

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

ESTUDIO DE LOS FUNDAMENTOS MATEMÁTICOS E IMPLEMENTACIÓN DEL MÉTODO PARAMÉTRICOS AR POR MEDIO DEL ALGORITMO DE YULE-WALKER Y SU COMPARACIÓN CON LOS MÉTODOS NO PARAMÉTRICOS BASADOS EN FOURIER (PERIODOGRAMA SIMPLE Y MODIFICADO)

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
TELECOMUNICACIONES**

MICHELLE STEFANNY CHILUISA MARMOL

michelle.chiluisa@epn.edu.ec

DIRECTOR: PhD. RICARDO XAVIER LLUGSI CAÑAR

ricardo.llugsi@epn.edu.ec

DMQ, agosto 2023

CERTIFICACIONES

Yo, Michelle Stefanny Chiluisa Mármol declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

Michelle Stefanny Chiluisa Mármol

Certifico que el presente trabajo de integración curricular fue desarrollado por Michelle Stefanny Chiluisa Mármol, bajo mi supervisión.

Ph.D. Ricardo Llugsi
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

MICHELLE CHILUISA

RICARDO LLUGSI

DEDICATORIA

Este trabajo está dedicado con profundo cariño a las personas que han sido fundamentales en mi vida y en mi formación académica:

A mis amados padres, Violeta Mármol y Abelardo Chiluisa, quienes han sido mi pilar y guía. Su inquebrantable apoyo y sacrificio han sido la fuente de mi motivación para salir adelante en cada paso de mi camino.

A mis queridos hermanos, Paul, Adriana, Gabriela y Vero, quienes han sido un ejemplo constante de responsabilidad y deseos de superación. Su influencia positiva ha sido un impulso importante para alcanzar mis metas académicas y personales.

A mi cuñada, Rosibel Chasin, por su constante apoyo y ayuda desinteresada. Su presencia y colaboración han sido un gran respaldo en los momentos de mayor desafío.

A mis adorables sobrinos, Heidy, Abigail, Grace, Edith, Keyra, Darío y Sebastián, quienes con su dulzura y alegría le dan luz y felicidad a mis días. Su presencia llena mi corazón de amor y me motiva a esforzarme para ser un ejemplo positivo en sus vidas.

A Esthela Mármol, quien con su cariño y amor incondicional siempre me ha brindado un segundo hogar. Su presencia ha sido un verdadero regalo que valoro enormemente.

Finalmente, dedico este trabajo a mi primo Sebastián, quien me brinda la mejor compañía, apoyo y cariño incondicional. Su memoria y legado continúan siendo una fuente de inspiración y fortaleza en mi camino. Sus consejos, aliento y amistad inquebrantable han dejado una huella imborrable en mi corazón y en mi determinación por seguir adelante. Sé que, desde el cielo, él sigue siendo mi guía y protector, motivándome a alcanzar mis metas con valentía y perseverancia.

AGRADECIMIENTO

En primer lugar, agradezco sinceramente a Dios por darme la fortaleza y determinación para avanzar en mi vida personal y profesional.

Quiero expresar mi profundo agradecimiento a mis queridos padres, quienes han sido mi apoyo incondicional en toda mi carrera.

También quiero extender mi agradecimiento a Alex Ramos y Alex Páez, cuya amistad y apoyo inquebrantable han sido invaluable durante los momentos difíciles. Los recuerdos compartidos con ellos siempre permanecerán en mi corazón y me siento afortunada de graduarme junto a ellos.

Además, quiero agradecer a mis demás compañeros, Daysi, Ibeth, Pauli y Rene, con quienes he compartido momentos felices a lo largo de este proceso de aprendizaje. No puedo dejar de mencionar a mi mejor amigo, Cristian Lema, cuyo cariño y apoyo sincero han sido una luz en mi vida desde el primer semestre.

Asimismo, mi gratitud se extiende hacia mi profesor Robin Álvarez, cuya sabiduría, dedicación y orientación han sido fundamentales para el desarrollo de mi proyecto de titulación. Su experiencia y consejos han sido una guía invaluable durante todo el proceso.

Finalmente, quiero agradecer también a mi profesor Ricardo Llugsí, quien me ha brindado su guía y apoyo en la etapa final de este proyecto. Su experiencia y conocimiento han sido de gran valor para culminar con éxito esta etapa académica.

ÍNDICE DE CONTENIDO

CERTIFICACIONESI

DECLARACIÓN DE AUTORÍAI

DEDICATORIAIII

AGRADECIMIENTOIV

ÍNDICE DE CONTENIDOV

RESUMENVII

ABSTRACTVIII

1 INTRODUCCIÓN1

1.1 OBJETIVO GENERAL2

1.2 OBJETIVOS ESPECÍFICOS2

1.3 ALCANCE2

1.4 MARCO TEÓRICO3

1.4.1 Métodos deterministas3

1.4.2 Periodograma Simple y Modificado4

1.4.3 El modelo autorregresivo4

1.4.4 Sistema de Ecuaciones Yule Walker6

1.4.5 Cálculo densidad espectral de potencia de modelos AR.7

1.4.6 Cálculo paso a paso del método paramétrico AR para un orden 2 y 108

2 METODOLOGÍA15

2.1 CÓDIGO PRINCIPAL16

2.1.1 Generación de la señal16

2.1.2 Uso de la función mi_yule y Yule-Walker de Matlab18

2.1.3 Implementación del Periodograma Simple y Modificado21

2.2 FUNCIÓN CÁLCULO DE AUTOCORRELACIÓN23

2.3 FUNCIÓN VARIANZA O POTENCIA DEL RUIDO24

2.4 FUNCIÓN CALCULO DENSIDAD ESPECTRAL DE POTENCIA25

3 RESULTADOS26

3.1 RESULTADOS26

3.1.1 Resultados en frecuencia y amplitud para diferentes ordenes26

3.1.2 Resultados de la prueba computacional para cada método29

3.1.3 Resultados en frecuencia y amplitud para diferentes ordenes con ruido AWGN31

4 CONCLUSIONES Y RECOMENDACIONES34

4.1	CONCLUSIONES	34
4.2	RECOMENDACIONES	35
5	REFERENCIAS BIBLIOGRÁFICAS	35
6	ANEXOS	36
	ANEXO I	37
	ANEXO II	40
	ANEXO III	41
	ANEXO IV	42

RESUMEN

Los fenómenos naturales se representan mediante señales, que contienen múltiples componentes con diferentes frecuencias y amplitudes. La extracción de características en dominio de la frecuencia brinda la capacidad para utilizar algoritmos de estimación espectral altamente eficaces, permitiendo identificar características de la señal que en otros dominios serían inaccesibles. Los diferentes métodos de estimación espectral no tienen un estudio que muestre claramente los pros y contras de estos.

Este proyecto de titulación tiene como objetivo evidenciar las ventajas y desventajas del método paramétrico AR, por medio del Algoritmo de Yule-Walker, en comparación con el método no paramétrico basado en el periodograma simple y modificado, mediante el estudio e implementación de estos.

Para el desarrollo de este proyecto, se hará una breve descripción de la clasificación y funcionamiento de los métodos de estimación espectral a compararse, y un ejemplo paso a paso del cálculo, desde la obtención de los coeficientes del modelo mediante el algoritmo de Yule-Walker, hasta la estimación espectral del modelo paramétrico AR.

Luego, se aplicarán estos conocimientos en la metodología, donde se implementarán diversos códigos, usados para el cálculo de la estimación espectral de los diferentes métodos. Adicionalmente, se empleará un código desarrollado específicamente denominado 'mi_yule' con el propósito de generar un material educativo que facilite la comprensión del algoritmo de Yule-Walker.

Como resultado de esta metodología, se pretende comparar estos métodos mediante cuatro pruebas que permitan discernir los pros y contras de cada uno. Las pruebas permitirán ver si hay una estimación correcta de las amplitudes, una resolución en frecuencia, analizar la carga computacional y observar el funcionamiento de los métodos frente al ruido AWGN.

PALABRAS CLAVE: Yule Walker, autocorrelación, paramétricos, periodograma simple, periodograma modificado, PSD.

ABSTRACT

Natural phenomena are represented by signals having multiple components with different frequencies and amplitudes. Frequency domain feature extraction supplies the ability to use highly efficient spectral estimation algorithms, allowing the identification of signal characteristics that would be inaccessible in other domains. However, the different spectral estimation methods lack a comprehensive study that clearly presents their pros and cons.

This degree project aims to prove the advantages and disadvantages of the parametric AR method using the Yule-Walker Algorithm, in comparison with the non-parametric method based on the simple and modified periodogram. This will be achieved through a detailed study and implementation of both methods.

For the development of this project, a brief description of the classification and operation of the spectral estimation methods to be compared will be provided. Additionally, a step-by-step example will be presented, covering the calculation process from obtaining the model coefficients using the Yule-Walker algorithm to the spectral estimation of the parametric AR model.

Then, this knowledge will be applied in the method, where several codes will be implemented and used for calculating the spectral estimation of the different methods. Additionally, a specifically developed code called 'mi_yule' will be used with the purpose of creating educational materials that ease the understanding of the Yule-Walker algorithm.

As a result of this method, we intend to compare these methods through four tests that will allow us to discern the pros and cons of each one. These tests will enable us to evaluate the accuracy of amplitude estimation, frequency resolution, analyze computational load, and see the performance of the methods against AWGN noise.

KEYWORDS: Yule-Walker, autocorrelation, parametric, simple periodogram, modified periodogram, PSD.

1 INTRODUCCIÓN

En el análisis de señales, diversos modelos pueden generar resultados semejantes; sin embargo, ciertos modelos demandan menos parámetros y demuestran mayor eficacia al representar el proceso [1]. La estimación de la PSD (*Power Spectral Density*, Densidad Espectral de Potencia) que se basa en la FFT (*Fast Fourier Transform*, Transformada Rápida de Fourier), es computacionalmente eficiente y adecuada para muchos procesos de señales. Sin embargo, este enfoque tiene limitaciones de resolución de frecuencia y una ventana implícita que causa 'fugas' y distorsiones en respuestas espectrales cercanas [2].

Una estrategia clave para abordar esta tarea, consiste en aprovechar las funciones ya existentes en Matlab y aplicarla directamente a la señal de prueba. Esto permitirá observar, cuáles de estos métodos generan los resultados más favorables y servirá como punto de partida para profundizar en la comprensión matemática y realizar implementaciones propias. De esta manera, se buscará obtener un documento didáctico que facilite la comprensión de estos algoritmos y, en un futuro, se podrán introducir mejoras.

En este proyecto, se comenzará con el estudio e implementación del modelo AR (*Autoregressive*, Autorregresivo), utilizando el algoritmo de Yule-Walker con la función 'pyulear', que entrega la PSD de una señal discreta en el dominio del tiempo [3]. Se compararán este algoritmo con los métodos tradicionales basados en Fourier, como el periodograma simple y modificado, que son ampliamente utilizados en diversas publicaciones.

Cabe resaltar que los métodos basados en Fourier, presentan dos importantes desafíos, como el problema de resolución en frecuencia, que restringe la habilidad para discernir entre las respuestas espectrales de dos o más señales y el problema de la ventana implícita de los datos, que ocurre cuando se procesa la FFT y afecta la capacidad de distinguir componentes muy cercanas en frecuencia, debido los lóbulos laterales se montan en el lóbulo principal positiva y negativamente perjudicando la PSD [4].

A través de este proyecto, se pretende analizar en profundidad estos algoritmos de estimación espectral y comparar su rendimiento en diferentes escenarios. Se realizarán pruebas, que evaluarán la correcta estimación de las amplitudes, la resolución en frecuencia, la carga computacional y el comportamiento frente al ruido AWGN (*Additive White Gaussian Noise*, Adición de Ruido Blanco Gaussiano) [5]. Este estudio permitirá discernir las ventajas y desventajas de cada método, brindando una perspectiva clara y fundamentada para su aplicación en el análisis de señales complejas en el futuro.

1.1 OBJETIVO GENERAL

El objetivo general de este proyecto es realizar un estudio detallado de los fundamentos matemáticos y la implementación del método paramétrico AR a través del Algoritmo de Yule-Walker, y compararlo con los métodos no paramétricos basados en Fourier, como el periodograma simple y modificado. El propósito es analizar sus ventajas y desventajas, evaluando su rendimiento en diferentes escenarios, con el fin de proporcionar una comprensión clara y fundamentada de estos algoritmos de estimación espectral y su aplicación en el análisis de señales complejas.

1.2 OBJETIVOS ESPECÍFICOS

1. Realizar un estudio exhaustivo de los fundamentos matemáticos del método paramétrico AR y del Algoritmo de Yule-Walker, comprendiendo su funcionamiento y los conceptos teóricos que sustentan su aplicación en la estimación espectral.
2. Implementar de manera práctica el método paramétrico AR utilizando el Algoritmo de Yule-Walker en el entorno de programación Matlab, desarrollando funciones y scripts que permitan calcular el espectro de potencias de señales de prueba y analizar los resultados obtenidos.
3. Comparar el rendimiento del método paramétrico AR con los métodos no paramétricos basados en Fourier, como el periodograma simple y modificado, mediante pruebas y simulaciones en diferentes escenarios. Evaluar la capacidad de detección de componentes de amplitud pequeña y la resolución en frecuencia de cada método para identificar sus ventajas y desventajas en la estimación espectral.

1.3 ALCANCE

Utilizando el software Matlab en su versión R2021a, se generarán señales discretas que constan de múltiples componentes con diferentes frecuencias y amplitudes. Estas señales serán sometidas a pruebas para evaluar la resolución temporal, la resolución en frecuencia y la capacidad de detección de cada uno de los métodos propuestos.

Las evaluaciones se realizarán mediante el uso de librerías y funciones propias o externas a Matlab. Además, se empleará una función desarrollada específicamente, llamada 'mi_yule', para obtener la PSD de la señal en tiempo discreto. A partir de los resultados obtenidos en las pruebas, se llevará a cabo una comparación detallada entre los cuatro métodos planteados, otorgando mayor importancia a aspectos como la resolución

temporal, la resolución en frecuencia, la capacidad de detección de componentes de baja amplitud y la carga computacional.

Como resultado de esta comparación, se determinará qué método demuestra mejores prestaciones y características en el análisis espectral de señales, contribuyendo a una comprensión más profunda de los fundamentos matemáticos y a una identificación más precisa de la eficacia de cada método.

