



Asunto: Notificación de Dictamen de Propuesta de Tesis

C. González Villagómez Jacob
Estudiante de la Maestría en Ingeniería Electrónica Aplicada
PRESENTE.

Una vez concluido el análisis y discusión sobre la propuesta del proyecto de tesis titulada: **"Identificación de la función de transferencia de un sistema utilizando técnicas adaptativas"**, bajo la revisión del *Asesor(es)*: RODRIGUEZ DOÑATE Carlos, Dr.; en su reunión ordinaria del semestre, el Comité de Ingreso y Permanencia de la acordó la siguiente resolución:

Con fundamento en las fracciones primera a cuarta del Artículo 67 y el Artículo 68 del Estatuto Académico, este comité acordó **POR UNANIMIDAD DE VOTOS ACEPTAR SU PROPUESTA DE TESIS** designando el siguiente jurado:

Presidente: CABAL YÉPEZ Eduardo, Dr.
Secretario: RODRIGUEZ DOÑATE Carlos, Dr.
Vocal: GURYEV V. Igor, Dr.

Se extiende la presente notificación de la resolución el 24 de noviembre del 2017.

ATENTAMENTE
"LA VERDAD OS HARÁ LIBRES"
EL DIRECTOR

DR. VÍCTOR AYALA RAMÍREZ

c.c.p. Dra. Rocío Alfonsina Lizárraga Morales – Secretaria Académica de la División de Ingenierías, – Asesor(es):
RODRIGUEZ DOÑATE Carlos, Dr.

Archivo

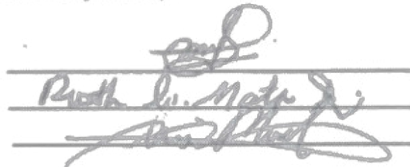
22 de noviembre de 2018, Yuriria, Gto.

Asunto: solicitud de cambio de jurado.

Dra. Rocío Alfonsina Lizárraga Morales
Secretaría Académica de la División de Ingenierías
del Campus Irapuato-Salamanca

Estimada Dra. Rocío por este medio me dirijo de la manera mas atenta y respetuosa, solicitando su apoyo para cambiar parte del jurado del proyecto de tesis del Ing. González Villagómez Jacob, quien espera presentar su defensa de tesis ante el jurado designado a más tardar el 14 de diciembre del 2018, ya que está en el proceso de ingreso al Doctorado de Ciencias de la Ingeniería. El cambio solicitado es al Dr. Igor V. Guryev, quien funge como vocal en el jurado y estará fuera del país a partir del 25 de noviembre del 2018. Cabe destacar que este cambio ya se le notificó al Dr. Igor y estuvo de acuerdo. Por otro lado, los doctores sugeridos para remplazar al Dr. Igor serían los siguientes (ya se les notificó que podrían ser parte del nuevo jurado):

Dr. Juan Manuel López Hernández
Dra. Ruth Ivonne Mata Chávez
Dr. Luis Manuel Ledesma Carrillo.



Esperando contar con su amable apoyo me despido con un cordial saludo y quedo a la espera de su respuesta, para que el Ing. Jacob envíe el documento de tesis a revisión con el nuevo jurado que se designe.

Atentamente



Dr. Carlos Rodríguez Doñate
Director de tesis



VoBo. Dr. Igor V. Guryev
Profesor DEM



Ing. González Villagómez Jacob
Estudiante de MIEA



VoBo. Dr. Geovanni Hernández Gómez
Coordinador de la MIEA

VoBo.
Rocío Lizárraga
23/11/18
Cambio
Dr. Guryev
por
Dr. Luis
Manuel Ledesma
Carrillo

Llenar en computadora con ayuda del oficio de modalidad.

La modalidad de tesis es única para los posgrados

Nivel:	
Licenciatura	
Maestría	X
Doctorado	

Modalidad:	Tesis
-------------------	-------

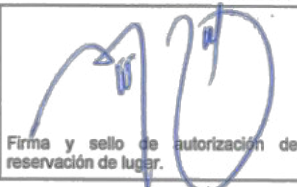
Año:	2018
-------------	------

Marcar con una X

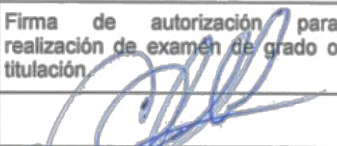


Poner el número de año p.e. 2015

Información sobre Obtención de Grado Académico:	
Nombre	Jacob González Villagómez
NUA	907211
Programa	Maestría en Ingeniería en Electrónica Aplicada

Para modalidades con Jurado completar la siguiente información:

Lugar, hora y fecha de la presentación		
Lugar	Aula magna sede Yuriria de la Universidad de Guanajuato	 Firma y sello de autorización de reservación de lugar.
Hora	11:00 a.m.	
Fecha	Jueves 13 de diciembre 2018	

Título del trabajo	Metodología para la identificación de la función de transferencia de un sistema utilizando técnicas adaptativas.
---------------------------	--

Jurado		
	Nombre con grado académico completo: p.e. Doctor en Informática Industrial Nombre Apellido Paterno Apellido Materno	Firma de autorización para realización de examen de grado o titulación
Presidente	Doctor in Philosophy Eduardo Cabal Yépez	
Secretario	Doctor en Ingeniería Carlos Rodríguez Doñate	
Vocal (1)	Doctor en Ingeniería Eléctrica Luis Manuel Ledesma Carrillo	
Vocal 2 (Doctorado)		
Vocal 3 (Doctorado)		

Asesoría	
Director del trabajo	Dr. Carlos Rodríguez Doñate
Codirector	

(No llenar para uso exclusivo de la Coordinación.)

Valida (nombre y firma): _____

Una vez terminado de llenar imprimir en dos tantos (uno para entregar al iniciar el trámite de autorización del examen de grado o titulación y otro para firma de recibido).

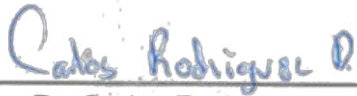
Yuriria, Gto., a 28 de Noviembre del 2018.

**M. en I. HERIBERTO GUTIÉRREZ MARTIN
JEFE DE LA UNIDAD DE ADMINISTRACIÓN ESCOLAR
P R E S E N T E.-**

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) Ing. Jacob González Villagómez del *Programa de Maestría en Ingeniería en Electrónica Aplicada* y cuyo número de *NUA* es: 907211 del cual soy director. El título de la tesis es: Metodología para la identificación de la función de transferencia de un sistema utilizando técnicas adaptativas

Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.

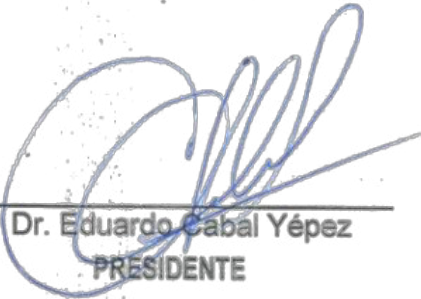
ATENTAMENTE



Dr. Carlos Rodríguez Doñate
DIRECTOR DE TESIS
SECRETARIO



Dr. Carlos Rodríguez Doñate
DIRECTOR DE TESIS



Dr. Eduardo Cabal Yépez
PRESIDENTE



Dr. Luis Manuel Ledesma Carrillo
VOCAL



UNIVERSIDAD DE GUANAJUATO

DIVISIÓN DE INGENIERÍAS

CAMPUS IRAPUATO – SALAMANCA

DEPARTAMENTO DE ESTUDIOS
MULTIDISCIPLINARIOS - YURIRIA

*“Metodología para la identificación de la función de transferencia de
un sistema utilizando técnicas adaptativas”*

TESIS PROFESIONAL

PARA OBTENER EL GRADO DE:

MAESTRÍA EN INGENIERÍA EN ELECTRÓNICA APLICADA

PRESENTA:

Ing. Jacob González Villagómez

ASESOR:

Dr. Carlos Rodríguez Doñate

Resumen

El control automático ha marcado un paso muy importante en el avance de la ingeniería, dando lugar a que algunos de los procesos industriales se automaticen. Aunque se han desarrollado diferentes esquemas de control con técnicas computacionales y de procesamiento de señales, no se ha logrado desplazar el uso de las leyes de control clásico en el diseño de controladores para la automatización. Sin embargo, para poder obtener los coeficientes que definen las leyes de control clásico en el diseño de controladores, es necesario conocer con certeza la función de transferencia del sistema a controlar. Por lo cual, en este trabajo se presenta una metodología para estimar la función de transferencia de un sistema de control utilizando técnicas de procesamiento digital de señales, la cual consiste en definir los parámetros para identificar la función de transferencia y la unidad de sincronización para enviar la entrada y salida de la planta hacia el algoritmo propuesto FIR-RLS. La validación del experimento se realiza en dos casos de estudio, obteniendo una herramienta fiable para estimar la función de transferencia de una planta, ya que se presentan errores absolutos de la respuesta de la estimación menores al 1%. Además, se propone un sistema basado en FPGA para realizar la estimación de la función de transferencia en tiempo real con bajo consumo de recursos.

Dedicatoria

Dedico esta tesis primeramente a mi Señor Jesucristo, porque tengo bien entendido que sin su ayuda no habría podido culminar con este proyecto de vida.

A mis padres, Salvador y Magdalena quienes me apoyaron todo el tiempo, en conjunto con mis hermanos Obed, Esaú y Karina.

A mis familiares que me apoyaron siempre, mis abuelitos Abel y Esther, mi tía Trinidad y mis tíos Ignacio y Eleazar por su apoyo y confianza.

A mi director de tesis Carlos Doñate por brindarme los consejos y la ayuda necesaria para la realización de este trabajo.

A mis maestros quienes nunca desistieron el enseñarme, depositando su esperanza en mí.

A los sinodales quienes estudiaron mi tesis y la aprobaron.

A mis compañeros y amigos de la carrera quienes fueron un gran apoyo emocional durante el tiempo en que escribía esta tesis.

A mi novia Celena Adileni quien me apoyo y alentó siempre para continuar.

A todos los que me apoyaron para escribir y concluir esta tesis, para ellos es esta dedicatoria de tesis, pues su apoyo fue incondicional.

Agradecimientos

Personales

Primeramente, agradecerle a Dios por la vida y la salud que me prestó a lo largo del transcurso de mi preparación educativa para culminar con una carrera profesional.

*Mis más sinceros agradecimientos a mis padres, **Salvador González Villafuerte** y **Ma. Magdalena Villagómez Hernández**, que se sacrificaron y me apoyaron en todo el transcurso de mi desarrollo profesional, sinceramente GRACIAS. A mis hermanos, familiares y amigos por el apoyo y los consejos para poder seguir adelante en este camino difícil que se ha logrado enfrentar para poder culminar con éxito este objetivo planteado en la vida.*

*A mi asesor y director de tesis **Dr. Carlos Rodríguez Doñate**, por los consejos y el apoyo brindado a lo largo de la realización de la presente tesis.*

Institucionales



*A la **Universidad de Guanajuato**, al Departamento de estudios multidisciplinares, por haberme dado la oportunidad de culminar mis estudios de la maestría en sus aulas llenas de sabiduría y glorias.*



*A la institución **Consejo Nacional de Ciencia y Tecnología (CONACYT)** por el apoyo financiero con el número de beca 447181 para lograr la culminación de este trabajo de investigación.*

Índice General

Resumen	i
Dedicatoria.....	ii
Agradecimientos.....	iii
Índice General.....	iv
Índice de Figuras	vi
Índice de Tablas.....	vii
I. Introducción	8
1.1 Antecedentes.....	9
1.2 Objetivos.....	13
1.1.1. Objetivo general	13
1.1.2. Objetivos específicos.....	13
1.3 Planteamiento General	13
II. Fundamento Teórico	15
2.1 Filtrado Digital.....	16
2.1.1 Sistemas con respuesta al impulso de duración finita (FIR)	17
2.2 Filtros Adaptativos.....	19
2.2.1 Aplicaciones de filtros adaptativos	23
2.2.2 Filtrado Óptimo	25
2.3 El algoritmo LMS	29
2.4 El algoritmo RLS	33
2.5 El Filtro Kalman	36
2.5.1 El origen computacional del filtro Kalman	37
2.5.2 El algoritmo del filtro Kalman discreto.....	38
2.6 Identificación de Sistemas	39
III. Metodología.	43
3.1 Desempeño de los algoritmos adaptativos.....	44
3.1.1. Algoritmo LMS	44
3.1.2. Algoritmo RLS.....	47

3.1.3. Filtro Kalman	48
3.2. Evaluación de los algoritmos	50
3.3. Implementación en FPGA	50
3.3.1 Optimización del algoritmo.....	50
3.3.2 Desarrollo del algoritmo RLS en FPGA	55
IV. Pruebas y Resultados	59
4.1. Metodología Propuesta	59
4.2. Etapa Experimental.....	61
4.2.1 Caso de estudio 1: Circuito RC	62
4.2.2 Caso de estudio 2: Motor de DC	64
4.3 Diseño de Controladores.....	65
4.4 Implementación en FPGA	71
4.4.1 Simulaciones en VHDL	72
V. Conclusiones.....	74
Bibliografía.....	76

Índice de Figuras

<i>Figura 1.-Proceso para la validación del trabajo de Investigación</i>	14
<i>Figura 2. Diagrama de bloques de los dos tipos de filtros digitales: (a) FIR y (b) IIR.</i>	17
<i>Figura 3. Sistema (a) No recursivo (b) Recursivo</i>	18
<i>Figura 4. Esquema general de un sistema adaptativo</i>	20
<i>Figura 5. Filtro transversal</i>	22
<i>Figura 6. Estructura de un sistema adaptativo para identificación</i>	23
<i>Figura 7. Estructura de un sistema adaptativo para la obtención de un modelo inverso.</i>	24
<i>Figura 8. Estructura de un sistema adaptativo para predicción</i>	24
<i>Figura 9. Estructura de un sistema adaptativo para la cancelación de interferencia.</i>	25
<i>Figura 10. Diagrama del bloque de la estructura de la estructura LMS</i>	29
<i>Figura 11. Diagrama del Filtro transversal</i>	30
<i>Figura 12. Estructura del mecanismo de control adaptativo para los coeficientes</i>	31
<i>Figura 13. Diagrama a bloques del algoritmo RLS</i>	33
<i>Figura 14. Diagrama a bloques de la formulación matemática del concepto espacio del estado</i>	36
<i>Figura 15. Diagrama de operación completa del filtro Kalman</i>	39
<i>Figura 16.- Proceso de Identificación</i>	41
<i>Figura 17. Esquema para la identificación de sistemas</i>	42
<i>Figura 18.- Desarrollo del proyecto de investigación</i>	43
<i>Figura 19.- Señal de entrada para la excitación del filtro FIR de 6 coeficientes.</i>	45
<i>Figura 20.- Señal de salida del filtro FIR de 6 coeficientes.</i>	45
<i>Figura 21.- Adaptación de los coeficientes mediante LMS.</i>	46
<i>Figura 22.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante LMS, b) Gráfica de Error entre la señal de salida del filtro FIR simulado y el estimado al usar LMS.</i>	46
<i>Figura 23.- Adaptación de los coeficientes mediante RLS.</i>	47
<i>Figura 24.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante RLS, b) Gráfica de error de la señal de salida entre el filtro simulado y el estimado al usar RLS.</i>	48
<i>Figura 25.- Señal de entrada para la identificación mediante el Filtro Kalman</i>	49
<i>Figura 26.- Adaptación de los coeficientes mediante el filtro Kalman.</i>	49
<i>Figura 27.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante el filtro Kalman, b) Gráfica de error de la señal de salida entre el filtro simulado y el estimado al usar el filtro Kalman.</i>	50
<i>Figura 28.- Diagrama a bloques Algoritmo RLS</i>	57
<i>Figura 29.- Diagrama general de la metodología para identificación mediante el algoritmo RLS</i>	60
<i>Figura 30.- Diagrama de la metodología propuesta</i>	61
<i>Figura 31.- Etapa experimental</i>	62
<i>Figura 32.- Circuito RC</i>	62
<i>Figura 33.- Resultados de la identificación de la FT del Circuito RC a) Respuesta de la FT para el Circuito RC, b) Error absoluto</i>	63
<i>Figura 34.- Resultados de la identificación de la FT del Motor de DC a) Respuesta de la FT para el Motor de DC, b) Error absoluto</i>	65
<i>Figura 35.- Caso de estudio 1: Respuesta al escalón y diagrama de Bode del sistema sin controlador.</i>	66
<i>Figura 36.- Caso de estudio 1: Respuesta al escalón del sistema con el controlador PI en lazo cerrado.</i>	66

Figura 37.- Caso de estudio 1: Respuesta al escalón del sistema con el controlador PID en lazo cerrado. 67

Figura 38.- Caso de estudio 1: Comparativa de la Planta con distintos controladores. 67

Figura 39.- Caso de estudio I: a) Comparativa entre la respuesta al escalón de la planta experimental y teórica con controlador PID, b) Grafica del error relativo de las respuestas al escalón de las plantas. 68

Figura 40.- Caso de estudio 2: Respuesta al escalón y diagrama de Bode del sistema sin controlador. 68

Figura 41.- Caso de estudio 2: Respuesta al escalón del sistema con el controlador PI en lazo cerrado. 69

Figura 42.- Caso de estudio 2: Respuesta al escalón del sistema con el controlador PID en lazo cerrado. 70

Figura 43.- Caso de estudio 2: Comparativa de la Planta con distintos controladores. 70

Figura 44.- Caso de estudio II: a) Comparativa entre la respuesta al escalón de la planta experimental y teórica con controlador PID, b) Grafica del error relativo de las respuestas al escalón de las plantas. 71

Figura 45.- Simulación en VHDL de la matriz de correlación del algoritmo RLS. 72

Figura 46.- Simulación en VHDL del vector de coeficientes del algoritmo RLS. 73

Índice de Tablas

Tabla 1.- CASO ESTUDIO I: COMPARATIVA DE CONTROLADORES..... 68

Tabla 2.- CASO ESTUDIO II: COMPARATIVA DE CONTROLADORES..... 71

Tabla 3.- CONSUMO DE RECURSOS DEL FPGA EP2C20F484C7N 72

1

CAPÍTULO

Introducción

El impacto del avance tecnológico en la economía y la sociedad, además de la demanda excesiva de los productos comerciales, tales como automóviles, dispositivos móviles, comestibles, etc., los cuales se desarrollan en diversas empresas, ha ocasionado que éstas automaticen sus procesos para acelerar la producción sin perder la calidad en el producto [1]. Este cambio ha generado el desarrollo de distintos sistemas de control automático, en los cuales se han implementado las leyes de control para regir la dinámica de estos. Un sistema de control automático se compone de tres partes fundamentales: el sensor, el controlador y el actuador. El controlador es la parte encargada de comparar el valor medido con el deseado (error), y de acuerdo con ese valor envía una señal de control. Sin embargo, una de las problemáticas más importantes en el diseño de un controlador en un sistema de control automático, es conocer la función de transferencia del sistema, la cual es una ecuación que relaciona la variable de entrada con la variable de salida del sistema [2].

Debido a la importancia que presenta el conocer la función de transferencia de un sistema, se han propuesto diferentes técnicas para la identificación de un sistema a controlar. Las técnicas utilizadas para la identificación de un sistema se eligen dependiendo de las

características lineales que éste presenta, por ejemplo, para sistemas lineales se utilizan las técnicas tradicionales, destacando de entre ellas los algoritmos de filtros adaptativos, mientras que para sistemas no lineales se han desarrollado algoritmos combinando las técnicas computacionales, tales como, lógica difusa, redes neuronales y algoritmos genéticos [3-4]. En el presente trabajo se analizarán sistemas lineales, por tal motivo las técnicas a utilizar son las relacionadas con los algoritmos de filtros adaptativos, entre los cuales podemos destacar el método de Mínimos Cuadrados Promedio (LMS por sus siglas en inglés), el método de Mínimos Cuadrados Recursivos (RLS), distintos filtros como el filtro de Wiener y el filtro Kalman [5], entre otros.

Con la propuesta del trabajo presentado, se obtendrá una metodología genérica, capaz de identificar la función de transferencia de sistemas de control, enfocando la metodología en el diseño de controladores basados en la ley de control clásico (Proporcional-Derivativo-Integrativo PID y sus variantes), debido a que este tipo de controladores son los más utilizados en el ámbito industrial [6-7]. La metodología obtenida podría ser útil en el ámbito académico, en el campo de investigación y el ámbito industrial. Los algoritmos de filtros adaptativos utilizados en la investigación estarán basados en el método de Mínimos Cuadrados Promedio (LMS por sus siglas en inglés), el método de Mínimos Cuadrados Recursivos (RLS por sus siglas en inglés) y el filtro Kalman; debido a su simplicidad y flexibilidad al ser desarrollados.

1.1 Antecedentes

El control automático ha marcado un paso muy importante en el avance de la ingeniería y la ciencia, por lo cual en la literatura ha sido caso de estudio en múltiples investigaciones. Esto ha promovido la incorporación del control en las distintas áreas de oportunidad que se presentan en la vida cotidiana de la sociedad [8].

Al pasar de los años, los sistemas a controlar en las industrias se han hecho más complejos, ya que las necesidades del ser humano van cambiando con el paso de los tiempos, además, como la población crece el incremento de la demanda de determinados productos (comestibles, de higiene, medicamentos, etc.) necesarios para la subsistencia del hombre también lo hace. Debido al incremento en la complejidad de los mecanismos utilizados en las industrias, es importante conocer la dinámica de dichos mecanismos para controlarlos [9].

La identificación de un mecanismo o sistema de control es la primera etapa para poder realizar el diseño de un controlador mediante las leyes de control convencionales [10]. Para llevar a cabo este proceso, se ha demostrado que el método experimental es uno de los más fiables para realizar la identificación de un sistema de control [3-4]. El método experimental

requiere de la señal de entrada y la de salida del sistema para desarrollar un modelo matemático confiable, el cual representará la dinámica del sistema a controlar.

El tema acerca de la identificación de los sistemas se ha venido desarrollando desde tiempos muy remotos. En las últimas décadas, se han propuesto diversos métodos para llevar a cabo la identificación de un sistema (función de transferencia). Un trabajo pionero acerca de la identificación de sistemas es el realizado por Astrom et al. en [3], el cual es la base de muchos de los trabajos descritos en los siguientes párrafos. Algunos ejemplos de los sistemas de control que pueden ser identificados son: los robots o maquinarias industriales, los mecanismos utilizados en aeronaves, los robots de exploración espacial, los sistemas económicos e incluso los dispositivos utilizados con biotecnología, entre otros [4]. En un principio, la función de transferencia se determinaba mediante la representación matemática de cada uno de los componentes del sistema, o por medio de la excitación del sistema debido a una señal de entrada definida y posteriormente se calculaba analíticamente [11-16]. Fue hasta mediados del siglo XX, cuando se empezaron a proponer diferentes metodologías para obtener la función de transferencia de un sistema de forma distinta, tal es el caso de Weygandt et al. en [17] donde describieron un sistema para determinar los parámetros de la función de transferencia, por medio de dividir una constante por un polinomio en la variable laplaciano. Posteriormente Nagumo et al. en [18] propuso un método de identificación basado en el error y la corrección, al cual llamo “identificación de aprendizaje”, los resultados los presentó mediante simulaciones computacionales. Ljung et al. en [19] determinaron un teorema de convergencia para los parámetros del método de Mínimos Cuadrados Promedio (LMS por sus siglas en inglés), ya que para esa época este método era el más conocido para identificar un sistema, así como las condiciones para determinar el funcionamiento del método en un sistema de lazo cerrado.

Después de un año, Gustavsson et al. en [20] hizo una recopilación de los distintos trabajos desarrollados aproximadamente hasta los años 70's para la identificación de sistemas de lazo cerrado, dando como conclusión que, los distintos enfoques de los métodos de predicción de errores, utilizados para la identificación de sistemas lineales de lazo cerrado, son buenos y eficientes. Además, se demostró que la identificación de un sistema también es posible prácticamente, y no solamente teóricamente, mostrando dos ejemplos donde se aplicaron estos métodos. Diez años más tarde Hunt et al. en [21] realizó una investigación acerca de la identificación de sistemas con algoritmos recursivos, donde compara cuatro métodos de algoritmos recursivos, los cuales son Mínimos Cuadrados Recursivos (RLS), Variables Instrumentales Recursivas (RIV), Máxima Probabilidad Recursiva (RML) y Cuadrados Mínimos Extendidos (ELS); demostrando las similitudes y diferencias entre ellos y remarcando los problemas en cada método, entre los cuales se encuentran la complejidad del modelo, la relación señal a ruido, la convergencia y la demanda computacional del modelo. Ya en los años noventa Unbehauen en [22] realizó una recopilación de trabajos acerca de los distintos enfoques para la identificación de sistemas en tiempo real, donde

menciona las técnicas basadas en la estimación de los parámetros del sistema a través de los modelos de error, los cuales se generan por la diferencia de dos señales del sistema. En el artículo también propuso otras técnicas para la identificación de sistemas no lineales. Cada uno de los trabajos mencionados anteriormente, han sido la base para el desarrollo de nuevos enfoques de las técnicas en la actualidad, con el fin de mejorar el modelo de los algoritmos o métodos de identificación.

Recientemente, en los trabajos de investigación se han propuesto diferentes enfoques basados en la teoría de filtros adaptativos para la identificación de sistemas de control o componentes de los mismos (perturbaciones, parámetros, etc), modificando la estructura o parámetros del modelo para la solución de las deficiencias que presentaban los trabajos mencionados anteriormente. Por ejemplo, en la propuesta presentada por Benesty et al. [23] realizaron el proceso de identificación para sistemas de múltiples entradas y una salida, el cual se basa en el filtro Wiener y un conjunto con un término bilineal definidos por las respuestas al impulso de un sistema, el artículo presenta el desarrollo de dos versiones del filtro Wiener.

El método del algoritmo basado en LMS, ha sido la base para diferentes propuestas, como la presentada por Prasad et al. [24] donde mostraron el desarrollo teórico del método y la implementación del mismo en Matlab con una señal aleatoria. Los resultados presentados son atractivos, aunque solo se reportan resultados de simulaciones al implementar el LMS en Matlab; de forma similar Ciochină et al. [25] presentaron la optimización del método para identificar un sistema en dos versiones. La optimización se basó en minimizar la desalineación del sistema, en el trabajo se presentan únicamente simulaciones en el contexto de la cancelación de eco, pero no en la identificación de un sistema. En [26] Dash et al. implementaron en una tarjeta de desarrollo de Arreglo de Compuertas Programables de Campo (FPGA) el filtro LMS de primer orden para identificar un sistema, sin embargo, no validan los resultados obtenidos con un sistema de control real, por lo cual solo mostraron simulaciones. Muy similar al trabajo anterior, Safarian et al. en [27] propusieron una arquitectura en FPGA más eficiente para la implementación del algoritmo, presentaron la comparación de los recursos y eficiencia de la arquitectura propuesta, aunque solo en simulaciones. El algoritmo LMS tiene muchas variaciones, por mencionar alguna, Deshmuk en [28] presentó una variación del algoritmo de Mínimos Cuadrados Proporcional Normalizado (MPNLMS) en el tamaño de paso (μ), resolviendo el problema de la lenta convergencia que tiene dicha variación del modelo cuando su entrada es altamente correlacionada.

Derivado del método anterior, el algoritmo basado en el método de RLS, es la base de la propuesta presentada por Wang et al. en [29], la cual se utiliza para identificar la función de transferencia de un robot, en el trabajo presentaron la simulación del algoritmo de cuarto grado en la trayectoria que debe seguir el mecanismo, las dos simulaciones tienen un comportamiento efectivo, una simulación la realizan sin carga y la otra con carga en el robot,

mostrando que la dinámica del robot es la adecuada en los dos casos. Utilizando también el algoritmo adaptativo basado en el método RLS, Mórales et al. en [30] desarrollaron un sistema en FPGA para la identificación de un sistema en línea, en el artículo se presenta dos casos de estudio con resultados favorables. En [31] Askari presentó una variación del algoritmo RLS, donde incorporó factores de olvido adaptativos (parámetro λ determina la memoria del algoritmo), los cuales ayudan al modelo para estimar cambios abruptos. Esta variación también permite la identificación simultánea de varios parámetros variables en el tiempo del sistema, además puede ser usado para sistemas lineales y no lineales. También el artículo proporciona la validación del sistema mediante simulaciones de diferentes casos de estudio.

