

**EL ALGORITMO ACELERADOR VERSIÓN  $\gamma$  (AR  $\gamma$ ) EN  
CONFIGURACIÓN *AFFINE* Y SU APLICACIÓN EN  
ESTRUCTURA DE CELOSÍA O *LATTICE***



**MARÍA FERNANDA RIVERA SANCLEMENTE**

Tesis de Doctorado en Ciencias de la Electrónica

Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Línea de investigación: Señales y Sistemas de Comunicaciones  
Popayán, Mayo 2021

**MARÍA FERNANDA RIVERA SANCLEMENTE**

**EL ALGORITMO ACELERADOR VERSIÓN  $\gamma$  (AR  $\gamma$ ) EN  
CONFIGURACIÓN *AFFINE* Y SU APLICACIÓN EN  
ESTRUCTURA DE CELOSÍA O *LATTICE***

Tesis presentada a la Facultad de Ingeniería  
Electrónica y Telecomunicaciones de la  
Universidad del Cauca para la obtención del Título de

Doctor en:  
Ciencias de la Electrónica.

Director  
PhD. Guefry Leider Agredo Méndez

Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Línea de investigación Señales y Sistemas de Comunicaciones  
Popayán, Mayo 2021

# Aceptación

---

---

---

# Dedicatoria

*“La suprema adquisición de la razón consiste en reconocer  
que hay una infinidad de cosas que la sobrepasan.  
Cuando no conoce esto, la razón es débil.  
Dúdese donde es debido, afirmese donde es debido,  
Empléese la sumisión donde es debido.  
Quién no hace esto, no entiende la fuerza de la razón.  
Hay quién peca contra estos tres principios,  
ya asegurándolo todo como evidente,  
por falta de habilidad en las demostraciones;  
ya dudando de todo, por falta de sumisión allí donde es debido;  
ya sometándose a todo,  
por falta de discernimiento allí donde hay lugar a éste. ...”*

*Blaise Pascal*

***A mis padres y hermana,  
A mi entorno familiar.  
A mis profesores, colegas y amigos.***

# Agradecimientos

Al Ing. Ph.D. Guefry Leider Agredo Méndez. Director de Tesis, profesor titular Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca.

Al Ing. PhD. Antonio Artés-Rodríguez. Director del grupo de Investigación de Tratamiento de Señal del Departamento de Teoría de la Señal y Comunicaciones de la Universidad Carlos III de Madrid (España) por su valioso aporte y asesoría durante la estancia-pasantía Internacional en investigación.

Al Ing. PhD. Julio Ariel Hurtado. Coordinador del Doctorado en Ciencias de la Electrónica de la Facultad de Ingeniería Electrónica y Telecomunicaciones Universidad del Cauca.

Al Ing. Francisco Pino Correa. Decano de la Facultad de Ingeniería Electrónica y Telecomunicaciones. Universidad del Cauca.

Al Centro de Posgrados Universidad del Cauca. Personal docente y administrativo.

Al Grupo de Investigación GNTT I+D. Grupo Nuevas Tecnologías en Telecomunicaciones y su director Ing. Pablo Emilio Jojoa Gomez. Adscrito a la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca.

A la Universidad Carlos III de Madrid (España), a la Facultad de Ingenierías y al Departamento de Teoría de la Señal y Comunicaciones por autorizarme la pasantía doctoral y facilitar la infraestructura necesaria para el éxito de la experiencia.

Al Hospital General Universitario Gregorio Marañón de Madrid (España) por proporcionar el material requerido del Departamento de Electrofisiología a través de la Universidad Carlos III de Madrid.

A todos aquellos, que con su aporte contribuyeron al logro de este importante proyecto para mi crecimiento personal y desarrollo profesional.

## Resumen

Para el procesamiento digital de señales se suele utilizar sistemas cuyas estructuras pueden modelar y analizar la señal digital con base en sus características o en la retroalimentación de su salida. Además de las estructuras conocidas (FIR e IIR) se tiene la estructura *Lattice* que tiene como ventajas un número reducido de coeficientes permitiendo que grandes bloques de datos puedan ser modelados en tiempo real, repetición de secciones idénticas logrando cierta simplificación del proceso (estructuras en cascada), coeficientes con magnitudes menores a la unidad, prueba de estabilidad por inspección, buenas propiedades de redondeo numérico y es particularmente útil para el filtrado adaptable dado que la solución recursiva de la estimación de los mínimos cuadrados produce naturalmente un filtro de estructura *Lattice* (Proakis y Manolakis, 2007). Así mismo, la estructura *Lattice* ortogonaliza la señal de entrada en una base de etapa a etapa, permitiendo capacidades de rápida convergencia y seguimiento. Este tipo de estructura utiliza los denominados coeficientes de reflexión y el análisis de predicción hacia adelante y hacia atrás para analizar la señal que pasa a través de él.

En las técnicas de estimación adaptables se modifican los coeficientes del filtro (sistema) de acuerdo con los datos que se actualizan cada vez que arriba una nueva muestra. En el caso de la estimación recursiva de un filtro *Lattice* son los coeficientes de reflexión los que se actualizan produciendo nuevos errores de predicción por cada orden del filtro. Es necesario cambiar cada coeficiente del filtro para cada nueva muestra en las aplicaciones donde la rápida convergencia o el seguimiento (*tracking*) de señales son altamente variables. (Cowan y Grant, 1986) Así entonces la estructura *Lattice* está motivada por la teoría del modelo autorregresivo de la señal.

Ahora bien, el algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* consiste en una suma ponderada de las salidas de dos filtros adaptativos sometidos a una misma entrada cuyo factor de ponderación es un parámetro calculado mediante un mecanismo adaptativo con el propósito de lograr un mejor rendimiento. La configuración *Affine* ofrece una buena adaptabilidad mejorando características como

velocidad, reducción del error de medida final, una buena capacidad de seguimiento (*tracking*) y buenos resultados en desajuste. El filtro implementado en esta configuración alivia el compromiso entre la velocidad de convergencia y el error residual inherente al filtro. (Rivera, 2013). Con base en lo anterior, se propone en el presente estudio, la implementación del algoritmo acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* sobre la estructura *Lattice*, incidiendo específicamente en los coeficientes de reflexión buscando potenciar su adaptabilidad y optimizar el rendimiento de sus propiedades de velocidad, reducción del error de medida final, capacidad de seguimiento (*tracking*) y mínimo desajuste.

Para efectos de obtener una implementación del algoritmo en tiempo real que permita un análisis cuali-cuantitativo de los algoritmos mencionados y, en consecuencia, evaluar su desempeño sobre señales reales, se lleva a cabo, una experiencia en la cual se hace aplicación de la estructura *Affine-Lattice* para la determinación de zonas de *frecuencia dominante* causantes de fibrilación auricular en situaciones de patologías cardíacas. La experiencia tiene como fundamento el hecho de que, la estructura *Affine-Lattice* minimiza el error de predicción, lo que se traduce en la identificación de las zonas con *frecuencia dominante* como causantes de fibrilación cardíaca. Se utilizan con tal propósito los registros de electrogramas intracardiaco obtenidos del servicio de electrofisiología cardíaca del Hospital General Universitario de Madrid (España), registros que permiten el análisis de fibrilación auricular obtenidas con un catéter intracardiaco. El análisis se apoya en el diseño de un programa desarrollado en MATLAB cuyas características básicas consisten en la aplicación de la estructura *Affine-Lattice*, la utilización de un filtro adaptativo y el análisis con base en la Teoría de Causalidad de Granger.

Con base en lo anterior, la formulación del problema de investigación responde a la pregunta: Cuál es el efecto sobre la respuesta de un sistema *Lattice* adaptativo cuando sobre ésta estructura se utiliza una configuración *Affine* soportada en el algoritmo adaptativo  $AR\ \gamma$ ? Como Hipótesis de solución se establece: La implementación del algoritmo acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* sobre la estructura *Lattice* optimiza la adaptabilidad y el rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento. Los objetivos para lograr dicho propósito se formulan como: 1. Optimizar las propiedades que ofrece el algoritmo acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* al

adecuarlo a la estructura de Celosía o *Lattice*, 2. Analizar el efecto de los parámetros de ajuste en el comportamiento de la estructura *Lattice-Affine* soportada en el algoritmo  $AR_{\gamma}$ , 3. Especificar las ecuaciones de diseño de la estructura adaptativa *Lattice* haciendo uso de la configuración *Affine* con el algoritmo  $AR_{\gamma}$  como soporte, 4. Comparar el comportamiento del Algoritmo Acelerador Versión  $\gamma$  ( $AR_{\gamma}$ ) en configuración *Affine* sobre la estructura *Lattice* con sistemas *Lattice* adaptativos convencionales, con base en, un análisis cualitativo y cuantitativo de los algoritmos implementados. La metodología utilizada en el estudio obedece al proceso de investigación tecnológica en ciencias de la Ingeniería, la cual se orienta a generar conocimientos aplicables a la producción de bienes y servicios. Contribuye específicamente a la solución de un problema mediado por una hipótesis de solución, buscando respuesta a los interrogantes, (Problema de Investigación) operacionalizando los objetivos propuestos, mediante la simulación soportada en la herramienta matemática computacional MATLAB, lo que se traduce en la implementación del algoritmo acelerador versión  $\gamma$  ( $AR_{\gamma}$ ) en configuración *Affine* sobre la estructura *Lattice*, incidiendo específicamente en los coeficientes de reflexión buscando potenciar su adaptabilidad y optimizar el rendimiento de sus propiedades de velocidad, reducción del error de medida final, capacidad de seguimiento (*tracking*) y mínimo desajuste.

Los resultados se obtienen, con base en, el desarrollo de las simulaciones requeridas para el logro de los objetivos en lenguaje de programación de alto nivel, MATLAB con la configuración de identificación del sistema y que permitieron obtener los registros en los que se evidencia el comportamiento de los filtros LMS y NLMS al implementarlos en la estructura *Lattice* respecto de la señal de error, los coeficientes propios del filtro y su respuesta frente al sistema de identificación, el comportamiento de los coeficientes de reflexión y predictores de adelanto y atraso; como también respecto al error cuadrático medio. Así mismo, los resultados permiten evidenciar el comportamiento del filtro  $AR_{\gamma}$  y su implementación en la arquitectura *Lattice* con respecto a la señal de error, los coeficientes propios del filtro y su respuesta frente al sistema de identificación; comportamiento de los coeficientes de reflexión y predictores de adelanto y atraso como también el error cuadrático medio. El nuevo producto, corresponde a un Filtro adaptativo para procesamiento digital de señales con optimización de la adaptabilidad y rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento (*tracking*). Se concluye por lo tanto que,

el algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine* sobre una estructura *Lattice* da como resultado un eficiente filtro adaptativo en el procesado digital de señales. Un modelo de filtro adaptativo en el que características como velocidad de convergencia, mínimo error de predicción y capacidad de seguimiento (*tracking*) se han optimizado. Al relacionar los filtros LMS, NLMS y acelerador AR  $\gamma$  en estructura *Lattice* se observa que al aumentar el número de etapas de la estructura *Lattice*, al igual que el orden del filtro, se reduce el error cuadrático medio para el filtro AR  $\gamma$  en arquitectura *Lattice*. La utilización de la Teoría de Causalidad de Granger en la aplicación del filtro Acelerador en configuración *Affine* a la estructura *Lattice*, permite concluir que la optimización de filtros adaptativos y específicamente la adecuación del Algoritmo Acelerador Regresivo en combinación *Affine* sobre una estructura *Lattice* resulta de gran utilidad en eventos en los cuales el factor de predicción permite la toma de decisiones en el procesado digital de señales y en otras situaciones, por ejemplo, de tipo biomédico (en la determinación de focos de fibrilación auricular), en eventos de tipo económico (predicción de índices económicos) y en señales geofísicas, entre otros.

**Palabras clave.** Procesamiento digital de señales. Filtros adaptativos. Configuración *Affine*. Estructura en Celosía o *Lattice*. Algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine*.

# Abstract

For digital signal processing, systems are often used whose structures can model and analyze the digital signal based on their characteristics or on the feedback of their output. In addition to the known structures (FIR and IIR) we have the lattice structure that has as advantages a reduced number of coefficients allowing large blocks of data to be modeled in real time, repeating identical sections achieving some simplification of the process (cascading structures), coefficients with magnitudes less than unity, stability test by inspection, good numerical rounding properties and is particularly useful for adaptive filtering since the recursive solution of least squares estimation naturally produces a Lattice structure filter (Proakis and Manolakis, 2007). Likewise, the Lattice structure orthogonalizes the input signal on a stage-by-stage basis, allowing rapid convergence and tracking capabilities. This type of structure uses the so-called reflection coefficients and forward and backward prediction analysis to analyze the signal that passes through it.

In the adaptive estimation techniques, the filter (system) coefficients are modified according to the data that is updated each time a new sample arrives. In the case of the recursive estimation of a Lattice filter, the reflection coefficients are updated, producing new prediction errors for each order of the filter. It is necessary to change each filter coefficient for each new sample in applications where fast convergence or tracking of signals are highly variable. (Cowan and Grant, 1986) Thus, the Lattice structure is motivated by the theory of the autoregressive model of the signal.

However, the algorithm accelerator version  $\gamma$  (AR  $\gamma$ ) in Affine configuration consists of a weighted sum of the outputs of two adaptive filters subjected to the same input whose weighting factor is a parameter calculated by an adaptive mechanism in order to achieve better performance. The Affine configuration offers a good adaptability improving characteristics such as speed, reduction of the final measurement error, a

good tracking ability and good results in mismatch. The filter implemented in this configuration relieves the compromise between the convergence speed and the residual error inherent in the filter. (Rivera, 2013). Based on the foregoing, we propose the implementation of the acceleration algorithm version  $\gamma$  ( $AR \gamma$ ) in Affine configuration on the Lattice structure, specifically focusing on the reflection coefficients in order to enhance their adaptability and optimize the performance of their speed properties. , reduction of the final measurement error, tracking capacity and minimum mismatch.

In order to obtain an implementation of the algorithm in real time that allows a qualitative analysis of the aforementioned algorithms and, consequently, to evaluate its performance on real signals, an experience is carried out in which the structure is applied. Affine-Lattice for the determination of dominant frequency zones causing atrial fibrillation in situations of cardiac pathologies. The experience is based on the fact that the Affine-Lattice structure minimizes the prediction error, which translates into the identification of areas with dominant frequency as a cause of cardiac fibrillation. The records of intracardiac electrograms obtained from the cardiac electrophysiology service of the General University Hospital of Madrid (Spain) are used for this purpose, records that allow the analysis of atrial fibrillation obtained with an intracardiac catheter. The analysis is based on the design of a program developed in MATLAB whose basic characteristics consist in the application of the Affine-Lattice structure, the use of an adaptive filter and the analysis based on Granger's Theory of Causality.

Based on the above, the formulation of the research problem answers the question: What is the effect on the response of an adaptive Lattice system when an Affine configuration supported in the adaptive algorithm  $AR \gamma$  is used on this structure? As solution hypothesis is established: The implementation of the algorithm accelerator version  $\gamma$  ( $AR \gamma$ ) in configuration Affine on the structure Lattice optimizes the adaptability and performance of its properties of speed of convergence and tracking

capacity. The objectives to achieve this purpose are formulated as: 1. Optimize the properties offered by the acceleration algorithm version  $\gamma$  ( $AR \gamma$ ) in Affine configuration when adapting it to the Lattice or Lattice structure, 2. Analyze the effect of the adjustment parameters on the behavior of the structure Lattice-Affine supported in the algorithm  $AR \gamma$ , 3. Specify the design equations of the Lattice adaptive structure making use of the configuration Affine with the algorithm  $AR \gamma$  as support, 4. Compare the behavior of the Algorithm Accelerator Version  $\gamma$  ( $AR \gamma$ ) in Affine configuration on Lattice structure with conventional adaptive lattice systems based on a qualitative and quantitative analysis of the algorithms implemented. The methodology used in the study is due to the process of technological research in engineering sciences, which is geared towards generating applicable knowledge to the production of goods and services. Contributes specifically to the solution of a problem mediated by a solution hypothesis, seeking answers to the questions, (Research Problem) operationalizing the proposed objectives, through the simulation supported in the mathematical computational tool MATLAB, which translates into the implementation of the algorithm accelerator version  $\gamma$  ( $AR \gamma$ ) in configuration Affine on the structure Lattice, specifically focusing on the coefficients of reflection seeking to enhance its adaptability and optimize the performance of its speed properties, reduction of the final measurement error, tracking capacity and minimum maladjustment.

The results are obtained, based on the development of the simulations required for the achievement of the objectives in programming language of high level, MATLAB with the configuration of system identification and that allowed to obtain the records in which the behavior is evidenced of the LMS and NLMS filters when implementing them in the Lattice structure with respect to the error signal, the coefficients of the filter and its response to the identification system, the behavior of the reflection coefficients and predictors of advancement and delay; as also with respect to the mean square error. Likewise, the results show the behavior of the AR filter and its implementation in the Lattice architecture with respect to the error signal, the filter's own coefficients and its response to the identification system; behavior of the

reflection coefficients and predictors of advancement and delay as well as the mean square error. The new product corresponds to an adaptive filter for digital signal processing with optimization of the adaptability and performance of its properties of convergence speed and tracking capacity. It is therefore concluded that the algorithm Accelerator version  $\gamma$  (AR  $\gamma$ ) in combination with Affine on a Lattice structure results in an efficient adaptive filter in the digital signal processing. An adaptive filter model in which features such as convergence speed, minimum prediction error and tracking capacity have been optimized. When relating the LMS, NLMS and AR  $\gamma$  accelerator filters in the Lattice structure, it is observed that by increasing the number of stages of the Lattice structure, as well as the filter order, the mean square error for the AR  $\gamma$  filter in the Lattice architecture is reduced. The use of Granger's Theory of Causality in the application of the Accelerator filter in the Affine configuration to the Lattice structure, allows us to conclude that the optimization of adaptive filters and specifically the adaptation of the Regressive Accelerator Algorithm in combination with Affine on a Lattice structure is very useful in events in which the prediction factor allows decision-making in the digital processing of signals and in other situations, for example, biomedical type (in the determination of foci of atrial fibrillation), in economic events (prediction of economic indexes) and in geophysical signals, among others.

**Keywords:** Feedback filter and digital signal processing. Adaptive filters. *Affine* combination. Algorithm accelerator version  $\gamma$  (AR  $\gamma$ ) in *Affine* combination.

# CONTENIDO

	Pág.
<b>Dedicatoria.</b> .....	IV
<b>Agradecimientos.</b> .....	V
<b>Resumen.</b> .....	VII
<b>Abstract.</b> .....	XI
<b>Lista de Tablas.</b> .....	XVIII
<b>Lista de Figuras.</b> .....	XIX
<b>Lista de Siglas y Abreviaturas.</b> .....	XXII
<b>Lista de Símbolos.</b> .....	XXIII
<b>Glosario.</b> .....	XXIV
<b>CAPÍTULO 1. Introducción.</b> .....	27
1.1. Planteamiento del problema, Justificación-Motivación. ....	27
1.2. Estado del arte. ....	29
1.3. Metodología. ....	33
1.3.1. Hipótesis de solución. ....	33
1.3.2. Objetivos. ....	33
1.3.2.1. General. ....	33
1.3.2.2. Específicos. ....	34
1.3.3. Ejecución-Simulación. ....	34
1.3.4. Producto. ....	34
1.4. Contribución de la tesis. ....	36
1.5. Organización de la tesis. ....	37
<b>CAPÍTULO 2. Características, usos y aplicaciones de la estructura en Celosía o <i>Lattice</i>.</b> .....	39
2.1. Características de la estructura en Celosía o <i>Lattice</i> . ....	40
2.1.1. Predictor lineal directo o de adelanto. ....	47
2.1.2. Predictor lineal inverso o de atraso. ....	49
2.2. Conversión de coeficientes de estructura <i>Lattice</i> a coeficientes de un	53

filtro FIR. ....	54
2.3. Conversión de los coeficientes de un filtro FIR a los coeficientes <i>Lattice</i> .	54
2.4. Usos, aplicaciones, ventajas y desventajas de la estructura <i>Lattice</i> .	55

**CAPÍTULO 3. El algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine* sobre la estructura *Lattice* y su relación con otros filtros adaptativos.** 57

3.1 Características del algoritmo Acelerador Regresivo versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> .	57
3.2 Adecuación del Algoritmo Acelerador Versión $\gamma$ (AR $\gamma$ ) en combinación <i>Affine</i> a la estructura <i>Lattice</i> respecto a otros filtros adaptativos.	61
3.3 Adaptabilidad de los coeficientes de reflexión de la estructura <i>lattice</i> en la implementación de un filtro adaptativo.	62
3.4 Aplicación de la estructura <i>lattice</i> al filtro AR $\gamma$ y <i>Affine</i> AR $\gamma$ .	63
3.4.1 Filtro AR $\gamma$ - <i>Lattice</i> .	63
3.4.2 Filtro <i>Affine-Lattice</i> AR $\gamma$ .	68
3.5 Aplicación de la estructura <i>lattice</i> a los filtros LMS y NLMS.	72
3.5.1 Filtro LMS- <i>Lattice</i> .	72
3.5.2 Filtro LMS <i>Affine-Lattice</i> .	76
3.5.3 Filtro NLMS <i>Lattice</i> .	79
3.5.4 Filtro NLMS <i>Affine-Lattice</i> .	82
3.6 Análisis de <i>Tracking</i> .	87
3.7 Funcionamiento del Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en estructura <i>Affine-Lattice</i> .	91

<b>CAPÍTULO 4. Adecuación del algoritmo Acelerador versión <math>\gamma</math> (AR <math>\gamma</math>) a la estructura <i>Affine-Lattice</i> y determinación de focos de fibrilación cardíaca.</b>	103
4.1 Teoría de Causalidad de Granger. (Modelo autorregresivo de la señal).	104
4.1.1 Modelo de retroalimentación. ....	106
4.1.1.1 Causalidad. ....	106
4.1.1.2 Prueba de causalidad. ....	108
4.2 Determinación de focos de fibrilación cardíaca. ....	110
4.2.1 Ritmos Irregulares. ....	113
4.2.1.1 Marcapasos migratorio. ....	113
4.2.1.2 Taquicardia auricular multifocal. ....	113
4.2.1.3 Fibrilación auricular. ....	113
4.3 Fibrilación auricular y frecuencias dominantes. ....	113
4.4 Registros de electrogramas con catéter. ....	114
4.5 Simulación. ....	119
<b>CAPITULO 5 Conclusiones y trabajo futuro.</b>	124
5.1 Conclusiones. ....	124
5.2 Trabajo Futuro. ....	125
<b>Anexos</b>	127
A.1 Pseudocódigo Algoritmo Acelerador Versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> . ....	127
A.2 Pseudocódigo Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en estructura <i>Affine-Lattice</i> . ....	131
A.3 Pseudocódigo Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en estructura <i>Affine-Lattice</i> y prueba de causalidad de Granger. ....	153
<b>Bibliografía.</b> ....	155

## Lista de Tablas

	<b>Pág.</b>
1.1 Metodología	35
2.2 Propiedades y otras características comparativas de la estructura <i>Lattice</i> .	56
3.1 Error Cuadrático Medio para diferentes valores de $\zeta$ del filtro AR $\gamma$ aplicando estructura <i>lattice</i> y <i>Affine-Lattice</i> .	88
3.2 Error Cuadrático Medio para diferentes valores de $\zeta$ del filtro LMS aplicando estructura <i>Lattice</i> y <i>Affine-Lattice</i> .	89
3.3 Error Cuadrático Medio para diferentes valores de $\zeta$ del filtro NLMS aplicando estructura <i>Lattice</i> y <i>Affine-Lattice</i> .	89
4.1 Error Cuadrático medio del Registro de las señales del corazón al aplicar el filtro AR $\gamma$ a la combinación <i>Affine-Lattice</i> y la estructura <i>lattice</i>	120
4.2 Registro de las señales del corazón al aplicar los filtros AR $\gamma$ a la combinación <i>Affine-Lattice</i> y test de Causalidad de Granger.	121

## Lista de Figuras

		Pág.
2.1	Estructura <i>Lattice</i> de $M - 1$ etapas.	40
2.2	Estructura <i>Lattice</i> de filtro FIR de primer orden.	41
2.3	Forma directa del filtro FIR de predicción.	42
2.4	Estructura <i>Lattice</i> de filtro FIR de segundo orden.	43
2.5	Predictor de adelanto.	49
2.6	Filtro de error de predicción.	49
3.1	Diagrama de flujo Algoritmo Acelerador Versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> .	60
3.2	Combinación <i>Affine</i> y Estructura <i>Lattice</i> .	61
3.3	Estructura <i>Lattice</i> y filtro adaptativo.	62
3.4	Comportamiento del filtro AR $\gamma$ y la estructura <i>lattice</i> . a) señal de error, b) coeficientes propios del filtro y c) coeficientes de reflexión.	64
3.5	Señal de predicción de adelanto (azul) y atraso (roja) del filtro AR $\gamma$ al aplicar la estructura <i>lattice</i> .	65
3.6	Señales de Error Filtros AR $\gamma$ (azul) y AR $\gamma$ - <i>Lattice</i> (rojo).	66
3.7	Convergencia de los filtros AR $\gamma$ (azul) y AR $\gamma$ - <i>Lattice</i> (rojo).	67
3.8	Comportamiento del filtro AR $\gamma$ y la estructura <i>lattice</i> . a) señal de error, b) coeficientes propios del filtro y c) coeficientes de reflexión.	69
3.9	Señales de Error Filtros AR $\gamma$ (azul), AR $\gamma$ - <i>Lattice</i> (rojo) y AR $\gamma$ - <i>Affine Lattice</i> (verde).	70
3.10	Convergencia de los filtros AR $\gamma$ (azul), AR $\gamma$ - <i>Lattice</i> (rojo) y AR $\gamma$ - <i>Affine Lattice</i> (verde).	71
3.11	Filtro LMS- <i>Lattice</i> . a) error LMS- <i>Lattice</i> , b) coeficientes filtro LMS- <i>lattice</i> y c) coeficientes de reflexión LMS- <i>Lattice</i> .	74
3.12	Error Filtro LMS (rojo) y error filtro LMS- <i>Lattice</i> (azul).	75
3.13	Convergencia filtro LMS (azul) y LMS- <i>Lattice</i> (rojo).	76
3.14	. Filtro LMS <i>Affine-Lattice</i> . a) error LMS <i>Affine-Lattice</i> , b) Variable combinación <i>Affine</i> , c) coeficientes de reflexión filtro LMS <i>Affine-lattice</i> y d) coeficientes LMS <i>Affine-Lattice</i> .	77
3.15	Error cuadrático medio del filtro LMS (azul) al aplicar la estructura <i>lattice</i> (rojo) y finalmente aplicar la combinación <i>Affine</i> y estructura <i>lattice</i> (azul claro).	78
3.16	Filtro NLMS- <i>Lattice</i> . a) error NLMS- <i>Lattice</i> , b) coeficientes filtro NLMS- <i>lattice</i> y c) coeficientes de reflexión NLMS- <i>Lattice</i> .	79
3.17	Error NLMS (azul) y Error NLMS- <i>Lattice</i> (rojo).	80
3.18	Convergencia filtro NLMS (azul) y NLMS- <i>Lattice</i> (rojo).	81

3.19	Filtro NLMS <i>Affine-Lattice</i> . a) error NLMS <i>Affine-Lattice</i> , b) Predictores de adelanto y atraso, c) Coeficientes filtro NLMS <i>Affine-lattice</i> y d) coeficientes de reflexión NLMS <i>Affine-Lattice</i> .	82
3.20	Error NLMS (azul), error NLMS <i>Lattice</i> (rojo) y Error NLMS <i>Affine-Lattice</i> (verde).	83
3.21	Convergencia de los filtros NLMS (azul), NLMS- <i>Lattice</i> (rojo) y NLMS <i>Affine-Lattice</i> (verde).	84
3.22	Convergencia de los filtros LMS <i>Affine-Lattice</i> (azul), AR $\gamma$ <i>Affine-Lattice</i> (rojo) y NLMS <i>Affine-Lattice</i> (verde).	85
3.23	Convergencia de los filtros LMS <i>Lattice</i> (rojo), NLMS <i>Lattice</i> (cian) y AR $\gamma$ <i>Lattice</i> (magenta).	86
3.24	Modelo del Proceso de Markov y sistema de identificación.	87
3.25	Algoritmo AR $\gamma$ en combinación <i>Affine</i> con estructura <i>Lattice</i> y proceso de Markov.	90
3.26	Algoritmo LMS en combinación <i>Affine</i> con estructura <i>Lattice</i> y proceso de Markov.	90
3.27	Algoritmo NLMS en combinación <i>Affine</i> con estructura <i>Lattice</i> y proceso de Markov.	91
3.28	Diagrama de flujo filtros básicos, estructura <i>Lattice</i> y estructura <i>Affine-Lattice</i> .	92
3.29	Diagrama de flujo filtros LMS <i>Lattice</i> .	94
3.30	Diagrama de flujo filtros NLMS <i>Lattice</i> .	95
3.31	Diagrama de flujo filtros AR $\gamma$ <i>Lattice</i> .	96
3.32	Diagrama de flujo filtro LMS <i>Affine-Lattice</i> .	98
3.33	Diagrama de flujo filtro NLMS <i>Affine-Lattice</i> .	100
3.34	Diagrama de flujo filtro AR $\gamma$ <i>Affine-Lattice</i> .	102
4.1	Diagrama causal de un proceso VAR <sub>5</sub> (1).	105
4.2	Partes Principales de un corazón humano.	110
4.3	Vías de conducción eléctrica del corazón.	111
4.4	Onda sinusal del corazón con sus componentes principales, onda P, complejo QRS y onda T.	112
4.5	a) Catéter circular en vena pulmonar junto a sonda de ablación. b) Señales del Lasso™ en fibrilación auricular.	115
4.6	Antro Inferior Izquierda.	116
4.7	Antro Superior Izquierda.	116
4.8	Antro Vena Pulmonar Inferior Derecha (VPID).	116

4.9	Cara Lateral Alta.	116
4.10	Cara Lateral Baja.	116
4.11	Cara Lateral entre Venas.	116
4.12	Cara Posterior.	117
4.13	Cara Posterior 2.	117
4.14	Cara Posterior 3.	117
4.15	Cara Posterior 4.	117
4.16	Lateral Orejuela.	117
4.17	SEPTO o Tabique.	117
4.18	Suelo 2.	118
4.19	Techo 2.	118
4.20	Techo anterior.	118
4.21	Techo Derecho.	118
4.22	Techo Izquierdo.	118
4.23	(VPID) Vena Pulmonar Inferior Derecha.	118
4.24	(VPIL 2) Vena Pulmonar Inferior Izquierda.	119
4.25	(VPSD) Vena Pulmonar Superior Derecha.	119
4.26	Diagrama de flujo filtro AR $\gamma$ <i>Affine-Lattice</i> y Teoría de causalidad.	123

## Lista de Siglas y abreviaturas

ATC.	( <i>Air Traffic Control</i> ) Control de Tráfico Aéreo
AR $\gamma$	Algoritmo Acelerador Regresivo versión $\gamma$ (AR $\gamma$ )
ALNF	( <i>Adaptative Lattice notch filter</i> ) Filtro notch <i>Lattice</i> adaptativo
ALRE	( <i>Adaptative Lattice-type respiratory rate estimator</i> ) Frecuencia respiratoria estimada
ALLMS	<i>Affine-Lattice</i> LMS
ALNLMS	<i>Affine-Lattice</i> NLMS
ALAR $\gamma$	<i>Affine-Lattice</i> AR $\gamma$
ANC	Cancelación del ruido de banda estrecha
A-V	Aurículo-ventricular
ECG	Electrocardiograma
EEG	Electro-encefalograma
EMG	Electromiogramas
FA	Fibrilación auricular
FIR	( <i>Finite Impulse Response</i> ) Respuesta finita al impulso
FPGA	( <i>Field Programmable Gate Array</i> )
HGGM	Hospital General Universitario Gregorio Marañón de Madrid
IIR	( <i>Infinite Impulse Response</i> ) Respuesta infinita al impulso
LANF	Filtro adaptativo <i>Lattice</i> de orden superior
LMS	( <i>Least Mean Squared</i> ) Filtro de Mínimos Cuadrados
LPEF	( <i>Linear prediction error filter</i> ) Filtro de error de predicción lineal
MCO	Mínimos cuadrados ordinarios
MIMO	( <i>Múltiple In-put - Múltiple Out-put</i> ), múltiple entrada y múltiple salida
MSE	( <i>Mean Squared Error</i> ), Error Cuadrático Medio
NLMS	( <i>Normalized least mean squared</i> )
PPG	( <i>Photoplethysmogram</i> ) Fotopleletismograma
RR	( <i>Respiratory Rate</i> ) Frecuencia Respiratoria
RRSS	<i>Restricted Residual Sum of Squares</i> , suma de residuales al cuadrado de una autorregresión univariada no restringida
SNR	( <i>Signal-to-noise ratio</i> , abreviado SNR o S/N) Relación Señal a Ruido
URSS	<i>Unrestricted Residual Sum of Squares</i> , suma de residuos al cuadrado de la regresión no restringida
VAR	<i>Vector Autorregresivo</i>
VPID	Vena Pulmonar Inferior Derecha
VPIL	Vena Pulmonar Inferior Izquierda
VPSP	Vena Pulmonar Superior Derecha

## Lista de símbolos

$\gamma$	Gamma
$k_m$	Coefficientes de reflexión
$x(n)$	Señal de entrada
$y(n)$	Señal de salida
$\hat{x}(n)$	Predictor lineal
$M$	Orden del filtro
$f_p(n)$	Predicción de adelanto
$g_m(n)$	Predicción de atraso
$\zeta$	Parámetro fijo
$v(n)$	Vector de ruido.
$w_0(n)$	Coefficientes del sistema a ser identificado
$\eta(n)$	Ruido blanco gaussiano
$\sigma_v^2$	Varianza del proceso de Markov
$\sigma_x^2$	Varianza de la señal de entrada
$\sigma_\eta^2$	Varianza del ruido de medida
$A$	Proceso estocástico estacionario
$\bar{A}$	Conjunto de valores pasados
$\bar{\bar{A}}_t$	Conjunto de valores pasados y presentes

# Glosario

Algoritmo	Del Latín, dixit algoríthmus y éste a su vez del matemático persa Muhammad ibn Musa al-Jwarizmi, es una lista bien definida, ordenada y finita de operaciones que permite hallar la solución a un problema.
Configuración <i>Affine</i> .	Es la Combinación de dos filtros adaptativos cuyo fundamento es la selección escalar de un parámetro de mezcla $\lambda(n)$ para combinar las salidas de los dos (2) filtros buscando obtener finalmente una tasa de convergencia más eficiente y un mejor estado de equilibrio. En una Combinación <i>Affine</i> se realiza una suma ponderada de las salidas de dos filtros adaptativos sometidos a una misma entrada cuyo factor de ponderación es un parámetro calculado mediante un mecanismo adaptativo (filtros adaptativos) con el propósito de lograr un mejor rendimiento.
Desajuste	Del inglés <i>Misadjustment</i> , es una medida cuantitativa de la cantidad del valor final del error cuadrático medio.
DSP	Del acrónimo de <i>Digital Signal Processor</i> , Procesador Digital de Señal, es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones mumericas a muy alta velocidad.
Estructura en Celosía o <i>Lattice</i>	La estructura de los filtros <i>Lattice</i> es una alternativa para realizar una función de transferencia de un filtro digital por ofrecer propiedades ventajosas como :Un número reducido de coeficientes permitiendo que grandes bloques de datos puedan ser modelados en tiempo real; Repetición de secciones idénticas logrando cierta simplificación del proceso (estructuras en cascada);Coeficientes con magnitudes menores a la unidad; prueba de estabilidad por inspección, buenas propiedades de redondeo numérico y son particularmente útiles para

el filtrado adaptable. Este tipo de estructura, utiliza los denominados coeficientes de reflexión y el análisis de predicción hacia adelante y hacia atrás para analizar la señal que pasa a través de él.

Filtrado	Proceso de eliminar alguna información específica de la señal.
Filtros Adaptativos	Estructuras ajustables con capacidad de cambiar su respuesta automáticamente para mejorar su desempeño.
FIR	Acónimo en inglés para <i>Finite Impulse Response</i> o Respuesta finita al impulso. Se trata de un tipo de filtros digitales cuya respuesta a una señal impulso como entrada tendrá un número finito de términos no nulos.
FPGA	Del inglés <i>Field Programmable Gate Array</i> es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad se puede programar.
Función de Transferencia	Modelo matemático que entrega la respuesta de un sistema a una señal de entrada o excitación exterior.
Hardware	Neologismo proveniente del inglés utilizado generalmente para describir componentes físicos de una tecnología.
IIR	Sigla en inglés para <i>Infinite Impulse Response</i> o Respuesta infinita al impulso. Se trata de un tipo de filtros digitales en el que, como su nombre indica, si la entrada es una señal impulso, la salida tendrá un número infinito de términos no nulos, es decir, nunca vuelve al reposo.
LMS	Del inglés, <i>Least-Mean-Square algorithm</i> , se usa en filtros adaptativos para encontrar los coeficientes del filtro que permiten obtener el valor esperado mínimo del cuadrado de la señal de error, definida como la diferencia entre la señal deseada y la señal producida a la salida del filtro.
MATLAB	Abeviatura de <i>Matrix Laboratory</i> (laboratorio de matrices). Es un programa de análisis numérico creado por <i>The Math Works</i> .

MSE	En estadística, el error cuadrático medio de un estimador mide el promedio de los errores al cuadrado, es decir, la diferencia entre el estimador y lo que se estima. El ECM es una función de riesgo, correspondiente al valor esperado de la pérdida del error al cuadrado o pérdida cuadrática. La diferencia se produce debido a la aleatoriedad o porque el estimador no tiene en cuenta la información que podría producir una estimación más precisa.
PPG	Trazado hecho por el pletismógrafo, aparato para medir las variaciones de tamaño de una parte por efecto del flujo sanguíneo.
Rango de convergencia	Se conoce así al número de iteraciones requeridas por un algoritmo, para converger lo suficiente a una solución óptima.
Robustez	Del inglés <i>Robustness</i> , es una medida cuantitativa de los errores de estimación de un algoritmo.
Seguimiento	Del inglés <i>Tracking</i> , se refiere a la propiedad de un algoritmo de rastrear variaciones estadísticas al trabajar en un medio no estacionario.
Software	Palabra de origen anglosajón con la cual se denomina al conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica.
SNR	La <b>relación señal/ruido</b> o <b>S/R</b> (en inglés <i>signal-to-noise ratio</i> , abreviado SNR o S/N) se define como la proporción existente entre la potencia de la señal que se transmite y la potencia del ruido que la corrompe. Este margen es medido en decibelios. Rango dinámico y relación señal/ruido para referirse a este margen que hay entre el ruido de fondo y nivel de referencia, pueden utilizarse como sinónimos. No ocurre lo mismo, cuando el rango dinámico indica la distancia entre el nivel de pico y el ruido de fondo. Que en las especificaciones técnicas de un equipo aparezca la relación señal/ruido indicada en decibelios no tiene significado si no se acompaña de los puntos de referencia utilizados y las ponderaciones.

# Capítulo 1

## Introducción

### 1.1. Planteamiento del problema. Motivación-Justificación

En el campo del procesamiento digital de señales es frecuente utilizar sistemas cuyas estructuras pueden modelar y analizar la señal digital con base en sus características o en la retroalimentación de su salida. Además de las estructuras conocidas (FIR e IIR) se tiene la estructura *Lattice* que tiene como ventajas un número reducido de coeficientes permitiendo que grandes bloques de datos puedan ser modelados en tiempo real, repetición de secciones idénticas logrando cierta simplificación del proceso (estructuras en cascada), coeficientes con magnitudes menores a la unidad, prueba de estabilidad por inspección, buenas propiedades de redondeo numérico y es particularmente útil para el filtrado adaptable dado que la solución recursiva de la estimación de los mínimos cuadrados produce naturalmente un filtro de estructura *Lattice* (Proakis y Manolakis, 2007). Así mismo, la estructura *Lattice* ortogonaliza la señal de entrada en una base de etapa a etapa, permitiendo capacidades de rápida convergencia y seguimiento. Este tipo de estructura, como se verá más adelante, utiliza los denominados coeficientes de reflexión y el análisis de predicción hacia adelante y hacia atrás para analizar la señal que pasa a través de él.

Muchas técnicas alternativas han sido desarrolladas para estimar los coeficientes de reflexión que parametrizan la estructura *Lattice*, el método de los mínimos cuadrados recursivos que actualiza la estimación de los mínimos cuadrados a través de la observación de cada muestra conlleva a una estimación óptima a pesar de que requiere únicamente un costo computacional ligeramente mayor al de las técnicas tradicionales (Galindo, 2008).

En las técnicas de estimación adaptables se modifican los coeficientes del filtro (sistema) de acuerdo con los datos que se actualizan cada vez que arriba una nueva muestra. En el caso de la estimación recursiva de un filtro *Lattice* son los coeficientes de reflexión los que se actualizan produciendo nuevos errores de predicción por cada orden del filtro. Es necesario

cambiar cada coeficiente del filtro para cada nueva muestra en las aplicaciones donde la convergencia rápida o el seguimiento (*tracking*) de señales son altamente variables. (Cowan y Grant, 1986) Así entonces la estructura *Lattice* está motivada por la teoría del modelo autorregresivo de la señal.

Ahora bien, el algoritmo acelerador versión  $\gamma$  ( $AR \gamma$ ) en configuración *Affine* consiste en una suma ponderada de las salidas de dos filtros adaptativos sometidos a una misma entrada cuyo factor de ponderación es un parámetro calculado mediante un mecanismo adaptativo con el propósito de lograr un mejor rendimiento. La configuración *Affine* ofrece una buena adaptabilidad mejorando características como velocidad, reducción del error de medida final, una buena capacidad de seguimiento (*tracking*) y buenos resultados en desajuste. El filtro implementado en esta configuración alivia el compromiso entre la velocidad de convergencia y el error residual inherente al filtro. (Rivera, 2013). Teniendo en cuenta la experiencia anterior, se propone para el presente estudio, la implementación del algoritmo acelerador versión  $\gamma$  ( $AR \gamma$ ) en configuración *Affine* sobre la estructura *Lattice*, incidiendo específicamente en los coeficientes de reflexión buscando potenciar su adaptabilidad y optimizar el rendimiento de sus propiedades de velocidad, reducción del error de medida final, capacidad de seguimiento (*tracking*) y mínimo desajuste.