1.4 MARCO TEÓRICO

El propósito de este capítulo es brindar una descripción de los métodos determinísticos paramétricos, enfocados en modelo AR por medio del algoritmo Yule Walker y hacer una comparación con los métodos determinísticos no paramétricos, enfocados en el periodograma simple y el periodograma modificado, siendo este último el más usado hoy en día.

La estimación espectral es una técnica utilizada en varias áreas de las comunicaciones, en el análisis de señales y series de tiempo, para obtener información sobre la distribución de energía o potencia de una señal en el dominio de la frecuencia. En particular, se utiliza para estimar la PSD de una señal [6].

1.4.1 Métodos deterministas

La estimación de las componentes de frecuencia de una señal se puede abordar desde diferentes modelos matemáticos, entre los cuales existen innumerables clasificaciones. Si se plantea desde la visión de si existe probabilidad o no, se pueden clasificar en métodos deterministas y estocásticos [7].

En el análisis de señales, los métodos estocásticos se basan en situaciones donde hay incertidumbre o aleatoriedad en los datos, mientras que, los métodos determinísticos se basan en ecuaciones matemáticas o modelos bien definidos [8].

Los métodos determinísticos a su vez se clasifican en:

-Métodos deterministas no paramétricos. – Son métodos que están basados en Fourier y hacen uso de estimaciones basadas en promedios, funciones de ventana, o métodos basados en transformadas. Entre estos métodos se tiene el periodograma simple y modificado [9].

-Métodos deterministas paramétricos. – Son métodos que representan a la señal mediante un modelo y, en base a los coeficientes de este, se procede a estimar las

componentes de frecuencia. Entre estos métodos se trabajará con el modelo autorregresivo o AR que se calcula a partir del algoritmo de Yule-Walker, con el fin de obtener los coeficientes del modelo AR [10].

1.4.2 Periodograma Simple y Modificado

-Periodograma Simple. – Es la estimación espectral más básica y directa, emplea el enventanado rectangular donde a un segmento de señal se aplica una ventana rectangular [11]. El problema de utilizar el periodograma simple, sin aplicar ventanas, radica en que el inicio y final del segmento analizado, pueden contener valores distintos de cero. Esto puede equivaler a funciones de tipo escalón, que afectan el contenido de altas frecuencias al realizar la FFT [12].

-Periodograma Modificado. – Consiste en aplicar distintos tipos de ventanas a una señal. Empezando por multiplicar dicha señal temporal por una de las ventanas antes estudiadas y luego obtener su FFT [13]. El propósito del periodograma modificado es abordar el problema de enmascaramiento espectral causado por el periodograma simple. Este problema se deriva de los altos niveles de lóbulos secundarios asociados con el uso de la ventana rectangular [14]. El periodograma modificado proporciona una estimación más precisa de la distribución de energía en el dominio de la frecuencia.

Tanto el periodograma simple como el modificado son métodos de estimación no paramétrica y, se basan en el análisis de la transformada de Fourier de una señal. Sin embargo, el periodograma modificado utiliza ventanas ponderadas y segmentación para reducir la fuga espectral en comparación con el periodograma simple [15].

1.4.3 El modelo autorregresivo

En la actualidad el modelo AR, es ampliamente utilizado en la estimación de la PSD. La señal discreta $x(n)$ se puede representar como la salida de un sistema que contiene una entrada $W(n)$ y vinculado mediante la siguiente relación [16]:

$$x(n) = - \sum_{k=0}^N a_k x(n-k) + W(n) \quad (1)$$

Donde:

- N : Orden del modelo
- $x(n)$: Muestras de la señal discreta real.

- a_k : Coeficientes AR (parámetros), donde, $k=1, 2, \dots, p$.
- $W(n)$: AWGN, proceso aleatorio con un espectro plano y una distribución normal $N(0,1)$, se asume que tiene varianza σ^2 y media cero

El modelo presentado en la ecuación (1) se denomina Modelo AR(N). A continuación, se explica cómo obtener las ecuaciones de Yule-Walker, que son el punto de partida para el desarrollo del método paramétrico [17].

Reescribiendo la ecuación (1), se tiene:

$$\sum_{k=0}^N a_k x(n-k) + x(n) = W(n); a_0 = 1 \quad (2)$$

De la ecuación (2) se puede decir que, la entrada del sistema se puede interpretar como la disparidad entre el corriente valor de $x(n)$ y una mezcla lineal de sus valores previos. Esta discrepancia también puede considerarse como una función que sigue una distribución gaussiana de errores [16].

En base a la ecuación (2) se aplica el principio de ortogonalidad para obtener los parámetros a_k , que indica que [16]:

$$\sum_{k=0}^N a_k E[x(n)x(n-l)] = 0; \quad (3)$$

A continuación, se multiplica por $x(n-l)$ y se obtiene la esperanza de cada lado de la ecuación (2) y se reemplaza la ecuación (3).

$$\sum_{k=0}^N a_k E[x(n-k)x(n-l)] + E[x(n)x(n-l)] = E[W(n)x(n-l)]; \quad (4)$$

$R_{xx}(l-k)$

0

$0, l > 0$
 $\sigma^2, l = 0$

Se procede a eliminar el término que se hizo 0 en la ecuación (4) y se tiene la ecuación (5).

$$x(n-l) = - \sum_{k=1}^N a_k x(n-l-k) + W(n-l); \quad (5)$$

Para un R_{xx} tal que,

$$R_{xx}(k) = \begin{cases} \sigma^2 & k = 0 \\ 0 & k \neq 0 \end{cases} \quad (6)$$

Reemplazando la ecuación (4) y (6) en la ecuación (5),

$$\sum_{k=0}^N a_k R_{xx}(l-k) = \begin{cases} 0, & l > 0 \\ \sigma^2, & l = 0 \end{cases}, a_0 = 1 \quad (7)$$

Para $l > 0$,

$$\sum_{k=1}^N a_k R_{xx}(l-k) = -r_{xx}(l) \quad (8)$$

1.4.4 Sistema de Ecuaciones Yule Walker

La función de autocorrelación discreta de la señal $x(n)$ está representada como r_{xx} y es una función par $r_{xx}(-m) = r_{xx}(m)$. Debido a esta simetría, la ecuación (8) puede expresarse de manera matricial de la siguiente forma [16]:

$$\begin{matrix} l = 1 \\ l = 2 \\ l = 3 \\ \vdots \\ l = N \end{matrix} \begin{bmatrix} R_{xx}(0) & R_{xx}(-1) & R_{xx}(-2) & \cdots & R_{xx}(1-N) \\ R_{xx}(1) & R_{xx}(0) & R_{xx}(-1) & \cdots & R_{xx}(2-N) \\ R_{xx}(2) & R_{xx}(1) & R_{xx}(0) & \cdots & R_{xx}(3-N) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{xx}(N-1) & R_{xx}(N-2) & R_{xx}(N-3) & \cdots & R_{xx}(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_N \end{bmatrix} \quad (9)$$

Sistema de ecuaciones Yule Walker con N ecuaciones y N incógnitas

Denotando la matriz de correlación como R_{xx} , el vector de coeficientes como a y el r_{xx} como vector, se tiene:

$$R_{xx}a = -r_{xx} \quad (10)$$

De la ecuación (10) se despeja a_k del modelo AR y se puede observar se pueden obtener invirtiendo la matriz R_{xx} .

$$a = R_{xx}^{-1} \cdot r_{xx} \quad (11)$$

dato a_k , se puede resolver la varianza o potencia de ruido a la entrada del filtro en la ecuación (12).

$$\sigma^2 = R_{xx}(0) + \sum_{k=0}^N a_k R_{xx}(-k) \quad (12)$$

Obtenidos a_k y σ^2 se procede al siguiente paso. $R_{xx}[0] = 1$

1.4.5 Cálculo densidad espectral de potencia de modelos AR.

La forma general la PSD de un modelo AR se expresa a continuación [18]:

$$PSD_{AR(z)} = \frac{1}{|A(z)|^2} * \sigma^2 * \Delta t \quad (13)$$

$A(z)$ es un polinomio que en el caso general contiene coeficientes complejos, coeficientes a_k , la frecuencia de la señal, el orden del modelo, la duración de la señal y se calcula como se observa en la siguiente ecuación (14):

$$A(z) = 1 + a_{k,1} * e^{-j2k\pi f \Delta t} + a_{k,2} * e^{-j2k\pi f \Delta t} + \dots + a_{k,k} * e^{-j2k\pi f \Delta t} \quad (14)$$

Donde

$$e^{-j2k\pi f \Delta t} = \cos(2k\pi f \Delta t) - j\text{sen}(2k\pi f \Delta t) \quad (15)$$

Reemplazando, la ecuación (15) en la ecuación (14) se obtiene

$$\begin{aligned} A(z) = & 1 + a_{k,1} * (\cos(2\pi f \Delta t) - j\text{sen}(2\pi f \Delta t)) + a_{k,2} \\ & * (\cos(4\pi f \Delta t) - j\text{sen}(4\pi f \Delta t)) + \dots + a_{k,k} \\ & * (\cos(2k\pi f \Delta t) - j\text{sen}(2k\pi f \Delta t)) \end{aligned} \quad (16)$$

Ya que la ecuación (16) es compleja, se puede separar su parte real de la imaginaria

$$A(z) = \quad (17)$$

Si se eleva al cuadrado esta expresión

$$A(z)^2 = \quad (18)$$

Finalmente, la PSD que se observa esta expresada en la ecuación (13), se la puede ver ampliada en la ecuación (19).

$$PSD_{AR(z)} = \frac{\sigma^2 * \Delta t}{|A(z)|^2} \quad (19)$$

La función $PSD_{AR(z)}$, se puede utilizar en todas las situaciones de estimación de parámetros AR cuando los datos son completamente reales. También es una función ininterrumpida en el dominio de frecuencia, lo cual es una característica distintiva del espectro AR. Los

programas creados emplean la relación $PSD_{AR(Z)}$ para calcular el espectro de potencia utilizando el modelo AR [18].

1.4.6 Cálculo paso a paso del método paramétrico AR para un orden 2 y 10

Se plantea la señal de ejemplo:

$$\begin{aligned}
 x(t) = & 1 \cdot \sin(2\pi \cdot 100 \cdot t) + 1 \cdot \sin(2\pi \cdot 108 \cdot t) + \dots \\
 & 0.5 \cdot \sin(2\pi \cdot 120 \cdot t) + 0.5 \cdot \sin(2\pi \cdot 127 \cdot t) + \dots \\
 & 0.1 \cdot \sin(2\pi \cdot 140 \cdot t) + 0.1 \cdot \sin(2\pi \cdot 146 \cdot t) + \dots \\
 & 0.05 \cdot \sin(2\pi \cdot 160 \cdot t) + 0.05 \cdot \sin(2\pi \cdot 165 \cdot t) + \dots \\
 & 0.01 \cdot \sin(2\pi \cdot 180 \cdot t) + 0.01 \cdot \sin(2\pi \cdot 184 \cdot t) + \dots \\
 & 0.005 \cdot \sin(2\pi \cdot 200 \cdot t) + 0.005 \cdot \sin(2\pi \cdot 203 \cdot t) + \dots \\
 & 0.001 \cdot \sin(2\pi \cdot 220 \cdot t) + 0.001 \cdot \sin(2\pi \cdot 222 \cdot t) + \dots \\
 & 0.0005 \cdot \sin(2\pi \cdot 240 \cdot t) + 0.0005 \cdot \sin(2\pi \cdot 241 \cdot t);
 \end{aligned}$$

La señal $x(t)$ es una señal periódica continua formada por una suma de senos, se necesita que esta señal sea discreta $x(n)$ para aplica técnicas de procesamiento de señales y análisis espectral [19]. En este caso, se usa directamente la función de autocorrelación que permite obtener los datos de la matriz en base a la señal $x(t)$.

Se introduce una función de autocorrelación, para ello se toma 2 variables aleatorias con diferentes tiempos t_1 y t_2 . De la diferencia de $t_2 - t_1$ se obtiene τ . La función de autocorrelación $R_{xx}(\tau)$, relaciona la potencia media del proceso con su distribución a lo largo de la frecuencia, a través del llamado teorema de Wiener-Khinchin [21].

En estos procesos, la energía tiende a ser infinita, por lo que el enfoque se centra en la distribución de potencia con la frecuencia. Para procesos aleatorios estacionarios, se utiliza la función de autocorrelación [4]. Se observa este proceso con la señal discreta en la ecuación (4).

$$R_{xx}(\tau) = E[x(t + \tau)x(t)] \quad (20)$$

proporciona la base para el análisis de espectro, en lugar del proceso aleatorio $x(t)$ en sí mismo. El teorema de Wiener-Khinchin relaciona $R_{xx}(\tau)$ a través de la transformada de Fourier. Para un proceso ergódico donde sus propiedades estadísticas se mantienen constantes a lo largo del tiempo y son iguales en promedio temporal para diferentes muestras, la función de autocorrelación estadística puede equipararse a [4]:

$$R_{xx}(\tau) = \lim_{N \rightarrow \infty} \frac{1}{N} \int_{-N/2}^{N/2} x(t + \tau)x(t) dt \quad (21)$$

La ecuación (21) muestra el proceso de muestreo que se debe seguir para la discretización de la señal y posteriormente la obtención del sistema de ecuaciones Yule Walker.

Un sistema correctamente formulado (con una matriz de coeficientes cuadrada R_{xx}) se caracteriza por tener un equilibrio entre el número de restricciones (ecuaciones o filas de R_{xx}) y el número de incógnitas (r_{xx}). Además, R_{xx} posee simetría y su rango es completo, asegurando así la invertibilidad del sistema [22].