Otra técnica utilizada para el proceso de identificación es el filtro Kalman. A pesar de su complejidad Chubich et al. en [32] presentaron el desarrollo matemático de la implementación del filtro Kalman para identificar un sistema discreto lineal Gaussiano. En el artículo no se presentan simulaciones ni aplicación del método, sino solamente la propuesta del sistema. De forma similar a los métodos mencionados en los párrafos anteriores, el filtro Kalman también ha sufrido variaciones para resolver los problemas con los que se enfrentan los expertos, tal es el caso de Bogdanski en [33], donde presentó una herramienta la cual se basa en el filtro Kalman extendido, esta herramienta sirve para la identificación de sistemas no lineales, la estructura presentada en la propuesta es un sistema completamente negro, es decir, no requiere el proceso de identificación exitosa mientras se asegure un buen acondicionamiento de parámetros.

Otras propuestas utilizan la combinación de diferentes técnicas para facilitar el proceso de identificación como lo desarrollaron Sun et al. en [34], donde presentaron las características y condiciones para realizar un algoritmo híbrido entre el filtro Kalman y lógica difusa para identificar un sistema, en el trabajo mostraron las bases teóricas de la propuesta, como lo son las reglas para implementar la lógica difusa, pero no mostraron las simulaciones o la implementación de la propuesta; también Davanipour et al. en [35] propusieron un algoritmo de aprendizaje híbrido usado para la identificación de un sistema no lineal. La propuesta se basa en inicializar el sistema mediante K-means y después actualizar los parámetros para que el sistema aprenda por las redes neuronales. El artículo presenta simulaciones del sistema, pero no aplicaciones en un sistema real.

El desarrollo de los algoritmos adaptativos se ha implementado en distintas plataformas, pero resalta más la utilización del software de MATLAB para probar dichos algoritmos. Tal es el caso de Millán en su proyecto final de carrera [36] donde desarrolló los algoritmos adaptativos LMS, RLS y el filtro Kalman en el software de MATLAB presentado simulaciones de cada uno de ellos. De forma similar Chuz et al. en [37] implementó los algoritmos adaptativos RLS, LMS, algoritmo Adaptativo de Modulo Constante (CMA) y el algoritmo adaptativo de Inversión de Matriz Directa (DMI) en el software de MATLAB, en el trabajo presentó las definiciones, simulaciones y ventajas-desventajas de cada uno de los

algoritmos mencionados, comparando diversos factores para la identificación de antenas en los algoritmos (convergencia, número de antenas, potencia, etc.).

Al revisar la literatura se puede notar que hay numerosos trabajos donde se ha implementado distintas técnicas para la identificación de la función de transferencia de un sistema control, sin embargo, la mayoría de los trabajos presentan resultados únicamente de simulaciones de sus propuestas, y no presenta una validación de los resultados presentados con un sistema de control real. En este trabajo se propondrá una metodología basada en técnicas adaptativas para realizar el proceso de identificación de la función de transferencia en un sistema de control en tiempo real. Además, la propuesta se implementará en un FPGA y se realizará una etapa de validación experimental en un sistema de control real.

1.2 Objetivos

1.1.1. Objetivo general

Proponer una metodología basada en técnicas adaptativas para realizar la identificación de la función de transferencia en un sistema de control, la cual se utilizará para la sintonización de un controlador.

1.1.2. Objetivos específicos

- Implementar en Matlab los algoritmos adaptativos Mínimos Cuadrados Promedio (LMS), el método de Mínimos Cuadrados Recursivos (RLS) y el filtro Kalman, para poder identificar un sistema de control.
- Comparar el desempeño de la función de transferencia identificada con los algoritmos adaptativos implementados en Matlab, con respecto a la función de transferencia identificada con el toolbox “IDENT” de Matlab, para corroborar su correcto funcionamiento.
- Optimizar el algoritmo adaptivo con mejor desempeño para diseñar un sistema de identificación en tiempo real, basado en un FPGA.
- Diseñar diferentes controladores utilizando la función de transferencia obtenida con el sistema de identificación implementado en FPGA en un sistema de control real, para validar el método.

1.3 Planteamiento General

El trabajo de investigación se desarrolló de acuerdo con el procedimiento presentado en la figura 1.

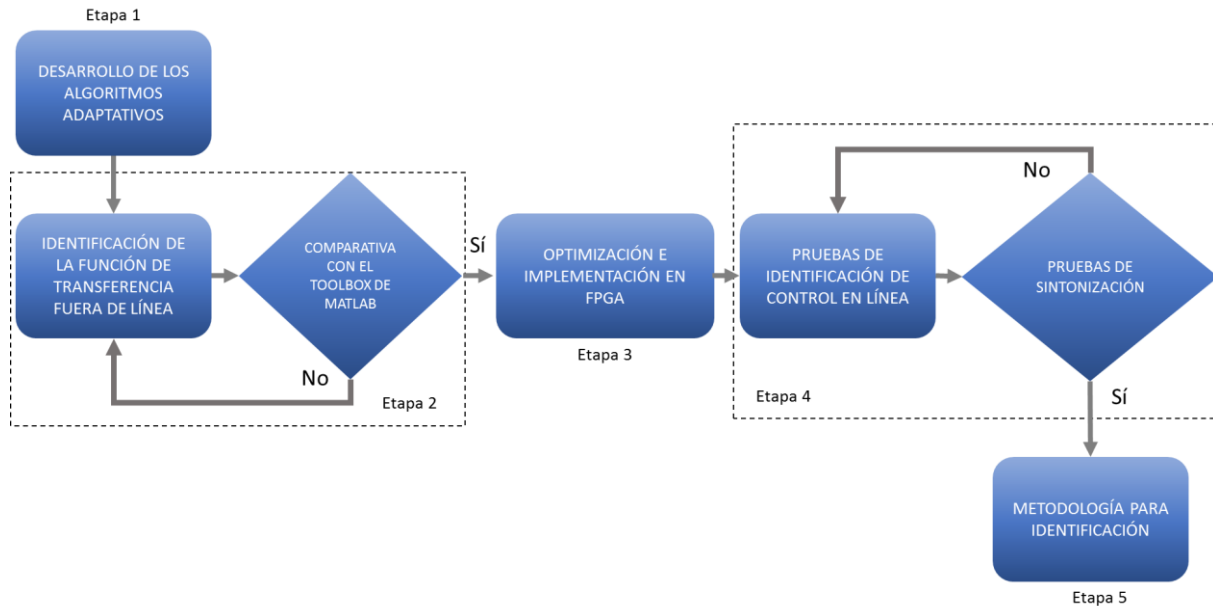


Figura 1.-Proceso para la validación del trabajo de Investigación

A continuación, se describirá brevemente el desarrollo de cada una de las etapas mostradas en la figura 1.

En la primera etapa se desarrollarán los algoritmos o filtros adaptativos en el software de MATLAB, los cuales son: el método Mínimos Cuadrados Promedio (LMS), el método de Mínimos Cuadrados Recursivos (RLS) y el filtro Kalman. En la etapa dos, se evaluarán los algoritmos mediante la identificación de la función de transferencia de distintos sistemas fuera de línea (simulados). Los aspectos para la evaluación de los algoritmos son: la precisión de la identificación, el tiempo de convergencia, la demanda computacional del filtro y la complejidad al implementarlo. La validación del algoritmo se hará mediante la comparación de la respuesta que presente el toolbox “IDENT” de MATLAB, al identificar el mismo sistema. En la validación del algoritmo se dictaminará si la identificación de la función de transferencia del sistema de control es óptima, si no es así se rediseñarán los algoritmos o parámetros de las técnicas adaptativas. Si el resultado es satisfactorio, Se proseguirá con la etapa 3, donde se diseñará un sistema basado en VHDL para implementar el algoritmo óptimo mediante un FPGA.

En la etapa 4 se realizarán pruebas de identificación de un sistema de control en tiempo real. El sistema de control se someterá a distintas señales de entrada para obtener su comportamiento en tiempo real. Y mediante el algoritmo adaptativo, se determinará los parámetros que regirán la función de transferencia de la planta del sistema. Obtenido el modelo del sistema, se estimará la función de transferencia del sistema y se propondrá una ley de control que rijan el comportamiento del sistema modelado. Para validar la metodología, se sintonizarán los parámetros de control para regir el comportamiento deseado del sistema y se implementará para observar el comportamiento con el controlador propuesto. Posteriormente en la etapa 5 se determinará y resumirá el proceso para obtener la metodología para identificar la función de transferencia de un sistema.

2

CAPÍTULO

Fundamento Teórico

El diseño de filtros para realizar estimaciones de señales es un problema que frecuentemente surge en el diseño de sistemas de comunicaciones, sistemas de control, en geofísica y en muchas otras aplicaciones y disciplinas. Para poder diseñar un filtro, se debe tener conocimiento de la señal que se desea procesar, para posteriormente mediante el filtro obtener la información de interés de dicha señal.

El término “filtro” se define habitualmente para describir un dispositivo (software o hardware) que se aplica a un conjunto de datos ruidosos (señal) para extraer una cantidad de información previamente definida como la respuesta deseada o la señal de interés. Un filtro puede ser usado para tres tareas básicas de procesamiento sobre alguna señal o información:

1. Filtrado, Consiste en extraer la información o datos de interés de una señal contaminada en un intervalo de tiempo determinado (Período de muestreo), y emplear esos datos, con el mismo intervalo de tiempo, para reconstruir la salida de la señal ya sin ruido.
2. Suavizado, Difiere del filtrado en que la información de interés no necesariamente debe estar disponible en el intervalo de tiempo determinado, y

los datos posteriores al intervalo de tiempo pueden ser usados para inferir la información de interés. En el caso del suavizado se produce un retardo en la salida.

3. Predicción, Este es un tratamiento preventivo que se le hace a la señal que contiene la información de interés. El objetivo aquí es obtener la información sobre cómo será la información de interés de la señal en algún momento en el futuro, a partir de la observación de los valores de los datos de la señal en estudio.

Cuando se busca una aproximación para resolver el problema de filtrado lineal, se dispone de ciertos parámetros estadísticos (por ejemplo, el promedio o media aritmética y la correlación) de la señal deseada y el ruido aditivo no deseado. El objetivo es diseñar un filtro lineal con los datos ruidosos como entrada y posteriormente, minimizar los efectos de dicho ruido a la salida del filtro según algún criterio estadístico. Una solución útil a este problema de optimización del filtro es minimizar el valor cuadrático medio de la señal de error que se define como la diferencia entre alguna respuesta deseada y la salida. En la literatura, mediante filtros adaptativos se han dado diversas soluciones a este problema de optimización, entre los cuales se destacan: el filtro Wiener, el filtro Kalman, y las estimaciones por medio de mínimos cuadrados (LMS, RLS).

2.1 Filtrado Digital

El filtrado de una señal puede realizarse de forma analógica o digital. En casi todas las aplicaciones donde son usados los filtros subyace una de las dos posibles formas de considerar un filtro:

- En el dominio de la frecuencia: los filtros actúan modificando la amplitud y fase de las componentes sinusoidales de una señal. Debido a que cualquier señal puede considerarse como suma de sinusoides a través de la transformada de Fourier, los filtros pueden ser usados para eliminar determinadas componentes espectrales, enfatizar o atenuar algunas componentes espectrales, desfasar de forma distinta las diferentes componentes espectrales, entre otras.
- En el dominio del tiempo: la relación entre la entrada y la salida de un filtro viene dada por la convolución, lo cual indica que es una suma (finita o infinita) de valores ponderados de la entrada en ciertos instantes de tiempo. En función de los coeficientes de ponderación los filtros realizan un promedio en un determinado intervalo de tiempo, o que buscan las transiciones en el mismo. Algunas de las aplicaciones son la limitación del ancho de banda, controles de tono en audio, ecualizadores de audio, entre otras.

Los filtros analógicos suelen implementarse con elementos que dependen de la frecuencia; típicamente se usan bobinas, condensadores, resistores entre otros. Mientras que un filtro digital se puede describir mediante una ecuación que relaciona una señal de entrada

con una señal de salida; la salida del filtro se especifica como un resultado de sumas, restas y multiplicaciones de muestras de entrada actuales y anteriores. Dichas ecuaciones se denominan técnicamente como ecuaciones lineales en diferencias, ya que, si la entrada de un filtro es la suma de dos funciones escaladas, la salida del filtro es igual a la suma escalada de las salidas del filtro para cada una de dichas funciones. Algunas de las ventajas que presenta el filtrado digital en comparación del analógico son [38]: la insensibilidad a condiciones externas (especialmente la temperatura), el filtrado de varias señales a la vez, la integración de varios filtros en un único chip, un gran margen dinámico (ya que en filtros analógicos es difícil lograr en la práctica atenuaciones mayores de 60–70 dB), la repetitividad (capacidad de realizar cientos o miles de filtros idénticos), la reprogramación, la realización de funciones adicionales, entre otras. Mientras algunas de las limitantes del filtrado digital son: la calidad y velocidad del conversor analógico-digital y digital-analógico (número de bits), los efectos de precisión finita (redondeo) y el tiempo y coste del desarrollo del hardware.

El funcionamiento de los filtros digitales se puede observar en la figura 2, donde se pueden clasificar de acuerdo con la duración de su respuesta al impulso en filtros con respuesta impulsional finita (FIR, del inglés Finite Impulse Response), en el cual se retarda ligeramente una copia de la señal de entrada y se suma con la nueva señal de entrada; o filtros con respuesta impulsional infinita (IIR, del inglés Infinite Impulse Response), en el cual se retarda una copia de la señal de salida y se suma con la nueva señal de entrada.

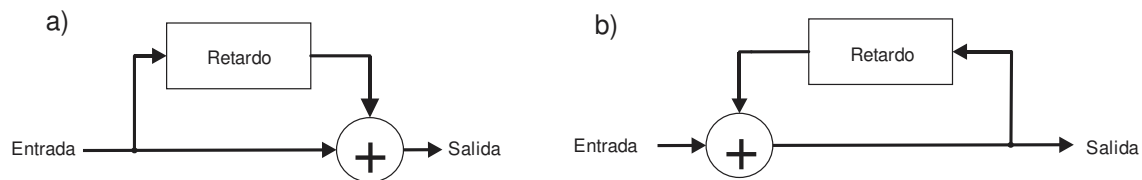


Figura 2. Diagrama de bloques de los dos tipos de filtros digitales: (a) FIR y (b) IIR.

2.1.1 Sistemas con respuesta al impulso de duración finita (FIR)

Un sistema FIR tiene una respuesta al impulso que es cero fuera de un determinado intervalo finito. La fórmula de la convolución para un sistema así se reduce a la ecuación 1.

$$y(n) = \sum_{k=0}^{M-1} h(k)u(n-k) \quad (1)$$

La ecuación 1 indica que la salida en cualquier instante n es simplemente una combinación lineal ponderada de las muestras de la señal de entrada $u(n), u(n-1), \dots, u(n-M+1)$, es decir, el sistema simplemente pondera mediante los valores de la respuesta al impulso $h(k), k = 0, 1, \dots, M-1$, las M muestras de la señal más recientes y suma los M productos resultantes. En efecto, el sistema actúa como una ventana que sólo ve

las M muestras de la señal de entrada más recientes para formar la salida, en otras palabras, un sistema FIR tiene memoria finita de M muestras, donde M representa la longitud del filtro. Un sistema FIR puede describirse mediante ecuaciones en diferencias basándose directamente en la convolución (Ec. 1), pues implica una realización de sumas, multiplicaciones y un número finito de posiciones de memoria, dando como resultado la ecuación 2.

$$y(n) = h(0)u(n) + h(1)u(n - 1) + \dots + h(M)u(n - M) \quad (2)$$

La fórmula de la convolución expresa la salida del sistema lineal invariante en el tiempo explícitamente y sólo en función de la señal de entrada (sistemas no recursivos, ver Fig. 2), sin embargo, existen muchos sistemas en los que es necesario o deseable expresar la salida del sistema no sólo en función de los valores actual y pasados de la entrada, sino también en función de los valores de la salida pasados ya disponibles, estos sistemas son llamados recursivos y deberán tener un número finito de retardos o necesitará un número finito de posiciones de memoria para poder ser implementado en la práctica (ver Fig. 3).

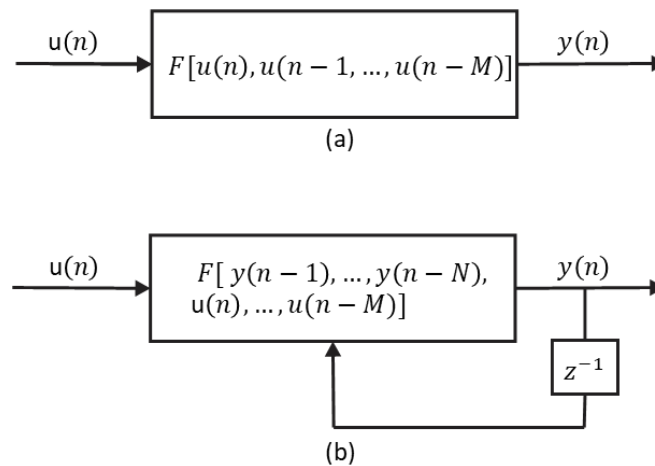


Figura 3. Sistema (a) No recursivo (b) Recursivo

La diferencia fundamental entre los dos sistemas presentados en la figura 2 es el lazo de realimentación existente en el sistema recursivo, que realimenta la salida del sistema a la entrada. Este lazo de realimentación contiene un elemento de retardo. La presencia de este retardo es crucial para poder implementar el sistema, ya que la ausencia de este forzaría al sistema a calcular $y(n)$ en función de $y(n)$, lo que no es posible en los sistemas discretos en el tiempo.

Existen varios métodos para diseñar filtros FIR entre los cuales se encuentran el método utilizando ventanas, el método basado en el muestreo en frecuencia, el método con rizado constante optimo, entre otros. A continuación, se describirá el método utilizando ventanas ya que fue el que se utilizó en el trabajo presentado.

El diseño de un filtro FIR consiste en determinar los coeficientes del filtro a partir de la especificación de la respuesta en frecuencia deseada ($H_d(\omega)$), la cual determina la respuesta correspondiente al impulso ($h_d(n)$). Así $h_d(n)$ está relacionada con $H_d(\omega)$ por la transformada de Fourier como se muestra en la ecuación 3.

$$H_d(\omega) = \sum_{n=0}^{\infty} h_d(n)e^{-j\omega n} \quad (3)$$

Donde

$$h_d(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(\omega) e^{j\omega n} d\omega \quad (4)$$

En general, la respuesta al impulso $h_d(n)$ obtenida a partir de (3) es infinita en duración y tiene que truncarse en algún punto, por ejemplo, en $n = M - 1$, para proporcionar un filtro FIR de longitud M . Truncar $h_d(n)$ a una longitud $M - 1$ es equivalente a multiplicar $h_d(n)$ por una ventana ($w(n)$), en este caso una rectangular. Al multiplicar la función de ventana $w(n)$ por $h_d(n)$ es equivalente a convolucionar $H_d(\omega)$ con $W(\omega)$, donde $W(\omega)$ es la representación en el dominio de la frecuencia (transformada de Fourier) de la función de ventana (ecuación 5).

$$W(\omega) = \sum_{n=0}^{M-1} w(n)e^{-j\omega n} \quad (5)$$

Luego la convolución de $H_d(\omega)$ con $W(\omega)$ proporciona la respuesta en frecuencia del filtro FIR (truncado). Al obtener el módulo de la función de ventana en su respuesta frecuencial, se puede observar que la anchura del lóbulo principal [la anchura que se mide en el primer cero de $W(\omega)$] cuando M aumenta se hace más estrecho. Sin embargo, la altura de los lóbulos secundarios de $|W(\omega)|$ es ligeramente mayor, pero no se ven afectados por un incremento de M . De hecho, incluso aunque la anchura de cada lóbulo secundario disminuye cuando se produce un incremento de M , su altura aumenta cuando aumenta M de tal manera que el área bajo el lóbulo secundario permanece invariante ante los cambios de M . Existen distintos tipos de ventanas entre las cuales se encuentran Bartlett (triangular), Blackman, Hamming, Hanning, Kaiser, Lanczos, Tukey, entre otras.

2.2 Filtros Adaptativos

Los filtros adaptativos han recibido una atención considerable por parte de los investigadores durante los últimos años debido a su amplio campo de aplicación, como resultado se han desarrollado muchos algoritmos de cálculo eficientes para el filtrado

adaptativo. La mayoría de los algoritmos que se emplean en este tipo de filtros son de coeficientes ajustables, es decir, permiten adaptar los coeficientes del filtro a los parámetros estadísticos de la señal. En otras palabras, un filtro adaptativo es aquel cuyos coeficientes son actualizados mediante un algoritmo que cumple con un criterio predefinido (por ejemplo, minimizar el error cuadrático medio). Todos los sistemas adaptativos independientemente de su aplicación responden a un esquema general, el cual se muestra en la figura 4.

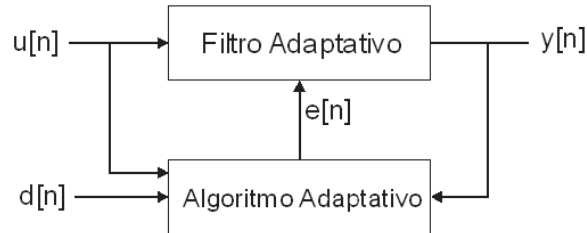


Figura 4. Esquema general de un sistema adaptativo

Donde

u = la entrada aplicada al filtro adaptativo.

y = la salida del filtro adaptativo.

d = respuesta deseada o señal de referencia.

$e = d - y$ = error de estimación.

En dicho esquema existen dos elementos básicos, el filtro adaptativo que contiene los coeficientes ajustables que se actualizan en cada iteración y el algoritmo de adaptación o adaptativo, el cual recibe como entrada la señal de entrada ($u(n)$), la salida del filtro ($y(n)$) y una señal auxiliar llamada señal de referencia o deseada ($d(n)$). El algoritmo de adaptación compara la salida y la señal de referencia y trata de calcular los mejores coeficientes del filtro para que $y(n)$ se parezca (de forma estadística) a $d(n)$.

La aplicación de coeficientes ajustables surge de la problemática en ciertos filtros al momento de su ejecución (Wiener), ya que requieren de información previa de los datos a procesar para lograr un funcionamiento óptimo. Una solución a este problema es realizar el proceso de “estimación y actualización”. Este es un proceso que consta de dos fases, primero el filtro “estima” los parámetros estadísticos de la señal (coeficientes) y después “actualiza” los resultados obtenidos en la fórmula no recursiva, para calcular el error y luego comenzar de nuevo la estimación. La velocidad del proceso de estimación y actualización puede variar según la implementación y el tipo de señales que se manejen, y es evidente que, si se actualizan los coeficientes con mayor frecuencia o se aumenta el número de coeficientes, la respuesta y el desempeño del filtro mejoran, sin embargo, recalculando los coeficientes con cada nueva muestra aumenta el número de cálculos matemáticos a realizar, aumentando el costo computacional. Por lo tanto, en la aplicación de un algoritmo recursivo, los parámetros del

filtro se vuelven datos dependientes de la señal de entrada del sistema y del error que se genera en la estimación de los parámetros.

En resumen, el funcionamiento de un algoritmo para filtrado adaptativo consta de dos procesos básicos: (1) un proceso de *filtrado*, diseñado para producir una salida correspondiente a los datos de entrada, y (2) un proceso *adaptativo*, el cual provee un mecanismo de control adaptativo en un conjunto de parámetros que se usarán en el proceso de filtrado. Estos dos procesos trabajan interactivamente entre sí dependiendo del error ($e(n)$) generado al modificarse o actualizar los coeficientes, ya que $e(n)$ permite definir las mejoras del filtro y determinar la forma en que han de modificarse dichos coeficientes.

En la literatura se han desarrollado una amplia variedad de algoritmos recursivos para la operación de filtros adaptativos, sin embargo, la elección de un algoritmo con respecto a otro está determinado por alguno de los siguientes factores [5]:

- ✚ *Velocidad de convergencia*. Se define como el número de iteraciones requeridas por el algoritmo para alcanzar una solución óptima. Una tasa o velocidad alta permite que la convergencia se alcance rápidamente, pero sacrificando el ajuste óptimo.
- ✚ *Desajuste (Precisión)*. Parámetro que proporciona una medida cuantitativa, acerca de cuanto se desvía el valor final del error cuadrático (promediado sobre un conjunto de filtros adaptativos) con respecto al medido.
- ✚ *Seguimiento*. Capacidad para adaptarse a las variaciones del entorno. El comportamiento de seguimiento del algoritmo, sin embargo, está influenciada por dos características opuestas: (a) la velocidad de convergencia, y (b) la fluctuación del estado estable debido al ruido del algoritmo.
- ✚ *Robustez*. Para que un filtro adaptativo sea robusto, debe cumplirse que pequeños disturbios (es decir, perturbaciones con poca energía) sólo causen errores de estimación pequeños. Las perturbaciones pueden originarse por una variedad de factores, internos o externos al filtro.
- ✚ *Requisitos computacionales*. Parámetro que depende de dos factores, (a) el número de operaciones (es decir, multiplicaciones, divisiones, y sumas/restas) requeridas en cada iteración completa del algoritmo y (b) el tamaño de la memoria disponible para guardar los datos y el programa.
- ✚ *Estructura*. Se refiere a la ruta a seguir por el flujo de información dentro del algoritmo y está determinada por los criterios de diseño del programa. Por ejemplo, un algoritmo podría ser de estructura de programación modular (con subrutinas), o lineal (procedimientos seguidos, uno después del otro), operaciones en paralelo, etc.
- ✚ *Propiedades numéricas*. Cuando se ejecuta un algoritmo, numéricamente se producen inexactitudes debido a los errores de cuantificación. Los errores de cuantificación se deben a la conversión analógico-digital de los datos de entrada y la manipulación digital de los cálculos interiores. Comúnmente, es la última fuente de errores que se evalúa, pero significa un problema muy serio de diseño si no se toma en cuenta.

La importancia de tomar en cuenta los factores mencionados con anterioridad, es que al elegir un filtro adaptativo para resolver un problema se obtenga el resultado deseado, y esta elección sea óptima. Por otra parte, y además de lo mencionado en los párrafos

anteriores, la eficiencia de un filtro adaptativo lineal también depende del tipo de filtro (IIR o FIR), la estructura de este (transversal, de celosía o sistólico), o la función de costo usada como criterio de adaptación (error cuadrático medio, mínimo error cuadrático, etc.). Algunas características de los filtros FIR al ser usadas en los filtros adaptativos son:

- El error cuadrático medio para un filtro transversal es una función cuadrática de los pesos del filtro. La superficie de error es un paraboloide con sólo un mínimo, y por ello, la búsqueda del error cuadrático medio mínimo es relativamente sencilla.
- Dado que los coeficientes del filtro son limitados, se puede controlar fácilmente la estabilidad del filtro.
- Existen algoritmos para la actualización de los coeficientes que con filtros FIR son mucho más simples y eficientes.
- Las prestaciones de estos algoritmos son perfectamente conocidas en términos de convergencia y estabilidad.

De los tres tipos de estructuras de filtro que se distinguen en el contexto de un filtro adaptativo lineal, en el presente trabajo se enfoca en la estructura transversal por las características presentadas anteriormente.

Los filtros transversales, también llamados filtros directos de pesos retardados, consisten en tres elementos básicos, como se observa en la figura 5: (a) elementos de unidad de retardo, (b) el multiplicador, y (c) la sumatoria. El número de elementos de retardo usados en el filtro, determinan la respuesta finita al impulso (FIR). El número de elementos de retardo, mostrado como $M - 1$ en figura 5, normalmente está referido al orden del filtro.