Para efectos de obtener una implementación del algoritmo en tiempo real que permita un análisis cuali-cuantitativo de los algoritmos mencionados y, en consecuencia, evaluar su desempeño sobre señales reales, se lleva a cabo, una experiencia en la cual se hace aplicación de la estructura *Affine-Lattice* para la determinación de zonas de *frecuencia dominante* causantes de fibrilación auricular en situaciones de patologías cardíacas. La experiencia tiene como fundamento el hecho de que, la estructura *Affine-Lattice* minimiza el error de predicción, lo que se traduce en la identificación de las zonas con *frecuencia dominante* como causantes de fibrilación cardíaca.

Se utilizan con tal propósito los registros de electrogramas intracardiacos obtenidos del servicio de electrofisiología cardíaca del Hospital General Universitario de Madrid, registros que permiten el análisis de fibrilación auricular obtenidas con un catéter intracardíaco. El análisis se apoya en el diseño de un programa desarrollado en MATLAB cuyas características básicas consisten en la aplicación de la estructura

*Affine-Lattice*, la utilización de un filtro adaptativo y el análisis con base en la Teoría de Causalidad de Granger.

Con base en lo anteriormente expuesto se realiza la formulación del problema de investigación, el cual, responde a la pregunta:

¿Cuál es el efecto sobre la respuesta de un sistema adaptativo cuando sobre esta estructura se utiliza una configuración *Affine-Lattice* soportada en el algoritmo adaptativo AR  $\gamma$ ?

## **1.2. Estado el Arte.**

Como antecedentes al problema de investigación formulado se pueden citar a Gibson y Haykin (1980) quienes describieron este tipo de estructura y su utilización para control de tráfico aéreo (ATC, *Air Traffic Control*) aportando una descripción amplia sobre las características de los filtros *Lattice*.

Friedlander, B (1982) en su obra "*Filtros Lattice para el procesamiento adaptativo*" (*Lattice filter for adaptive processing*) concluye que los filtros *Lattice* constituyen aproximaciones a la solución de mínimos cuadrados.

Park.Y., y Sommerfeld, S. (1996) en el estudio titulado "*Algoritmo de control de ruido adaptativo rápido basado en la estructura Lattice*" (*A Fast Adaptive Noise Control Algorithm Based on the Lattice Structure*) utilizan un predictor reticular para desacoplar la señal de referencia filtrada en aplicaciones acústicas. Debido a su propiedad de desacoplamiento este algoritmo es capaz de entregar un buen rendimiento a pesar de las amplias variaciones en el valor de propagación de la matriz de correlación de la entrada de referencia. Las simulaciones por computador permitieron evaluar el rendimiento del algoritmo de control de ruido basado en la estructura *Lattice*, obteniendo ventajas significativas sobre el algoritmo LMS en términos de velocidad de convergencia.

En 2002, Lu, J., Shen Ch., Qiu, X y Xu, B en "*Control activo de ruido mediante algoritmo de filtrado adaptativo IIR en estructura lattice*" (*Lattice form adaptative infinite impulse response filtering algorithm for active noise control*) hacen referencia a que un gran número de algoritmos se han derivado de la aplicación de filtros IIR en el control activo de ruido., sin embargo, la mayoría de ellos utilizan la estructura del

filtro IIR de forma directa, que se enfrenta a las dificultades de control de estabilidad y la velocidad de convergencia lenta para el ruido de componentes de banda estrecha. Para superar estas dificultades, además de utilizar los filtros IIR de forma directa proponen un nuevo algoritmo adaptativo que utiliza y actualiza el filtro IIR adaptativo en un sistema *Lattice* para control activo de ruido concluyendo la superioridad del algoritmo propuesto. Se prueba utilizando funciones de transferencia medidas a partir de un sistema de control activo de ruido. Los resultados de la simulación muestran que la estructura *Lattice* no solo converge más rápidamente que los de uso común, sino que también ofrece un mínimo error cuadrático medio, también muestra que el algoritmo propuesto da como resultado un sistema más robusto en la práctica.

El trabajo titulado “*Filtro Lattice con ganancia ajustable y aplicación en procesamiento de señales ópticas*” (*Lattice filter with adjustable gains and its application in optical signal processing*) de los autores Panahi, I. M. S., Kannan, G., Hunt, L. R., MacFaelane, D. L. y Tong, J. (2005) ofrece métodos recursivos para derivar la función de transferencia de entrada-salida del filtro en función de los parámetros de la estructura *Lattice*, tales como tiempo de retardo, ganancias, coeficientes de reflexión, etc. Proponen para estabilidad del filtro una clase de filtro reticular MIMO con modelos de ganancias ajustables integrados por dispositivos fotónicos para la comunicación óptica y aplicación de procesamiento de señales de alta velocidad que también pueden ser utilizados en la realización del filtro por FPGA o procesadores programables para aplicaciones acústicas o de habla. La estructura *Lattice* ilustrada en este estudio también se puede usar para modelar adecuadamente los canales de propagación de la señal en otras aplicaciones no ópticas, tales como radar, acústica o procesamiento de la señal sísmica.

Sukhumalwong, S., Benjangkprasert, C., Anantrasirichai, N y Janchitrapongvej, K. (2006) en el estudio titulado “*Echo cancellation using the new variable step-size algorithm adaptive lattice filters*” afirman que los algoritmos adaptativos se utilizan ampliamente para diversas aplicaciones; una de ellas es la cancelación de eco en la red de telefonía de larga distancia. La pérdida de retorno y la respuesta de impulso son características importantes de un trayecto de eco al momento de diseñar los canceladores de eco y la efectividad de los canceladores de eco depende de la elección del algoritmo de filtrado adaptativo. En este trabajo proponen un nuevo

algoritmo de paso de adaptación variable para filtros *Lattice* de cancelación de eco que ofrece una excelente velocidad de convergencia y proporcionan un alto rendimiento.

Fujii., K, Tanaka., M, Sasaoka, N y Itoh, Y (2007) proponen un método para la aplicación de algoritmos adaptativos convencionales en la estimación de los coeficientes de reflexión para el cual se hace uso del algoritmo NLMS ofreciendo el mismo rendimiento que el del método convencional. Así mismo encuentran que los filtros *Lattice* ofrecen una ventaja en comparación a los filtros FIR en sensibilidad al error de redondeo.

Posteriormente Qi, H (2009) utilizando la estructura *Lattice* en un algoritmo adaptativo de gradiente, encuentra superiores los índices de rendimiento incluyendo la capacidad de seguimiento y velocidad de convergencia con menos recursos de hardware en comparación a los filtros convencionales.

Takemoto, T., Sasaoka, N., Itoh, Y., y Fujii, K. (2010) investigan en un sistema de reconstrucción de ruido la aplicación del filtro *Lattice* como error de predicción lineal (LPEF, *Linear prediction error filter*) debido a que la estructura de un filtro *Lattice* se aproxima a un filtro de tracto vocal para el proceso de producción del habla, logran mejorar el error residual y la capacidad de mejora del habla con el uso de este tipo de filtro.

Otros investigadores como Lee y Sung (2011) usaron la estructura *Lattice* para el modelado de canal y cancelación de ruido con rápida velocidad de convergencia y otras propiedades de su estructura modular.

En mayo de 2013, se registra la patente No. US 8,452,034 B2 a nombre de Lalin Theverappurama (Mineapolis MN, US) consistente en el diseño de un dispositivo de asistencia auditiva para el procesamiento de una señal de entrada a partir de un micrófono y un receptor. El sistema utiliza un filtro adaptativo de gradiente con uno o más coeficientes de reflexión para medir la retroalimentación acústica desde el receptor al micrófono del dispositivo de asistencia auditiva.

La patente No. US 8,797,673 B2 de Agosto 5 de 2014 cuyo inventor es Supino.,L, describe la implementación de un circuito constituido por un sensor de vibración que genera una señal a partir de una perturbación, un filtro *Lattice* que descorrelaciona la señal del sensor generando un error de predicción hacia atrás o de retroceso y un filtro de regresión múltiple con estimación de una señal de corrección de la perturbación utilizando los errores de predicción hacia atrás.

Kim., S, Park., Y, Seo., Y y Youn., D. (2014) en su trabajo titulado *A robust high-order lattice adaptive notch filter and its application to narrowband noise cancellation*, proponen un filtro adaptativo *Lattice* de orden superior (LANF) que realiza el seguimiento de múltiples señales diferenciándola de la estructura en cascada convencional porque ofrece un seguimiento independiente de las señales de referencia y las señales iniciales. El filtro LANF aplica una cancelación del ruido de banda estrecha (ANC) para mitigar el efecto de la perturbación en la señal de referencia. Así entonces, la combinación de un algoritmo de gradiente y el filtro LANF permite obtener un sistema ANC de banda estrecha muy eficiente desde el punto de vista computacional. Los resultados experimentales demuestran la robustez del filtro LANF propuesto y la eficiencia del filtro ANC de banda estrecha.

Park., C y Lee., B (2014) realizan la estimación en tiempo real de la frecuencia respiratoria (RR) de un fotopletoxiograma (PPG) usando un filtro *notch Lattice* adaptativo. Se tenía conocimiento de que muchos investigadores intentaron adquirir información de la RR a partir del PPG en razón a que la respiración afecta la forma de onda del PPG. La mayoría de estos registros son difíciles de operar en tiempo real debido a su complejidad o requerimientos computacionales. Teniendo en cuenta lo anterior los autores buscan obtener información de la RR adoptando una estructura de filtrado y una técnica de estimación de frecuencia secuencial que extrae una frecuencia dominante de la señal dada; para el efecto, utilizan un filtro *notch* adaptativo en celosía (ALNF) para estimar la RR de un PPG junto con un ritmo cardiaco adicional que se utiliza como parámetro de adaptación del método. Adicionalmente diseñan un sistema de filtrado *notch* IIR para eliminar la componente cardiaca y sus armónicos de la PPG; se compara el método propuesto con el modelo AR de Burg utilizado en la estimación de la RR de un PPG. Se determina estadísticamente que la red de tipo adaptativo de la frecuencia respiratoria estimada (ALRE) es significativamente más precisa que el modelo AR de Burg.

La revisión bibliográfica al respecto y específicamente relacionada con la adecuación del Algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* adaptado a una estructura de Celosía o *Lattice*, no registra a la fecha, aplicaciones de algoritmos adaptativos en configuración *Affine* sobre este tipo de estructura, constituyendo el presente estudio, un nuevo aporte en el comportamiento de los filtros adaptativos para el procesamiento digital de señales.

### **1.3. Metodología**

La metodología utilizada en el estudio obedece al proceso de investigación tecnológica en ciencias de la ingeniería, la cual se orienta a generar conocimientos aplicables a la producción de bienes y servicios. Se direcciona específicamente a la solución de un problema, mediado por una hipótesis de solución, buscando una respuesta a los interrogantes, mediante el logro de objetivos (Tamayo y Tamayo., M, 1.999) ejecutados a través de la simulación apoyándose en la herramienta matemática computacional MATLAB. Lo anterior se traduce en la implementación del algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* sobre la estructura de Celosía o *Lattice*, incidiendo específicamente en los coeficientes de reflexión buscando potenciar su adaptabilidad y optimizar el rendimiento de sus propiedades de velocidad, reducción del error de medida final, capacidad de seguimiento (*tracking*) y mínimo desajuste. (ver Tabla 1).

#### **1.3.1. Hipótesis de Solución:**

La implementación del algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* sobre la estructura *Lattice* optimiza la adaptabilidad y el rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento.

#### **1.3.2. Objetivos:**

##### **1.3.2.1. General.**

Estudiar las propiedades que ofrece el algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* al adecuarlo a la estructura de Celosía o *Lattice*.

### 1.3.2.2. Específicos.

- Establecer el efecto de los parámetros de ajuste en el comportamiento de la estructura *Lattice-Affine* soportada en el algoritmo  $AR\ \gamma$ .
- Especificar las ecuaciones de diseño de la estructura adaptativa *Lattice* haciendo uso de la configuración *Affine* con el algoritmo  $AR\ \gamma$  como soporte.
- Ejecutar la simulación del Algoritmo Acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* sobre la estructura *Lattice* evaluando su desempeño sobre señales reales, haciendo uso de la herramienta computacional MATLAB.
- Comparar el comportamiento del Algoritmo Acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* sobre la estructura *Lattice* con sistemas *Lattice* adaptativos convencionales con base en un análisis cualitativo y cuantitativo de los algoritmos implementados.

### 1.3.3. Ejecución-Simulación.

La verificación del comportamiento del algoritmo Acelerador versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine* sobre la estructura *Lattice*, se soporta en la herramienta matemática computacional MATLAB, con especificaciones relativas a:

- Convergencia de la señal de error.
- Coeficientes de los filtros
- Coeficientes de reflexión.
- Señales de predicción (adelanto y atraso).
- Error cuadrático medio.

### 1.3.4. Producto.

Un filtro adaptativo para procesamiento digital de señales con optimización de la adaptabilidad y rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento (*tracking*).

Tabla 1.1. Metodología

El Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> y su aplicación en estructura de Celosía o <i>Lattice</i>					
Marco de Referencia	Formulación del Problema	Hipótesis de Solución	Objetivos	Ejecución-Simulación Herramienta matemática computacional MATLAB	Nuevo Producto
Propiedades del algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i>	¿Cuál es el efecto sobre la respuesta de un sistema adaptativo cuando sobre esta estructura se utiliza una configuración <i>Affine-Lattice</i> soportada en el algoritmo adaptativo AR $\gamma$ ?	La implementación, del algoritmo acelerador versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> sobre la estructura <i>Lattice</i> , optimiza la adaptabilidad y el rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento.	<b>General:</b> Optimizar las propiedades que ofrece el algoritmo acelerador versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> al adecuarlo a la estructura de Celosía o <i>Lattice</i> .	Verificación del comportamiento del algoritmo acelerador versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> sobre la estructura <i>Lattice</i>	Filtro adaptativo para procesamiento digital de señales con optimización de la adaptabilidad y rendimiento de sus propiedades de velocidad de convergencia y capacidad de seguimiento ( <i>tracking</i> ).
Características, usos y aplicaciones de la estructura de Celosía o <i>Lattice</i>			<b>Específicos:</b> Analizar el efecto de los parámetros de ajuste en el comportamiento de la estructura <i>Lattice-Affine</i> soportada en el algoritmo AR $\gamma$	Verificación del comportamiento de: - Convergencia de la señal de error. - Coeficientes de los filtros. - Coeficientes de reflexión.	
Concepto de Causalidad de Granger			Especificar las ecuaciones de diseño de la estructura adaptativa <i>Lattice</i> haciendo uso de la configuración <i>Affine</i> con el algoritmo AR $\gamma$ como soporte.	- Señales de predicción (adelanto y atraso). - Error cuadrático medio.	
Estado del Arte			Comparar el comportamiento del Algoritmo Acelerador Versión $\gamma$ (AR $\gamma$ ) en configuración <i>Affine</i> sobre la estructura <i>Lattice</i> con sistemas <i>lattice</i> adaptativos convencionales.		

Fuente: Rivera. S.M.F. 2016

## 1.4. Contribución de la tesis.

En investigación previa titulada “Efecto de la configuración *Affine* en el Algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ )” (Rivera, 2013) se analizó el comportamiento del algoritmo adaptativo AR  $\gamma$  en una configuración *Affine*. Se observó que, aunque se tienen cuatro parámetros de ajuste (tres del algoritmo AR  $\gamma$  y uno de la configuración *Affine*) la respuesta obtenida del sistema ofrecía una buena adaptabilidad, mejoramiento de características como velocidad, reducción del error de medida final, buena capacidad de seguimiento (*tracking*) y desajuste; identificándolo como un filtro que ofrece un buen comportamiento entre la velocidad de convergencia y el error residual inherente al filtro. Con base en estos resultados, el presente trabajo de investigación contribuye a analizar el efecto de implementar la configuración *Affine* sobre una estructura en celosía o *Lattice* teniendo como soporte de adaptación el algoritmo AR  $\gamma$ .

Para analizar las propiedades y el comportamiento en la adaptabilidad de los coeficientes de reflexión y el componente de error de predicción de la estructura *Lattice* cuando se hace uso del algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ ) en configuración *Affine*, se recurre al concepto de Causalidad de Granger. el cual, se fundamenta en que: en el tiempo la causa siempre precede a los efectos, es decir, si una serie de tiempo es causa de otra, el conocimiento del primer proceso ayudaría a predecir los valores futuros del otro (Granger, 1964).

Teniendo en cuenta, la conformación de la arquitectura *Lattice*, específicamente, las ecuaciones que la componen, además de su funcionamiento, la obtención de los predictores tanto de adelanto como de atraso, así como los coeficientes de reflexión, que se presuponen compatibles con las características de la configuración *Affine* potencializada con el uso del algoritmo AR  $\gamma$ , el propósito de este trabajo es combinar las dos estructuras con el fin de analizar la viabilidad de mejorar los resultados obtenidos en sistemas *Lattice* convencionales (Fujii., K, Tanaka., M, Sasaoka., N y Itoh., Y, 2007).

En este contexto, se busca validar el comportamiento del algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* sobre una estructura *Lattice* en el marco de una metodología de implementación de algoritmos basada en la simulación para la

evaluación de la dinámica de las variables de los algoritmos y validación de las implementaciones.

## 1.5. Organización de la tesis

El desarrollo de la investigación: “El algoritmo acelerador versión  $\gamma$ (AR  $\gamma$ ) en configuración Affine y su aplicación en estructura de Celosía o Lattice” se estructura en cinco (5) capítulos en los que se desarrollan los contenidos que a continuación se ilustran.

**Capítulo 1.** Constituido por la **Introducción**, que comprende e ilustra el **Planteamiento del Problema, Justificación, Motivación**; la metodología y sus componentes de Marco de referencia, el cual, desarrolla contenidos relativos a Estado del arte, concepto de Causalidad de Granger, características y propiedades del algoritmo acelerador versión  $\gamma$ (AR  $\gamma$ ) en configuración Affine y de la estructura Lattice; Hipótesis de solución; Objetivos, la ejecución o simulación soportada en herramienta matemática computacional MATLAB.

Así mismo, se ilustra en este capítulo, bajo la denominación de “*Contribución de la Tesis*” los aspectos fundamentales del proceso investigativo relacionados con el análisis, efectos y validación (mediante simulación) al momento de implementar la configuración Affine sobre una estructura Lattice teniendo como soporte de adaptación el algoritmo AR  $\gamma$ . La finalidad es, combinar las dos estructuras con el fin de analizar la viabilidad de mejorar los resultados obtenidos en sistemas Lattice convencionales. El Capítulo 1, igualmente, relaciona y describe bajo el ítem “*Organización de la Tesis*” los contenidos desarrollados en cada uno de los Capítulos que componen el estudio. Como se describe a continuación:

**Capítulo 2.** Titulado: **Características, usos y aplicaciones de la estructura de Celosía o Lattice.** En este capítulo, se desarrollan contenidos ilustrativos sobre: Características, usos y aplicaciones de la estructura Lattice, las ecuaciones constitutivas, Predictor de adelanto, el predictor de atraso, la conversión de coeficientes en estructura Lattice a coeficientes de un filtro FIR, como también, la Conversión de los coeficientes de un filtro FIR a los coeficientes Lattice. Se ilustra

igualmente, acerca de los usos, aplicaciones, ventajas y desventajas de la estructura *Lattice*.

**Capítulo 3.** Hace referencia al **Algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ ) en configuración Affine sobre la estructura *Lattice* y su relación con otros filtros adaptativos.** En este capítulo se describen las características del Algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ ) en configuración Affine y se se compara el comportamiento del Algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* sobre la estructura *Lattice* con sistemas *Lattice* adaptativos convencionales con base en un análisis cualitativo y cuantitativo de los algoritmos implementados que permita determinar entre otros criterios: error cuadrático medio, velocidad de convergencia y *tracking*, así mismo, el código que identifica el algoritmo para su implementación en MATLAB.

**Capítulo 4.** **Adecuación del algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ ) a la estructura *Affine-Lattice* y determinación de focos de fibrilación cardíaca.** Este capítulo ilustra el desempeño del algoritmo en estudio sobre señales reales a través de una experiencia que consiste en la determinación de zonas de frecuencia dominante causantes de fibrilación cardíaca. El propósito es el análisis cuali-cuantitativo del algoritmo mencionado y evidenciar el comportamiento predictivo de la estructura *Affine-Lattice* como también su capacidad de minimizar el error predicción.

**Capitulo 5. Conclusiones y Trabajo Futuro.** Se relacionan en este aparte, los resultados obtenidos como respuesta a los objetivos propuestos y enfocados en el comportamiento del Algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* y su aplicación en estructura de Celosía o *Lattice*, igualmente se propone algunas recomendaciones para el desarrollo de trabajos de investigación futuros que permitan continuar con el desarrollo y validación del algoritmo obtenido.

## Capítulo 2.

### Características, usos y aplicaciones de la estructura de Celosía o *Lattice*.

El procesamiento digital de señales requiere la utilización de estructuras de filtro que pueden modelar y analizar la señal digital con base en sus características. Este tipo de estructuras de filtro ofrece ventajas especialmente para efectos de la cuantificación. Como caso de particular importancia se destacan las denominadas estructuras de filtrado adaptable en cascada, en paralelo y en celosía o *Lattice*.

La estructura de filtros *Lattice* es una alternativa para realizar una *función de transferencia* de un filtro digital que no tiene un número mínimo de multiplicadores y sumadores para la realización de una función de transferencia pero sí ofrece muchas propiedades ventajosas como por ejemplo cascadeo de secciones idénticas, coeficientes con magnitudes menores a la unidad, prueba de estabilidad por inspección, buenas características de redondeo numérico y es particularmente útil para el filtrado adaptable dado que la solución recursiva de la estimación de los mínimos cuadrados produce naturalmente un filtro de estructura *Lattice*. Así mismo, se caracteriza porque ortogonaliza la señal de entrada en una base de *etapa a etapa* permitiendo capacidades de rápida convergencia y seguimiento. No obstante que muchas técnicas alternativas han sido desarrolladas para estimar los coeficientes de reflexión que parametrizan la estructura *Lattice*, el método de los mínimos cuadrados recursivos actualiza la estimación a través de la observación de cada muestra. Este procedimiento conlleva una estimación óptima y requiere únicamente un costo computacional ligeramente mayor al de otras técnicas. (Galindo, 2008)

Las técnicas de estimación adaptables modifican los parámetros estimados del filtro de acuerdo con los datos observados actualizados, para cada nueva muestra, la

estimación recursiva utilizando el filtro *Lattice* genera nuevos coeficientes de reflexión y errores de predicción para cada orden del filtro. Cambiar cada coeficiente del filtro para cada nueva muestra resulta importante en las aplicaciones donde la rápida convergencia o el seguimiento (*tracking*) de señales altamente variables son necesarias. Sin embargo, para aplicaciones donde la dinámica es lenta los resultados de observar la señal por cierto tiempo resultan importantes. (Cowan y Grant, 1986) Así entonces, la estructura *Lattice*, está motivada por la teoría del modelo auto-regresivo de la señal.

## 2.1. Características de la estructura en Celosía o *Lattice*<sup>1</sup>.

El desarrollo de sistemas digitales ha permitido producir grandes avances en la tecnología, generando robustez en la fabricación de circuitos integrados, razón por la cual han surgido gran variedad de aplicaciones en cuanto al procesamiento de señales (voz, imagen, control, medición y/o adquisición, etc.)

Los filtros digitales constituyen una de las herramientas utilizadas en el procesamiento de señales y su clasificación se realiza con base en estructuras o topologías. Una de estas clasificaciones corresponde a la configuración de celda determinada por la teoría del modelado de señales autorregresivas de Markel y Gray (1976). Esta estructura modular, se fundamenta en un bloque denominado cruz o escalera que se va añadiendo uno a uno o periódicamente dependiendo del orden del filtro (Ver figura 2.1). Dicha estructura está constituida por bloques de retardos unidos a través de líneas horizontales, puntos de suma, líneas cruzadas de coeficientes, denominados coeficientes de reflexión ( $k_m$ ), como se muestra en la figura 2.1.

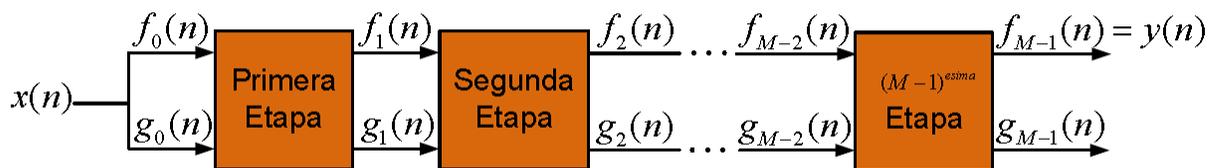


Figura 2.1. Estructura *Lattice* de  $M-1$  etapas (Apolo y Córdova, 2010)

<sup>1</sup> En adelante se seguirá usando el termino Lattice.

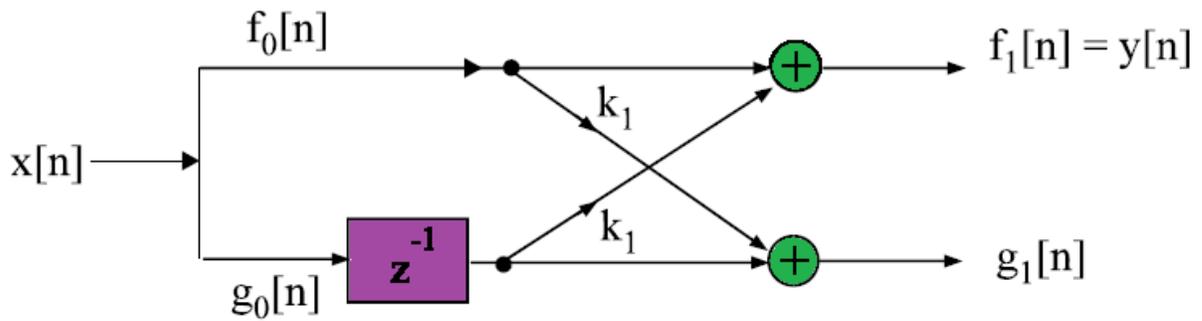


Figura 2.2. Estructura *Lattice* de filtro FIR de primer orden

Es importante destacar que el modelo *Lattice* corresponde a una estructura del filtro FIR, denominada realización *Lattice* o filtro *Lattice*. Su uso está direccionado hacia aplicaciones de tratamiento de voz y filtros adaptativos. Su desarrollo se inicia considerando una secuencia de filtros FIR cuyas funciones son:

$$H_m(z) = A_m(z), \quad m = 0, 1, 2, \dots, M - 1 \quad (2.1)$$

siendo por definición  $A_m(z)$  el polinomio

$$A_m(z) = 1 + \sum_{k=1}^m \alpha_m(k)z^{-k}, \quad m \geq 1 \quad (2.2)$$

$A_0(z) = 1$ . La respuesta al impulso unitario del filtro  $m$  es  $h_m(0) = 1$  y  $h_m(k) = \alpha_m(k)$ ,  $k = 1, 2, \dots, m$

El grado del polinomio  $A_m(z)$  es indicado por el subíndice  $m$ . Se define  $\alpha_m(0) = 1$  por conveniencia matemática.

Si la señal de entrada es  $x(n)$  y  $y(n)$  es la señal de salida, se tiene

$$y(n) = x(n) + \sum_{k=1}^m \alpha_m(k)x(n - k) \quad (2.3)$$

En la figura 2.3 se observa, la forma directa del filtro FIR de predicción.

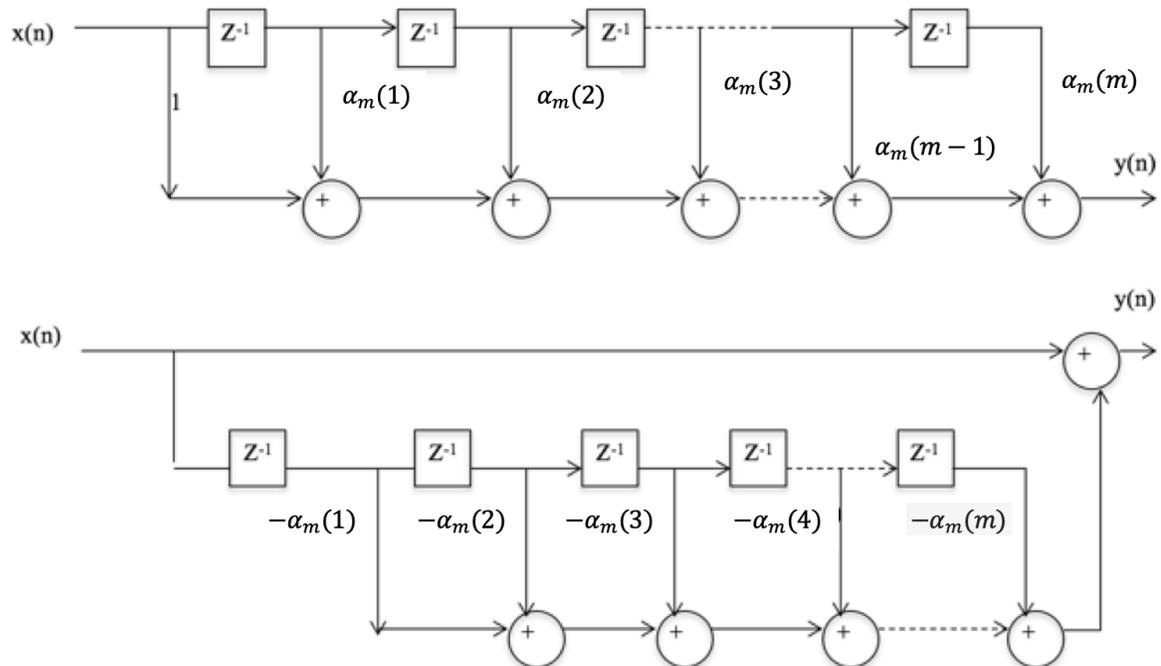


Figura 2.3. Forma directa del filtro FIR de predicción

Siendo el predictor lineal

$$\hat{x}(n) = - \sum_{k=1}^m \alpha_m(k) x(n - k) \quad (2.4)$$

Expresión que corresponde al valor predictivo de un escalón de  $x(n)$ , basándose en las m entradas pasadas y en  $y(n) = x(n) - \hat{x}(n)$  dadas por la ecuación (2.3) representando el error de predicción.

Si se tiene un filtro de orden  $M - 1$ , la salida del filtro corresponde a

$$y(n) = x(n) + \alpha_1(1)x(n - (M - 1)) \quad (2.5)$$

Para el filtro de celosía en la primera etapa, esta ecuación sería también su salida como se puede ver en la figura 2.2, siendo la señal de entrada  $x(n)$  y  $y(n)$  la salida de la rama superior. La salida es exactamente la ecuación (2.5), si se toma  $K_1 = \alpha_1(1)$  el parámetro  $K_1$  se denominaría coeficiente de reflexión.

Si se considera un filtro FIR con  $M = 2$ , la salida sería

$$y(n) = x(n) + \alpha_1(1)x(n - 1) + \alpha_2(2)x(n - 2) \quad (2.6)$$

Se asigna a  $K_1 = \alpha_1(1)$  llamando  $K_1$  como el parámetro denominado coeficiente de reflexión.

Al conectar las dos etapas de celosía en cascada (Figura 2.4) se obtiene la misma salida de la ecuación (2.6).

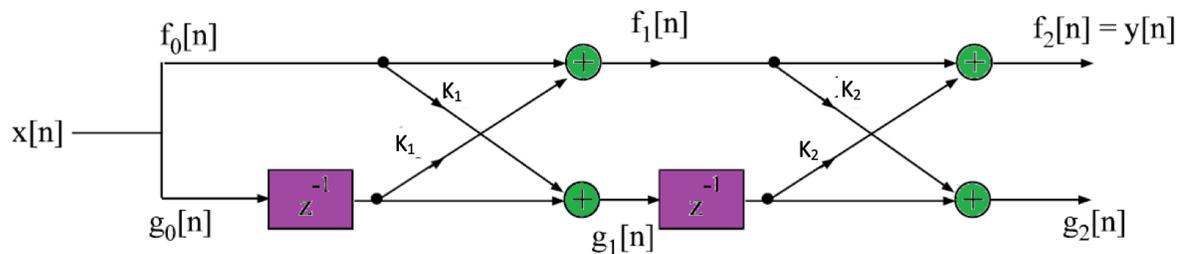


Figura 2.4. Estructura *Lattice* de filtro FIR de segundo orden

Luego la salida para la primera etapa es

$$f_1(n) = x(n) + K_1 x(n - 1) \quad (2.7)$$

Para la segunda etapa

$$\begin{aligned} f_2(n) &= f_1(n) + K_2 g_2(n - 1) \\ g_2(n) &= K_2 f_1(n) + g_1(n - 1) \end{aligned} \quad (2.8)$$

Al ser  $f_2(n)$  la salida y al sustituir  $f_1(n)$  y  $g_1(n - 1)$  dadas por las ecuaciones (2.7) y (2.8) se obtiene

$$\begin{aligned} f_2(n) &= x(n) + K_1 x(n - 1) + K_2 [K_1 x(n - 1) + x(n - 2)] \\ &= x(n) + K_1(1 + K_2)x(n - 1) + K_2 x(n - 2) \end{aligned} \quad (2.9)$$

Si se igualan los coeficientes con la ecuación (2.9) se obtiene la misma salida del filtro FIR dada por (2.6), es decir:

$$\alpha_2(2) = K_2, \quad \alpha_2(1) = K_1(1 - K_2) \quad (2.10)$$

Equivalente a

$$K_2 = \alpha_2(2), \quad K_1 = \frac{\alpha_2(2)}{1 + \alpha_2(2)} \quad (2.11)$$

Así se demuestra la equivalencia entre un filtro FIR de orden  $m$  y un filtro *Lattice* de orden  $m$  ó de  $m$  etapas.

Las ecuaciones para un filtro *Lattice* son:

$$f_0(n) = g_0(n) = x(n) \quad (2.12)$$

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1), \quad m = 1, 2, \dots, M-1 \quad (2.13)$$

$$g_m(n) = K_m f_{m-1}(n) + g_{m-1}(n-1), \quad m = 1, 2, \dots, M-1 \quad (2.14)$$

En donde la salida del filtro (M-1) etapas es la salida de un filtro FIR de orden (M-1), es decir:

$$y(n) = f_{M-1}(n) \quad (2.15)$$

La figura 2.1 muestra el filtro *Lattice* de (M-1) etapas en diagrama de bloques donde se ilustra una etapa general presentando los cálculos específicos para las ecuaciones (2.13) y (2.14) del filtro *Lattice*.

Al hacer la equivalencia entre un filtro FIR y un filtro *Lattice*, la salida  $f_m(n)$  de la estructura en celosía de  $m$  etapas se puede expresar como se indica en la ecuación 2.16:

$$f_m(n) = \sum_{k=0}^m \alpha_m(k)x(n-k), \quad \alpha_m(0) = 1 \quad (2.16)$$

Por tratarse de una suma de convolución (ecuación 2.16), la transformada  $z$  es:

$$F_m(z) = A_m(z)X(z) \quad (2.17)$$

Que es igual a

$$A_m(z) = \frac{F_m(z)}{X(z)} = \frac{F_m(z)}{F_0(z)} \quad (2.18)$$

El otro componente del filtro *Lattice*  $g_m(n)$  se puede expresar también en forma de convolución de la ecuación (2.16). Retomando a las ecuaciones (2.7) y (2.8) para utilizar otro conjunto de coeficientes, por ejemplo  $\beta_m(k)$  se observa que en la ecuación (2.7) los coeficientes del filtro *Lattice* que produce  $f_1(n)$  son  $(1, K_1) = (1, \alpha_1(1))$  y los coeficientes del filtro que producen la salida  $y(n)$  evidencian que el orden de los coeficientes es inverso. Si se consideran dos etapas con salida dada por la ecuación (2.8) se comprueba que  $g_2(n)$  puede expresarse así:

$$\begin{aligned}
 g_2(n) &= K_2 f_1(n) + g_1(n-1) \\
 &= K_2 [x(n) + K_1 x(n-1)] + K_1 x(n-1) + x(n-2) \\
 &= K_2 x(n) + K_1 (1 + K_2) x(n-1) + x(n-2) \\
 &= \alpha_2(2) x(n) + \alpha_2(1) x(n-1) + x(n-2)
 \end{aligned} \tag{2.19}$$

Al comparar los coeficientes que se obtienen de la salida  $f_2(n)$  con los que se obtienen de la salida  $g_2(n)$  se observa que también son inversos.

De igual forma se deduce entonces que la salida  $g_m(n)$  de un filtro *Lattice* de  $m$  etapas se puede expresar por la suma de convolución así:

$$g_m(n) = \sum_{k=0}^m \beta_m(k) x(n-k) \tag{2.20}$$

Donde los coeficientes  $\beta_m(k)$  se asocian con el filtro de donde se obtiene  $f_m(n) = y(n)$  pero en orden inverso, así

$$\beta_m(k) = \alpha_m(m-k), \quad k = 0, 1, \dots, m \tag{2.21}$$

con  $\beta_m(m) = 1$ .

Al abordar el predictor lineal, se suponen los datos  $x(n), x(n-1), \dots, x(n-m+1)$  que se usan para predecir linealmente el valor de la señal  $x(n-m)$  usando un filtro lineal con coeficientes  $-\beta_m(k)$  donde el valor predictivo es

$$\hat{x}(n-m) = - \sum_{k=0}^{m-1} \beta_m(k)x(n-k) \quad (2.22)$$

Como los datos se procesan de forma inversa por el predictor, la predicción especificada por la ecuación (2.22) se conoce como *predicción inversa*. Para el filtro FIR del sistema  $A_m(z)$  se denomina predictor directo.

El dominio en la transformada  $z$ , (2.21) se convierte en

$$G_m(z) = B_m(z)X(z) \quad (2.23)$$

Luego así

$$B_m(z) = \frac{G_m(z)}{X(z)} \quad (2.24)$$

En donde  $B_m(z)$  representa la función del filtro FIR con coeficientes  $\beta_m(k)$ , así

$$B_m(z) = \sum_{k=0}^m \beta_m(k)z^{-k} \quad (2.25)$$

Al ser  $\beta_m(k) = \alpha_m(m-k)$ , (24) se puede escribir

$$\begin{aligned} B_m(z) &= \sum_{k=0}^m \alpha_m(m-k)z^{-k} \\ &= \sum_{l=0}^m \alpha_m(l)z^{l-m} \\ &= z^{-m} \sum_{l=0}^m \alpha_m(l)z^l = z^{-m} A_m(z^{-1}) \end{aligned} \quad (2.26)$$

La relación dada en (2.26) implica que los ceros del filtro FIR donde cuya función es  $B_m(z)$  son los recíprocos de  $A_m(z)$ , así  $B_m(z)$  es el polinomio inverso de  $A_m(z)$ .

Volviendo a las ecuaciones recursivas del filtro en celosía dadas por (2.12), (2.13) y (2.14) pasando al dominio de  $z$ , se tiene,

$$F_0(z) = G_0(z) = X(z) \quad (2.27)$$

$$F_m(z) = F_{m-1}(z) + K_m z^{-1} G_{m-1}(z) \quad (2.28)$$

$$G_m(z) = K_m F_{m-1} + z^{-1} G_{m-1}(z) \quad (2.29)$$

Dividiendo por  $X(z)$  el resultado es de la forma

$$A_0(z) = B_0(z) = 1 \quad (2.30)$$

$$A_m(z) = A_{m-1}(z) + K_m z^{-1} B_{m-1}(z) \quad m = 1, 2, \dots, M - 1 \quad (2.31)$$

$$B_m(z) = K_m A_{m-1} + z^{-1} B_{m-1}(z) \quad m = 1, 2, \dots, M - 1 \quad (2.32)$$

Describiendo de forma matricial

$$\begin{bmatrix} A_m(z) \\ B_m(z) \end{bmatrix} = \begin{bmatrix} 1 & K_m \\ K_m & 1 \end{bmatrix} \begin{bmatrix} A_{m-1}(z) \\ z^{-1} B_{m-1}(z) \end{bmatrix} \quad (2.33)$$

### 2.1.1. Predictor lineal directo o de adelanto.

El problema es predecir un valor futuro (predicción hacia adelante) de un proceso aleatorio estacionario a partir de valores pasados del proceso. Se considera que el *predictor de adelanto de una muestra*, es la predicción del valor  $x(n)$  por medio de una combinación lineal de los valores pasados  $x(n-1), x(n-2), \dots, x(n-p)$ . El valor predictivo de adelanto sería

$$\hat{x}(n) = - \sum_{k=1}^p a_p(k) x(n-k) \quad (2.34)$$

de donde  $-a_p(k)$  son los pesos de la combinación lineal. Estos son los pesos de los *coeficientes de predicción* del predictor de adelanto de una muestra de orden  $p$ .