- **Calculo para un orden 2**

Debido a que la discretización de la señal es un proceso que involucra un extenso cálculo matemático. Se realiza el cálculo de los coeficientes AR a partir del Sistema de Ecuaciones Yule Walker para un orden 2, mediante Matlab con la función 'mi_rxx' se obtiene la ecuación (10).

$$\begin{bmatrix} 2.2601 & 1.2377 \\ 1.2377 & 2.2601 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -1.2377 \\ -1.0766 \end{bmatrix} \quad (22)$$

Entonces, se despeja la ecuación (1.22)

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 2.2601 & 1.2377 \\ 1.2377 & 2.2601 \end{bmatrix}^{-1}}_{R_{xx}^{-1}} \underbrace{\begin{bmatrix} -1.2377 \\ -1.0766 \end{bmatrix}}_{r_{xx}} \quad (23)$$

Se resuelve la inversa por el método de adjunción que esta dado por la siguiente formula:

$$R_{xx}^{-1} = \frac{(R_{xx})^T}{|R_{xx}|} \quad (24)$$

Donde

- R_{xx}^{-1} : Matriz inversa
- $R_{xx} \vee$: Determinante de la matriz
- R_{xx} : Matriz adjunta
- $(R_{xx})^T$: Transpuesta de la matriz adjunta

Se tiene

$$R_{xx} = \begin{bmatrix} 2.2601 & 1.2377 \\ 1.2377 & 2.2601 \end{bmatrix} \quad (25)$$

Se calcula, R_{xx}

$$R_{xx} = \begin{bmatrix} (-1)^{1+1} * \det(R_{xx_{22}}) & (-1)^{1+2} * \det(R_{xx_{21}}) \\ (-1)^{2+1} * \det(R_{xx_{12}}) & (-1)^{2+2} * \det(R_{xx_{11}}) \end{bmatrix} \quad (26)$$

$$R_{xx} = \begin{bmatrix} (-1)^2 * \det(2.2601) & (-1)^3 * \det(1.2377) \\ (-1)^3 * \det(1.2377) & (-1)^4 * \det(2.2601) \end{bmatrix} \quad (27)$$

$$R_{xx} = \begin{bmatrix} 1 * 2.2601 & -1 * 1.2377 \\ -1 * 1.2377 & 1 * 2.2601 \end{bmatrix} \quad (28)$$

$$R_{xx} = \begin{bmatrix} 2.2601 & -1.2377 \\ -1.2377 & 2.2601 \end{bmatrix} \quad (29)$$

A continuación, se calcula la transpuesta y en este caso debido a que la matriz es simétrica, la transpuesta es igual a la adjunta.

$$(R_{xx} | \quad)^T = \begin{bmatrix} 2.2601 & -1.2377 \\ -1.2377 & 2.2601 \end{bmatrix} \quad (30)$$

Se calcula el determinante

$$|R_{xx}| = (2.2601)^2 - (1.2377)^2 = 3.5761 \quad (31)$$

Se reemplaza las ecuaciones (1.30) y (1.31) en nuestra ecuación (1.32)

$$R_{xx}^{-1} = \frac{\begin{bmatrix} 2.2601 & -1.2377 \\ -1.2377 & 2.2601 \end{bmatrix}}{3.5761} = \begin{bmatrix} 0.6320 & -0.3461 \\ -0.3461 & 0.6320 \end{bmatrix} \quad (32)$$

Finalmente se calcula los coeficientes a_k

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0.6320 & -0.3461 \\ -0.3461 & 0.6320 \end{bmatrix} * \begin{bmatrix} -1.2377 \\ -1.0766 \end{bmatrix} \quad (33)$$

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} (0.6320 * -1.2377) + (-0.3461 * -1.0766) \\ (-0.3461 * -1.2377) + (0.6320 * -1.0766) \end{bmatrix} \quad (34)$$

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -0.4096 \\ -0.2520 \end{bmatrix} \quad (35)$$

```
>> parm
parm =
    1.0000
   -0.4096
   -0.2520
```

Figura 1. Coeficientes AR obtenidos por el código de Matlab orden 2

Se puede observar en la Figura 1 que los datos calculados coinciden con los obtenidos con el código.

- **Calculo para un orden 10**

Se obtiene el siguiente sistema de ecuaciones Yule Walker para un orden 10, mediante Matlab.

$$\begin{bmatrix} 2.25321.20421.05790.81140.52890.1850 - 0.1611 - 0.4886 - 0.7820 & (36) \\ 1.20422.25321.20421.05790.81140.52890.1850 - 0.1611 - 0.4886 - \\ 1.05791.20422.25321.20421.05790.81140.52890.1850 - 0.1611 - (\\ 0.81141.05791.20422.25321.20421.05790.81140.52890.1850 - 0. \\ 0.52890.81141.05791.20422.25321.20421.05790.81140.52890.1 \\ 0.18500.52890.81141.05791.20422.25321.20421.05790.81140.5 \\ -0.16110.18500.52890.81141.05791.20422.25321.20421.05790.8 \\ -0.4886 - 0.16110.18500.52890.81141.05791.20422.25321.20421 \\ -0.7820 - 0.4886 - 0.16110.18500.52890.81141.05791.20422.2532 \\ -0.9997 - 0.7820 - 0.4886 - 0.16110.18500.52890.81141.05791.204 \end{bmatrix}$$

$$* \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \end{bmatrix} = \begin{bmatrix} -1.2042 \\ -1.0579 \\ -0.8114 \\ -0.5289 \\ -0.1850 \\ 0.1611 \\ 0.4886 \\ 0.7820 \\ 0.9997 \\ 1.1511 \end{bmatrix}$$

Entonces

$$\begin{aligned}
 & \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \end{bmatrix} \tag{37} \\
 = & \begin{bmatrix} 2.25321.20421.05790.81140.52890.1850 - 0.1611 - 0.4886 - 0.7820 \\ 1.20422.25321.20421.05790.81140.52890.1850 - 0.1611 - 0.4886 - \\ 1.05791.20422.25321.20421.05790.81140.52890.1850 - 0.1611 - \\ 0.81141.05791.20422.25321.20421.05790.81140.52890.1850 - 0. \\ 0.52890.81141.05791.20422.25321.20421.05790.81140.52890. \\ 0.18500.52890.81141.05791.20422.25321.20421.05790.81140. \\ -0.16110.18500.52890.81141.05791.20422.25321.20421.05790 \\ -0.4886 - 0.16110.18500.52890.81141.05791.20422.25321.2042 \\ -0.7820 - 0.4886 - 0.16110.18500.52890.81141.05791.20422.253 \\ -0.9997 - 0.7820 - 0.4886 - 0.16110.18500.52890.81141.05791.20 \\ \end{bmatrix} \\
 * & \begin{bmatrix} -1.2042 \\ -1.0579 \\ -0.8114 \\ -0.5289 \\ -0.1850 \\ 0.1611 \\ 0.4886 \\ 0.7820 \\ 0.9997 \\ 1.1511 \end{bmatrix} \\
 & \underbrace{\hspace{15em}}_{R_{xx}^{-1}} \underbrace{\hspace{2em}}_{r_{xx}}
 \end{aligned}$$

Se resuelve la inversa por el método de adjunción que esta dado por la siguiente formula:

$$R_{xx}^{-1} = \frac{(R_{xx})^T}{|R_{xx}|} \tag{38}$$

Se tiene

$$\begin{aligned}
 & R_{xx} \tag{39} \\
 = & \begin{bmatrix} 2.25321.20421.05790.81140.52890.1850 - 0.1611 - 0.4886 - 0. \\ 1.20422.25321.20421.05790.81140.52890.1850 - 0.1611 - 0. \\ 1.05791.20422.25321.20421.05790.81140.52890.1850 - 0.1 \\ 0.81141.05791.20422.25321.20421.05790.81140.52890.18 \\ 0.52890.81141.05791.20422.25321.20421.05790.81140.5 \\ 0.18500.52890.81141.05791.20422.25321.20421.05790.8 \\ -0.16110.18500.52890.81141.05791.20422.25321.20421.0 \\ -0.4886 - 0.16110.18500.52890.81141.05791.20422.25321 \\ -0.7820 - 0.4886 - 0.16110.18500.52890.81141.05791.2042 \\ -0.9997 - 0.7820 - 0.4886 - 0.16110.18500.52890.81141.057 \\ \end{bmatrix}
 \end{aligned}$$

Por lo tanto, R_{xx}

$$R_{xx} \quad (40)$$

$$= \begin{bmatrix} 43.7820 & -10.3441 & -8.6703 & -5.6714 & -3.7625 & -0.6089 & 0.2514 \\ -10.3441 & 144.8332 & -9.4607 & -8.0792 & -5.1437 & -3.5100 & -0.4163 \\ -8.6703 & -9.4607 & 45.5756 & -8.9641 & -7.6362 & -4.9322 & -3.3497 \\ -5.6714 & -8.0792 & -8.9641 & 45.9076 & -8.6709 & -7.4990 & -4.8337 \\ -3.7625 & -5.1437 & -7.6362 & -8.6709 & 46.1372 & -8.5904 & -7.4990 \\ -0.6089 & -3.5100 & -4.9322 & -7.4990 & -8.5904 & 46.1372 & -8.6709 \\ 0.2514 & -0.4163 & -3.3497 & -4.8337 & -7.4990 & -8.6709 & 45.9076 \end{bmatrix}$$

Ahora, se procede a calcular la transpuesta y en este caso debido a que la matriz es simétrica, la transpuesta es igual a la adjunta.

$$(R_{xx} | I)^T \quad (41)$$

$$= \begin{bmatrix} 43.7820 & -10.3441 & -8.6703 & -5.6714 & -3.7625 & -0.6089 & 0.2514 \\ -10.3441 & 144.8332 & -9.4607 & -8.0792 & -5.1437 & -3.5100 & -0.4163 \\ -8.6703 & -9.4607 & 45.5756 & -8.9641 & -7.6362 & -4.9322 & -3.3497 \\ -5.6714 & -8.0792 & -8.9641 & 45.9076 & -8.6709 & -7.4990 & -4.8337 \\ -3.7625 & -5.1437 & -7.6362 & -8.6709 & 46.1372 & -8.5904 & -7.4990 \\ -0.6089 & -3.5100 & -4.9322 & -7.4990 & -8.5904 & 46.1372 & -8.6709 \\ 0.2514 & -0.4163 & -3.3497 & -4.8337 & -7.4990 & -8.6709 & 45.9076 \end{bmatrix}$$

Se calcula el determinante

$$|R_{xx}| = 55.0259 \quad (42)$$

Se reemplaza las ecuaciones (41) y (42) en nuestra ecuación (43)

$$R_{xx}^{-1} \quad (43)$$

$$= \begin{bmatrix} 0.7957 & -0.1880 & -0.1576 & -0.1031 & -0.0684 & -0.0111 & 0.0368 \\ -0.1880 & 0.8148 & -0.1719 & -0.1468 & -0.0935 & -0.0638 & -0.0076 \\ -0.1576 & -0.1719 & 0.8283 & -0.1629 & -0.1388 & -0.0896 & -0.0609 \\ -0.1031 & -0.1468 & -0.1629 & 0.8343 & -0.1576 & -0.1363 & -0.0878 \\ -0.0684 & -0.0935 & -0.1388 & -0.1576 & 0.8385 & -0.1561 & -0.1363 \\ -0.0111 & -0.0638 & -0.0896 & -0.1363 & -0.1561 & 0.8385 & -0.1576 \\ 0.0368 & -0.0076 & -0.0609 & -0.0878 & -0.1363 & -0.1576 & 0.8343 \end{bmatrix}$$

Finalmente se procede al cálculo los coeficientes a_k

$$\begin{aligned}
 & \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \end{bmatrix} \tag{44} \\
 = & \begin{bmatrix} 0.7957 - 0.1880 - 0.1576 - 0.1031 - 0.0684 - 0.01110.03680 \\ -0.18800.8148 - 0.1719 - 0.1468 - 0.0935 - 0.0638 - 0.00760 \\ -0.1576 - 0.17190.8283 - 0.1629 - 0.1388 - 0.0896 - 0.0609 - \\ -0.1031 - 0.1468 - 0.16290.8343 - 0.1576 - 0.1363 - 0.0878 - (\\ -0.0684 - 0.0935 - 0.1388 - 0.15760.8385 - 0.1561 - 0.1363 - 0. \\ -0.0111 - 0.0638 - 0.0896 - 0.1363 - 0.15610.8385 - 0.1576 - 0. \\ 0.0368 - 0.0076 - 0.0609 - 0.0878 - 0.1363 - 0.15760.8343 - 0.1 \\ 0.07630.0372 - 0.0073 - 0.0609 - 0.0896 - 0.1388 - 0.16290.82 \\ 0.11870.07640.0372 - 0.0076 - 0.0638 - 0.0935 - 0.1468 - 0.1 \\ 0.14190.11870.07630.0368 - 0.0111 - 0.0684 - 0.1031 - 0.15 \\ -1.2042 \\ -1.0579 \\ -0.8114 \\ -0.5289 \\ -0.1850 \\ 0.1611 \\ 0.4886 \\ 0.7820 \\ 0.9997 \\ 1.1511 \end{bmatrix} \\
 * & \begin{bmatrix} -1.2042 \\ -1.0579 \\ -0.8114 \\ -0.5289 \\ -0.1850 \\ 0.1611 \\ 0.4886 \\ 0.7820 \\ 0.9997 \\ 1.1511 \end{bmatrix}
 \end{aligned}$$

$$\begin{aligned}
 & \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \end{bmatrix} = \begin{bmatrix} -2.0411 \\ 1.7347 \\ -0.1941 \\ -0.1466 \\ -0.0439 \\ 0.0670 \\ 0.1400 \\ 0.1492 \\ -0.1686 \\ 0.0982 \end{bmatrix} \tag{45}
 \end{aligned}$$

```

parm =
    1.0000
   -2.0411
    1.7347
   -0.1941
   -0.1466
   -0.0439
    0.0670
    0.1400
    0.1492
   -0.1686
    0.0982

```

Figura 1. Coeficientes AR obtenidos por el código de Matlab orden 10

Se puede observar en la Figura 2 que los datos calculados coinciden con los obtenidos con el código.

2 METODOLOGÍA

En el presente capítulo, se explicarán a detalle los pasos llevados a cabo en la implementación de los códigos usados para el cálculo de la autocorrelación, su uso para la conformación del sistema de ecuaciones de Yule-Walker y la solución para determinar los coeficientes AR mediante la matriz inversa para a continuación estimar el espectro de potencias en base a estos coeficientes AR. Finalmente se llevará a cabo una comparativa entre el método Yule Walker y los Periodogramas Simple y Modificado para obtener la PSD. Además, se analizará la carga computacional empleada por cada método.

Para esta comparación, se realizarán 4 pruebas con el objetivo de evaluar diferentes aspectos de los métodos y su desempeño frente a distintos escenarios.

Para esta comparación se realizará 4 pruebas donde se evaluará la estimación correcta de amplitudes (amplitudes en pares), resolución en frecuencia con reducción paulatina de separación entre frecuencias de 8,7,6,5,4,3,2,1 para la detección de componentes pequeños: amplitudes fueron reduciéndose (en pares, desde 1 hasta 0.0005), la carga computacional de cada uno de estos métodos y la prueba de estos métodos frente al ruido AWGN.

