

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

SIMULACIÓN DE ESTIMACIÓN DE CANAL Y SINCRONIZACIÓN EN LA TÉCNICA DE COMUNICACIÓN INALÁMBRICA SEFDM SIMULACIÓN DE ESTIMACIÓN CFO DENTRO DE UN SISTEMA SEFDM

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

JONATHAN JOAQUÍN GUERRERO MORALES

jonathan.guerrero02@epn.edu.ec

DIRECTOR: DIEGO JAVIER REINOSO CHISAGUANO

diego.reinoso@epn.edu.ec

DMQ, octubre 2022

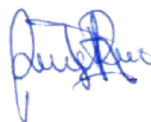
CERTIFICACIONES

Yo, Jonathan Joaquín Guerrero Morales 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.



Jonathan Joaquín Guerrero Morales

Certifico que el presente trabajo de integración curricular fue desarrollado por Jonathan Joaquín Guerrero Morales, bajo mi supervisión.



Diego Javier Reinoso Chisaguano
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.

Jonathan Joaquín Guerrero Morales

Diego Javier Reinoso Chisaguano

DEDICATORIA

Dedico mi proyecto de titulación a Dios, a mis padres, a mi hermana, a mi familia y amigos, a la memoria de mis abuelitos, Joaquín, Angelita, Luis y a mi perrita pelusa. Sin su apoyo, en este largo y duro camino en la universidad, no hubiera logrado ser el hombre que soy hoy en día.

AGRADECIMIENTO

A Dios, por brindarme fuerza y fortaleza en los momentos donde no podía seguir adelante.

A mis padres, Wilson y Yolanda, a mi pequeña hermana Wendy, a mis perritos Cuqui y Body porque solo nosotros sabemos lo que sufrimos y perdimos en estos últimos años. Sin su amor, apoyo y guía, no hubiese logrado levantarme y salir adelante. Todo lo que he logrado y lograré es para ustedes.

A mi familia, mi abuelita Angelita, tíos y tías, primos y primas, supieron brindarnos su apoyo cuando no pasábamos por un buen momento y nosotros más los necesitábamos. Nunca olvidaré que estuvieron ahí para ayudarnos, estaré eternamente agradecido.

A mis profesores, a cada uno de los que me formaron desde mi primer día en la Escuela Politécnica Nacional. Además, un especial agradecimiento a mi tutor de tesis, al Dr. Diego Reinoso porque sin sus consejos, exigencia, perfección y guía, no hubiese logrado realizar mi proyecto de titulación.

A mis amigos y amigas, a los que siguen en la universidad y los que no, con los que sufrimos en nivelación y con los que nos reuníamos varias veces entre edificios, porque con cada uno de ustedes viví grandes experiencias en este largo camino, me ayudaron a crecer y espero haberlos ayudado en lo mismo.

A un gran amigo, mi hermano, Alex, gracias por ser ese compañero de apoyo en las buenas y en las malas, pasamos un sinfín de anécdotas dentro y fuera de la U, formamos una gran amistad como estudiantes y estoy seguro que lo seguiremos siendo como profesionales.

A una increíble persona que conocí en la universidad, Dayanara, en mis momentos difíciles me escuchaste y apoyaste cuando más lo necesitaba, vivimos buenos y malos momentos en nuestro tiempo dentro de la universidad, estuviste para mí y yo estuve para ti. Gracias por haber hecho de la universidad aún un lugar más especial.

Gracias y un abrazo a todos.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VII
ABSTRACT	VIII
1. INTRODUCCIÓN.....	1
1.1. OBJETIVO GENERAL	2
1.2. OBJETIVOS ESPECÍFICOS	2
1.3. ALCANCE	2
1.4. MARCO TEÓRICO.....	3
1.4.1. SISTEMA SEFDM	3
1.4.1.1. Etapa de transmisión y recepción del sistema SEFDM.....	5
1.4.1.2. Ventajas y desventajas del sistema SEFDM.....	8
1.4.1.3. Comparación con OFDM.....	8
1.4.2. CARRIER FREQUENCY OFFSET	9
1.4.2.1. Técnicas de estimación.....	10
1.4.2.2. Aplicación en SEFDM	12
2. METODOLOGÍA.....	13
2.1. SCRIPT PRINCIPAL	13
2.2. FUNCION SEFDM_AWGN	16
2.2.1. VALORES INICIALES	16
2.2.2. OBTENCIÓN DE PREÁMBULO.....	18
2.2.3. TRANSMISOR SEFDM	19
2.2.3.1. Generación de bits y codificación FEC	19
2.2.3.2. Modulación digital	20
2.2.3.3. Formación de símbolos SEFDM y adición de ceros.....	20
2.2.3.4. IFFT	21
2.2.3.5. Adición de prefijo cíclico.....	22
2.2.3.6. Adición del preámbulo.....	22
2.2.4. CANAL AWGN.....	23

2.2.5.	RECEPTOR SEFDM	23
2.2.5.1.	Inserción CFO	24
2.2.5.2.	Estimación CFO	24
2.2.5.3.	Compensación CFO.....	28
2.2.5.4.	Eliminación del preámbulo	28
2.2.5.5.	Eliminación prefijo cíclico	29
2.2.5.6.	Adición de ceros y bloque FFT.....	29
2.2.5.7.	Detector MMSE.....	29
2.2.5.8.	Demodulación digital y decodificación FEC	31
2.2.6.	CÁLCULO DE BITS ERRADOS	31
2.3.	GRÁFICAS DE RESULTADOS EN SCRIPT PRINCIPAL.....	32
3.	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	34
3.1.	ESCENARIOS DE SIMULACIÓN	34
3.1.1.	PRUEBAS DE SIMULACIÓN EN LAS 3 TÉCNICAS DE ESTIMACIÓN CON VARIOS VALORES DE α Y UN VALOR DE CFO.....	34
3.1.2.	PRUEBAS DE SIMULACIÓN EN LAS 3 TÉCNICAS DE ESTIMACIÓN CON UN VALOR DE α Y CFO.....	36
3.1.3.	PRUEBAS DE SIMULACIÓN EN LAS 2 TÉCNICAS DE ESTIMACIÓN CFO CON VARIOS VALORES DE α Y UN VALOR DE CFO.	37
3.1.4.	PRUEBAS DE SIMULACIÓN EN LAS 2 TÉCNICAS DE ESTIMACIÓN CFO CON UN VALOR DE α Y DISTINTOS VALORES DE CFO.	39
3.2.	CONCLUSIONES.....	40
3.3.	RECOMENDACIONES	41
4.	REFERENCIAS BIBLIOGRÁFICAS	43
5.	ANEXOS.....	45
	ANEXO I.....	45

RESUMEN

El presente trabajo de titulación realiza un análisis y comparación de la eficiencia entre 2 técnicas de estimación del CFO (*Carrier Frequency Offset*), con prefijo cíclico y preámbulo, y una estimación perfecta dentro de un sistema SEFDM (*Spectrally Efficient Frequency Division Multiplexing*). Para el análisis, se procede a simular, en el software de simulación Matlab, el sistema SEFDM a través de un canal AWGN (*Additive White Gaussian noise*) con las técnicas de estimación CFO en el dominio del tiempo y ocupando en recepción un detector MMSE (*Minimum Mean Square Error*). Los resultados de la simulación son presentados en curvas dentro de una gráfica BER (*Bit Error Rate*) vs E_b/N_0 (*Energy per Bit to Noise Power Spectral Density*) con diferentes escenarios de simulación donde, se van variando parámetros como: valor de CFO, tipo de modulación, técnica de estimación, valor de factor de compresión de ancho de banda, etc.

El capítulo 1, muestra información acerca del sistema SEFDM, sus características principales, ventajas y desventajas dentro de una comparación con el sistema OFDM (*Orthogonal Frequency Division Multiplexing*). Además, se detalla los componentes que conforman las etapas de transmisión y recepción del sistema SEFDM

El capítulo 2, presenta una descripción acerca del código desarrollado en Matlab para la simulación del sistema SEFDM. Se describe cada función ocupada para la realización del script principal y el código usado para presentar las curvas en las gráficas BER vs E_b/N_0 .

El capítulo 3, describe los resultados obtenidos dentro de cada escenario de simulación y una breve comparación entre las curvas de las técnicas de estimación CFO y la curva de la estimación perfecta obtenidas en las gráficas BER vs E_b/N_0 . Además, las conclusiones y recomendaciones obtenidas del desarrollo del presente proyecto, así como, una idea para el desarrollo de un futuro proyecto de titulación.

PALABRAS CLAVE: SEFDM, CFO, Estimación, CP, Preámbulo.

ABSTRACT

This degree project performs an analysis and comparison of the efficiency between two CFO (*Carrier Frequency Offset*) estimation techniques, with cyclic prefix and preamble, and a perfect estimation within a SEFDM (*Spectrally Efficient Frequency Division Multiplexing*) system. For the analysis, the SEFDM system is simulated, in the Matlab simulation software, through an AWGN channel (*Additive White Gaussian noise*) with the CFO estimation techniques in the time domain and using an MMSE detector in reception (*Minimum Mean Square Error*). The simulation results are presented in curves within a BER (*Bit Error Rate*) vs. E_b/N_0 (*Energy per Bit to Noise Power Spectral Density*) graph with different simulation scenarios and varying parameters such as: CFO value, modulation type, estimation technique, bandwidth compression factor value, etc.

Chapter 1 shows information about the SEFDM system, its main characteristics, advantages and disadvantages in comparison with the OFDM (*Orthogonal Frequency Division Multiplexing*) system. In addition, the components that compose SEFDM system transmission and reception stage are described.

Chapter 2 presents a description of the code developed in Matlab for the simulation of the SEFDM system. Each function used to develop the main script and the code used to present the curves in the BER vs E_b/N_0 graphs are described.

Chapter 3 describes the results obtained within each simulation scenario and a brief comparison between the CFO estimation techniques curves and the perfect estimate curve obtained in the BER vs E_b/N_0 graphs. In addition, the conclusions and recommendations obtained from the development of this project, as well as an idea for the development of a future degree project.