La diferencia entre  $x(n)$  y el valor predictivo de  $x(n)$  es el *error de predicción de adelanto*, designado por  $f_p(n)$

$$\begin{aligned} f_p(n) &= x(n) - \hat{x}(n) \\ &= x(n) + \sum_{k=1}^p a_p(k) x(n-k) \end{aligned} \quad (2.35)$$

Se observa la predicción de adelanto de forma equivalente al filtrado lineal donde el predictor se integra en el filtro lineal, como se ve en la figura 2.5, que corresponde a un filtro de error de predicción con una señal de entrada  $x(n)$  y una señal de salida  $f_p(n)$  una equivalencia al filtro de error de predicción se observa en la figura 2.6, con una función de sistema

$$A_p(z) = \sum_{k=0}^p a_p(k)z^{-k}, \quad a_p(0) = 1 \quad (2.36)$$

Dado que la ecuación anterior es una suma de convolución, su transformada z es

$$F_p(z) = A_p(z)X(z) \quad (2.37)$$

Entonces se tiene

$$A_p(z) = \frac{F_p(z)}{X(z)} = \frac{F_p(z)}{F_0(z)} \quad (2.38)$$

El valor cuadrático medio del error de predicción de adelanto  $f_p(n)$  es

$$\begin{aligned} \mathcal{E}_p^f &= E \left[ |f_p(n)|^2 \right] \\ &= \gamma_{xx}(0) + 2\Re \left[ \sum_{k=1}^p a_p^*(k)\gamma_{xx}(k) \right] + \sum_{k=1}^p \sum_{l=1}^p a_p^*(l)a_p(k)\gamma_{xx}(l-k) \end{aligned} \quad (2.39)$$

$\mathcal{E}_p^f$  es una función cuadrática de los coeficientes del predictor y al minimizarlo llevará al conjunto de ecuaciones lineales

$$\gamma_{xx}(l) = - \sum_{k=1}^p a_p(k)\gamma_{xx}(l-k), \quad l = 1, 2, \dots, p \quad (2.40)$$

Estas son las ecuaciones normales para los coeficientes del predictor de adelanto. El mínimo error de predicción cuadrático es

$$\min[\mathcal{E}_p^f] \equiv E_p^f = \gamma_{xx}(0) + \sum_{k=1}^p a_p(k)\gamma_{xx}(-k) \quad (2.41)$$

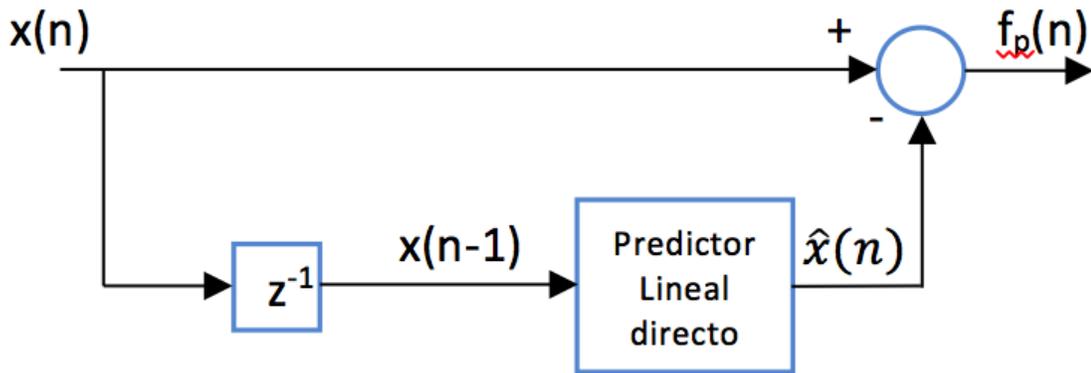


Figura 2.5. Predictor de adelanto

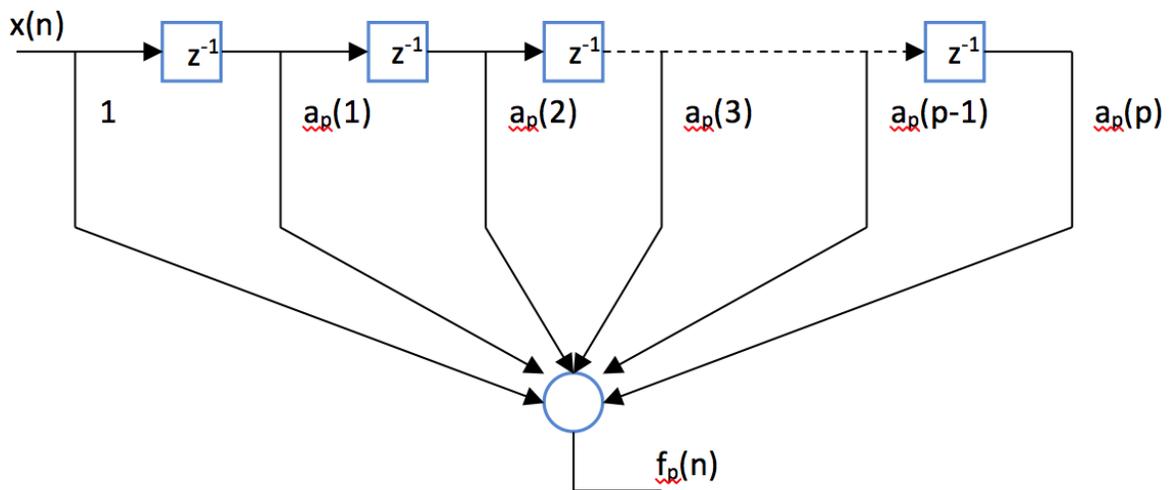


Figura 2.6. Filtro de error de predicción

### 2.1.2. Predictor lineal Inverso o de atraso

Se tiene la secuencia de datos  $x(n), x(n-1), \dots, x(n-p+1)$  de un proceso estacionario y se desea predecir el valor  $x(n-p)$  del proceso. Al emplear un *predictor de atraso* de una muestra de orden  $p$ , se tiene

$$\hat{x}(n-p) = - \sum_{k=0}^{p-1} b_p(k)x(n-k) \quad (2.42)$$

El *error de predicción de atraso* es la diferencia entre el valor  $x(n-p)$  y el estimado  $\hat{x}(n-p)$ , designado como

$$\begin{aligned} g_p(n) &= x(n-p) + \sum_{k=0}^{p-1} b_p(k)x(n-k) \\ &= \sum_{k=0}^p b_p(k)x(n-k), \quad b_p(p) = 1 \end{aligned} \quad (2.43)$$

Los coeficientes del predictor de atraso son los complejos conjugados de los coeficientes del predictor de adelanto, pero de forma inversa. Así se tiene

$$b_p(k) = a_p^*(p-k), \quad k = 0, 1, \dots, p \quad (2.44)$$

en el dominio de  $z$ , la suma de convolución dada en (2.43) se transforma en

$$G_p(z) = B_p(z)X(z) \quad (2.45)$$

Así entonces

$$B_p(z) = \frac{G_p(z)}{X(z)} = \frac{G_p(z)}{G_0(z)} \quad (2.46)$$

La función del sistema del filtro FIR con coeficientes  $b_p(k)$  es representada por  $B_p(k)$

Como  $b_p(k) = a_p^*(p-k)$ ,  $G_p(z)$  se relaciona con  $A_p(z)$

$$\begin{aligned} B_p(z) &= \sum_{k=0}^p b_p(k)z^{-k} \\ &= \sum_{k=0}^p a_p^*(p-k)z^{-k} \\ &= z^{-p} \sum_{k=0}^p a_p^*(k)z^k \\ &= z^{-p} A_p^*(z^{-1}) \end{aligned} \quad (2.47)$$

Se puede ver que  $B_p(k)$  son los recíprocos (conjugados) de  $A_p(z)$ .

Volviendo a las ecuaciones en Celosía recursivas dadas en (2.27), (2.28) y (2.29) y transformándolas en el dominio de  $z$ , se tiene

$$\begin{aligned} F_0(z) &= G_0(z) = X(z) \\ F_m(z) &= F_{m-1}(z) + K_m z^{-1} G_{m-1}(z), & m = 1, 2, \dots, p \\ G_m(z) &= K_m^* F_{m-1}(z) + z^{-1} G_{m-1}(z) & m = 1, 2, \dots, p \end{aligned} \quad (2.48)$$

dividiendo por  $X(z)$ , se obtiene

$$\begin{aligned} A_0(z) &= B_0(z) = 1 \\ A_m(z) &= A_{m-1}(z) + K_m z^{-1} B_{m-1}(z), & m = 1, 2, \dots, p \\ B_m(z) &= K_m^* A_{m-1}(z) + z^{-1} B_{m-1}(z) & m = 1, 2, \dots, p \end{aligned} \quad (2.49)$$

Descrito de forma matricial

$$\begin{bmatrix} A_m(z) \\ B_m(z) \end{bmatrix} = \begin{bmatrix} 1 & K_m z^{-1} \\ K_m^* & z^{-1} \end{bmatrix} \begin{bmatrix} A_{m-1} \\ B_{m-1} \end{bmatrix} \quad (2.50)$$

La estructura *Lattice* con parámetros  $K_1, K_2, \dots, K_p$  corresponde a un filtro FIR en forma directa de orden  $p$  con funciones  $A_1(z), A_2(z), \dots, A_p(z)$ . Una caracterización de esta clase de filtro FIR de orden  $p$  en forma directa requiere  $\frac{p(p+1)}{2}$  coeficientes del filtro.

Por otro lado, la caracterización en la estructura *Lattice* sólo necesita los  $p$  coeficientes de reflexión  $K_i$ . La razón de que la estructura *Lattice* sea de forma compacta para filtros FIR de orden  $p$  es que al adicionar etapas a la estructura *Lattice* no altera los parámetros de las etapas anteriores. Por otra parte, al adicionar una etapa  $p$  a una estructura *Lattice* ( $p-1$ ) etapas es incrementar la longitud del filtro FIR en un coeficiente. El filtro FIR con funciones  $A_p(z)$  tiene coeficientes diferentes del filtro FIR de orden superior con una función de sistema  $A_{p-1}(z)$ .

Para determinar los coeficientes del filtro  $a_p(k)$  de forma recursiva se deducen a partir de (2.49), así se tiene

$$\begin{aligned} A_m(z) &= A_{m-1}(z) + K_m z^{-1} B_{m-1}(z) \\ \sum_{k=0}^m a_m(k) z^{-k} &= \sum_{k=0}^m a_{m-1}(k) z^{-k} + K_m \sum_{k=0}^m a_{m-1}^*(m-1-k) z^{-(k+1)} \end{aligned} \quad (2.51)$$

Igualando coeficientes de potencias iguales a  $z^{-1}$  y teniendo en cuenta que  $a_m(0)=1$  para  $m=1,2,\dots,p$  de forma que para los coeficientes del filtro FIR deseados sus ecuaciones recursivas son:

$$\begin{aligned} a_m(0) &= 1 \\ a_m(m) &= K_m \\ a_m(k) &= a_{m-1}(k) + K_m a_{m-1}^*(m-k) \\ &= a_{m-1}(k) + a_m(m) a_{m-1}^*(m-k), \quad 1 \leq k \leq m-1; \quad m = 1, 2, \dots, p \end{aligned} \quad (2.52)$$

Para la estructura *Lattice* de  $p$  etapas se obtienen los coeficientes de reflexión  $K_p = a_p(p)$ . Para obtener estos coeficientes se necesitan los polinomios  $A_m(z)$  para  $m = p-1, \dots, 1$  a partir de (2.50)

$$A_{m-1}(z) = \frac{A_m(z) - K_m B_m(z)}{1 - |K_m|^2}, \quad m = p, \dots, 1 \quad (2.53)$$

siendo una recursión de orden descendente. Al calcular los polinomios de grado inferior  $A_m(z)$  iniciando por  $A_{p-1}(z)$  y obteniendo los coeficientes de reflexión a partir de la relación  $K_m = a_m(m)$ . El procedimiento funciona cuando  $|K_m| \neq 1$  para  $m = 1, 2, \dots, p-1$ . Luego la fórmula para obtener  $K_m, m = p-1, \dots, 1$  de forma recursiva a partir de la recursión descendente se tiene

$$\begin{aligned} K_m &= a_m(m) \\ a_{m-1}(k) &= \frac{a_m(k) - K_m b_m(k)}{1 - |K_m|^2} \end{aligned} \quad (2.54)$$

$$= \frac{a_m(k) - a_m(m)a_m^*(m-k)}{1 - |a_m(m)|^2}$$

La ecuación recursiva dada en (2.54) falla si cualquier parámetro *Lattice* cumple que  $|K_m| = 1$ . Esto indica que el polinomio  $A_{m-1}(z)$  tiene una raíz sobre la circunferencia de radio = 1.

Al considerar la minimización del error cuadrático medio en un predictor de atraso se tiene:

$$\begin{aligned} g_p(n) &= x(n-p) + \sum_{k=0}^{p-1} b_p(k)x(n-k) \\ &= x(n-p) + \sum_{k=1}^p a_p^*(k)x(n-p+k) \end{aligned} \quad (2.55)$$

y su valor cuadrático medio es:

$$\mathcal{E}_p^b = E \left[ |g_p(n)|^2 \right] \quad (2.56)$$

La minimización de  $\mathcal{E}_p^b$  con respecto a los coeficientes de predicción permite obtener el mismo conjunto de ecuaciones lineales que (2.40), así el mínimo error cuadrático medio (MSE, *minimum mean-square error*) es

$$\text{mín}[\mathcal{E}_p^b] \equiv E_p^b = E_p^f \quad (2.57)$$

Que queda determinado por la ecuación (2.41)

## 2.2. Conversión de coeficientes en estructura *Lattice* a coeficientes de un filtro FIR.

Para obtener los coeficientes  $\{b_m(k)\}$  de un filtro FIR en forma directa comenzando por los coeficientes *Lattice*  $\{k_m\}$  se aplican las siguientes ecuaciones:

$$F_0(z) = G_0(z) = 1 \quad (2.58)$$

$$F_m(z) = F_{m-1}(z) + K_m z^{-1} G_{m-1}(z) \quad m = 1, 2, \dots, M-1 \quad (2.59)$$

$$G_m(z) = z^{-1} F_{m-1}(z^{-1}) \quad m = 1, 2, \dots, M-1 \quad (2.60)$$

Siendo la solución una secuencia  $(M-1)$  filtros FIR, logrados recursivamente comenzando con  $m=1$ .

Una colección de  $m$  filtros FIR en forma directa requiere  $\frac{m(m+1)}{2}$  coeficientes de reflexión  $\{k_m\}$ , se advierte que la adición de etapas no altera los parámetros de las etapas previas, ofreciendo una estructura más compacta

### 2.3. Conversión de los coeficientes de un filtro FIR a los coeficientes *Lattice*

Los coeficientes  $\{k_m\}$  *Lattice* a partir  $\{b_m(k)\}$  de un filtro FIR en forma directa se relacionan así:

$$F_{m-1}(z) = \frac{F_m(z) - K_m F_m(z)}{1 + K_m^2}, \quad m = M-1, M-2, \dots, 1 \quad (2.61)$$

Siempre que  $|k_m| \neq 1$  para  $m=1, 2, \dots, M-1$ , iniciando por  $F_{M-1}(z)$  calculando los polinomios de orden inferior  $F_m(z)$  y el proceso iterativo continua para el sistema de orden reducido permitiendo obtener los coeficientes deseados *Lattice* a partir de la relación  $K_m = b_m(m)$ .

Se puede calcular  $K_m$  recursivamente para  $m = M - 1, M - 2, \dots, 1$

$$b_{m-1}(k) = \frac{K_m = b_m(m) \quad b_{m-1} = 1}{1 + K_m^2} = \frac{b_m(k) - K_m a_m(k)}{1 - b_m^2(m)} = \frac{b_m(k) - b_m(m)b_m(m-k)}{1 - b_m^2(m)} \quad (2.62)$$

## 2.4. Usos, aplicaciones, ventajas y desventajas de la estructura *Lattice*.

La estructura *Lattice*, ofrece una serie de propiedades de gran interés para aplicaciones de filtrado adaptativo. La Tabla 2 ilustra de manera comparativa dichas propiedades con las correspondientes a los filtros FIR e IIR.

Es importante observar que en lo relativo a la implementación, la estructura *Lattice* por su alto grado de modularidad permite canalizar los datos, ofrece propiedades deseables de estabilidad, excelente precisión numérica y rápida convergencia. Otras propiedades son por ejemplo el cascadeo de secciones idénticas, coeficientes con magnitudes menores a la unidad, prueba de estabilidad por inspección, buenas características de redondeo numérico resultando particularmente útiles para el filtrado adaptable dado que la solución recursiva de la estimación de los mínimos cuadrados produce naturalmente un filtro de estructura *Lattice*.

Tabla 2.2. Propiedades comparativas de la estructura *Lattice*.

	IIR	FIR	<i>LATTICE</i>
<b>USOS Y APLICACIONES</b>	Equivalentes a filtros análogos.	Transmisión de datos. Audio digital. Procesamiento de imágenes.	Procesamiento digital de señales de voz. Implementación de Filtros adaptativos. Tratamiento de señales geofísicas.
<b>VENTAJAS</b>	No requiere memoria extra  En un mismo diseño de filtro, la longitud del filtro IIR más corta que en FIR.  Mejor atenuación que los FIR.  Pueden proporcionar pendientes de corte muy pronunciadas.	Estructura del filtro siempre es estable.  Diseño para tener respuesta en fase lineal.  No entran en oscilación debido a que no tienen realimentación.	Número reducido de coeficientes permite que grandes bloques de datos puedan ser modelados en tiempo real.
<b>DESVENTAJAS</b>	Producen en general distorsión de fase, es decir la fase no es lineal con la frecuencia.  Transitorio que no se limita a un intervalo de tiempo finito.	Para especificaciones dadas requieren un filtro de orden mucho mayor que los filtros IIR.  Memoria finita.  Número de operaciones y tiempo de procesamiento altos comparados con los de IIR.	<b>El filtro de estructura <i>Lattice</i> puede ser FIR o IIR por tanto también posee sus desventajas.</b>

## Capítulo 3

### **El algoritmo acelerador versión $\gamma$ (AR $\gamma$ ) en combinación *Affine* sobre la estructura *Lattice* y su relación con otros filtros adaptativos.**

Se describe en este capítulo las propiedades y características del Algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine*, como también el comportamiento observado al adaptarlo sobre una estructura *Lattice* y comparativamente con otros sistemas *Lattice* adaptativos convencionales, con base en, un análisis cuali-cuantitativo de los algoritmos implementados con el propósito de determinar entre otros criterios error cuadrático medio, velocidad de convergencia y *tracking*, así mismo, el diagrama de flujo que identifica el algoritmo para su implementación en MATLAB.

#### **3.1. Características del algoritmo Acelerador regresivo versión $\gamma$ (AR $\gamma$ ) en configuración *Affine*.**

El algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine* (Rivera, 2013) se fundamenta en la suma ponderada de las salidas de dos filtros adaptativos sometidos a una misma entrada cuyo factor de ponderación es un parámetro calculado mediante un mecanismo adaptativo con el propósito de lograr un mejor rendimiento, el filtro combinado resultante tiende a comportarse como el mejor de los filtros y en algunos casos se comporta mejor que los filtros a combinar y alivia el compromiso entre la velocidad de convergencia y el error residual inherente al filtro.

El algoritmo se puede describir con las ecuaciones siguientes para cada filtro:

Ecuaciones Filtro 1

$$y_1(n) = x(n)w_1(n) \quad (3.1)$$

$$e_1(n) = y_1(n) - d(n) \quad (3.2)$$

$$g_1(n) = \frac{e_1(n)\gamma_1 x^T(n)q_1(n-1)}{1 + \alpha_1 \gamma_1 x^T(n)m_1 x(n)} \quad (3.3)$$

$$q_1(n) = \frac{\gamma_1}{\alpha_1 + \gamma_1} [q_1(n-1) - \alpha_1 g_1(n)m_1 x(n)] \quad (3.4)$$

$$w_1(n) = w_1(n-1) + \alpha_1 q_1(n) \quad (3.5)$$

Ecuaciones Filtro 2

$$y_2(n) = x(n)w_2(n) \quad (3.6)$$

$$e_2(n) = y_2(n) - d(n) \quad (3.7)$$

$$g_2(n) = \frac{e_2(n)\gamma_2 x^T(n)q_2(n-1)}{1 + \alpha_2 \gamma_2 x^T(n)m_2 x(n)} \quad (3.8)$$

$$q_2(n) = \frac{\gamma_2}{\alpha_2 + \gamma_2} [q_2(n-1) - \alpha_2 g_2(n)m_2 x(n)] \quad (3.9)$$

$$w_2(n) = w_2(n-1) + \alpha_2 q_2(n) \quad (3.10)$$

Para la configuración *Affine* se tiene

$$y(n) = y_1(n)\lambda(n) + y_2(1 - \lambda(n)) \quad (3.11)$$

$$e(n) = d(n) - y(n) \quad (3.12)$$

$$\lambda(n+1) = \lambda(n) + \mu_\lambda e(n)(y_1(n) + y_2(n)) \quad (3.13)$$

$\mu_\lambda$  corresponde al parámetro de adaptación del filtro en combinación *Affine* y cada algoritmo acelerador AR  $\gamma$  tienen sus propios parámetros de ajuste que son  $\alpha, \gamma$  y  $m_1$ .

El Algoritmo Acelerador Versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine* requiere aproximadamente  $12M + 9$  multiplicaciones,  $11M + 4$  sumas y  $4M$  divisiones, ya que los parámetros  $\mu_\lambda, \alpha, \gamma$  y  $m_1$  permanecen constantes durante la operación del sistema.

Concluyendo; para el funcionamiento del algoritmo se considera un escenario de identificación de sistema, el algoritmo inicia su proceso iterativo, formando los vectores de pesos  $w_1(n)$ ,  $w_2(n)$  y  $\lambda(n)$  que se muestran en las ecuaciones (3.5, 3.10 y 3.13) respectivamente, los cuales se irán actualizando. Con cada iteración se establece un rango para el factor de convergencia garantizando su estabilidad, con este factor de convergencia se logra un equilibrio entre el desajuste final y la velocidad de convergencia lo cual es característico de la combinación *Affine*, permitiendo finalmente obtener el mínimo error cuadrático medio, como se observa en la figura 3.1 en la que se aprecian las etapas de ejecución del algoritmo.

Dadas las ecuaciones planteadas se inicia con la lectura de las señales de entrada ( $x$ ), deseada ( $d$ ), el parámetro de adaptación de *Affine* ( $\mu$ ), el algoritmo AR  $\gamma$  requiere de dos (2) filtros por ser su configuración *Affine* ( $m_1$ ,  $m_2$ ,  $\alpha_1$ ,  $\alpha_2$ ,  $\gamma_1$ ,  $\gamma_2$ ), el orden de los filtros ( $M$ ), los coeficientes para los filtros ( $w$ ) y el número de muestras ( $N$ ), seguidamente se inicia un contador según el número de muestras a tomar, donde se realizan los procesos para calcular los filtros AR  $\gamma$  1 y 2, y la configuración *Affine* de donde se obtienen los datos para las señales de error ( $e$ ), salida ( $Y$ ) y parámetro de mezcla ( $\lambda$ ).

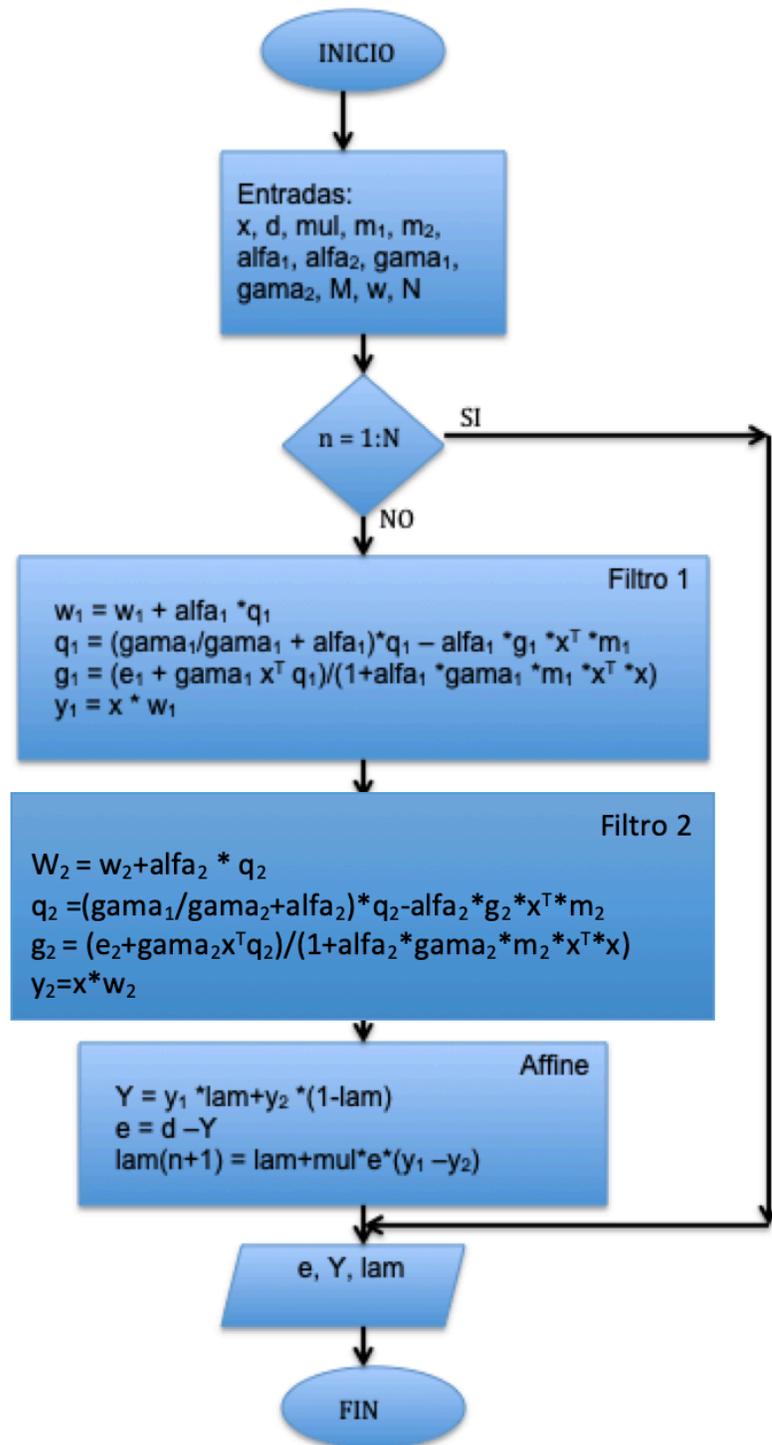


Figura 3.1. Diagrama de flujo Algoritmo Acelerador Versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine*.

### 3.2. El algoritmo acelerador versión $\gamma$ (AR $\gamma$ ) en combinación *Affine* sobre la estructura *Lattice* y su relación con otros filtros adaptativos.

El algoritmo acelerador versión  $\gamma$  (AR  $\gamma$ ) en combinación *Affine* se fundamenta en la suma ponderada de las salidas de dos filtros adaptativos sometidos a una misma entrada cuyo factor de ponderación es un parámetro calculado mediante un mecanismo adaptativo con el propósito de lograr un mejor rendimiento. La configuración *Affine* ofrece una buena adaptabilidad mejorando características como velocidad, reducción del error, una buena capacidad de seguimiento (*tracking*) y buenos resultados en desajuste. La estructura *Lattice* se caracteriza por un bloque denominado cruz o escalera el cual se va añadiendo uno a uno dependiendo del orden del filtro; está constituida por bloques y cada bloque consta de un retardo unidos a través de líneas horizontales, puntos de suma de líneas cruzadas de coeficientes, denominados coeficientes de reflexión los cuales son estimados en cada muestra con el fin de minimizar el error de predicción de adelanto o de atraso (ver figura 3.2). El presente capítulo ilustra de manera comparativa el comportamiento de otros filtros adaptativos como el LMS y NLMS sometidos a la estructura *Affine-Lattice*.

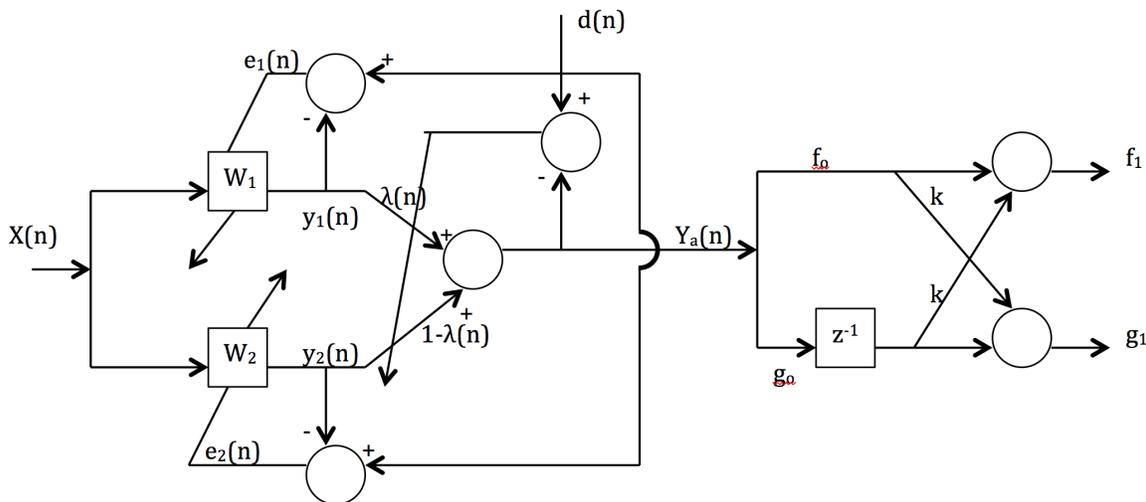


Figura 3.2. Combinación *Affine* y Estructura *Lattice*

### 3.3. Adaptabilidad de los coeficientes de reflexión de la estructura *Lattice* en la implementación de un filtro adaptativo.

Con base en las características de la estructura *Lattice*, las ecuaciones que la componen, su funcionamiento, la obtención de los predictores tanto de adelanto como de atraso, los coeficientes de predicción, además, el conocimiento de las características del algoritmo acelerador regresivo en configuración *Affine*; se tiene como propósito lograr una adecuación de la estructura *Lattice* con el algoritmo acelerador regresivo versión  $\gamma$  ( $AR\ \gamma$ ) en configuración *Affine*, para determinar el comportamiento que tienen los coeficientes de reflexión en el ámbito adaptativo y establecer su comportamiento.

Para el efecto se inicia con la adecuación de un filtro adaptativo a la estructura *Lattice*, como se ilustra en la figura 3.3, donde se puede precisar la adaptabilidad de los coeficientes de reflexión de la estructura *Lattice*.

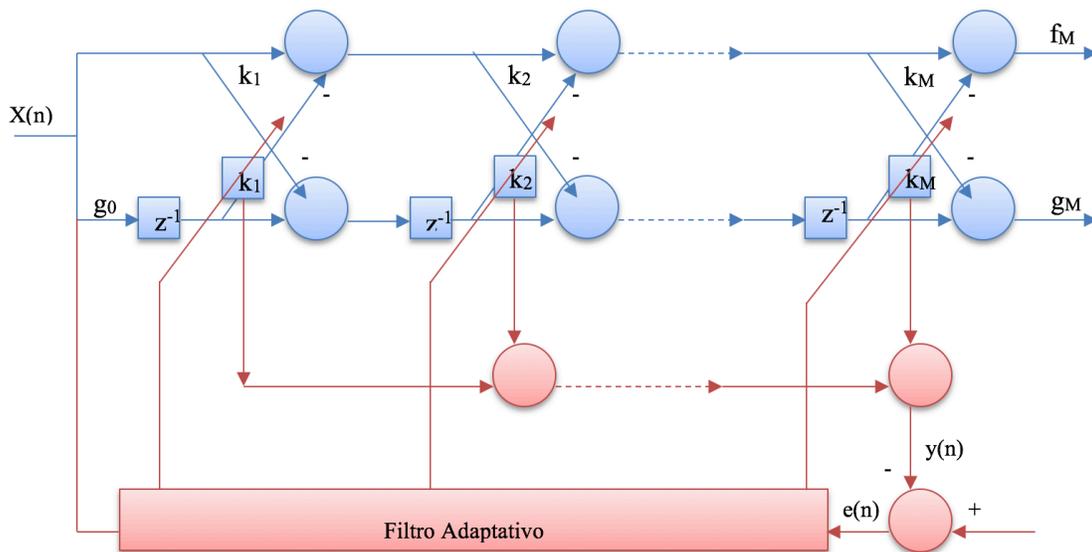


Figura 3.3. Estructura *Lattice* y filtro adaptativo.

### 3.4. Aplicación de la estructura *Lattice* al filtro $AR_\gamma$ y *Affine-AR* $\gamma$ .

Se describen a continuación las aplicaciones a la estructura *Lattice* de los filtros  $AR_\gamma$  y  $AR_\gamma$  en combinación *Affine* con ilustraciones graficas de los resultados obtenidos.

#### 3.4.1. Filtro $AR_\gamma$ -*Lattice*

Las ecuaciones que identifican al filtro  $AR_\gamma$  en estructura *Lattice* son:

$$y(n) = f_m(n)K(n) \quad (3.14)$$

$$e(n) = y(n) - d(n) \quad (3.15)$$

$$g_m(n) = \frac{e(n)\gamma f_m^T(n)q(n-1)}{1 + \alpha\gamma f_m^T(n)m_1 f_m(n)} \quad (3.16)$$

$$q(n) = \frac{\gamma}{\alpha + \gamma} [q(n-1) - \alpha g(n)m_1 f_m(n)] \quad (3.17)$$

$$K_m(\mathbf{n}) = K_m(\mathbf{n}-1) + \alpha [f_m(\mathbf{n})g_{m-1}(\mathbf{n}-1) + g_m(\mathbf{n})f_{m-1}(\mathbf{n})] \quad (3.18)$$

En la figura 3.4 se observa el comportamiento del filtro AR $\gamma$  al hacer la aplicación de la estructura *Lattice*, inicialmente se tiene la señal de error presentando una buena convergencia (fig. 3.4-a), los coeficientes propios del filtro mostrando un buen comportamiento para el sistema de identificación implementado (fig. 3.4-b) y finalmente los coeficientes de reflexión mostrando un comportamiento similar a los coeficientes propios del filtro (fig 3.4-c), siendo éste, un comportamiento singular ya que los coeficientes de reflexión siempre toman el mismo valor según la teoría presentada.

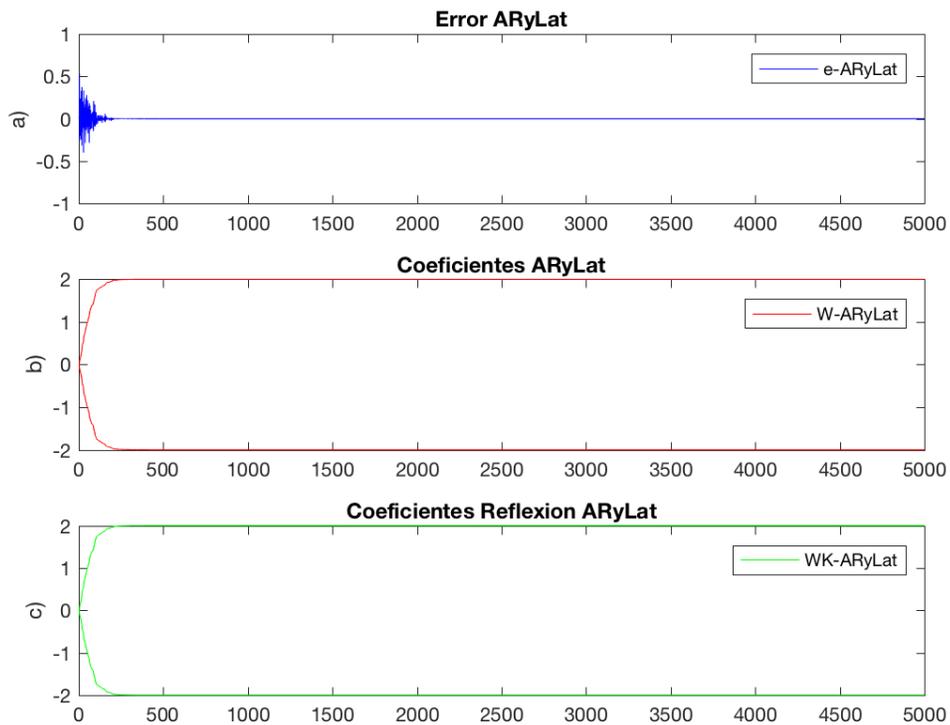


Figura 3.4. Comportamiento del filtro AR $\gamma$  y la estructura *Lattice*. a) señal de error, b) coeficientes propios del filtro y c) coeficientes de reflexión.

En la figura 3.5. Se presenta la aplicación del filtro  $AR\gamma$  y su implementación a la estructura *Lattice*, observándose el comportamiento de las señales de adelanto (azul) y de atraso (roja), es observable la buena adecuación a la estructura *Lattice* al aplicarla al filtro  $AR\gamma$  ya que la señal de atraso precede a la señal de adelanto, según lo que se espera al aplicar la estructura

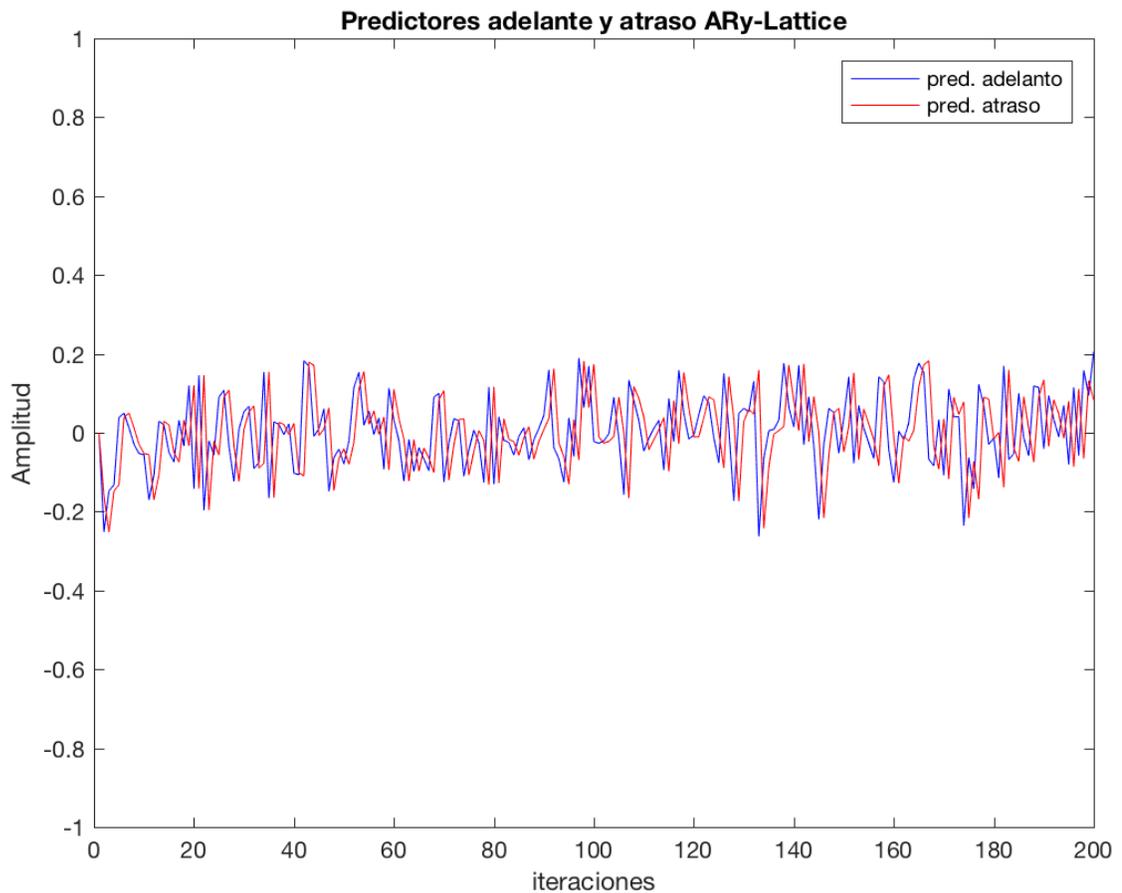


Figura 3.5 Señal de predicción de adelanto (azul) y atraso (roja) del filtro  $AR\gamma$  al aplicar la estructura *Lattice*.

La señal de error del filtro  $AR\gamma$  en su forma básica y la señal de error del filtro  $AR\gamma$  al aplicar la estructura *Lattice* se ilustra en la figura 3.6 donde se evidencia que la velocidad de convergencia presenta un comportamiento similar.

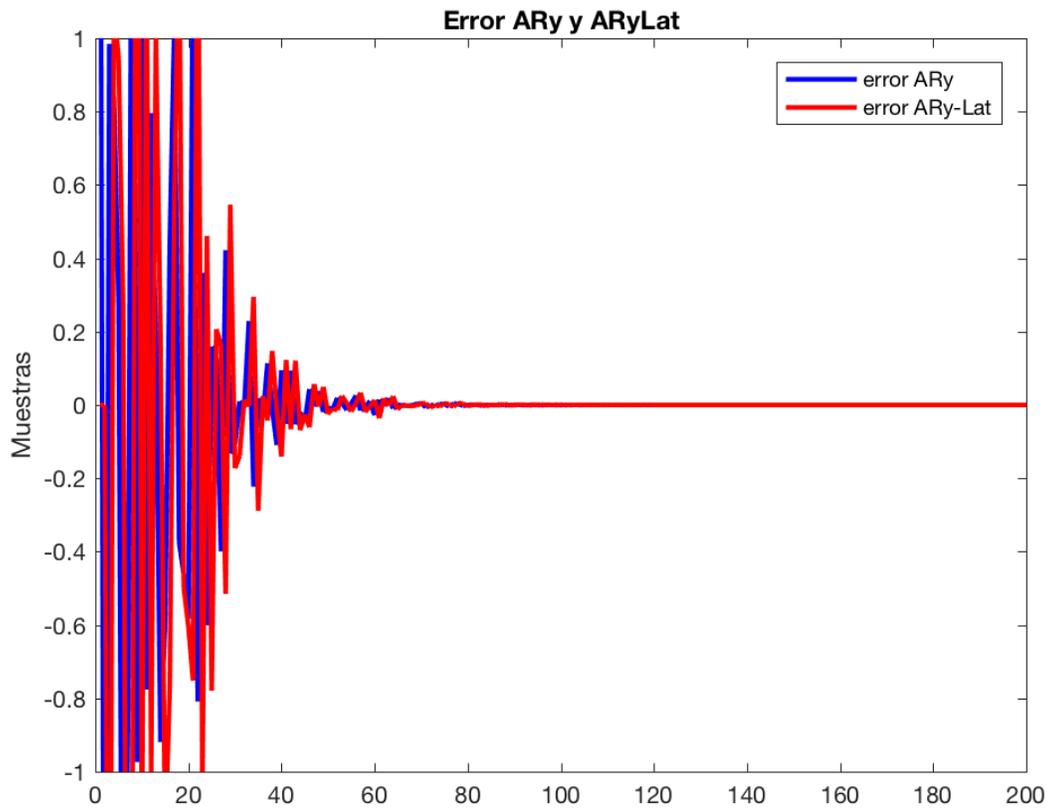


Figura 3.6. Señales de Error Filtros  $AR\gamma$  (azul) y  $AR\gamma$ -*Lattice* (rojo).