1. Estimación correcta de amplitudes

En la primera prueba, se evaluará la capacidad de los métodos para estimar correctamente las amplitudes. Se utilizarán 8 amplitudes diferentes, que variarán entre las 16 frecuencias, con el propósito de observar de manera más precisa la correcta estimación de estas. Las amplitudes utilizadas serán: 1, 0.5, 0.1, 0.05, 0.01, 0.005, 0.001 y 0.0005.

2. Resolución en frecuencia

La segunda prueba se centrará en evaluar la resolución en frecuencia de los métodos. Se utilizarán las 16 componentes de frecuencia en pares, con una reducción gradual de la separación entre ellas, siguiendo el patrón descendente de 8, 7, 6, 5, 4, 3, 2 y 1. Esto permitirá detectar componentes pequeñas que estén lo suficientemente alejadas entre sí.

3. Prueba de la carga computacional de cada uno de los métodos

En la tercera prueba, se evaluará la carga computacional de cada método. Se utilizarán 16 componentes de frecuencia relativamente altas en un rango de 100 a 241 Hz, lo cual permitirá evaluar el límite computacional de cada método propuesto.

4. Métodos frente al ruido (AWGN).

Por último, en la cuarta prueba se aplicará ruido AWGN y se observará la estimación de frecuencia y amplitud por parte de cada método.

Estas pruebas permitirán comparar el desempeño de los diferentes métodos en términos de estimación de amplitudes, resolución en frecuencia, carga computacional y robustez frente al ruido. El análisis de estos aspectos proporcionará información valiosa sobre las fortalezas y limitaciones de cada método en el contexto de la estimación de la PSD.

2.1 CÓDIGO PRINCIPAL

2.1.1 Generación de la señal

Primero se inicia el código importando las bibliotecas necesarias y limpiando el espacio de trabajo. A continuación, se definen algunas variables, como la duración de la señal, la frecuencia máxima a muestrear, la frecuencia de muestreo y el tiempo de muestreo.

Para la señal de prueba compuesta por varias sinusoides con 16 diferentes frecuencias donde la separación entre estas será cada vez menor y 8 diferentes amplitudes. A continuación, se genera ruido AWGN utilizando la función 'randn' [23]. Luego, se agrega el ruido a la señal original. Se solicita al usuario que ingrese el grado del modelo de autorregresión. A continuación, se grafican la señal original y la señal con ruido en las figuras 3 y 4 respectivamente.

```
%ESCUELA POLITECNICA NACIONAL
%Michelle Chiluisa
clc%Limpia la ventana de comandos,
clear all%Borra todas las variables del espacio de trabajo
close all%Cierra todas las figuras existentes.
duracion = 10;%Duracion 10 segundos
fmax=241;%Frecuencia maxima a muestrear
Fs=10*fmax;%Frecuencia de muestreo
ts=1/Fs;%Tiempo de muestreo
t=0:ts:duracion;%Vector de tiempo

xn=1*sin(2*pi*100*t)+1*sin(2*pi*108*t)+...
0.5*sin(2*pi*120*t)+0.5*sin(2*pi*127*t)+...
0.1*sin(2*pi*140*t)+0.1*sin(2*pi*146*t)+...
0.05*sin(2*pi*160*t)+0.05*sin(2*pi*165*t)+...
```

```

0.01*sin(2*pi*180*t)+0.01*sin(2*pi*184*t)+...
0.005*sin(2*pi*200*t)+0.005*sin(2*pi*203*t)+...
0.001*sin(2*pi*220*t)+0.001*sin(2*pi*222*t)+...
0.0005*sin(2*pi*240*t)+0.0005*sin(2*pi*241*t);
%Señal con 8 diferentes amplitudes y 16 frecuencias diferentes

N=length(xn);% Calcula la longitud de la señal 'x'
awgn=randn(1,length(xn));% Genera ruido blanco gaussiano de longitud N
x=xn+awgn;% Suma la señal 'x' con el ruido 'awgn'
l=input('Ingrese el grado del modelo:');% Solicita al usuario que ingrese el
grado del modelo
figure(1)% Crea una nueva figura
stem(t,xn)% Grafica la señal 'x'
grid minor % Agrega una cuadrícula de fondo al gráfico
xlabel('tiempo [segundos]')% Establece una etiqueta en el eje x del gráfico
ylabel('Amplitud [u]')% Establece una etiqueta en el eje y del gráfico.
title('Señal Discreta ')% Establece un título para el gráfico
figure(2)% Crea una nueva figura
stem(t,x)% Grafica la señal 'x1' (señal original + ruido)
grid minor % Agrega una cuadrícula de fondo al gráfico
xlabel('tiempo [segundos]')% Establece una etiqueta en el eje x del gráfico
ylabel('Amplitud [u]')% Establece una etiqueta en el eje y del gráfico
title('Señal Discreta + Ruido')% Establece un título para el gráfico

```

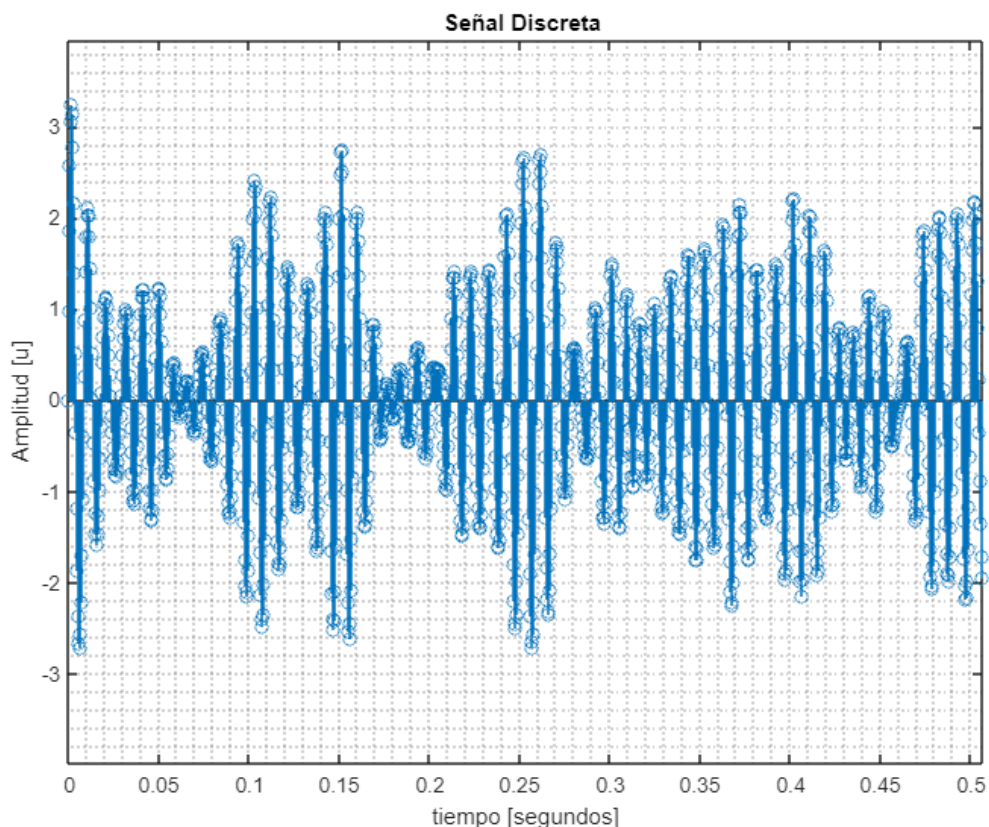


Figura 3. Señal discreta de 16 componentes

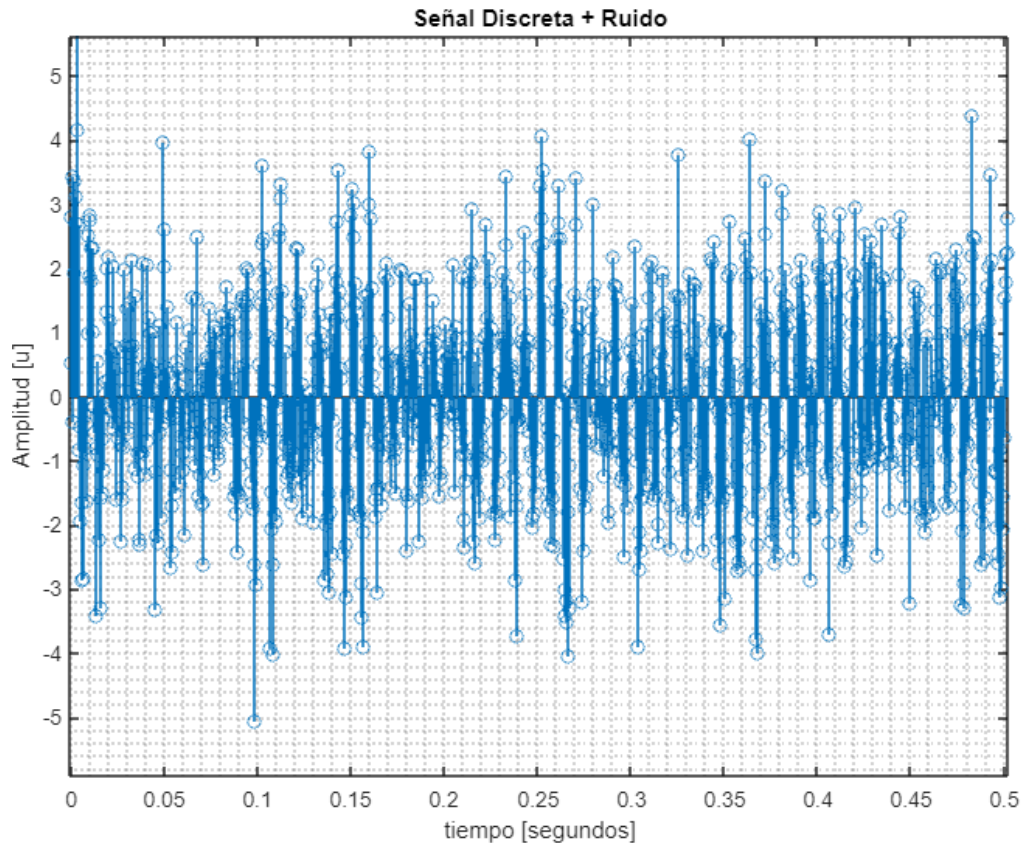


Figura 4. Señal discreta de 16 componentes más ruido

2.1.2 Uso de la función `mi_yule` y Yule-Walker de Matlab

Este código realiza una comparación entre la PSD utilizando dos métodos distintos: el método Yule-Walker personalizado y el método Yule-Walker implementado en MATLAB.

En la primera parte del código:

1. Se llama a la función '`mi_yule`' con los argumentos '`l`' y '`x`' para calcular los coeficientes del modelo AR y otros parámetros, donde `l` es el orden del método y `x` la señal discreta.
2. Se calcula la varianza del ruido en la señal utilizando la función '`pot_ruido`'.
3. Se realiza un bucle para calcular la estimación de la PSD para diferentes frecuencias desde 0 hasta la mitad de la frecuencia de muestreo ($F_s/2$), con un paso de 0.1.
4. En cada iteración del bucle, se utiliza la función '`mi_pds`' con los parámetros calculados previamente para obtener la PSD en la frecuencia actual (`f`) y se almacena en el vector '`pss`'.
5. Finalmente, se grafica la estimación de la PSD obtenida con el método Yule-Walker personalizado.

En la segunda parte del código:

1. Se utiliza la función 'aryule' de MATLAB para calcular los coeficientes del modelo AR y la varianza del ruido [24].
2. Se utiliza la función 'pyulear' de MATLAB para calcular la PSD utilizando el método Yule-Walker implementado en MATLAB.
3. Se grafica la estimación de la PSD obtenida con el método Yule-Walker de MATLAB.

En ambos gráficos, se muestra la estimación de la PSD para un rango de frecuencias específico y se comparan los resultados obtenidos con los dos métodos Yule-Walker. El eje x del gráfico representa la frecuencia en kHz y el eje y representa la PSD en decibelios por Hertz.

```
%CALCULO CON MI PROPIA FUNCION MI YULE
[RR,rr,parm]=mi_yule(l,x)
% Llama a la función 'mi_yule' con los argumentos 'l' y 'x'
sigma=pot_ruido(l,parm,x);% Calcula la varianza del ruido utilizando la
función 'pot_ruido'
aux=0;% Inicializa la variable 'aux'
disp('Carga computacional MI YULE')% Muestra un mensaje en la ventana de
comandos
tic
for f=0:0.1:Fs/2
    aux=aux+1;
    %CALCULO DE LA DENSIDAD ESPECTRAL DE POTENCIA DE MODELOS AR
    pss(aux)=mi_pds(f,sigma,parm,l,ts);%Uso de la funcion mi_pds
end
toc
f=0:0.1:Fs/2;
figure(3)
subplot(1,2,1)
plot(f,10*log10(pss));
axis([90 250 -75 10])
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
title('Estimacion de PSD (MI YULE)')
legend('PDS (mi yule)')
grid minor
%%funciones de matlab
disp('Carga computacional YULE WALKER MATLAB')
tic
%CALCULO CON LA FUNCION YULE DE MATLAB
[coef,variance] = aryule(x,l);
[Pxx,F] = pyulear(x,l,N,Fs);%nfft- Número de puntos DFT
toc
subplot(1,2,2)
plot(F,10*log10(Pxx));
axis([90 250 -75 10])
grid minor
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
```

```
title('Estimacion de PSD (MATLAB)')
legend('PDS (YULE WALKER MATLAB)')
```