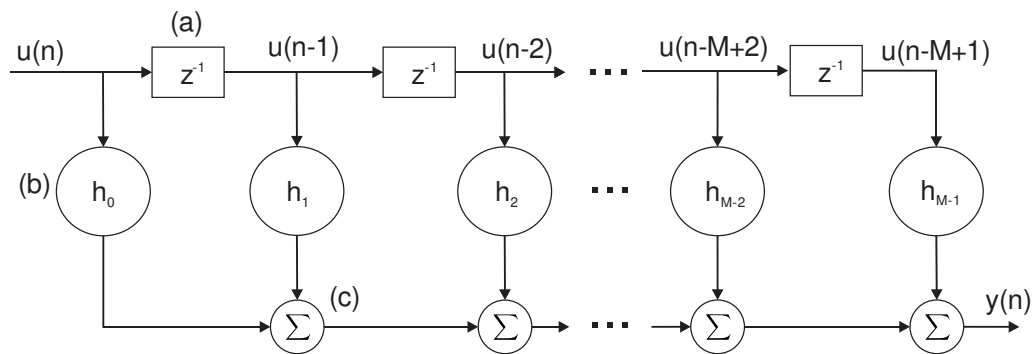


Figura 5. Filtro transversal

En la figura 4, los elementos de retardo son identificados por el operador de la unidad de retraso z^{-1} . En particular, cuando z^{-1} opera en la entrada $u(n)$, la salida resultante es $u(n - 1)$. El papel de cada multiplicador en el filtro es realizar el producto del valor de entrada por un coeficiente del filtro, así un multiplicador conectado k veces a entrada retardadas $u(n - k)$ produce la versión del escalar de que es el producto interno, $h_k u(n - k)$, donde h_k va desde $k=1$ a M . El papel combinado de las sumatorias en el filtro es sumar los resultados de los productos individuales y producir una salida total del filtro. La salida está dada por la ecuación 1. La estructura transversal es la más sencilla de implementar, conduciendo a algoritmos igualmente sencillos.

Sin embargo, no existe una solución única al problema de filtros adaptativos lineales. Más bien, se tiene un “conjunto de herramientas” representado por una variedad de algoritmos recursivos, donde cada uno de los cuales ofrecen ventajas sobre las desventajas de los demás. El desafío es, primero, poder entender las capacidades y limitaciones de un algoritmo y segundo, usar esta información para hacer la mejor selección del algoritmo más apropiado para la aplicación.

2.2.1 Aplicaciones de filtros adaptativos

La habilidad de un filtro adaptativo de operar de manera satisfactoria en un ambiente desconocido rastreando las variaciones estadísticas en el tiempo de una entrada, hace a los filtros adaptativos poderosos dispositivos para aplicaciones de procesamiento de señales y control. Es por lo cual los filtros adaptativos han sido aplicados en diversos campos, por ejemplo, en las comunicaciones, radar, sonar, sismología, y la ingeniería biomédica, por mencionar algunos. Aunque estas aplicaciones son de hecho bastante diferentes en naturaleza, no obstante, tienen un rasgo común básico: un vector de entrada y una respuesta deseada, las cuales se usan para calcular el error de estimación, que a su vez se usa para controlar los valores de un conjunto de coeficientes ajustables. Los coeficientes ajustables pueden tomar la forma de pesos regulables (del inglés tap), coeficientes de reflexión, parámetros de rotación, o pesos sinápticos, dependiendo de la estructura del filtro empleada. Sin embargo, la diferencia esencial entre las varias aplicaciones de filtrado adaptativo comienza con la manera como se extrae la respuesta deseada. En este contexto, podemos distinguir cuatro clases básicas de aplicaciones de filtros adaptativos.

Las cuatro aplicaciones básicas de filtros adaptativos son:

- I. **Identificación.** La notación de un modelo matemático es fundamental para la ciencia e ingeniería. En la identificación, un filtro adaptativo se usa para proporcionar el mejor modelo lineal que represente o adapte al comportamiento o dinámica de una planta (Función de transferencia) o sistema desconocido. La planta y el filtro adaptativo poseen la misma entrada. La salida del filtro es un conjunto de coeficientes que caracterizan el sistema desconocido. Si la planta es de naturaleza dinámica (sistema o señal no estacionaria), el modelado será variante en el tiempo. La estructura de la aplicación de los filtros adaptativos para la identificación se observa en la figura 6.

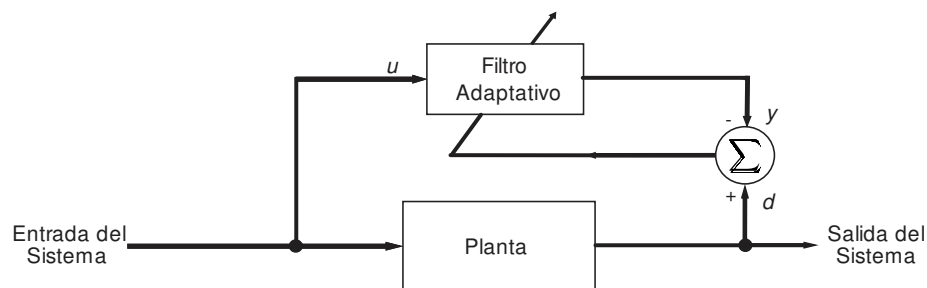


Figura 6. Estructura de un sistema adaptativo para identificación.

II. **Modelado inverso.** En esta aplicación, la función del filtro adaptativo es proporcionar el mejor modelo inverso que representa o adapta a una planta ruidosa desconocida. Idealmente, en el caso de un sistema lineal, el modelo inverso tiene una función de transferencia igual al recíproco (inverso) de la función de transferencia de la planta. Una versión retardada de la planta (sistema) es la señal de entrada al filtro adaptativo que constituye la respuesta deseada. En algunas aplicaciones, la entrada de la planta se usa sin retardo como la respuesta deseada. La estructura de la aplicación de los filtros adaptativos para la obtención de un modelo inverso se observa en la figura 7.

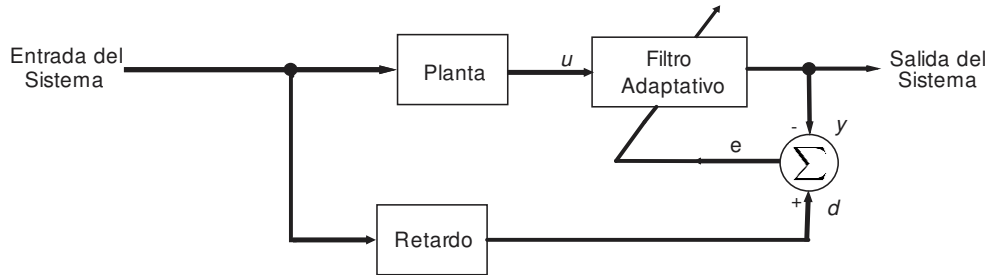


Figura 7. Estructura de un sistema adaptativo para la obtención de un modelo inverso.

III. **Predicción.** Aquí la función del filtro adaptativo es proporcionar la mejor predicción (hasta donde sea posible) del valor presente de una señal aleatoria. El valor presente de la señal es la respuesta deseada para el filtro adaptativo. Los valores pasados de la señal son aplicados a la entrada del filtro adaptativo. Dependiendo de la aplicación de interés, la salida del filtro adaptativo o la estimación (predicción) del error, podría ser la salida que se busca. En el primer caso, el sistema opera como un predictor; en el último caso, opera como un filtro de predicción-error. Estas dos posibilidades existen, gracias a que el predictor lo que ha hecho, en realidad, es separar dos señales. La estructura de la aplicación de los filtros adaptativos para la predicción se observa en la figura 8.

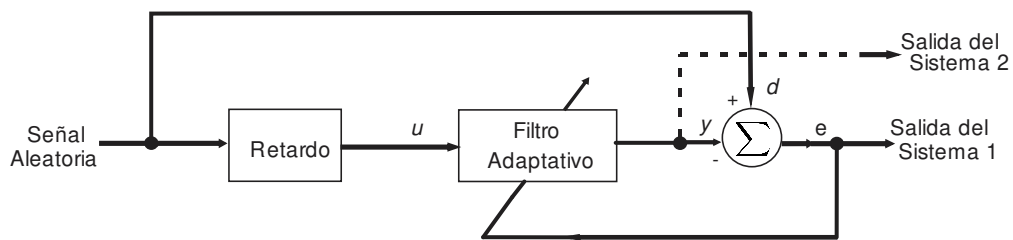


Figura 8. Estructura de un sistema adaptativo para predicción.

IV. **Cancelación de Interferencia.** En esta última aplicación, el filtro adaptativo se usa para cancelar una interferencia desconocida contenida (sumada a una señal con información de interés) en una señal primaria. Con la cancelación se comienza a optimizar la salida hasta cierto grado. La señal primaria sirve como la respuesta

deseada para el filtro adaptativo. Una señal de referencia (auxiliar) es empleada como la entrada al filtro adaptativo. La señal de referencia se deriva de un sensor o conjunto de sensores localizados muy cerca del sensor que proporciona la señal primaria, solo que la componente del ruido en la señal primaria es débil o esencialmente indetectable y la componente del ruido en la señal de referencia resulta ser una proporción de la señal interferente desconocida. La estructura de la aplicación de los filtros adaptativos para la cancelación de interferencia se observa en la figura 9.

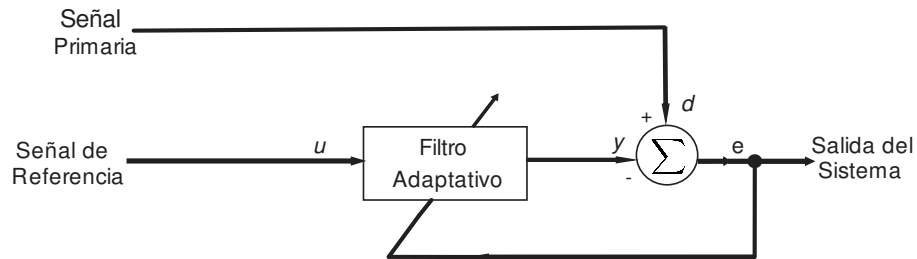


Figura 9. Estructura de un sistema adaptativo para la cancelación de interferencia.

2.2.2 Filtrado Óptimo

En la aplicación de filtrado adaptativo, las entradas en un filtro transversal, como el que se muestra en la figura 5, son muestras con diferentes retardos de la señal de entrada u . Durante el proceso de adaptación, si los coeficientes h_k dependen de los valores de las entradas u_k , el sistema dejará de ser lineal; por lo tanto, dependerán del tiempo. La ecuación 1, la cual es el resultado del filtro transversal descrito puede expresarse matricialmente como la ecuación 6.

$$y(n) = H^T[n]U[n] = U^T[n]H[n] \quad (6)$$

Donde el producto de matrices está definido en la forma habitual, $H[n]$ es un vector columna que contiene los pesos de los coeficientes del filtro adaptativo, (ver ecuación 7)

$$H^T[n] = (h_0[n], h_1[n], \dots, h_k[n]) \quad (7)$$

Y $U[n]$ es un vector que contiene muestras de la señal de entrada como se observa en la ecuación 8.

$$U^T[n] = (u[n], u[n - 1] \dots, u[n - k]) \quad (8)$$

El objetivo de los algoritmos adaptativos será minimizar la potencia media de la señal error $e[n]$. Es decir, se determinarán los coeficientes del filtro transversal de modo que la señal $y[n]$ se cancele lo más posible con $d[n]$.

Supongamos inicialmente que $u[n]$ y $d[n]$ son procesos aleatorios estacionarios. La potencia de la señal $e[n]$ depende de los coeficientes que contiene filtro adaptativo

transversal. A la función que expresa la potencia de la señal error en función de los coeficientes del filtro se le denomina Superficie de Error. Supongamos momentáneamente que el vector de pesos del filtro H no varía con el tiempo n . Por lo tanto, la señal de error se describe en la ecuación 9.

$$e(n) = d[n] - y[n] = d[n] - H^T U[n] \quad (9)$$

La potencia instantánea de la señal de error se observa en la ecuación 10.

$$(e(n))^2 = (d[n])^2 + H^T U[n] U^T [n] H - 2d[n] U^T [n] H \quad (10)$$

Tomando esperanzas matemáticas en la ecuación 10 y teniendo en cuenta que el vector de coeficientes H esta fijo (es decir no es aleatorio) resulta la ecuación 11.

$$E\{(e(n))^2\} = E\{(d[n])^2\} + H^T E\{U[n] U^T [n]\} H - 2E\{d[n] U^T [n]\} H \quad (11)$$

La matriz que se forma por la esperanza matemática de la señal de entrada en el segundo término de la ecuación 11 se denomina matriz de autocorrelación (R) y se expresa en las ecuaciones 12-14.

$$R = E\{U[n] U^T [n]\} = E \left\{ \begin{bmatrix} u[n] \\ u[n-1] \\ \vdots \\ u[n-k] \end{bmatrix} \begin{bmatrix} u[n] & u[n-1] & \cdots & u[n-k] \end{bmatrix} \right\} \quad (12)$$

$$R = E \left\{ \begin{bmatrix} u[n]u[n] & u[n]u[n-1] & \cdots & u[n]u[n-k] \\ u[n-1]u[n] & u[n-1]u[n-1] & \cdots & u[n-1]u[n-k] \\ \vdots & \vdots & \ddots & \vdots \\ u[n-k]u[n] & u[n-k]u[n-1] & \cdots & u[n-k]u[n-k] \end{bmatrix} \right\} \quad (13)$$

$$R = \begin{bmatrix} R_u[0] & R_u[1] & R_u[2] & \cdots & R_u[k] \\ R_u[1] & R_u[0] & R_u[1] & \cdots & R_u[k-1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_u[k] & R_u[k-1] & R_u[k-2] & \cdots & R_u[0] \end{bmatrix} \quad (14)$$

Análogamente, se define el vector de correlaciones cruzadas (P) entre la señal de entrada y la señal de referencia o deseada, el cual está en el tercer término de la ecuación 11 y se describe en las ecuaciones 15-16.

$$P = E\{d[n] U[n]\} = E \left\{ \begin{bmatrix} d[n] \\ d[n] \end{bmatrix} \begin{bmatrix} u[n] \\ u[n-1] \\ \vdots \\ u[n-k] \end{bmatrix} \right\} = E \left\{ \begin{bmatrix} d[n]u[n] \\ d[n]u[n-1] \\ \vdots \\ d[n]u[n-k] \end{bmatrix} \right\} \quad (15)$$

$$P = \begin{bmatrix} R_{dx}[0] \\ R_{dx}[1] \\ \vdots \\ R_{dx}[k] \end{bmatrix} \quad (16)$$

De acuerdo con las ecuaciones 14 y 16 es posible escribir la potencia media de la señal de error (ξ) en función de los coeficientes como la ecuación 17.

$$E\{(e(n))^2\} = \xi = E\{(d[n])^2\} + H^T R H - 2P^T H \quad (17)$$

La ecuación 17 expresa el valor medio de la potencia de la señal error en el caso de que los coeficientes no varíen y las señales sean estacionarias. En la práctica, y dado que el filtro es adaptativo, lo que sucederá es que un algoritmo de adaptación tratará de encontrar el valor de los coeficientes H que minimice la señal de error. Además, será suficiente con que las señales tengan una estadística que varíe suficientemente despacio como para que el algoritmo de adaptación pueda converger correctamente y seguir las variaciones estadísticas de la señal. En el caso de señales no estacionarias lo que sucederá es que la función de error irá modificándose lentamente con el tiempo.

La ecuación 17 expresa una función que tiene una forma denominada cuádrica, una cuádrica es el lugar geométrico de los puntos del espacio (x, y, z) que verifican una ecuación de segundo grado (un paraboloides), esta superficie indica cuál es la potencia media de la señal error en función de los coeficientes de un filtro FIR con un único mínimo local que coincide con el mínimo global.

Dado que la función del valor cuadrático medio de la señal error (ξ) tiene un único mínimo local, podremos hallar los coeficientes óptimos W^* , es decir los que minimizan la potencia de la señal error, sin más que calcular el gradiente de la función ξ , anularlo y despejar el valor de los pesos óptimos. Por lo tanto, se define en la ecuación 18 el gradiente de una función escalar que depende de un vector.

$$\nabla \xi = \left[\frac{\partial \xi}{\partial h_0}, \frac{\partial \xi}{\partial h_1}, \dots, \frac{\partial \xi}{\partial h_k} \right]^T \quad (18)$$

El vector gradiente en un cierto punto tiene las siguientes propiedades: la dirección es la de máxima pendiente en un punto (crecimiento), su módulo es el valor de esa máxima pendiente y los mínimos (y máximos) locales de la función se corresponden con puntos de gradiente nulo. Para calcular el gradiente, a la ecuación 17 se le calcula las derivadas parciales formando el vector gradiente. Dicho gradiente se puede expresar de nuevo en forma matricial como se observa en la ecuación 19.

$$\nabla \xi = 2RH - 2P \quad (19)$$

El error cuadrático medio mínimo se producirá cuando los pesos del filtro anulen el gradiente como se observa en la ecuación 20.

$$\nabla \xi = 0 = 2RH^* - 2P \quad (20)$$

Suponiendo que R sea no singular (es decir, invertible) entonces el vector de coeficientes óptimo, H^* queda definido por la ecuación 21.

$$H^* = R^{-1}P \quad (21)$$

Al vector de pesos obtenido por la ecuación 21 se le denomina Filtro Óptimo de Wiener, y representa el mejor conjunto de pesos posible en el sentido de minimizar la potencia de la señal error. El valor de la potencia media cuando los coeficientes del filtro son los óptimos, es decir, cuando ξ_{\min} se obtiene mediante la ecuación 22 teniendo en cuenta que la matriz R es simétrica.

$$\xi_{\min} = E\{d^2\} - P^T H^* \quad (22)$$

Otra posibilidad para encontrar los pesos óptimos consiste en realizar un algoritmo iterativo, en el que se mueve en dirección contraria al gradiente (es decir, cuesta abajo) con pasos de un tamaño proporcional al gradiente. Éste se puede escribir como la ecuación 23.

$$\nabla \xi = 2R(H - H^*) \quad (23)$$

Es decir, el gradiente es proporcional a la distancia entre el vector de pesos H y el vector de pesos de error cuadrático medio mínimo H^* . El hecho de que el gradiente sea proporcional a la diferencia entre los pesos y los pesos óptimos justifica el hecho de que los pasos de los algoritmos iterativos sean de tamaño proporcional al gradiente.

Existen procedimientos de adaptación que estiman la matriz de autocorrelación y el vector de correlaciones cruzadas muestra a muestra determinando los pesos óptimos mediante la ecuación 21 con las matrices estimadas. Estos métodos emplean técnicas eficientes que evitan el tener que invertir una matriz cada vez que llega una nueva muestra. Una de las técnicas se denominada algoritmos de gradiente, es nombrado así porque busca el vector de pesos óptimo de forma recursiva, moviéndose pequeños pasos de longitud proporcional al gradiente y en dirección contraria al mismo. El vector de pesos se actualiza mediante la ecuación 24.

$$H[n + 1] = H[n] - \mu \nabla \xi[n] \quad (24)$$

Donde el parámetro μ controla el tamaño de los pasos del algoritmo de la siguiente manera:

- Si μ aumenta, la velocidad de convergencia será más rápida. Si es excesivamente grande, el método puede hacerse inestable y divergir en lugar de converger. Además, en la práctica la solución final alcanzada es más inestable.

- Si μ disminuye, la convergencia se ralentiza, pero a cambio, la solución final es más estable.

La estrategia de variar los pesos iterativamente en dirección contraria al gradiente es el fundamento del algoritmo adaptativo LMS (Least Mean Square), el cual se detallará a continuación.

2.3 El algoritmo LMS

El algoritmo de Mínimos Cuadrados Medios (LMS, Least Mean Squares) fue creado por Widrow y Hoff en el año de 1960 y pertenece a la familia de algoritmos de gradiente estocástico. El término “gradiente estocástico” pretende distinguir el algoritmo LMS del método de máxima pendiente que usa un gradiente determinista en un cálculo recursivo de los filtros Wiener para entradas estocásticas. El algoritmo fue ideado por Widrow y demostrado por Hoff en 1959. Todo comenzó con el estudio de un esquema de reconocimiento de un patrón, conocido como elemento lineal adaptativo (lógica de umbral) comúnmente referido dentro de la literatura científica como ADALINE (del inglés: ADaptive LINear Element) [5].

Una característica muy importante y a la vez atractiva del algoritmo LMS es su simplicidad. No requiere medición de funciones de correlación, y tampoco necesita de inversión de matrices. El diagrama a bloques de la estructura del algoritmo LMS se presenta en la figura 10.

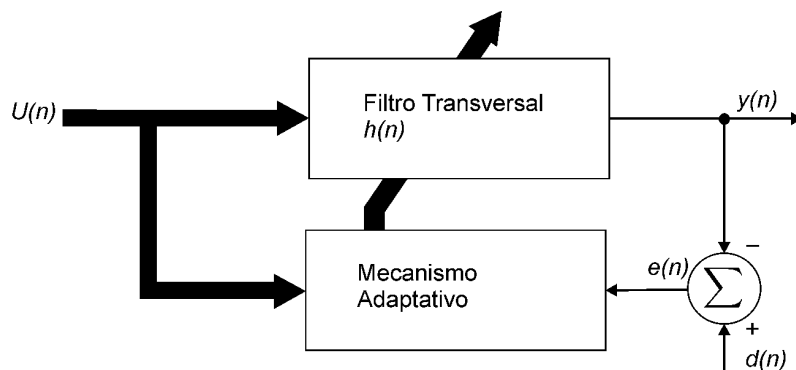


Figura 10. Diagrama del bloque de la estructura de la estructura LMS

En la figura 10 se presenta un filtro transversal, alrededor del cual se construye el algoritmo LMS, el cual es responsable de desempeñar el proceso de filtrado. Luego se tiene un mecanismo que desempeña el proceso de control adaptativo de los valores de los coeficientes del filtro transversal. Los detalles de los componentes del filtro transversal son presentados en la figura 11, donde la entrada es representada por $u(n)I, u(n-1), \dots, u(n-M+1)$ que son los elementos del vector de entrada, donde $M-1$ es el número de elementos de retardo.

Correspondientemente, los valores $h_0(n)$, $h_1(n)$, ..., $h_{M-1}(n)$ son los elementos del vector de coeficientes. El valor calculado para el vector de coeficientes $h(n)$ utilizando el algoritmo LMS representa una estimación del valor que se aproxima a la solución óptima con un número de n iteraciones.

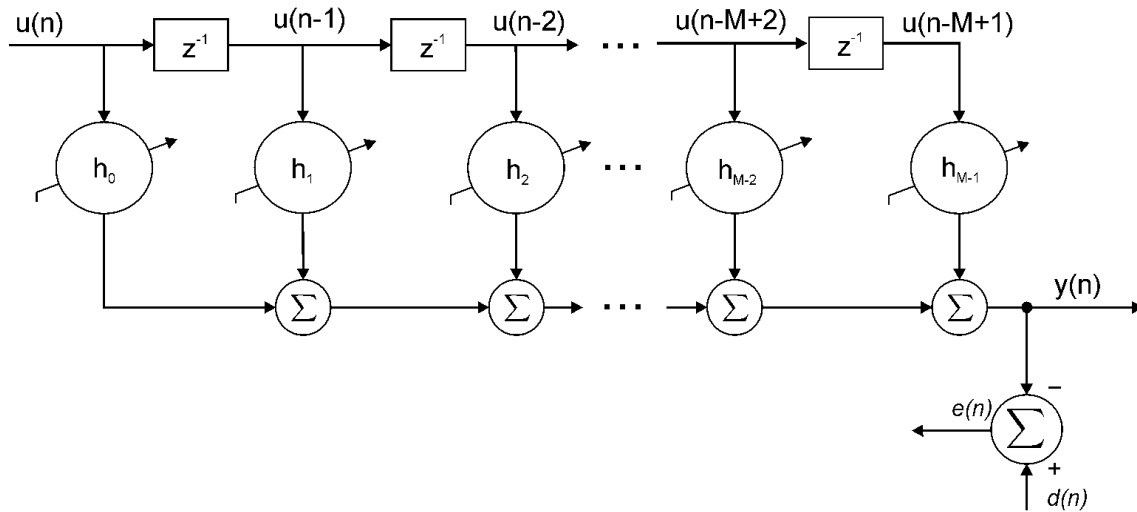


Figura 11. Diagrama del Filtro transversal

El filtro transversal produce la salida $y(n)$ utilizada como una estimación de la respuesta deseada $d(n)$. Consecuentemente se define una estimación de error $e(n)$ como la diferencia entre la respuesta deseada y la salida del filtro, como se indica la salida en la figura 11 la estimación del error $e(n)$ y los valores del vector de entrada $u(n)$ son aplicados al mecanismo de control y el lazo de realimentación alrededor de los valores de los coeficientes es cerrado consecuentemente.

La figura 12 presenta los detalles del mecanismo de control adaptativo para los coeficientes. Específicamente, una versión escalar del producto interno de la estimación del error $e(n)$ y la entrada $u(n-k)$ es calculada para $k=0, 1, 2, \dots, M-2, M-1$. El resultado así obtenido define la correlación $\delta h_k(n)$ aplicada a los valores de los coeficientes $h_k(n)$ en la iteración $n+1$. El factor de escala utilizado en este cálculo se designa por μ , mejor conocido como parámetro del tamaño de paso de adaptación o velocidad de adaptación.

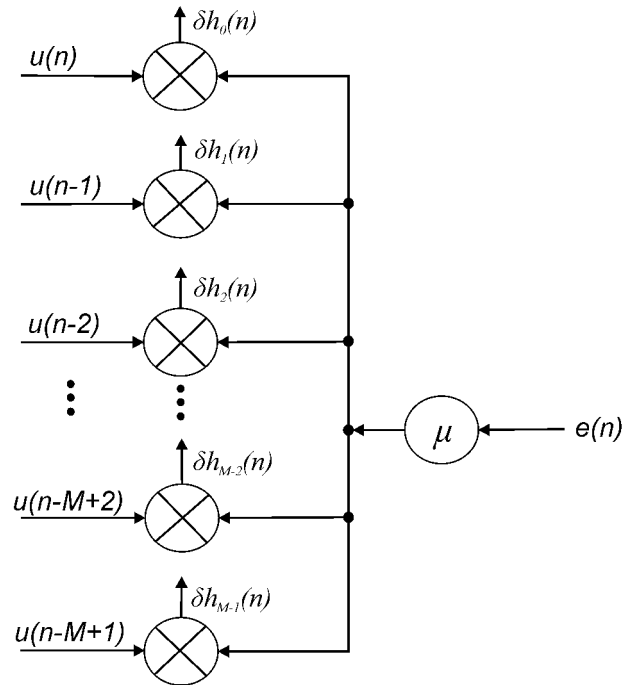


Figura 12. Estructura del mecanismo de control adaptativo para los coeficientes

Por lo tanto, el algoritmo LMS consiste en sustituir el gradiente en la ecuación 24 por una estima de este obtenida con muestras de las señales. En concreto, para actualizar los pesos, se sigue la ecuación 25 [39].

$$H[n + 1] = H[n] + \frac{1}{2} \mu e[n] U[n] \quad (25)$$

Donde la estima del gradiente está representada por la ecuación 26.

$$\nabla \xi = -\frac{1}{2} e[n] U[n] \quad (26)$$

La media de la estima del gradiente de la ecuación 26 coincide con el gradiente teórico de la ecuación 23 con un factor de dos, por lo tanto, se propone un medio en la ecuación 26 para facilitar los cálculos y quede multiplicado por la unidad. Esta comprobación se presenta en la ecuación 27 .

$$E\{\nabla \xi[n]\} = E\left\{-\frac{1}{2} e[n] U[n]\right\} = E\left\{-\frac{1}{2} (d[n] - U^T[n]H)U[n]\right\} = \frac{1}{2} (RH - P) \quad (27)$$

Tomando en cuenta las ecuaciones 6, 9, 27 y 25 se obtienen las tres ecuaciones que rigen el comportamiento del algoritmo LMS, estas relaciones básicas se presentan en las ecuaciones 28-30.