KEYWORDS: SEFDM, CFO, Estimation, CP, Preamble.

1. INTRODUCCIÓN

Con el avance tecnológico de las comunicaciones inalámbricas se busca utilizar sistemas inalámbricos de próxima generación que puedan funcionar con las nuevas tecnologías, como es 5G. Como técnica prometedora para estas tecnologías, implementada ya en sistemas ópticos e inalámbricos, se presenta la técnica de modulación multiportadora SEFDM (*Spectrally Efficient Frequency Division Multiplexing*) que se diferencia de otras técnicas, ya que, utiliza subportadoras no ortogonales con el propósito de aumentar la eficiencia espectral del sistema [1].

Esta técnica, considerada una extensión de OFDM (*Orthogonal Frequency Division Multiplexing*) con espaciamiento de frecuencia no ortogonal, basa su funcionamiento en utilizar un menor ancho de banda en comparación con la técnica OFDM para transmitir la misma tasa de datos [2]. La reducción de ancho de banda está relacionada con un parámetro llamado factor de compresión α , cuyo valor en señales OFDM es $\alpha = 1$ y en señales SEFDM es $\alpha < 1$. Sin embargo, la desventaja que presenta esta reducción en la ocupación de ancho de banda es una degradación en el BER (*Bit Error Rate*) por la presencia de ICI (*Intercarrier Interference*) debido a la interferencia de las subportadoras no ortogonales [3].

Al igual que OFDM, un sistema SEFDM presenta sensibilidad frente a sincronización en tiempo (STO- *Symbol Time Offset*) y frecuencia (CFO- *Carrier Frequency Offset*). CFO se presenta cuando la señal modulada se distorsiona en el receptor debido a un desajuste de frecuencia, este desajuste corresponde a la diferencia de la frecuencia portadora entre el oscilador del receptor y el oscilador del emisor [4], este desfase en frecuencia produce la degradación dentro del sistema SEFDM. La estimación de CFO puede ser realizada, tanto en el dominio de tiempo como en frecuencia mediante el empleo de prefijo cíclico, secuencia de entrenamiento y la comparación del cambio de fase en el símbolo por consecuencia del desplazamiento de frecuencia en la portadora [5].

La solución para disminuir la degradación es la utilización de técnicas de generación, técnicas de recuperación y detección dentro de los sistemas SEFDM, por ejemplo, el detector ML (*Maximum Likelihood*) cuya complejidad aumenta exponencialmente a medida que el número de subportadoras se incrementa [6] y el detector lineal MMSE (*Minimum Mean Square Error*) que presenta una tasa de error inferior a los detectores ML [7].

1.1. OBJETIVO GENERAL

Simular en Matlab la estimación y compensación del CFO dentro de un sistema SEFDM.

1.2. OBJETIVOS ESPECÍFICOS

- Diseñar el diagrama de bloques de la simulación desarrollada para el sistema SEFDM.
- Describir las técnicas de estimación CFO dentro del sistema SEFDM para su implementación.
- Implementar la simulación del sistema SEFDM con estimación CFO utilizando un código convolucional de tasa $\frac{1}{2}$ y canal con ruido AWGN (*Additive White Gaussian noise*).
- Analizar los resultados obtenidos en las pruebas de funcionamiento variando los parámetros de modulación, factor de compresión y CFO.

1.3. ALCANCE

Este componente implementa una simulación en Matlab de un sistema SEFDM con estimación y compensación de CFO. El transmisor utiliza código convolucional de tasa $\frac{1}{2}$, diferentes técnicas de modulación como QPSK (*Quadrature Phase-Shift Keying*), 16QAM (*Quadrature Amplitude Modulation*) o 64QAM, un factor variable de compresión y canal con ruido AWGN. En la etapa de recepción se implementa la estimación CFO y la técnica de detección MMSE. Para realizar la comparación entre los resultados obtenidos de la simulación con los resultados dentro de un escenario con sincronización perfecta se variará los parámetros de modulación, factor de compresión y CFO. El desarrollo de este componente se trabaja con 3 fases:

Fase de diseño

En base a la información recolectada en un análisis bibliográfico previo, se realizará el diagrama de bloque tanto para la etapa de transmisión y recepción detallando cada componente de la simulación presente en estas etapas.

Fase de implementación

En esta fase se implementa la simulación en Matlab del sistema SEFDM con los procesos descritos en el diagrama de bloques, tanto para transmisor y el empleo de preámbulo en la

sincronización para poder realizar la estimación CFO en receptor junto con la técnica de detección MMSE.

Fase de prueba

Se realizará la simulación del componente variando parámetros de modulación y factor de compresión con el fin de obtener datos que puedan ser comparados en una simulación donde el escenario sea ideal. El resultado de esta comparación permite observar cómo afecta los parámetros usados en la simulación del sistema SEFDM con respecto a la estimación CFO.

1.4. MARCO TEÓRICO

En esta sección se describe el sistema SEFDM con sus características principales, ventajas, desventajas, descripción de las etapas de transmisión y recepción con los respectivos bloques que los conforman. Además, se presenta una comparación con el sistema OFDM para mostrar aspectos que los diferencian en su funcionamiento. Finalmente, se presenta información sobre la componente CFO, sus técnicas de estimación tanto en dominio de tiempo como de frecuencia y su aplicación en SEFDM.

1.4.1. SISTEMA SEFDM

El sistema SEFDM es la técnica de modulación multiportadora con mayor proyección para su uso en redes de última generación [3], su funcionamiento se caracteriza en la reducción de la utilización del ancho de banda debido a la no ortogonalidad que presentan sus subportadoras. Al no poseer ortogonalidad, el pico de potencia de una subportadora no coincide con el lugar donde la potencia de la subportadora adyacente es cero [8], caso contrario de lo que sucede en sistema OFDM donde sus subportadoras presentan ortogonalidad. En la Figura 1.1, se observa el solapamiento en las subportadoras del sistema SEFDM como consecuencia de la no ortogonalidad y de la reducción del ancho de banda del sistema ocasionando así la presencia de ICI en las subportadoras. El espaciado entre subportadoras adyacentes en SEFDM viene dado por [2]:

$$\Delta f = \frac{\alpha}{T} \quad (1.1)$$

donde, interviene un parámetro que es característico del sistema SEFDM denominado factor de compresión de ancho de banda α . Este factor consiste en un número racional que determina el ahorro de ancho de banda utilizado para transmitir la misma cantidad de datos en comparación con OFDM. La obtención del ahorro de ancho de banda es calculada con $(1 - \alpha) \times 100\%$ [6].

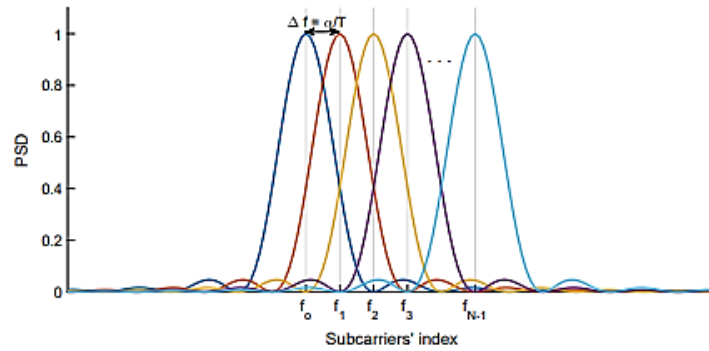


Figura 1.1. Sobrelapamiento de subportadoras en sistema SEFDM [7].

La relación que tiene el factor de compresión con el ahorro de ancho de banda se puede observar en la Figura 1.2, donde se muestra el espectro de potencia de un sistema SEFDM con subportadoras igual a $N = 1024$. Al tener un valor menor de factor de compresión, la compresión será mayor en el sistema SEFDM y por consiguiente también el ahorro de ancho de banda.

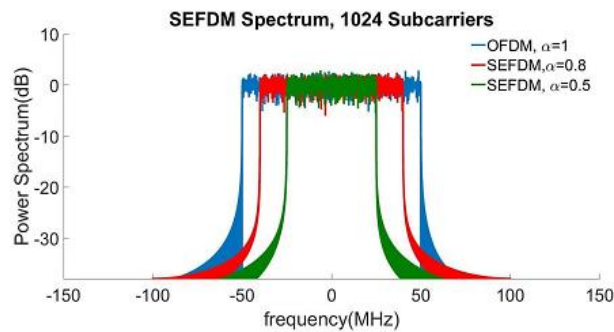


Figura 1.2. Espectro de potencia SEFDM con diferentes valores de α [6].

La reducción del ancho de banda trae consigo beneficios, como el aumento en la eficiencia espectral y la capacidad del sistema. Sin embargo, la desventaja que presenta es la degradación del BER debido al alto nivel de interferencia en sus subportadoras provocando un problema en la recepción de símbolos SEFDM. La solución para evitar esta degradación viene siendo investigada desde el año 2003, año en el que se desarrolló la técnica de SEFDM [1], [6], donde se ha probado diferentes métodos como detectores lineales MMSE o enfoques en detectores ML sin importar la complejidad computacional que estos puedan producir [3]. Un símbolo SEFDM consiste en un bloque de N símbolos complejos que son denotados como $s = s_{\Re} + js_{\Im}$ [1], cada uno de estos símbolos complejos son modulados en una subportadora. En forma discreta, al ocupar el primer símbolo SEFDM y considerando un periodo de muestreo de $t_s = \frac{T}{Q}$, donde $Q = \rho N$ y $\rho \geq 1$ es el factor de sobremuestreo, se tiene la siguiente fórmula del símbolo SEFDM [1]:

$$X_{sef\text{dm}}[k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} s_n \exp\left(\frac{j2\pi n\alpha k}{N}\right) \quad (1.2)$$

donde, los parámetros son [1]:

- α es el factor de compresión del ancho de banda.
- N es el número de subportadoras.
- T es el periodo de un símbolo SEFDM.
- s_n es el símbolo modulado QAM en la n^{th} subportadora.
- $k = [0, 1, \dots, N - 1]$ corresponde al índice de las muestras de tiempo en un símbolo SEFDM [7]

1.4.1.1. Etapa de transmisión y recepción del sistema SEFDM

El sistema SEFDM posee etapas de transmisión y recepción, cada una con bloques que cumplen una función determinada como se muestra en la Figura 1.3.