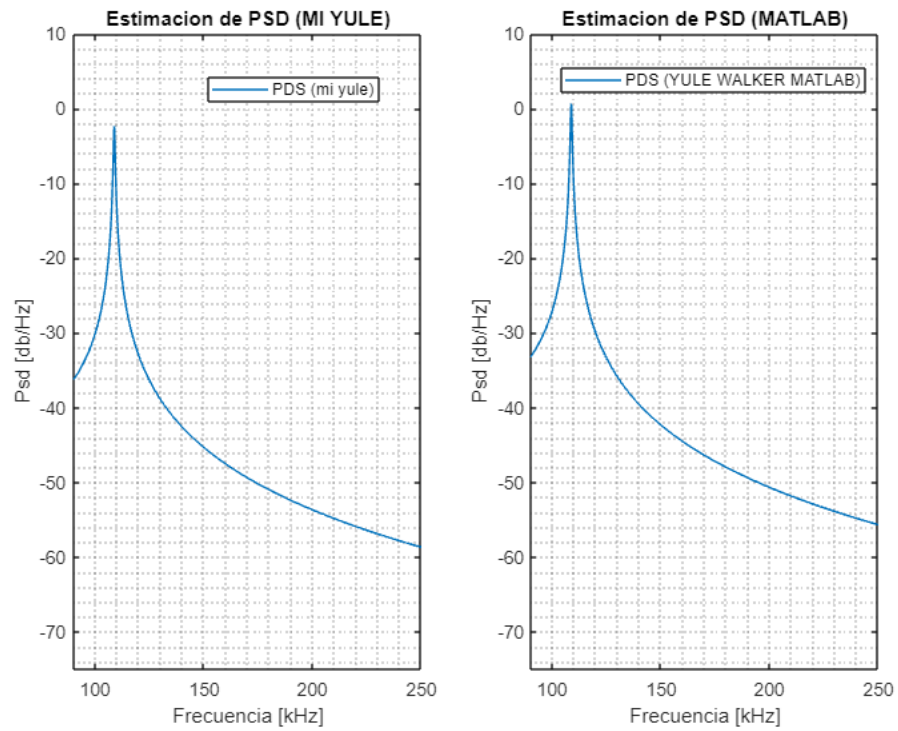


Figura 5. Comparación de 'mi_yule' vs yule-walker Matlab para un orden 2

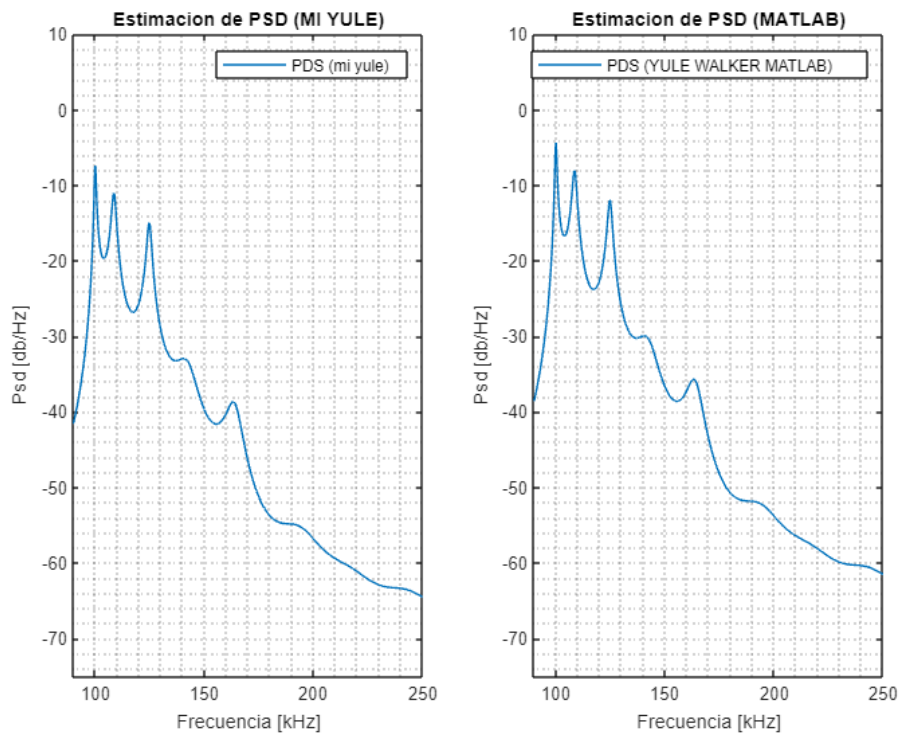


Figura 6. Comparación de 'mi_yule' vs yule-walker Matlab para un orden 100

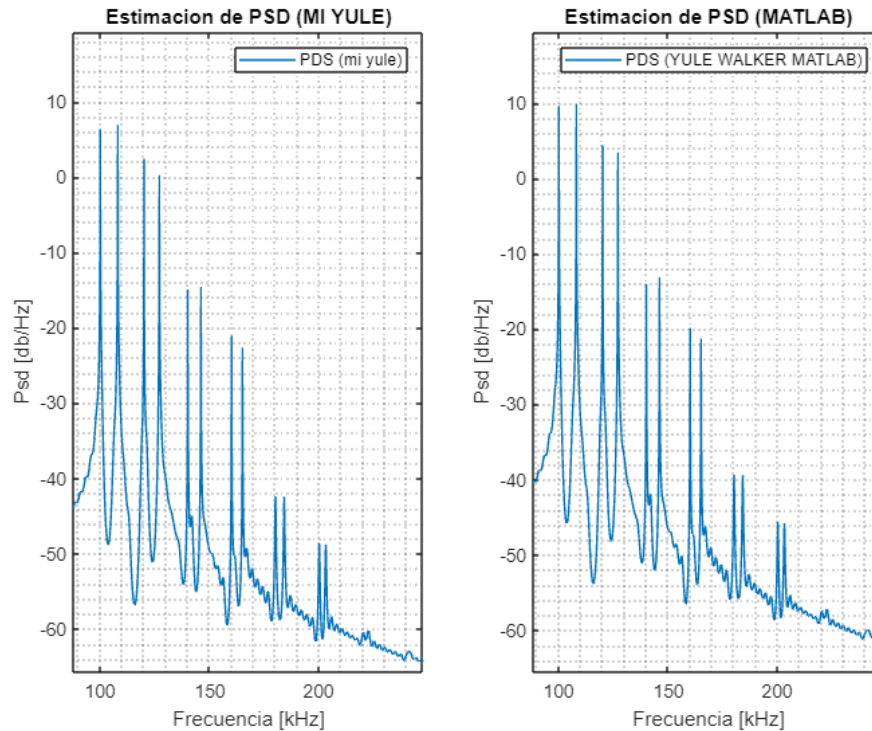


Figura 7. Comparación de mi_yule vs yule-walker Matlab para un orden 999

Se puede observar en la Figura 5, 6 y 7 que se realizó una comparación del método Yule-Walker personalizado y el método Yule-Walker implementado en MATLAB a tres órdenes diferentes: 2, 100 y 999 respectivamente. Los resultados revelaron que, en los casos de los órdenes 2 y 100, los métodos no fueron capaces de capturar todas las frecuencias presentes en las señales. Sin embargo, cuando se incrementó el orden a 999, ambos algoritmos lograron identificar y representar con precisión todas las frecuencias y amplitudes presentes en las señales analizadas.

Un detalle interesante surgió al comparar las funciones de representación. La función personalizada 'mi_yule' demostró un desplazamiento de 4 dB/Hz hacia abajo en el eje y, en comparación con el método Yule-Walker implementado en MATLAB. Esta diferencia en la representación visual sugiere una variación en la magnitud de las amplitudes espectrales entre ambos enfoques, lo cual es un aspecto importante por considerar al seleccionar el método más adecuado para análisis espectrales específicos.

2.1.3 Implementación del Periodograma Simple y Modificado

Este código implementa una comparación entre el periodograma simple y el periodograma modificado, así como la generación y aplicación de diferentes ventanas en la señal temporal antes de calcular el periodograma modificado.

Primero, se inicializa la variable $nFFT$ con un valor de 2 y luego se entra en un bucle `while` que multiplica $nFFT$ por 32 y 2 de manera iterativa hasta que $nFFT$ sea mayor o igual que N (donde N es la longitud de la señal x). Esto se realiza para encontrar un valor de $nFFT$ que sea una potencia de 2 y lo suficientemente grande para una estimación más precisa del espectro de frecuencia.

Luego, se calcula el periodograma simple de la señal x mediante la función `fft`. Se utiliza el valor $nFFT$ obtenido en el bucle y se almacena en la variable $P1$, que contiene el espectro de potencia de la señal.

A continuación, se generan tres ventanas diferentes: `rectwin`, `hanning` y `blackmanharris`, de igual longitud que la señal x . Estas ventanas se aplican a la señal x para obtener tres versiones enventanadas, almacenadas en las variables $y1_enventanada$, $y2_enventanada$ y $y3_enventanada$, respectivamente [25].

La utilización de la ventana `rectwin` muestra la estimación de la densidad espectral de potencia (PSD) del periodograma con una ventana rectangular. La ventana se especifica como un vector de fila o columna con la misma longitud que la señal de entrada. Si se deja la ventana especificada como vacía, el periodograma utiliza automáticamente una ventana rectangular [26].

La ventana `Hanning` tiene un efecto algo más intenso sobre los valores de la secuencia a los que se aplica en comparación con la ventana de `Hanning`. Al analizar la ecuación de la ventana, se mira una distinción clara en comparación con la de `Hanning`, y una mayor similitud con la de `Blackman`, la cual se examina en una etapa posterior [27].

`BlackmanHarris` es una ventana utilizada en el procesamiento de señales para mejorar la representación espectral y reducir los efectos de fugas espectrales en la Transformada de Fourier de una señal. Funciona aplicando una ponderación específica a los valores de una señal antes de realizar la Transformada de Fourier, con el objetivo de mejorar la representación espectral y minimizar la fuga espectral en el análisis de señales [28].

Luego, se calcula el periodograma modificado para cada una de las señales enventanadas, utilizando la función `fft` con el mismo valor $nFFT$ utilizado para el periodograma simple. Los resultados se almacenan en las variables $Pxx1$, $Pxx2$ y $Pxx3$, que contienen los espectros de potencia correspondientes a las señales enventanadas.

Finalmente, se superponen los resultados en un gráfico para comparar el periodograma simple (almacenado en $P1$) con los Periodogramas modificados de las señales

enventanadas (almacenados en Pxx1, Pxx2 y Pxx3). Se muestra cómo varían los espectros de potencia en función de las diferentes ventanas aplicadas a la señal original.

Es importante destacar que la función `fft` se utiliza para calcular la Transformada Rápida de Fourier, que es el cálculo principal para obtener los espectros de potencia en el análisis espectral de la señal. El código proporcionado realiza una comparación visual entre el periodograma simple y el periodograma modificado utilizando diferentes ventanas, lo que permite evaluar cómo afecta el enventanado al resultado espectral.

```
%PERIODOGRAMA SIMPLE
nFFT=2;
while nFFT<N
nFFT = 32*nFFT*2;
end
% PERIODOGRAMA SIMPLE:
P1=abs(fft(x,nFFT));
f1=linspace(0,Fs,nFFT);
toc
disp('Carga computacional Periodograma Modificado')
tic
% PERIODOGRAMA MODIFICADO:
% 1. GENERACION DE VENTANAS:
window_1=rectwin(length(x));
window_2=hanning(length(x));
window_3=blackmanharris(length(x));
% 2. Enventanado de la senial temporal:
y1_enventanada=x.*window_1';
y2_enventanada=x.*window_2';
y3_enventanada=x.*window_3';
% 3. Periodograma simple vs modificado:
Pxx1=abs(fft(y1_enventanada,nFFT));
Pxx2=abs(fft(y2_enventanada,nFFT));
Pxx3=abs(fft(y3_enventanada,nFFT));
%Superposición
figure(4)
hold on
plot(F,10*log10(Pxx))
plot(f1,10*log10(Pxx3),'r');
legend('Yule-Walker','Periodograma Modificado');
```

2.2 FUNCIÓN CÁLCULO DE AUTOCORRELACIÓN

Este código implementa una función llamada `'mi_rxx'` que calcula la autocorrelación de una señal discreta `'x'` hasta un cierto retraso `'K'`. Primero, se inicializa el valor de `'rxxi'` como cero. Luego, se verifica si el valor de `'K'` es negativo y, en caso afirmativo, se multiplica por `-1` para convertirlo en positivo. A continuación, se realiza un bucle para calcular la suma de los productos de los elementos de la señal desplazados `'K'` posiciones hacia la derecha y los elementos originales de la señal. Este cálculo se realiza para todos los desplazamientos posibles dentro del rango permitido. Finalmente, se normaliza la suma dividiéndola por la

longitud total de la señal 'N' para obtener el valor final de la autocorrelación 'rxxi'. Si el valor de 'K' es mayor que el rango permitido (N-1), se muestra un mensaje de error.

```
function [rxxi]=mi_rxx(x,K)
rxxi=0;
N=length(x);
if K<0
    K=K*-1;
end
if K<=N-1
    for n=0:N-K-1
        rxxi=rxxi+(x(n+K+1)*x(n+1));
    end
else
    fprintf('rango no permitido');
end
rxxi=rxxi/N;
```

```
>> rr
rr =
    -1.2122
    -1.0653
>> RR
>>
RR =
    1.2626    1.2122
    1.2122    1.2626
```

Figura 8. Obtención de la matriz de autocorrelación y su vector

2.3 FUNCIÓN VARIANZA O POTENCIA DEL RUIDO

Para calcular la varianza del ruido de una señal discreta x utilizando los coeficientes del modelo AR.

La función toma tres argumentos de entrada:

- N : El grado del modelo AR, es decir, el número de coeficientes AR utilizados en el modelo.
- ak : Una matriz que contiene los coeficientes AR, donde $ak(k + 1,1)$ representa el coeficiente correspondiente a k (desplazamiento temporal negativo).
- x : La señal discreta de entrada para la cual se desea calcular la varianza del ruido.

El código utiliza un bucle for para calcular la varianza del ruido (sigma) sumando los productos de los coeficientes AR (ak) y las funciones de autocorrelación de la señal en diferentes desplazamientos temporales (-k) desde -N hasta 0.