Salida del Filtro:

$$y(n) = U^T[n]H[n] \quad (28)$$

Estimación de error:

$$e(n) = d[n] - y[n] \quad (29)$$

Adaptación de los valores de los coeficientes:

$$H[n + 1] = H[n] + \mu e[n]U[n] \quad (30)$$

Donde μ es el tamaño de paso fijo, ya que, en el proceso de filtrado adaptativo utilizar un algoritmo de tamaño de paso fijo es adecuado debido a dos razones. La primera de ellas es que un algoritmo con tamaño de paso fijo se implementa fácilmente por hardware o software y la segunda es que un tamaño de paso fijo es apropiado para hacer un seguimiento de los parámetros estadísticos de la señal variante en el tiempo.

La principal limitación del algoritmo LMS es su lenta velocidad de convergencia. Hay dos factores principales que inciden en el comportamiento del algoritmo LMS: el parámetro de tamaño de paso μ y el valor propio de la matriz de correlación R del vector de entrada. Dentro de la perspectiva del análisis del algoritmo LMS se resumen sus características a continuación.

- La convergencia del algoritmo LMS dentro del cuadrado medio de la entrada es asegurada por la selección del parámetro de tamaño de paso μ de acuerdo con la condición práctica: $0 < \mu < \frac{2}{\text{potencia de entrada}}$ donde la potencia de entrada es la suma de los valores del cuadrado medio de todos los valores de entrada dentro del filtro transversal.
- Cuando un valor pequeño se asigna a μ , la adaptación es lenta, ya que es equivalente a tener un algoritmo LMS con bastante “memoria”. Respectivamente, el exceso del error cuadrático medio después de la adaptación es pequeño en promedio, por la gran cantidad de datos utilizados por el algoritmo para la estimación del gradiente. En otras palabras, cuando μ es grande, la adaptación es relativamente rápida, pero hay un costo de incremento dentro del promedio del exceso del error cuadrático medio después de la adaptación. En este caso, menos datos entran para calcular la estimación, dentro de un desempeño de estimación de error degradado. Así, el recíproco del parámetro μ debe verse como memoria del algoritmo LMS.
- Cuando el valor propio de la matriz de correlación R presenta una dispersión grande, el exceso del error cuadrático medio producido por el algoritmo LMS es primeramente determinado por valores propios grandes y a la vez tomado por el promedio de los coeficientes, para que ocurra la convergencia se limita a un valor propio pequeño. Sin embargo, la velocidad de convergencia del error cuadrático medio es afectada por una dispersión de los valores propios de R de menor magnitud que la convergencia de los coeficientes. Cuando la dispersión de los valores propios es grande la convergencia del algoritmo LMS es de baja velocidad.

2.4 El algoritmo RLS

El algoritmo adaptativo RLS tiene como característica fundamental actualizar los coeficientes del filtro en cada iteración minimizando la suma del error cuadrático y se basa en la inversión de matrices. En la forma recursiva del método de los cuadrados (RLS), se inicia el cálculo con las condiciones iniciales conocidas y se usa la información contenida en las nuevas muestras de datos para actualizar las estimaciones anteriores, por lo tanto, se encuentra que la longitud de los datos observables es variable. El diagrama a bloques del algoritmo RLS se puede observar en la figura 13.

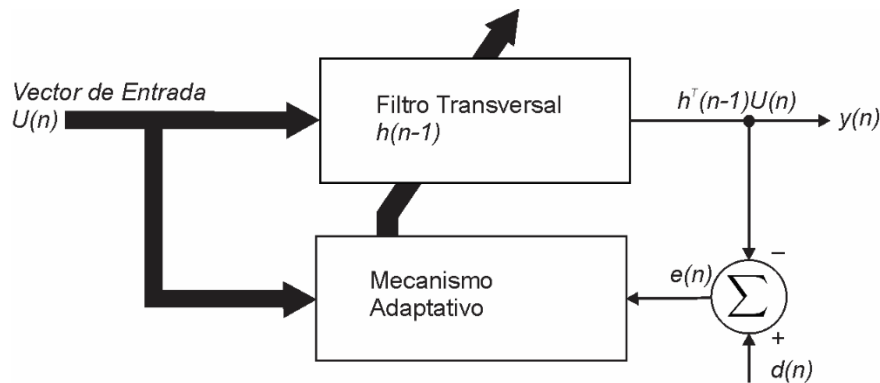


Figura 13. Diagrama a bloques del algoritmo RLS

La función de costo que debe minimizarse para lograr obtener los coeficientes del filtro adaptativo mediante el algoritmo RLS es \mathcal{E} y queda representada en la ecuación 31, donde n es la capacidad de los datos observables. Además, se introduce un factor de ponderación (λ , factor de olvido) en la definición de la función de costo, el uso del factor λ está destinado a asegurar que los datos en el pasado distante sean olvidados para permitir la posibilidad de seguir las variaciones estáticas de los datos observables cuando el filtro opera en un entorno no estacionario.

$$\mathcal{E} = \sum_{l=1}^n \lambda^{n-l} |e_M[l]|^2 \quad (31)$$

En la ecuación 31 λ es un valor cercano a 1, pero menor a la unidad, ya que cuando $\lambda=1$ se tiene el ordinario método de mínimos cuadrados. El error ($e_M[n]$) se define como la diferencia entre la secuencia deseada $d[n]$ y la estimada ($H_M[n-1]U[n]$).

Tomando en cuenta las ecuaciones 7 y 8 presentadas anteriormente y considerando que $H_M[n] = H[n]$, el algoritmo RLS preselecciona los datos observados de entrada ($U[n]$) y trata de determinar el vector de coeficientes del filtro $H_M[n]$ que minimiza la suma ponderada de los módulos al cuadrado de los errores (ecuación 31). La minimización de \mathcal{E} con respecto

al vector de coeficientes del filtro $H[n]$ proporciona el conjunto de ecuaciones lineales presentadas en la ecuación 32.

$$R_M[n]H_M[n] = D_M[n] \quad (32)$$

Donde $R_M[n]$ es la matriz de correlación de la señal, queda definida por la ecuación 33.

$$R_M[n] = \sum_{l=1}^n \lambda^{n-l} U[n]U^T[n] \quad (33)$$

Y $P_M[n]$ es el vector de correlación cruzada representado en la ecuación 34

$$D_M[n] = \sum_{l=1}^n \lambda^{n-l} U[n]d[n] \quad (34)$$

La solución para determinar los coeficientes del filtro de la ecuación 32 se muestra en la ecuación 35

$$H_M[n] = R_M^{-1}[n]D_M[n] \quad (35)$$

Evidentemente, la matriz $R_M[n]$ es parecida a la matriz de autocorrelación presentada en la ecuación 12, y el vector $P_M[n]$ es parecido al vector de correlación cruzada presentado en la ecuación 15. Sin embargo, hay que destacar que $R_M[n]$ no es una matriz Toeplitz. También cabe mencionar que, para valores pequeños de n , $R_M[n]$ puede estar mal condicionada, por lo que su inverso no se puede calcular. En tal caso, es habitual añadir inicialmente la matriz $\delta^{-1}I_M$ a $R_M[n]$, donde I_M es una matriz identidad y δ es una constante positiva pequeña. Con una ponderación exponencial hacia atrás, el efecto de añadir $\delta^{-1}I_M$ se disipa con el tiempo [5].

El cálculo de la matriz y los vectores se hacen de forma recursiva para hacer más eficiente el algoritmo, en primer lugar, $R_M[n]$ puede calcularse recursivamente con la ecuación 36.

$$R_M[n] = \lambda R_M[n-1] + U[n]U^T[n] \quad (36)$$

Puesto que se necesita en inverso de $R_M[n]$, se utiliza el lema de inversión de matrices, obteniendo la ecuación 37.

$$R_M^{-1}[n] = \frac{1}{\lambda} \left(R_M^{-1}[n-1] - \frac{R_M^{-1}[n-1]U[n]U^T[n]R_M^{-1}[n-1]}{\lambda + U^T[n]R_M^{-1}[n-1]U[n]} \right) \quad (37)$$

Por conveniencia, definimos $P_M[n] = R_M^{-1}[n]$. Resultando la ecuación 38.

$$P_M[n] = \frac{1}{\lambda} \left(P_M[n-1] - \frac{P_M[n-1]U[n]U^T[n]P_M[n-1]}{\lambda + U^T[n]P_M[n-1]U[n]} \right) \quad (38)$$

Así como $R_M[n]$, $D_M[n]$ puede también calcularse recursivamente como se ve en la ecuación 39

$$D_M[n] = \lambda D_M[n-1] + d[n]U[n] \quad (39)$$

Tomando en cuenta las ecuaciones 35, 38 y 39 se obtiene una ecuación para calcular los coeficientes del filtro en forma recursiva, la cual se presenta en la ecuación 40.

$$H_M[n] = H_M[n-1] + P_M[n]U[n](d[n] - U^T[n]H_M[n-1]) \quad (40)$$

Obsérvese que $U^T[n]H_M[n-1]$ es la salida del filtro adaptativo en el instante n basada en el uso de los coeficientes el filtro en el instante $n-1$. Puesto que

$$e_M[n] = d[n] - U^T[n]H_M[n-1] \quad (41)$$

Se puede sustituir en la ecuación 40 obteniendo la ecuación 42.

$$H_M[n] = H_M[n-1] + P_M[n]U[n]e_M[n] \quad (42)$$

Por lo tanto, el cálculo recursivo para los coeficientes del filtro es el que se muestra a continuación:

- 1.- Calcular la salida del filtro $U^T[n]H_M[n-1]$.
- 2.- Calcular el error $e_M[n]$. Ecuación 41
- 3.- Calcular y actualizar la inversión de la matriz de correlación $P_M[n]$. Ecuación 38
- 4.- Actualizar el vector de coeficientes del filtro $H_M[n]$. Ecuación 42

Una ventaja importante del algoritmo RLS en la forma directa sobre el algoritmo LMS es su rápida velocidad de convergencia. A pesar de su superior velocidad de convergencia, los algoritmos RLS para el filtrado adaptativo FIR descritos en la sección anterior presentan dos importantes desventajas. Una es su complejidad de cálculo y la segunda desventaja de los algoritmos es su sensibilidad a los errores de redondeo que acumulan como resultado de los cálculos recursivos. En algunos casos, los errores de redondeo hacen que estos algoritmos sean inestables. El algoritmo RLS en la forma directa se hace inestable y, por tanto, no funciona apropiadamente con aritmética de punto fijo de 16 bits. Para este algoritmo, se determinó experimentalmente que son necesarios aproximadamente de 20 a 24 bits de precisión para que el algoritmo funcione bien [38].

2.5 El Filtro Kalman

En 1960, R. E. Kalman publicó su trabajo que describía una solución para el problema de filtrado lineal en datos discretos [Kalman60-Welch-Bishop]. El filtro Kalman ha sido ha sido objeto de muchas investigaciones y aplicaciones, particularmente en el área de autonomía o navegación asistida.

El filtro de Kalman es un conjunto de ecuaciones matemáticas que proporciona una solución computacional eficiente (recursiva) del método de mínimos cuadrados. El filtro es muy poderoso en varios aspectos: admite estimaciones de pasado, presente e incluso futuro estados, y puede hacerlo incluso cuando se desconoce la naturaleza precisa del sistema modelado. La dinámica que rige el algoritmo se resume en dos pasos:

1. Estimar las variables de estado utilizando su propia dinámica (etapa de predicción).
2. Mejorar la primera estimación utilizando la información de las variables observables (etapa de corrección).

Una característica muy atractiva del filtro de Kalman es su carácter recursivo. Una vez que el algoritmo pronostica el nuevo estado en un momento determinado, añade un término de corrección y el nuevo estado “corregido” sirve como condición inicial en la siguiente etapa. De esta forma, la estimación de la variable de estado utiliza toda la información disponible hasta el momento y no solo la información hasta la etapa anterior al momento en el que se realiza la estimación. En la figura 14 se muestra el diagrama a bloques de la formulación matemática en términos del concepto espacio del estado del filtro Kalman.

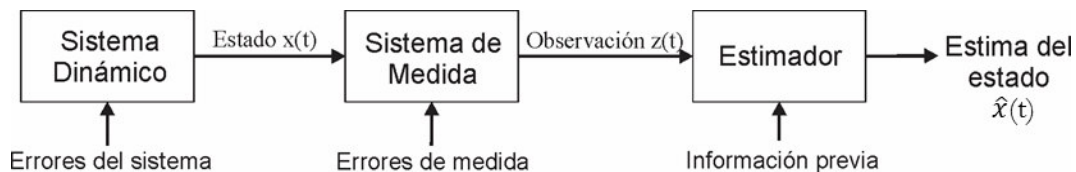


Figura 14. Diagrama a bloques de la formulación matemática del concepto espacio del estado

El filtro de Kalman aborda el problema general de tratar de estimar el estado $x \in R^n$ de un proceso controlado de tiempo discreto que se rige por la ecuación de diferencia estocástica lineal presentada en la ecuación 43 llamada ecuación del proceso, a partir de un conjunto de medidas $z \in R^m$ representado por la ecuación 44 llamada ecuación de medida.

$$x_{k+1} = A_k x_k + B u_k + w_k \quad (43)$$

$$z_k = H_k x_k + v_k \quad (44)$$

Las variables aleatorias w_k y v_k representan el ruido de proceso y el ruido de medición (respectivamente). Se supone que son independientes (entre sí), son ruidos blancos y con distribuciones de probabilidad normales (gaussiana).

La matriz A de $n \times n$ en la ecuación de diferencia 43 relaciona el estado en el paso de tiempo k con el estado en el paso $k + 1$, en ausencia de una función de accionamiento o ruido de proceso. La matriz B de $n \times l$ relaciona la entrada de control $u \in R^l$ con el estado x . La matriz H de $m \times n$ en la ecuación de medición 44 relaciona el estado con la medición z_k .

2.5.1 El origen computacional del filtro Kalman

Se definirá $\hat{x}_k^- \in R^n$ (nótese el "súper menos") como la estimación *a priori* del estado en el paso k dado el conocimiento del proceso anterior al paso k , y $\hat{x}_k \in R^n$ para ser la estimación *a posteriori* del estado en el paso k de medición dada por z_k . Por lo tanto, se pueden definir los errores de estimación *a priori* y *a posteriori* en las ecuaciones 45 y 46 respectivamente.

$$e_k^- = x_k - \hat{x}_k^- \quad (45)$$

$$e_k = x_k - \hat{x}_k \quad (46)$$

La covarianza de error de estimación *a priori* queda representada por la ecuación 47, mientras que la covarianza de error de estimación *a posteriori* está representada en la ecuación 48.

$$P_k^- = E[e_k^- e_k^{-T}] \quad (47)$$

$$P_k = E[e_k e_k^T] \quad (48)$$

Al derivar las ecuaciones para el filtro de Kalman, el objetivo es encontrar una ecuación que calcule una estimación de estado \hat{x}_k *a posteriori* como una combinación lineal de una estimación \hat{x}_k^- *a priori* y una diferencia ponderada entre una medición real z_k y una predicción de medición $H_k \hat{x}_k^-$. La ecuación que evalúa lo anteriormente dicho se presenta en la ecuación 49.

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H_k \hat{x}_k^-) \quad (49)$$

La diferencia $z_k - H_k \hat{x}_k^-$ en la ecuación 49 se llama innovación de medición, o el residuo. El residual refleja la discrepancia entre la medida predicha y la medida real.

La matriz K de $n \times m$ en la ecuación 49 es mejor conocida como la ganancia de Kalman, y es el factor de ganancia o combinación que minimiza la covarianza de error *a posteriori* en la ecuación 48 [40-42] Una de las formas que resulta al minimizar K esta dado por la ecuación 50 [40], cabe mencionar que todas las ecuaciones del filtro Kalman pueden manipularse algebraicamente en varias formas, dando como resultado otras representaciones matemáticas del mismo.

$$K_k = \frac{P_k^- H_k^T}{H_k P_k^- H_k^T + R_k} \quad (50)$$

La ecuación 50 determina con respecto a K que a medida que la covarianza del error de medición R_k se aproxima a cero, la medición real z_k es "confiable", mientras que la medida prevista $H_k \hat{x}_k^-$ es cada vez menos confiable. Por otro lado, como la covarianza de error de estimación a priori P_k^- se aproxima a cero, cada vez se confía menos en la medición real z_k , mientras que la medida prevista $H_k \hat{x}_k^-$ es cada vez más confiable.

2.5.2 El algoritmo del filtro Kalman discreto

El filtro de Kalman estima un proceso mediante el uso de una forma de control de retroalimentación: el filtro estima el estado del proceso en algún momento y luego obtiene retroalimentación en forma de mediciones (ruidosas). Como tal, las ecuaciones para el filtro de Kalman se dividen en dos grupos: ecuaciones de actualización de tiempo y ecuaciones de actualización de medición. Las ecuaciones de actualización de tiempo son responsables de proyectar hacia adelante (en el tiempo) las estimaciones de covarianza de estado actual y error para obtener las estimaciones *a priori* para el siguiente paso de tiempo. Las ecuaciones de actualización de medición son responsables de la retroalimentación, es decir, para incorporar una nueva medida en la estimación *a priori* para obtener una estimación *a posteriori* mejorada. Las ecuaciones de actualización de tiempo también pueden considerarse ecuaciones de predicción, mientras que las ecuaciones de actualización de medición pueden considerarse como ecuaciones de corrección.

El funcionamiento del filtro Kalman está representado en la figura 15, donde A_k es una matriz de $n \times n$ llamada matriz de transición, z_k es el vector de medidas tomadas en un instante, H_k es la llamada matriz de medidas de $m \times n$, K es la ganancia de Kalman, la matriz Q_k se usa para representar la incertidumbre en el modelo de proceso (ruido del proceso) y R_k representa la matriz de covarianza de error de medición.

Las ecuaciones de actualización de tiempo proyectan las estimaciones de estado y covarianza desde el paso de tiempo k hasta el paso de $k + 1$. La primera tarea durante la actualización de la medición es calcular la ganancia de Kalman. El siguiente paso es medir realmente el proceso para obtener y luego generar una estimación de estado *a posteriori* incorporando la medición z_k . El último paso es obtener una estimación de covarianza de error *a posteriori*. Después de cada par de actualización de tiempo y medición, el proceso se repite con las estimaciones previas *a posteriori* utilizadas para proyectar o predecir las nuevas estimaciones *a priori*. El filtro de Kalman condiciona recursivamente la estimación actual en todas las mediciones pasadas. En la implementación real del filtro, cada una de las matrices de covarianza de error de medición y el ruido del proceso pueden medirse antes de la operación del filtro.

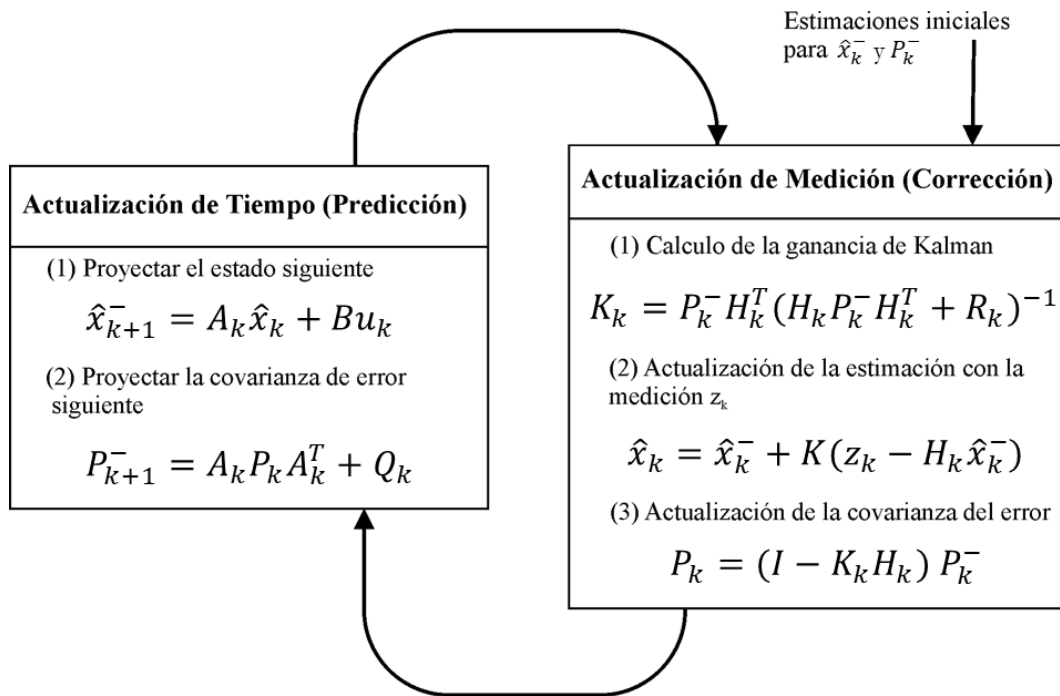


Figura 15. Diagrama de operación completa del filtro Kalman

2.6 Identificación de Sistemas

La identificación de sistemas es la teoría y el arte de construir modelos matemáticos de sistemas dinámicos basándonos en las entradas y salidas observadas, por lo cual construir modelos para sistemas desconocidos es un objetivo importante de la Ingeniería de control. Estos modelos necesitan simular el comportamiento real en los casos en que existe un conocimiento previo limitado de la estructura del sistema. Puede decirse que la identificación de sistemas quedó establecida como un campo de investigación reconocido dentro del área de control automático a mediados de los sesenta: en el tercer congreso de la IFAC en Londres, 1966 en el que fue presentado un artículo de visión general sobre identificación de sistemas [43]. La teoría sobre identificación de sistemas está reunida, por ejemplo, en los libros de Ljung [44] y de Söderstrom y Stoica [45], donde también se pueden encontrar numerosas referencias. El término Identificación de sistemas fue acuñado por Lofti Zadeh [46] en 1962, como:

“Identificación es la determinación, en base a la entrada y la salida, de un sistema, dentro de una clase de sistemas especificada, al cual el sistema probado es equivalente.”

Existen dos métodos principales para obtener el modelo de un sistema:

1. Modelado teórico. Se trata de un método analítico, en el que se recurre a leyes básicas de la física para describir el comportamiento dinámico de un fenómeno o proceso.

2. Identificación del sistema. Se trata de un método experimental que permite obtener el modelo de un sistema a partir de datos reales recogidos de la planta bajo estudio.

El modelado teórico tiene un campo de aplicación restringido a procesos muy sencillos de modelar, o a aplicaciones en que no se requiera gran exactitud en el modelo obtenido. En muchos casos, además, la estructura del modelo obtenido a partir del conocimiento físico de la planta posee un conjunto de parámetros desconocidos y que sólo se pueden determinar experimentando sobre el sistema real. De ahí la necesidad de recurrir a los métodos de identificación de sistemas. Los modelos obtenidos mediante técnicas de identificación tienen, sin embargo, las siguientes desventajas:

1. Su rango de validez suele ser limitado (sólo son aplicables a un determinado punto de trabajo, un determinado tipo de entrada o un proceso concreto).

2. En muchos casos es difícil dar significado físico al modelo obtenido, puesto que los parámetros identificados no tienen relación directa con ninguna magnitud física. Estos parámetros se utilizan sólo para dar una descripción aceptable del comportamiento conjunto del sistema.

En la práctica, lo ideal es recurrir a una mezcla de ambos métodos de modelado para obtener el modelo final. El uso de datos reales para identificar los parámetros del modelo provee a éste de una gran exactitud, pero el proceso de identificación se ve tanto más facilitado cuanto mayor sea el conocimiento sobre las leyes físicas que rigen el proceso.

En términos generales, el proceso de identificación comprende los siguientes pasos:

1. Obtención de datos de entrada - salida. Para ello se debe excitar el sistema mediante la aplicación de una señal de entrada y registrar la evolución de sus entradas y salidas durante un intervalo de tiempo.

2. Tratamiento previo de los datos registrados. Los datos registrados están generalmente acompañados de ruidos indeseados u otro tipo de imperfecciones que puede ser necesario corregir antes de iniciar la identificación del modelo. Se trata, por tanto, de 'preparar' los datos para facilitar y mejorar el proceso de identificación.

3. Elección de la estructura del modelo. Si el modelo que se desea obtener es un modelo paramétrico (modelos que describen las relaciones entre las variables del sistema mediante expresiones matemáticas como pueden ser ecuaciones diferenciales o en diferencias), el primer paso es determinar la estructura deseada para dicho modelo. Este punto se facilita en gran medida si se tiene un cierto conocimiento sobre las leyes físicas que rigen el proceso.

4. Obtención de los parámetros del modelo. A continuación, se procede a la estimación de los parámetros de la estructura que mejor ajustan la respuesta del modelo a los datos de entrada-salida obtenidos experimentalmente.

5. Validación del modelo. El último paso consiste en determinar si el modelo obtenido satisface el grado de exactitud requerido para la aplicación en cuestión. Si se llega a la conclusión de que el modelo no es válido, se deben revisar los siguientes aspectos como posibles causas:

a) El conjunto de datos de entrada-salida no proporciona suficiente información sobre la dinámica del sistema.

b) La estructura escogida no es capaz de proporcionar una buena descripción del modelo.

c) El criterio de ajuste de parámetros seleccionado no es el más adecuado. Dependiendo de la causa estimada, deberá repetirse el proceso de identificación desde el punto correspondiente.

Por tanto, el proceso de identificación es un proceso iterativo, cuyos pasos pueden observarse en el organigrama de la figura 16.

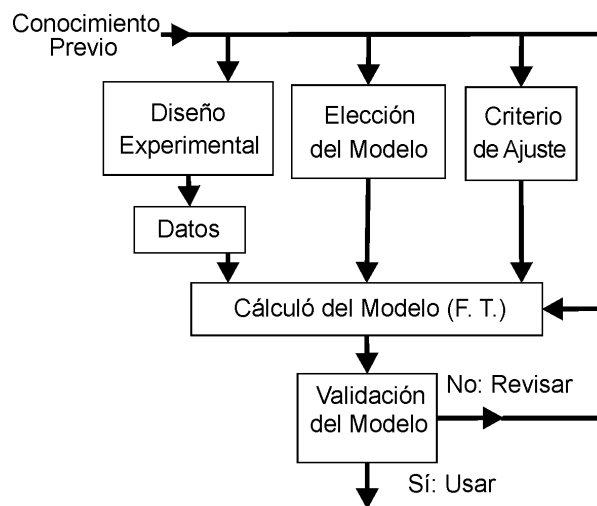


Figura 16.- Proceso de Identificación

Una de la clasificación de los métodos de identificación puede determinarse según distintos criterios:

- Dependiendo del tipo de modelo obtenido:

1. Métodos no paramétricos, que permiten obtener modelos no paramétricos del sistema bajo estudio. Algunos de estos métodos son: análisis de la respuesta transitoria, análisis de la respuesta en frecuencia, análisis de la correlación, análisis espectral, análisis de Fourier, etc.

2. Métodos paramétricos, que permiten obtener modelos paramétricos. Estos métodos requieren la elección de una posible estructura del modelo, de un criterio de ajuste de parámetros, y por último de la estimación de los parámetros que mejor ajustan el modelo a los datos experimentales.

- Dependiendo de la aplicación:

1. Métodos de identificación off-line (a posteriori), utilizados en aquellas aplicaciones en que no se requiera un ajuste continuado del modelo. En estos casos, se realiza la identificación previa de la planta, considerándose que la validez de los parámetros obtenidos no se verá alterada con el paso del tiempo.

2. Métodos de identificación on-line (identificación recursiva), en los que los parámetros se van actualizando continuamente a partir de los nuevos datos de entrada-salida obtenidos durante la evolución del proceso. Estos métodos son muy utilizados en sistemas de control adaptativo.

En el capítulo 3 se mostrará la simulación, desarrollo e implementación de los algoritmos presentados en el presente capítulo. El esquema que se utilizará en la identificación de los sistemas de control se muestra en la figura 17, donde una señal conocida (u) estimula la planta para conocer la respuesta del sistema (d) y a partir de estas dos señales se estiman los coeficientes (Θ) de los algoritmos. Como se puede ver, el algoritmo adaptativo estimará los coeficientes de la planta que caracterizarán la FT a partir de la señal u y la señal de error (ϵ) que es estimado a partir de la diferencia de la señal d y la respuesta (y) del algoritmo.