En la figura 3.7 se observa con mayor precisión que el filtro  $AR\gamma$  ofrece una mejor convergencia que cuando se aplica la estructura *Lattice* y con respecto al valor numérico del error cuadrático medio se reafirma lo dicho anteriormente por cuanto su valor es muy próximo. Como lo ilustran los valores obtenidos ( $eqm\_AR\gamma = 9.0530e-31$ ,  $eqm\_AR\gamma_{Lat} = 9.0544e-31$ ).

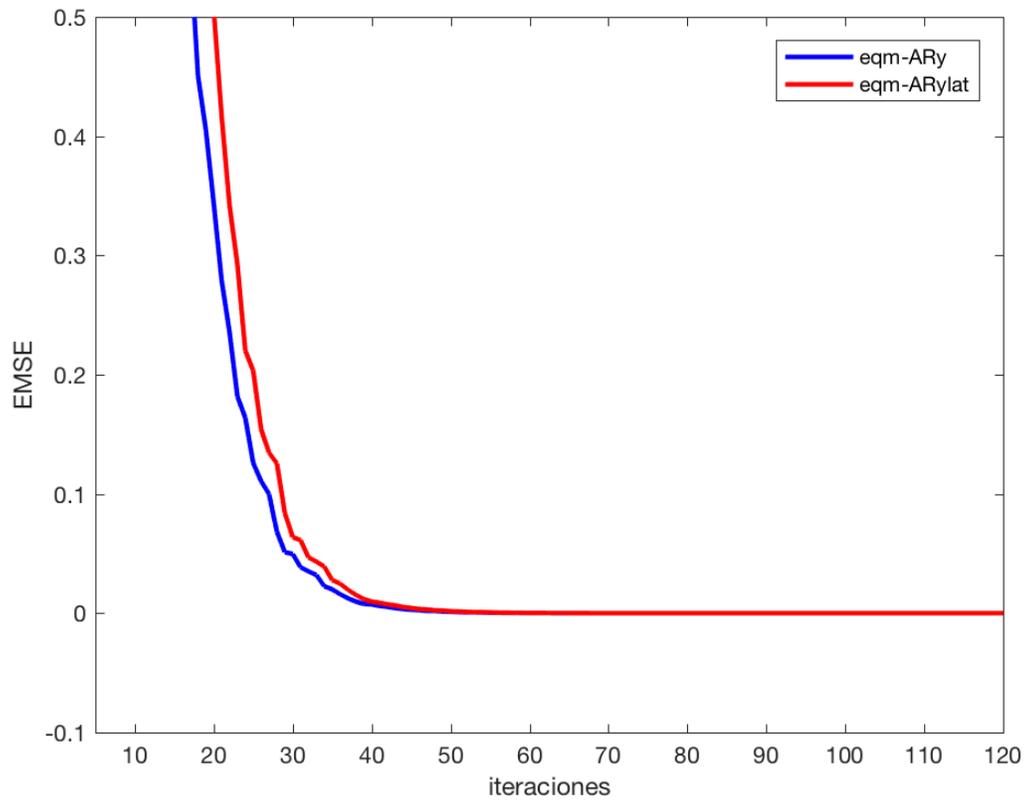


Figura 3.7. Convergencia de los filtros  $AR\gamma$  (azul) y  $AR\gamma$ -*Lattice* (rojo).

### 3.4.2. Filtro *Affine-Lattice-AR $\gamma$*

Las ecuaciones que identifican al filtro *Affine-AR $\gamma$*  en estructura *Lattice* son:

$$y(n) = f_m(n)K(n) \quad (3.19)$$

$$e(n) = y(n) - d(n) \quad (3.20)$$

$$g_m(n) = \frac{e(n)\gamma f_m^T(n)q(n-1)}{1 + \alpha\gamma f_m^T(n)m_1 f_m(n)} \quad (3.21)$$

$$q(n) = \frac{\gamma}{\alpha + \gamma} [q(n-1) - \alpha g(n)m_1 f_m(n)] \quad (3.22)$$

$$y(n) = y_1(n)\lambda(n) + y_2(1 - \lambda(n)) \quad (3.23)$$

$$e(n) = d(n) - y(n) \quad (3.24)$$

$$\lambda(n+1) = \lambda(n) + \mu_\lambda e(n)(y_1(n) + y_2(n)) \quad (3.25)$$

$$\mathbf{K}_m(n) = \mathbf{K}_m(n-1) + \alpha[\mathbf{f}_m(n)\mathbf{g}_{m-1}(n-1) + \mathbf{g}_m(n)\mathbf{f}_{m-1}(n)] \quad (3.26)$$

En la figura 3.8 se ilustra el comportamiento del filtro  $AR_{\gamma}$  al aplicarlo a la estructura *Affine-Lattice*, en la cual se observa el registro de la señal de error (fig. 3.8-a), seguido de las señales de predicción de adelanto y atraso (fig. 3.8-b) donde se evidencia el buen comportamiento que afrece la adecuación del algoritmo en estudio. En lo referente a los coeficientes de reflexión se observa que el registro correspondiente a la adecuación *Affine-Lattice* no difiere en sus valores comparativamente como se muestra en la figura 3.4-c, comentada anteriormente. Finalmente, la figura 3.8-c registra el comportamiento esperado de los coeficientes del filtro al aplicar el sistema de identificación aplicado durante la simulación.

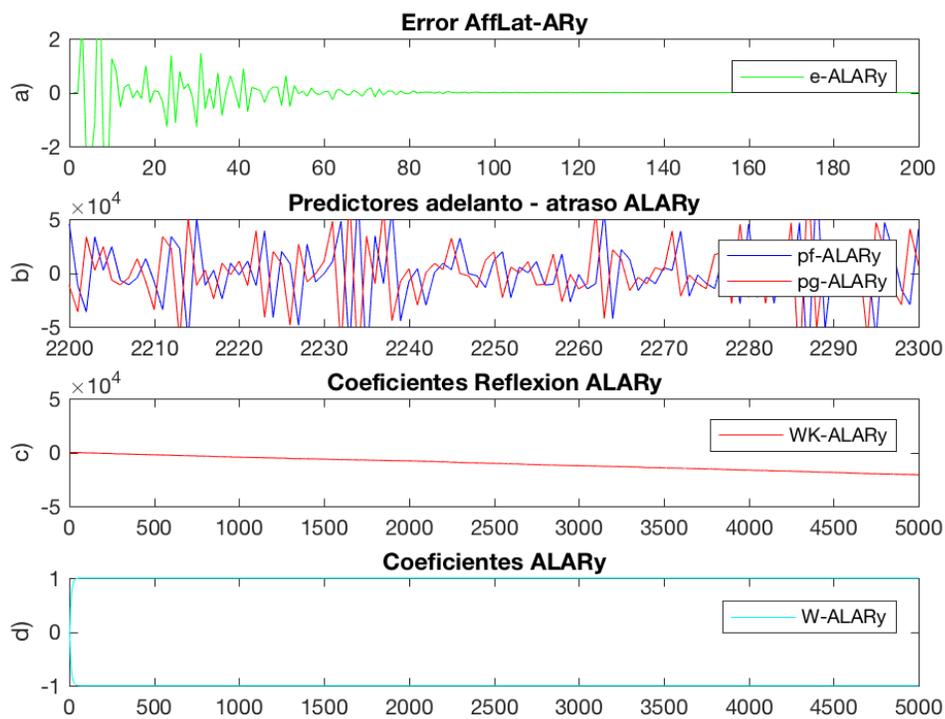


Figura 3.8. Comportamiento del filtro  $AR_{\gamma}$  y la estructura *lattice*. a) señal de error, b) coeficientes propios del filtro y c) coeficientes de reflexión.

El registro observado en la figura 3.9 permite concluir una estabilidad mucho más rápida al aplicar la adecuación *Affine-Lattice* al filtro  $AR_\gamma$  comparativamente con la aplicación del filtro  $AR_\gamma$  en *Lattice* y el filtro  $AR_\gamma$  en su forma convencional.

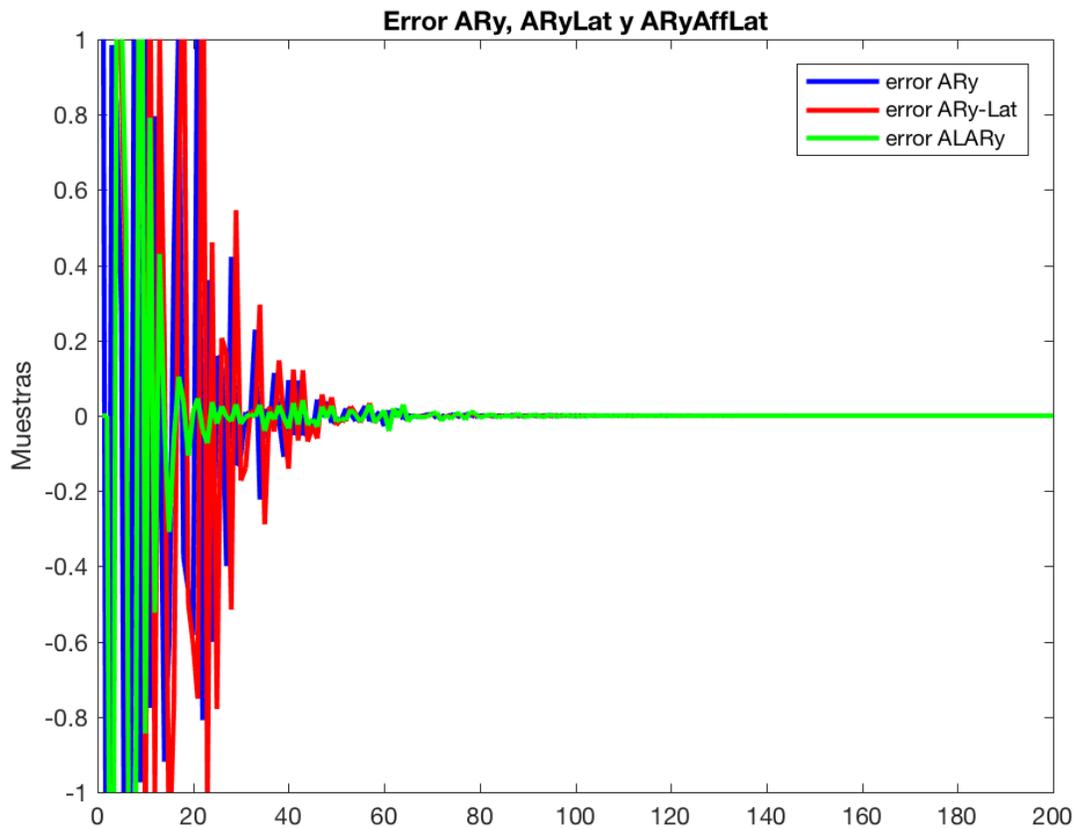


Figura 3.9. Señales de Error Filtros  $AR_\gamma$  (azul),  $AR_\gamma$ -*Lattice* (rojo) y  $AR_\gamma$ -*Affine Lattice* (verde).

Lo observado en el registro de la figura 3.10 permite afirmar que el filtro  $AR_\gamma$  *Affine-Lattice* ofrece una mejor convergencia si se compara con los resultados obtenidos de los filtros  $AR_\gamma$ -*Lattice* y  $AR_\gamma$  básico. Los valores numéricos para el error cuadrático medio responden a  $eqm\_AR_\gamma = 8.0413e-31$ ,  $eqm\_AR_\gamma Lat = 8.0166e-31$ ,  $eqm\_AffLatAR_\gamma = 4.4517e-31$  donde es evidente que el menor valor es el obtenido al aplicar la adecuación *Affine-Lattice*.

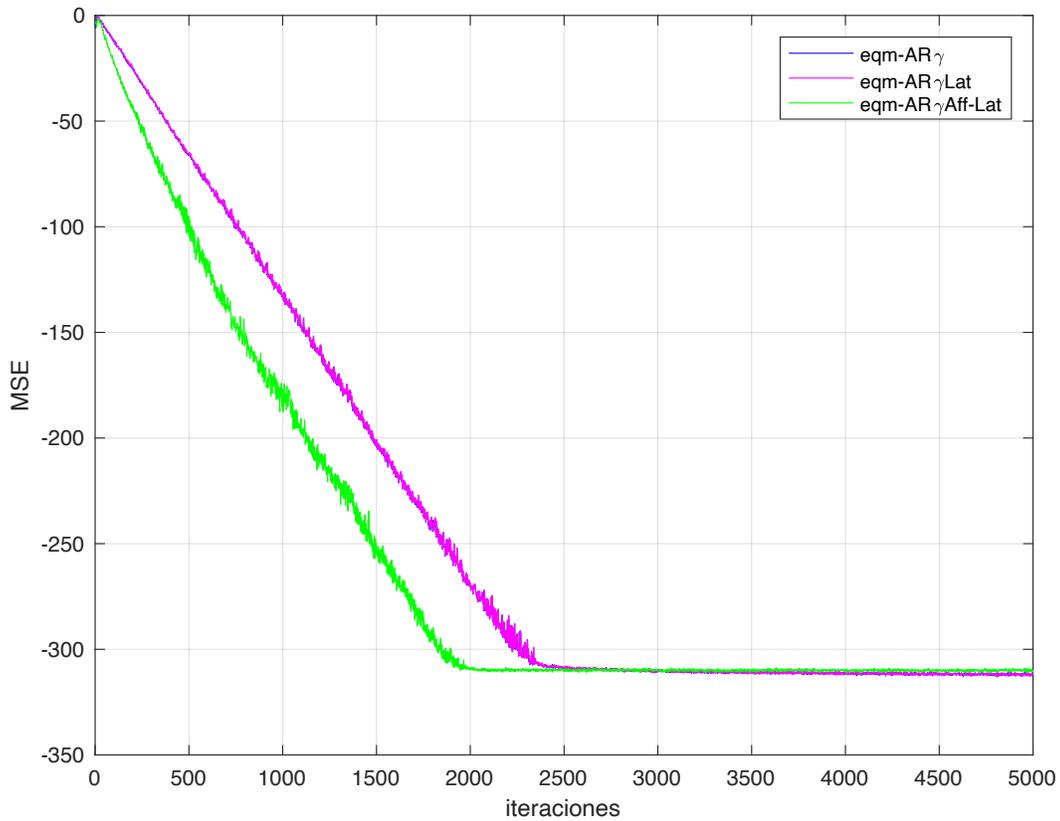


Figura 3.10. Convergencia de los filtros  $AR_\gamma$  (azul),  $AR_\gamma$ -*Lattice* (magenta) y  $AR_\gamma$ -*Affine Lattice* (verde).

### 3.5. Aplicación de la estructura *Lattice* a los filtros LMS y NLMS.

Se realiza a continuación la adecuación de la estructura *Lattice* a los filtros LMS y NLMS ilustrando gráficamente su adaptación.

#### 3.5.1. Filtro LMS-*Lattice*.

Un filtro adaptativo básico para realizar esta adecuación es el LMS cuyas ecuaciones son:

$$e_0(n) = d(n) - g_0(n)w_0(n) \quad (3.27)$$

$$e_m(n) = e_{m-1}(n) - g_m(n)w_m(n) \quad (3.28)$$

$$w_m(n+1) = w_m(n) + \mu e_m(n)g_m(n) \quad (3.29)$$

Ahora bien, las ecuaciones que describe la estructura *Lattice* son las siguientes:

$$f_m(n+1) = f_{m-1}(n+1) + K_m(n+1)g_{m-1}(n) \quad (3.30)$$

$$g_m(n+1) = g_{m-1}(n) + K_m(n+1)f_{m-1}(n+1) \quad (3.31)$$

$$f_0(n) = g_0(n) = x(n) \quad (3.32)$$

La estimación de los coeficientes de reflexión aplicando los métodos convencionales de algoritmos adaptativos, resulta de la aplicación de la estructura *Lattice* al algoritmo LMS, para el efecto, los coeficientes de reflexión se calculan como coeficientes de correlación cruzada entre los predictores de adelanto y atraso en cada etapa del filtro *Lattice* adaptativo.

La configuración del filtro *Lattice* ilustrada en la figura 2.2, se puede expresar de forma general como:

$$y(n) = f_m(n) = f_0(n) - \sum_{m=1}^M K_m(n)g_{m-1}(n-1) \quad (3.33)$$

En donde  $f_m(n)$  es la salida del filtro *Lattice* con sus componentes  $f_0(n)$  que es un valor de predicción y la sumatoria de los coeficientes de reflexión  $K_m(n)$  por el error de predicción de atraso  $g_{m-1}(n-1)$ . El segundo componente de la ecuación (3.33) se puede considerar como un filtro FIR de error de predicción como se ilustró en las secciones anteriores. Con base en las metodologías utilizadas en los tratamientos de los filtros FIR en lo relativo a la estimación de coeficientes, para los filtros *Lattice* el procedimiento resulta similar.

Así mismo, al minimizar el error de predicción de adelanto final se obtendrían los coeficientes de reflexión usando la siguiente estimación al hacer uso del algoritmo LMS:

$$K_m(n+1) = K_m(n) + \mu[f_m(n)g_{m-1}(n-1) + g_m(n)f_{m-1}(n)], \quad 0 < m \leq M \quad (3.34)$$

$$e_0(n) = d(n) - g_0(n)w_0(n) \quad (3.35)$$

$$e_m(n) = e_{m-1}(n) - g_m(n)w_m(n) \quad (3.36)$$

$$w_m(n+1) = w_m(n) + \mu e_m(n)g_m(n) \quad (3.37)$$

El método de Burg garantiza que los coeficientes de reflexión sean menores a la unidad.

La figura 3.11 ilustra el comportamiento del filtro LMS en estructura *Lattice* observándose la señal de error (fig. 3.11-a), coeficientes propios del filtro LMS-*Lattice* (fig. 3.11-b) y finalmente los coeficientes de reflexión que permiten concluir la generación de una sola señal (fig. 3.11-c).

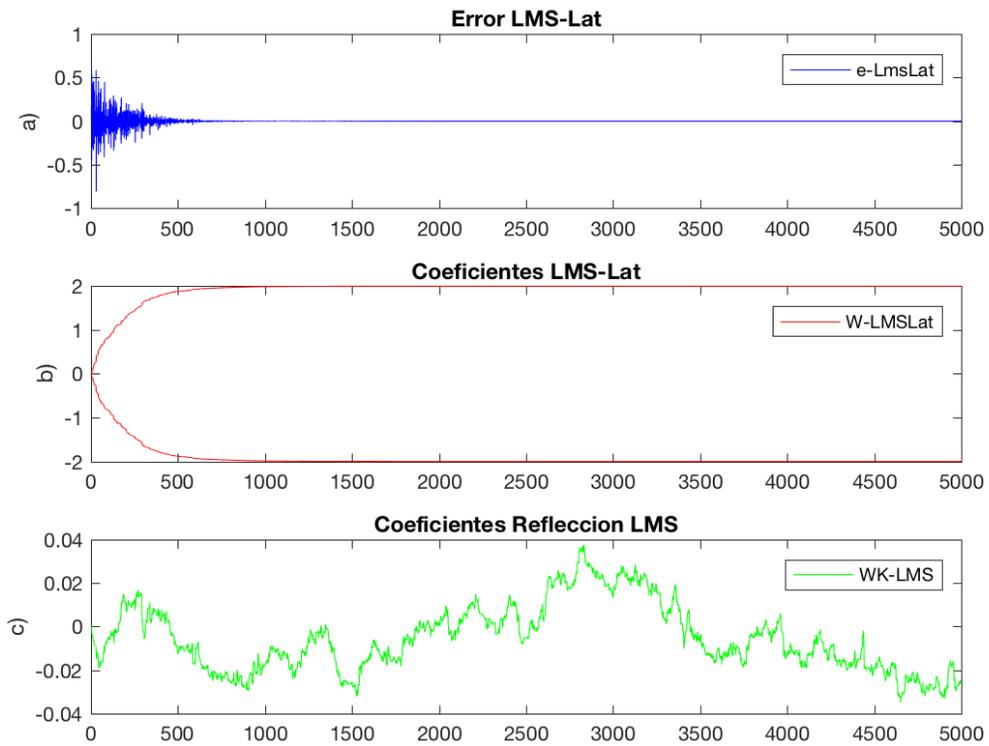


Figura 3.11. Filtro LMS-*Lattice*. a) error LMS-*Lattice*, b) coeficientes filtro LMS-*Lattice* y c) coeficientes de reflexión LMS-*Lattice*.

La figura 3.12 registra el comportamiento de la señal de error del filtro LMS en su forma básica y al aplicar la estructura *Lattice*. Se concluye que ofrecen un comportamiento similar con respecto a la convergencia, característica que se observa igualmente en la figura 3.13 donde se destaca el comportamiento del error cuadrático medio del filtro LMS en estructura *Lattice* y el filtro LMS en su diseño convencional. El filtro en estudio evidencia una buena adecuación no obstante que su convergencia no ofrece mejora significativa.

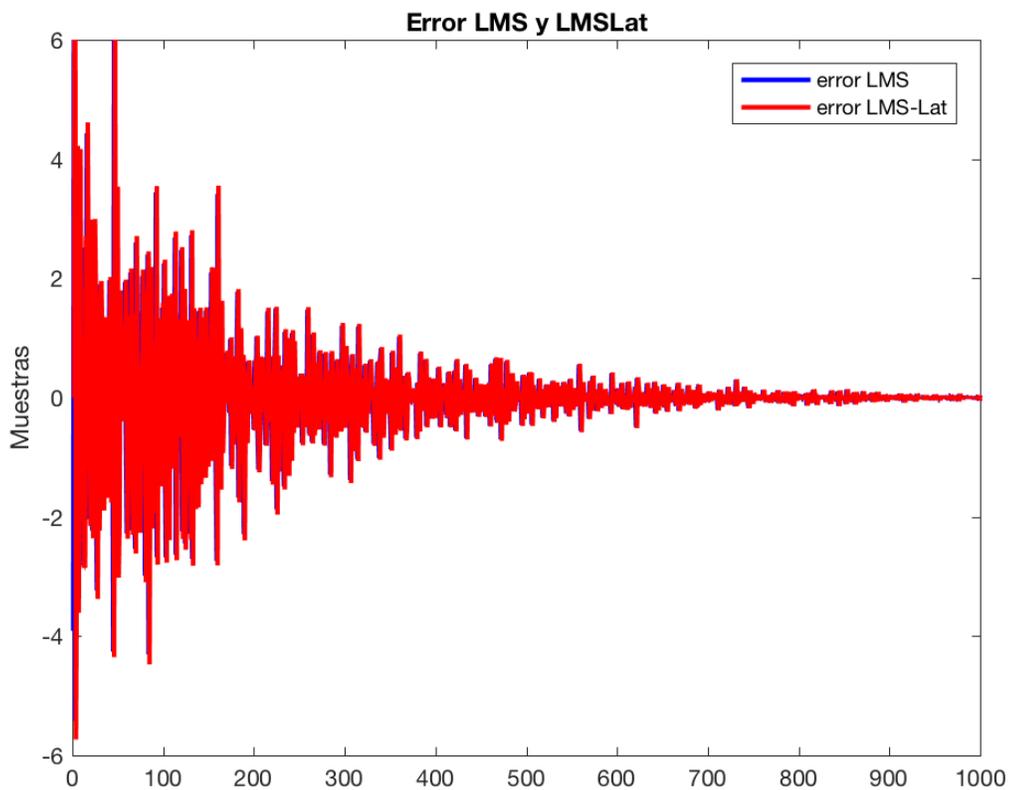


Figura 3.12. Error Filtro LMS (rojo) y error filtro LMS-*Lattice* (azul).

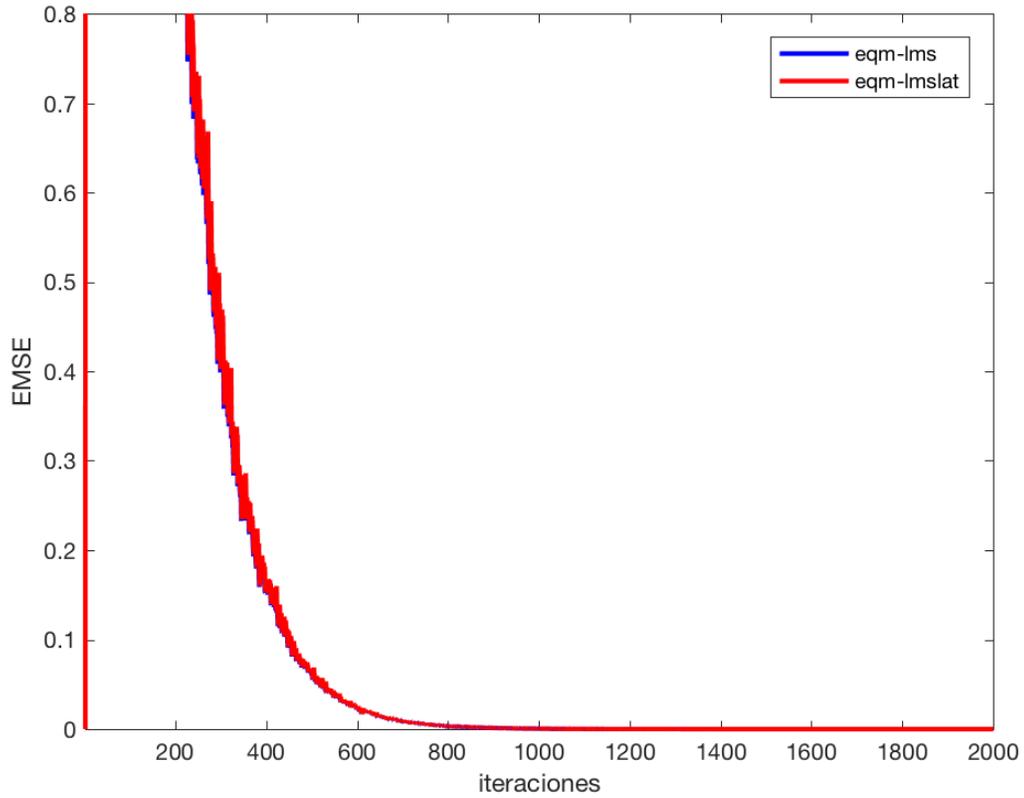


Figura 3.13. Convergencia filtro LMS (azul) y LMS-Lattice (rojo).

### 3.5.2. Filtro LMS *Affine-Lattice*

A continuación se muestran las ecuaciones que describen el algoritmo LMS al adecuarlo a la estructura *Affine-Lattice*.

$$e_0(n) = d(n) - g_0(n)w_0(n) \quad (3.38)$$

$$e_m(n) = e_{m-1}(n) - g_m(n)w_m(n) \quad (3.39)$$

$$w_m(n+1) = w_m(n) + \mu e_m(n)g_m(n) \quad (3.40)$$

$$y(n) = y_1(n)\lambda(n) + y_2(1 - \lambda(n)) \quad (3.41)$$

$$e(n) = d(n) - y(n) \quad (3.42)$$

$$\lambda(n+1) = \lambda(n) + \mu_\lambda e(n)(y_1(n) + y_2(n)) \quad (3.43)$$

$$\mathbf{K}_m(n+1) = \mathbf{K}_m(n) + \mu[\mathbf{f}_m(n)\mathbf{g}_{m-1}(n-1) + \mathbf{g}_m(n)\mathbf{f}_{m-1}(n)], \quad (3.44)$$

$$\mathbf{0} < m \leq M$$

La aplicación de la estructura *Affine-Lattice* al filtro LMS se ilustra en la figura 3.14 con: Señal de error (fig.3.14-a), variable *Affine* (fig.3.14-b), coeficientes de reflexión (fig.3.14-c) y coeficientes propios del filtro (fig.3.14-d).

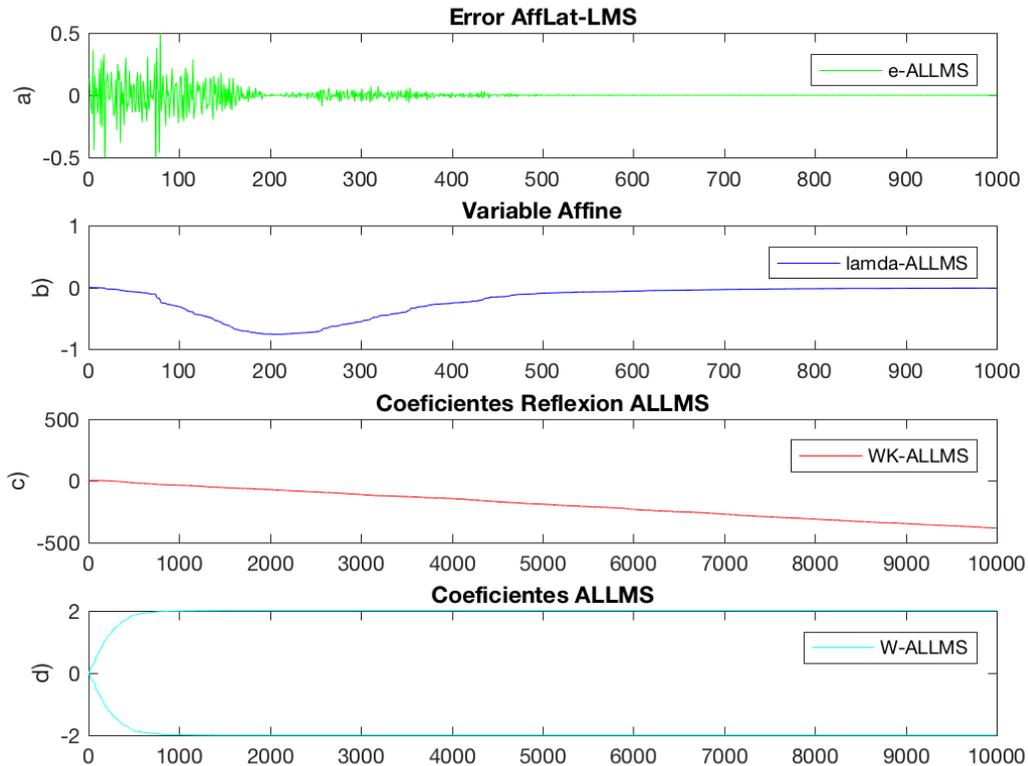


Figura 3.14. Filtro LMS *Affine-Lattice*. a) error LMS *Affine-Lattice*, b) Variable combinación *Affine*, c) coeficientes de reflexión filtro LMS *Affine-Lattice* y d) coeficientes LMS *Affine-Lattice*.

Al realizar la adecuación del filtro LMS en combinación *Affine* y luego al aplicar la estructura *Lattice* se observa una mejora significativa respecto a la velocidad de convergencia como se evidencia en la figura 3.15.

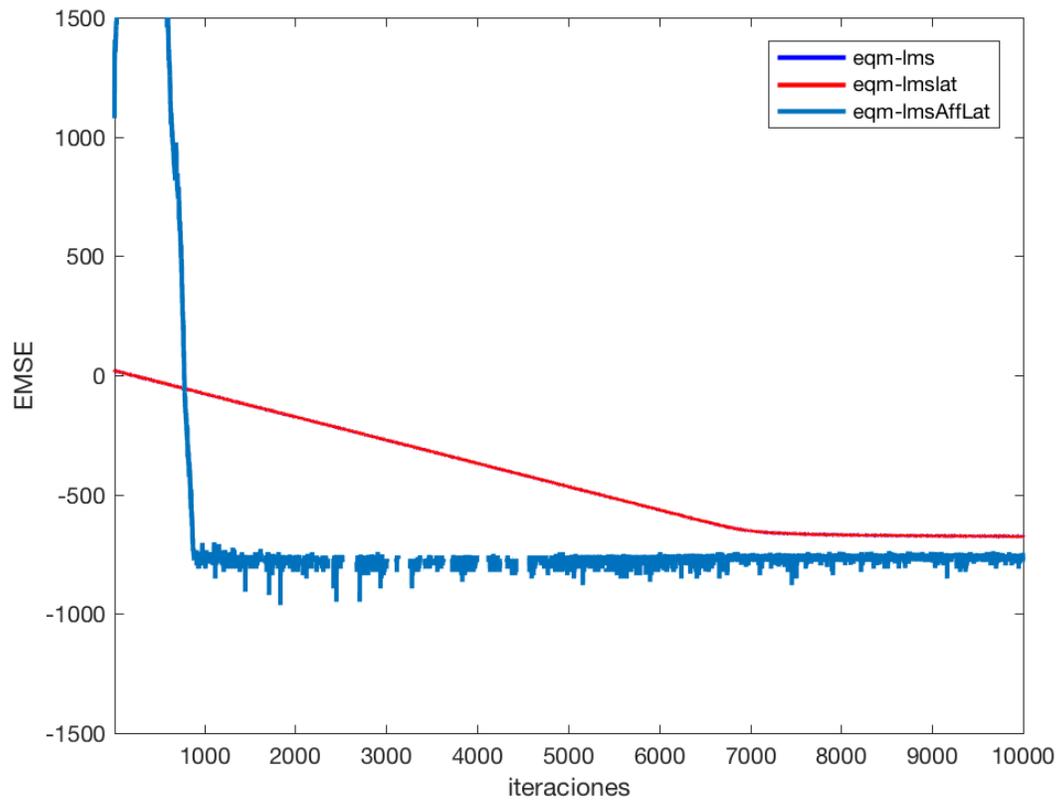


Figura 3.15. Error cuadrático medio del filtro LMS (azul) al aplicar la estructura *Lattice* (rojo) y con la aplicación de la combinación *Affine* y estructura *Lattice* (azul claro).

Respecto a los coeficientes propios del filtro LMS se puede observar que al aplicar el sistema de identificación se obtiene la respuesta esperada (fig. 4.13-d y fig.4.10-b) y con respecto a los coeficientes de reflexión de los filtros LMS-*Lattice* y LMS-*Affine-Lattice*, el registro presenta el mismo valor luego de la aplicación de la combinación *Affine* (Ver figuras 4.10-c y 4.13-c).

### 3.5.3. Filtro NLMS-Lattice

Al hacer uso del filtro NLMS y aplicar la estructura *Lattice* los coeficientes de reflexión están dados por la siguiente ecuación:

$$K_m(n+1) = K_m + \mu \frac{f_m(n)g_m(n-1)}{(g_{m-1}(n-1))^2} \quad (3.45)$$

La figura 3.16 registra el comportamiento del filtro NLMS adecuado a la estructura *Lattice* donde se observa la señal de error (fig. 3.16-a), los coeficientes propios del filtro (fig. 3.16-b) y coeficientes de reflexión (fig. 3.16-c).

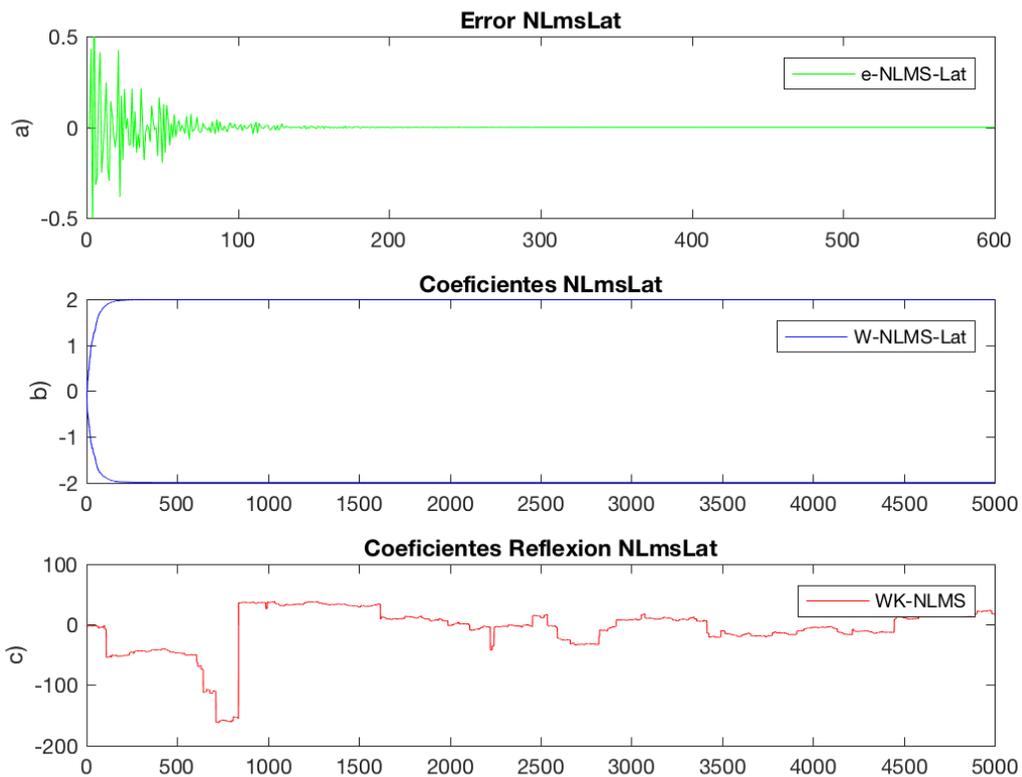


Figura 3.16. Filtro NLMS-Lattice. a) error NLMS-Lattice, b) coeficientes filtro NLMS-Lattice y c) coeficientes de reflexión NLMS-Lattice.

La figura 3.17 ilustra la comparación del comportamiento similar de la señal de error que ofrecen los filtros NLMS y NLMS en estructura *Lattice*.

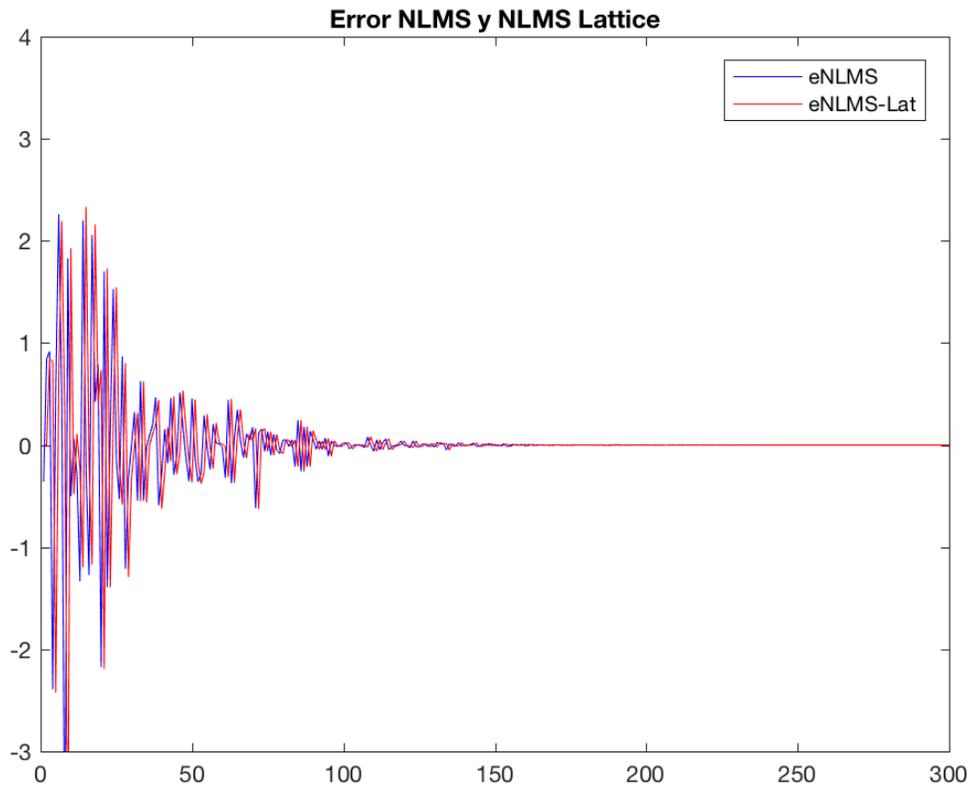


Figura 3.17. Error NLMS (azul) y Error NLMS-*Lattice* (rojo)

Cuando se aplica la estructura *Lattice* al filtro NLMS se observa una buena adecuación (ver figura 3.16) aunque su velocidad de convergencia no es significativa como se ilustra en la figura 3.18.

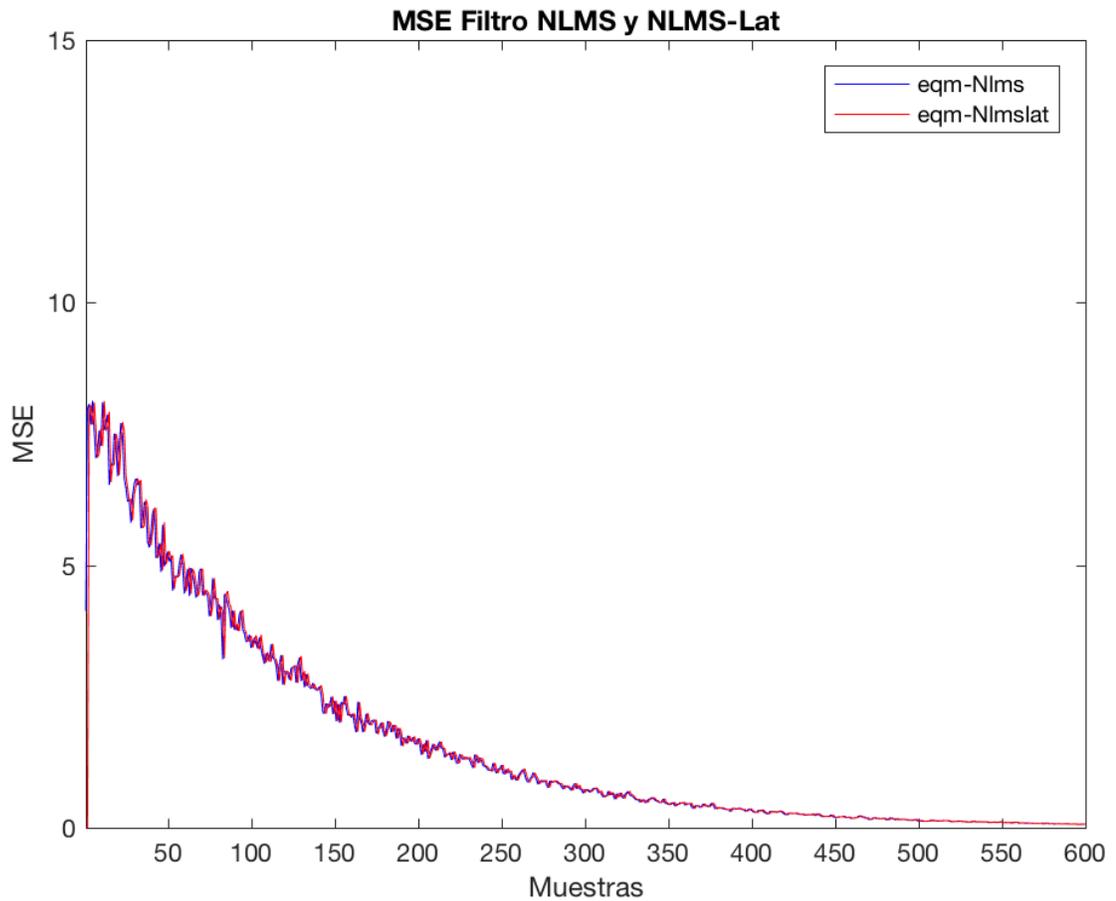


Figura 3.18. Convergencia filtro NLMS (azul) y NLMS-*Lattice* (rojo).