El resultado final de la función es la varianza del ruido (sigma), que se devuelve como resultado.

```
function [sigma]=pot_ruido(N,ak,x)%grado ,parametros, muestras
sigma=0;
for k=1:N
    sigma=sigma+ak(k+1,1)*mi_rxx(x,-k);
end
sigma=sigma+mi_rxx(x,0);
```

2.4 FUNCIÓN CALCULO DENSIDAD ESPECTRAL DE POTENCIA

Este código implementa una función llamada 'mi_pds' que calcula la PDS para una frecuencia 'f' específica, utilizando los coeficientes autorregresivos 'parm' de un modelo AR de orden 'l', y la varianza del ruido 'sigma'. Primero, se inicializan las variables 'A' y 'B' como 1 y 0 respectivamente. Luego, se realiza un bucle en el que se calcula la suma de los términos coseno y seno de la serie autorregresiva para la frecuencia 'f' y el tiempo 't'. Estos términos se utilizan para calcular los módulos al cuadrado 'A' y 'B'. Luego, se realiza la suma de estos módulos al cuadrado para obtener el módulo total. Finalmente, se calcula la PDS dividiendo la varianza del ruido 'sigma' por el módulo total y multiplicando por el intervalo de tiempo 't'. El resultado final se devuelve como 'pss', que representa la PSD para la frecuencia 'f'.

```
function [pss]=mi_pds(f,sigma,parm,l,ts)
%pds
%modulo al cuadrado , denominador de la pds
A=1;
B=0;
for i=1:l
    A=A+parm(i+1,1)*cos(2*pi*f*i*ts);
    B=B+parm(i+1,1)*sin(2*pi*f*i*ts);
end
A=A^2;
B=B^2;
modulo=A+B;
pss=((sigma)*ts)/modulo;
```

3 RESULTADOS

Este capítulo está destinado a la discusión de los resultados obtenidos con cada uno de los métodos tratados, haciendo énfasis resolución en frecuencia, estimación correcta de amplitudes, carga computacional y obtención de frecuencias y amplitudes frente a ruido AWGN.

3.1 RESULTADOS

3.1.1 Resultados en frecuencia y amplitud para diferentes ordenes

En el análisis comparativo de métodos paramétricos, se han utilizado el método AR basado en el Algoritmo de Yule-Walker, el periodograma simple y el periodograma modificado para estimar la densidad espectral de potencia (PSD) de una señal. Los resultados obtenidos al comparar la correcta estimación en frecuencia y amplitud se presentan en diferentes figuras que muestran la evolución de la estimación con distintos órdenes.

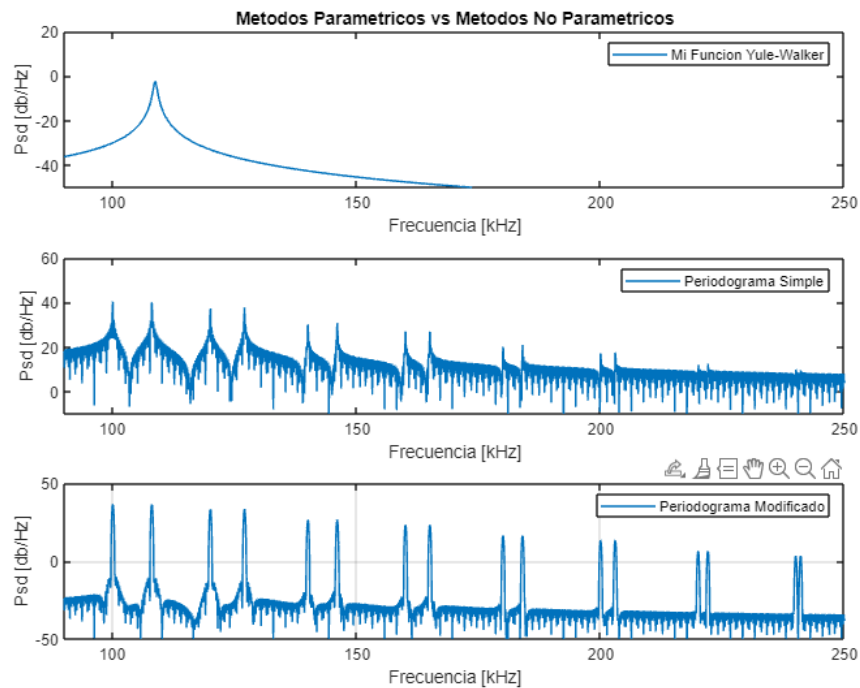


Figura 9. PSD para un método paramétrico de orden 2 frente a métodos no paramétricos

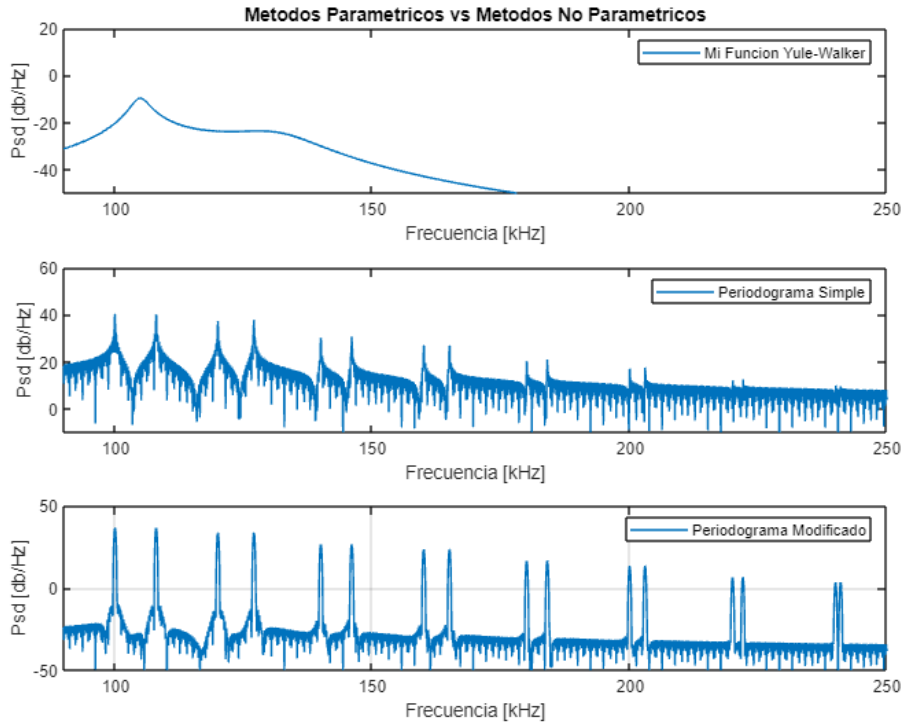


Figura 10. PSD para un método paramétrico de orden 10 frente a métodos no paramétricos

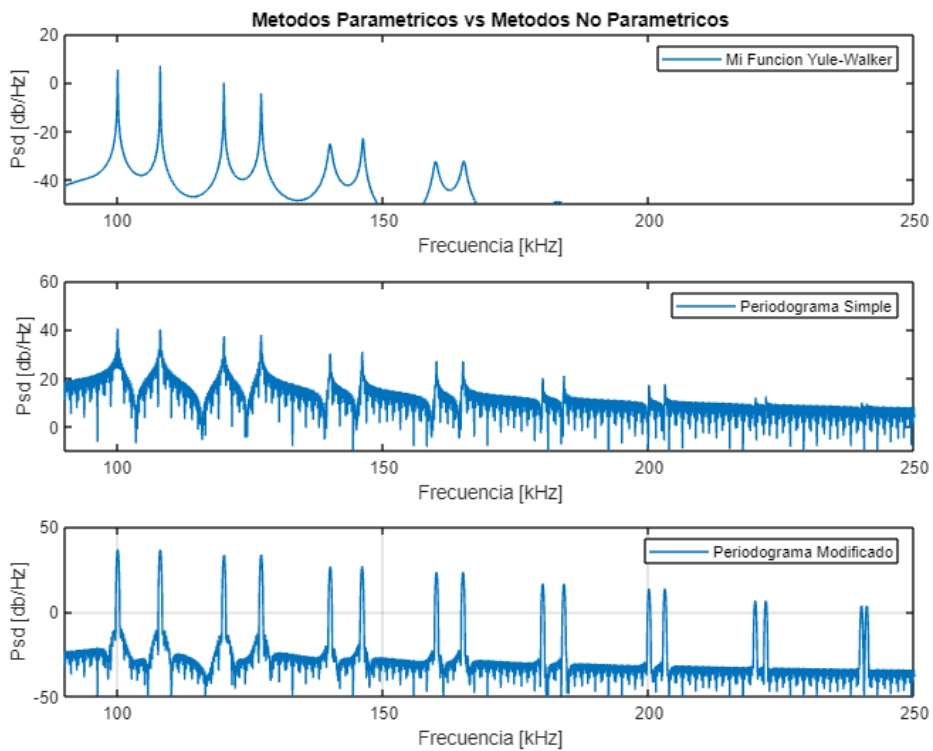


Figura 11. PSD para un método paramétrico de orden 100 frente a métodos no paramétricos

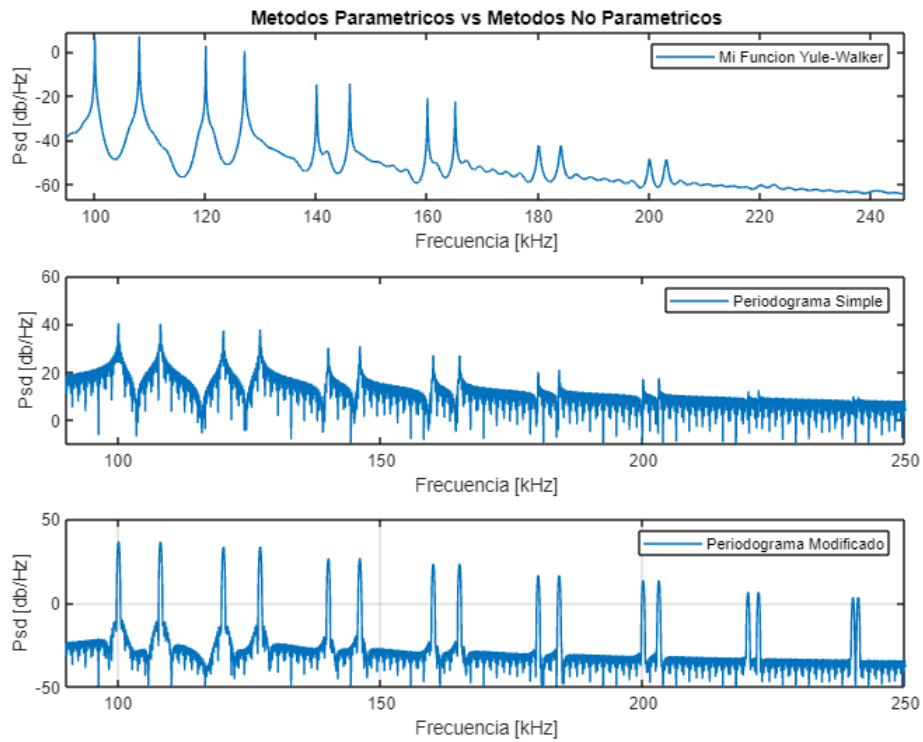


Figura 12. PSD para un método paramétrico de orden 999 frente a métodos no paramétricos

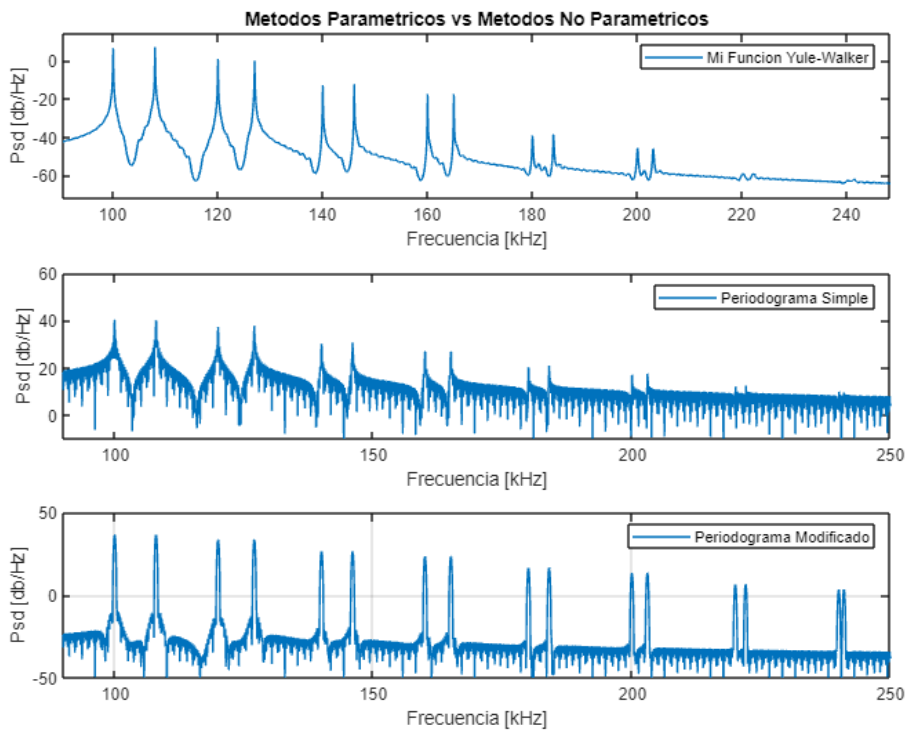


Figura 13. PSD para un método paramétrico de orden 2000 frente a métodos no paramétricos

En la Figura 9 se puede apreciar que el método AR con el Algoritmo de Yule-Walker no logra identificar ninguna de las frecuencias presentes en la señal, en contraste con el periodograma simple, que reconoce todas las frecuencias y amplitudes de la señal. Sin embargo, el periodograma modificado destaca por su capacidad para identificar adecuadamente todas las frecuencias presentes, aunque no logra estimar correctamente las amplitudes.

En un intento por mejorar la estimación, se aumenta el orden del modelo AR a 10, como se muestra en la Figura 10 A pesar del incremento en el orden, el método AR aún sigue sin lograr estimar correctamente las frecuencias y amplitudes de la señal.

No obstante, en la Figura 11 al aumentar el orden del modelo AR a 100, se observa un avance significativo en la estimación de las frecuencias y amplitudes. En esta figura, se aprecia cómo el método AR ha logrado estimar ambas características de 8 componentes de la señal, lo que sugiere que aumentar el orden puede mejorar la precisión de la PSD.

Se continúa experimentando con órdenes más altos, como 999 y 2000, como se muestra en las Figuras 12. y 13. respectivamente. En ambos casos, se identifican claramente las 16 frecuencias presentes en la señal y sus 8 amplitudes diferentes. Además, se observa que los lóbulos principales de la PSD son más evidentes y no están enmascarados por glóbulos secundarios, lo que ayuda a evitar distorsiones en la información.

En resumen, se ha demostrado que el método paramétrico AR basado en el Algoritmo de Yule-Walker puede lograr una estimación precisa de la PSD cuando se utiliza un orden suficientemente alto. A medida que se aumenta el orden del método, se mejora la capacidad para reconocer las características de la señal, pero también se incrementa la complejidad computacional y la cantidad de datos requeridos para obtener estimaciones confiables. Es importante encontrar un equilibrio entre el orden del modelo y la precisión deseada en la estimación para obtener resultados óptimos en el análisis de señales y series de tiempo.