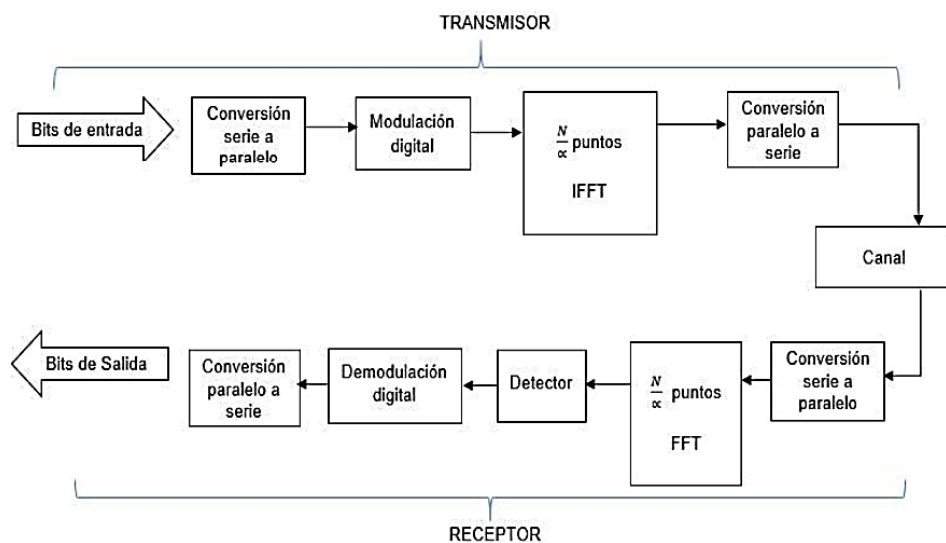


Figura 1.3. Transmisor y receptor de un sistema SEFDM [9].

a) Conversión serie a paralelo/ paralelo a serie

SEFDM es una técnica multiportadora que transmite flujo de datos o bits en las subportadoras [1]. En primer lugar, los bits son enviados a un bloque de conversión serie a paralelo (S/P) antes de ser modulados. Este flujo de bits en paralelo es convertido nuevamente a serie antes de ingresar al canal como se observa en la Figura 1.3. En la etapa de recepción se repite de igual manera el proceso de conversión para que al final se obtenga el flujo de bits en serie después de la demodulación.

b) Modulación/ demodulación digital

La modulación en SEFDM se basa en el proceso IFFT (*Inverse Fast Fourier Transform*), donde se transforma los flujos de datos convertidos a paralelo después de la salida del conversor en L muestras en el dominio del tiempo de un símbolo SEFDM [2]. Con esta transformación, se asigna un símbolo de modulación complejo a cada conjunto de símbolos binarios (0 o 1) [10], estos son organizados en grupos de D bits ocupando diferentes técnicas de modulación con un tamaño de $M = 2^D$ [2].

Por lo general, la señal SEFDM consta de un flujo de símbolos SEFDM con la presencia de N símbolos complejos generados por la utilización de la técnica de modulación QAM en el modulador [11]. La demodulación es el proceso inverso a la modulación que en el sistema SEFDM consta del bloque FFT (*Fast Fourier Transform*), donde las muestras de frecuencia son generadas a partir de muestras de tiempo recibidas del símbolo SEFDM [2].

c) IFFT/FFT

La conversión de señales puede ser realizada del dominio del tiempo al dominio de la frecuencia mediante el uso de la FFT, empleada en el receptor, o viceversa mediante el uso de la IFFT empleada en el transmisor [10]. En la conversión, la señal capturada es sincronizada en el dominio del tiempo para posteriormente pasar al bloque de conversión S/P y ser modulada. A la señal modulada compleja se le añade un determinado número de ceros en base a la función de $\frac{1-\alpha}{\alpha} * N$ [9] cómo se observa en la Figura 1.4.

```
-1.0000 - 1.0000i
 1.0000 + 1.0000i
 0.0000 + 0.0000i
 0.0000 + 0.0000i
 0.0000 + 0.0000i
 0.0000 + 0.0000i
```

Figura 1.4. Vector modulado complejo con la inserción de ceros [10].

La señal compensada es demodulada en el receptor, donde los símbolos SEFDM se obtienen después de la eliminación de los ceros insertados en el transmisor y finalmente, el flujo de bits es obtenido al pasar la señal por el convertidor P/S [1]. Sin embargo, la generación de símbolos modulados en un sistema SEFDM no posee un enfoque tradicional de conversión debido a la pérdida de ortogonalidad que este presenta. Para solventar este inconveniente se propone en [1] el empleo de un algoritmo usando un IFFT donde cada vector de entrada es rellenado con una cantidad

específica de ceros y otro usando múltiples IFFT que se basa en la interpolación de varios ceros en lugar de rellenar un símbolo de entrada.

d) Canal

La transmisión de información en las comunicaciones inalámbricas se lo realiza mediante un canal, cuyo medio más común en este tipo de comunicaciones es el aire, desde el transmisor al receptor [9]. Uno de los modelos de canal usados es el AWGN que añade o contamina a la señal SEFDM con ruido gaussiano, este ruido es denotado como $w(t)$ y la señal con ruido se representa como [12]:

$$r(t) = x(t) + w(t) \quad (1.3)$$

donde, $r(t)$ es la señal SEFDM que llega al receptor y $x(t)$ es la señal que sale del transmisor.

e) Detector

Esta componente, ubicada en el receptor como se muestra en la Figura 1.5, es la encargada de eliminar los efectos de ICI que posee la señal que sale del bloque FFT mediante técnicas de detección o ecualización, con el fin de recuperar la señal que fue transmitida originalmente al pasar previamente por la conversión P/S [10].

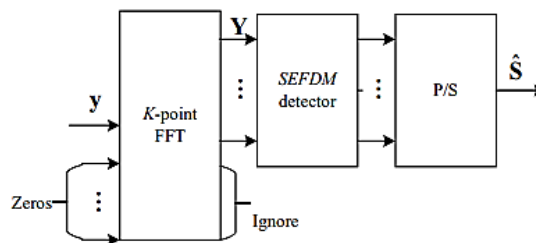


Figura 1.5. Bloque de detección en receptor en un sistema SEFDM [14].

Para garantizar un desempeño óptimo y contrarrestar los efectos negativos producidos por la presencia de ruido en el canal se utilizan técnicas de detección, como la detección ML. Sin embargo, esta técnica presenta una alta complejidad, la cual va aumentando exponencialmente con el aumento del número de subportadoras [6]. Entre otras opciones se tiene los detectores lineales, ZF (*Zero Forcing*) y MMSE, los cuales tienen un rendimiento inferior en la tasa de error en comparación con los detectores ML [7] y restringen el ahorro en ancho de banda para obtener una BER considerada aceptable dentro de las aplicaciones de comunicaciones inalámbricas [13].

1.4.1.2. Ventajas y desventajas del sistema SEFDM

A continuación, se presenta algunas ventajas y desventajas de esta técnica multiportadora.

Ventajas

- Debido a la pérdida de ortogonalidad en sus subportadoras, presenta una alta eficiencia espectral en comparación con otras técnicas multiportadora, por ejemplo, OFDM, GFDM (*Generalized Frequency Division Multiplexing*), etc [3].
- Esta técnica se considera prometedora para aplicaciones en tecnología 5G [1], [2], ya que, en futuros sistemas inalámbricos se requiere una alta eficiencia espectral además de ser compatibles para aplicaciones ópticas y sistemas híbridos [1].
- Transmite la misma tasa de datos que un sistema OFDM, pero, con un ahorro de ancho de banda en función del factor de compresión α [7].

Desventajas

- Al poseer violación en la ortogonalidad, se presenta una degradación en el BER el cual es producido por la presencia de ICI que es proporcional a la compresión del ancho de banda y al número de subportadoras [13].
- Presenta una recepción con pobre desempeño, ya que, tiene una alta pérdida de energía en el receptor en comparación con el sistema OFDM [2].
- Algunas de las técnicas para la detección de las señales presentan alta complejidad en su implementación o restringen al sistema SEFDM con el fin de mantener un BER adecuado [12].

1.4.1.3. Comparación con OFDM

En el dominio de la frecuencia, los símbolos SEFDM se encuentran comprimidos debido a la reducción de ancho de banda lo cual produce ICI y genera la no ortogonalidad [15]. Sin embargo, la técnica OFDM también es propensa a presentar ICI debido a la sensibilidad que tiene al desfase en frecuencia lo que afecta a su rendimiento [11]. La comparación de los símbolos de estas dos técnicas multiportadora se puede ver en la Figura 1.6, donde los símbolos están representados en tiempo-frecuencia y la comparación en la ortogonalidad se puede apreciar en la Figura 1.7.

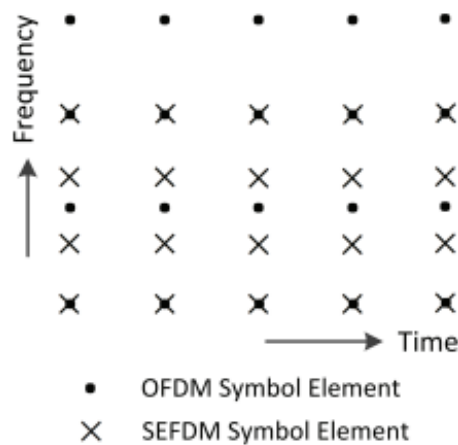


Figura 1.6. Representación tiempo-frecuencia de símbolos SEFDM y OFDM [15].