### 3.5.4. Filtro NLMS *Affine-Lattice*

A continuación se describen las ecuaciones características al aplicar la estructura *Affine-Lattice* al filtro NLMS.

$$y(n) = y_1(n)\lambda(n) + y_2(1 - \lambda(n)) \quad (3.46)$$

$$e(n) = d(n) - y(n) \quad (3.47)$$

$$\lambda(n+1) = \lambda(n) + \mu_\lambda e(n)(y_1(n) + y_2(n)) \quad (3.48)$$

$$K_m(n+1) = K_m + \mu \frac{f_m(n)g_m(n-1)}{(g_{m-1}(n-1))^2} \quad (3.49)$$

Los registros presentados en la figura 3.19 obedecen al comportamiento del filtro NLMS en configuración *Affine-Lattice* en sus variables de señal de error (fig. 3.19-a), predictores de adelanto y atraso (fig. 4.18-b), coeficientes propios del filtro (fig. 3.19-c) y coeficientes de reflexión (fig. 3.19-d).

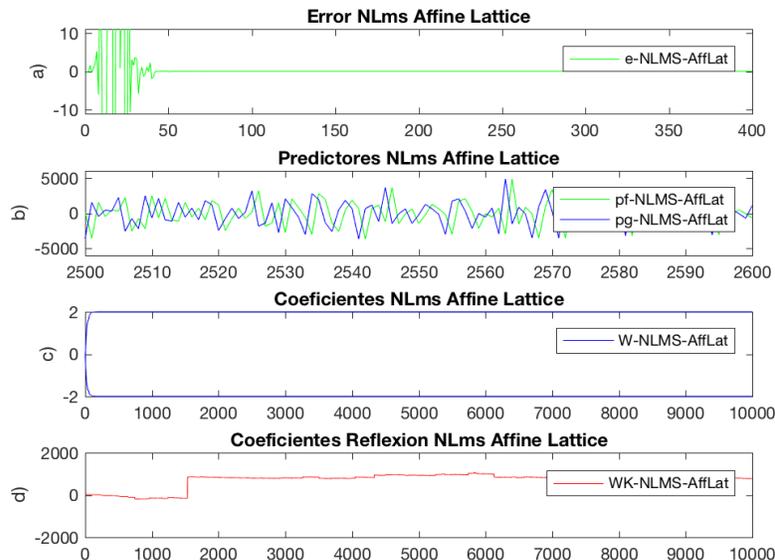


Figura 3.19. Filtro NLMS *Affine-Lattice*. a) error NLMS *Affine-Lattice*, b) Predictores de adelanto y atraso, c) Coeficientes filtro NLMS *Affine-Lattice* y d) coeficientes de reflexión NLMS *Affine-Lattice*.

La convergencia de la señal de error para los filtros NLMS, NLMS-*Lattice* y NLMS-*Affine-Lattice* se puede observar en la figura 3.20 destacando el mejor comportamiento dado por el filtro NLMS-*Affine-Lattice* en comparación con los filtros NLMS y NLMS-*Lattice*.

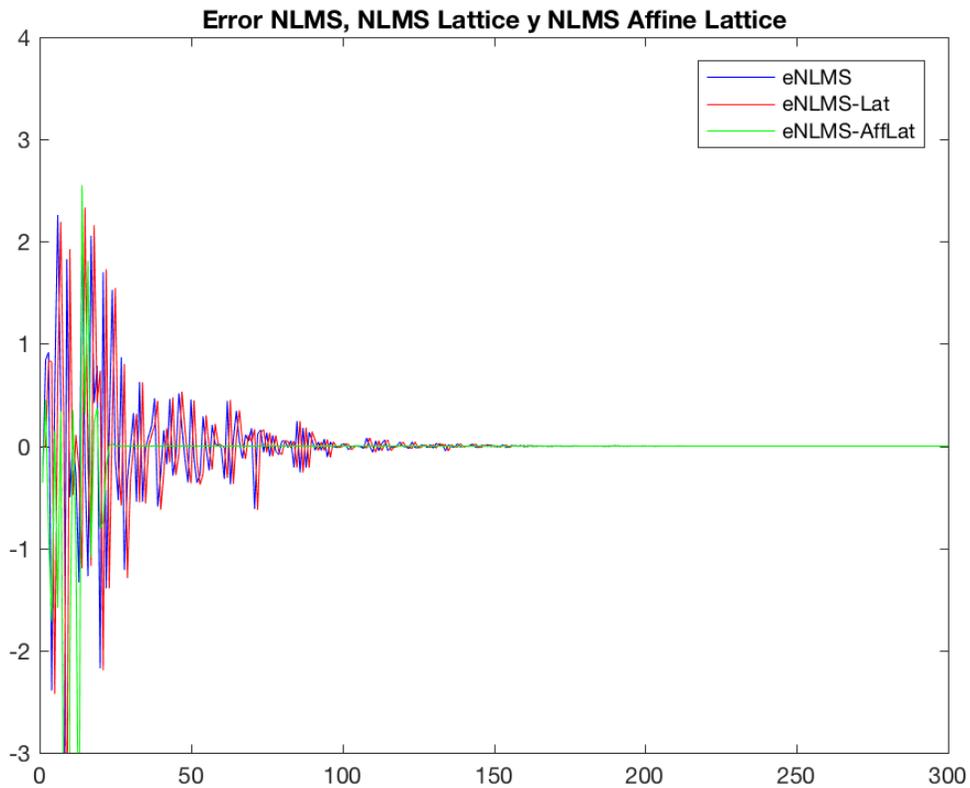


Figura 3.20. Error NLMS (azul), error NLMS Lattice (rojo) y Error NLMS *Affine-Lattice* (verde)

En la figura 3.21 se observa la aplicación de la combinación *Affine* y la respuesta para los filtros NLMS-*Affine* y NLMS-*Lattice* los resultados son prácticamente iguales sin notarse una diferencia significativa entre estas dos señales (Azul y Rojo respectivamente). Al aplicar la estructura *Affine-Lattice* al filtro NLMS se obtiene una convergencia más rápida, que solo aplicando la estructura *Lattice*.

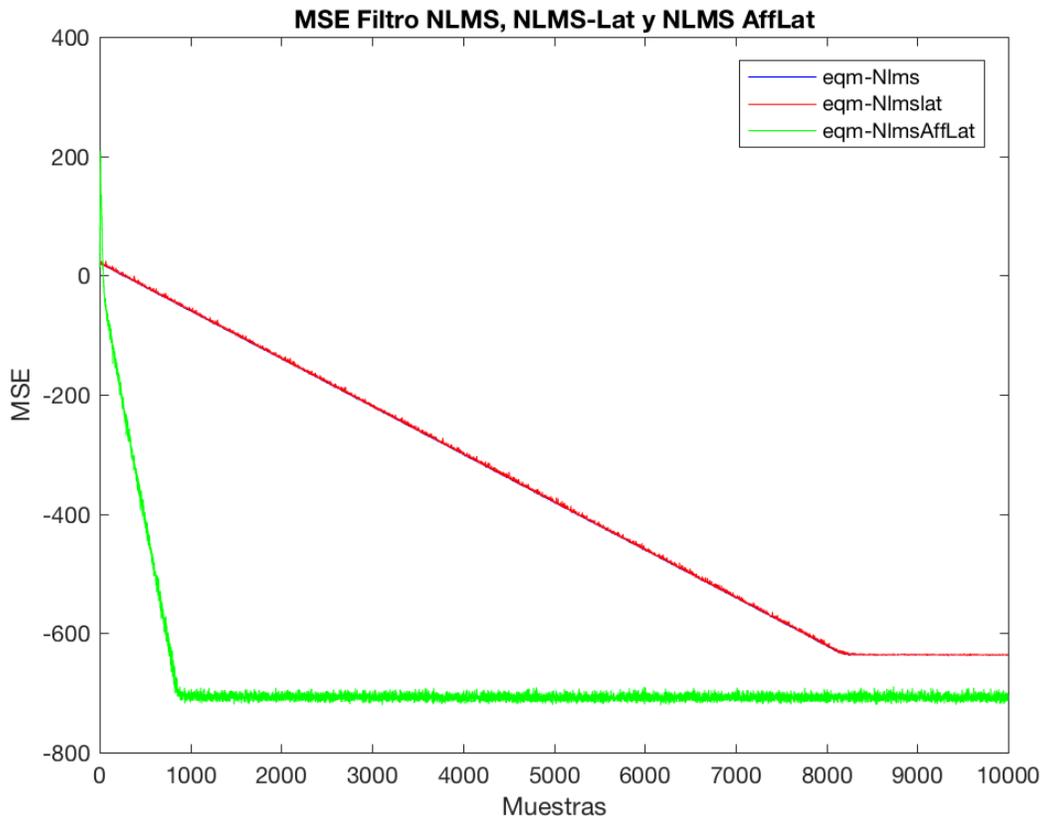


Figura 3.21. Convergencia de los filtros NLMS (azul), NLMS-*Lattice* (rojo) y NLMS *Affine-Lattice* (verde).

La figura 3.22 permite deducir que con respecto a la convergencia el filtro  $AR\gamma$  *Affine-Lattice* ofrece el mejor comportamiento al relacionarlo con los filtros LMS y NLMS en configuración *Affine-Lattice*.

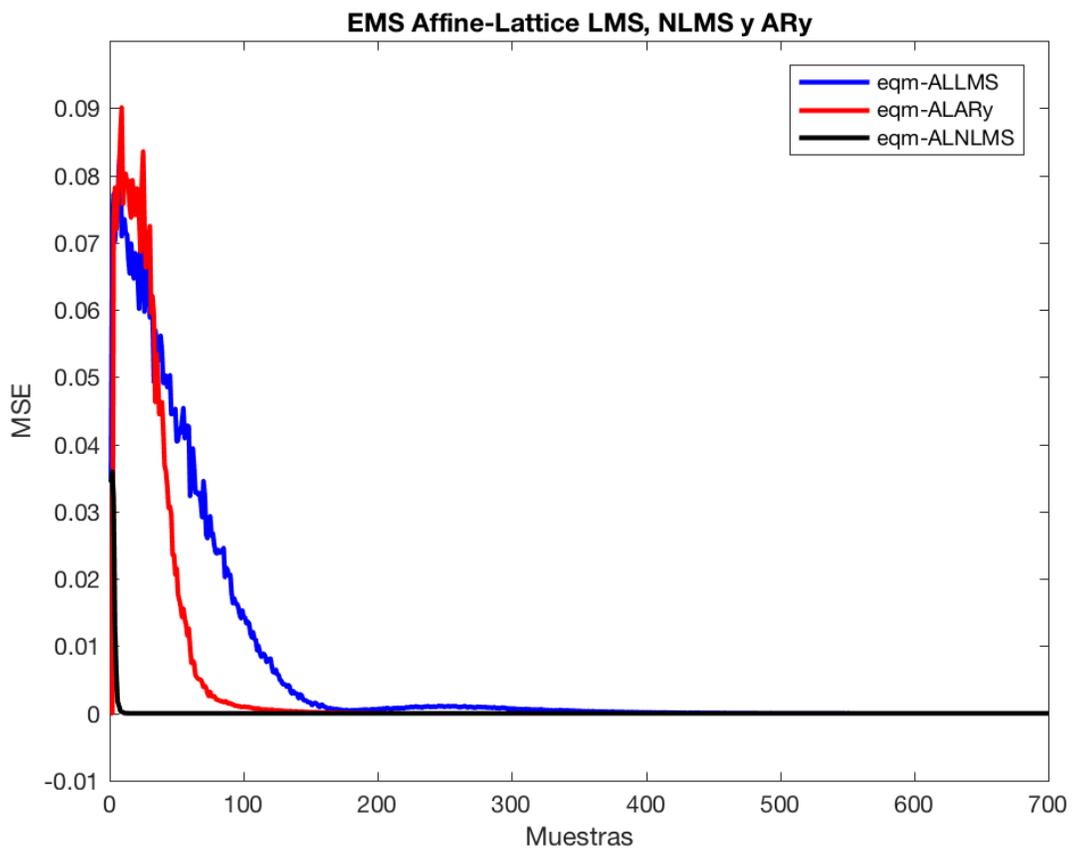


Figura 3.22. Convergencia de los filtros LMS *Affine-Lattice* (azul),  $AR\gamma$  *Affine-Lattice* (rojo) y NLMS *Affine-Lattice* (negro).

Después de la aplicación de la estructura *Lattice* y la combinación *Affine-Lattice* a los filtros LMS, NLMS y AR $\gamma$  y de observar el comportamiento de la señal de error cuadrático medio en las figuras 3.22 y 3.23 se concluye que la convergencia más rápida se obtiene con el filtro AR $\gamma$  al compararlo con el filtro LMS, hay rangos donde el filtro NLMS es mejor como se aprecia en la figura 3.22

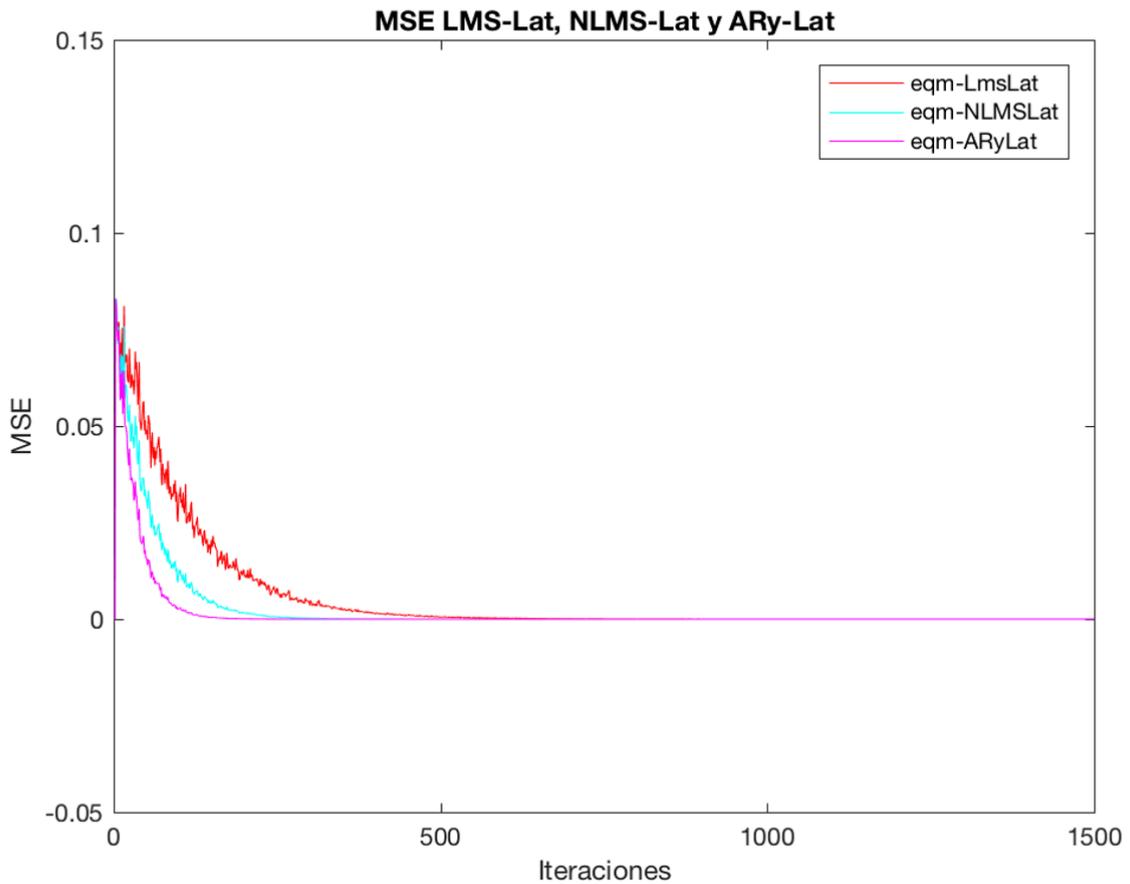


Figura 3.23. Convergencia de los filtros LMS *Lattice* (rojo), NLMS *Lattice* (cian) y AR $\gamma$  *Lattice* (magenta).

### 3.6. Análisis de Tracking

Se examina el funcionamiento de los algoritmos en un ambiente no estacionario. El *tracking* es un fenómeno de estado estacionario que debe contrastarse con la convergencia como fenómeno transitorio. Para que un filtro adaptativo realice su *tracking*, inicialmente debe pasar de un estado transitorio a un estado estable y debe haber un ajuste continuo de los parámetros del filtro. En general, la velocidad de convergencia y el *tracking* son propiedades diferentes del algoritmo. Un algoritmo de filtrado adaptativo con buenas propiedades de convergencia no necesariamente posee una buena capacidad de seguimiento (*tracking*) y viceversa.

A continuación, se describe un sistema de identificación para un modelo de variación en el tiempo siguiendo un proceso de Markov el cual se describe en la figura 3.24, que seguidamente se usa como base para evaluar el rendimiento de la capacidad de seguimiento (*tracking*) de los algoritmos LMS, NLMS y AR $\gamma$  que operan en un ambiente no estacionario.

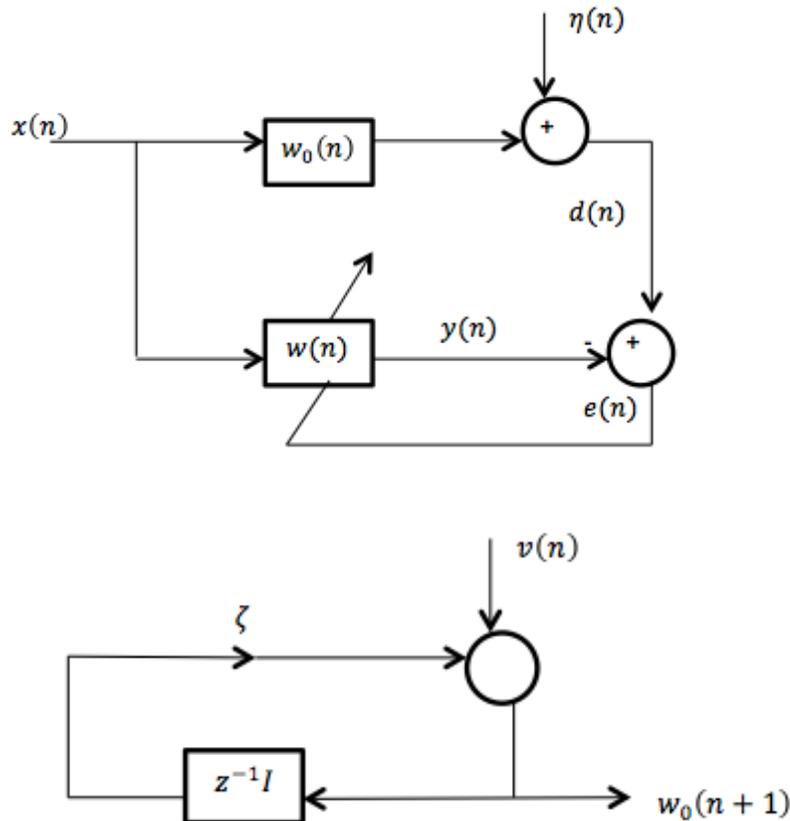


Figura 3.24. Modelo del Proceso de Markov y sistema de identificación

El proceso de Markov se describe de forma vectorial así:

$$w_0(n + 1) = \zeta w_0(n) + v(n) \quad (3.50)$$

$\zeta$  = parámetro fijo (aproximado a 1)

$v(n)$  = vector de ruido con media cero.

$w_0(n)$  = coeficientes del sistema a ser identificado

$\eta(n)$  = ruido blanco gaussiano con media cero y varianza  $\sigma_\eta^2$

$x(n)$ ,  $v(n)$  y  $\eta(n)$  = estadísticamente independientes.

Para verificar el análisis se efectúan diferentes experiencias con las siguientes condiciones:

$L$  = longitud de la señal = 30000

$\sigma_v^2$  = varianza del proceso de Markov = 0.01

$\sigma_x^2$  = varianza de la señal de entrada = 0.1

$\sigma_\eta^2$  = varianza del ruido de medida = 0.05

**Tabla 3.1. Error Cuadrático Medio para diferentes valores de  $\zeta$  del filtro ARy aplicando estructura *Lattice* y *Affine-Lattice*.**

$\zeta$	Eqm ARy	Eqm ARy <i>Lattice</i>	Eqm ARy <i>Affine - Lattice</i>
0.99	1.2008e-31	1.1955e-31	1.1028e-31
0.9998	1.4038e-31	1.4035e-31	1.2194e-31
1	1.3879e-31	1.3953e-31	1.2215e-31

seguidamente se observan los valores para el error cuadrático medio de los filtros LMS , LMS en estructura *Lattice* y LMS en estructura *Affine-Lattice*, tomando diferentes valores de zeta ( $\zeta$ ) siendo el mejor resultado cuando  $\zeta = 0.99$ , en la tabla 3.2 se registran estos valores.

**Tabla 3.2. Error Cuadrático Medio para diferentes valores de  $\zeta$  del filtro LMS aplicando estructura *Lattice* y *Affine-Lattice*.**

$\zeta$	Eqm LMS	Eqm LMS <i>Lattice</i>	Eqm LMS <i>Affine - Lattice</i>
0.99	6.4837e-13	6.6236e-13	1.4458e-14
0.9998	6.8851e-13	7.0190e-13	1.5335e-14
1	7.1655e-13	7.3095e-13	1.5959e-14

Posteriormente se indican los valores para el error cuadrático medio de los filtros , NLMS, NLMS en estructura *Lattice* y NLMS en estructura *Affine-Lattice*, variando el valor de zeta ( $\zeta$ ) entre 0.99 a 1 obteniendo el mejor resultado cuando  $\zeta = 0.99$ , como se observa en la tabla 3.3.

**Tabla 3.3. Error Cuadrático Medio para diferentes valores de  $\zeta$  del filtro NLMS aplicando estructura *Lattice* y *Affine-Lattice*.**

$\zeta$	Eqm NLMS	Eqm NLMS <i>Lattice</i>	Eqm NLMS <i>Affine - Lattice</i>
0.99	0.0796	30.8524	6.8844e-34
0.9998	0.0790	137.1053	6.6239e-34
1	0.0794	4.6045e+04	6.7269e-34

Para los valores de  $\zeta$  tomados se observa un mínimo valor para el error cuadrático medio con el filtro NLMS al aplicarlo a la combinación *Affine* y la estructura *Lattice*.

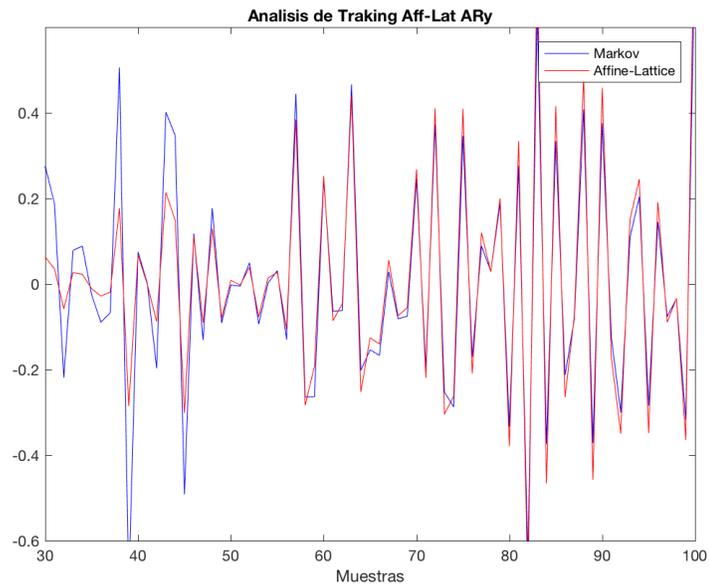


Figura 3.25. Algoritmo ARy en combinación *Affine* con estructura *Lattice* y proceso de Markov.

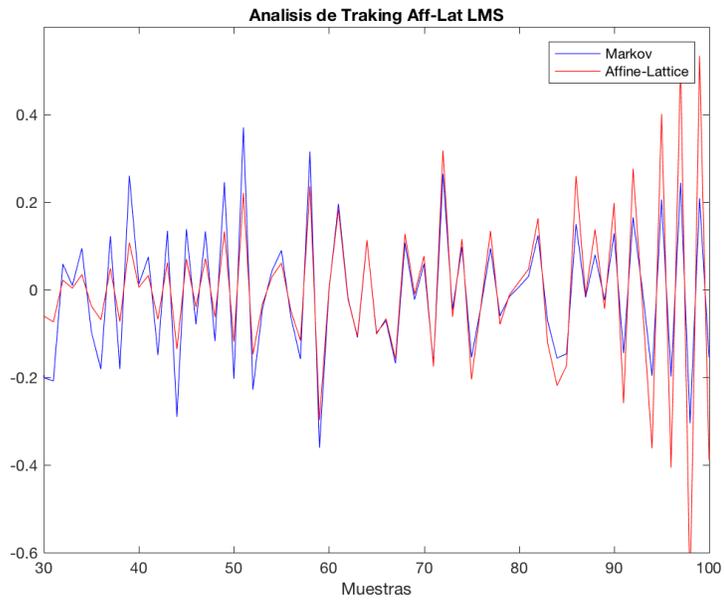


Figura 3.26. Algoritmo LMS en combinación *Affine* con estructura *Lattice* y proceso de Markov.

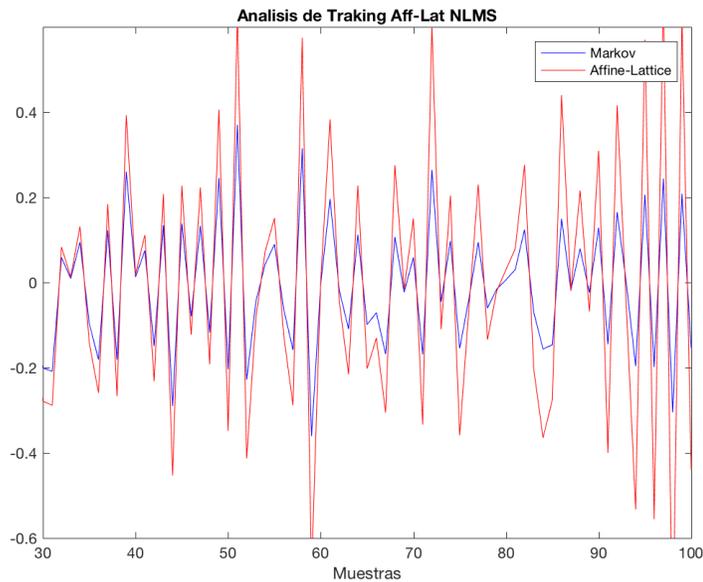


Figura 3.27. Algoritmo NLMS en combinación *Affine* con estructura *Lattice* y proceso de Markov.

En las figuras 3.25, 3.26 y 3.27 se observa que el mejor seguimiento se obtiene del filtro  $AR\gamma$  en combinación *Affine* en estructura *Lattice*.

### 3.7. Funcionamiento del Algoritmo Acelerador versión $\gamma$ ( $AR\gamma$ ) en estructura *Affine-Lattice*

Se inicializan las variables a operar estableciendo sus valores correspondientes, se comienza con el filtro LMS seguido de los filtros NLMS,  $AR\gamma$  con los pasos de adecuación para cada filtro a la combinación *Affine*. Se establece el número de muestras a considerar (N), el orden del filtro (M) a tener en cuenta, el valor inicial a calcular para el error cuadrático medio ( $n_2$ ) y el número de iteraciones o experiencias a realizar (S). Una vez iniciado el contador (n) se ingresan las señales de entrada ( $u_1$ ) y de referencia (d); se establecen los parámetros para la realización del proceso de Markov el cual permite realizar el análisis de *tracking*, seguidamente, se hace llamado a los algoritmos de los filtros LMS, NLMS y  $AR\gamma$  en su forma básica; a continuación se ejecutan los algoritmos LMS, NLMS y  $AR\gamma$  con su adecuación a la

estructura *Lattice* para posteriormente ejecutar los algoritmo LMS, NLMS y ARy en combinación *Affine-Lattice* y así finalmente, obtener un vector para el error cuadrático medio de cada experiencia; una vez se finaliza el contador de experiencias se obtiene el mínimo error cuadrático medio en la área de convergencia llegando de esta forma, al final del algoritmo. El proceso descrito anteriormente puede apreciarse en el diagrama de flujo que se ilustra en la figura 3.28.

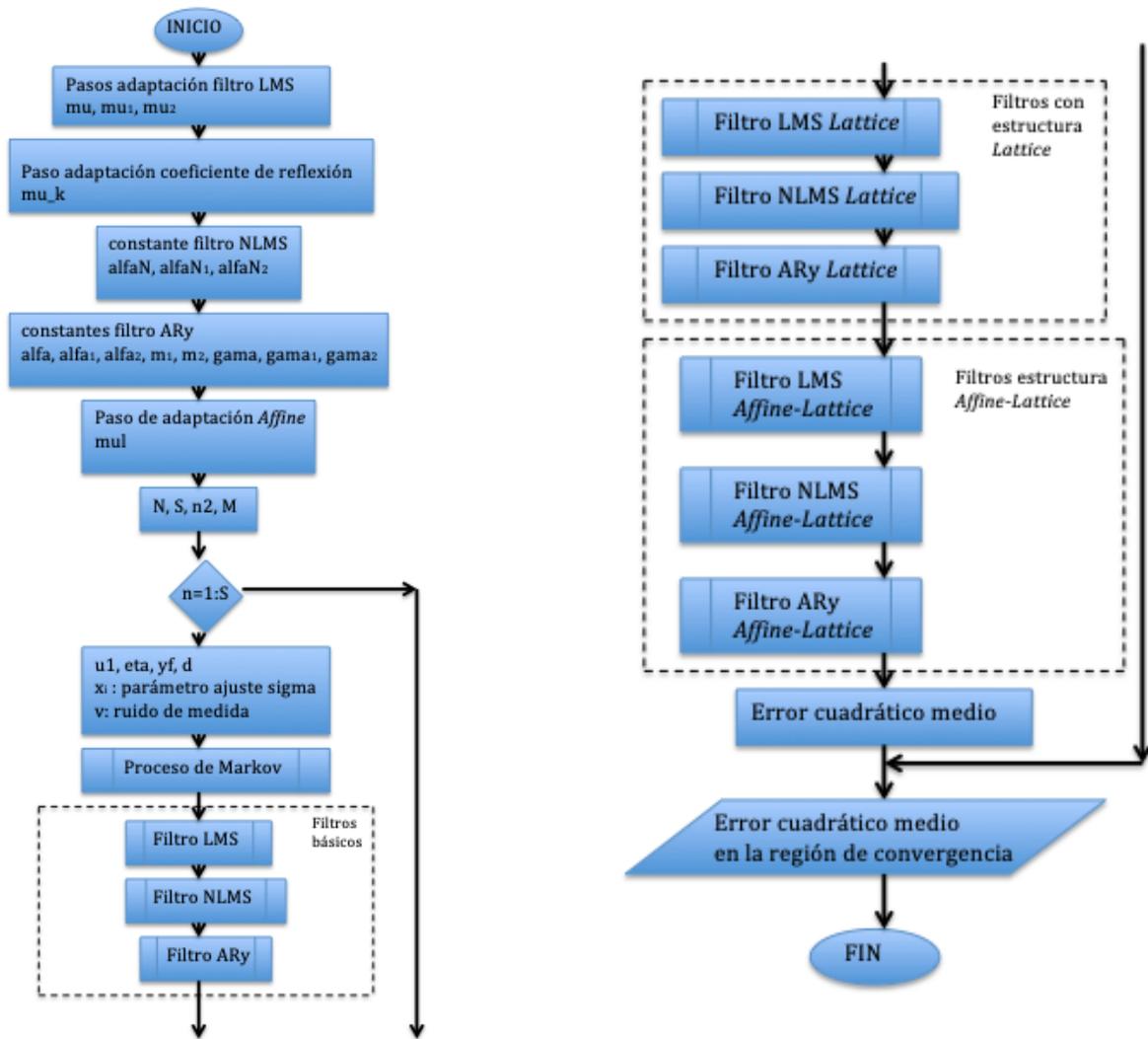


Figura 3.28. Diagrama de flujo filtros básicos, estructura *Lattice* y estructura *Affine-Lattice*.

La adecuación del algoritmo LMS a la estructura *Lattice* se inicia con la señal de entrada ( $x$ ), la señal deseada ( $d$ ), el paso de adaptación del filtro LMS ( $\mu$ ) el paso de adaptación para la estructura *Lattice* ( $\mu_k$ ), el orden a tener en cuenta del filtro ( $M$ ), el número de muestras a tener en cuenta ( $N$ ) y los coeficientes para el filtro inicializados en cero, inicializa el contador y se comienzan a formar los vectores denominados de adelanto ( $p_{flms}$ ) y atraso ( $p_{glms}$ ). Así mismo, se comienza con la actualización de los coeficientes de reflexión hasta que el contador llega al número de muestras determinadas inicialmente; por ultimo, se obtienen las señales de error ( $e$ ), los coeficientes propios del filtro LMS ( $w_{lms}$ ) los coeficientes de reflexión ( $w_K$ ) y la señal de salida del filtro ( $y_l$ ), ilustrada en la figura 3.29 como diagrama de flujo correspondiente a las etapas de ejecución del algoritmo.

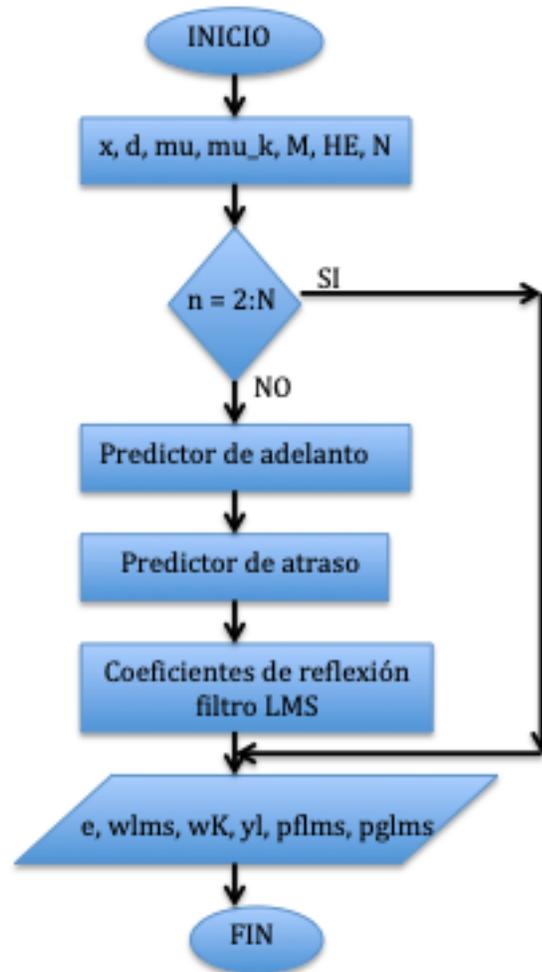


Figura 3.29. Diagrama de flujo filtros LMS *Lattice*.

Para el algoritmo NLMS y su adecuación a la estructura *Lattice* se inicia con la señal de entrada ( $x$ ), la señal deseada ( $d$ ), el paso de adaptación del filtro NLMS ( $\alpha N$ ) el paso de adaptación para la estructura *Lattice* ( $\mu_k$ ), el orden a tener en cuenta del filtro ( $M$ ), el número de muestras ( $N$ ) a tener en cuenta y los coeficientes para el filtro inicializados en cero ( $HE$ ), seguidamente se inicializa el contador ( $n$ ) y se comienzan a formar los vectores denominados de adelanto ( $pflms$ ) y atraso ( $pglms$ ), se continúa con la actualización de los coeficientes de reflexión ( $wKN$ ) hasta que el contador llega al número de muestras determinadas inicialmente; se finaliza

obteniendo las señales de error (eNLat), los coeficientes propios del filtro NLMS (wNlms) los coeficientes de reflexión (wKN) y la señal de salida del filtro (yNI), como se muestra en la figura 3.30 que corresponde al flujograma de las etapas de ejecución del algoritmo.

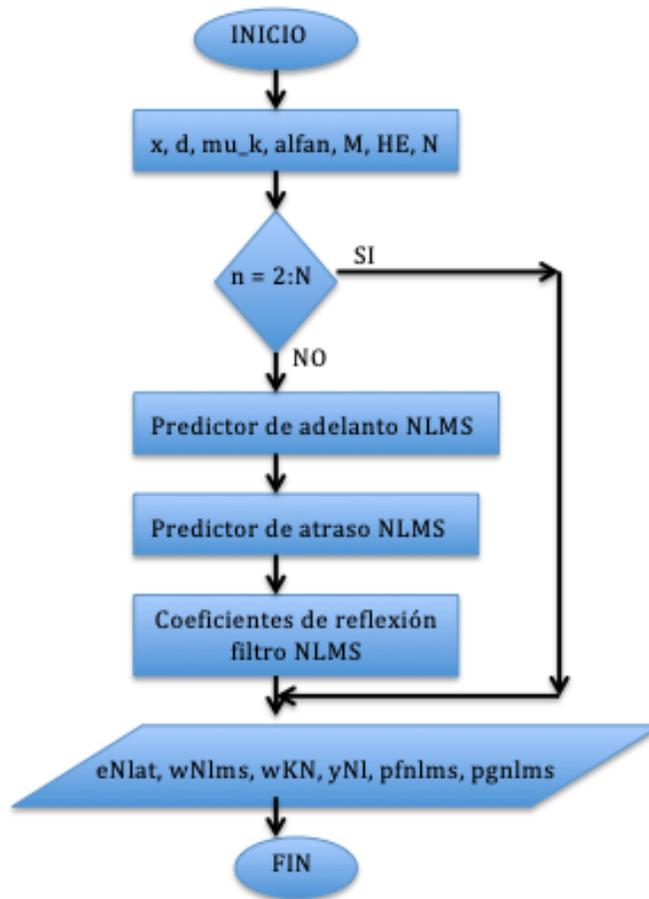


Figura 3.30. Diagrama de flujo filtros NLMS *Lattice*.

La adecuación del algoritmo  $AR_{\gamma}$  a la estructura *Lattice* se inicia con la señal de entrada ( $x$ ), la señal deseada ( $d$ ), las variables del filtro  $AR_{\gamma}$  ( $m_1$ , alfa y gama) el paso de adaptación para la estructura *Lattice* ( $\mu_k$ ), el orden del filtro a tener en cuenta ( $M$ ), el número de muestras ( $N$ ) y los coeficientes para el filtro inicializados en cero ( $HE$ ), inicializa el contador y se comienzan a formar los vectores denominados de adelanto ( $pfAR_{\gamma}$ ) y atraso ( $pgAR_{\gamma}$ ); así mismo, se continúa con la actualización de los coeficientes de reflexión ( $wK$ ) hasta que el contador llega al numero de muestras

determinadas inicialmente. Se finaliza obteniendo las señales de error (e), los coeficientes propios del filtro  $AR_\gamma$  ( $wAR_\gamma$ ) los coeficientes de reflexión ( $wK$ ) y la señal de salida del filtro ( $ys$ ), como se indica en la figura 3.31 como diagrama de ejecución del algoritmo.

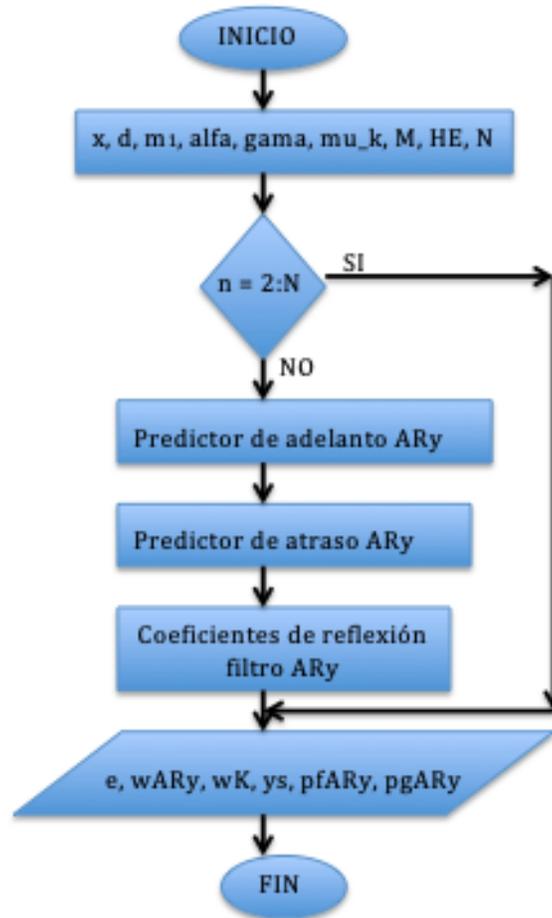


Figura 3.31. Diagrama de flujo filtros  $AR_\gamma$  *Lattice*.