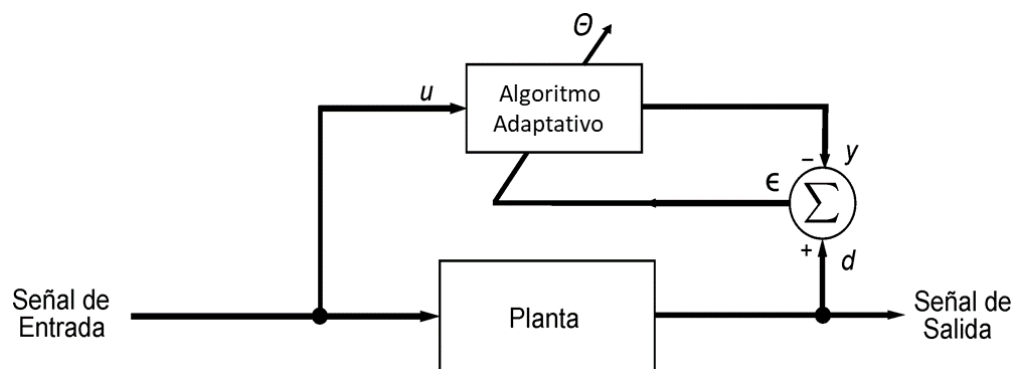


Figura 17. Esquema para la identificación de sistemas

3

CAPÍTULO

Metodología.

El proyecto de investigación presentado en este trabajo de tesis consiste en el desarrollo e implementación de algoritmos adaptativos con el fin de identificar la planta o dinámica de un sistema de control. Entre los algoritmos utilizados en este proyecto fueron el algoritmo LMS, RLS y el filtro Kalman. El diagrama de bloques que representa la evolución de dicho proyecto se presenta en la figura 18.

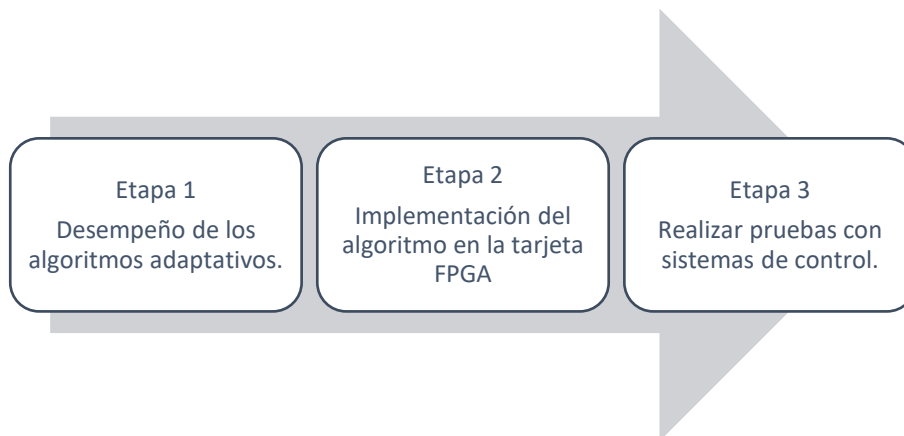


Figura 18.- Desarrollo del proyecto de investigación

En la etapa 1 se desarrollaron en la plataforma de Matlab los algoritmos adaptativos LMS, RLS y el filtro Kalman, con el objetivo de evaluar el desempeño de cada uno de ellos en la identificación de sistemas de control simulados. Posteriormente, en la etapa 2 se implementó el algoritmo adaptativo con mejor desempeño y con menor consumo computacional en una tarjeta de diseño FPGA. Por último, en la etapa 3 se realizaron las pruebas para probar el diseño del algoritmo y su desempeño en la identificación de sistemas de control. A continuación, se detallarán cada una de las etapas descritas anteriormente.

3.1 Desempeño de los algoritmos adaptativos

La primera etapa del desarrollo del proyecto es analizar los algoritmos adaptativos presentados y observar el desempeño que estos presentan en la identificación de distintos sistemas o plantas. Por lo cual, mediante el software de Matlab se diseñarán sistemas simulados, con el objetivo de probar cada uno de los algoritmos adaptativos propuestos para identificarlos.

El primer sistema que se identificó mediante los algoritmos adaptativos fue un filtro FIR de 6 coeficientes. Los coeficientes del filtro se generaron aleatoriamente y la señal de entrada del sistema también se generó con datos aleatorios. La ecuación que rige el sistema esta descrita en la ecuación 2 de la sección 2. La cual queda representada en la ecuación 51.

$$y(n) = 0.014 u(n) + 0.0974 u(n - 1) + 0.2634 u(n - 2) + 0.2094 u(n - 3) + 0.2613 u(n - 4) + 0.1545 u(n - 5) \quad (51)$$

Donde 0.014, 0.0974, 0.2634, 0.2094, 0.2613 y 0.1545 son los coeficientes que el algoritmo adaptativo debe identificar, $u(n)$ representa la señal de entrada la cual es una señal aleatoria, la cual se definió de una longitud de 120 datos y $y(n)$ representa la señal de salida del sistema.

3.1.1. Algoritmo LMS

Al aplicar el algoritmo LMS se definieron los siguientes parámetros: el número de coeficientes a identificar (6), el valor de μ (0.2) que define el paso con el cual se adaptará el algoritmo adaptativo, la señal de entrada $u(n)$ y la salida del sistema $y(n)$.

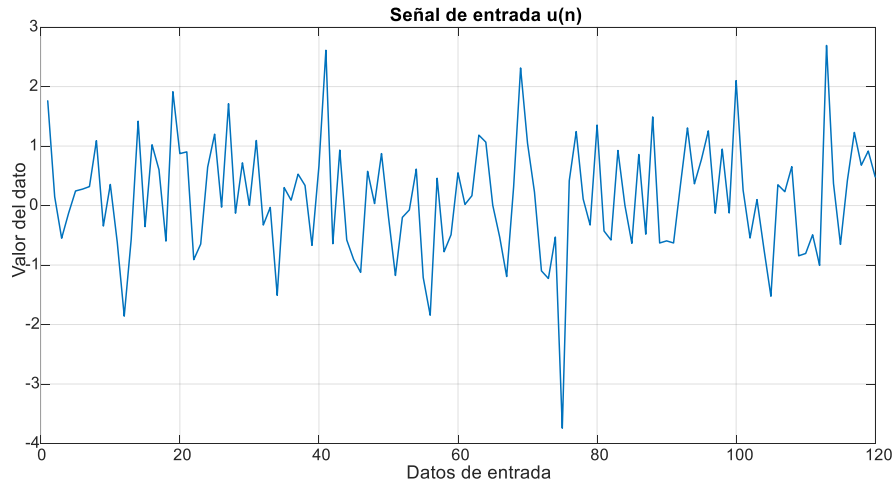


Figura 19.- Señal de entrada para la excitación del filtro FIR de 6 coeficientes.

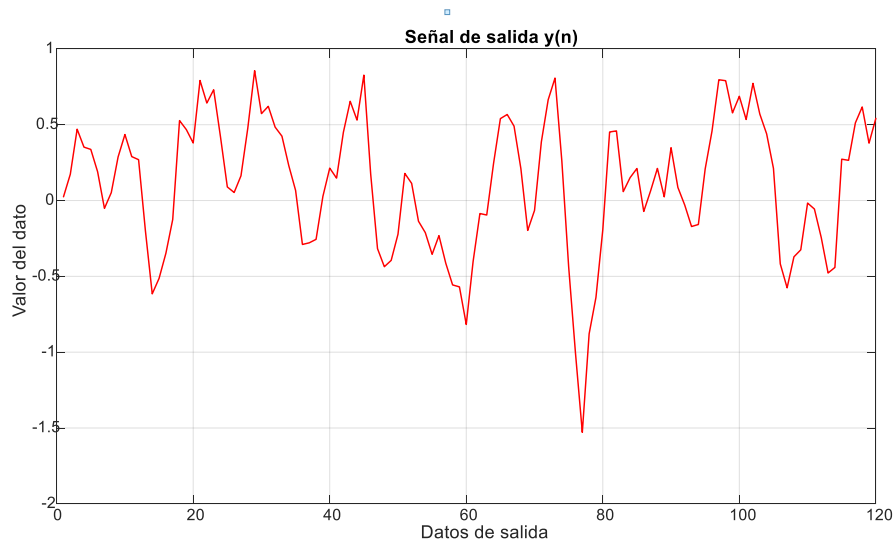


Figura 20.- Señal de salida del filtro FIR de 6 coeficientes.

El resultado que proporciona el algoritmo adaptativo al ingresar los parámetros necesarios y las señales de entrada y salida del sistema a identificar, son los siguientes coeficientes: 0.0139, 0.0974, 0.2634, 0.2094, 0.2613 y 0.1545. La adaptación de los coeficientes se presenta en la figura 21.

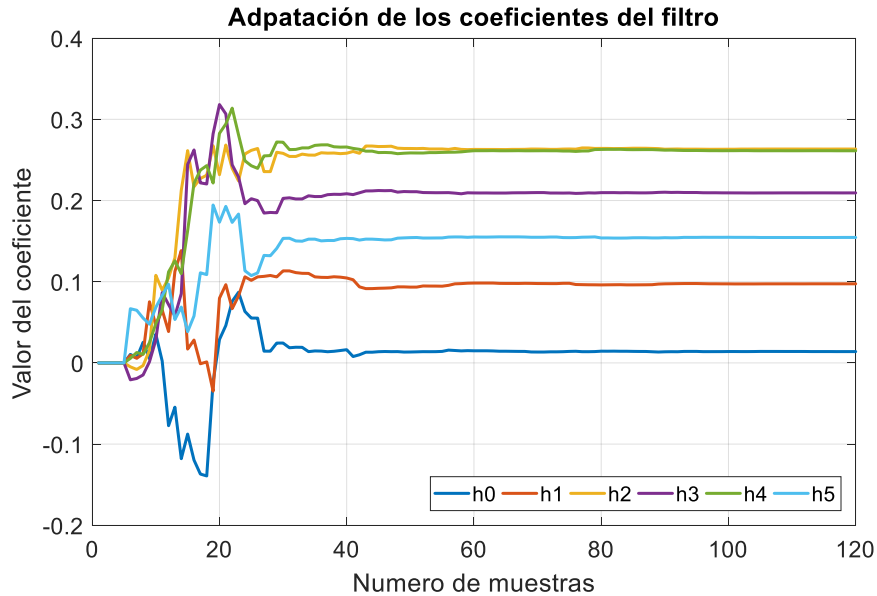


Figura 21.- Adaptación de los coeficientes mediante LMS.

En la figura 21 podemos observar que cada uno de los coeficientes son inicializados en cero y van adaptándose al valor real de cada uno mediante el algoritmo LMS. Además, se puede percibir que a partir de la muestra 40 de los datos tomados ya hay poca variación en cuanto al valor final de cada uno de los coeficientes. La comparativa entre la señal de salida obtenida con respecto a la que se va obteniendo en cada uno de los pasos del algoritmo LMS se presenta en la figura 22.

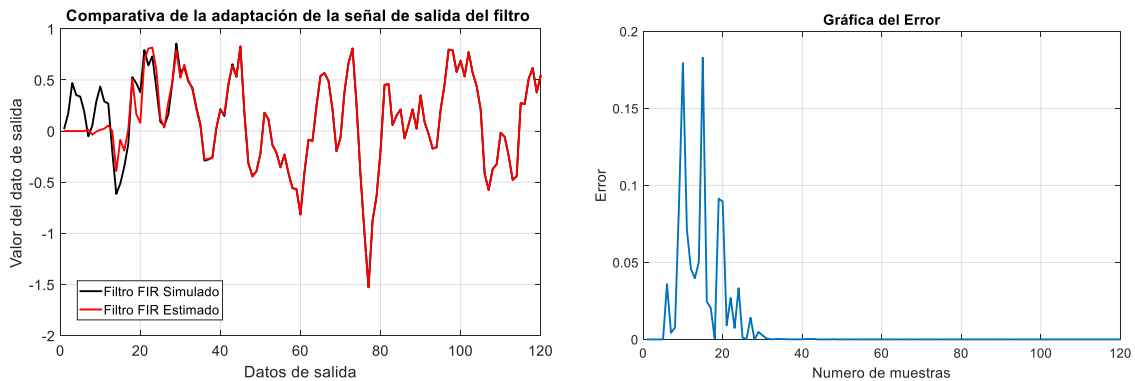


Figura 22.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante LMS, b) Gráfica de Error entre la señal de salida del filtro FIR simulado y el estimado al usar LMS.

En la figura 22(a) se puede observar como el algoritmo LMS intenta obtener la misma salida que el filtro FIR, por lo cual modifica los valores de los coeficientes estimados hasta lograrlo. En dicha figura se puede observar como a partir de la muestra 40, la señal de salida del filtro estimado es similar a la que se ha simulado, concordando con la variación que tienen los coeficientes en dicho punto.

El error generado en el transcurso de la adaptación de los coeficientes entre la señal de salida del filtro estimado con respecto al filtro simulado se presenta en la figura 22(b). En la cual se puede notar como el error disminuye al transcurso de los pasos que el algoritmo LMS llevó a cabo.

A través de los resultados podemos observar que el algoritmo LMS trabaja de manera satisfactoria para identificar sistemas como filtros FIR.

3.1.2. Algoritmo RLS

Al utilizar el algoritmo RLS para identificar el filtro FIR de seis coeficientes descrito en la sección 4.1, se definieron los siguientes parámetros: el número de coeficientes a identificar (6), el valor de λ (0.998) que define el paso con el cual se adaptará el algoritmo adaptativo, la señal de entrada $u(n)$ y la salida del sistema $y(n)$, estas señales pueden observarse en las figuras 19 y 20 respectivamente.

El resultado que proporciona el algoritmo adaptativo RLS al ingresar los parámetros necesarios y las señales de entrada y salida del sistema a identificar, son los siguientes coeficientes: 0.0140, 0.0974, 0.2634, 0.2094, 0.2613 y 0.1545. La adaptación de los coeficientes se presenta en la figura 23.

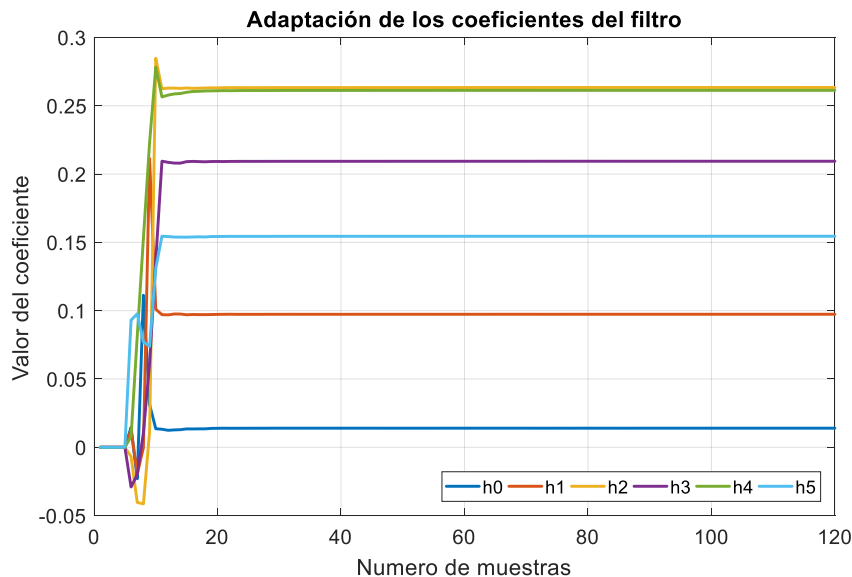


Figura 23.- Adaptación de los coeficientes mediante RLS.

En la figura 23 podemos observar que cada uno de los coeficientes son inicializados en cero y van adaptándose al valor real de cada uno mediante el algoritmo RLS. Además, se puede percibir que a partir de la muestra 10 de los datos tomados ya hay poca variación en cuanto al valor final de cada uno de los coeficientes. La comparativa entre la señal de salida

obtenida con respecto a la que se va obteniendo en cada una de las iteraciones del algoritmo RLS se presenta en la figura 24(a).

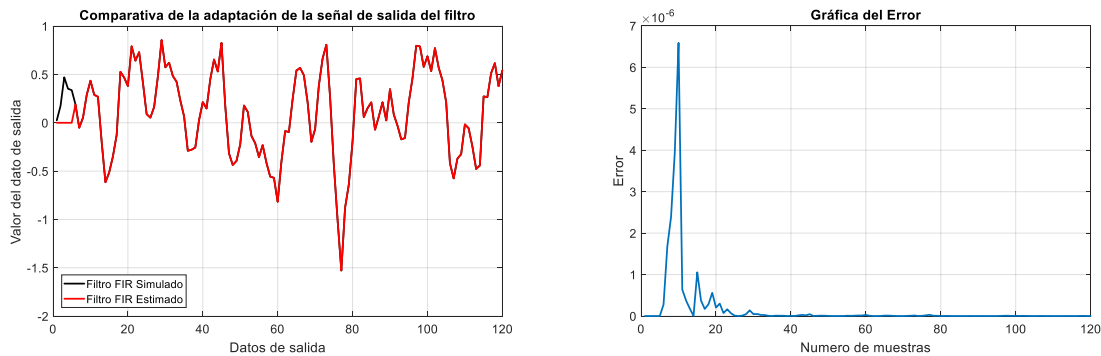


Figura 24.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante RLS, b) Gráfica de error de la señal de salida entre el filtro simulado y el estimado al usar RLS.

En la figura 24(a) se puede observar como el algoritmo RLS intenta obtener la misma salida que el filtro FIR, por lo cual modifica los valores de los coeficientes estimados hasta lograrlo. En dicha figura se puede observar como a partir de la muestra 10, la señal de salida del filtro estimado es similar a la que se ha simulado, concordando con la variación que tienen los coeficientes en dicho punto.

El error generado en el transcurso de la adaptación de los coeficientes entre la señal de salida del filtro estimado con respecto al filtro simulado se presenta en la figura 24(b). En la cual se puede notar como el error disminuye al transcurso de las iteraciones que el algoritmo RLS llevó a cabo. Además, se puede apreciar que el error es muy pequeño a partir de la muestra 10, aunque hay pequeñas variaciones, pero son sumamente pequeñas.

3.1.3. Filtro Kalman

Para observar el desempeño del filtro Kalman en la identificación de los coeficientes de un sistema, se establecieron los vectores para el modelado del sistema en el espacio de estados lo cual proporciono los coeficientes a identificar, los cuales fueron: 0.0395, 0.2072, 0.1957, 0.1734, 0.0211, 0.1404. Además, se definió la incertidumbre en el modelo del proceso y el error de medición como señales aleatorias con un factor de 0.001. La señal de entrada $u(n)$ se definió como una señal aleatoria de 120 muestras y se presenta en la figura 25.

El resultado que proporciona el filtro Kalman al ingresar los parámetros descritos y las señales de entrada y salida del sistema a identificar, son los siguientes coeficientes: $h_0=0.0395$, $h_1=0.2072$, $h_2=0.1957$, $h_3=0.1734$, $h_4=0.0211$, $h_5=0.1404$. La adaptación de los coeficientes se presenta en la figura 26.

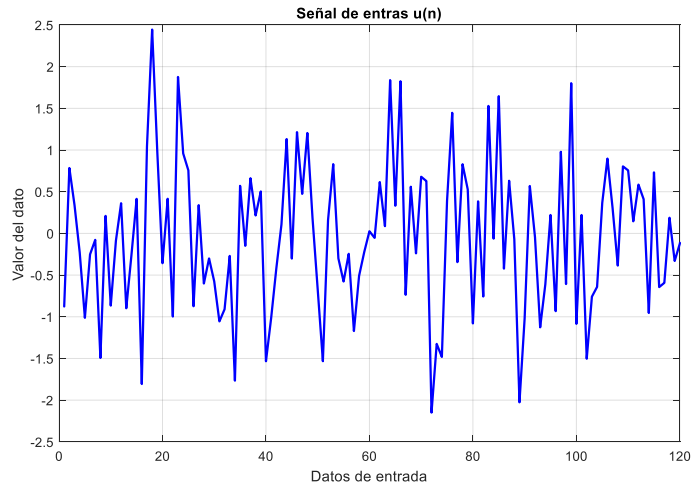


Figura 25.- Señal de entrada para la identificación mediante el Filtro Kalman

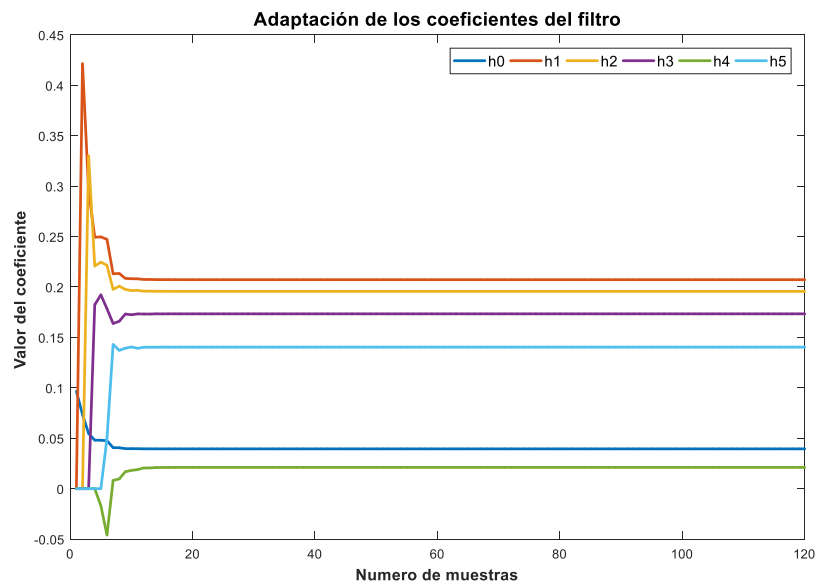


Figura 26.- Adaptación de los coeficientes mediante el filtro Kalman.

En la figura 26 podemos observar que cada uno de los coeficientes son inicializados en cero y van adaptándose al valor real de cada uno mediante el filtro Kalman. Además, se puede percibir que a partir de la muestra 12 de los datos tomados ya hay poca variación en cuanto al valor final de cada uno de los coeficientes, esto depende también del valor de incertidumbre y ruido que se ingresó al sistema. La comparativa entre la señal de salida obtenida con respecto a la que se va obteniendo en cada una de las iteraciones del filtro Kalman se presenta en la figura 27(a).

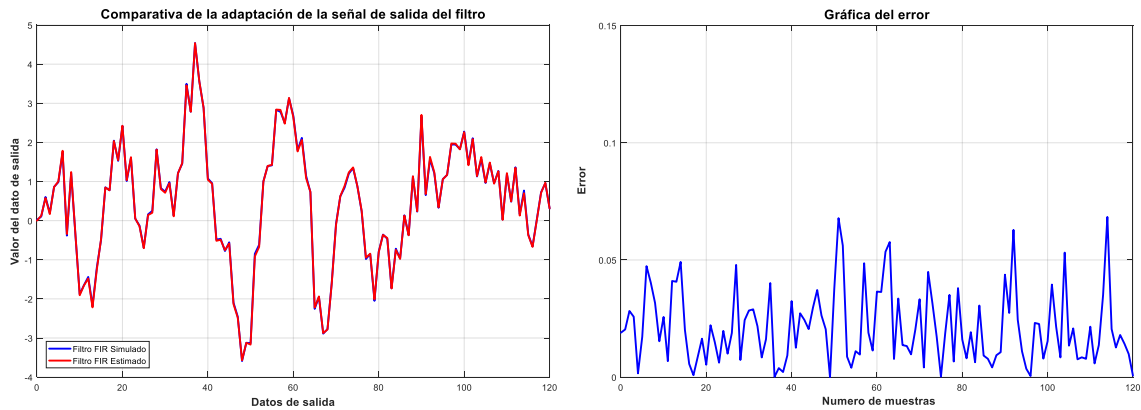


Figura 27.- a) Adaptación de la señal de salida por medio de los coeficientes estimados mediante el filtro Kalman, b) Gráfica de error de la señal de salida entre el filtro simulado y el estimado al usar el filtro Kalman.

En la figura 27(a) se puede observar como el filtro Kalman adapta los coeficientes para obtener la misma respuesta de la salida del sistema. El error generado en el transcurso de la adaptación de los coeficientes entre la señal de salida del filtro estimado con respecto al filtro simulado se presenta en la figura 27(b). En la cual se puede notar como el error es muy pequeño a lo largo de la identificación.

3.2. Evaluación de los algoritmos

Al observar el desempeño de cada una de las propuestas para la identificación descritas en la subsección anterior, podemos observar que el algoritmo RLS y el filtro Kalman presentan mejores resultados, además de que el filtro Kalman puede aún mejorar la identificación modificando sus parámetros. Sin embargo, para la implementación de las dos propuestas es más complicado implementar el filtro Kalman que el algoritmo RLS, ya que consume mayor número de recursos computacionales debido a que usa un mayor número de ecuaciones, operaciones y parámetros para la adaptación de los coeficientes. Por lo cual se optó por implementar el algoritmo RLS para la identificación de sistemas de control.

3.3. Implementación en FPGA.

3.3.1 Optimización del algoritmo.

De acuerdo con los resultados obtenidos anteriormente, se eligió implementar el algoritmo RLS para la identificación de sistemas de control, debido a que este presenta un mejor desempeño que el algoritmo LMS. Su costo computacional es mayor que el LMS, pero es compensado con el ciclo de reloj de la tarjeta FPAG utilizada para la implementación de dicho algoritmo. A continuación, se describirán los diagramas a bloques para la

implementación del algoritmo RLS en lenguaje VHDL (VHSIC – Very High Speed Integrated Circuit – Hardware Descriptio Language).

Antes de la implementación del algoritmo RLS en lenguaje VHDL, se analizaron las ecuaciones 38, 41 y 42 que integran dicho algoritmo, con el objetivo de reducir el número de operaciones como la multiplicación, suma y resta de matrices, las cuales conllevan un mayor consumo de recurso computacionales. Este análisis se realizó con el fin de mejorar la eficiencia del cálculo de los coeficientes que definen la función de transferencia de la planta a identificar. A continuación, se describe el análisis hecho en la reducción de operaciones elementales en el cálculo con el algoritmo RLS.

Para realizar el análisis del cálculo matemático que presenta el algoritmo RLS, se definirán las matrices involucradas con variables constantes. Las matrices que se presentan en el proceso matemático dentro del cálculo y la actualización de la matriz de correlación son P_M la matriz de correlación y U el vector que contiene las entradas de la señal de entrada del sistema, así como H_M vector que contiene los coeficientes con los cuales definirán la función de transferencia de la planta del sistema y que se utiliza en el cálculo del error y la actualización de dichos coeficientes.