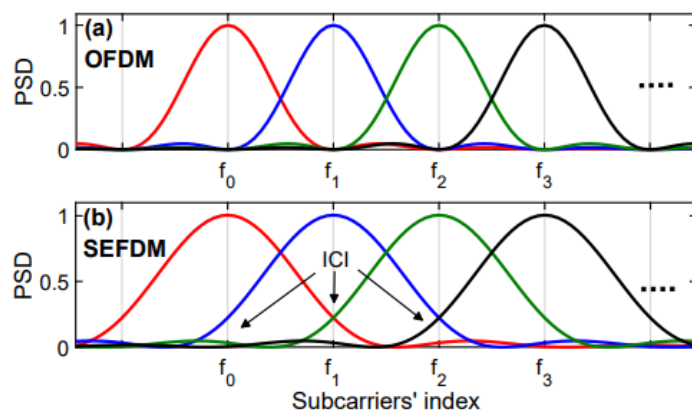


Figura 1.7. Ilustración de subportadoras en a) OFDM y b) SEFDM [7].

SEFDM comparada con OFDM, presenta un ahorro en la utilización del ancho de banda, es decir, para un mismo número de subportadoras se transmite un mayor flujo de datos [2]. Este ahorro viene dado por el factor de compresión α , el cual en OFDM es $\alpha = 1$ y para SEFDM es $\alpha < 1$ [6]. La relación del factor de compresión con el número de subportadoras de SEFDM y OFDM viene dado por la siguiente fórmula [16]:

$$N = \frac{N_o}{\alpha} \quad (1.4)$$

donde, N_o corresponde al número de subportadoras en OFDM y N representa el número extra de subportadoras SEFDM que ocuparán el mismo ancho de banda [16]

1.4.2. CARRIER FREQUENCY OFFSET

El problema de la estimación CFO aparece cuando la señal modulada se distorsiona en el receptor debido a un desajuste de frecuencia, este desajuste corresponde a la diferencia de la frecuencia portadora entre el oscilador del receptor y el oscilador del emisor [4]. El

error de sincronización en frecuencia destruye la ortogonalidad en las subportadoras produciendo el desfase de frecuencia y como consecuencia se genera ICI [17]. En la ecuación de CFO normalizado ε se toma en cuenta la fórmula del desfase de frecuencia o CFO, la cual consiste en la diferencia entre la frecuencia portadora en el transmisor f_c y la frecuencia portadora en el receptor f'_c . La ecuación de este desfase es: $f_{offset} = f_c - f'_c$ [17]. Definida la fórmula del desfase de frecuencia, se establece el CFO normalizado como la relación entre el CFO y el espaciamiento entre subportadoras Δf . La ecuación es [4], [17]:

$$\varepsilon = \frac{f_{offset}}{\Delta f} \quad (1.5)$$

El valor de ε influye directamente en el comportamiento de la señal y la diferencia de fase en el dominio de tiempo. Por otra parte, existe un comportamiento ambiguo cuando se tiene un valor de $\varepsilon > 0.5$. ya que, el rango de valores de CFO es de $[-0.5, 0.5)$ [17].

1.4.2.1. Técnicas de estimación

Debido a la importancia que tiene la estimación CFO para combatir la ICI dentro del sistema SEFDM, se realiza mediante diferentes técnicas, tanto en el dominio de tiempo y frecuencia [4]. A continuación, se muestra información acerca de estas.

1.4.2.1.1. Técnicas de estimación en dominio de tiempo

Técnica usando Prefijo cíclico (CP- *Cyclic Prefix*)

Dentro de una sincronización perfecta de símbolo, CFO posee una rotación de fase de $\frac{2\pi n \varepsilon}{N}$. Sin embargo, al asumir un canal insignificante, la diferencia de fase entre el prefijo cíclico y la parte lejana o final de un símbolo OFDM que se encuentra separada N muestras es de $\frac{2\pi \varepsilon N}{N} = 2\pi \varepsilon$. Por lo tanto, el CFO puede ser determinado por el ángulo de fase del producto del CP y su conjugada en la parte final del símbolo OFDM [5], [17]. En la Ecuación 1.6, se muestra un ejemplo de estimación CFO usando CP con un valor de $n = -1, -2, \dots, -N_g$.

$$\hat{\varepsilon} = \frac{1}{2\pi} \arg \left\{ \sum_{n=-N_g}^{-1} y^*[n] y[n + N] \right\} \quad (1.6)$$

donde, el \arg viene dado por la operación $\tan^{-1}()$ y el rango de la estimación CFO es de $[-0.5, 0.5)$ [17].

Técnica usando símbolos de entrenamiento

El empleo de símbolos de entrenamiento presenta una baja complejidad computacional, ya que, se limita el número de cálculos en el receptor [4]. Esta técnica logra cubrir un amplio rango de CFO gracias al empleo de símbolos de entrenamiento repetitivos dentro de un intervalo corto de tiempo [5]. Para la estimación CFO se establece la variable D que consiste en la relación entre la longitud del símbolo OFDM y la longitud del patrón repetitivo [17] cómo se puede ver en la Ecuación 1.7.

$$\hat{\varepsilon} = \frac{D}{2\pi} \arg \left\{ \sum_{n=0}^{\frac{N}{D}-1} y^*[n] y \left[n + \frac{N}{D} \right] \right\} \quad (1.7)$$

donde, el rango de estimación es $|\varepsilon| \leq \frac{D}{2}$ y para aumentar dicho rango se sacrifica el rendimiento del MSE (*Mean Square Error*) aumentando el valor de D y reduciendo $\frac{1}{D}$ el número de muestras [17] cómo se observa en la Figura 1.8.

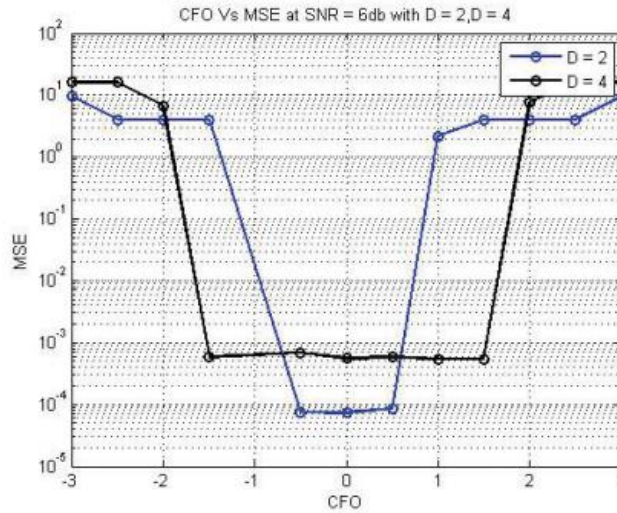


Figura 1.8. Relación entre MSE y CFO [5].

1.4.2.1.2. Técnica de estimación en dominio de la frecuencia

Técnica usando símbolos de entrenamiento

Al transmitir dos símbolos de entrenamiento idénticos consecutivamente, las señales de CFO ε se relacionan entre sí [5], [17] como se muestra en la Ecuación 1.8:

$$y_2[n] = y_1[n] e^{\frac{j2\pi\varepsilon N}{N}} \leftrightarrow Y_2[k] = Y_1[k] e^{j2\pi\varepsilon} \quad (1.8)$$

Para expresar la estimación de CFO, donde el rango de estimación es $|\varepsilon| \leq 0.5$ [5] y el rendimiento del MSE se deteriora [17], en esta técnica se hace uso de la Ecuación 1.8 y se obtiene:

$$\hat{\varepsilon} = \frac{1}{2\pi} \tan^{-1} \left(\frac{\sum_{k=0}^{N-1} \text{Im}[Y_2[k] Y_1^*[k]]}{\sum_{k=0}^{N-1} \text{Re}[Y_2[k] Y_1^*[k]]} \right) \quad (1.9)$$

1.4.2.2. Aplicación en SEFDM

El desfase de frecuencia afecta tanto al sistema OFDM y SEFDM. En el caso del sistema SEFDM, se ve afectado debido a la no ortogonalidad que este sistema presenta y como consecuencia se produce ICI [4]. Además, la presencia de ICI degrada al sistema SEFDM, razón por la cual, se han establecido diferentes técnicas de estimación CFO para poder combatir con este problema. Dichas técnicas, tanto en el dominio del tiempo y dominio de la frecuencia, son aplicadas en sistemas OFDM, pero también son empleadas en sistemas SEFDM con ciertas variaciones que permitan minimizar el efecto de ICI.

2. METODOLOGÍA

En el presente capítulo se detalla el código desarrollado para la simulación de un sistema SEFDM con estimación CFO en el software de simulación Matlab. El código posee las siguientes subsecciones: ingreso de variables iniciales, transmisor SEFDM, adición del preámbulo, ruido AWGN, receptor SEFDM con inserción, estimación y compensación de CFO, eliminación de preámbulo, cálculo de errores y generación de gráficas. Además, se presenta en detalle las funciones y variables que son utilizadas en el script principal.

El enfoque que presenta este trabajo investigativo es cuantitativo, ya que, se trabaja con estimación CFO, donde dicha variable afecta a la señal SEFDM transmitida. Con los datos obtenidos de esta señal se procede al análisis de comparación entre un escenario con estimación perfecta y uno con un tipo de estimación CFO explicados en el capítulo 1. Este análisis se lo realiza mediante la comparación de gráficas BER vs Eb/No con valores de BER previamente calculados y promediados con el empleo de una función desarrollada.

Para el tipo de alcance del proyecto se tiene un enfoque mixto, ya que, se tiene un tipo de alcance descriptivo y explicativo porque se describe el efecto que la estimación CFO tiene sobre la señal SEFDM transmitida explicado mediante la utilización de los diferentes tipos de estimación (prefijo cíclico y preámbulo) que se utilizan. Finalmente, el tipo correlacional, ya que, se compara el BER de las señales obtenidas mediante dos escenarios de simulación: el primero con estimación perfecta, es decir, el CFO no influye en la señal y el segundo con estimación empleando una de las técnicas mencionadas anteriormente. Gracias a esta comparación se analiza el efecto degradante en el BER que la estimación CFO tiene sobre las señales recibidas con diferente valor de α .