La figura 3.32 registra el proceso de adecuación de la estructura *Affine-Lattice* al filtro LMS, el cual se inicia con la señal de entrada (x), la señal deseada (d), los pasos de adaptación para la combinación *Affine* ( $\mu_1$ ), la estructura *Lattice* ( $\mu_k$ ), como también el proceso de adecuación para los filtros tipo LMS y sus pasos de adaptación ( $\mu_1$  y  $\mu_2$ ), se determina el orden del filtro (M), el número de muestras (N) y los coeficientes para los filtros inicializados en cero (HE), inicializa el primer

contador y comienzan a formarse los vectores de los coeficientes para el filtro 1 LMS y el filtro 2 LMS llegando a la etapa de la combinación *Affine* la cual determina cual de los dos filtros tiene mejor comportamiento, seguidamente se inicia el segundo contador y se comienzan a formar los vectores denominados de adelanto (pflmsAL) y atraso (pglmsAL). Se procede con la actualización de los coeficientes de reflexión ( $w_k$ ) hasta que el contador llega al número de muestras determinadas inicialmente, se finaliza al obtener las señales de error ( $e$ ), los coeficientes para cada filtro ( $w_1$  y  $w_2$ ), la señal de salida del filtro después de la combinación *Affine* ( $ya$ ) y el parámetro de mezcla ( $lam$ ).

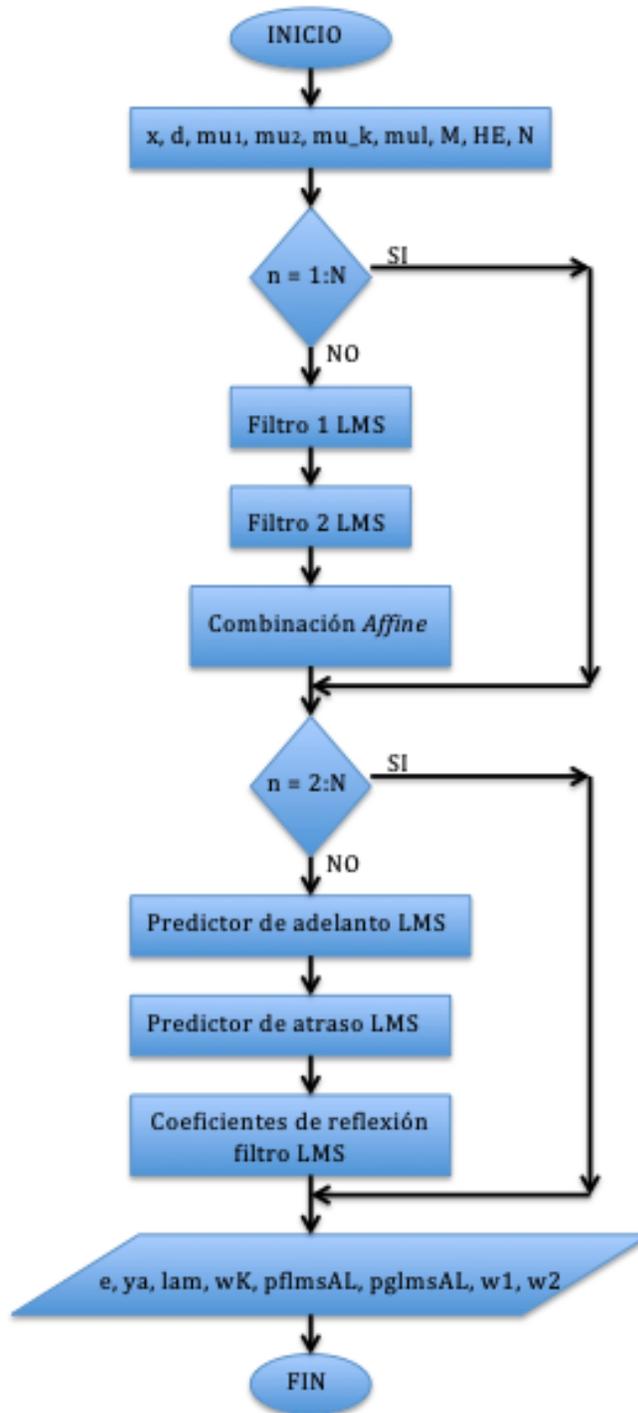


Figura 3.32. Diagrama de flujo filtro LMS *Affine-Lattice*.

La adecuación de la estructura *Affine-Lattice* al filtro NLMS y las etapas de ejecución del algoritmo se ilustran en la figura 3.33. El proceso se inicia con la señal de entrada ( $x$ ), la señal deseada ( $d$ ), los pasos de adaptación para la combinación *Affine* ( $\mu$ ), la estructura *Lattice* ( $\mu_k$ ), y los pasos de adaptación para los filtros NLMS ( $\alpha_{N_1}$  y  $\alpha_{N_2}$ ), se determina el orden del filtro ( $M$ ), el número de muestras ( $N$ ) y los coeficientes para los filtros inicializados en cero ( $H_E$ ), inicializa el primer contador y comienzan a formarse los vectores de los coeficientes para el filtro 1 NLMS y el filtro 2 NLMS llegando a la etapa de la combinación *Affine* la cual determina cual de los dos filtros tiene mejor comportamiento. Seguidamente se inicia el segundo contador y se comienzan a formar los vectores denominados de adelanto ( $p_{nlmsAL}$ ) y atraso ( $p_{nlmsAL}$ ), de igual forma se comienza con la actualización de los coeficientes de reflexión ( $w_k$ ) hasta que el contador llega al número de muestras determinadas inicialmente para finalmente obtener las señales de error ( $e$ ), los coeficientes para cada filtro ( $w_a$  y  $w_b$ ), la señal de salida del filtro después de la combinación *Affine* ( $y_a$ ) y el parámetro de mezcla ( $\lambda$ ).

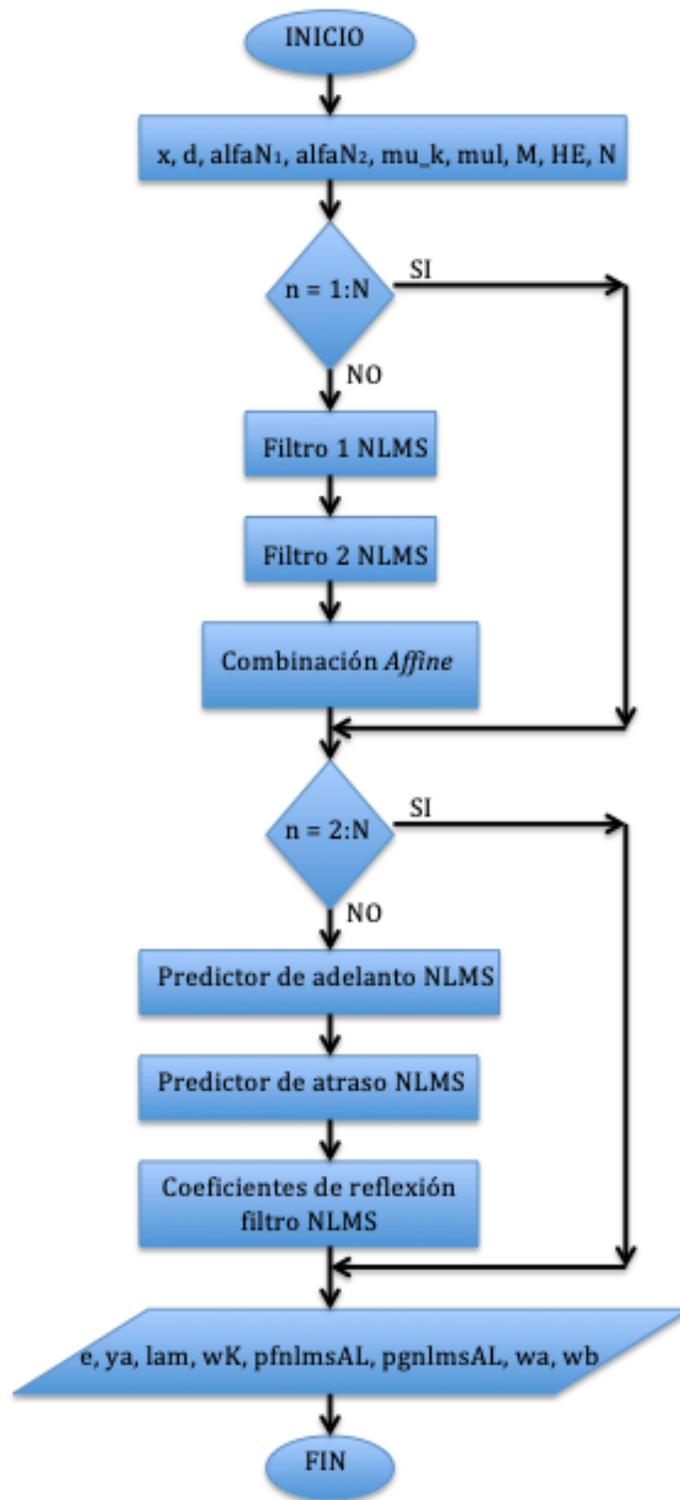


Figura 3.33. Diagrama de flujo filtro NLMS *Affine-Lattice*.

La implementación de la configuración *Affine-Lattice* para el filtro  $AR_{\gamma}$  y las etapas de ejecución del algoritmo se observan en la figura 3.34. El proceso inicia con la señal de entrada ( $x$ ), la señal deseada ( $d$ ), los pasos de adaptación para la combinación *Affine* ( $mul$ ), la estructura *Lattice* ( $mu_k$ ), y los pasos de adaptación para los filtros  $AR_{\gamma}$  ( $m_1, m_2, \alpha_1, \alpha_2, \gamma_1$  y  $\gamma_2$ ), se determina el orden del filtro ( $M$ ), el número de muestras ( $N$ ) y los coeficientes para los filtros inicializados en cero ( $HE$ ). Inicializa el primer contador y comienzan a formarse los vectores de los coeficientes para el filtro 1  $AR_{\gamma}$  y el filtro 2  $AR_{\gamma}$  llegando a la etapa de la combinación *Affine* en la que se determina cual de los dos filtros tiene mejor comportamiento. Seguidamente, se inicia el segundo contador y se comienzan a formar los vectores denominados de adelanto ( $pfARyAL$ ) y atraso ( $pgARyAL$ ), de igual forma, se comienza con la actualización de los coeficientes de reflexión ( $w_k$ ) hasta que el contador llega al número de muestras determinadas inicialmente. Por último se obtienen las señales de error ( $e$ ), los coeficientes para cada filtro ( $w_1$  y  $w_2$ ), la señal de salida del filtro después de la combinación *Affine* ( $ya$ ) y el parámetro de mezcla ( $lam$ ).

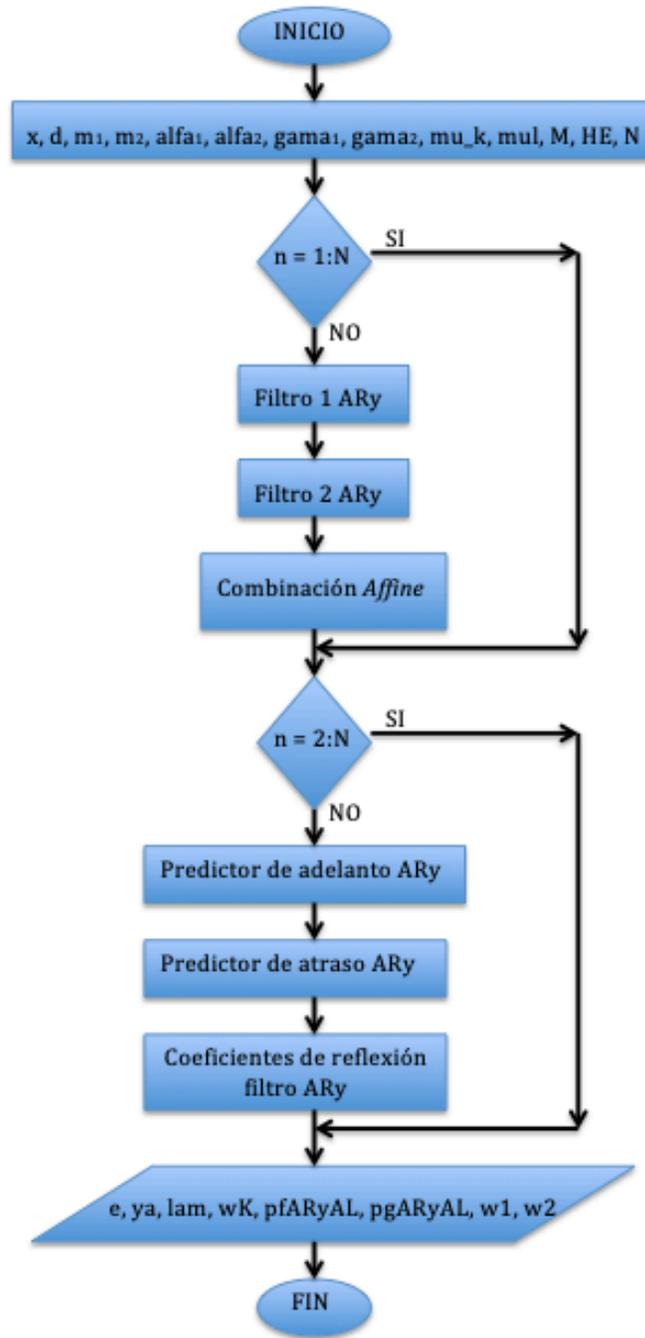


Figura 3.34. Diagrama de flujo filtro AR  $\gamma$  *Affine-Lattice*.

## Capítulo 4

### **Adecuación del algoritmo Acelerador versión $\gamma$ ( $AR_{\gamma}$ ) a la estructura *Affine-Lattice* y determinación de focos de fibrilación cardíaca.**

Con la finalidad de obtener un análisis cualitativo y cuantitativo de los algoritmos mencionados y para efectos de evaluar su desempeño sobre señales reales, se procede a la implementación del Algoritmo Acelerador versión  $\gamma$  ( $AR_{\gamma}$ ) a la estructura *Affine-Lattice* haciendo uso de señales reales, mediante una experiencia que consiste en la determinación de *zonas de frecuencia dominante* causantes de fibrilación auricular en situaciones de patologías cardíacas. La experiencia permite evidenciar el comportamiento predictivo (Modelo autorregresivo de la señal y la teoría de causalidad de Granger) de la estructura *Affine-Lattice*, la cual ofrece la propiedad de minimizar el error de predicción, lo que se traduce en la identificación de las zonas con *frecuencia dominante*, como causantes de fibrilación cardíaca.

Para efectos de materializar la experiencia anterior, se utilizan los registros de electrogramas intracardiacos obtenidos del Servicio de electrofisiología cardíaca del Hospital General Universitario Gregorio Marañón de Madrid (HGGM), registros que proporcionan el análisis de la fibrilación auricular obtenida mediante un catéter intracardiaco (PentaRay). El análisis se apoya en el diseño de un programa desarrollado en MATLAB cuyas características básicas consisten en la aplicación de la estructura *Affine-Lattice*; el empleo de un filtro adaptativo y el análisis soportado en la Teoría de causalidad de Granger. La adaptación de los algoritmos a partir de datos reales se lleva a cabo, sobre un procesador digital de señales (DSP).

## 4.1 Teoría de Causalidad de Granger. (Modelo autorregresivo de la señal.)

El fundamento de esta teoría es la idea de que, en el tiempo la causa siempre precede a los efectos: si una serie de tiempo es causa de otra, el conocimiento del primer proceso ayudaría a predecir los valores futuros del otro. (Granger, 1964) Se recurre a este modelo autorregresivo de la señal para comprobar las propiedades que ofrece el algoritmo acelerador versión  $\gamma$  ( $AR_{\gamma}$ ) en combinación *Affine* en la adaptabilidad de los coeficientes de reflexión y el componente de error de predicción específicos de la arquitectura *Affine*.

Por el lado de la economía, Hsiao (1982), especifica que generalmente hay una dificultad para precisar la discriminación respecto a la información de los datos, enfatizando la problemática que emerge de la colinealidad y los pocos grados de libertad entre otros. Es así que la información se extrae enfocándose solamente en las series de tiempo.

La forma más común de describir como se relaciona la causalidad con las series de tiempo es refiriéndose a los modelos de vectores autorregresivos (VAR), mencionados por Sims (1980) y el concepto de causalidad de Granger (1969).

Por medio de gráficas o grafos (Wright 1934) se pueden representar sistemas complejos y con ellos ver sus relaciones dinámicas, estos grafos constan de vértices que representan variables y flechas dirigidas en sentido a otras variables. De esta forma se expresa la influencia causal entre dichas variables.

Pearl (1995) usó los modelos gráficos para mostrar la causalidad entre variables y como se infiere no paramétricamente. Sin embargo, estos modelos dependen de suposiciones propias.

Eichler (2006 y 2007) hace su análisis para series de tiempo multivariadas introduciendo diagramas causales para visualizar estructuras autorregresivas de procesos estacionarios, lo que da una base ideal para el análisis de variables y su relación de causalidad propuesta por Granger. La figura 4.1 muestra la estructura causal en un proceso  $VAR_5(1)$ .



Granger basándose en modelos econométricos investigó su relación y la implicación que tienen al retroalimentarse y las funciones que se plantean en el análisis espectral. Su mayor aportación a la econometría fue la inclusión de una serie de definiciones y las pruebas respecto a la retroalimentación y causalidad entre series de tiempo, él logra probar haciendo uso del análisis espectral que el efecto del espectro cruzado de dos series de tiempo puede descomponerse en dos partes, relacionada cada una en una sencilla sección causal de una situación de retroalimentación, es decir, para dos variables, el mecanismo de retroalimentación puede descomponerse en dos relaciones causales y que su espectro cruzado se considera la suma de dos espectros cruzados, cada uno estrechamente vinculado con una de las causalidades.

#### 4.1.1. Modelos de retroalimentación.

Se tienen dos series de tiempo estacionarias,  $x_t$  y  $y_t$ , con media cero. Un modelo de retroalimentación consiste en un conjunto de ecuaciones lineales en las cuales ambas series de tiempo son explicadas en términos presentes y pasados de ellas mismas. Se incorpora un vector de ruido blanco para dar explicación a la parte del modelo faltante, se determinará por medio de  $\varepsilon_t$  que satisface  $E[\varepsilon_t' \varepsilon_s] = 0$  para  $t \neq s$  y  $E[\varepsilon_t' \varepsilon_s] = I$  para  $t = s$ , donde  $0$  es la matriz de ceros e  $I$  es matriz unitaria. Es así que se puede considerar el modelo de la siguiente forma:

$$\begin{aligned} x_t + b_0 y_t &= \sum_{j=1}^m a_j x_{t-j} + \sum_{j=1}^m b_j y_{t-j} + \varepsilon_t' \\ y_t + c_0 x_t &= \sum_{j=1}^m c_j x_{t-j} + \sum_{j=1}^m d_j y_{t-j} + \varepsilon_t'' \end{aligned} \quad (4.1)$$

##### 4.1.1.1. Causalidad.

Granger determina una serie de definiciones de causalidad y retroalimentación que permiten entender ésta de una forma mas clara ya que por otros métodos como son los espectrales resulta ser más dificultoso.

Se inicia determinando un proceso estocástico estacionario  $A$ , el conjunto de valores pasados es  $\bar{A} = \{A_{t-j}, j = 1, 2, \dots, \infty\}$  y el conjunto de valores pasados y presentes estará representado así  $\bar{\bar{A}}_t = \{A_{t-j}, j = 0, 1, 2, \dots, \infty\}$ , además se tendrá presente el conjunto  $\bar{A}(k) = \{A_{t-j}, j = k, k + 1, \dots, \infty\}$ . El predictor de mínimos cuadrados (insesgado y eficiente)  $P_t(A|B)$  de  $A_t$  usando el conjunto de valores  $B_t$ . Si se dice que  $P_t(x|\bar{x})$  es el mejor predictor de  $x_t$  usando solamente los valores pasados de  $x_t$ . Se denota la serie de error de predicción así  $\epsilon_t(A|B) = A_t - P_t(A|B)$  y la varianza sería  $\sigma^2 = (A|B)$ .

Sea  $v_t$  toda la información del universo acumulado desde el tiempo  $t - 1$  y sea  $v_t - y_t$  toda la información excepto la especificada en  $y_t$ .

### **Definición 1. Causalidad.**

Si  $\sigma^2(x|v) < \sigma^2(x|\bar{v} - y)$  se dice que  $y_t$  está causando  $x_t$  ( $y_t \xrightarrow{GC} x_t$ ). Se dice que  $y_t$  causa a  $x_t$  si se es más capaz de predecir  $x_t$  usando toda la información disponible que si la información exceptuando aquella de  $y_t$  hubiera sido usada. (Rodríguez, 2011).

### **Definición 2. Retroalimentación.**

Si  $\sigma^2(x|\bar{v}) < \sigma^2(x|\bar{v} - y)$  y  $\sigma^2(y|\bar{v}) < \sigma^2(y|\bar{v} - x)$  se dice que está ocurriendo una retroalimentación, la cual se denota como  $y_t \leftrightarrow x_t$ . Cuando  $y_t \xrightarrow{GC} x_t$  y  $x_t \xrightarrow{GC} y_t$  se dice que existe una retroalimentación entre  $x_t$  y  $y_t$ . (Rodríguez, 2011).

### **Definición 3. Causalidad instantánea.**

Si  $\sigma^2(x|\bar{v}, \bar{y}) < \sigma^2(x|\bar{v})$  se dice que una causalidad instantánea está ocurriendo, se denota  $y_t \xrightarrow{GC} x_t$ . Es decir, el valor actual de  $x_t$  es predice mejor si el valor presente de  $y_t$  es incluido en la predicción que si no lo fuera. (Rodríguez, 2011).

### **Definición 4. Causalidad rezagada.**

Sea  $m$  el rezago causal, siendo el mínimo valor de  $k$ , entonces si  $y_t \xrightarrow{GC} x_t$ , se define como causalidad rezagada con rezago  $m$  si  $\sigma^2(x|v - y(k)) < \sigma^2(x|v - y(k + 1))$ . Entonces el conocer los valores  $y_{t-j}, j = 0, 1, \dots, m - 1$  no serán de ayuda para mejorar la predicción de  $x_t$ . (Rodríguez, 2011).

Las definiciones anteriores han asumido que las series de tiempo son estacionarias, por lo que la causalidad en el sentido de Granger teniendo en cuenta una perspectiva no estacionaria no fue estudiada originalmente. Siendo que dicha situación no ha sido explorada de manera sistemática, salvo contadas excepciones como en He (2000), y mediante simulaciones Monte Carlo en Zapata (1988), el presente trabajo estudia la causalidad de Granger en presencia de procesos no estacionarios.

Hay que tener en cuenta que las definiciones anteriores exigen el uso de predictores lineales, por lo que el mejor predictor lineal de  $x_t$  usando solamente el pasado de  $x_t$  y el pasado de  $y_t$  sería así:

$$P_t(x|\bar{x}, \bar{y}) = \sum_{j=1}^{\infty} a_j x_{t-j} + \sum_{j=1}^{\infty} b_j y_{t-j} \quad (4.2)$$

donde  $a_j$  y  $b_j$  se eligen de forma que minimicen  $\sigma^2(x|\bar{x}, \bar{y})$ .

#### 4.1.1.2. Prueba de Causalidad

La prueba de causalidad mide la relación entre dos o más variables donde una, causa a la(s) otra(s).

Granger (1969) con su teoría de causalidad ha sido un hito en el análisis de series de tiempo en procesos estocásticos en diferentes campos de la ciencia, a pesar de haber sido citado numerosas veces, no es hasta que se desarrollan pruebas estadísticas que vuelven más sencilla la forma de estudiar las relaciones causales.

Sims, (1972), fue el primero en tratar de probar una relación causal simple. Esta prueba estuvo basada en representaciones de promedios móviles. Pero se encontró con algunos problemas en su procedimiento lo que no permitió que fuese exitosa dicha investigación. Sargent, (1976), propone una derivación de la definición de Granger, es conocido como procedimiento directo de Granger. Haugh y Pierce, (1977) proponen una prueba estadística la cual usa los residuales de los modelos univariados para  $x$  y  $y$ . Hsiao (1979) propone un procedimiento para identificar y estimar modelos bivariados los cuales se basan en representaciones autorregresivas y también pueden ser interpretadas como pruebas de causalidad.

A continuación, se describirá el procedimiento tenido en cuenta para probar la causalidad de Granger.

Se tienen las variables  $x$  e  $y$  luego para probar que  $x$  no causa  $y$  ( $x \overset{GC}{\nrightarrow} y$ ) se debe examinar si los valores rezagados de  $x$  en la regresión de  $y$  sobre los valores rezagados de  $x$  e  $y$  reduce significativamente el error de varianza.

Se asume el proceso autorregresivo de orden  $p$ , tanto en  $x$  como en  $y$ . Para poder usar los métodos de mínimos cuadrados ordinarios (MCO) atribuida a Carl Friederich Gauss, se estima la siguiente ecuación:

$$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + \dots + \alpha_p x_{t-p} + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \dots + \beta_p y_{t-p} + u_t \quad (4.3)$$

ecuación denominada *regresión no restringida*.

Entonces una prueba F debe aplicarse para probar la hipótesis nula,

$$H_0: \beta_1 = \beta_2 = \dots = \beta_p = 0 \quad (4.4)$$

Hamilton (1994) usa una forma de implementar esta prueba calculando la suma de residuos al cuadrado de la regresión no restringida (5.3) (*URSS, Unrestricted Residual Sum of Squares*).

$$URSS = \sum_{t=1}^T \hat{u}_t^2, \quad (4.5)$$

y compararla con la suma de residuales al cuadrado de una autorregresión univariada no restringida para  $x_t$  (*RRSS, Restricted Residual Sum of Squares*)

$$RRSS = \sum_{t=1}^T \hat{e}_t^2, \quad (4.6)$$

Donde

$$x_t = \gamma_0 + \gamma_1 x_{t-1} + \gamma_2 x_{t-2} + \dots + \gamma_p x_{t-p} + e_t \quad (4.7)$$

ecuación denominada *regresión restringida*. Es también estimada mediante MCO.

Ahora bien, si

$$S \equiv \frac{RRSS - URSS/P}{URSS/(T - 2p - 1)} \quad (4.8)$$

es mayor que el  $\alpha\%$  del valor crítico para una distribución  $F(p, T - 2p - 1)$ , entonces se debe rechazar la hipótesis nula que y no causa x. La prueba estadística (4.8) tiene una distribución F exacta para una regresión con regresores fijos y errores con distribución normal.

## 4.2. Determinación de focos de fibrilación cardíaca.

A continuación, se presenta la aplicación del algoritmo Acelerador  $\gamma$ (AR $_{\gamma}$ ) a la estructura *Affine-Lattice* en la determinación de focos de fibrilación cardíaca.

En la figura 4.2 se presenta el flujo de sangre (seguido por la flecha) a seguir cuando hay un funcionamiento normal del corazón (Dubin, 2000). La sangre entra al corazón por la vena Cava inferior hacia la aurícula derecha, de aquí pasa al ventrículo derecho que la impulsa hacia los pulmones. Una vez que la corriente sanguínea ha sido oxigenada, vuelve a la aurícula izquierda a través de las venas pulmonares. Desde la aurícula pasa al ventrículo izquierdo, éste se encarga de eyectar la sangre al resto del sistema circulatorio a través de la arteria aorta.

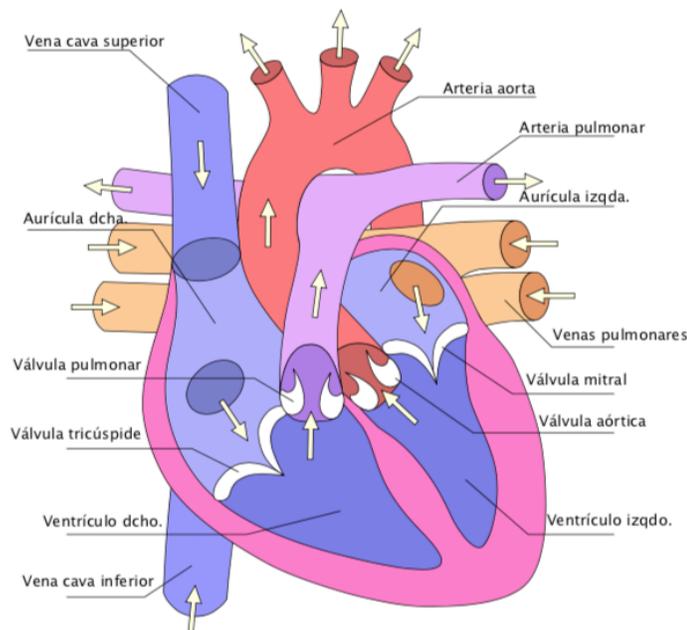


Figura 4.2. Partes Principales de un corazón humano. (Cumplido,2012)

Los movimientos periódicos que realiza el corazón para efectuar este recorrido son diástole y sístole. Durante la diástole auricular estas cavidades se relajan, dejando pasar la sangre a su interior. En la diástole ventricular, la relajación permite que pase parte de la sangre de las aurículas. La sístole es una contracción rápida, que impulsa la sangre a seguir el circuito descrito. El ventrículo izquierdo es la cavidad que realiza un mayor esfuerzo, ya que debe asegurar el riego al resto del cuerpo. Este movimiento es la parte más crítica del corazón.

Este movimiento sanguíneo y cardiaco es regulado mediante impulsos eléctricos, controlados desde los sistemas simpáticos y parasimpáticos (Houssay et al, 1972). En funcionamiento normal cardiaco se dice que el corazón late a ritmo sinusal. El nodo sinusal situado en la aurícula derecha es el encargado de generar la estimulación cardiaca que se propaga por el miocardio, a través del sistema de haces que se puede ver en la figura 4.3.

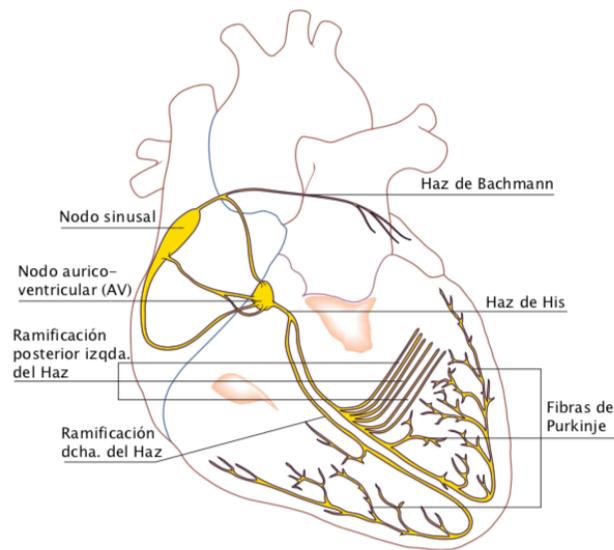


Figura 4.3. Vías de conducción eléctrica del corazón. (Cumplido,2012)

Esta estimulación eléctrica provoca las contracciones de las distintas cavidades y consecuentemente las sístoles auriculares y ventriculares. En el nodo aurículo-ventricular (A-V), situado en la confluencia de aurículas y ventrículos, este impulso se ve retenido debido a la distinta configuración de las células que la componen. De esta forma, la sangre dispone del tiempo suficiente para abandonar las aurículas e

ingresar en los ventrículos. Una vez superado el nodo A-V, el Haz de His distribuye el impulso eléctrico por un camino de fibras de conducción eléctrica rápida, conocido como fibras de Purkinje, que permite que los impulsos eléctricos provoquen una contracción rápida y coordinada del tejido ventricular.

Si se registra la actividad eléctrica del corazón de manera no invasiva, se obtienen trenes de ondas similares al electrocardiograma (ECG) de la figura 4.4. La onda P representa la despolarización auricular (contracción). El complejo QRS es la despolarización ventricular. El tiempo entre P y QRS es el necesario por la estimulación para atravesar el nodo A-V, de conducción mas lenta. Durante el QRS se produce la sístole ventricular. Finalmente, la onda T marca el final de la repolarización ventricular.

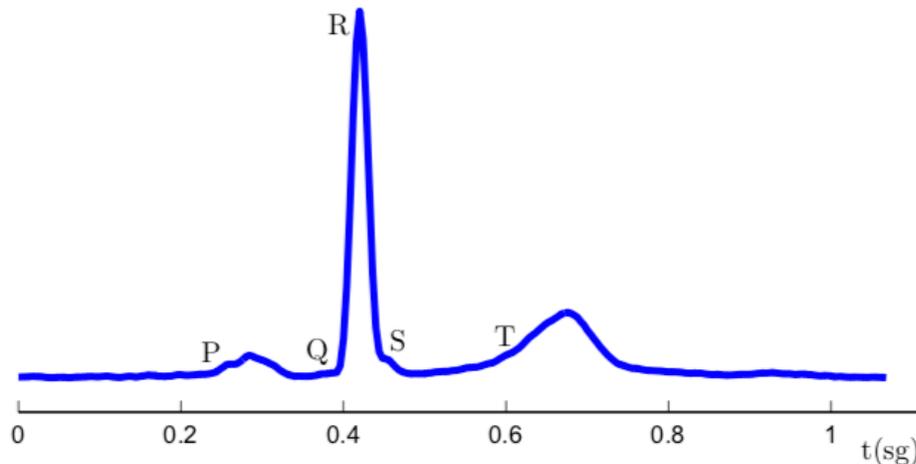


Figura 4.4. Onda sinusal del corazón con sus componentes principales, onda P, complejo QRS y onda T. (Cumplido,2012)

Lo anterior es la descripción del funcionamiento normal de las corrientes de estimulación cardiaca. Son varios los movimientos que comprometen el funcionamiento del corazón y de diverso origen.

En general los ritmos eléctricos anómalos pueden provocar dolencias cardiacas que se dividen en ritmos lentos, rápidos o de alto grado de descoordinación, llamados respectivamente: bradicardias, taquicardias o fibrilaciones, los que se definirán de manera breve a continuación.

### **4.2.1. Ritmos irregulares.**

Son ritmos generados por varios puntos de estimulación anómala con un patrón de latido mayor que la variabilidad normal del ritmo sinusal. En un corazón sano, la variabilidad cardiaca es necesaria y suele ser del 10% del ritmo, valores mayores entran en la categoría de irregulares los que se pueden dividir en:

#### **4.2.1.1. Marcapasos migratorio.**

En ocasiones el foco de estimulación del nodo sinusal se desplaza de ubicación, migra y produce variaciones en el ciclo cardiaco y por lo tanto en la longitud de la onda P (despolarización auricular) del ECG, aun cuando el ritmo general se mantiene en valores normales

#### **4.2.1.2. Taquicardia auricular multifocal.**

En esta patología son varios los focos activos de la aurícula, que además laten a un ritmo superior a 100 lat/min. Es particular en pacientes con enfermedad de obstrucción pulmonar crónica.

#### **4.2.1.3. Fibrilación auricular.**

En este caso el ritmo se dispara a valores superiores a 150 lat/min; los mecanismos de generación provocan controversia entre la comunidad científica. La apariencia de los electrogramas son ondas caóticas, con un ritmo muy irregular, que no son capaces de producir una sístole auricular eficaz.

## **4.3. Fibrilación auricular y frecuencias dominantes.**

Mediante técnicas estadísticas y de tratamiento de señal, se han podido obtener resultados al analizar señales bioeléctricas, como las producidas en el ser humano y generadas por el corazón (ECG-Electrocardiograma), el cerebro (EEG-Electroencefalograma) o los músculos (EMG- Electromiogramas). La captación cada vez más eficiente de estas señales unido a las técnicas avanzadas de tratamiento, han permitido obtener información relevante sobre diversas enfermedades y

patologías, así como mejorar las técnicas de diagnóstico de las mismas. (De-Prado-Cumplido y Artés-Rodríguez, 2010)

La **fibrilación auricular** (FA) es la arritmia cardíaca más frecuente que afecta la calidad de vida y puede generar otras patologías graves que incrementan el riesgo de mortalidad. Debido a condiciones particulares de excitabilidad presentes en un momento dado, puede resultar un tipo de actividad miocárdica completamente desordenada, durante la cual existen a un mismo tiempo fibras en contracción y fibras en reposo. Esta anomalía tiene tendencia a perpetuarse, especialmente si la masa muscular afectada es voluminosa y se conoce con el nombre de *fibrilación*, puede afectar a las aurículas, a los ventrículos o a ambas cámaras a la vez. (Houssay, et-al-1972)

Las **Frecuencias dominantes**, corresponden a zonas de la pared cardíaca que hipotéticamente, tienen una mayor velocidad en la generación de impulsos eléctricos y constituyen el foco de la arritmia.

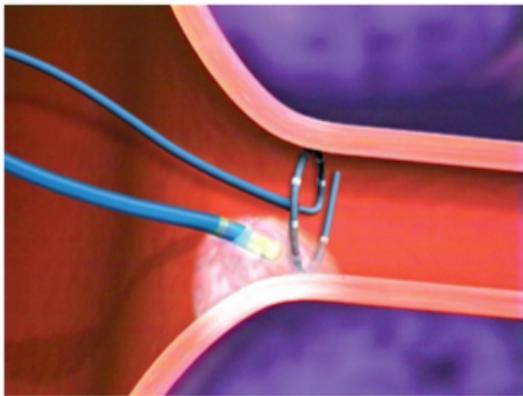
La arquitectura *Lattice* y las ecuaciones que la componen, su funcionamiento, los predictores de avance y retroceso, así como los coeficientes de reflexión, compatibles con las características de la configuración *Affine* potencializada con la utilización del algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) describen la estructura *Affine-Lattice* que ofrece la propiedad de minimizar el error de predicción permitiendo determinar zonas con *frecuencia dominante* causantes de la *fibrilación auricular cardíaca*. (Movimientos irregulares convulsivos de las aurículas debidos a que gran número de estímulos actúan sobre ellas. Las fibras musculares actúan independientemente, es decir, frente a condiciones particulares de excitabilidad existentes en un momento dado, puede resultar un tipo de actividad miocárdica completamente desordenada, durante la cual existen a un mismo tiempo fibras en contracción y fibras en reposo. Diccionario-Ciencias médicas-Salvat-Edit. S.A,1975. Houssay, et-al-1972)

El análisis se apoya, como se mencionó anteriormente, en el diseño de un programa desarrollado en MATLAB cuyas características básicas consisten en la aplicación de la estructura *Affine-Lattice* teniendo como soporte de adaptación el algoritmo

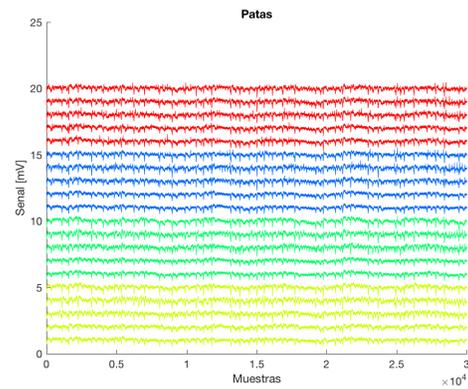
Acelerador versión  $\gamma$  (AR  $\gamma$ ) como filtro adaptativo y el análisis con base en la teoría de causalidad de Granger.

#### 4.4. Registro de electrogramas con catéter.

Para realizar este estudio se dispone de señales de electrogramas (EGM) obtenidos durante estudios médicos, en venas pulmonares, techo y orejuela de la aurícula izquierda. Las medidas se han registrado con catéteres circulares marca Lasso™ (Biosense Webster, Inc.) y Optima™ (St. Jude Medical, Inc), de 20 sensores con registro simultáneo. La disposición de los sensores se engloba en una circunferencia de diámetro variable entre 15 y 25 mm. En la figura 4.5 se muestra una recreación virtual del catéter circular situado en una vena pulmonar de la aurícula.



a)



b)

Figura 4.5. a) Catéter circular en vena pulmonar junto a sonda de ablación. b) Señales del Lasso™ en fibrilación auricular

A continuación, en la figuras 4,6 a 4,25 se observa cada una de las señales obtenidas por cada sensor del catéter en diferentes secciones del corazón.

