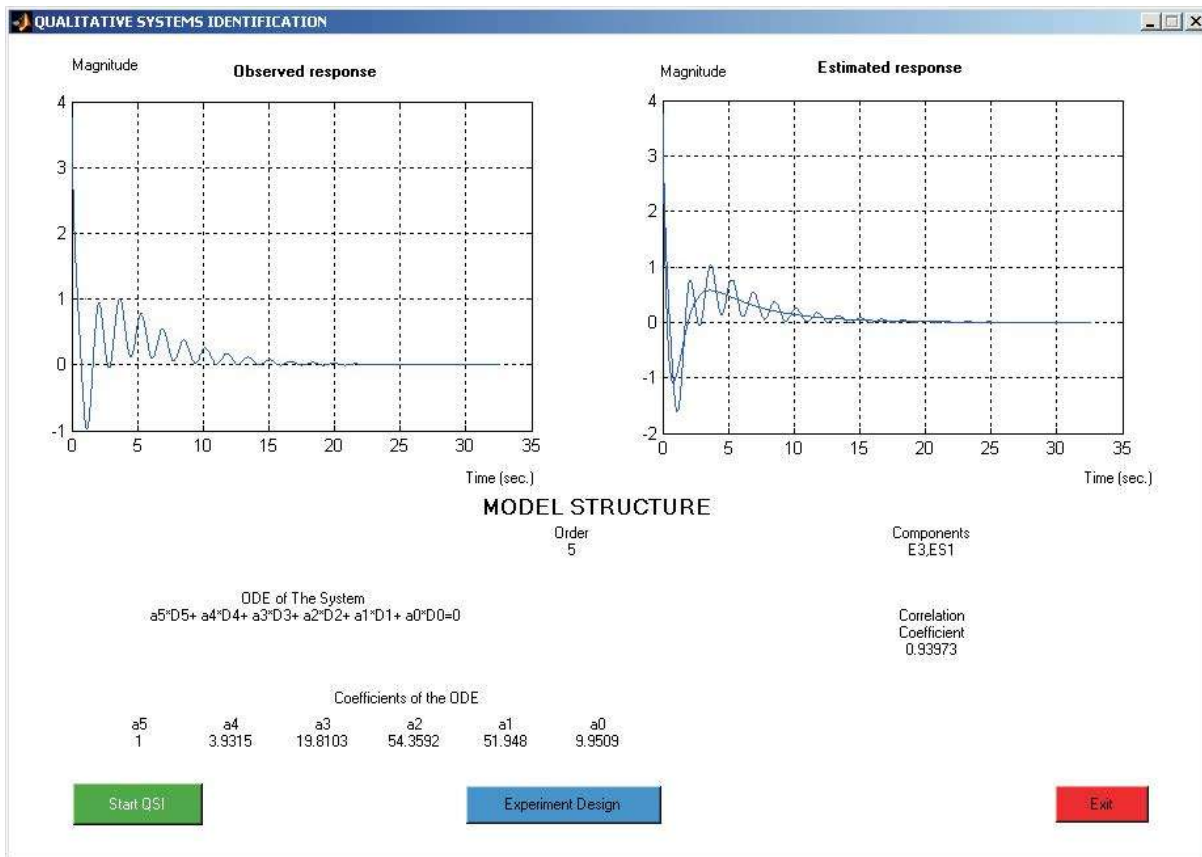


Figura A.6: Interfaz gráfica para desplegar el resultado de la identificación

La Figura A.6 muestra la señal de entrada, el orden del sistema, la Ecuación Diferencial Ordinaria que describe al sistema, el coeficiente de correlación entre el comportamiento observado y el comportamiento del modelo obtenido y la forma cualitativa del modelo. Para representar las componentes de la respuesta se usa la nomenclatura definida en la sección ??; las exponenciales se representan usando la letra *E* y para

las senoidales se usan las letras ES . La notación será de la forma; En_1ESn_2 , donde n_1 y n_2 son números enteros positivos que indican la cantidad de componentes exponenciales y senoidales respectivamente.

A.4.2. El botón Experiment Design

Al elegir este botón se ejecuta la función `gen_exp` con la cual se generan respuestas de sistemas con comportamientos interesantes. La primera acción que realiza esta función es solicitar los datos para generar la respuesta, mediante el cuadro de diálogo de la Figura A.7



Figura A.7: Cuadro de diálogo para configurar la respuesta simulada

Mediante este cuadro de diálogo se indica el número de componentes deseados en la respuesta que se va a generar. Con esta información el algoritmo genera en forma aleatoria los parámetros para la primera exponencial y las componentes sucesivas se calculan de acuerdo a lo planteado en la sección 4.6. A continuación ilustraremos como generar una señal transitoria con dos componentes exponenciales y dos senoidales. La Figura A.7 presenta el cuadro de diálogo donde se solicita la cantidad de cada componente deseados en la respuesta y la Figura A.8 presenta la señal generada en la gráfica correspondiente a la señal estimada.