En el análisis se comenzará con el caso más simple, el cual se da cuando se tiene solamente dos coeficientes a calcular en un sistema (Sistema de primer orden). Por lo tanto, se definirán las matrices P_M , H_M y U , como se muestra en las ecuaciones 52, 53 y 54:

$$H_M = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix}, P_M = \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix}, U = \begin{bmatrix} y_1 \\ u_1 \end{bmatrix} \quad (52, 53, 54)$$

La ecuación 38 presenta tres cálculos independientes, la multiplicación por el inverso de λ , la resta entre las dos matrices resultantes y la división de una serie de multiplicaciones de matrices entre una constante. Primeramente, se realizó el desarrollo de la serie de multiplicaciones dentro del numerador de la división, lo cual dará como resultado una matriz la cual llamaremos N , esta se presenta en la ecuación 55.

$$N = P_M[n-1]U[n]U^T[n]P_M[n-1] = \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} \begin{bmatrix} y_1 \\ u_1 \end{bmatrix} \begin{bmatrix} y_1 & u_1 \end{bmatrix} \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} \quad (55)$$

Al realizar las operaciones y factorizaciones necesarias se obtiene la matriz presentada en la ecuación 56

$$N = \begin{bmatrix} (y_1 a_2 + u_1 b_2)(y_1 a_2 + u_1 c_2) & (y_1 a_2 + u_1 b_2)(y_1 b_2 + u_1 d_2) \\ (y_1 c_2 + u_1 d_2)(y_1 a_2 + u_1 c_2) & (y_1 c_2 + u_1 d_2)(y_1 b_2 + u_1 d_2) \end{bmatrix} \quad (56)$$

Tomando ahora el denominador de la división, dará como resultado una constante que dividirá la matriz N . La constante resultante la llamaremos D y queda definida por la ecuación 57.

$$D = \lambda + U^T [n] P_M [n-1] U [n] = \lambda + [y_1 \quad u_1] \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} \begin{bmatrix} y_1 \\ u_1 \end{bmatrix} \quad (57)$$

Al realizar las multiplicaciones y agrupaciones necesarias en la ecuación 57 resulta la ecuación 58.

$$D = \lambda + y_1(y_1 a_2 + u_1 c_2) + u_1(y_1 b_2 + u_1 d_2) \quad (58)$$

Para facilitar los cálculos posteriores se realizaron cambios de variables dentro de las ecuaciones 56 y 58, estos cambios se presentan en las ecuaciones 59-62.

$$x_1 = (y_1 a_2 + u_1 b_2) \quad (59) \quad x_2 = (y_1 a_2 + u_1 c_2) \quad (60)$$

$$x_3 = (y_1 b_2 + u_1 d_2) \quad (61) \quad x_4 = (y_1 c_2 + u_1 d_2) \quad (62)$$

Al reescribir las ecuaciones 56 y 58 utilizando las ecuaciones 59-62, y colocándolas en forma de fracción, se obtiene la expresión presentada en la ecuación 63, la cual nombraremos B .

$$B = \frac{N}{D} = \frac{\begin{bmatrix} x_1 x_2 & x_1 x_3 \\ x_4 x_2 & x_4 x_3 \end{bmatrix}}{\lambda + y_1 x_2 + u_1 x_3} = \begin{bmatrix} \frac{x_1 x_2}{\lambda + y_1 x_2 + u_1 x_3} & \frac{x_1 x_3}{\lambda + y_1 x_2 + u_1 x_3} \\ \frac{x_4 x_2}{\lambda + y_1 x_2 + u_1 x_3} & \frac{x_4 x_3}{\lambda + y_1 x_2 + u_1 x_3} \end{bmatrix} \quad (63)$$

Se puede reescribir la ecuación 38 como en la ecuación 64.

$$P_M [n] = \frac{1}{\lambda} \left(\begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} - B \right) \quad (64)$$

Realizando la operación de sustracción entre matrices se obtiene la ecuación 65.

$$P_M [n] = \frac{1}{\lambda} \left(\begin{bmatrix} a_2 - \frac{x_1 x_2}{\lambda + y_1 x_2 + u_1 x_3} & b_2 - \frac{x_1 x_3}{\lambda + y_1 x_2 + u_1 x_3} \\ c_2 - \frac{x_4 x_2}{\lambda + y_1 x_2 + u_1 x_3} & d_2 - \frac{x_4 x_3}{\lambda + y_1 x_2 + u_1 x_3} \end{bmatrix} \right) \quad (65)$$

Se puede observar que el término en el denominador de cada uno de los términos dentro de la matriz de autocorrelación es el mismo en cada uno de los elementos. Además, aplicando la propiedad de la multiplicación de una constante por una matriz, se agrega la multiplicación de λ por cada uno de los denominadores de la matriz. Por lo tanto, se hizo un cambio de variable en el cual $m = \lambda(\lambda + y_1 x_2 + u_1 x_3)$. Reduciendo los términos que se restan y haciendo el cambio de variable definido anteriormente (m), se obtiene la ecuación 66.

$$P_M[n] = \begin{bmatrix} \frac{a_2\lambda + a_2y_1x_2 + a_2u_1x_3 - x_1x_2}{m} & \frac{b_2\lambda + b_2y_1x_2 + b_2u_1x_3 - x_1x_3}{m} \\ \frac{c_2\lambda + c_2y_1x_2 + c_2u_1x_3 - x_4x_2}{m} & \frac{d_2\lambda + d_2y_1x_2 + d_2u_1x_3 - x_4x_3}{m} \end{bmatrix} \quad (66)$$

A continuación, se agrupan los términos semejantes y se realiza la sustitución de algunas variables que se definieron en las ecuaciones 58-62, y se obtiene la ecuación 67.

$$P_M[n] = \begin{bmatrix} \frac{a_2\lambda + a_2u_1x_3 + x_2(\cancel{a_2y_1} - \cancel{a_2y_1} - b_2u_1)}{m} & \frac{b_2\lambda + b_2y_1x_2 + x_3(\cancel{b_2u_1} - \cancel{b_2u_1} - a_2y_1)}{m} \\ \frac{c_2\lambda + c_2u_1x_3 + x_2(\cancel{c_2y_1} - \cancel{c_2y_1} - d_2u_1)}{m} & \frac{d_2\lambda + d_2y_1x_2 + x_3(\cancel{d_2u_1} - \cancel{d_2u_1} - c_2y_1)}{m} \end{bmatrix} \quad (67)$$

Se realizan las reducciones necesarias y se obtiene la ecuación 68.

$$P_M[n] = \begin{bmatrix} \frac{a_2(\lambda + u_1\{x_3 - \frac{b_2}{a_2}x_2\})}{m} & \frac{b_2(\lambda + y_1\{x_2 - \frac{a_2}{b_2}x_3\})}{m} \\ \frac{c_2(\lambda + u_1\{x_3 - \frac{d_2}{c_2}x_2\})}{m} & \frac{d_2(\lambda + y_1\{x_2 - \frac{c_2}{d_2}x_3\})}{m} \end{bmatrix} \quad (68)$$

Tomando uno de los elementos de la matriz de correlación para ejemplificar el proceso de reducción, se tomó el primer elemento $\left(\frac{a_2(\lambda + u_1\{x_3 - \frac{b_2}{a_2}x_2\})}{m}\right)$ y se realizaron las sustituciones y operaciones como se muestra en la ecuación 69.

$$\frac{a_2(\lambda + u_1\{x_3 - \frac{b_2}{a_2}x_2\})}{m} = \frac{a_2(\lambda + u_1\{\cancel{y_1}b_2 + u_1d_2 - \frac{b_2}{a_2}(\cancel{y_1}a_2 + u_1c_2)\})}{m} = \frac{a_2(\lambda + u_1\{u_1d_2 - \frac{b_2c_2}{a_2}(u_1)\})}{m} = \frac{a_2\lambda + u_1^2(a_2d_2 - b_2c_2)}{m} \quad (69)$$

Realizando el mismo proceso para cada uno de los elementos de la matriz se obtiene la ecuación 70.

$$P_M[n] = \begin{bmatrix} \frac{a_2\lambda + u_1^2(a_2d_2 - b_2c_2)}{m} & \frac{b_2\lambda - u_1y_1(a_2d_2 - b_2c_2)}{m} \\ \frac{c_2\lambda - u_1y_1(a_2d_2 - b_2c_2)}{m} & \frac{d_2\lambda + y_1^2(a_2d_2 - b_2c_2)}{m} \end{bmatrix} \quad (70)$$

De acuerdo con la ecuación 70 se puede observar que si se diseña un sistema que realice los cálculos para determinar el primer elemento de la matriz, se puede reutilizar para determinar todos los otros elementos de la matriz, lo cual facilita en el diseño en FPGA.

La segunda parte del algoritmo de RLS es la actualización de los coeficientes de la función de transferencia H_M , la cual se realiza después de la actualización de la matriz de correlación presentada en la ecuación 70. De igual manera se desarrollaron cada una de las multiplicaciones de las matrices para encontrar una ecuación representada sin matrices. Primeramente, se obtiene el error generado por la diferencia del dato deseado $d[n]$ y la

multiplicación de los datos de entrada (Ec. 54) y el vector de coeficientes de entrada (Ec. 52) la cual se presentó en la ecuación 41. Haciendo las sustituciones que se definieron en las ecuaciones 52,53 y 54, el error se puede escribir como se presenta en la ecuación 71.

$$e_M[n] = d[n] - [y_1 \quad u_1] \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = d[n] - (y_1 a_1 + u_1 b_1) \quad (71)$$

Se puede observar que la ecuación 71 genera un valor numérico que posteriormente multiplica a la matriz resultado de multiplicar la matriz de correlación actualizada por los datos de entrada. Por lo tanto, la ecuación 42 se puede escribir como la ecuación 72.

$$H_M[n] = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} + \begin{bmatrix} a_2 & b_2 \\ c_2 & d_2 \end{bmatrix} \begin{bmatrix} y_1 \\ u_1 \end{bmatrix} e_M[n] \quad (72)$$

De donde al realizar las operaciones involucradas da como resultado la ecuación 73.

$$H_M[n] = \begin{bmatrix} a_1 + e_M[n](a_2 y_1 + b_2 u_1) \\ b_1 + e_M[n](c_2 y_1 + d_2 u_1) \end{bmatrix} \quad (73)$$

Al sustituir el error resulta la ecuación 74.

$$H_M[n] = \begin{bmatrix} a_1 + (d[n] - (y_1 a_1 + u_1 b_1))(a_2 y_1 + b_2 u_1) \\ b_1 + (d[n] - (y_1 a_1 + u_1 b_1))(c_2 y_1 + d_2 u_1) \end{bmatrix} \quad (74)$$

De la ecuación 74 se puede deducir la misma analogía que se realizó en la ecuación 70, es decir, se puede diseñar un sistema para encontrar el valor del primer elemento del vector H_M , y el posterior se calcula reutilizando el sistema diseñado, pero con los parámetros que le corresponde. También se observa que el cálculo del error es similar en ambos elementos del vector, por lo cual, se pueden evitar algunas operaciones para optimizar el algoritmo.

Para un sistema de tres coeficientes, se definen las matrices P_M , H_M y U , como se muestra en las ecuaciones 75, 76 y 77.

$$H_M = \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix}, P_M = \begin{bmatrix} a_2 & b_2 & e_2 \\ c_2 & d_2 & g_2 \\ f_2 & h_2 & i_2 \end{bmatrix}, U = \begin{bmatrix} y_1 \\ u_1 \\ u_2 \end{bmatrix} \quad (75, 76, 77)$$

Al realizar un análisis similar se obtuvieron las ecuaciones 78 y 79, las cuales representan la matriz de correlación resultante y el vector con los coeficientes actualizados respectivamente.

$$P_M = \begin{bmatrix} \frac{a_2 \lambda + n_1 p_1 + n_4 p_2 + n_4 p_3 + n_5 p_4}{m} & \frac{b_2 \lambda - n_2 p_1 + n_4 p_5 - n_6 p_3 + n_5 p_6}{m} & \frac{e_2 \lambda - n_2 p_2 - n_1 p_5 - n_6 p_4 - n_4 p_6}{m} \\ \frac{c_2 \lambda - n_2 p_1 - n_6 p_2 + n_4 p_7 + n_5 p_8}{m} & \frac{d_2 \lambda + n_3 p_1 - n_6 p_5 - n_6 p_7 + n_5 p_9}{m} & \frac{g_2 \lambda + n_3 p_2 + n_2 p_5 - n_6 p_8 - n_4 p_9}{m} \\ \frac{f_2 \lambda - n_2 p_3 - n_6 p_4 - n_1 p_7 - n_4 p_8}{m} & \frac{h_2 \lambda + n_3 p_3 - n_6 p_6 + n_2 p_7 - n_4 p_9}{m} & \frac{i_2 \lambda + n_3 p_4 + n_2 p_6 + n_2 p_8 + n_1 p_9}{m} \end{bmatrix} \quad (78)$$

Donde $n_1 = u_1^2$, $n_2 = u_1y_1$, $n_3 = y_1^2$, $n_4 = u_1u_2$, $n_5 = u_2^2$, $n_6 = u_2y_1$,
 $p_1 = (a_2d_2 - b_2c_2)$, $p_2 = (a_2g_2 - c_2e_2)$, $p_3 = (a_2h_2 - b_2f_2)$, $p_4 = (a_2i_2 - e_2f_2)$,
 $p_5 = (b_2g_2 - d_2e_2)$, $p_6 = (b_2i_2 - h_2e_2)$, $p_7 = (c_2h_2 - d_2f_2)$, $p_8 = (c_2i_2 - g_2f_2)$,
 $p_9 = (d_2i_2 - h_2g_2)$ y $m = \lambda(\lambda + y_1(y_1a_2 + u_1c_2 + u_2f_2) + u_1(y_1b_2 + u_1d_2 + u_2h_2) +$
 $u_2(y_1e_2 + u_1g_2 + u_2i_2))$.

$$H_M = \begin{bmatrix} a_1 + (d[n] - m_2)(a_2y_1 + b_2u_1 + c_2u_2) \\ b_1 + (d[n] - m_2)(d_2y_1 + e_2u_1 + f_2u_2) \\ c_1 + (d[n] - m_2)(g_2y_1 + h_2u_1 + i_2u_2) \end{bmatrix} \quad (79)$$

Donde $m_2 = (y_1a_1 + u_1b_1 + u_2c_1)$.

Al desarrollar las operaciones en el cálculo de los coeficientes en el algoritmo RLS, se pudo observar que se pueden evitar algunas operaciones para optimizar el consumo de recursos en la implementación del algoritmo en una plataforma de programación, así como también se observó que el cálculo de los elementos dentro de la matriz resultante y el vector con los coeficientes actualizados, son repetitivos y puede diseñarse un sistema fundamental y reutilizarlo para obtener el cálculo de los otros elementos involucrados. A continuación, se presenta la propuesta desarrollada en lenguaje VHDL para una tarjeta FPGA.

3.3.2 Desarrollo del algoritmo RLS en FPGA

Las ecuaciones 78 y 79 obtenidas en la sección anterior son la base del desarrollo del algoritmo en lenguaje VHDL del algoritmo adaptativo RLS. EL diagrama general a bloques se muestra en la figura 28. De acuerdo con las ecuaciones 78 y 79 se puede observar que para realizar el cálculo de la matriz de correlación del algoritmo RLS se ocupan obtener 25 cálculos generales, los cuales se reutilizan en la mayoría de las ecuaciones que representan el cálculo de cada elemento de la matriz; estos cálculos son n_1 , n_2 , n_3 , n_4 , n_5 , n_6 , p_1 , p_2 , p_3 , p_4 , p_5 , p_6 , p_7 , p_8 , p_9 y m , donde m incluye $a_2\lambda$, $b_2\lambda$, $c_2\lambda$, $d_2\lambda$, $e_2\lambda$, $f_2\lambda$, $g_2\lambda$, $h_2\lambda$, $i_2\lambda$ y λ^2 . A partir de estos cálculos generales solo se necesita hacer la combinación correcta de sumas y multiplicaciones que representen el valor para cada elemento de la matriz de correlación.

Debido a este análisis, en el diagrama general que se diseñó, primeramente, se obtuvieron los datos de entrada iniciales del algoritmo, estos se adquieren cada periodo de muestreo (señal Fs) y es la señal de inicio para la FSM_1. Cabe mencionar, que debido a que el algoritmo RLS es cíclico, la primera vez que llega la señal Fs, el algoritmo toma los valores iniciales determinados por el usuario, en el siguiente periodo de muestreo el algoritmo toma los valores obtenidos de la matriz de correlación calculados en el primer periodo de tiempo. Los datos adquiridos se guardaron en la RAM_1. La RAM_1 se reordeno mediante una ROM_1 en la RAM_2 para poder utilizar un bloque general (Bloque Determinantes) y

obtener los cálculos generales mencionados anteriormente. Para ingresar los datos al Bloque Determinantes, se utilizó un registro (Registro Dezp_1) que guardara los datos involucrados en la operación para obtener un cálculo general (por ejemplo, para p_1 se necesitan los datos a_2, d_2, b_2, c_2), al obtener un cálculo general se guardó en RAM_3. Lo anterior mencionado se realizó mediante contadores y la máquina de estados FSM_1 que sincronizó todo el proceso.

A continuación, la FMS_1 genera una señal de salida que da comienzo a la FSM_2, la cual se encarga de sincronizar el cálculo de los valores del denominador y numerador de cada uno de los elementos de la matriz de correlación. En esta sección hay dos bloques, uno de multiplicación (MULTIP) y otro de sumas y restas (Bloque Suma-Resta). En el bloque MULTIP se realizaron las multiplicaciones que implica el multiplicar un cálculo general n y un cálculo general p (por ejemplo n_1 por p_1), además de las multiplicaciones para obtener m (por ejemplo $a_2\lambda$ por n_3), por lo tanto, de forma similar se utilizó un registro (Registro Dezp_2) que guardara los datos involucrados en la multiplicación. Posteriormente, los productos obtenidos se guardaron en un registro (Registro Dezp_3), ya que después fueron utilizados para obtener la suma o resta que implica obtener la parte del numerador para el cálculo de cada elemento de la matriz de correlación (por ejemplo $a_2\lambda + n_1p_1 + n_4p_2 + n_4p_3 + n_5p_4$). De igual manera se obtuvo en el mismo proceso los datos que representaron la parte del denominador (m) de cada uno de los elementos de la matriz de correlación (se obtuvo en dos partes, ya que el bloque de Suma-Resta tiene cinco elementos de entrada y para el cálculo de m se ocupan diez elementos). Cabe resaltar que m es igual para todos los elementos, por lo cual, solo es necesario calcularlo una vez y reutilizarlo en el cálculo de la división de cada uno de los elementos de la matriz. Por último, se guardaron los datos de los numeradores de cada uno de los elementos de la matriz, así como los dos datos que representaron el denominador, en un registro (Registro Dezp_4).

Posteriormente, la FMS_2 genera una señal de salida que da comienzo a la FSM_3, la cual se encarga de sincronizar el funcionamiento del algoritmo COORDIC (COordinate Rotation DIgital Computer) para realizar la división que está presente en cada elemento de la matriz de correlación. Antes que comience el algoritmo COORDIC a trabajar, se realiza la suma de los elementos que se obtuvieron en el bloque anterior del denominador de la división para el cálculo de cada uno de los elementos de la matriz. Al efectuarse la división los datos obtenidos se guardan en un registro (Registro D.), el cual servirá para retroalimentar el algoritmo en el siguiente periodo de muestreo, además de proporcionar los datos de entrada para calcular la actualización de los coeficientes, los cuales determinaran la función de transferencia de la planta.

La actualización de los coeficientes se efectúa cuando la FMS_3 genera una señal de salida que da comienzo a la FSM_4. De igual forma como se planteó la arquitectura para efectuar el cálculo de la matriz de correlación, en esta sección se generó un sistema para obtener la actualización general de uno de los coeficientes y posteriormente reutilizar ese

sistema para obtener la actualización de los otros coeficientes. Aunque cabe mencionar que la actualización se realiza de forma semi-paralela (se va realizando paso a paso para cada coeficiente, pero con un solo elemento de multiplicación, suma o resta) de cada uno de los coeficientes. Primeramente, se realizaron las multiplicaciones que están presente en la ecuación 79 mediante el bloque MULTIP, las cuales son 12, ya que tres multiplicaciones son comunes en el cálculo de los tres coeficientes (m_2). El resultado es guardado en un registro (Registro Dezp_6) para posteriormente ser sumado en el bloque de suma (B_Suma). Al tener las sumas de las multiplicaciones que interactúan en la ecuación de la actualización de coeficientes en un registro (Registro Dezp_7), lo siguiente es obtener el error. El error se obtiene de la resta de la señal deseada (D_{ns}) y m_2 (B_Resta), el cual es el mismo para los tres coeficientes, por lo tanto, solo es necesario calcularlo una vez. A continuación, se multiplica el error por los datos obtenidos en el proceso de actualización mediante el bloque de MULTIP. Finalmente se suma el dato de corrección a los coeficientes para generar los coeficientes actualizados que se utilizaron en el próximo periodo de muestreo para proseguir actualizando los coeficientes hasta obtener los parámetros de la planta a identificar.

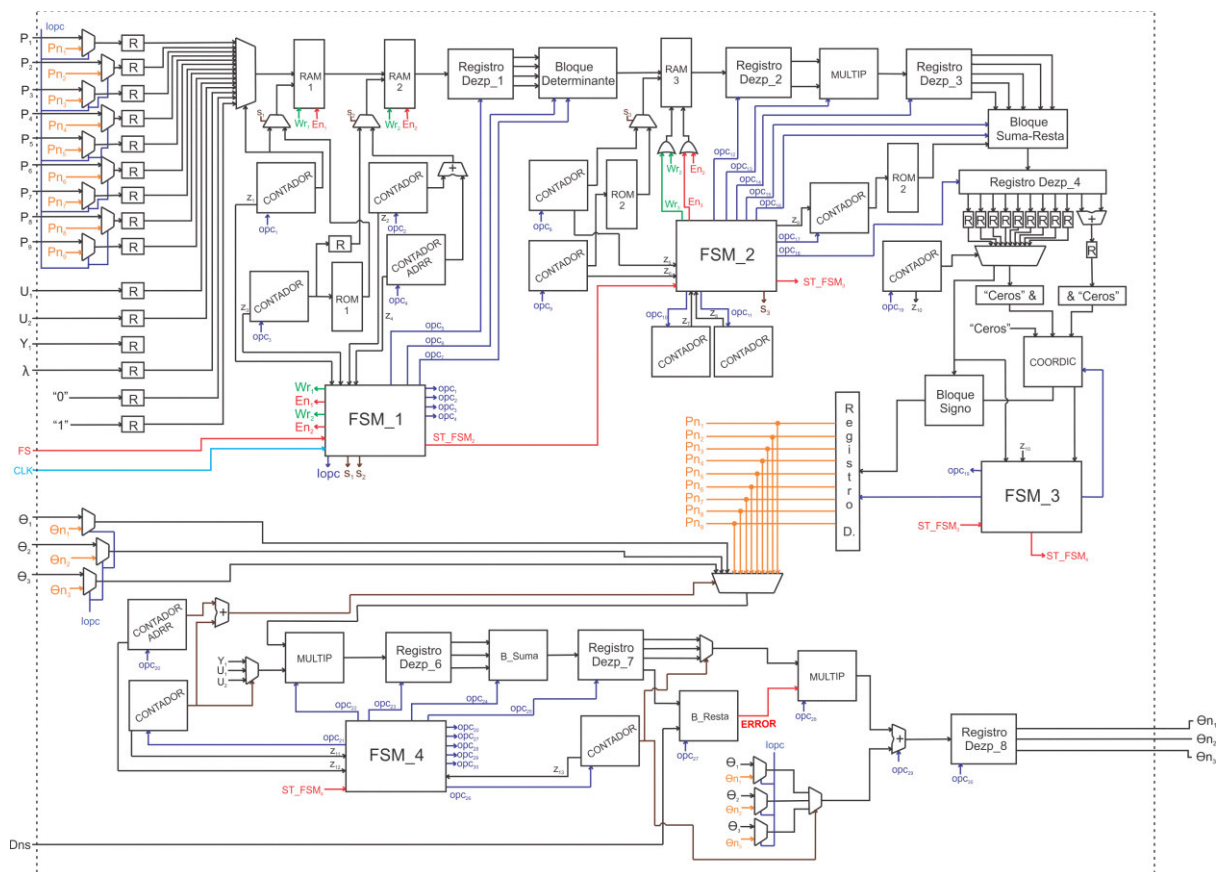


Figura 28.- Diagrama a bloques Algoritmo RLS

Concluida la propuesta para la implementación del algoritmo adaptativo en la tarjeta programable, se continuo con las pruebas experimentales para probar el funcionamiento de la metodología de identificación de dos sistemas de control, un circuito de primer orden RC y la velocidad de un motor de DC. En el siguiente capítulo se presenta los resultados obtenidos en los dos casos de estudio mencionados, como la propuesta metodológica para obtener la función de transferencia de un sistema de control.

4

CAPÍTULO

Pruebas y Resultados

En este capítulo se presentarán los resultados obtenidos al llevar a cabo la metodología propuesta en este trabajo de tesis. Primeramente, se presentará la propuesta de la metodología utilizada para la obtención de la función de transferencia de dos sistemas de control en estudio. Posteriormente, se presentarán los resultados obtenidos en los casos de estudio, los cuales fueron la identificación de un sistema de primer orden (Circuito RC) y la velocidad de un motor de DC, mediante una etapa experimental y la metodología propuesta. Por último, se presentan los recursos utilizados en la tarjeta programable donde se implementó la metodología propuesta

4.1. Metodología Propuesta

El diagrama general de la metodología propuesta en este proyecto de investigación se muestra en la figura 29, en la primera etapa se definen los parámetros para identificar al sistema, en la segunda etapa se envían y almacenan las señales de entrada y salida de la planta mediante un sistema de adquisición de datos (DAS, Data Acquisition System), finalmente en la tercera etapa se filtra la señal de salida de la planta y a partir de esta señal y la señal de entrada se estima la función de transferencia (FT) con los algoritmos adaptativos.

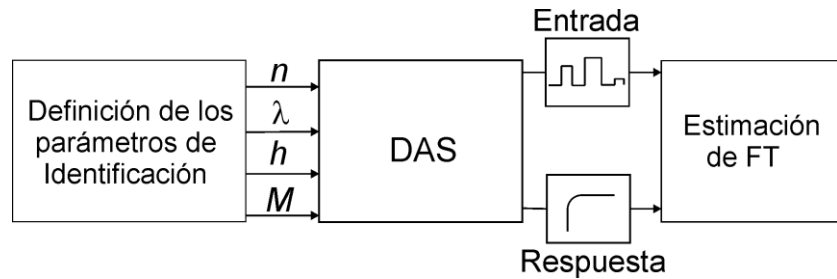


Figura 29.- Diagrama general de la metodología para identificación mediante el algoritmo RLS

A continuación, se presenta más detalladamente la estructura de la metodología propuesta en este trabajo de investigación. Cabe mencionar que se agregó una etapa de filtrado digital para obtener mejores resultados en la identificación del sistema, los resultados se presentan más adelante en este capítulo.

La propuesta para el sistema de identificación se ha hecho con el uso de un FPGA y una PC, lo cual se muestra en la figura 30. Parte de la primera etapa de esta propuesta es desarrollada en el entorno Matlab, donde se definen los siguientes parámetros: orden de la FT, el cual define qué orden tendrá la función de transferencia estimada; criterio de ajuste, se puede definir como el paso con el cual el algoritmo adaptativo ajusta los parámetros para definir la función de transferencia (λ); señal de entrada, es la señal con la cual se excitara el sistema para obtener la respuesta del mismo; y por último el orden y coeficientes del filtro FIR, los cuales son los parámetros para el filtro que filtrará la señal obtenida experimentalmente para obtener un mejor resultado en la identificación del sistema. La etapa dos se desarrolló en el FPGA, a donde llegan los parámetros deseados para la identificación de la FT, esta interacción se realizó mediante el protocolo de comunicación Rs-232. A partir de esta información, el sistema de adquisición de señales (DAS) sincroniza el envío de la entrada y respuesta de la planta, los cuales son almacenados en una memoria (SDRAM, synchronous dynamic random-access memory). Por otro lado, el módulo de entrada se encarga de enviar la señal de excitación hacia la etapa de potencia de la planta, mientras tanto el DAS de forma paralela almacena esta señal y la respuesta de la planta; la señal de respuesta puede ser enviada como una señal analógica o digital, ya que el decodificador se encarga de interpretar esta información.

Finalmente, la tercera etapa es comandada por el DAS quien envía las señales hacia el procesador FIR-RLS. Previo a realizar la identificación del sistema la señal de salida de la planta es filtrada mediante un filtro FIR que puede ser reconfigurado desde un orden 4 hasta un orden 32, con lo cual se logra mejorar la estimación de la FT. Una vez filtrada la señal de salida de la planta, se estima la FT utilizando el algoritmo RLS con los parámetros que se definieron en la etapa 1.