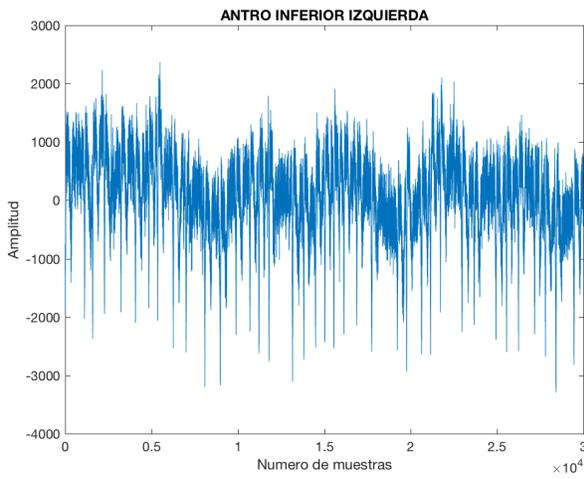


Figura 4.6. Antro Inferior Izquierda

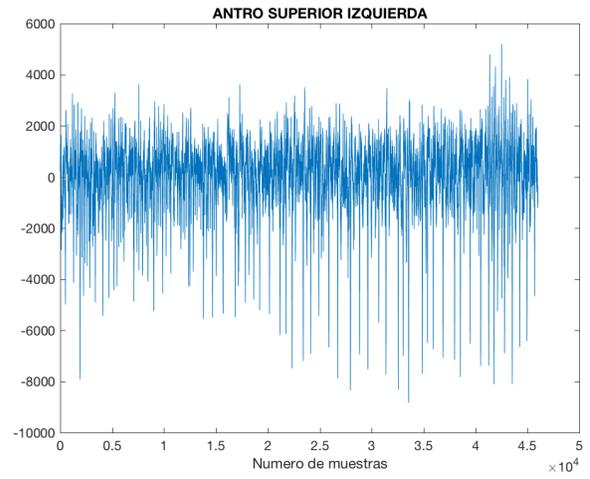


Figura 4.7. Antro Superior Izquierda

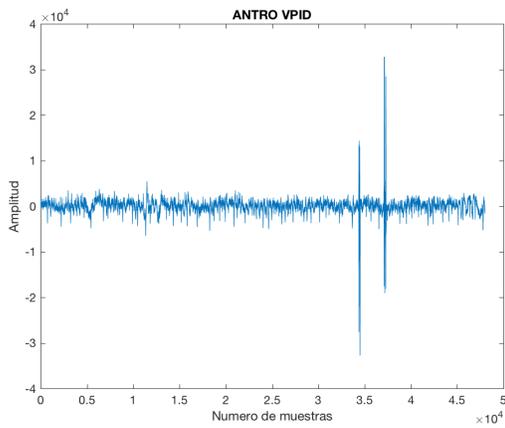


Figura 4.8. Antro Vena Pulmonar Inferior Derecha (VPID)

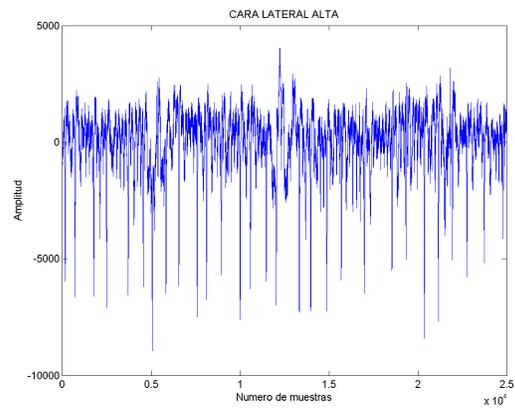


Figura 4.9. Cara Lateral Alta

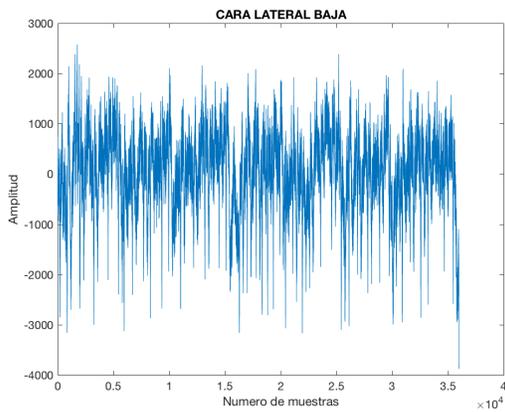


Figura 4.10. Cara Lateral Baja

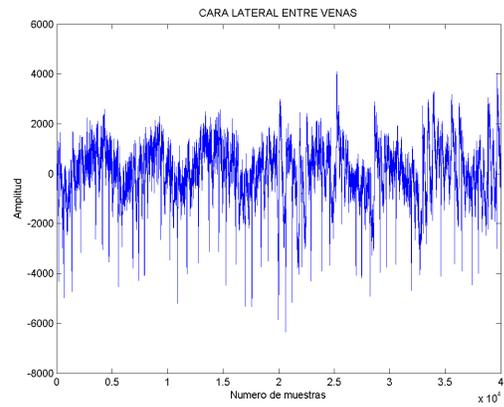


Figura 4.11. Cara Lateral entre Venas

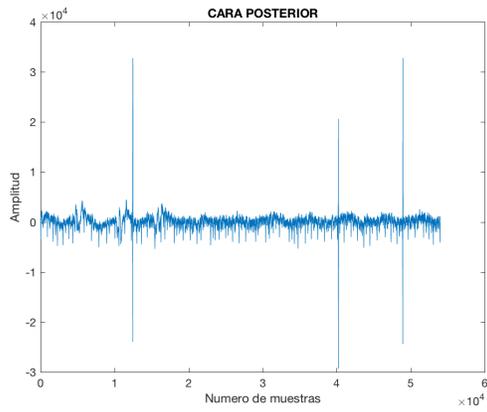


Figura 4.12. Cara Posterior

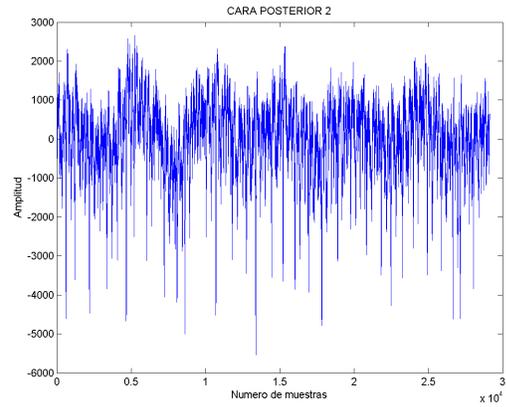


Figura 4.13. Cara Posterior 2

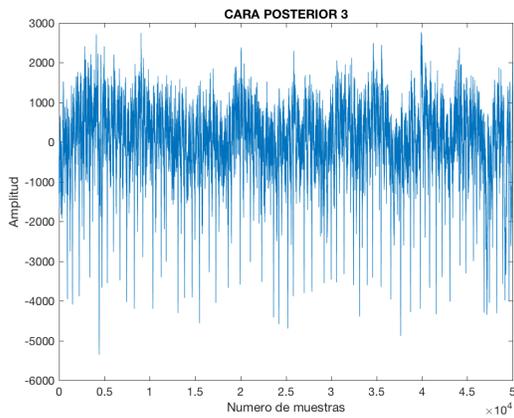


Figura 4.14. Cara Posterior 3

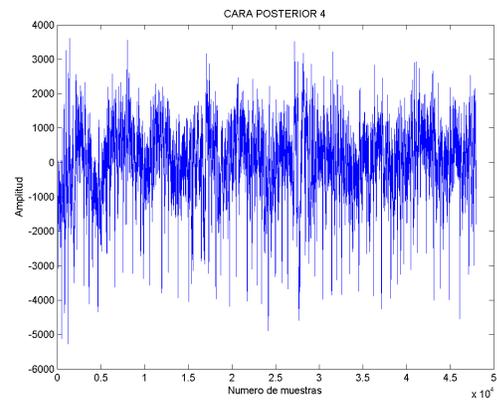


Figura 4.15. Cara Posterior 4

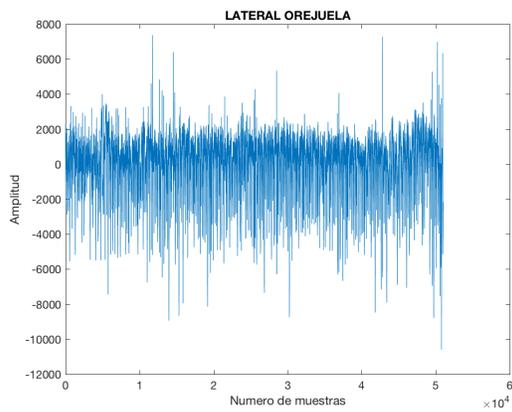


Figura 4.16. Lateral Orejuela

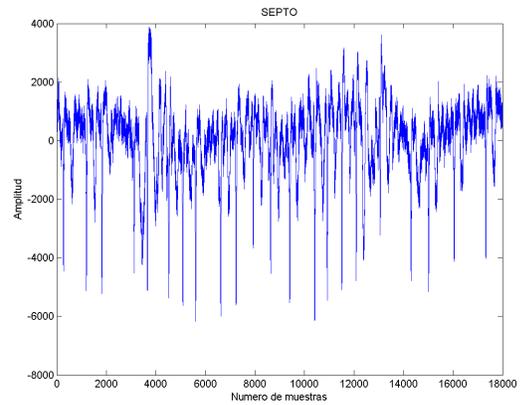


Figura 4.17. SEPTO o Tabique

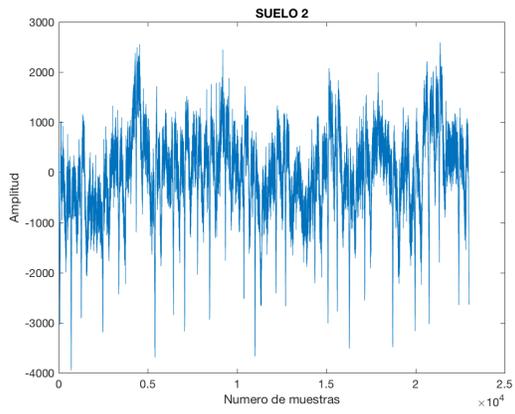


Figura 4.18. Suelo 2

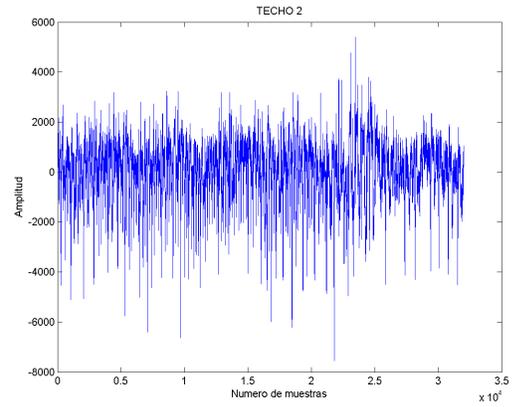


Figura 4.19. Techo 2

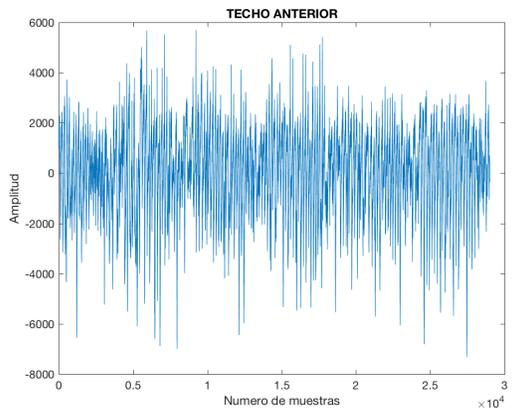


Figura 4.20. Techo anterior

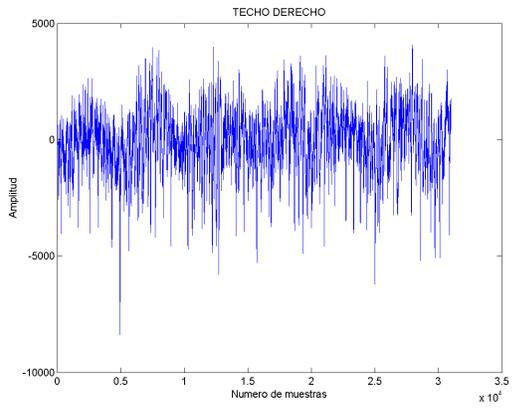


Figura 4.21. Techo Derecho

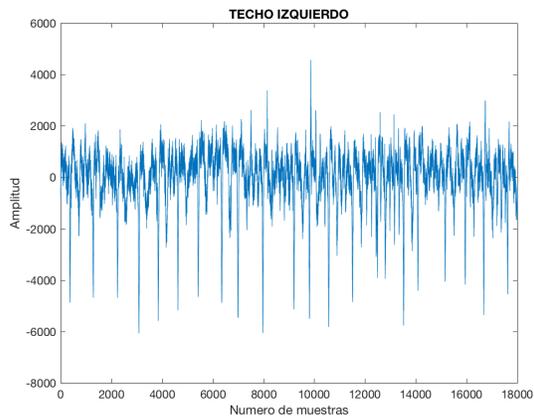


Figura 4.22. Techo Izquierdo

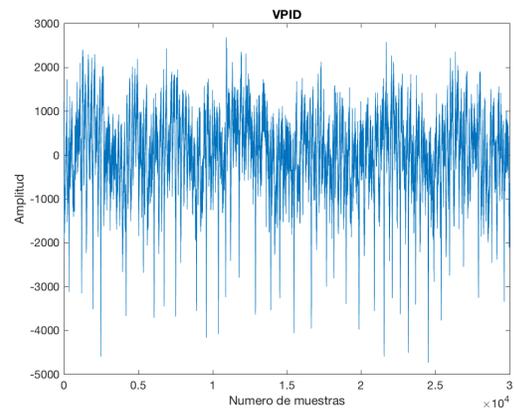


Figura 4.23. (VPID) Vena Pulmonar Inferior Derecha

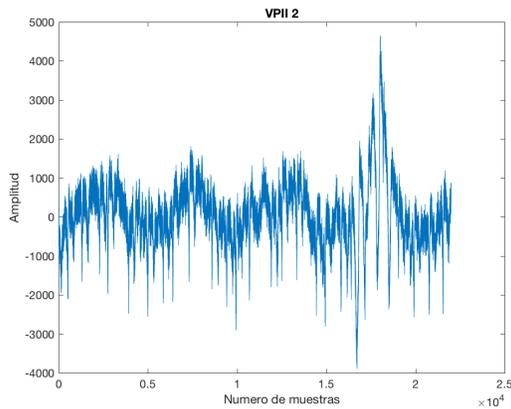


Figura 4.24. (VP11 2) Vena Pulmonar Inferior Izquierda

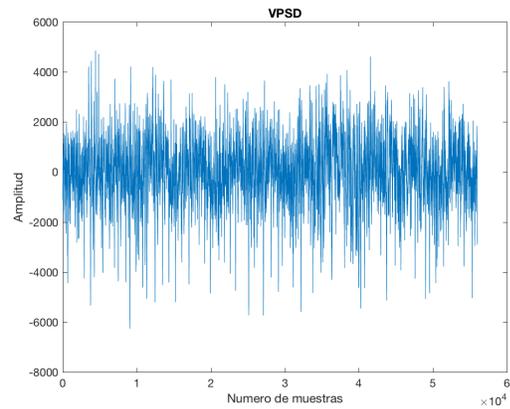


Figura 4.25. (VPSD) Vena Pulmonar Superior Derecha

## 4.5. Simulación

En capítulos anteriores se ha descrito la implementación de los algoritmos LMS, NLMS y ARy en combinación *Affine-Lattice*, en estructura *Lattice* y el test de causalidad de Granger; se obtienen los datos de error cuadrático medio, el valor estadístico F y valor crítico, lo anterior se realiza haciendo uso de MATLAB (programa interactivo orientado para llevar a cabo proyectos donde se encuentren implicados elevados cálculos matemáticos y la visualización gráfica de los mismos) en una configuración de identificación de sistema, este está constituido por un filtro FIR de respuesta impulsiva  $h$ , señal deseada  $d$  con media cero y varianza  $\sigma^2 = 10^4$ , los parámetros del sistema a identificar se suponen invariantes en el tiempo con valores entre -2 y 2 dado por  $y(k) = 2u(k) - 2u(k - 1)$ , orden del filtro  $M = 2$ , número de muestras  $N = 30000$ , número de experiencias  $K = 500$ . Las señales de entrada  $x(n)$  utilizadas son las obtenidas por el catéter; para el test de causalidad se usa  $\alpha = 0.8$  y *retardos = numero de orden del filtro*.

En la tabla 4.1 se observan los datos obtenidos al hacer aplicación del test de causalidad de Granger a los algoritmos LMS, NLMS y ARy aplicando las estructuras *Lattice*, *Affine-Lattice* y el error cuadrático medio para cada filtro.

**Tabla. 4.1 Error Cuadrático medio del Registro de las señales del corazón al aplicar el filtro ARy a la combinación *Affine-Lattice* y la Estructura *Lattice***

Señales Catéter <b>pentaray</b>	<i>Affine - Lattice</i>		<i>Lattice</i>
		ARy	ARy
3 – ANTRO INFERIOR IZQUIERDA	MSE	6.2081e-07	3.8642e-05
4 – ANTRO SUPERIOR IZQ	MSE	2.0267e-10	2.3609e-18
5 – ANTRO VPID	MSE	4.0792e-10	2.1533e-18
6 – CARA LATERAL ALTA	MSE	5.6427e-10	3.5778e-18
7 – CARA LATERAL BAJA	MSE	1.2290e-10	3.3456e-17
8 – CARA LATERAL ENTRE VENAS	MSE	4.8153e-10	5.8223e-19
9 – CARA POSTERIOR	MSE	4.3659e-10	2.1759e-18
10 – CARA POSTERIOR2	MSE	2.4887e-04	4.4813e-07
11 – CARA POSTERIOR3	MSE	2.3799e-09	7.3434e-18
12 – CARA POSTERIOR4	MSE	3.1473e-10	1.7805e-18
13 – LATERAL OREJUELA	MSE	2.4555e-10	1.7706e-17
14 – SEPTO	MSE	6.6534e-10	2.2911e-18
15 – SUELO2	MSE	3.5711e-10	1.9038e-18
16 – TECHO2	MSE	2.4275e-11	9.9256e-18
17 – TECHO ANTERIOR	MSE	2.3100e-10	3.4233e-18
18 – TECHO DERECHO	MSE	2.6410e-10	2.3084e-18
19 – TECHO IZQUIERDO	MSE	5.8471e-10	8.1776e-18
20 – VPID	MSE	9.0064e-10	4.9877e-18
21 – VPII2	MSE	8.7179e-10	4.0947e-18
22 – VPSD	MSE	1.3678e-04	4.9908e-08

Se seleccionan 5 señales con menor error cuadrático medio para hacer aplicación del test de causalidad.

**Tabla 4.2. Registro de las señales del corazón al aplicar los filtros ARy a la combinación *Affine-Lattice* y test de Causalidad de Granger.**

<b>Affine-Lattice ARy</b>		<b>4. Antro Sup Izq</b>	<b>7. Cara Lateral Baja</b>	<b>13. Lateral Orejuela</b>	<b>16. Techo 2</b>	<b>17. Techo Anterior</b>
<b>4. Antro Sup Izq</b>	F		17.1219	13.4799	8.7968	30.0191
	C_V		0.2231	0.2231	0.2231	0.0642
<b>7. Cara Lateral Baja</b>	F	20.4716		9.5641	159.7054	105.7101
	C_V	0.0642		0.2231	0.2231	0.2231
<b>13. Lateral Orejuela</b>	F	18.8808	20.2340		65.5940	6.0926
	C_V	0.2231	0.2231		0.2231	0.0642
<b>16. Techo 2</b>	F	190.8649	35.8623	59.3778		49.3212
	C_V	0.2231	0.0642	0.0642		0.2231
<b>17. Techo Anterior</b>	F	34.9447	178.5775	48.9832	144.5353	
	C_V	0.0642	0.2231	0.2231	0.2231	

Como se observa en la tabla 4.2 con los datos de F y C\_V y sabiendo que la teoría de causalidad establece que cuando F es mayor que C\_V la señal Y causa la señal X, teniendo en cuenta lo anterior y los datos de la tabla 4.2 se puede determinar lo siguiente:

Considerando un valor representativo de F (mayor valor) se podría decir que para las señales de la tabla 4.2 la señal 16. Techo 2 está causando las señales 7. Cara lateral baja la señal 13. Lateral orejuela pudiéndose considerar un foco de fibrilación.

El Diagrama de flujo que ilustra la figura 4.26 hace referencia a las etapas de ejecución realizadas en MATLAB, del Algoritmo Acelerador versión  $\gamma$  ( $AR_{\gamma}$ ) en estructura *Affine-Lattice* y prueba de causalidad de Granger. Se inicia estableciendo los valores para los parámetros de los filtros  $AR_{\gamma}$ , los pasos de adaptación para el combinación *Affine* ( $\mu_l$ ) y estructura *Lattice* ( $\mu_k$ ), con  $M$  se determina el orden del filtro y el número de etapas de la estructura *Lattice*, el número de muestras ( $N$ ) y el valor desde donde se inicia a calcular el error cuadrático medio ( $n_2$ ). Se continúa con el proceso, determinando las señales cardíacas que se tendrán en consideración, éstas deben ser de igual longitud. Una vez seleccionadas estas señales, se inicia con un contador que determinará el número de iteraciones a realizar permitiendo con ello una mejor definición de los resultados a considerar. En este ciclo se determina el sistema a identificar, se sigue con el algoritmo  $AR_{\gamma}$  en estructura *Affine-Lattice*, para así obtener un vector con los datos del error cuadrático, finalizando el contador se determina el mínimo error cuadrático medio en la región de convergencia, zona donde se ha estabilizado el filtro y se pasa a la etapa de causalidad, comenzando por dar valores a los parámetros de causalidad, nivel de probabilidad del valor crítico ( $\alpha$ ) y número de retardos ( $ret$ ), siendo este proceso el que permite determinar si la señal  $Y$  causa la señal  $X$ .

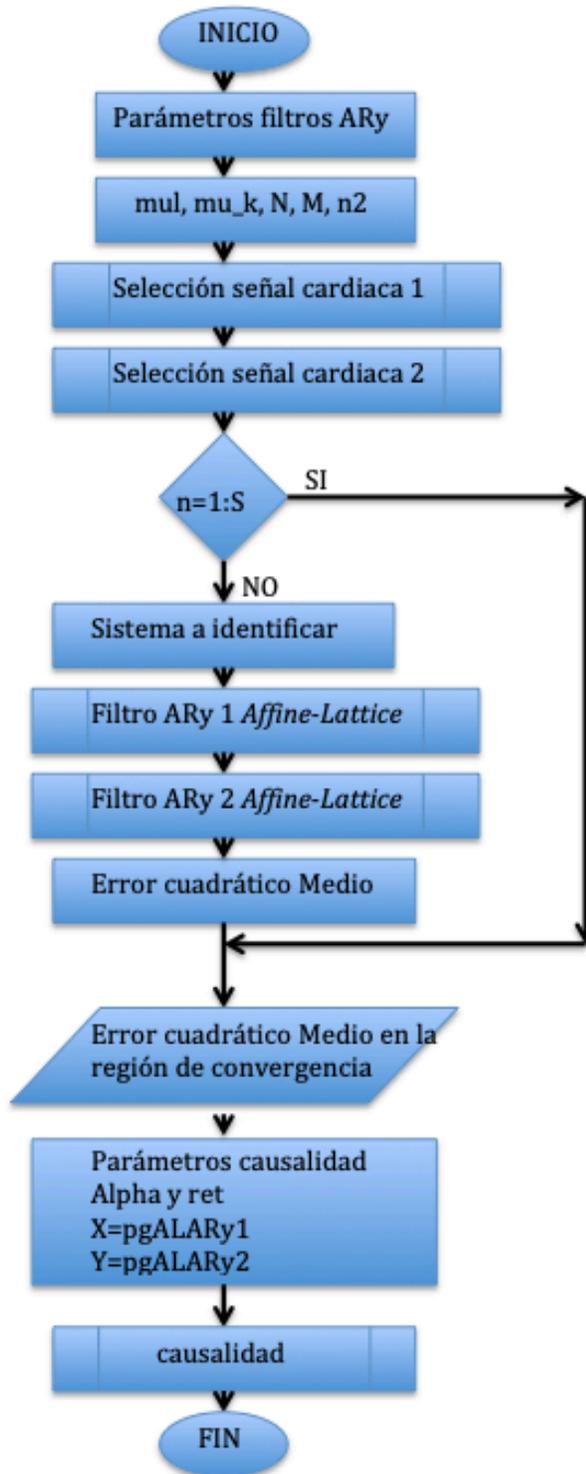


Figura 4.26. Diagrama de flujo filtro AR  $\gamma$  *Affine-Lattice* y Teoria de causalidad.

# Capítulo 5

## Conclusiones y Trabajo Futuro.

### 5.1. Conclusiones

El Algoritmo Acelerador versión  $\gamma$  (AR  $\gamma$ ) en configuración *Affine-Lattice* responde a un filtro adaptativo para procesamiento digital de señales con optimización de la adaptabilidad y rendimiento de sus propiedades de velocidad de convergencia, capacidad de seguimiento y mínimo error cuadrático medio.

El registro de la señal de error permite concluir que al aplicar la adecuación *Affine-Lattice* al filtro AR  $\gamma$  ofrece una mas rápida estabilidad si se compara con la de los filtros AR  $\gamma$ -*Lattice* y AR  $\gamma$  básico.

La aplicación de la estructura *Lattice* a los filtros LMS, NLMS y AR  $\gamma$  se traduce en un mejor comportamiento de los filtros adaptativos en lo relativo a mínimo error cuadrático medio, velocidad de convergencia y capacidad de seguimiento, destacándose el algoritmo AR  $\gamma$  como el mejor de los filtros. Así mismo se puede afirmar que la adecuación de los filtros anteriores a la estructura *Affine-Lattice* ofrece una mejora notable en el comportamiento de los mismos y continúa observándose el predominio del algoritmo AR  $\gamma$  como el mejor.

Las representaciones en forma de grafos permiten plantear relaciones causales entre señales o variables, orientando los enlaces según el mecanismo que direcciona el flujo de la información de una forma mas fácil de comprender que las relaciones

establecidas estadísticamente, facilitando una comprensión mas profunda del problema planteado.

Es importante recalcar la validación del comportamiento del algoritmo AR  $\gamma$  en configuración *Affine-Lattice* al utilizarse con señales reales en la determinación de los focos de frecuencia dominante causantes de fibrilación cardiaca.

Se ha realizado un aporte en el desarrollo de un nuevo algoritmo de búsqueda causal, integrando la estructura *Lattice* que de forma implícita lleva a la predicción lineal adaptativa mostrando una rápida convergencia a pesar del costo computacional que se presentan en otro tipo de filtros adaptativos como el LMS.

Se ha aplicado el nuevo algoritmo como herramienta en la búsqueda causal para series de tiempo de señales de fibrilación cardiaca, como una mejor forma de identificar la generación de problemas del corazón.

## 5.2. Trabajo Futuro

Realizar implementaciones del algoritmo AR  $\gamma$  en configuración *Affine-Lattice* en otros ambientes como cancelación de ruido y ecualización de canal, incluyendo el efecto el ruido con diferentes SNR, así mismo en otro tipo de señales como las geofísicas.

En el campo de la investigación biomédica resulta de gran utilidad para la medicina preventiva continuar con la adecuación del algoritmo AR  $\gamma$  en configuración *Affine-Lattice* en la determinación de zonas dominantes causantes de fibrilación cardíaca, como también en el área de la electroencefalografía en la determinación de zonas generadoras de focos convulsivos.

Verificar su implementación al hacer uso de sistemas de comunicación con múltiples antenas.

Otro ámbito a tener en cuenta es el que se relaciona a técnicas de remuestreo o submuestreo en el caso de interferencia de redes.

Se recomienda realizar la implementación del algoritmo RLS en combinación *Affine-Lattice*.

Al momento de realizar inferencia causal las áreas de investigación se amplían significativamente permitiendo el desarrollo de algoritmos a partir de conjuntos de muestras o de señales, los métodos varían según los datos a tomar ya sean discretos o de señales continuas.

# Anexos

## A.1. Pseudocódigo Algoritmo Acelerador Versión $\gamma$ (AR $\gamma$ ) en configuración *Affine*.

```
%programa para análisis de un identificador de sistema
%uso del algoritmo AR_gamma

%*****
%CONDICIONES INICIALES
%*****
clear all;
N = 3000;      %número de muestras
M = 7;        %orden del filtro adaptativo
nM = M;       %orden del filtro a identificar
n2 = 1000;    %posición inicial desde donde se calcula
              %el error cuadrático medio

n = 1:N;
K = 1         %Número de experiencias

%condiciones parámetro de ajuste algoritmo AR_gamma
m1=30;
m2=30;
alfa1=0.05;
alfa2=0.01;
gama1=2/(m1*alfa1);
gama2=2/(m2*alfa2);

%condiciones parámetros de ajuste algoritmo Affine
mul = 0.001;

%valor inicial de los coeficientes para los algoritmos
HE=zeros(M,1);

%condiciones para las señales aleatorias
sig=1;
sig2=0.05;

% otras inicializaciones
% estas variables almacenaran los valores cuadráticos de los errores
% medidos en cada experiencia
sel2=0;
sea2=0;
sea3=0;
seaf2=0;
```

```

%*****
%SISTEMA QUE SE DESEA IDENTIFICAR
%*****
B=[0.9 0.54 0.21 0.03 0.78 0.52 0.09];
A=1;
wo=B';

for k=1:K      %contador de número de experiencias

%*****
%SEÑAL DE ENTRADA (SEÑALES ALEATORIAS)
%*****

%señal de entrada
ul=sig*randn([N 1]);    %ruido blanco con desviación estándar=sig
x=ul;                  %señal de entrada al sistema

%ruido de medida (señal gaussiana de variancia sig^2
v2 = sig2*randn(size(x));

%*****
%SEÑAL ENTREGADA POR EL SISTEMA A IDENTIFICAR
%EN ESTE CASO (IDENTIFICADOR) ES LA SEÑAL DESEADA
%*****
yf=filter(B,A,x);
d=yf;          %señal deseada sin ruido de medida
d=yf+v2;      %señal deseada con ruido de medida

%*****
%REPUESTA ENTREGADA POR LOS ALGORITMOS ADAPTATIVOS:
%ERROR Y COEFICIENTES DEL FILTRO
%*****

[e_arg1,w_arg1,ys1]=ARy(x,d,m1,alfa1,gama1,M,HE); %algoritmo AR_gamma
[e_arg2,w_arg2,ys2]=ARy(x,d,m2,alfa2,gama2,M,HE); %algoritmo AR_gamma

%-----
[e_aff,YS_aff,w_lam]=aff(x,ys1,ys2,d,mul,M,HE); %Algoritmo Affine
%-----

%calculo del error cuadrático medio de la experiencia
ea2=e_arg1.^2;
ea3=e_arg2.^2;
eaf2=e_aff.^2;

%almacenamiento temporal del error cuadrático medio de las experiencias
%realizadas hasta ahora
sea2=sea2+ea2;
sea3=sea3+ea3;
seaf2=seaf2+eaf2;
end      %fin de la experiencia

```

```

%calculo del error cuadrático para obtener el error cuadrático medio final
mea2=sea2/K;
mea3=sea3/K;
meaf2=seaf2/K;

%valor medio del error cuadrático medio en la región de convergencia
eqm_arg1=mean(mea2(n2:N))
eqm_arg2=mean(mea3(n2:N))
eqm_aff=mean(meaf2(n2:N))

%grafica de los coeficientes ARy
figure(1)
f1=plot(n,w_arg1,'b',n,w_arg2,'g',n,w_lam,'m');
legend([f1(1), f1(M+1), f1(2*M+1)], 'ARy1', 'ARy2', 'Aff')
axis([0 3000 -2 2])
grid;

%grafica de salidas ARy y Affine
figure(2)
plot(n,ys1,'-*r',n,ys2,'-.b',n,YS_aff,'g');
axis([0 300 -6 6])
legend('ys1','ys2','ys_aff')
title ('Salidas ARy1-2 y Affine')
grid;

%grafica de errores ARy y Affine
figure(3)
plot(n,e_arg1,'r',n,e_arg2,'b',n,e_aff,'g');
axis([0 300 -6 6])
legend('e-arg1','e-arg2','e-aff')
title ('Errores ARy1-2 y Affine')
grid;

%Algoritmo Affine

% ecuaciones
%  $y(n) = y1(n)*lam(n)+y2(n)*(1-lam)$ 
%  $e(n) = d(n)-y(n)$ 
%  $lam(n+1) = lam(n)+mul*e(n)*(y1(n)-y2(n))$ 
% [e,YS,lam] =aff(xx,ys1,ys2,d,mul,M,HE)
% parámetros entrada:
% ys1 = ys2 = salidas de los filtros
% d = señal deseada
% mul = parámetro de adaptación

% parámetros de salida
% e = error
% YS = salida affine
% lam = parámetro de mezcla

function [es,YS,lami]=aff(xx,ys1,ys2,d,mu);
ai=0;
N = length(xx); %longitud de la señal
lami = 1/(1+exp(-ai));

```

```

for n=1:N,
    Yf(n)    = ys1(n)-ys2(n);
    YS(n)    = (ys1(n)*lami)+(ys2(n)*(1-lami));
    es(n)    = d(n)-YS(n);
    lami=lami+ mu*es(n)*Yf(n);
    lam(n)=lami;
end;
%algoritmo ARy

% [e,h] = ARy(x,d,m1,alfa,gama,M,HE)
% parámetros
% x      = señal de entrada
% d      = señal deseada
% m1     = parámetro de ajuste
% alfa   = parámetro de ajuste
% gama   = parámetro de ajuste
% M      = número de coeficientes
% HE     = condición inicial de los coeficientes

% g = escalar auxiliar
% q = vector auxiliar

% parámetros de salida
% e = error
% h = coeficientes del filtro ARy

function [es,HE1,ys] = ARy(xx,dd,m1,alfa,gama,M,HE)
N=length(xx); %longitud de la señal
XE=zeros(M,1); %vector de regresión
q1=zeros(M,1); %Vector de error q[n-1] inicial

for n=1:N
    XE=[xx(n); XE(1:M-1)];
    ys(n)=XE'*HE;
    es(n)=ys(n)-dd(n);
    g=(es(n)+gama*XE'*q1)/(1+(alfa*gama*m1*XE'*XE));
    qn=(gama/(gama+alfa))*(q1-(alfa*g*XE*m1));
    HE1(:,n)=HE;
    HE=HE+alfa*qn;
    q1=qn;
end

```

## A.2. Pseudocódigo Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en estructura *Affine-Lattice*

```

%programa para obtener los coeficientes de reflexión lattice al hacer uso
%de un sistema de identificación usando los algoritmos LMS, NLMS, ARy,
%LMS-Lattice, NLMS-Lattice, ARy-Lattice, LMS-AffineLattice, NLMS-
%AffineLattice y ARy-AffineLattice

%*****
%CONDICIONES INICIALES
%*****

clear all
close all

mu      = 0.005;           % paso adaptación filtro LMS
mu1     = 0.0009;
mu2     = 0.5;

mu_k    = 0.1;           % paso adaptación coeficientes reflexión

alfaN   = 0.008;         % constante filtro NLMS
alfaN1  = 0.08;
alfaN2  = 0.8;

alfa    = 0.5;           % Variables ARy
alfa1   = 0.5;
alfa2   = 0.8;
m1      = 0.8;
m2      = 0.5;
gama    = 2/(m1*alfa);
gama1   = 2/(m1*alfa1);
gama2   = 2/(m2*alfa2);

mul     = 0.5;           % condiciones Affine

N       = 5000;          % Numero de muestras
p       = 1:N;
S       = 500;           % Numero de experiencias
n2      = 2000;          % posición inicial desde donde se calcula

se11 = 0;
se12 = 0;
se13 = 0;
se14 = 0;
se15 = 0;
se16 = 0;
se17 = 0;
se18 = 0;
se19 = 0;
se20 = 0;

```

```

M    = 2;                % numero de etapas y orden del filtro
HE   = zeros(M,1);
K    = zeros(M,1);
A    = 1;                % numerador
B    = [-2 2];          % denominador

%Condiciones para las señales aleatorias
sig  = 0.99;            % desviación estándar x(n) (entrada)
sig2 = 0.05;           % desviación estándar ruido blanco n(n)
sigv = 0.01;           % desviación estándar Markov

for n = 1:S            %contador numero de experiencias

%-----
    u1 = sig*randn([N 1]);    % ruido blanco con desviación estándar=sig
    x  = u1;
%-----
    eta = sig2*randn(size(x)); % ruido de medida (señal gaussiana de
                                % variancia sig2^2)
    yf  = filter(B,A,x);      % sistema a identificar
    d   = yf;                 % señal deseada sin ruido
    %d  = yf+eta;             % señal deseada con ruido de medida

    %% -----
    xi  = 1;                  % parámetro de ajuste sigma
    v   = sigv*randn([size(x) M]); % ruido de medida markov
    [yf,cm1]=markov(x,xi,v,B); %Proceso de Marcov
    % -----

    %% -----FILTROS LMS, NLMS Y ARY-----
    [eLms,w,ylms] = lms(x,d,mu,M,HE);
    [enlms,wnlms,ynlms] = Nlms(x,d,alfaN,M,HE);
    [eARy,wARy,yARy] = ARy(x,d,m1,alfa,gama,M,HE);

    %-----FILTROS CON ESTRUCTURA LATTICE-----
    [eLat,wlmslat,wK,yslat,pf,pg] = lmslat(x,d,mu,mu_k,M,HE);
    [eNLat,wNlmslat,wKN,ysNlat,pfN,pgN] = Nlmslat(x,d,mu_k,alfaN,M,HE);
    [eALat,wALat,wKARy,ysALat,pfARy,pgARy]=ARylat(x,d,m1,alfa,gama,mu_k,M,HE);
    %-----

    %-----FILTROS CONBINACION AFFINE LATTICE-----
    [ea,yaALLMS,lam,wk,wlms1,wlms2,pfALLMS,pgALLMS] =
    ALLMS(x,d,mu1,mu2,mu_k,mul,M,HE);

    [eaARy,lamARy,pfALARy,pgALARy,KARy,wARy1]=
    ALARy(x,d,mul,m1,m2,mu_k,alfa1,alfa2,gama1,gama2,M,HE);

    [ealNLMS,lamNLMS,wKNLMS,wALNLMS1,wALNLMS2,pfALN,pgALN,yaALNLMS] =
    ALNLMS(x,d,alfaN1,alfaN2,mu_k,mul,M,HE);
    %-----

```

```

%calculo del error cuadrático medio de la experiencia
e11 = eLms.^2;
e12 = eLat.^2;
e13 = enlms.^2;
e14 = eNLat.^2;
e15 = eARy.^2;
e16 = eALat.^2;
e18 = ea.^2;
e19 = eaARy.^2;
e20 = ealNLMS.^2;

%almacenamiento temporal del error cuadrático medio de las experiencias
%realizadas hasta ahora
se11 = se11+e11;
se12 = se12+e12;
se13 = se13+e13;
se14 = se14+e14;
se15 = se15+e15;
se16 = se16+e16;
se18 = se18+e18;
se19 = se19+e19;
se20 = se20+e20;

end %Fin experiencias

%calculo del error cuadrático para obtener el error cuadrático medio final
me11 = se11/S;
me12 = se12/S;
me13 = se13/S;
me14 = se14/S;
me15 = se15/S;
me16 = se16/S;
me18 = se18/S;
me19 = se19/S;
me20 = se20/S;

%valor medio del error cuadrático medio en la región de convergencia
eqm_Lms = mean(me11(n2:N))
eqm_latlms = mean(me12(n2:N))
eqm_Nlms = mean(me13(n2:N))
eqm_NlmsLat = mean(me14(n2:N))
eqm_ARy = mean(me15(n2:N))
eqm_ARyLat = mean(me16(n2:N))
eqm_AffLatLMS = mean(me18(n2:N))
qm_AffLatARy = mean(me19(n2:N))
eqm_AffLatNLMS = mean(me20(n2:N))

```

```
%-----GRAFICAS-----
```

```
%Señal de entrada
```

```
figure(1)
plot(p,x)
title('Entrada')
ylabel('Amplitud')
xlabel('Muestras')
axis([0 1000 -0.4 0.4])
print('Señal Entrada','-dpng')
```

```
% Señal Filtro LMS-Lattice
```

```
figure (6)
subplot(3,1,1),plot(p,eLat,'b')
legend('e-LmsLat')
title('Error LMS-Lat')
ylabel('a')
axis([0 5000 -1 1])

subplot(3,1,2),plot(p,wlmslat,'r')
legend('W-LMSLat')
title('Coeficientes LMS-Lat')
ylabel('b')

subplot(3,1,3),plot(p,wK,'g')
legend('WK-LMS')
title('Coeficientes Refleccion LMS')
ylabel('c')

print('Filtro LMS-Lattice','-dpng')
```

```
%Señales de error LMS y Lattice
```

```
figure (2)
plot(p,eLms,'b',p,eLat,'r','LineWidth',2)
legend('error LMS','error LMS-Lat')
title('Error LMS y LMSLat')
ylabel('Muestras')
axis([0 1000 -6 6])
print('Error Filtro LMS y LMS-Lattice','-dpng')
```

```
% Señal Error Filtro LMS, LMS-Lattice y LMS-AffLat'
```

```
figure (2)
plot(p,eLms,'b',p,eLat,'r',p,ea,'g','LineWidth',2)
legend('error LMS','error LMS-Lat','error LMSAffLat')
title('Error LMS,LMS-Lattice y LMSAffLat')
ylabel('Muestras')
axis([0 1000 -10 10])
print('Error Filtro LMS, LMS-Lattice y LMS-AffLat','-dpng')
```

```

% Señal Filtro Affine-Lattice LMS

figure (38)
subplot(4,1,1),plot(p,ea,'g')
title('Error AffLat-LMS')
legend('e-ALLMS')
ylabel('a')

subplot(4,1,2),plot(p,pfALLMS,'b',p,pgALLMS,'r')
title('Predictores ALLMS')
legend('pf-ALLMS','pg-ALLMS')
ylabel('b')

subplot(4,1,3),plot(p,wk,'r')
title('Coeficientes Reflexion ALLMS')
legend('WK-ALLMS')
ylabel('c')

subplot(4,1,4),plot(p,wlms1,'c')
title('Coeficientes filtro ALLMS')
legend('w-ALLMS')
ylabel('d')

print('Filtro Affine-Lattice LMS','-dpng')

```

```

% Graficas filtro NLMS

```

```

%graficas filtro NLMS-LAT

figure (10)
subplot(3,1,1),plot(p,eNLat,'g')
title('Error NLmsLat')
legend('e-NLMS-Lat')
ylabel('a')
axis([0 600 -0.5 0.5])

subplot(3,1,2),plot(p,wNlmslat,'b')
title('Coeficientes NLmsLat')
legend('W-NLMS-Lat')
ylabel('b')

subplot(3,1,3),plot(p,wKN,'r')
title('Coeficientes Reflexion NLmsLat')
legend('WK-NLMS')
ylabel('c')
print('Filtro NLMS-Lattice','-dpng')

```

```

% Señal Error Filtro NLMS y NLMS-LAt

```

```

figure (9)
plot(p,enlms,'b',p,eNLat,'r')
title('Error NLMS y NLMS Lattice')
legend('eNLMS','eNLMS-Lat')
axis([0 300 -3 4])
print('Error Filtro NLMS y NLMS-LAt','-dpng')