El código de simulación SEFDM con estimación CFO, es una modificación del código de la tesis del Sr. Logacho [10], se conforma de un script principal llamado *SEFDM_varios_alfa_awgn* y de varias funciones desarrolladas para el transmisor y receptor SEFDM.

2.1. SCRIPT PRINCIPAL

El programa principal *SEFDM_varios_alfa_awgn* fue desarrollado para configurar parámetros iniciales de simulación, llamar a la función *SEFDM_awgn.m* la cual recibe estos parámetros que sirven para generar las diferentes curvas que posteriormente son graficadas ocupando el comando *semilogy* para mostrar gráficas BER vs Eb/No. El proceso del script principal se observa en la Figura 2.1.

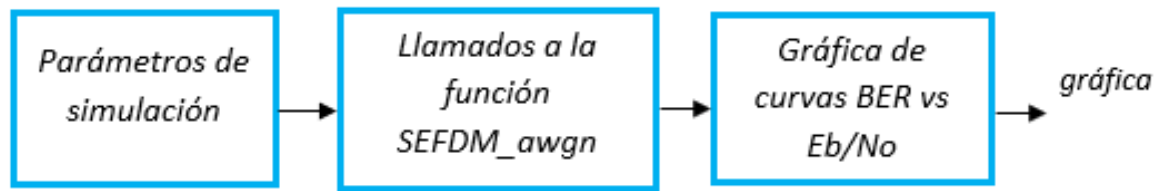


Figura 2.1. Diagrama de bloque del script principal *SEFDM_varios_alfa_awgn*.

Dentro de los parámetros de simulación se tiene: número de subportadoras, números de bits agrupados para determinar la modulación, técnica de estimación CFO, valor de CFO, factor de compresión del ancho de banda, número de iteraciones, valor de E_b/N_0 , número de símbolos SEFDM por paquete como se observa en la Tabla 2.1:

Tabla 2.1. Parámetros de configuración para simulación SEFDM con estimación CFO.

Parámetro	Descripción
N	Número de subportadoras
m	Número de bits agrupados para determinar la modulación
<i>estimacion</i>	Técnica de estimación CFO (PC o Preámbulo)
<i>CFO</i>	Valor CFO para técnica de estimación (PC o Preámbulo)
<i>alfa</i>	Valor de compresión de ancho de banda
E_b/N_0	Valor de E_b/N_0
N_{sym}	Número de símbolos SEFDM por paquete

Las variables N , E_b/N_0 y N_{sym} son constantes para todas las gráficas de simulación teniendo así un número N de 16 subportadoras y un valor N_{sym} de 5 símbolos SEFDM por paquete. Las demás variables son configurables según se requiera para la simulación, como se puede ver en el segmento de código 2.1. Además, se utiliza un número de iteraciones igual a 20000 para obtener un resultado aceptable en las curvas de las gráficas BER vs E_b/N_0 , ya que, un número menor a 10000 iteraciones aproximadamente presentan unas curvas no definidas correctamente.

```

%INICIALIZACIÓN DEL PROGRAMA
clc;
clear;
close all;

% INGRESO DE VARIABLES
%Datos del transmisor
N=16;           %Número de Subportadoras
m=6;           %Número de bits agrupados según la modulación
               % 2-->QPSK | 4--->16QAM | 6-->64QAM

%Selección de modulación
if m==2
    modulaciones='Modulación: QPSK';
elseif m==4
    modulaciones='Modulación: 16QAM';
elseif m==6
    modulaciones='Modulación: 64QAM';
end

%Selección técnica de estimación
    %1---> Estimación PC dominio tiempo
    %2---> Estimación Preámbulo dominio tiempo
estimacion=1;

%Indicar valor CFO para técnica de estimación PC o Preámbulo
CFO=0.1;

%Ingreso de valores de alfa
a=[0,2];       %Número de ceros añadidos (Definición SEFDM)
alfa=(N./(N+a)); %Definición del Factor de Compresión de AB

%Datos para la obtención de gráficas
Ntrials=20000; %Número de iteraciones para cada valor de SNR
EbNo=0:2:30;  %Eb/No
N_sym=5;      %numero de simbolos SEFDM por paquete

```

Segmento de código 2.1. Ingreso de los parámetros de simulación del script principal

SEFDM_varios_alfa_awgn.m

Una vez definidos los parámetros de simulación, se llama varias veces a la función *SEFDM_awgn.m*, ya que, es de suma importancia para la generación de curvas BER vs Eb/No con las diferentes técnicas de estimación CFO y consta de varias secciones que permiten el funcionamiento, por ejemplo, del transmisor y receptor SEFDM. Finalmente, se grafica las curvas mostrando la comparación BER vs Eb/No para el análisis de los resultados en los diferentes escenarios de simulación.

2.2. FUNCION SEFDM_AWGN

Esta función cumple un rol importante, como se mencionó anteriormente, dentro del script principal de la simulación. La función posee secciones de código relacionadas con el transmisor y receptor SEFDM, valores iniciales de simulación, agregación/eliminación de preámbulo, matriz de subportadoras, canal AWGN y promedio de tasa de errores. A continuación, se muestra las variables de entrada y se detalla cada sección que compone esta función.

```
function [BER] = SEFDM_awgn(N,m,alfa,Ntrials,EbNo,N_sym,estimacion,CFO)
```

Segmento de código 2.2. Parámetros de entrada de función *SEFDM_awgn.m*

2.2.1. VALORES INICIALES

Estos valores ayudan al desarrollo del código en ciertas secciones del mismo, ya que, contiene: orden de modulación, tamaño de la IFFT/FFT que se muestra en la Figura 1.3, vector con valores de SNR y la matriz donde se almacenan los bits errados obtenidos en cada iteración de simulación. Los valores iniciales se muestran en el segmento de código 2.3.

```
%VALORES INICIALES
M=2^m;                               %Orden de la modulación
IFFT_FFT_size=round(N/alfa);         %Tamaño de la IFFT/FFT
Snr=EbNo+10*log10(m);                %Obtención del vector SNR
BER_T=zeros(Ntrials,length(Snr));    %Creación de la matriz
```

Segmento de código 2.3. Valores iniciales de la función *SEFDM_awgn.m*

La sección de matriz de subportadoras viene dada en base a la Ecuación 1.2 de la fórmula del símbolo SEFDM, donde se forma llenando una matriz de ceros creada previamente. Esto se aprecia en el segmento de código 2.4.

```
%MATRIZ DE SUBPORTADORAS F
F=zeros(N,N);
for k=0:N-1
    for n=0:N-1
        F(k+1,n+1)=1/sqrt(N)*exp(1i*2*pi*n*alfa*(k/N));
    end
end
```

Segmento de código 2.4. Matriz de subportadoras.

El alcance del proyecto define utilizar tasa de codificación $\frac{1}{2}$. Además, existe la sección de codificación FEC (*Forward Error Correction*) y codificación convolucional para garantizar una transmisión confiable y correcta de los símbolos SEFDM a través del canal AWGN como se observa en el segmento de código 2.5, donde se establece los objetos del codificador y decodificador convolucional

```
%codificacion
tc=1/2;      %tasa codificacion
traceBack = 32;  %parametro de codificador FEC
trellis = poly2trellis(7,[171 133]);

%codificador convolucional
convEncoder =
comm.ConvolutionalEncoder('TrellisStructure',trellis);
vitDecoder = comm.ViterbiDecoder('TrellisStructure',trellis, ...
    'InputFormat','unquantized','TracebackDepth',traceBack);
```

Segmento de código 2.5. Tasa de codificación $\frac{1}{2}$ y codificador/decodificador convolucional.

Finalmente, se forma la matriz del canal basada en la fórmula de la forma matricial de una señal SEFDM que es [1]:

$$X = FS \tag{2.1}$$

donde, cada componente representa [1]:

- X es un vector Q -dimensional de muestras en el dominio del tiempo.
- S es un vector N -dimensional de símbolos transmitidos.
- F es una matriz de subportadoras de tamaño $Q \times N$ con elementos iguales a $e^{\frac{j2\pi n a k}{Q}}$.

En el segmento de código 2.6, se obtiene la matriz de canal con el producto entre la matriz de subportadoras F , formada en el segmento de código 2.4 y su respectiva conjugada F^* sin la intervención de ruido gaussiano que se añadirá posteriormente después de la etapa de transmisión.

```
%Matriz del Canal
H=F'*F;
```

Segmento de código 2.6. Formación de la matriz de canal.

2.2.2. OBTENCIÓN DE PREÁMBULO

Para la obtención del preámbulo, en el segmento de código 2.7 se utiliza una función llamada *get_pream.m*,

```
%Obtencion Preámbulo  
[pream]=get_pream;
```

Segmento de código 2.7. Obtención del preámbulo ocupando la función *get_pream.m*

La función *get_pream.m* ocupa una secuencia de pseudo ruido (*PN- Pseudo Noise*) con tamaño 16, donde las frecuencias pares son transmitidas con ceros. En la simulación, la secuencia es almacenada en la variable *PN_seq*. Con la secuencia definida, se realiza el proceso de ifft para pasar al dominio del tiempo y luego añadir el prefijo cíclico con tamaño de $\frac{N}{4} = 4$.

La forma final del preámbulo se puede observar en la Figura 2.2.