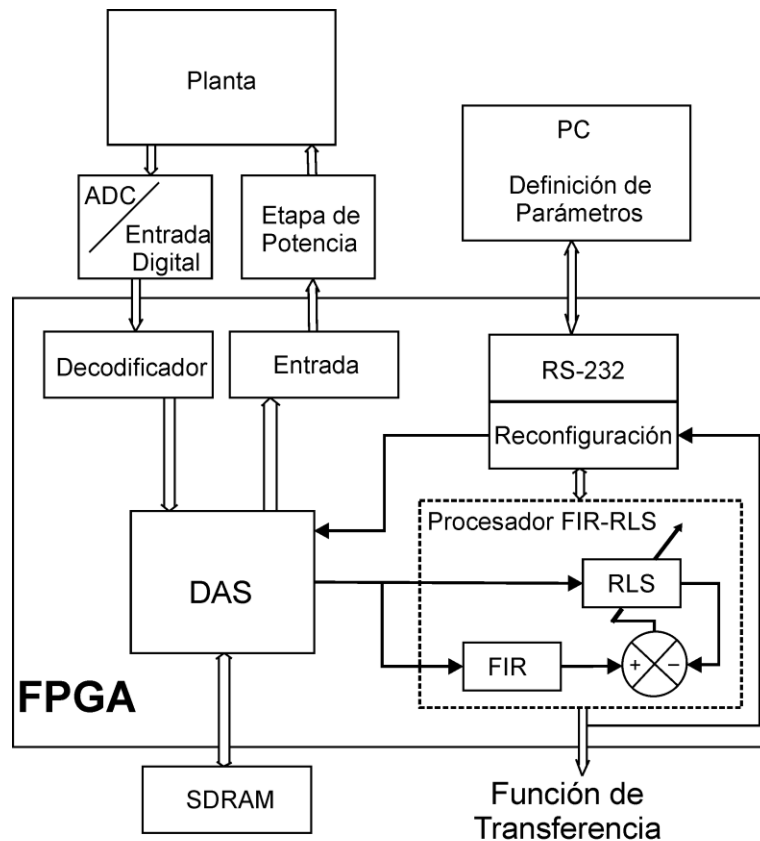


Figura 30.- Diagrama de la metodología propuesta

Obtenida la metodología para la identificación de un sistema de control, se paso a la etapa experimental para obtener la función de transferencia de dos sistemas de control en casos de estudios.

4.2. Etapa Experimental

La etapa experimental de esta metodología se presenta en la figura 31, donde se pueden observar los dos casos de estudio utilizados para validar esta propuesta. El primer caso consiste en un circuito RC, donde la salida de esta planta es analógica, esta es capturada con el ADC128S022 que tiene una resolución de 12 bits. El segundo caso de estudio es un motor de DC de 11000 RPM, donde la salida digital es proporcionada por un encoder en cuadratura de 64 cuentas por revolución. Para implementar el sistema desarrollado se utilizó el Kit de desarrollo DE1, que cuenta con el FPGA EP2C20F484C7N, una memoria SDRAM de 8Mbyte y un transceptor para comunicación RS-232.

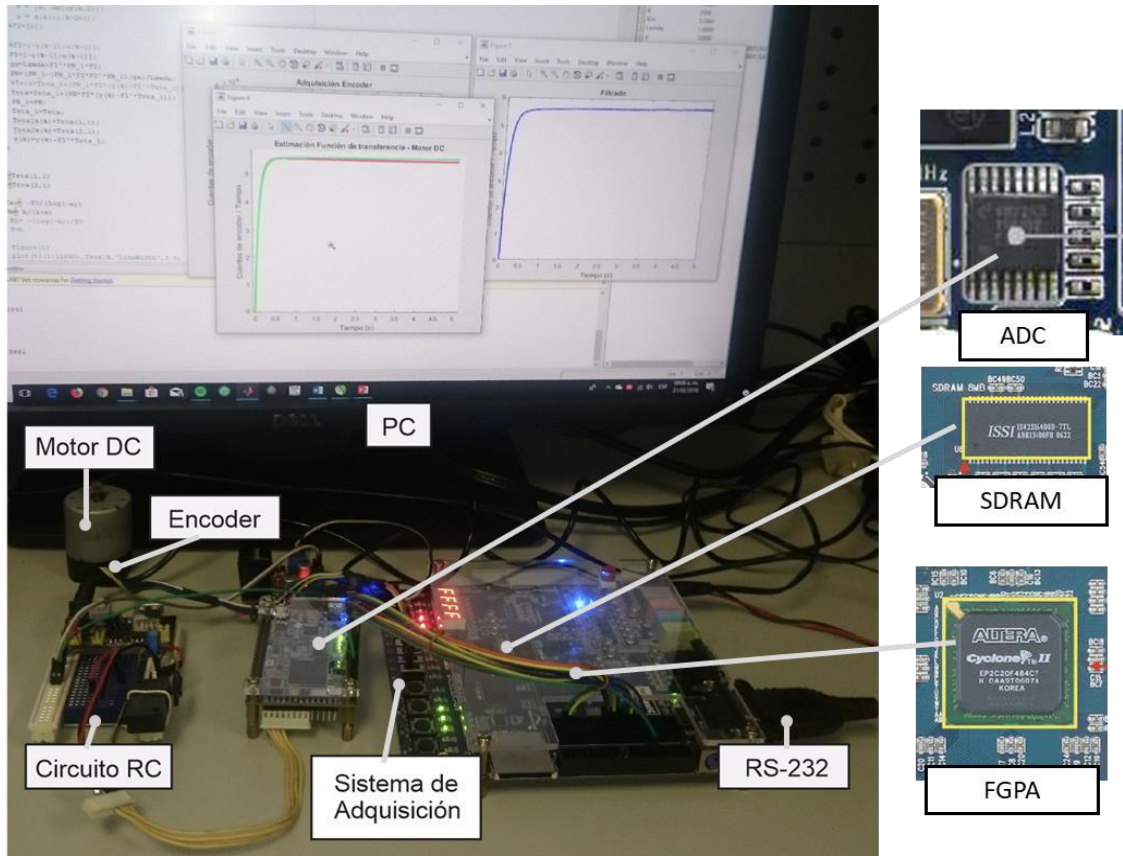


Figura 31.- Etapa experimental

4.2.1 Caso de estudio 1: Circuito RC

Un circuito RC es un circuito compuesto de elementos pasivos conectados por una fuente eléctrica. Su esquema general más simple se presenta en la figura 32.

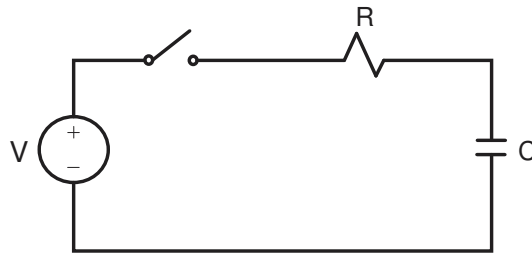


Figura 32.- Circuito RC

La función de transferencia analítica de un circuito RC queda definida por la ecuación 80.

$$G(s) = \frac{K}{\tau s + 1} \quad (80)$$

Donde, τ es la constante de tiempo de la planta y está definida por el producto de la resistencia y capacitancia de los elementos que lo componen y K es la ganancia de la planta. Si la ecuación 80 se discretiza se obtiene la ecuación en diferencia descrita por la ecuación 81.

$$y(n) = -h_2 y(n-1) + h_1 u(n-1) \quad (81)$$

Donde, los coeficientes quedan definidos por las ecuaciones 82 y 83

$$h_1 = K \left(1 - e^{-\frac{T}{\tau}}\right), \quad h_2 = -e^{-\frac{T}{\tau}} \quad (82, 83)$$

A partir de estas ecuaciones se obtiene la respuesta analítica del sistema; por lo tanto, con la propuesta se estiman los coeficientes h_1 y h_2 , los cuales corresponden al vector de coeficientes H_M estimados por el RLS. Para obtener la respuesta teórica de la planta, se definió una señal escalón de 3.3 V, una resistencia de 1 Mohm y un capacitor de 1uF. Para realizar la estimación de la FT con la metodología propuesta se definieron los siguientes parámetros una frecuencia de muestreo de 2 KHz, un filtro FIR pasa-bajo de orden 8, 16 y 32 y un valor de $\lambda = 0.9998$. Los resultados obtenidos se presentan en la figura 33.

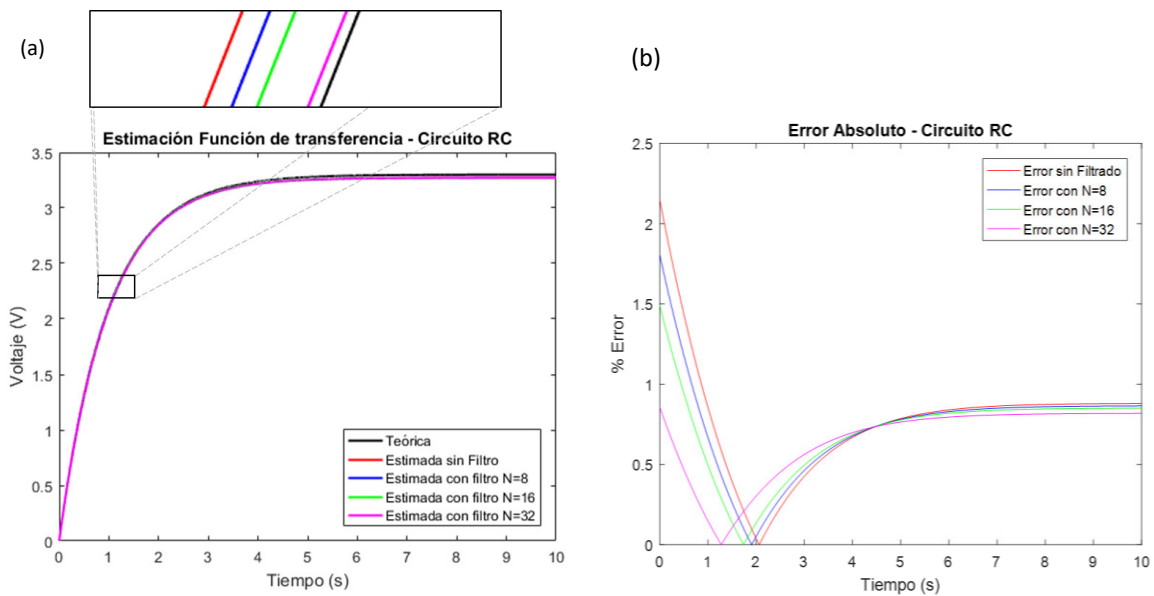


Figura 33.- Resultados de la identificación de la FT del Circuito RC a) Respuesta de la FT para el Circuito RC, b) Error absoluto

Utilizando los parámetros definidos en el párrafo anterior, en la figura 33 (a) se muestra la comparación de la respuesta del sistema con la FT estimada contra la respuesta del sistema obtenida de forma teórica. Con el filtro de orden 8 se obtuvo una ganancia $K=3.2594$ y un $\tau=0.9645$, con el de orden 16 se obtuvo una ganancia $K=3.26$ y un $\tau=0.9678$ y con el de orden 32 se obtuvo una ganancia $K=3.2713$ y un $\tau=0.9733$. Como se puede observar en la figura, el comportamiento de las respuestas es casi imperceptible; sin embargo,

al hacer un acercamiento se puede ver claramente como mejora la estimación de la planta al aumentar el orden del filtro. De forma cuantitativa, la figura 33 (b) muestra el error absoluto de la respuesta del sistema obtenido de forma teórica, con las respuestas de la FT utilizando solo el RLS y la respuesta de las FT estimada con la metodología propuesta; donde el error más grande, se obtiene cuando la salida del sistema no es estimada con la propuesta y el más pequeño es obtenido cuando se utiliza la metodología propuesta con un filtro de orden 32.

4.2.2 Caso de estudio 2: Motor de DC

El motor DC es uno de los sistemas dinámicos más utilizados en la prueba de algoritmos de control debido a que su función de transferencia puede ser reducida a un sistema lineal de orden dos [47]. Por lo tanto, la FT de un motor de DC con respecto al voltaje de entrada queda representada por la ecuación 84, tomando en consideración que en la práctica la inductancia de la armadura del motor es muy pequeña [48].

$$G(s) = \frac{K_m}{\tau s + 1} \quad (84)$$

Donde, K_m representa la ganancia del motor que incluye valores como la resistencia de armadura, el coeficiente de amortiguamiento y la constante de torque. Y τ representa la constante de tiempo del motor. Las ecuaciones para obtener la función de transferencia son las mismas que se presentaron en el caso de estudio anterior (Ecs. 81-83). Para este caso de estudio, la estimación de la FT con la propuesta se definieron los siguientes parámetros frecuencia de muestreo de 1 KHz, un filtro FIR pasa-bajo de distinto orden 8, 16 y 32 y un valor de $\lambda = 0.9998$. Los resultados obtenidos se presentan en la figura 34.

Al utilizar los parámetros definidos en el párrafo anterior para la FT estimada del segundo caso de estudio se obtuvieron los siguientes coeficientes, con el filtro de orden 8 se obtuvo una ganancia $K_m = 11.2417$ y un $\tau = 0.0809$, con el de orden 16 se obtuvo una ganancia $K_m = 11.2840$ y un $\tau = 0.0964$ y con el de orden 32 se obtuvo una ganancia $K_m = 11.3187$ y un $\tau = 0.1142$. Para este caso de estudio tanto la señal de entrada como la salida de la planta fueron enviadas hacia la PC, donde se estimó la FT con el toolbox IDENT de Matlab y a partir de la respuesta de esta FT y las repuestas estimadas con la propuesta se obtuvieron las gráficas que se muestra en la figura 34 (a), donde se puede ver que prácticamente todas las respuestas son similares, sin embargo al realizar un acercamiento se ve como mejora la respuesta de la estimación realizada utilizando la metodología propuesta. De igual forma, en la figura 34 (b) se muestran los errores absolutos de las estimaciones realizadas tomado como referencia la estimación del toolbox IDENT. Para este caso el error más grande es cuando la FT es estimada utilizando solo el algoritmo RLS y el error disminuye al utilizar la propuesta de este trabajo.

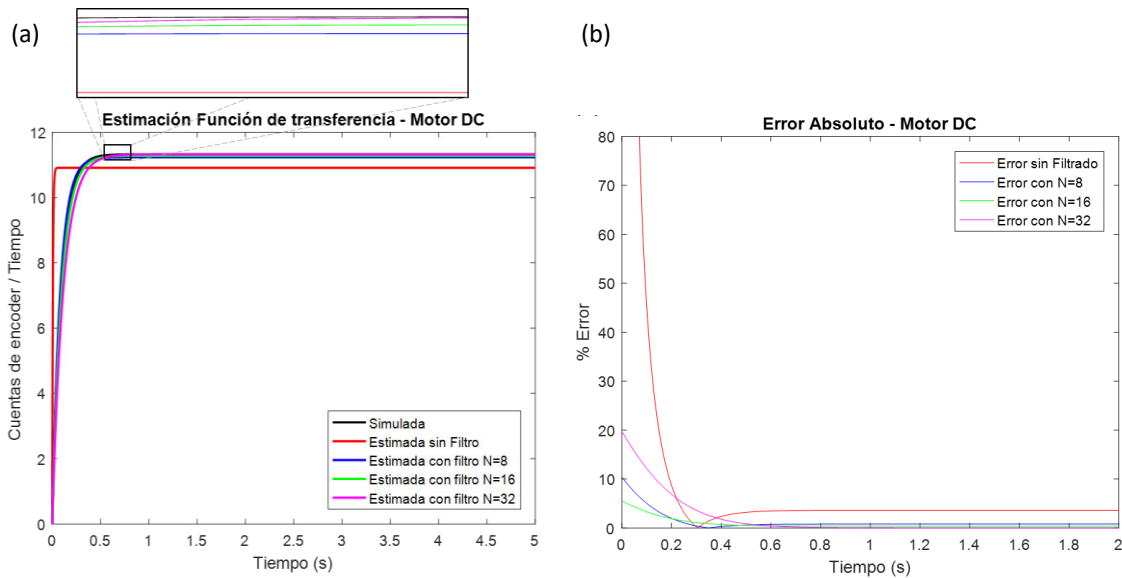


Figura 34.- Resultados de la identificación de la FT del Motor de DC a) Respuesta de la FT para el Motor de DC, b) Error absoluto

4.3 Diseño de Controladores

Obtenida la función de transferencia de cada uno de los casos de estudio, se diseñaron algunos controladores para mejorar la dinámica del sistema en lazo cerrado. El diseño de los controladores se basó en el método en frecuencia [2, 49].

Para el caso de estudio 1 se obtuvo la función de transferencia descrita en la ecuación 85 mediante el método experimental:

$$G(s) = \frac{3.2713}{s+1.02743} \quad (85)$$

Donde la respuesta de este sistema en lazo cerrado se presenta en la figura 35. En la figura 35 se puede observar que el sistema presenta error en estado estable, ya que no llega a la referencia. Por lo tanto, se diseñaron dos controladores para corregir este error. El primer controlador propuesto fue un PI, donde se propuso un tiempo de respuesta de 0.5 segundos y un margen de fase de 70° . La función de transferencia que representa el controlador se presenta en la ecuación 86.

$$G_{PI} = \frac{3.141s+16.71}{s} \quad (86)$$

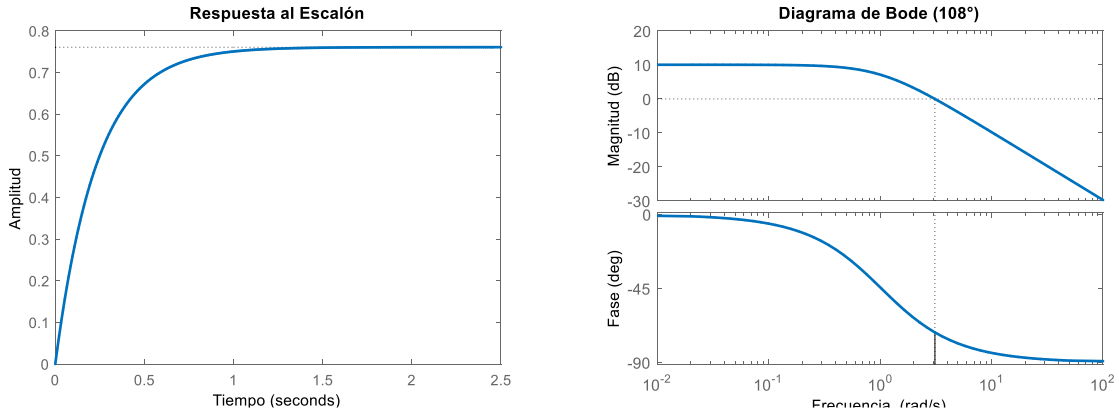


Figura 35.- Caso de estudio 1: Respuesta al escalón y diagrama de Bode del sistema sin controlador.

La respuesta del sistema se presenta en la figura 36, donde se puede observar que el error en estado estable se corrige, sin embargo, presenta un sobre pico en su estado transitorio, la respuesta del sistema es más rápida.

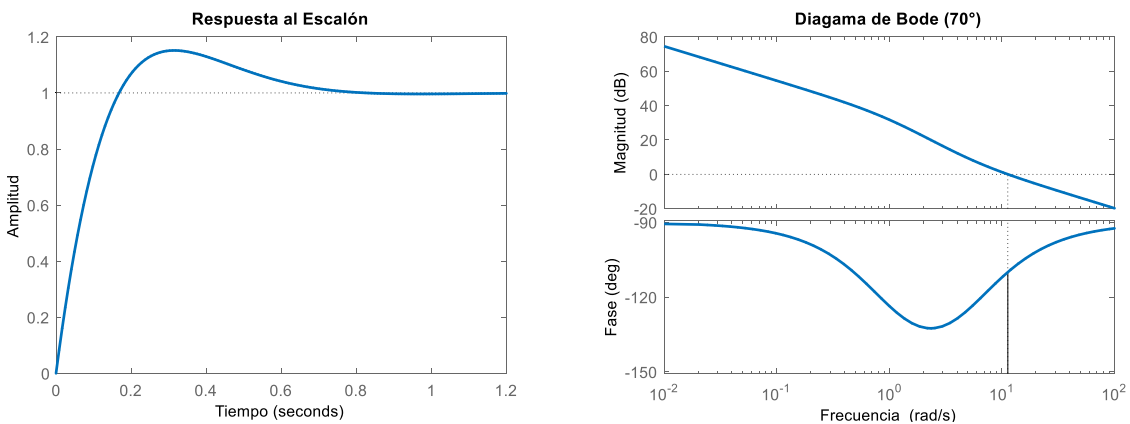


Figura 36.- Caso de estudio 1: Respuesta al escalón del sistema con el controlador PI en lazo cerrado.

El segundo controlador propuesto fue un PID, donde se propuso uno de los ceros en 1, un tiempo de respuesta de 0.1 segundos y un margen de fase de 90.5°. La función de transferencia que representa el controlador se presenta en la ecuación 87.

$$G_{PID} = \frac{0.002519s^2 + 17.29s + 17.29}{s} \quad (87)$$

La respuesta del sistema se presenta en la figura 37. Donde se puede observar que el error en estado estable se corrige y la respuesta del sistema es más rápida.

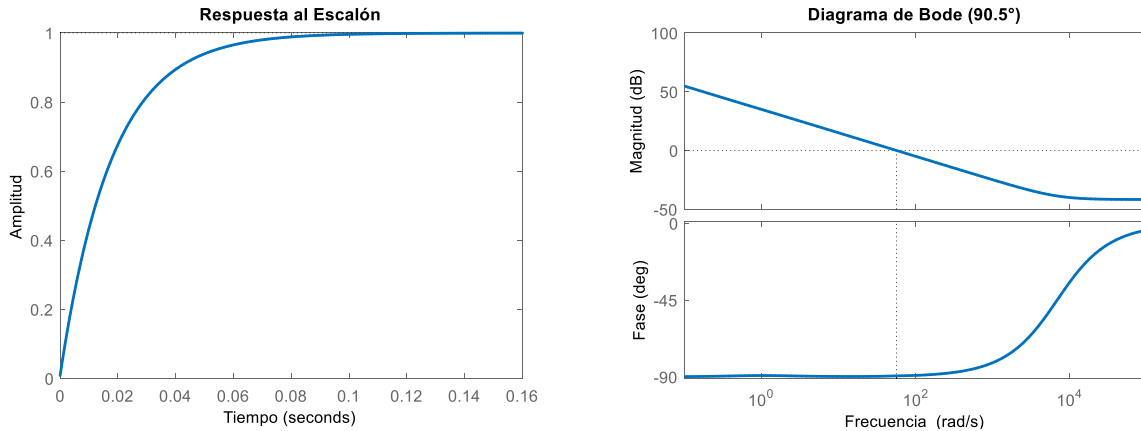


Figura 37.- Caso de estudio 1: Respuesta al escalón del sistema con el controlador PID en lazo cerrado.

La comparación de la respuesta del sistema en el primer caso de estudio sin controlador, con el controlador PI y con el controlador PID se presenta en la figura 38.

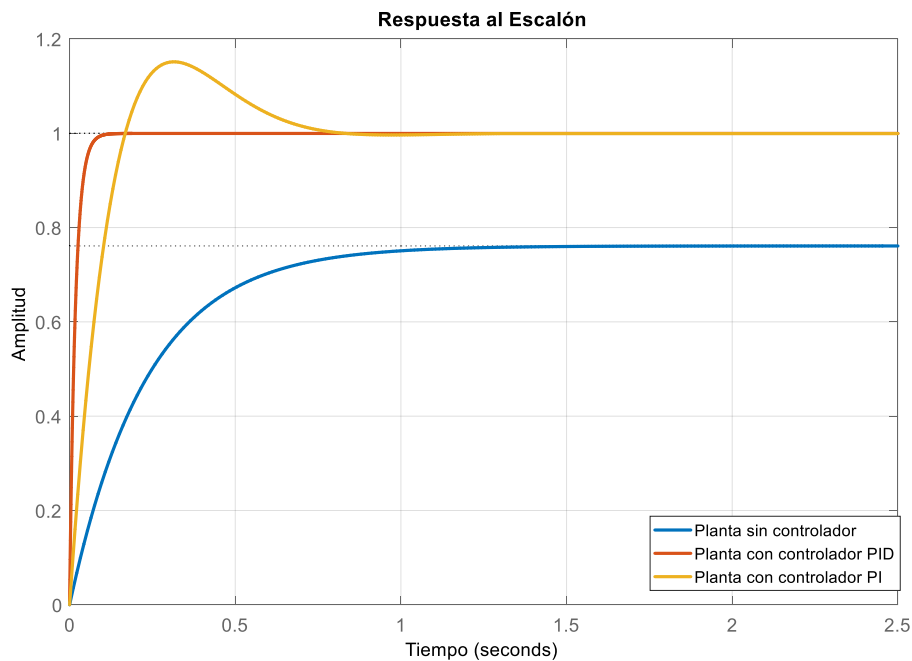


Figura 38.- Caso de estudio 1: Comparativa de la Planta con distintos controladores.

Por último, se realizó la comparación entre la respuesta al escalón del controlador diseñado a base de la etapa experimental y la que se obtuvo teóricamente, la comparación se presenta en la figura 39 (a). Se puede observar que la sintonización del controlador es muy parecida, esta comparativa se presenta en la Tabla 1. La figura 39 (b) muestra el error que se genera entre la respuesta al escalón de la planta teórica con el controlador PID y la respuesta al escalón de la planta estimada experimental.

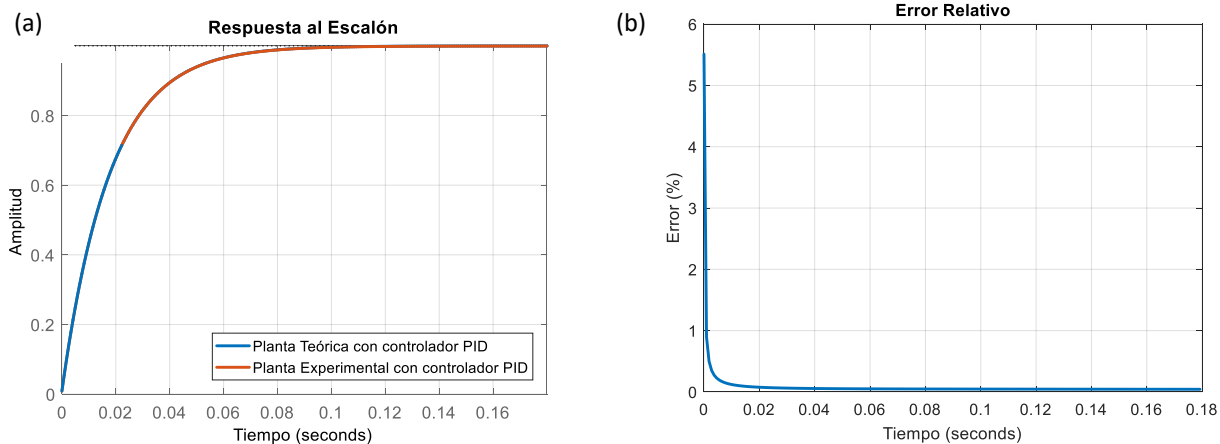


Figura 39.- Caso de estudio I: a) Comparativa entre la respuesta al escalón de la planta experimental y teórica con controlador PID, b) Grafica del error relativo de las respuestas al escalón de las plantas.

Tabla 1.- CASO ESTUDIO I: COMPARATIVA DE CONTROLADORES

	K_d	K_p	K_i
Controlador PID con Planta Teórica	0.002644	17.14	17.14
Controlador PID con Planta Experimental	0.002519	17.29	17.29

Para el caso de estudio 2 se obtuvo la función de transferencia descrita en la ecuación 88 mediante el método experimental:

$$G(s) = \frac{11.3187}{s+8.756567} \quad (88)$$

Donde la respuesta de este sistema en lazo cerrado se presenta en la figura 40.