```

```

% Señal Error Filtro NLMS, NLMS-LAt y NLMS-AffLat

figure (10)
plot(p,enlms,'b',p,enLat,'r',p,ealNLMS,'g')
title('Error NLMS, NLMS Lattice y NLMS Affine Lattice')
legend('eNLMS','eNLMS-Lat','eNLMS-AffLat')
axis([0 300 -3 4])
print('Error Filtro NLMS, NLMS-LAt y NLMS-AffLat','-dpng')

% Señal Filtro NLMS- AffineLattice

figure (10)
subplot(4,1,1),plot(p,ealNLMS,'g')
title('Error NLms Affine Lattice')
legend('e-NLMS-AffLat')
ylabel('a')
axis([0 400 -11 11])

subplot(4,1,2),plot(p,pfALN,'g',p,pgALN,'b')
title('Predictores NLms Affine Lattice')
legend('pf-NLMS-AffLat','pg-NLMS-AffLat')
ylabel('b')
axis([2500 2600 -6000 6000])

subplot(4,1,3),plot(p,wALNLMS1,'b')
title('Coeficientes NLms Affine Lattice')
legend('W-NLMS-AffLat')
ylabel('c')

subplot(4,1,4),plot(p,wKNLMS,'r')
title('Coeficientes Reflexion NLms Affine Lattice')
legend('WK-NLMS-AffLat')
ylabel('d')
print('Filtro NLMS- AffineLattice','-dpng')

%-----
% Graficas error cuadrático medio

% MSE Filtro LMS y LMS-Lattice

figure (11)
plot(p,mel1,'b',p,mel2,'r')
legend('eqm-lms','eqm-lmslat')
title('EMS Filtro LMS y LMS-Lat')
xlabel('Muestras')
ylabel('MSE')
axis([0 2000 0 0.15])
print('MSE Filtro LMS y LMS-Lattice','-dpng')

```

```

% MSE Filtro NLMS y NLMS-Lattice

figure (12)
plot(p,me13,'b',p,me14,'r')
legend('eqm-Nlms','eqm-Nlmslat')
title('MSE Filtro NLMS y NLMS-Lat')
xlabel('Muestras')
ylabel('MSE')
axis([1 600 0 15])
print('MSE Filtro NLMS y NLMS-Lattice','-dpng')

% MSE Filtro NLMS, NLMS-Lattice y NLMS Affine Lattice

figure (13)
plot(p,10*log(me13),'b',p,10*log(me14),'r',p,10*log(me20),'g')
legend('eqm-Nlms','eqm-Nlmslat','eqm-NlmsAffLat')
title('MSE Filtro NLMS, NLMS-Lat y NLMS AffLat')
xlabel('Muestras')
ylabel('MSE')
print('MSE Filtro NLMS, NLMS-Lattice y NLMS Affine Lattice','-dpng')

%-----
% Graficas del predictor de adelanto y atraso

% Predictores Filtro LMS-Lattice

figure (13)
plot(p,pf,'b',p,pg,'r')
title ('Predictores adelante y atraso LMS-Lattice')
legend('pred. adelanto','pred. atraso')
xlabel('Muestras')
ylabel('Amplitud')
axis([0 200 -0.4 0.4])
axis([0 500 -0.4 1])
print('Predictores Filtro LMS-Lattice','-dpng')

% Predictores Filtro NLMS-Lattice

figure (14)
plot(p,pfN,'b',p,pgN,'r')
title ('Predictores adelante y atraso NLMS-Lattice')
legend('pred. adelanto','pred. atraso')
xlabel('Muestras')
ylabel('Amplitud')
axis([0 250 -40 40])
print('Predictores Filtro NLMS-Lattice','-dpng')

```

```

%-----
% Graficas ARy

% Señal Error Filtro ARy y ARy-Lattice'

figure (15)
plot(p,eARY,'b',p,eALat,'r','LineWidth',2)
legend('error ARy','error ARy-Lat')
title('Error ARy y ARyLat')
ylabel('Muestras')
axis([0 200 -1 1])
print('Error Filtro ARy y ARy-Lattice','-dpng')

% Señal Error Filtro ARy, ARy-Lattice y ARy Affine-Lattice'

figure (16)
plot(p,eARY,'b',p,eALat,'r',p,eaARY,'g','LineWidth',2)
legend('error ARy','error ARy-Lat','error ALARY')
title('Error ARy, ARyLat y ARyAffLat')
ylabel('Muestras')
axis([0 200 -1 1])
print('Error Filtro ARy, ARy-Lattice y ARy Affine-Lattice','-dpng')

% Filtro ARy-Lattice

figure (16)
subplot(3,1,1),plot(p,eALat,'b')
legend('e-ARYLat')
title('Error ARyLat')
ylabel('a')
axis([0 5000 -1 1])

subplot(3,1,2),plot(p,wAlat,'r')
legend('W-ARYLat')
title('Coeficientes ARyLat')
ylabel('b')

subplot(3,1,3),plot(p,wKARY,'g')
legend('WK-ARYLat')
title('Coeficientes Reflexion ARyLat')
ylabel('c')
print('Filtro ARy-Lattice','-dpng')

% Predictores Filtro ARy-Lattice'

figure (17)
plot(p,pfARY,'b',p,pgARY,'r')
legend('pred. adelanto','pred. atraso')
title('Predictores adelante y atraso ARy-Lattice')
xlabel('iteraciones')
ylabel('Amplitud')
axis([0 200 -1 1])
print('Predictores Filtro ARy-Lattice','-dpng')

```

```

% MSE Filtro ARy y ARy-Lattice'

figure (18)
plot(p,me15,'b',p,me16,'r')
title ('MSE Filtros ARy y ARy-Lattice')
axis([0 1000 0 0.13])
xlabel('iteraciones')
ylabel('MSE')
legend('eqm-ARy','eqm-ARyLat')
print('MSE Filtro ARy y ARy-Lattice','-dpng')

% MSE Filtros LMS, LMS-lat

figure(21)
plot(p,me11,'b',p,me12,'r','LineWidth',2)
axis([2 2000 0 0.8])
legend('eqm-lms','eqm-lmslat')
xlabel('iteraciones')
ylabel('EMSE')
print('MSE Filtros LMS, LMS-lat','-dpng')

% MSE Filtros LMS, LMS-lat, LMS AffLat'

figure(22)
plot(p,10*log(me11),'b',p,10*log(me12),'r',p,10*log(me18),'LineWidth',2)
axis([15 10000 -1500 1500])
legend('eqm-lms','eqm-lmslat','eqm-lmsAffLat')
xlabel('iteraciones')
ylabel('EMSE')
print('MSE Filtros LMS, LMS-lat, LMS AffLat','-dpng')

% MSE Filtros ARy, ARy-lat, ARy-AffLat'

figure(29)
plot(p,10*log(me15),'b',p,10*log(me16),'r',p,10*log(me19),'g','LineWidth',2)
axis([0 2000 -800 100])
legend('eqm-ARy','eqm-ARyLat','eqm-ARyAffLat')
xlabel('iteraciones')
ylabel('EMSE')
print('MSE Filtros ARy, ARy-lat, ARy-AffLat','-dpng')

% MSE Filtros LMS, NLMS, LMS-Lat y NLMS-Lat

figure (33)
plot(p,me11,'b',p,me12,'r',p,me13,'k',p,me14,'c')
legend('eqm-lms','eqm-lmslat','eqm-Nlms','eqm-Nlmslat')
axis([0 1000 -0.05 0.15])
title('Error Cuadratico Medio LMS, NLMS, LMS-Lat, NLMS-Lat')
xlabel('iteraciones')
ylabel('EMSE')
print('MSE Filtros LMS, NLMS, LMS-Lat y NLMS-Lat ','-dpng')

```

```

% Predictores LMS-Lat y NLMS-Lat'

figure (34)
subplot(2,1,1),plot(p,pf,'b',p,pg,'c')
title ('Predictores adelante y atraso LMS Lattice')
axis([0 500 -0.5 0.5])

subplot(2,1,2),plot(p,pfN,'g',p,pgN,'r')
title ('Predictores adelante y atraso NLMS Lattice')
axis([2000 3000 -150 150])
print('Predictores LMS-Lat y NLMS-Lat','-dpng')

%-----Graficas Affine-Lattice-----

% Filtro Affine-Lattice-LMS'

figure (36)
subplot(4,1,1),plot(p,ea,'g')
title('Error AffLat-LMS')
legend('e-ALLMS')
ylabel('a')
axis([0 1000 -0.5 0.5])

j=1:length(lam);
subplot(4,1,2),plot(j,lam,'b')
title('Variable Affine')
legend('lamda-ALLMS')
ylabel('b')
axis([0 1000 -1 1])

subplot(4,1,3),plot(p,wk,'r')
title('Coeficientes Reflexion ALLMS')
legend('WK-ALLMS')
ylabel('c')

subplot(4,1,4),plot(p,wlms2,'c')
title('Coeficientes ALLMS')
legend('W-ALLMS')
ylabel('d')
print('Filtro Affine-Lattice-LMS','-dpng')

% Filtro Affine-Lattice ARy

figure (37)
subplot(4,1,1),plot(p,eaARy,'g')
title('Error AffLat-ARy')
legend('e-ALARy')
ylabel('a')
axis([0 1000 -0.5 0.5])

j=1:length(lamARy);
subplot(4,1,2),plot(j,lamARy,'b')
title('Variable Affine')
legend('lamda-ALARy')
ylabel('b')
axis([0 1000 -0.08 0.05])

```

```

    subplot(4,1,3),plot(p,KARy,'r')
    title('Coeficientes Reflexion ALARy')
    legend('WK-ALARy')
    ylabel('c')

    subplot(4,1,4),plot(p,wARy1,'c')
    title('Coeficientes filtro ALARy')
    legend('w-ALARy')
    ylabel('d')
    print('Filtro Affine-Lattice ARy','-dpng')

```

```
% Filtro Affine-Lattice NLMS'
```

```

figure (38)
subplot(4,1,1),plot(p,ealNLMS,'g')
title('Error AffLat-NLMS')
legend('e-ALNLMS')
ylabel('a')
axis([0 200 -0.5 0.5])

j=1:length(lamNLMS);
subplot(4,1,2),plot(j,lamNLMS,'b')
title('Variable Affine')
legend('lamda-ALNLMS')
ylabel('b')
axis([0 500 -0.01 0.01])

subplot(4,1,3),plot(p,wKNLMS,'r')
title('Coeficientes Reflexion ALNLMS')
legend('WK-ALNLMS')
ylabel('c')

subplot(4,1,4),plot(p,wALNLMS1,'c')
title('Coeficientes filtro ALNLMS')
legend('w-ALNLMS')
ylabel('d')
axis([0 5000 -3 3])
print('Filtro Affine-Lattice NLMS','-dpng')

```

```
% MSE Aff-Lat LNS,NLMS y ARy'
```

```

figure (39)
hc=plot(p,10*log(me18),'b',p,10*log(me19),'r',p,10*log(me20),'g');
set(hc, 'LineWidth',2)
legend('eqm-ALLMS','eqm-ALARy','eqm-ALNLMS')
title('EMS Affine-Lattice LMS, NLMS y ARy')
xlabel('Muestras')
ylabel('MSE')
axis([20 5000 -1500 1500])
print('MSE Aff-Lat LNS,NLMS y ARy','-dpng')

```

```

% ----- Graficas proceso de Markov -----

% Tracking ARy Lattice

figure(41)
plot(p,yf,'b',p,ysAlat,'r');
legend('Markov','Lattice')
title('Análisis de Tracking Lattice ARy')
xlabel('Muestras')
axis([0 150 -11 11])
print('Tracking ARy Lattice','-dpng')

% Tracking LMS Lattice'

figure(42)
plot(p,yf,'b',p,yslat,'r');
legend('Markov','Lattice')
title('Análisis de Tracking Lattice LMS')
xlabel('Muestras')
axis([0 100 -11 11])
print('Tracking LMS Lattice','-dpng')

% Tracking NLMS Lattice'

figure(43)
plot(p,yf,'b',p,ysNlat,'r');
legend('Markov','Lattice')
title('Análisis de Tracking Lattice NLMS')
xlabel('Muestras')
axis([0 100 -11 11])
print('Tracking NLMS Lattice','-dpng')

% Lattice LMS
% [e,wlms,wK,ys] = lmslat(x,d,mu,mu_k,M,HE)
%  $f_m(n)=f_{m-1}(n)+k_m(n)*g_{m-1}(n-1)$ ; Predictor de error adelanto
%  $g_m(n)=g_{m-1}(n-1)-k_m(n)*f_{m-1}(n)$ ; Predictor de error atraso
%  $k_m(n+1)=k_m(n)+mu*(f_m(n)*g_{m-1}(n-1)+g_m(n)*f_{m-1}(n))$ ; Coeficientes
% Lattice
%
% PARAMETROS DE ENTRADA:
% x = señal de entrada
% d = señal deseada
% mu = paso de adaptación para el filtro
% mu_k = paso de adaptación para coeficientes de reflexión menor a 1
% M = número de etapas lattice y numero de coeficientes filtro
% HE = condición inicial coeficientes LMS
% K = vector de coeficientes predictor lattice [Mx1]

% PARAMETROS DE SALIDA:
% e = error  $e(n)=d(n)-y(n)$ 
% w_lms = coeficientes del filtro LMS
% wK = coeficientes predictor lattice
% ys = salida del filtro

%-----

```

```

function [e,wlms,wK,y1,pf,pj]=lmslat(xx,dd,mu,mu_k,M,HE);

N      = length(xx);
XE     = zeros(M,1);
wK     = zeros(1,M);
epri   = zeros(1,M);
K      = zeros(1,M);
G      = zeros(M,1);
ys     = zeros;

for n=2:N

    wlms(:,n) = HE;
    wK(:,n)   = K(n);
    e(n)      = epri(1);
    y1(n)     = ys(1);
    XE       = [xx(n);XE(1:M-1)];

    f        = xx;
    g        = xx;

    %PREDICTOR ADELANTO
    f(n+1)   = f(n)+K(n)*g(n-1);
    pf(n)    = f(n+1);

    %PREDICTOR DE ATRASO
    g(n+1)   = g(n-1)+K(n)*f(n);
    pg(n)    = g(n+1);

    %-----LMS-----

    %COEFICIENTES DE REFLEXION
    K(n+1)   = K(n)+mu_k*(f(n)*g(n-1)+g(n)*f(n-1));

    %-----

    G        = [g(n);G(1:M-1)];
    ys       = G'*HE;           %salida filtro LMS
    epri     = dd(n)-ys;
    aa       = epri*mu;
    bb       = G*aa;
    HE       = HE+bb;         %coeficientes filtro Lms

end;

```

```

% Lattice NLMS
% [eNlat,wNlms,wKN,ysNlat] = Nlmslat(x,d,mu,M,HE,K)
% f_m(n)=f_m-1(n)+k_m(n)*g_m-1(n-1);          Predictor de error adelanto
% g_m(n)=g_m-1(n-1)-k_m(n)*f_m-1(n);          Predictor de error atraso
% k_m(n+1)=k_m(n)+mu*(f_m(n)*g_m-1(n-1)+g_m(n)*f_m-1(n)); Coeficientes
% Lattice
%
% PARAMETROS DE ENTRADA:
% x      = señal de entrada
% d      = señal deseada
% mu     = paso de adaptación para el filtro
% M      = número de etapas lattice y numero de coeficientes filtro
% HE     = condición inicial coeficientes NLMS
% K      = vector de coeficientes predictor lattice

% PARAMETROS DE SALIDA:
% e      = error e(n)=d(n)-y(n)
% wNlms  = coeficientes del filtro NLMS
% wK     = coeficientes predictor lattice
% ys     = salida del filtro

%-----

function [eNlat,wNlms,wKN,yNl,pf,pg]=Nlmslat(xx,dd,mu_k,alfa,M,HE);

N      = length(xx);
XE     = zeros(M,1);
wKN    = zeros(1,M);
epri   = zeros(1,M);
G      = zeros(M,1);
K      = zeros(1,M);
ys     = zeros;

for n=2:N
    wNlms(:,n) = HE;
    wKN(:,n)   = K(n);
    eNlat(n)   = epri(1);
    XE        = [xx(n);XE(1:M-1)];
    yNl(n)    = ys(1);

%-----Lattice-----
    f      = xx;
    g      = xx;

%PREDICTOR ADELANTO
    f(n+1)  = f(n)+K(n)*g(n-1);
    pf(n)   = f(n+1);

%PREDICTOR DE ATRASO
    g(n+1)  = g(n-1)+K(n)*f(n);
    pg(n)   = g(n+1);

```

```

%-----NLMS-----

%COEFICIENTES DE REFLEXIÓN
cc      = (f(n)*g(n-1))/((g(n-1))^2);
K(n+1)  = K(n)+mu_k*cc;

%-----
G        = [g(n);G(1:M-1)];
ys       = HE'*G;
epri     = dd(n)-ys;
HE       = HE+alfa/(eps+G'*G)*G*conj(epri);
end;

% Lattice ARy
% [e,wARy,wK,ys] = ARylat(x,d,m1,alfa,gama,mu_k,M,HE,K)
% f_m(n)  = f_m-1(n)+k_m(n)*g_m-1(n-1);      Predictor de error adelanto
% g_m(n)  = g_m-1(n-1)-k_m(n)*f_m-1(n);      Predictor de error atraso
% k_m(n+1) = k_m(n)+mu*(f_m(n)*g_m-1(n-1)+g_m(n)*f_m-1(n)); Coeficientes
% Lattice
%
% PARAMETROS DE ENTRADA:
% x        = señal de entrada
% d        = señal deseada
% m1       = parametro de ajuste
% alfa     = parametro de ajuste
% gama     = parametro de ajuste
% mu_k     = paso de adaptación para coeficientes de reflexión menor a 1
% M        = número de etapas lattice y numero de coeficientes filtro
% HE       = condición inicial coeficientes
% K        = vector de coeficientes predictor lattice
% v        = escalar auxiliar
% q        = vector auxiliar

% PARAMETROS DE SALIDA:
% e        = error e(n)=d(n)-y(n)
% W_ARy    = coeficientes del filtro ARy
% wK       = coeficientes predictor lattice
% ys       = salida del filtro

function [e,wARy,wK,ys,pf,pg]=ARylat(xx,dd,m1,alfa,gama,mu_k,M,HE);

N      = length(xx);          %longitud de la senal
XE     = zeros(M,1);
wK     = zeros(M,1);
epri   = zeros;
K      = zeros(M,1);
G      = zeros(M,1);
ys     = zeros;
cc     = zeros;
q1     = zeros(M,1);          % Vector de error q[n-1] inicial

```

```

for n=2:N

    XE      = [xx(n);XE(1:M-1)];
    wARy(:,n) = HE;
    wK(:,n)  = K;
    e(n)     = epri(1);
    yl(n)    = ys(1);
    f        = xx;
    g        = xx;

    %PREDICTOR ADELANTO
    f(n+1)   = f(n)+K(1)*g(n-1);
    pf(n)    = f(n+1);

    %PREDICTOR DE ATRASO
    g(n+1)   = g(n-1)+K(1)*f(n);
    pg(n)    = g(n+1);

    %-----ARy-----

    G        = [g(n);G(1:M-1)];
    ys(n)    = G'*HE;
    epri     = ys(n)-dd(n);
    aa       = epri+(gama*G'*q1);
    bb       = 1+(alfa*gama*m1*G'*G);
    v        = aa/bb;
    qn       = (gama/(gama+alfa))*(q1-(alfa*v*G*m1));
    HE       = HE+alfa*qn;
    q1       = qn;

    %-----
    %COEFICIENTES DE REFLEXI?N
    K        = K+(alfa*q1);

end;

```

```

% Affine - Lattice LMS
% [elf,y1f,wk,wlms,pf,pg,lam] = ALLMS(x,d,mu,mu_k,mul,M,HE)
% f_m(n)=f_m-1(n)+k_m(n)*g_m-1(n-1);          Predictor de error adelanto
% g_m(n)=g_m-1(n-1)-k_m(n)*f_m-1(n);          Predictor de error atraso
% k_m(n+1)=k_m(n)+mu*(f_m(n)*g_m-1(n-1)+g_m(n)*f_m-1(n)); Coeficientes
% Lattice
%
% PARAMETROS DE ENTRADA:
% x      = señal de entrada
% d      = señal deseada
% mu     = paso de adaptación para el filtro LMS
% mu_k   = paso de adaptación para coeficientes de reflexion menor a 1
% M      = numero de etapas lattice y numero de coeficientes filtro
% HE     = condicion inicial coeficientes LMS
% mul    = paso de adaptacion combinacion

% PARAMETROS DE SALIDA:
% e      = error Lattice - Affine LMS e(n)=d(n)-y(n)
% ys     = salida del filtro Lattice - Affine
% w_lms  = coeficientes del filtro LMS
% wK     = coeficientes predictor lattice
% pf     = senal adelanto lattice
% pg     = senal atraso Lattice
% lam    = parametro combinaci?n Affine

%-----

function [epri,ya,lam,wK,wa,wb,pf,pg] = ALLMS(xx,dd,mul,mu2,mu_k,mul,M,HE);

N      = length(xx);
XE     = zeros(M,1);
epri   = zeros(1,M);
epri1  = zeros(1,M);
epri2  = zeros(1,M);
K      = zeros(1,M);

lam    = 0;
HEa    = HE;
HEb    = HE;

for n=1:N

%-----
%AFFINE
%-----

wa(:,n) = HEa;
wb(:,n) = HEb;
XE      = [xx(n);XE(1:M-1)];

%-----FILTRO 1-----
y1(n)   = XE'*HEa;
epri1(n) = dd(n)-y1(n);
aa      = epri1(n)*mul;

```

```

HEa      = HEa+XE*aa;

%-----FILTRO 2-----
y2(n)    = XE'*HEb;
epri2(n) = dd(n)-y2(n);
bb       = epri2(n)*mu2;
HEb      = HEb+XE*bb;

%-----
ya(n)    = y1(n)*lam(n)+y2(n)*(1-lam(n));
epri(n)  = dd(n)-ya(n);
lam(n+1) = lam(n)+mul*epri(n)*[y1(n)-y2(n)];
end

%-----LMS Lattice -----

for n=2:N
f      = ya;
g      = ya;
wK(:,n) = K(n);

%PREDICTOR ADELANTO
f(n+1) = f(n)+K(n)*g(n-1);
pf(n)  = f(n+1);

%PREDICTOR DE ATRASO
g(n+1) = K(n)*f(n)+g(n-1);
pg(n)  = g(n+1);

%COEFICIENTES DE REFLEXION
K(n+1) = K(n)+mu_k*[f(n)*g(n-1)+g(n)*f(n-1)];

end;

```

```

% Algoritmo Affine - Lattice ARy
% parametros entrada:
% x = señal de entrada
% d = señal deseada
% m1 = parametro de ajuste filtro 1
% m2 = parametro de ajuste filtro 2
% alfa1 = parametro de ajuste filtro 1
% alfa2 = parametro de ajuste filtro 2
% gama1 = parametro de ajuste filtro 1
% gama2 = parametro de ajuste filtro 2
% g = escalar auxiliar
% q = vector auxiliar
% mul = paso de adaptación combinacion Affine
% M = numero de coeficientes
% HE = condicion inicial coeficientes

% parametros de salida
% e = error
% h = coeficientes del filtro
% y = salida de combinacion

function
[el,lam,pf,pg,wK,wARy1]=ALARY(xx,dd,mul,m1,m2,mu_k,alfa1,alfa2,gama1,gama2,
M,HE);

N = length(xx);
XE = zeros(M,1);
q1 = zeros(M,1); % Vector de error q[n-1] inicial
q2 = zeros(M,1); % Vector de error q[n-1] inicial
lam = 0;
HEa = HE;
HEb = HE;
K = zeros(M,1);
wK = zeros(M,1);
epri = zeros;
epril = zeros;
G = zeros(M,1);
ys = zeros;
cc = zeros;

for n=1:N

XE = [xx(n);XE(1:M-1)];

%FILTRO 1
wARy1(:,n) = HEa;
ys1(n) = XE'*HEa;
es1(n) = ys1(n)-dd(n);
g1 = (es1(n)+gama1*XE'*q1)/(1+(alfa1*gama1*m1*XE'*XE));
qn1 = (gama1/(gama1+alfa1))*(q1-(alfa1*g1*XE*m1));
HEa = HEa+alfa1*qn1;
q1 = qn1;

```

```

%FILTRO 2
wARy2(:,n) = HEb;
ys2(n)     = XE'*HEb;
es2(n)     = ys2(n)-dd(n);
g2         = (es2(n)+gama2*XE'*q2)/(1+(alfa2*gama2*m2*XE'*XE));
qn2        = (gama2/(gama2+alfa2))*(q2-(alfa2*g2*XE*m2));
HEb        = HEb+alfa2*qn2;
q2         = qn2;

%AFFINE
ya(n)      = ys1(n)*lam(n)+ys2(n)*(1-lam(n));
epri(n)    = dd(n)-ya(n);
lam(n+1)   = lam(n)+mul*epri(n)*[ys1(n)-ys2(n)];

end;

%-----Lattice -----

for n=2:N

    wK(:,n) = K;
    el(n)   = epril(1);
    yl(n)   = ys(1);
    f       = ya';
    g       = ya';

    %PREDICTOR ADELANTO
    f(n+1)  = f(n)+K(1)*g(n-1);
    pf(n)   = f(n+1);

    %PREDICTOR DE ATRASO
    g(n+1)  = g(n-1)+K(1)*f(n);
    pg(n)   = g(n+1);

    %-----ARy-----

    G       = [g(n);G(1:M-1)];
    ys(n)   = G'*HE;
    epril   = ys(n)-dd(n);
    aa      = epril+(gamal*G'*q1);
    bb      = 1+(alfal*gamal*m1*G'*G);
    v       = aa/bb;
    qn      = (gamal/(gamal+alfal))*(q1-(alfal*v*G*m1));
    HE      = HE+alfal*qn;
    q1      = qn;

    %-----
    %COEFICIENTES DE REFLEXI?N
    K       = K+mu_k*(alfal*((f(n)*g(n-1))+(g(n)*f(n-1))));

end;

```

```

% Affine - Lattice (NLMS)
% [elf,ylf,wk,wnlms,pf,pg,lam] = ALNLMS(x,d,alfa,mu_k,mul,M,HE)
% f_m(n)=f_m-1(n)+k_m(n)*g_m-1(n-1);          Predictor de error adelanto
% g_m(n)=g_m-1(n-1)-k_m(n)*f_m-1(n);          Predictor de error atraso
% k_m(n+1)=k_m(n)+mu*(f_m(n)*g_m-1(n-1)+g_m(n)*f_m-1(n)); Coeficientes
% Lattice
%
% PARAMETROS DE ENTRADA:
% x      = señal entrada
% d      = señal deseada
% alfa   = paso de adaptación para el filtro NLMS
% mu_k   = paso de adaptación para coeficientes de reflexion menor a 1
% M      = número de etapas lattice y numero de coeficientes filtro
% HE     = condición inicial coeficientes NLMS
% mul    = paso de adaptación combinacion Affine

% PARAMETROS DE SALIDA:
% e      = error Affine - Lattice NLMS
% ys     = salida del filtro Affine - Lattice
% wnlms  = coeficientes del filtro NLMS
% wK     = coeficientes predictor lattice
% pf     = señal adelanto lattice
% pg     = señal atraso Lattice
% lam    = parametro combinacion Affine

```

```

%-----

```

```

function [epri,lam,wK,wa,wb,pf,pg,ya] =
ALNLMS(xx,dd,alfa1,alfa2,mu_k,mul,M,HE);

```

```

N      = length(xx);
XE     = zeros(M,1);
epri   = zeros(1,M);
epri1  = zeros(1,M);
epri2  = zeros(1,M);
epriL  = zeros(1,M);
K      = zeros(M,1);
ys     = zeros;
G      = zeros(M,1);
lam    = 0;
HEa    = HE;
HEb    = HE;

```

```

for n=1:N

```

```

%-----AFFINE-----

```

```

wa(:,n) = HEa;
wb(:,n) = HEb;
XE      = [xx(n);XE(1:M-1)];

```

```

%-----FILTRO 1-----

```

```

y1(n)   = HEa'*XE;
epri1(n) = dd(n)-y1(n);
HEa     = HEa+alfa1/(eps+XE'*XE)*XE*conj(epri1(n));

```

```

%-----FILTRO 2-----
y2(n)      = HEb'*XE;
epri2(n)   = dd(n)-y2(n);
HEb        = HEb+alfa2/(eps+XE'*XE)*XE*conj(epri2(n));

%-----
ya(n)      = y1(n)*lam(n)+y2(n)*(1-lam(n));
epri(n)    = dd(n)-ya(n);
lam(n+1)   = lam(n)+mul*epri(n)*[y1(n)-y2(n)];
end

%-----NLMS Lattice -----

for n=2:N
f          = ya';
g          = ya';
wK(:,n)    = K(n);
wALnlms(:,n) = HE;

%PREDICTOR ADELANTO
f(n+1)     = f(n)+K(n)*g(n-1);
pf(n)      = f(n+1);

%PREDICTOR DE ATRASO
g(n+1)     = g(n-1)+K(n)*f(n);
pg(n)      = g(n+1);

%COEFICIENTES DE REFLEXI?N
cc         = (f(n)*g(n-1))/(eps+(g(n-1))^2);
K(n+1)     = K(n)+mu_k*cc;
end;

```

### A.3. Pseudocódigo Algoritmo Acelerador versión $\gamma$ (AR $\gamma$ ) en estructura *Affine-Lattice* y prueba de causalidad de Granger.

```
% Programa para verificar la adecuación del algoritmo acelerador versión  $\gamma$  y
% AR $\gamma$  y en estructura Affine-Lattice y la Teoría de Causalidad de Granger

%*****
%CONDICIONES INICIALES
%*****

clear all
close all

% condiciones filtro LMS
mu = 0.05; % paso adaptacion filtro LMS
mu1 = 0.005;
mu2 = 0.5;

%condiciones filtro NLMS
alfaN = 0.1; % constante filtro NLMS
alfaN1= 0.8;
alfaN2= 0.08;

% condiciones filtro AR $\gamma$ 
alfa = 0.5;
alfa1 = 0.5;
alfa2 = 0.8;
m1 = 1;
m2 = 0.5;
gama = 2/(m1*alfa);
gama1 = 2/(m1*alfa1);
gama2 = 2/(m2*alfa2);

% condicones Affine
mul = 0.005;

% condiciones Lattice
mu_k = 0.1; % paso adaptaci0n coeficientes reflexion
mu_k1 = 0.1;
mu_k2 = 0.05;

N = 5000; % Número de muestras
p = 1:N;
S = 200; %Número de experiencias
n2 = S; %posición inicial desde donde se calcula

se11 = 0;
se12 = 0;
se13 = 0;
se14 = 0;
se15 = 0;
se16 = 0;
```

```

sel7 = 0;
sel8 = 0;

A = 1; % numerador
B = [-2 2]; % denominador
M = 2; % número de etapas y orden del filtro
HE = zeros(M,1);
K = zeros(M,1);

%Condiciones para las señales aleatorias
sig = 0.1; % desviación estandar x(n) (entrada)
sig2 = 0.005; % desviación estandar ruido blanco n(n)

for n = 1:S %contador número de experiencias

yf1 = filter(B,A,x); % sistema a identificar
d1 = yf1; % señal deseada sin ruido

yf2 = filter(B,A,y); % sistema a identificar
d2 = yf2; % señal deseada sin ruido

%-----
[eaARY,lamARY,pfALARY,pgALARY,KARY,wALARY]=
ALARY(x,d1,mul,m1,m2,mu_k,alfa1,alfa2,gama1,gama2,M,HE);

[eaARY2,lamARY2,pfALARY2,pgALARY2,KARY2,wALARY2]=
ALARY(y,d2,mul,m1,m2,mu_k,alfa1,alfa2,gama1,gama2,M,HE);

%-----
e16 = eaARY.^2;
e18 = eaARY2.^2;
%-----
se16 = sel6+e16;
se18 = sel8+e18;

end %Fin experiencias
%-----
me16 = se16/S;
me18 = se18/S;
% -----
eqm_AffLatARY1 = mean(me16(n2:N))
eqm_AffLatARY2 = mean(me18(n2:N))

%% Causalidad

alpha=0.8;
ret=M;
x1=pgALARY;
y1=pgALARY2;

% Aplicando ARy - Affine - Lattice

[F,c_v] = Causa_Granger(x1,y1,alpha,ret)

```

## Bibliografía

Apolo, H. N. y Córdova, A. E. (2010). *Modelación matemática y simulación de un filtro digital híbrido FIR adaptativo lineal óptimo*. Tesis. Universidad Politécnica Salesiana, Cuenca, Ecuador.

Ayala, I, L. (1982). *On a New Adaptive Lattice Algorithm for Recursive Filters*. IEEE Transactions on Acoustics, speech, and signal processing, Vol. ASSP-30, No. 2, Abril.

Berberidis, K y Theodoridis, S. (1990). *A Normalized Lattice Algorithm for AR Apectral Analysis and System Identifacation Filters with Symmetric Impulse Response*. IEEE Transactions on Acoustics, speech, and signal processing. Vol. 38. No 3. Marzo.

Chakraborty, M y Mitra, A. (2004). *The Gradient Adaptive Lattice Algorithm in Block Floating Point Format*. IEEE ICASSP. Pp. 849-852.

Cifuentes, V., Iarussi, E y Clause, A. (2012) *Metodo de Lattice-Boltzmann para segmentación de texturas de imágenes*. Asociación Argentina de Mecanica Computacional. Vol 31. Pags. 3027-3036. Argentina. Noviembre

Cowan, C. F. Ñ. and Grant, P. M. (1986) *Adaptive Filters*. Prentice – Hall.

Dubin, Dale. (2000). *Rapid Interpretation of EKG's*. Cover Publishing, 6 ed. ISBN 978-0912912066.

Dedieu, H y Chételat, O. (1993) *Automatic derevation of adaptive algorithmn for a large class of filter structures*. IEEE. Suiza.

Friedlander, B. (1982). *Lattice Filters for Adaptive Prpcessing*. Proceedings of the IEEE. Vol. 70. No 8. Agosto.

Fujii, K., Tanaka, M., Sasaoka, N. y Itoh, Y. (2007). *Method estimating reflection coefficients of adaptive lattice filter and its application to system identifacation*. Acoust. Sci. & Tech. Vol. 28. Págs. 98-104. Febrero.

Galindo, H. (2008) *Estudio e implementación de algoritmos de filtrado adaptable en aritmética de punto fijo*. Tesis de Maestría, universidad Nacional Autónoma de México, México.

Gibson, C y Haykin, S. (1980). *Learning Characteristics of Adaptive Lattice Filtering Algorithms*. IEEE Transactions on Acoustics, speech, and signal processing. Vol. ASSP-28. No. 6. Diciembre.

Gunther, J. Song, W y Bose, T. (2006). *Stopping and Restarting Adaptive Updates to Recursive Least-Squares Lattice Adaptive Filtering Algorithms*. IEEE.

Haykin, S, (2002). *Adaptive Filter Theory*, 4a. Ed., Upper Saddle River, Prentice Hall, New Jersey

Houssay, B.A., Caldeyro-Barcia, R., Covian, M. R., Fasciolo, J. C., Foglia, V. G., Houssay, A. B., Hug. E., Leloir, L. F., Lewis, J. T y De Soldati, L. (1972). *Fisiología Humana*. 4 ed. Editorial el Ateneo. Argentina.

Honig, M. (1983). Convergence Models for Lattice Joint Process Estimator and Least Squares Algorithms. IEEE Transactions on acoustics, speech and signal processing. Vol ASSp-31. No 2. Abril.

Jojoa, P.E. (2003). Um Algoritmo Acelerador de Parâmetros, *Tesis de Doctorado*. Escola Politécnica da Universidade de Sao Paulo, Brasil.

Ketchum, J. y Proakis, J. (1982). *Adaptive Algorithms for Estimating and Suppressing Narrow-Band Interference in PN Spread-Spectrum Systems*. IEEE Trans. Communications, Vol. COM-30, págs. 913-923, Mayo.

Kimura, H. (1985). *Generalized Schwarz Form and Lattice-Ladder Realizations of Digital Filters*. IEEE Transactions on circuits and systems, Vol. CAS-32. No 1. Noviembre

Kuo, S y Luan, J. (1993). Cross-Coupled Filtered-X LMS Algorithm and Lattice Structure for Active Noise Control Systems. IEEE. Illinois. USA.

Lee, D y Sung, W. (2011). *Parallel Computation of Adaptive Lattice Filters*. IEEE ICASSP. Pp. 1601-1604

Lev-Ari, H. Chiang, K. F. y Kailath, T. (1987). *On the Stability of Adaptive Lattice Filters*. IEEE. Pp. 395-398.

Lu, J., Shen, C., Qiu, X, y Xu, B. (2003). *Lattice form Adaptive Infinite Impulse Response Filtering Algorithm for Active noise Control*. Acoustical Society of America. Pp. 327-335. Enero.

Martinez, M., Gomez, L., Gomez, J., Serrano, A y Vila, J. (2010). *Filtros Digitales*. Ingenieria Electrónica. Escuela Tecnica Superior de Ingeniería.

Markel, J. D., y Gray, A. H. (1976). *Linear Prediction of Speech*. Springer Verlag, Pp. Xii, 288.

Mathews, V y Douglas, S. (2003) *Adaptive Filters*. Capitulo 3.

Mathews, V y Xie, Z. (1990) *Fixed-Point Error Analysis of Stochastic Gradient Adaptive Lattice Filters*. IEEE Transactions on acoustics speech and signal processing. Vol 38. No 1. Enero.

Nichols, H. y Proakis, J. (1977). *MLD and MSE Algorithms for Adaptive Detection of Digital Signals in the Presence of Interchannel Interference*. IEEE Trans. Information Theory, Vol. IT-23, págs. 563-575, septiembre.

North, R., Zeidler, J., Albert, T., y Ku, W. (1992). *Comparison of Adaptive Lattice Filters to LMS Transversal Filters for Sinusoidal Cancellation*. IEEE. Pp. 33- 36.

Paleologu, C., Vladeanu, C. y El Assad, S. (2008). *Fast Convergence Least-Mean-Square Algorithms for MMSE Receivers in DS-CDMA Systems*. International Journal on Advances in Networks and Services, Vol 1.

Parikh, D., Ahmed, N., y Stearns, S. D. (1980). *An adaptive Lattice Algorithm for Recursive Filters*. IEEE Transactions on Acoustics, speech, and signal processing, vol. ASSP 28, No. 1. Febrero.

Picinbono, B. y Skwirzynski, J. K. (1978). *Adaptive Signal Processing for Detection and Communication*. En Communication Systems and Random Process Theory, Paises Bajos.

Proakis, J. G. y Manolakis, D. G. (2007) Tratamiento Digital de Señales, 4ed, Pearson Education S.A. Madrid.

Qi, H. (2008). *Analysis and Application of Gradient Adaptive Lattice Filtering Algorithm*. International Conference on Computer and Electrical Engineering. IEEE.

Qi, H. (2009). *Performance Analysis of Gradient Adaptive Lattice Joint Processing Algorithm*. International Asia Conference on Informatics in control, Automation and Robotics. Pp. 199-202.

Rao, H y Farhang-Boroujeny. (2009) Fast LMS/Newton Algorithms for Stereophonic Acoustic Echo Cancellation. IEEE Transactions on signal processing. Vol 57. No 8. Agosto.

Rivera, M. F. (2013). Efecto de la configuración *Affine en el Algoritmo Acelerador versión  $\gamma$ (AR  $\gamma$ )*. Tesis de Maestría, Universidad del Cauca, Popayán, Colombia.

Rodriguez, C. V. (2011) Ensayos sobre la Granger Causalidad. Tesis de Maestría, Universidad de Guanajuato, Mexico.

Rutter, M. J. Grant, P. M. Renshaw, D y Denyer, P. B. (1983). *Desing and Realisation of Adaptive Lattice Filters*. IEEE. ICASSP 83, pp. 21-24. Boston.

Salvat Editores S.A. (1975). Diccionario Terminologico de Ciencias Medicas. Barcelona. España.

Signell, S y Harnefors, L. (1977) *Implementation Aspects of Lattice Bilinear Digital Ladder Filters*. IEEE DSP 97-649. Suecia.

Sukhumaiwong, S., Benjangkprasert, C., Anantrasirichai, N y Janchitrapongvej, K. (2006). *Acho Cancellation Using the new Variable Step-siza Algorithm Adaptive Lattice Filters*. SICE-ICASE International Joint Conference, Octubre 18-21, Bexco (Busan-Korea).

Takemono, T., Sasaoka, N., Itoh, Y y Fujii, K. (2010). Speech Enhancement System Base don Lattice Filter and System Identification. IEEE. ISCIT 2010.

Watanabe, N., Nakachi, T y Hamada, N. (1996). *2-D Adaptive Line Enhancer using A New Lattice Structure*. Proceedings of IEEE Asia Pacific Conference on circuits and Systems '96. Noviembre 18-21. Seul. Korea.

Windrow, B., Mantey, P. y Griffiths, L. (1967). *Adaptative Antenna systems*. Proc. IEEE, Vol. 55, págs. 2143-2159, diciembre.

Zelaya, W. (2004). *Diseño de un filtro digital adaptativo como cancelador de ruido basado en el algoritmo LMS*. Trabajo de Grado. Universidad de el Salvador. Escuela de Ingenieria Electrica. Enero.

Zeidler, J., Chew, K., Axford, R y Ku, W. (1991) *Frequency Tracking Performance of Adaptive Lattice Filters*. IEEE. San Diego. California.

Zhi-Jin, Z. (2000). *An Adaptive Lattice Algorithm Base don Third-Order Cumulants*. IEEE Proceeding of ICSP2000.

Monzon, S. Trigano, T. Luengo, D y Artés-Rodriguez, A. (2012) *Sparse Apectral Analysis of Atrial Fibrilation Electrograms*. IEEE International Workshop on Machine Learning for Signal Processing. Pp. 1-6.

Prado Cumplido, M. (2012). *Busqueda de Relaciones Causales para aplicaciones en cardiología y Psiquiatría*. Tesis Doctoral. Universidad Carlos III de Madrid. Leganés. Enero.

Giles., David (2011). Econometrics Beat: Dave Giles' Blog, A resource for econometrics students & practitioners, recuperado de: <http://davegiles.blogspot.com.co/>

Lütkepohl, H. (2006). *New Introduction to Multiple Time Series Analysis*. Springer, Berlin.

Toda, H. Y and T. Yamamoto (1995). Statistical inferences in vector autoregressions with possibly integrated processes. *Journal of Econometrics*, 66, 225-250.

Ray, Sarbapriya (2012). *Testing Granger Causal Relationship between Macroeconomic Variables and Stock Price Behaviour: Evidence from India*. Advances in Applied Economics and Finance (AAEF), Vol. 3, No. 1, ISSN 2167-6348. India

Reader, Thomas. (2012). *Granger causality testing*. Economy & Finance, Technology. Recuperado de: <https://www.slideshare.net/ThomasReader/granger-causality-testing>.