3.1.2 Resultados de la prueba computacional para cada método

En este capítulo, se presentarán y analizarán los resultados obtenidos a través de la prueba computacional realizada para cada uno de los métodos de estimación espectral. Además, también se explorará la importancia de medir la carga computacional asociada a cada método, lo que permitirá comprender mejor su eficiencia en términos de tiempo y recursos.

En última instancia, esta evaluación exhaustiva contribuirá a una comprensión más completa de la aplicabilidad de estos métodos en el análisis de señales complejas.

Tabla 1. Carga computacional de los diferentes métodos de estimación espectral

ORDEN	MI YULE	YULE MATLAB	PERIODOGRAMA	PERIODOGRAMA MODIFICADO
2	0.008058	0.143116	0.000994	0.040652
10	0.014949	0.118895	0.001640	0.034073
100	0.046769	0.084581	0.000994	0.020981
999	0.413809	0.102108	0.001015	0.025699
2000	0.825222	0.104345	0.000997	0.024931

Según datos proporcionados sobre el tiempo de ejecución (carga computacional) de cada método a medida que el orden aumenta, se puede extraer las siguientes conclusiones:

1. Método Yule-Walker (MI YULE):

- A medida que aumenta el orden, la carga computacional también aumenta significativamente.
- El tiempo de ejecución aumenta de manera más notable a medida que el orden se incrementa considerablemente, como se evidencia en los órdenes 999 y 2000, donde el tiempo es considerablemente mayor que en órdenes más bajos.
- El método Yule-Walker implementado requiere más carga computacional en comparación con el método Yule-Walker en MATLAB, especialmente a órdenes altos.

2. Método Yule-Walker en MATLAB (YULE WALKER MATLAB):

- A medida que aumenta el orden, la carga computacional también aumenta, pero no de manera tan pronunciada como el método 'mi_yule'.
- El tiempo de ejecución sigue siendo razonable incluso para órdenes más altos, aunque también muestra un aumento con el orden.

3. Periodograma Simple:

- La carga computacional del periodograma simple es bastante baja y se mantiene estable a medida que aumenta el orden.
- Este método es computacionalmente más eficiente en comparación con los métodos Yule-Walker, especialmente a órdenes altos.

4. Periodograma Modificado:

- Similar al periodograma simple, la carga computacional del periodograma modificado es baja y no muestra un aumento significativo a medida que aumenta el orden.

- Al igual que el periodograma simple, este método es computacionalmente más eficiente en comparación con los métodos Yule-Walker.

En resumen, a medida que aumenta el orden, los métodos Yule-Walker, especialmente el implementado, experimentan un incremento notable en la carga computacional. Por otro lado, los Periodogramas simple y modificado no son afectados por la variación del orden por lo que se entiende que no varíen y muestren una carga computacional más baja y estable. Si la eficiencia computacional es una preocupación, los Periodogramas pueden ser una opción más adecuada, especialmente para órdenes más altos. Sin embargo, la elección del método también debe basarse en la precisión y confiabilidad de las estimaciones requeridas para el análisis específico que se esté realizando.

3.1.3 Resultados en frecuencia y amplitud para diferentes ordenes con ruido AWGN

En este capítulo, se examinan los resultados concernientes a la frecuencia y la amplitud, obtenidos para diferentes órdenes y en presencia de ruido AWGN. A través de un análisis minucioso, se lleva a cabo una comparativa exhaustiva de la resolución en frecuencia y la estimación de amplitudes empleando los modelos 'mi_yule', el periodograma simple y el periodograma modificado. Esta evaluación brinda una comprensión más profunda de cómo cada método se desempeña al identificar componentes frecuenciales y estimar amplitudes, especialmente en contextos donde las señales en la naturaleza están invariablemente acompañadas de interferencias de ruido.

Es esencial reconocer que, en la naturaleza, las señales rara vez se presentan de forma pura, y en su lugar, casi siempre están acompañadas por algún grado de ruido. Incluso cuando se logra una mejora en la fidelidad espectral mediante enfoques alternativos, resulta imperativo considerar los requerimientos computacionales inherentes a dichos métodos, ya que en muchas ocasiones estos pueden ser considerablemente más altos que los asociados con la Transformada Rápida de Fourier (FFT). Esta discusión subraya la continua relevancia de analizar señales en ambientes con presencia de ruido, lo cual es esencial para el avance del campo de la estimación espectral y su aplicación práctica.

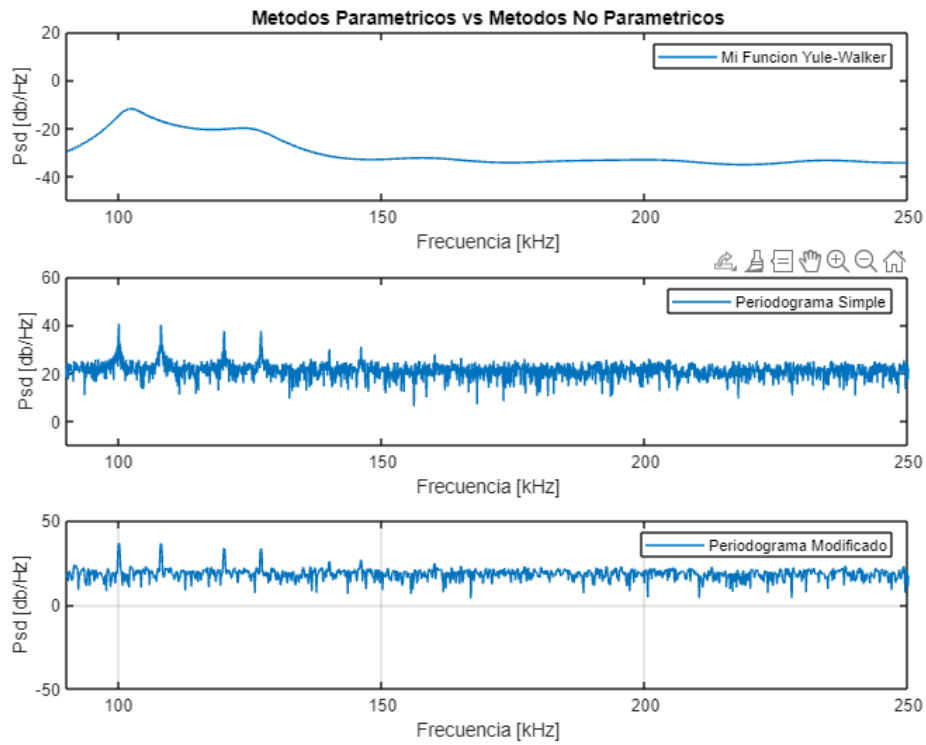


Figura 14. PSD para un método paramétrico de orden 100 frente a métodos no paramétricos con ruido AWGN

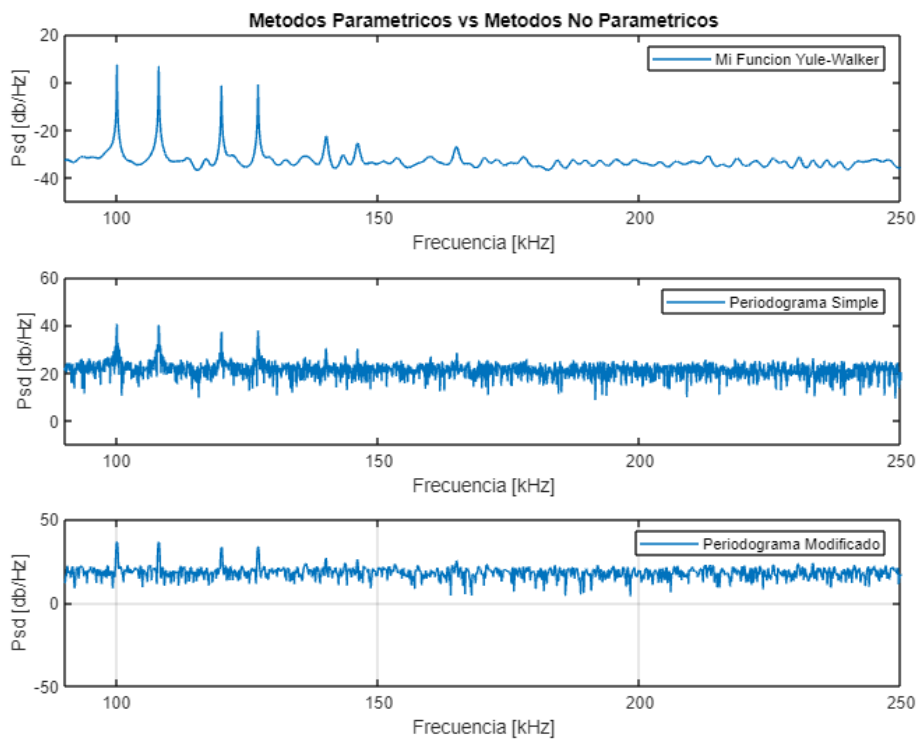


Figura 15. PSD para un método paramétrico de orden 999 frente a métodos no paramétricos con ruido AWGN

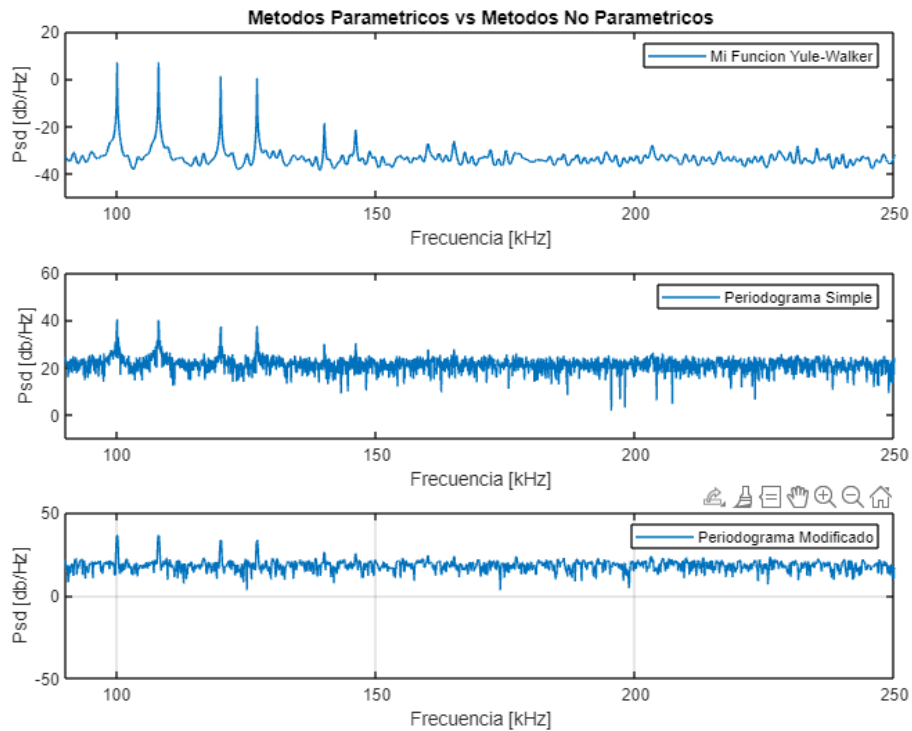


Figura 16. PSD para un método paramétrico de orden 2000 frente a métodos no paramétricos con ruido AWGN

En la Figura 14 se puede observar claramente que, para un orden de 100, el método 'mi_yule' no logra proporcionar resultados tanto en términos de frecuencia como de amplitud. Por el contrario, los Periodogramas, aunque no logran discernir todas las señales, muestran las seis primeras frecuencias con mayor amplitud.

En la Figura 15 al aumentar el orden a 999, se nota una mejora sustancial en el método Yule Walker en términos de resolución de frecuencia y estimación de amplitud. Esto se debe a que presenta menos ruido, permitiendo una representación más nítida en comparación con los periodogramas simples y modificados.

Finalmente, en la Figura 16 el método 'mi_yule' logra identificar más frecuencias y estimar sus amplitudes con mayor precisión. Aunque todavía no captura todas las frecuencias debido al deterioro de la señal por el ruido, su representación es más clara en comparación con los Periodogramas. Sin embargo, es importante mencionar que esta mejora en la calidad de la estimación viene acompañada de una carga computacional más elevada.

En resumen, a través de estas figuras es evidente cómo la elección del orden afecta la capacidad de cada método para discernir componentes frecuenciales y estimar amplitudes en presencia de ruido. La Figura 14 demuestra las limitaciones de 'mi_yule' en comparación con los Periodogramas para órdenes bajos. La Figura 15 destaca la ventaja del método

Yule Walker con un orden más alto, mientras que la Figura 16 muestra la mejora del 'mi_yule' a costa de una mayor carga computacional. Estos resultados ofrecen una comprensión más sólida de cómo estos métodos se comportan bajo diversas condiciones y la influencia del orden en su rendimiento.

En contraste, el periodograma simple y modificado exhiben limitaciones al no mejorar al variar el orden, como ocurre en el caso del método Yule-Walker, donde se aprecia una mejora en la visualización de las componentes en frecuencia y las variaciones de amplitud de la señal a medida que aumenta el orden.

En la Figura 16 se ha optado por un orden más alto para explorar si se mejora la estimación de la PSD. En los gráficos del periodograma simple y modificado, es difícil distinguir claramente las 6 frecuencias debido a la presencia de lóbulos secundarios y ruido en la PSD, que es más pronunciado en el periodograma simple. Por otro lado, el método Yule-Walker presenta una PSD con menos ruido, lo que permite una mejor visualización de las frecuencias y sus amplitudes.

4 CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

El método paramétrico AR basado en el Algoritmo de Yule-Walker muestra una mejora significativa en la estimación de la densidad espectral de potencia (PSD) a medida que se aumenta el orden del modelo. En comparación con la Figura 9 donde el método AR de orden 2 no logra identificar ninguna frecuencia, la Figura 13 con un orden de 2000 muestra una estimación más precisa de las frecuencias y amplitudes de la señal.

Los periodogramas simple y modificado presentan limitaciones en la estimación de la PSD, especialmente cuando hay una alta densidad de frecuencias y ruido en la señal. A pesar de no verse afectados por el orden del modelo paramétrico AR, los periodogramas muestran una mayor cantidad de lóbulos secundarios y ruido en comparación con el método AR de alto orden.