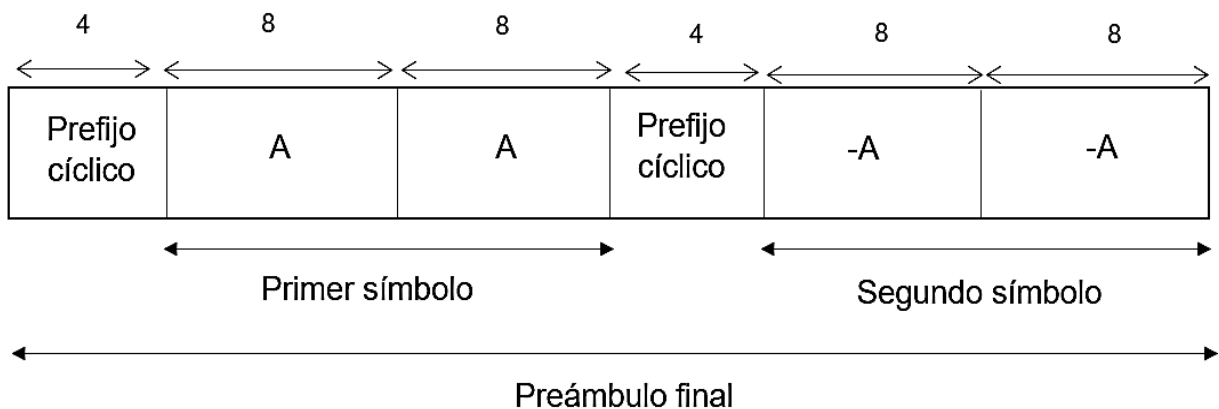


Figura 2.2. Forma de preámbulo final empleado en la función *get_param.m*.

El preámbulo final trabaja con dos prefijos cíclicos colocados antes de cada símbolo. Cada símbolo tiene un tamaño de 16 y cada prefijo cíclico de 4, por lo tanto, se tiene un tamaño final de preámbulo de 40. Esto se puede ver en el segmento de código 2.8.

```

function [pream] = get_pream

%Secuencia de PseudoRuido
PN_seq = [1 1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 -1]; %Tamaño 16
PN_seq(2:2:end) = 0; %Transmision de ceros en frecuencia
pares
%Mapeo de subportadoras.
N=16;
prel = ifft(PN_seq, N); %Dominio del tiempo
prel_cp=[prel(N-(N/4)+1:end)]; %CP

% Forma del Preámbulo --> [CP A A CP -A -A]
pream=[prel_cp prel prel_cp -prel];

end

```

Segmento de código 2.8. Formación del preámbulo dentro de la función *get_pream.m*

2.2.3. TRANSMISOR SEFDM

En el desarrollo del transmisor SEFDM, se ocupan secciones de código de los valores iniciales expuestos anteriormente. Además, en la simulación se inicia el sistema SEFDM con dos lazos for, el primer lazo for va de 1 hasta la longitud del vector con valores SNR y el otro lazo for corresponde al número de iteraciones. A continuación, en la Figura 2.3 se presenta el diagrama de bloques del funcionamiento general del transmisor SEFDM.

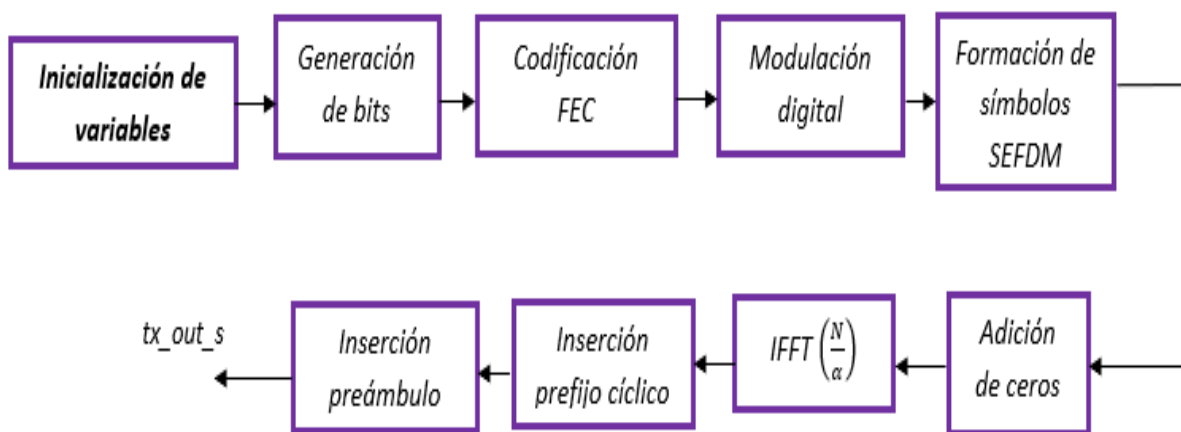


Figura 2.3. Diagrama de bloque del transmisor SEFDM.

2.2.3.1. Generación de bits y codificación FEC

En la generación de bits se ocupa el comando *randi*, el cual genera números aleatorios enteros dentro de un determinado intervalo, el cual es definido entre 0 y 1. Este comando genera un vector columna de tamaño $N * m * tc * N_{sym}$. Después, los bits generados son codificados usando la técnica FEC, donde en Matlab se ocupa el objeto *covEncoder*,

generado en variables iniciales, y estos bits codificados son almacenados dentro de la variable *data_cod*. Ambos procesos se pueden observar en el segmento de código 2.9.

```
bit_tx_s=randi([0,1],N*m*tc*N_sym,1); %Generación de bits
data_cod = convEncoder(bit_tx_s); %codificación FEC
```

Segmento de código 2.9. Generación de bits y codificación FEC.

2.2.3.2. Modulación digital

Los bits codificados son modulados según el orden de modulación definido en las variables iniciales $M = 2^m$. El orden de modulación, dentro de la función *SEFDM_awngn.m* depende del ingreso de la variable *m* dentro el script principal, ya que, en la simulación se maneja 3 órdenes de modulación:

Tabla 2.2. Opciones de orden de modulación según el ingreso de la variable *m*.

Valor de ingreso de m	Orden de modulación
2	Modulación QPSK
4	Modulación 16-QAM
6	Modulación 64-QAM

Una vez definido el orden de modulación, se ocupa el comando *qammod*, como se observa en el segmento de código 2.10, el cual permite modular digitalmente una señal de entrada, la cual corresponde a los bits codificados con técnica FEC *data_cod* con orden de modulación *M* definido. Además, dentro de este comando se establecen los parámetros como *InputType* que indica que los datos de entrada son de tipo binario y *UnitAveragePower* indica que se escala a una potencia promedio unitaria. Los bits modulados son almacenados en la variable *tx_mod*.

```
%Modulación digital
tx_mod=qammod(data_cod,M,'InputType','bit','UnitAveragePower',...
true);
```

Segmento de código 2.10. Modulación digital de los bits codificados.

2.2.3.3. Formación de símbolos SEFDM y adición de ceros

En primer lugar, se establece un vector de ceros en base a la función [9]:

$$ceros = \frac{1-\alpha}{\alpha} * N = \frac{N}{\alpha} - N \quad (2.2)$$

Generado el vector fila de ceros, dentro de la variable *ceros*, como se observa en el segmento de código 2.11, se procede a realizar la conversión paralelo/serie de los datos modulados *tx_mod*. Este cambio es almacenado en la variable *tx_mod_s*.

```
ceros=zeros(1,IFFT_FFT_size-N);
tx_mod_s=tx_mod.';
```

Segmento de código 2.11. Generación de ceros y conversión paralelo/serie.

La generación de símbolos SEFDM se realiza con un lazo *for* que va de 1 hasta *N_sym*, cuyo valor es 5. Al trabajar con un valor fijo de 16 subportadoras, se va tomando cada 16 elementos de la variable *tx_mod_s*, como se observa en la Tabla 2.3, para cada símbolo SEFDM por separado y estos son almacenados en *tx_sefdm*.

Tabla 2.3. Número de elementos tomados de la variable *tx_mod_s* en cada iteración.

Valor de <i>jj</i> del lazo <i>for</i>	Elementos de <i>tx_mod_s</i>
1	<i>tx_mod_s</i> (1:16)
2	<i>tx_mod_s</i> (17:32)
3	<i>tx_mod_s</i> (33:48)
4	<i>tx_mod_s</i> (49:64)
5	<i>tx_mod_s</i> (65:80)

Después, la matriz de ceros creada previamente es agregada a cada símbolo SEFDM contenido dentro de la variable *temp*, como se observa en el segmento de código 2.12

```
for jj=1:N_sym %simbolos
    temp=tx_mod_s((jj-1)*N+1:jj*N);
    tx_sefdm=[temp ceros]; %adición de
ceros
```

Segmento de código 2.12. Formación de símbolos SEFDM.

2.2.3.4. IFFT

Con el bloque IFFT, se busca trasladar del dominio de la frecuencia al dominio del tiempo de la señal SEFDM. Este proceso se consigue usando el comando *ifft*, cuyo vector de entrada es la variable *tx_sefdm* y el tamaño viene dado por la fórmula $\frac{N}{\alpha}$. Después, se desconoce los últimos valores del vector resultante, es decir, se busca obtener una señal de tamaño *N* en cada símbolo SEFDM. Este proceso es almacenado en la variable *tx_out* como se observa en el segmento de código 2.13.

```

ifft_tx=ifft(tx_sefdm,IFFT_FFT_size);      %Bloque IFFT
tx_out=ifft_tx(1:N);                       %Desconocer los
últimos valores

```

Segmento de código 2.13. Bloque IFFT.

2.2.3.5. Adición de prefijo cíclico

La adición del prefijo cíclico se realiza en cada símbolo SEFDM, es decir, en cada iteración se añade el CP al comienzo del símbolo, como se observa en la Figura 2.4. El tamaño del prefijo cíclico en la simulación viene dado por $\frac{N}{4} = 4$, dando como resultado un tamaño de símbolo SEFDM con CP de 20 y un tamaño total de 100 con los símbolos agrupados al finalizar el lazo for.

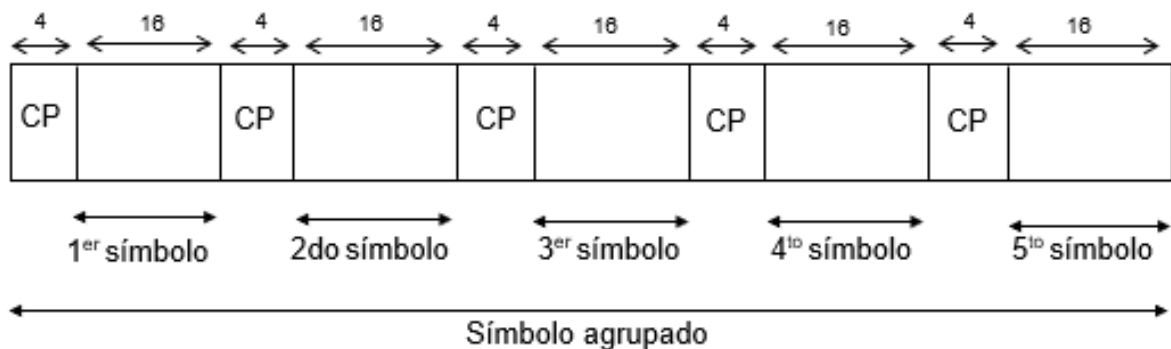


Figura 2.4. Adición del CP a cada símbolo SEFDM.