Una vez que se ha generado la respuesta, el software solicita un nombre de archivo en el cual se guarda la variable que contiene el tiempo t y la respuesta $y(t)$, la Figura A.8 presenta el resultado de activar el botón *Experiment Design*. La respuesta

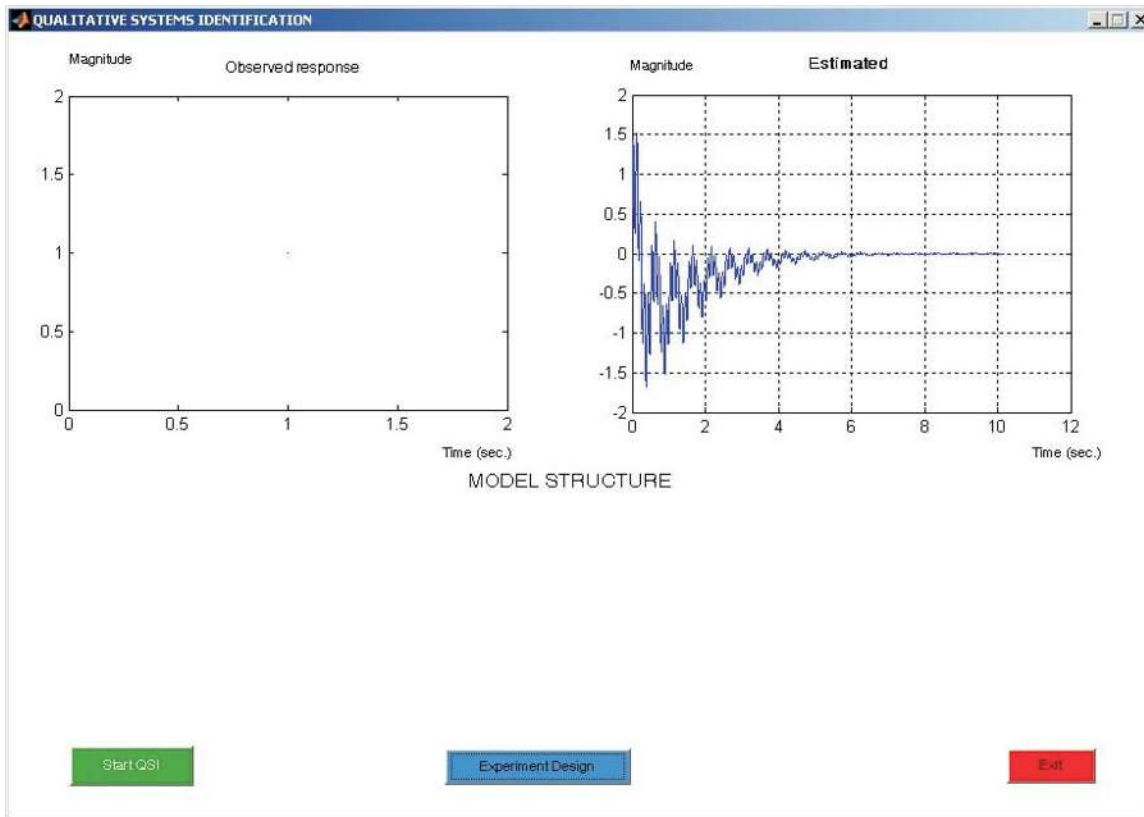


Figura A.8: Interfaz gráfica de la generación de respuestas transitorias

generada se presenta en el recuadro derecho de la interfaz gráfica, el resto de la interfaz no presenta ningún otro resultado.

A.4.3. El botón Exit

Este botón se usa para terminar la ejecución del programa. Cuando se elige este botón se ejecutan dos funciones: *close all* y *closereq*. Con la función *close all* se cierran todas ventanas que esten abiertas incluyendo la de la interfaz gráfica con el usuario. La función *closereq* cierra todas la figuras que se hayan solicitado o generado durante la ejecución del software.

La interfaz gráfica desarrollada facilita la visualización de los resultados y hace más amable la interacción con el software mediante el uso de cuadros de diálogo para

la entrada de datos y el despliegado de los resultados mediante figuras y gráficas. Por otro lado, también se ofrece el uso del sistema por medio de la terminal de comandos la cual hace más versátil el uso de las funciones y la manipulación de las gráficas para personalizar los resultados de acuerdo con la aplicación que se le vaya a dar posteriormente.

Se ha presentado la descripción del funcionamiento de los tres botones mediante los cuales se controla la interfaz gráfica. La función principal de este software es el proceso que se activa con el botón *Start QSI*.