En cuanto a la carga computacional, el método Yule-Walker implementado muestra una mayor carga a medida que aumenta el orden en comparación con el Yule-Walker en MATLAB. En contraste, los periodogramas simple y modificado mantienen una carga computacional baja y estable independientemente del orden.

El análisis de la PSD revela que el método paramétrico AR puede ser una opción más adecuada cuando se busca una estimación más precisa y menos afectada por ruido y

lóbulo secundario. Sin embargo, se debe tener en cuenta la eficiencia computacional y la cantidad de datos disponibles al seleccionar el orden del modelo AR.

En general, el método paramétrico AR utilizando Yule-Walker muestra ventajas en términos de precisión en la estimación de la PSD, pero se debe sopesar con la carga computacional y el orden del modelo para obtener un equilibrio adecuado entre precisión y eficiencia. Los periodogramas, aunque más eficientes computacionalmente, pueden presentar desafíos en términos de precisión y robustez, especialmente en escenarios con alta densidad de frecuencias y ruido en la señal. La elección del método dependerá de las necesidades y requisitos específicos del análisis espectral que se esté realizando.

4.2 RECOMENDACIONES

En el proyecto que implementa el método paramétrico AR utilizando Yule-Walker en comparación con los métodos no paramétricos (periodograma simple y modificado), se recomienda realizar una investigación exhaustiva sobre los fundamentos teóricos y aplicaciones de cada método antes de la implementación. Además, evaluar la precisión, eficiencia y robustez de los métodos y documentar cada paso del proceso y comunicar claramente los hallazgos mediante gráficos y visualizaciones efectivas.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] W. H. Foy, "Comparison of methods for spectral estimation with interrupted data," Located in Los Gatos, CA, USA, Transactions on Signal Processing IEEE, March 1993, pp. 1449-1453.
- [2] S. L. H. Jr, "A Tutorial Overview of Modern Spectral Estimation," *Acoustics, Speech, and Signal Processing*, vol. 4, no. 23-26, pp. 2152 - 2157, May 1989.
- [3] MATLAB, "Autoregressive power spectral density estimate — Yule-Walker method," MathWorks América Latina, [Online]. Available: <https://la.mathworks.com/help/signal/ref/pyulear.html>. [Accessed 03 Agosto 2023].
- [4] S. Kay and S. J. Marple, "Spectrum analysis—A modern perspective," *Proceedings of the IEEE*, vol. 69, p. 1380 – 1419, Nov 1981.
- [5] A. M. a. R. Roberts, «A comparison of spectral estimators for real data,» de *Conference on Acoustics, Speech, and Signal Processing*, Atlanta, GA, USA, IEEE International Conference on Acoustics, 1981, pp. 492-495.
- [6] J. P. Alvarado Moya, "Procesamiento Digital de Señales," Tecnológico de Costa Rica, Escuela de Ingeniería Electrónica, Costa Rica, 2011.
- [7] M. d. Torres, "Los Reyes de la pasarela" Modelos Matemáticos en la Ciencias, vol. 1a ed, Buenos Aires: Fundación de Historia Natural Félix de Azara, 2015, p. 17.
- [8] I. S. Seroussi, «Spectral Analysis of a Non-Equilibrium Stochastic Dynamics on a General Network,» *Sci Rep* 8, 2018, p. 14333.
- [9] M.-Y. Chow, B. Ayhan, T. Joel and M.-H. Song, "A Case Study on the Comparison of Non-

- parametric Spectrum Methods for Broken," *IEEE Xplore*, p. 2835, December 2003.
- [10] R. Álvarez, "Capítulo 11" Estimación Espectral, Quito: Clase Procesamiento Digital de, 2021, p. 1.
- [11] G. G. D. L. B. Froyland, «Spectral analysis of climate dynamics with operator-theoretic approaches.,» *Nat Commun* 12, 2021, p. 6570 .
- [12] P. Gomis, Estimación Espectral De Señales Biomédicas Métodos Clásicos (Fft) Y, Castelldefels: Aplicaciones Prácticas Con Matlab, Universitat Politècnica de, 2013, p. 45.
- [13] R. Álvarez, "Capítulo 9" Periodograma Modificado, Quito: Clase Procesamiento Digital de Señales, 2021, p. 24.
- [14] M. Rios, Análisis espectral multivariable aplicado a señales cerebrales reales, Sevilla: Universidad De Sevilla Escuela Superior De Ingenieros, 2005, p. 32.
- [15] B. Najafi and H. Hossein, "A comparative study of non-parametric espectral estimators for application in machine vibration analysis," *Electrical and Computer Engineering Department, Worcester Polytechnic Institute,,* vol. 6, pp. 551-574, 1991.
- [16] A. P. a. G. Espinosa, Linear signal processing methods and decay ratio, Iztapalapa: Linear signal processing methods, 2019, p. 282.
- [17] B. Van Veen, Director, *Autoregressive Models: The Yule-Walker Equations*. [Film]. https://www.youtube.com/watch?v=PFyp4t16_xk: AllSignalProcessing.com, 2013.
- [18] M. Bayas, Modelos Autorregresivos de Análisis Espectral, Quito: Escuela Politecnica Nacional, 1984, p. 16.
- [19] R. Álvarez, "Capítulo 6" Dominio de la Frecuencia- Teoría de Fourier, Quito: Clase Procesamiento Digital de Señales, 2021, p. 21.
- [20] G. LOTHIAN, «Spectral Analysis,» *Nature*, pp. 234, 278, 1971.
- [21] A. Khinchin, Korrelationstheorie der stationären stochastischen Prozesse, Alemania: *Mathematische Annalen* 109, 1934, p. 604–615.
- [22] G. Eshel, "The Yule Walker Equations for the AR Coefficients," Wharton University of Pennsylvania, [Online]. Available: <https://statistics.wharton.upenn.edu/>. [Accessed 23 Agosto 2023].
- [23] MATLAB, "awgn-Add white Gaussian noise to signal," MathWorks, [Online]. Available: <https://la.mathworks.com/help/comm/ref/awgn.html>. [Accessed 01 08 2023].
- [24] MATLAB, "aryule-Autoregressive all-pole model parameters — Yule-Walker method," MathWorks, [Online]. Available: <https://www.mathworks.com/help/signal/ref/aryule.html>. [Accessed 01 08 2023].
- [25] G. Battista Rossi, F. Crenna, V. Piscopo and A. Scamardella, "Comparison of Spectrum Estimation Methods for the," *SENSORS*, pp. 4-10, 2020.
- [26] Mathworks, "Periodogram power spectral density estimate," The MathWorks, Inc., [Online]. Available: <https://www.mathworks.com/help/signal/ref/periodogram.html>. [Accessed 23 Agosto 2023].
- [27] M. Estevez, A. Machado y J. Estevez, «El periodograma como método para la estimación de la densidad espectral de potencia,» *Libros Abiertos*, 2008.
- [28] F. J. HARRIS, «On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform,» *PROCEEDINGS OF THE IEEE*, vol. 66, nº 1, p. 63, 1978.

6 ANEXOS

ANEXO 1: Código Principal

ANEXO 2: Función mi_rxx

ANEXO 3: Función mi_pds

ANEXO 4: Función mi_yule

ANEXO I

%ESCUELA POLITECNICA NACIONAL
%Michelle Chiluisa

```

clc%Limpia la ventana de comandos,
clear all%Borra todas las variables del espacio de trabajo
close all%Cierra todas las figuras existentes.
duracion = 10;%Duracion 10 segundos
fmax=241;%Frecuencia maxima a muestrear
Fs=10*fmax;%Frecuencia de muestreo
ts=1/Fs;%Tiempo de muestreo
t=0:ts:duracion;%Vector de tiempo

xn=1*sin(2*pi*100*t)+1*sin(2*pi*108*t)+...
0.5*sin(2*pi*120*t)+0.5*sin(2*pi*127*t)+...
0.1*sin(2*pi*140*t)+0.1*sin(2*pi*146*t)+...
0.05*sin(2*pi*160*t)+0.05*sin(2*pi*165*t)+...
0.01*sin(2*pi*180*t)+0.01*sin(2*pi*184*t)+...
0.005*sin(2*pi*200*t)+0.005*sin(2*pi*203*t)+...
0.001*sin(2*pi*220*t)+0.001*sin(2*pi*222*t)+...
0.0005*sin(2*pi*240*t)+0.0005*sin(2*pi*241*t);
%Señal con 8 diferentes amplitudes y 16 frecuencias diferentes

N=length(xn);% Calcula la longitud de la señal 'x'
awgn=randn(1,length(xn));% Genera ruido blanco gaussiano de longitud N
x=xn+awgn;% Suma la señal 'x' con el ruido 'awgn'
l=input('Ingrese el grado del modelo:');% Solicita al usuario que ingrese el
grado del modelo
figure(1)% Crea una nueva figura
stem(t,xn)% Grafica la señal 'x'
grid minor % Agrega una cuadrícula de fondo al gráfico
xlabel('tiempo [segundos]')% Establece una etiqueta en el eje x del gráfico
ylabel('Amplitud [u]')% Establece una etiqueta en el eje y del gráfico.
title('Señal Discreta ')% Establece un título para el gráfico
figure(2)% Crea una nueva figura
stem(t,x)% Grafica la señal 'x1' (señal original + ruido)
grid minor % Agrega una cuadrícula de fondo al gráfico
xlabel('tiempo [segundos]')% Establece una etiqueta en el eje x del gráfico
ylabel('Amplitud [u]')% Establece una etiqueta en el eje y del gráfico
title('Señal Discreta + Ruido')% Establece un título para el gráfico

%CALCULO CON MI PROPIA FUNCION MI YULE
[RR,rr,parm]=mi_yule(l,x)
% Llama a la función 'mi_yule' con los argumentos 'l' y 'x'
sigma=pot_ruido(l,parm,x);% Calcula la varianza del ruido utilizando la
función 'pot_ruido'
aux=0;% Inicializa la variable 'aux'
disp('Carga computacional MI YULE')% Muestra un mensaje en la ventana de
comandos
tic
for f=0:0.1:Fs/2
    aux=aux+1;
    %CALCULO DE LA DENSIDAD ESPECTRAL DE POTENCIA DE MODELOS AR
    pss(aux)=mi_pds(f,sigma,parm,l,ts);%Uso de la funcion mi_pds
end
toc
f=0:0.1:Fs/2;
figure(3)
subplot(1,2,1)
plot(f,10*log10(pss));
axis([90 250 -75 10])
xlabel('Frecuencia [kHz]')

```

```

ylabel('Psd [db/Hz]')
title('Estimacion de PSD (MI YULE)')
legend('PDS (mi yule)')
grid minor
%%funciones de matlab
disp('Carga computacional YULE WALKER MATLAB')
tic
%CALCULO CON LA FUNCION YULE DE MATLAB
[coef,variance] = aryule(x,1);
[Pxx,F] = pyulear(x,1,N,Fs);%nfft- Número de puntos DFT
toc
subplot(1,2,2)
plot(F,10*log10(Pxx));
axis([90 250 -75 10])
grid minor
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
title('Estimacion de PSD (MATLAB)')
legend('PDS (YULE WALKER MATLAB)')
disp('Carga computacional Periodograma Simple')
%-----
tic
%PERIODOGRAMA SIMPLE
nFFT=2^(ceil(log2(N)))*1;
%Aplico la función FFT de matlab
Xn1=fft(x,nFFT);
% Se calcula el módulo de los complejos llamado PERIODOGRAMA
Periodograma=abs(Xn1);
%Cálculo del EJE de frecuencias: misma cantidad de puntos que PERIODOGRAMA y
que están entre 0 y Fs:
f1=linspace(0,Fs,nFFT);
toc
disp('Carga computacional Periodograma Modificado')
tic
% PERIODOGRAMA MODIFICADO:
% 1. GENERACION DE VENTANAS:
window_1=rectwin(length(x));
window_2=hanning(length(x));
window_3=blackmanharris(length(x));
% 2. Eventanado de la senial temporal:
y1_eventanada=x.*window_1';
y2_eventanada=x.*window_2';
y3_eventanada=x.*window_3';
% 3. Periodograma simple vs modificado:
Pxx1=abs(fft(y1_eventanada,nFFT));
Pxx2=abs(fft(y2_eventanada,nFFT));
Pxx3=abs(fft(y3_eventanada,nFFT));
toc
%Superposición
figure(4)
hold on
subplot(3,1,1)
plot(F,10*log10(pss))
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
title('Metodos Parametricos vs Metodos No Parametricos')
legend('Mi Funcion Yule-Walker');
axis([90 250 -50 20])

```



```
subplot(3,1,2)
plot(f1,10*log10(Periodograma));
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
legend('Periodograma Simple');
axis([90 250 -10 60])
subplot(3,1,3)
plot(f1,10*log10(Pxx3));
xlabel('Frecuencia [kHz]')
ylabel('Psd [db/Hz]')
legend('Periodograma Modificado');
axis([90 250 -50 50])
grid on
```

ANEXO II

```
function [rxxi]=mi_rxx(x,K)
rxxi=0;
```

```
N=length(x);  
if K<0  
    K=K*-1;  
end  
if K<=N-1  
    for n=0:N-K-1  
        rxxi=rxxi+(x(n+K+1)*x(n+1));  
    end  
else  
    fprintf('rango no permitido');  
end  
rxxi=rxxi/N;
```

ANEXO III

```
function[pss]=mi_pds(f,sigma,param,l,t)
```

```

%pds
%modulo al cuadrado , denominador de la pds
A=1;
B=0;
    for i=1:l
        A=A+parm(i+1,1)*cos(2*pi*f*i*t);
        B=B+parm(i+1,1)*sin(2*pi*f*i*t);
    end
A=A^2;
B=B^2;
modulo=A+B;
pss=((sigma)*t)/modulo;

```

ANEXO IV

```

function [R,r,aux]=mi_yule(l,x)%lgrado de la autoregresion

```

```

N=1;
for k1=1:N
    for k2=1:length(0:-1:1-N)
        %fprintf('rxx%d ',k1-k2);
        R(k1,k2)=mi_rxx(x,k1-k2);
    end
    fprintf('\n');
end
%para k=0 en las ec de yule walker(ladp derecho de las ec)
for k3=1:N
    %fprintf('-rxx%d \n',k3);
    r(k3,1)=-mi_rxx(x,k3);
end
%parametros del modelo
ar=(R^-1)*r;%inversa
aux=[1;ar(:,1)];

```