Esto se puede observar en el segmento de código 2.14, donde la formación del CP viene dada por los últimos 4 valores del `tx_out` y colocados al inicio del mismo, almacenando la señal resultante en la variable `tx_out_s`.

```

tx_out_cp=[tx_out(N-(N/4)+1:end) tx_out]; %CP
tx_out_s((jj-
1)*(N+N/4)+1:jj*(N+N/4))=tx_out_cp;

```

Segmento de código 2.14. Adición de prefijo cíclico.

2.2.3.6. Adición del preámbulo

Para esta sección se utiliza la variable `pream`, declarada en el segmento de código 2.7, donde se almacena el preámbulo obtenido gracias a la función `get_pream.m` desarrollada en el segmento de código 2.8. El preámbulo es añadido a la agrupación de símbolos SEFDM, dando como resultado la variable `txSig` cuyo tamaño total es 140, ya que, 100 son de los símbolos SEFDM agrupados y 40 del preámbulo.

La adición del preámbulo se puede ver en el segmento de código 2.15.

```
%Adición preambulo
txSig=[pream, tx_out_s];
```

Segmento de código 2.15. Adición de preámbulo en la agrupación de símbolos SEFDM.

2.2.4. CANAL AWGN

Una vez obtenida la señal $txSig$ que sale del transmisor, se añade ruido gaussiano a dicha señal. Este proceso se realiza con el comando *awgn*, donde los parámetros de la función son: la señal de entrada $txSig$, los valores en decibelios del vector SNR establecido en las variables iniciales y el *signalpower* como *measured* para elegir un nivel de ruido adecuado según el valor del vector SNR. Además, se calcula la potencia promedio de los datos modulados que es almacenada en la variable pt .

```
%Canal AWGN
canal_rx=awgn(txSig, Snr(i), 'measured', 'db'); %adición ruido
pt=mean(abs(tx_mod_s.^2));
```

Segmento de código 2.16. Canal AWGN.

2.2.5. RECEPTOR SEFDM

Al igual que en el transmisor SEFDM, en el receptor se ocupa secciones de código establecidos en las variables iniciales además de la señal con presencia de ruido AWGN. A continuación, en la Figura 2.5 se presenta el diagrama de bloques general del receptor SEFDM.

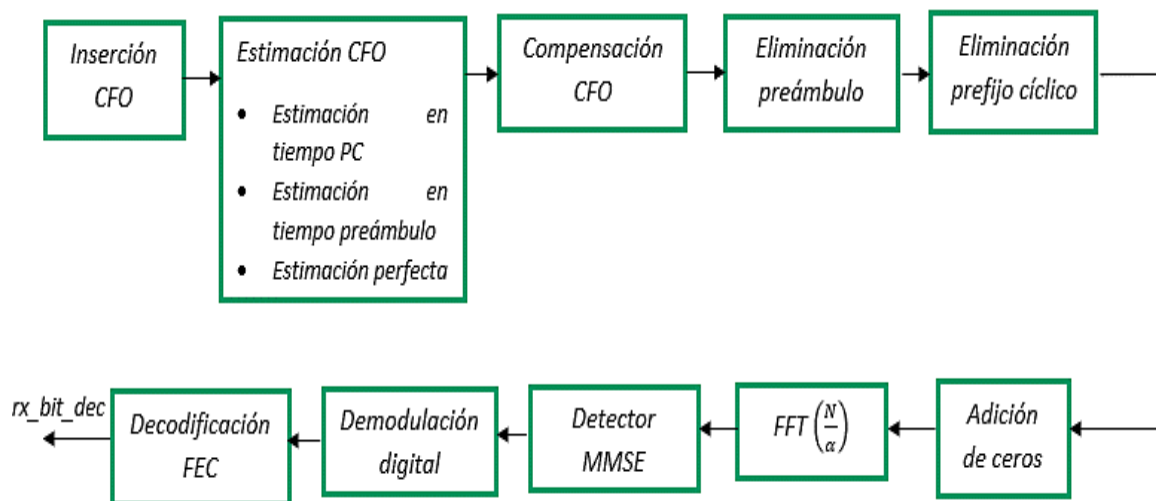


Figura 2.5. Diagrama de bloques del receptor SEFDM.

2.2.5.1. Inserción CFO

Es el primer paso de la sincronización en recepción. Para la inserción CFO se ocupa la función *ins_CFO.m*. En esta función se recibe la señal que sale del canal AWGN, la señal *canal_rx*, para multiplicarla por la expresión $\exp\left(\frac{j2\pi n\epsilon}{N}\right)$ [8]. Como se observa en el segmento de código 2.17:

```
function [sym_CFO]=ins_CFO(rxSigu, CFO, N)
    %Inserción del CFO
    nn=[0:19 0:19 0:19 0:19 0:19 0:19 0:19]; %Separación
    cada símbolo
    sym_CFO = rxSigu.*exp(1i*2*pi*nn*CFO/N);
end
```

Segmento de código 2.17. Inserción CFO.

donde, los parámetros de simulación son:

- *rxSigu*: Es la señal que sale del canal AWGN.
- *nn*: Corresponde a la separación de cada símbolo SEFDM porque el tamaño total de la señal que llega del canal AWGN es de 140, por tal razón, se ocupa 7 intervalos de tamaño 20 (0:19) para que pueda aplicarse el efecto del CFO a cada símbolo SEFDM por separado, ya que, la expresión utilizada es dividida para *N*.
- *CFO*: Representa al ϵ y equivale al valor de CFO ingresado en el script principal *SEFDM_varios_alfa_awgn*. Este valor de CFO influye en los dos tipos de estimación (CP y preámbulo), sin embargo, cuando se selecciona la opción de estimación perfecta el valor de CFO es igual a 0.
- *N*: Corresponde al número total de subportadoras.

2.2.5.2. Estimación CFO

En esta sección, dependiendo del valor ingresado en la variable inicial *estimacion*, se selecciona un tipo de estimación, los cuales se observan en la Tabla 2.4.

Tabla 2.4. Técnicas de estimación.

Valor variable <i>estimacion</i>	Técnica de estimación
1	Estimación en el dominio del tiempo con prefijo cíclico
2	Estimación en el dominio del tiempo con preámbulo
3	Estimación perfecta

Para realizar los tipos de estimación, se desarrolló una función para cada uno de los tipos de estimación que son afectados por CFO, ya que, en estimación perfecta no se tiene una afectación del CFO.

2.2.5.2.1. Técnica de estimación en dominio de tiempo usando prefijo cíclico

En esta técnica se estima el CFO, basándose en la Ecuación 1.6 donde, se obtiene el ángulo de fase determinado por el producto entre el CP y su conjugada en la parte final del símbolo SEFDM. Para la simulación de esta técnica, se utiliza la función *CFO_tiempo_PC.m* como se observa en el segmento de código 2.18, cuyos parámetros de entrada son la señal insertada el CFO y el *N*. Se ocupa la variable *delay* que indica la posición del CP en cada uno de los símbolos SEFDM. El primer *delay* es en 40, ya que, en primer lugar, se tiene el preámbulo con tamaño 40 y se va sumando 20 (16 de símbolo + 4 de CP) para la posición del CP en otro símbolo SEFDM. La correlación se realiza con la técnica de la ventana deslizante, la cual empieza definiendo su tamaño en la variable *nn* que es $40 + (1:4)$. El recorrido de la ventana parte de la posición 41 recorriendo 16 posiciones y así hasta recorrer el símbolo SEFDM. A continuación, se muestra en la Tabla 2.5 el recorrido de la ventana dentro del primer símbolo SEFDM.

Tabla 2.5. Recorrido de la ventana dentro del primer símbolo SEFDM.

Posición dentro del símbolo SEFDM	Recorrido de la ventana
41	41-57
42	42-58
43	43-59
44	44-60

Como se observa en la Tabla 2.5, el recorrido de la ventana termina en la posición 60, ya que, es donde inicia el CP del segundo símbolo SEFDM. Además, se observa la estimación que se realiza para los 5 símbolos SEFDM y en la parte final se calcula un promedio de los 5 para tener un mejor resultado al momento de la simulación.

```

function CFO_est=CFO_tiempo_PC(sym_CFO,N)

PC=1/4; %CP
Ng = N*PC; %Tamaño del CP
delay = 40 ; %Retraso posición del CP
nn = delay+(1:Ng); %Tamaño de la ventana
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+N); %Correlacion
CFO_est1 = 1/(2*pi)*angle(sum(tmp)); %Primer símbolo SEFDM con CFO

delay = 60 ; %Retraso posición del CP
nn = delay+(1:Ng);
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+N);
CFO_est2 = 1/(2*pi)*angle(sum(tmp)); %Segundo símbolo SEFDM con CFO

delay = 80 ; %Retraso posición del CP
nn = delay+(1:Ng);
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+N);
CFO_est3 = 1/(2*pi)*angle(sum(tmp)); %Tercer símbolo SEFDM con CFO

delay = 100 ; %Retraso posición del CP
nn = delay+(1:Ng);
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+N);
CFO_est4 = 1/(2*pi)*angle(sum(tmp)); %Cuarto símbolo SEFDM con CFO

delay = 120 ; %Retraso posición del CP
nn = delay+(1:Ng);
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+N);
CFO_est5 = 1/(2*pi)*angle(sum(tmp)); %Quinto símbolo SEFDM con CFO

%Promedio
CFO_est=(CFO_est1+CFO_est2+CFO_est3+CFO_est4+CFO_est5)/5;
end

```

Segmento de código 2.18. Técnica de estimación en dominio de tiempo usando CP.