A.5. Instalación del Software QSI

Se presenta la forma para instalar y operar QSI. El software es independiente de la plataforma del sistema operativo, el requisito más importante es que esté instalado Matlab en sus versiones 5.3, 6.0, 6.5 ó 7.0. El software funciona para las plataformas windows, linux, unix y OSX de mac.

A.5.1. Instalación

1. Crear un subdirectorio para QSI.
2. Desempaquetar el archivo qsi.zip, usando Winzip, dentro del subdirectorio creado en el paso anterior.
3. Agregar este subdirectorio en el path de matlab (“Set Path” in menu “File”).

Apéndice B

Teorema de Nyquist

De acuerdo con el teorema del muestreo de Nyquist [Nyquist46] una señal debe ser muestreada con una frecuencia de al menos $\Omega_s = 2\Omega$, siendo Ω la frecuencia más alta presente en la señal. De esta manera para capturar una señal durante un tiempo t , el número N de muestras necesarias será:

$$N = f_s t = \frac{t}{T} \quad (\text{B.1})$$

pero

$$f_s = \frac{\Omega_s}{2\pi} \geq \frac{2\Omega}{2\pi} \quad (\text{B.2})$$

con lo cual

$$N \geq \frac{\Omega t}{\pi} \quad (\text{B.3})$$

Este teorema es conocido también como el teorema de Shannon.

Una sugerencia práctica es seleccionar la frecuencia de muestreo de al menos 10 veces el ancho de banda del sistema. Lo cual corresponde aproximadamente a por lo menos 5 o 7 muestras durante el tiempo de elevación de la respuesta a una señal escalón [Ljung99].

Apéndice C

Algoritmo Genético del Criador

En esta tesis se usa este esquema de optimización para estimar los coeficientes de la ecuación característica del modelo variante en el tiempo. Este esquema de los algoritmos genéticos se basa en la selección racional que llevan a cabo los criadores de ganado cuando escogen a los padres de la siguiente generación. El Algoritmo Genético del Criador (AGC) usa el método del truncamiento para llevar a cabo la selección. Esto garantiza que solo los mejores individuos sean seleccionados para ser padres. Opuesto a los AG's estándar, en el AGC, los operadores de recombinación y mutación no son operaciones que le dan cierta oportunidad a los individuos, más bien, son operadores que generan individuos con mejores características. Es decir, podemos considerarlos como operadores de construcción, ya que en cada generación van construyendo las soluciones.

A continuación se hace una descripción de los principales operadores del AGC usado en este trabajo.

Selección por truncamiento

Este esquema es un método conocido como de selección artificial por la forma como se realiza. El método selecciona solo a los mejores individuos, i.e., solo se seleccionan los primeros individuos de la población. El parámetro para este método es

la variable Tru llamada umbral de truncamiento. Tru indica cual será la proporción de la población que será seleccionada como los padres, y puede tomar valores entre el 10 y el 50 % [MuhDirk93]. Los individuos fuera de este rango no participan en la reproducción.

El valor para Tru en este trabajo es del 40 %, i.e., el 40 % de los individuos que encabezan la población serán los seleccionados para ser los padres de la siguiente generación.

Recombinación intermedia extendida

Este método es aplicable solo para valores reales. Aquí los valores de los hijos se obtienen en los alrededores y entre los padres. Los hijos son calculados de acuerdo con la formula:

$$hijo = padre1 + \alpha(padre2 - padre1) \quad (C.1)$$

donde:

$padre1$ y $padre2$ son individuos \in a la selección delimitada por Tru .

$\alpha \in [-0.25, 1.25]$, y es el factor de escala seleccionado aleatoriamente dentro del intervalo propuesto por Mühlenbein [MuhDirk93].

Cada parámetro en la variable $hijo$ es el resultado de la combinación de los parámetros en los padres de acuerdo con la expresión de la ecuación C.1 con un diferente α para cada parámetro calculado [MuhDirk93].

Mutación

Los parámetros en los hijos son mutados por medio de la adición de pequeños valores aleatorios dentro del intervalo $[-rango_j, rango_j]$. El valor de la variable $rango_j$ esta calculada como:

$$rango_j = 0.1 \cdot Errormedio_j \quad , \quad j = 1, 2, \dots, Num_gen \quad (C.2)$$

donde $Errormedio_j$ es el promedio del error de los individuos seleccionados como padres de la generación en cuestión.

El nuevo valor para los hijos se calcula entonces como:

$$hijo_i = hijo_i + rango_j \quad (C.3)$$

Esta operación de mutación se aplica para cada una de las variables que forman al hijo, e.i., para cada uno de los parámetros.

Reinserción

El esquema de reinserción utilizado es el conocido como reinserción elitista. Este método se combina con el esquema de ordenamiento previniendo que se pierda información en cada generación. El 40 % de los individuos menos aptos son sustituidos por los hijos obtenidos anteriormente.