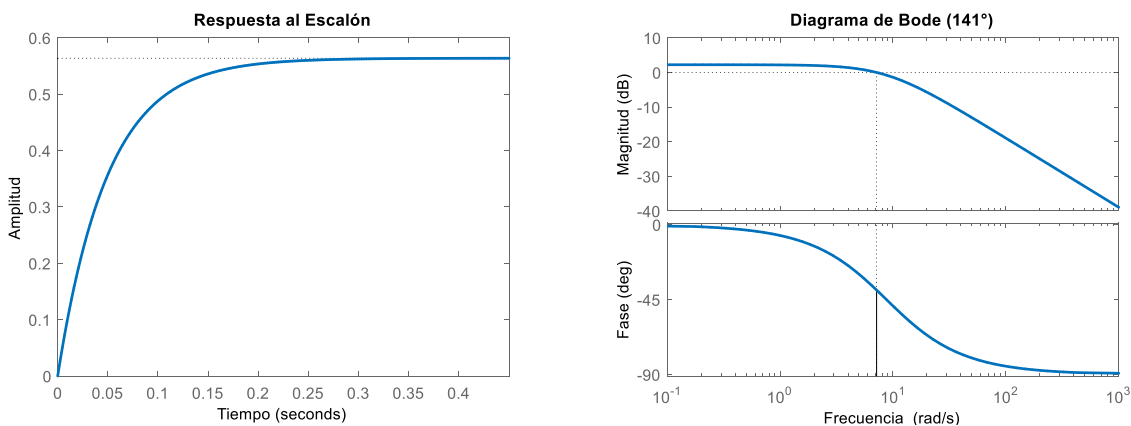


Figura 40.- Caso de estudio 2: Respuesta al escalón y diagrama de Bode del sistema sin controlador.

En la figura 39 se puede observar que el sistema presenta un error mayor en estado estable que en el caso de estudio 1, por lo tanto, no llega a la referencia. Debido a esto se diseñaron dos controladores para corregir este error. El primer controlador propuesto fue un PI, donde se propuso un tiempo de respuesta de 0.2 segundos y un margen de fase de 60°. La función de transferencia que representa el controlador se presenta en la ecuación 89.

$$G_{PI} = \frac{1.777s+54.26}{s} \quad (89)$$

La respuesta del sistema se presenta en la figura 41. Donde se puede observar que el error en estado estable se corrige, sin embargo, presenta un sobre pico en su estado transitorio.

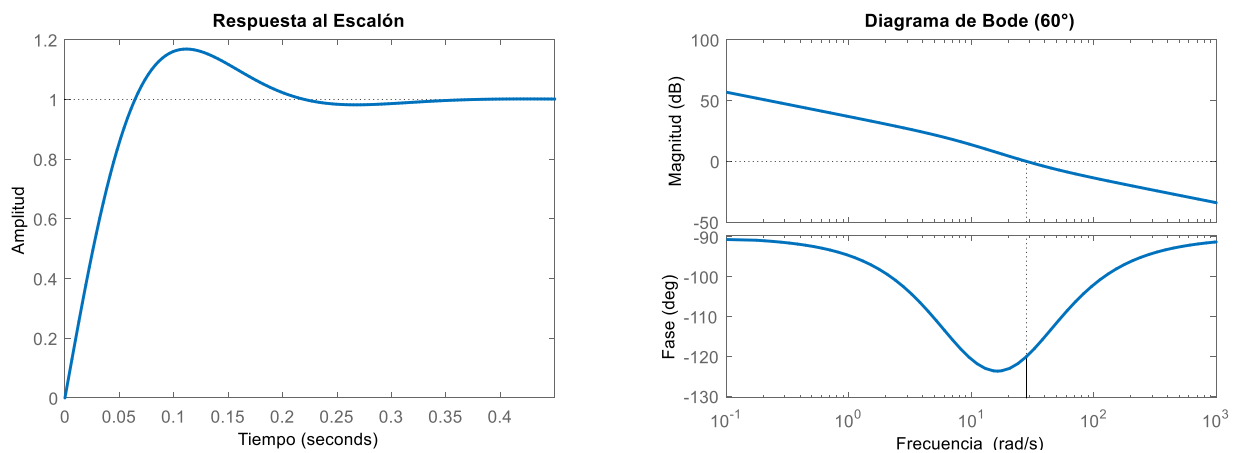


Figura 41.- Caso de estudio 2: Respuesta al escalón del sistema con el controlador PI en lazo cerrado.

El segundo controlador propuesto fue un PID, donde se propuso uno de los ceros en 8, un tiempo de respuesta de 0.01 segundos y un margen de fase de 91°. La función de transferencia que representa el controlador se presenta en la ecuación 90.

$$G_{PID} = \frac{0.001424s^2 + 49.97s + 399.6}{s} \quad (90)$$

La respuesta del sistema se presenta en la figura 42. Donde se puede observar que el error en estado estable se corrige y la respuesta del sistema es más rápida.

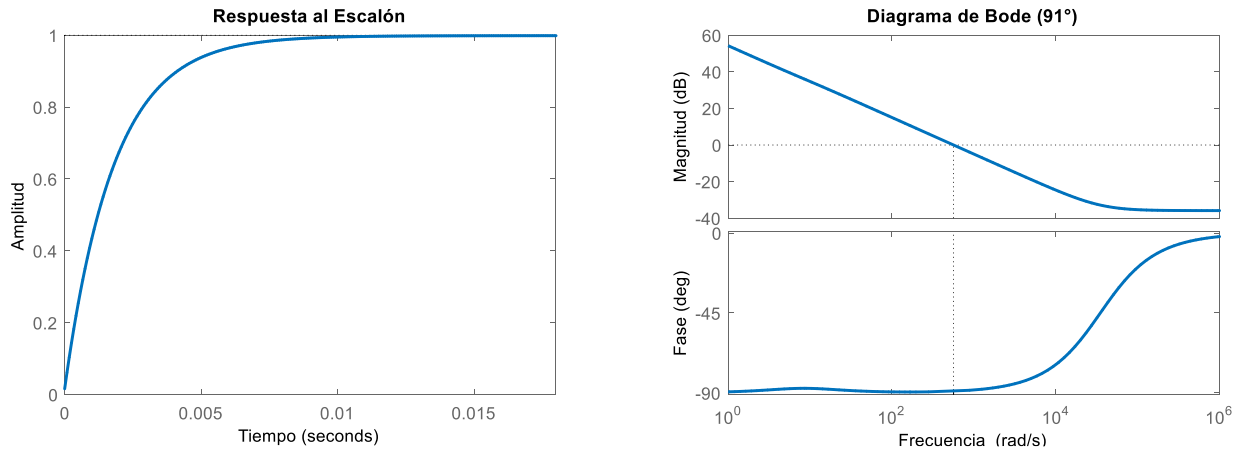


Figura 42.- Caso de estudio 2: Respuesta al escalón del sistema con el controlador PID en lazo cerrado.

La comparación de la respuesta del sistema en el segundo caso de estudio sin controlador, con el controlador PI y con el controlador PID se presenta en la figura 43.

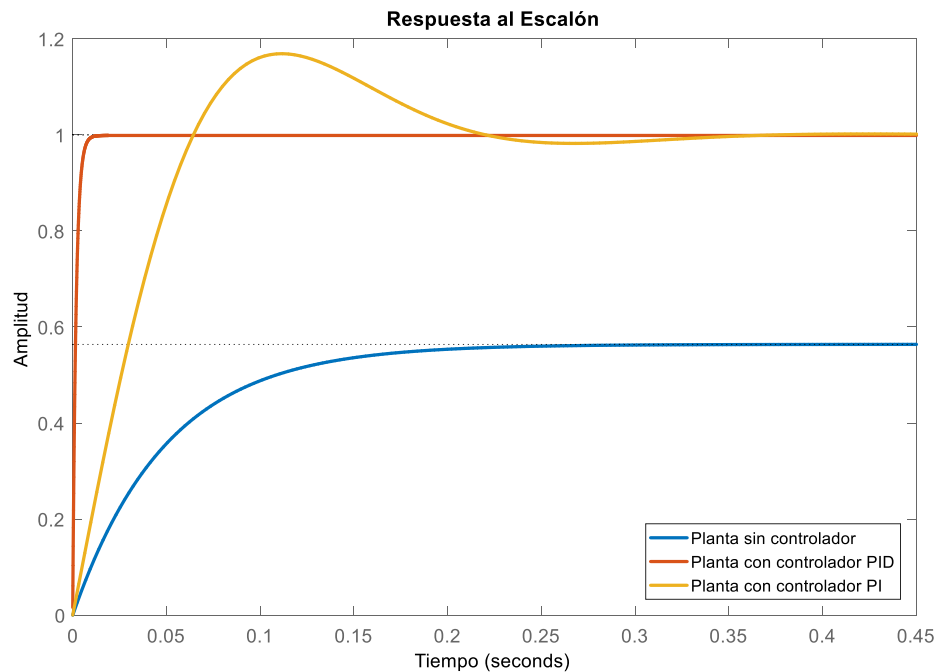


Figura 43.- Caso de estudio 2: Comparativa de la Planta con distintos controladores.

Por último, se realizó la comparación entre la respuesta al escalón del controlador diseñado a base de la etapa experimental y la que se obtuvo teóricamente, la comparación se presenta en la figura 44 (a). Se puede observar que la sintonización del controlador es muy parecida, esta comparativa se presenta en la Tabla 2. La figura 44 (b) muestra el error que se

genera entre la respuesta al escalón de la planta teórica con el controlador PID y la respuesta al escalón de la planta estimada experimental.

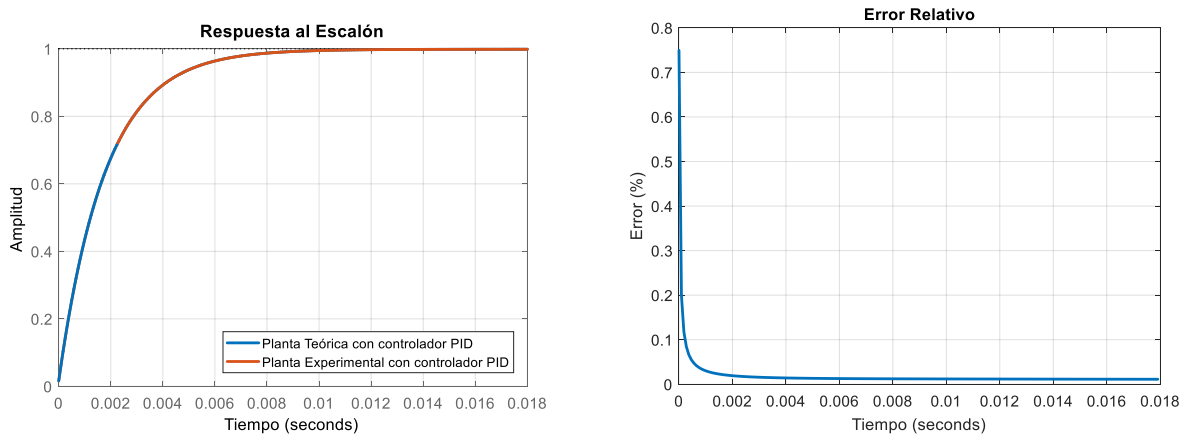


Figura 44.- Caso de estudio II: a) Comparativa entre la respuesta al escalón de la planta experimental y teórica con controlador PID, b) Grafica del error relativo de las respuestas al escalón de las plantas.

Tabla 2.- CASO ESTUDIO II: COMPARATIVA DE CONTROLADORES

	K_d	K_p	K_i
Controlador PID con Planta Teórica	0.001424	49.97	399.6
Controlador PID con Planta Experimental	0.001435	49.97	399.7

4.4 Implementación en FPGA

Afín de probar la metodología propuesta bajo condiciones de operación reales, el sistema se implementó en el FPGA de bajo costo EP2C20F484C7N, donde se encuentran embebidos los elementos que se muestran en la figura 27. Cabe señalar que todas las arquitecturas digitales embebidas en el FPGA fueron desarrolladas con la finalidad de mantener un bajo costo del sistema propuesto. Para realizar la estimación de la FT en tiempo real, el sistema realiza una iteración cada de que el DAS le envía una muestra. El total de recursos utilizados del FPGA se muestra en la Tabla 3. En base a esta tabla el sistema se puede implementar en un FPGA de bajo costo, ya que solo consume el 19% de las unidades lógicas y 29% de los multiplicadores de este dispositivo, además puede trabajar a una máxima frecuencia de operación de 36.69 MHz.

Tabla 3.- CONSUMO DE RECURSOS DEL FPGA EP2C20F484C7N

	Recursos disponibles	Recursos utilizados	Porcentaje utilizado
Elementos lógicos	18752	3573	19 %
Multiplicadores de 9 bits	52	15	29%

4.4.1 Simulaciones en VHDL

Simulación elemento de la Matriz de Correlación. – En la figura 45 se puede observar la simulación del comportamiento que presenta uno de los elementos de la matriz de correlación del algoritmo RLS, el cual va modificándose con respecto al tiempo. Se tomo una matriz de prueba solo para verificar el comportamiento del algoritmo implementado en VHDL, así como señales aleatorias para la entrada y salida del sistema.

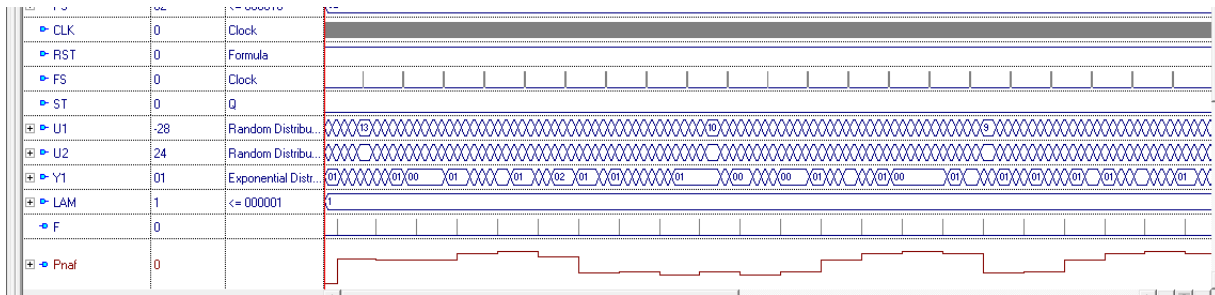


Figura 45.- Simulación en VHDL de la matriz de correlación del algoritmo RLS.

Simulación de la actualización de coeficientes. – En la figura 46 se observa el comportamiento que presenta los coeficientes del filtro adaptativo ante una entrada y salida aleatoria. Se puede observar que el valor de los coeficientes se va ajustando con respecto a la señal de entrada y salida del sistema cada periodo de muestres (FS), así como también el ajuste depende del factor de olvido.

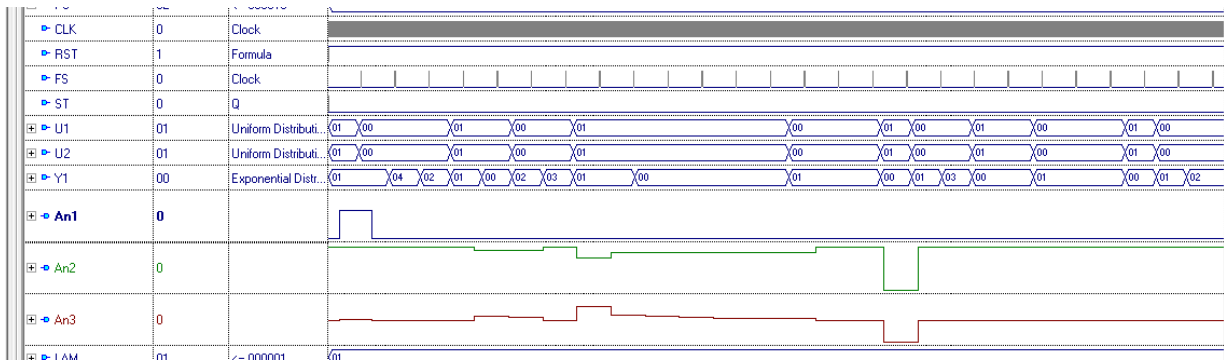


Figura 46.- Simulación en VHDL del vector de coeficientes del algoritmo RLS.

Conclusiones

Como es conocido en el área de control, la etapa de identificación es un proceso fundamental para poder realizar la sintonización de controladores digitales, por lo tanto, en este trabajo de investigación se propuso una metodología para la identificación de la función de transferencia de un sistema a partir de técnicas de procesamiento digital de señales, la cual solo requiere de la señal de entrada con la que se excita a la planta y la señal de respuesta de la planta. Dentro de la metodología, la etapa de filtrado mejora considerablemente la estimación de la función de transferencia, ya que disminuye considerablemente el ruido de cuantización de la señal de salida de la planta. Además, con el sistema propuesto se logra realizar una estimación de la función de transferencia en tiempo real, utilizando bajo consumo de recursos computacionales de un FPGA.

La validación de esta propuesta demuestra que la metodología y el sistema desarrollado es una herramienta confiable para estimar la función de transferencia de un sistema de control, ya que al aumentar el orden del filtro a 32 se pueden obtener errores absolutos menores al 1%, lo cual mejora considerablemente la estimación de la FT comparado con las propuestas donde solo utilizan el algoritmo RLS que se puede obtener un error de hasta un 5%.

A diferencia de los trabajos que realizan la estimación de la función de transferencia utilizando solo el RLS en este trabajo se propone una etapa de filtrado en la señal de salida

de la planta. Con esta etapa, se logra reducir el error de cuantización de la señal de salida de la planta con lo que se logra mejorar la estimación de la función de transferencia del sistema. Por lo tanto, si se estima la función de transferencia utilizando el RLS de forma tradicional se obtiene errores absolutos de 2.1 % para el primer caso de estudio y 5 % para el segundo caso de estudio; y con la metodología propuesta se obtienen errores absolutos de 0.8 % para el primer caso de estudio y 1 % para el segundo caso de estudio. A partir de estos errores se valida la metodología propuesta ya que la estimación de la función de transferencia mejora considerablemente. Además, a diferencia de los trabajos presentados se logró obtener una metodología que puede ser implementada en un dispositivo FPGA con bajo consumo de recursos computacionales.

En trabajos futuros se utilizará la función de transferencia estimada para realizar la sintonización e implementación de controladores digitales en tiempo real sobre sistemas de control. Además, se buscará otras técnicas de procesamiento digital de señales para mejorar más la estimación de la función de transferencia. Posteriormente, de la misma manera que se llevó a cabo en este proyecto de investigación, se pretende obtener una metodología para la sintonización de controladores, los cuales deberán ajustarse dependiendo de la función de transferencia obtenida experimentalmente. Obtenidas estas dos metodologías (Identificación y Sintonización), se fusionarán para obtener la metodología para un sistema auto sintonizable, el cual tendrá dos etapas que dependerán entre sí, la identificación experimental y la auto sintonización en línea.

Bibliografía

- [1] P. Mikell Groover, “Automation, production systems, and computer-integrated manufacturing”, Editorial Prentice-Hall, 2da edición, ISBN: 9780130889782.
- [2] K. OGATA, “Ingeniería de Control Moderno”, Editorial Prentice-Hall, 5ta edición, ISBN: 978-848-32-2660-5.
- [3] K. J. ÅSTRÖM, P. EYKHOFF, “System identification—a survey”, *Automatica*, vol. 7, no 2, pp. 123-162, 1971.
- [4] Fu, Li, P. Li, "The research survey of system identification method.", 5th International Conference in Intelligent Human-Machine Systems and Cybernetics (IHMSC), vol. 2, pp. 397-401, 2013.
- [5] S. Haykin, “Adaptive filter theory”, Editorial Prentice-Hall, 4ta edición, ISBN: 0-13-090126-1.
- [6] M. Kocur, S. Kozak, B. Dvorscak, “Design and Implementation of FPGA digital based PID controller”, In Proceedings of 15th International Carpathian Control Conference (ICCC), (Velke Karlovice, Czech Republic), pp. 233-236, 2014.
- [7] M. Aboelaze, M. G. Shehata, “Implementation of multiple PID controllers on FPGA”, In Proceedings of IEEE International Conference on Electronics, Circuits, and Systems (ICECS), pp. 446-449, 2015.
- [8] A. J. Carrillo, “Sistemas Automáticos de Control Fundamentos Básicos de Análisis y Modelado”, Editorial UNERMB, 2da edición, ISBN: 978-980-6792-12-8.
- [9] P. Van den Hof, “Closed-loop issues in system identification”, *Annual reviews in control*, vol. 22, pp. 173-186, 1998.
- [10] K. Burn, L. Maerte, C. Cox, “A Matlab toolbox for teaching modern system identification methods for industrial process control”, *International journal of mechanical engineering education*, vol. 38, no. 4, pp. 352-364, 2010.
- [11] R. H. Barker, “The pulse transfer function and its application to sampling servo systems”, In Proceedings of the IEE-Part IV: Institution Monograph, vol. 99, no. 4, pp. 302-317, 1952.
- [12] G. L. Matthaei, “Filter transfer function synthesis”, In Proceedings of the IRE, vol. 41, no. 3, pp. 377-382, 1953.
- [13] R. J. Ehret, E. F. Hochschild, J. M. Embree, E. C. Grogan, “Transfer function measuring and recording system”, *Electrical Engineering*, vol. 72, no. 9, pp. 796-796, 1953.
- [14] E. Ho, “RLC transfer function synthesis.”, *IRE Transactions on Circuit Theory*, vol. 3, no. 3, pp. 188-190, 1956.
- [15] M. Bender, F. P. Hercules, “An Experimental Method for Obtaining the Transfer Function of a Rate Gyro”, *IRE Transactions on Instrumentation*, vol. 1, no. 1, pp. 35-42, 1957.

- [16] J. G. Henderson, C. J. Pengilly, "The experimental determination of system transfer functions from normal operating data", *Journal of the British Institution of Radio Engineers*, vol. 18, no. 3, pp. 179-188, 1958.
- [17] C. Weygandt, N. Purin "Transfer-function tracking and adaptive control systems", *IRE Transactions on Automatic Control*, vol. 6, no. 2, pp. 162-166, 1961.
- [18] J. I. Nagumo, A. Noda, "A learning method for system identification", *IEEE Transactions on Automatic Control*, vol. 12, no. 3, pp. 282-287, 1967.
- [19] L. Ljung, "Consistency of the least-squares identification method", *IEEE Transactions on Automatic Control*, vol. 21, no. 5, pp. 779-781, 1976.
- [20] I. Gustavsson, L. Ljung, T. Söderström, "Identification of processes in closed loop—identifiability and accuracy aspects", *Automatica*, vol. 13, no. 1, pp. 59-72, 1977.
- [21] K. J. Hunt, "A survey of recursive identification algorithms", *Transactions of the Institute of Measurement and Control*, vol. 8, no. 5, pp. 273-278, 1986.
- [22] H. Unbehauen, G. P. Rao, "Continuous-time approaches to system identification—A survey", *Automatica*, vol. 26, no. 1, pp. 23-35, 1990.
- [23] J. Benesty, C. Paleologu, S. Ciochină, "On the Identification of Bilinear Forms with the Wiener Filter", *IEEE Signal Processing Letters*, vol. 24, no.5, pp. 653-657, 2017.
- [24] Prasad, R. Saurabh, A. P. Shrinivas, "Implementation of LMS algorithm for system identification", *International Conference on Signal and Information Processing (IConSIP)*, pp. 1-5, 2016.
- [25] Ciochină, Silviu, et al. "A family of optimized LMS-based algorithms for system identification", *24th European IEEE on Signal Processing Conference (EUSIPCO)*, pp. 1803-1807, 2016.
- [26] P. K. Dash, S. P. Sashank, E. Y. Sahoo, "Design and implementation of single order LMS based adaptive system identification on altera based cyclone II FPGA", *2014 International Conference on Communications and Signal Processing (ICCSIP)*, pp. 1605-1609, 2014.
- [27] C. Safarian, T. Ogunfunmi, W.J. Kozacky, B.K. Mohanty, "FPGA implementation of LMS-based FIR adaptive filter for real time digital signal processing applications", *In Conference on International Digital Signal Processing (DSP)*, pp. 1251-1255, 2015.
- [28] A. Deshmukh, K. Ramakrishnan, G. Ray, "Improved convergence of an adaptive identification algorithm for a large sparse system", *In 9th International Conference on Electrical and Computer Engineering (ICECE)*, pp. 34-37, 2016.
- [29] Wang, Yina, et al. "Digital acceleration controller based on recursive least squares (RLS) identification for an excretion care support robot", *International Conference on Mechatronics and Automation (ICMA)*, pp.1639-1644, 2012.
- [30] L. Morales, R. J. Romero, R. A. Osornio, G. Herrera and J. J. Santiago, "Special purpose processor for parameter identification of CNC second order servo systems on a low-cost FPGA platform", *Mechatronics*, vol. 20, no. 2, pp. 265-272, 2010.

- [31] M. Askari, J. Li, B. Samali, "A compact self-adaptive recursive least square approach for real-time structural identification with unknown inputs.", *Advances in Structural Engineering*, vol. 19, no. 7, pp. 1118-1129, 2016.
- [32] Chubich, M. Vladimir, S. Oksana Chernikova, "Adaptive fading Kalman filter with applications in identification discrete system", 13th International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE), vol. 2, pp. 385-389, 2016.
- [33] K. Bogdanski, M. C. Best, "A new structure for non-linear black-box system identification using the extended Kalman filter", In *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 0954407017692219, 2017.
- [34] Sun, Chuen-Tsai, "Rule-base structure identification in an adaptive-network-based fuzzy inference system", *IEEE Transactions on Fuzzy Systems* vol. 2, no.1, pp. 64-73, 1994.
- [35] Davanipour, Mehmoosh, M. Zekri, F. Sheikholeslam, "Fuzzy wavelet neural networks with hybrid algorithm in nonlinear system identification", *IEEE International Conference on Computer Science and Automation Engineering (CSAE)*, vol. 1, pp. 153-156, 2011.
- [36] D. Millán, "Estudio y comparativa de diferentes algoritmos adaptativos para la identificación de sistemas. Ingeniería Técnica de Telecomunicaciones", *Universidad Politécnica de Cataluña*, 2012.
- [37] M. Chuez, J. Alvarez, P. Vargas, "Implementaciones en matlab de los algoritmos adaptativos para los sistemas de antenas inteligentes", 2012.
- [38] E. B. Albertí, "Procesado digital de señales-II: Fundamentos para comunicaciones y control", *Editorial Universitat Politècnica de Catalunya*, ISBN: 84-8301-851-9.
- [39] J. G. Proakis, D. G. Manolakis, "Tratamiento digital de señales", *Pearson Educación*, Cuarta Edición, ISBN: 978-84-8322-347-5.
- [40] G. Bishop, G. Welch, "An introduction to the Kalman filter", *Proc of SIGGRAPH*, Course, 2000.
- [41] O. L. R. Jacobs, "Introduction to control theory", *Editorial Oxford: Oxford University Press*, Segunda edición, ISBN: 978-01-9856-249-8.
- [42] R. G. Brown, P. Y. Hwang, "Introduction to random signals and applied Kalman filtering", *Editorial New York: Wiley*, Tercera Edición, ISBN: 978-04-7112-839-7.
- [43] P. Eykhoff, P. M. E. M. Van der Grinten, H. Kwakernaak, B. P. Veltman, "Th.: Systems Modeling and Identification", *Survey paper in Third Congress IFAC*, London, 1966.
- [44] L. Ljung, "System identification: theory for the user", *Editorial Prentice-hall*, Segunda Edición, ISBN: 978-01-3244-053-0.
- [45] T. Söderström, P. Stoica, P. "System identification", *Prentice-Hall*, London, UK, 1989.
- [46] L. A. Zadeh, "From circuit theory to system theory" *Proceedings of the IRE*, vol. 50, no. 5, pp. 856-865, 1962.
- [47] J. D. Rairán, Y. P. Sierra, N. I. Moreno, "Implementación de dos estrategias de control para la velocidad de un motor DC", *Ingeniería e Investigación*, vol. 29, no. 2, 2009.

- [48] G. Huang, S. Lee, "PC-based PID speed control in DC motor" In International Conference on Audio, Language and Image Processing (ICALIP), pp. 400-407, 2008.
- [49] J. González, "Diseño de un controlador basado en FPGA para un robot cartesiano", Tesis de Licenciatura, Departamento de Estudios Multidisciplinarios, UG, Yuriria, Gto, 2016.