2.2.5.2.2. Técnica de estimación en dominio de tiempo usando preámbulo

Esta técnica es similar a la anterior, con la diferencia que se utiliza la variable D , la cual indica el número de símbolos idénticos que se tiene dentro del preámbulo utilizado. Para esta técnica se ocupa la variable $delay$ que indica la posición de inicio después del CP dentro del preámbulo, como se observa en la Figura 2.2. Además, la variable nn que, al igual que la técnica anterior, indica el tamaño de la ventana. En esta técnica se ocupa la mitad de símbolo con tamaño 8 para el recorrido de la ventana, dando así un tamaño de ventana de $4 + (1:8)$.

El recorrido de la ventana parte de la posición 4, después del CP, donde se recorre 8 posiciones que corresponden a la mitad del símbolo usando la primera parte del preámbulo $[A A]$. La otra mitad del símbolo recorre de la posición 13 hasta la 20. Para la segunda parte del preámbulo $[-A -A]$ se realiza el mismo proceso, con la diferencia que el recorrido de la

ventana parte de la posición 25, después del CP, recorriendo los dos símbolos idénticos hasta la posición 40, donde termina el preámbulo como se puede ver en la Tabla 2.6. Finalmente, se obtiene el promedio de estos dos cálculos para mejorar el resultado final de la simulación. En el segmento de código 2.19, se observa la aplicación de esta técnica en base a la Ecuación 1.7.

Tabla 2.6. Recorrido de la ventana dentro de las partes que conforman el preámbulo.

Posición de inicio dentro de la primera parte del preámbulo [A A]	Recorrido de la ventana	Posición de inicio dentro de la primera parte del preámbulo [-A -A]	Recorrido de la ventana
4	5-12	24	25-32
	13-20		33-40

```
function CFO_est=CFO_tiempo_pream(sym_CFO)

% Forma del Preámbulo generado
% [CP A A CP -A -A]

N=16; %Tamaño del patrón repetitivo
D = 2; %Numero de simbolos identicos

% [A A]
delay = 4; %retraso indica posición de inicio
Preambulo
nn = delay+(1:8); %Tamaño de la ventana
tmp = conj(sym_CFO(nn)).*sym_CFO(nn+(N/D)); %Correlación
CFO_est1 = D/(2*pi)*angle(sum(tmp));

% [-A -A]
delay1 = 24; %retraso indica posición de inicio
Preambulo
nn1 = delay1+(1:8); %Tamaño de la ventana
tmp = conj(sym_CFO(nn1)).*sym_CFO(nn1+(N/D)); %Correlación
CFO_est2 = D/(2*pi)*angle(sum(tmp));

%Promedio
CFO_est= (CFO_est1+CFO_est2)/2;

end
```

Segmento de código 2.19. Técnica de estimación en dominio de tiempo usando preámbulo.

2.2.5.3. Compensación CFO

Es el proceso contrario a la inserción de CFO, ya que, en este caso se divide la señal con CFO para la expresión $\exp\left(\frac{j2\pi n\epsilon}{N}\right)$ [8]. Para la compensación, se utiliza la función *comp_CFO* cuyos parámetros de entrada son la señal con CFO, la señal estimada con una de las técnicas de estimación y *N*.

En el segmento de código 2.20, se observa el mismo proceso al usado en el segmento de código 2.17 con la diferencia que se usan las siguientes variables:

- *sym_CFO*: Corresponde a la señal insertada con CFO.
- *CFO_est*: Es la señal que sale del bloque de estimación con una de las técnicas de estimación CFO detalladas en la Tabla 2.4. En estimación perfecta, el valor de este parámetro es 0, ya que, no se tiene afectación de CFO.

```
function [Sym_CFO_comp]=comp_CFO(sym_CFO, CFO_est, N )  
  
    %Compensacion de CFO  
    nn=[0:19 0:19 0:19 0:19 0:19 0:19 0:19];      %Seperación cada  
símbolo  
    Sym_CFO_comp=sym_CFO./exp(1i*2*pi*CFO_est*nn/N);  
  
end
```

Segmento de código 2.20. Compensación CFO.

2.2.5.4. Eliminación del preámbulo

Para la eliminación del preámbulo, se trabaja con la señal compensada que sale de la sección de compensación CFO. Para esto, se toma los datos de la señal desde la posición 41, ya que, el tamaño del preámbulo definido es de 40 y se almacena en la variable *canal_rx_1*. Esto se puede ver en la sección de código 2.21.

```
canal_rx_1=Sym_CFO_comp(length(pream)+1 :end); %Quita preambulo señal con  
CFO
```

Segmento de código 2.21. Eliminación preámbulo.

2.2.5.5. Eliminación prefijo cíclico

Se ocupa la señal sin preámbulo y dentro de un bloque for, se elimina el CP, cuyo tamaño es $\frac{N}{4} = 4$, de cada uno de los símbolos SEFDM. Una vez eliminado el CP, se almacena en la variable *temp*, tal como se observa en el segmento de código 2.22.

```
for jj=1:N_sym
    temp=canal_rx_1((jj-1)*(N+N/4)+1+N/4:jj*(N+N/4)); %quita PC
```

Segmento de código 2.22. Eliminación prefijo cíclico.

2.2.5.6. Adición de ceros y bloque FFT

Para la definición del bloque FFT, se une la matriz de ceros creada previamente en los valores iniciales con la señal sin CP *temp*, para guardar el resultado en la variable *rx_sefdm_vec*. Con el proceso de FFT se busca pasar del dominio del tiempo al dominio de la frecuencia, para ello se utiliza el comando *fft* donde, los parámetros de entrada son: la señal con el bloque de ceros *rx_sefdm_vec* y el tamaño que viene dado por la fórmula $\frac{N}{\alpha}$. Realizado el proceso FFT, se guarda el resultado en la variable *fft_rx* para luego tomar los *N* valores porque se quiere tener cada símbolo SEFDM con un valor *N*, en este caso de 16. La obtención de los *N* valores de la variable se guarda en la variable *Rx* como se observa en el segmento de código 2.23.

```
rx_sefdm_vec=[temp,ceros]; %Definición FFT
para SEFDM
fft_rx=fft(rx_sefdm_vec,IFFT_FFT_size); %Bloque FFT
Rx=fft_rx(1:N); %Desconocer los
últimos valores
```

Segmento de código 2.23. Bloque FFT.

2.2.5.7. Detector MMSE

En primer lugar, para desarrollar la detección MMSE se utiliza la función *MMSE.m*, como se puede ver en el segmento de código 2.24, donde los parámetros de entrada de esta función son:

- *R*: Corresponde a la señal de entrada.
- *I*: Es la matriz identidad.
- *snrdb*: Son cada uno de los valores en decibelios del vector SNR establecido en las variables iniciales.

- H : Matriz del canal.
- pt : Es la potencia promedio de la señal recibida.

```
function [Sv] = MMSE(R,I,snrdb,H,pt)
    snr=10^(snrdb/10);           %Valor de SNR en veces
    W=((H'*H)+(pt/snr)*I)\H';    %Aplicación de la definición detector
MMSE
    Sv=W*R;                      %Obtención de símbolos
end
```

Segmento de código 2.24. Función MMSE.

Después de la aplicación de la función *MMSE.m*, se realiza la conversión serie/paralelo de la variable Rx obteniendo así la nueva variable Rx_p . Como se observa en el segmento de código 2.25, se debe definir la matriz identidad con el comando *eye* y se aplica el detector MMSE a la señal Rx_p usando la función *MMSE.m*, guardando el resultado en la variable rx_det , a la cual se realiza la conversión paralelo/serie para la demodulación digital.

```
Rx_p=Rx.';
I=eye(N);                                     %Matriz Identidad
rx_det=MMSE(Rx_p,I,Snr(i),H,pt/(N*4));      %Detector MMSE
rx_det_s((jj-1)*N+1:jj*N,1)=rx_det.';
end
```

Segmento de código 2.25. Detección MMSE.

Al finalizar el lazo for, como se observa en el segmento de código 2.26, se obtendrá una señal con símbolos SEFDM de tamaño 80, ya que, se ha eliminado el CP y preámbulo.

```
for jj=1:N_sym
    temp=canal_rx_1((jj-1)*(N+N/4)+1+N/4:jj*(N+N/4)); %quita PC
    rx_sefdm_vec=[temp,ceros];                       %Definición FFT
para SEFDM
    fft_rx=fft(rx_sefdm_vec,IFFT_FFT_size);         %Bloque FFT
    Rx=fft_rx(1:N);                                 %Desconocer los
últimos valores
    Rx_p=Rx.';
    I=eye(N);                                       %Matriz
Identidad
    rx_det=MMSE(Rx_p,I,Snr(i),H,pt/(N*4));         %Detector MMSE
    rx_det_s((jj-1)*N+1:jj*N,1)=rx_det.';
end
```

Segmento de código 2.26. Lazo for para la obtención de símbolos SEFDM sin CP y preámbulo.

2.2.5.8. Demodulación digital y decodificación FEC

Para el proceso de demodulación, en Matlab se utiliza el comando *qamdemod* cuyos parámetros de entrada son los mismos a los usados en la modulación: el vector fila de la señal que sale del detector MMSE y el orden de modulación definido en las variables iniciales $M = 2^m$. La demodulación digital se almacena en la variable *rx_dem*.

Realizada la demodulación digital, el siguiente proceso es la decodificación la cual utiliza el objeto *vitDecoder*, creado en las *variables iniciales*, cuyo parámetro de entrada es la señal demodulada *rx_dem*. La decodificación con técnica FEC se guarda en la variable *rx_bit_dec*. Ambos procesos se observan en el segmento de código 2.27.

```
%Demodulación digital
rx_dem=qamdemod(rx_det_s,M,'OutputType','approxllr','UnitAveragePower',...
. true,'NoiseVariance',noiseVar);
rx_bit_dec = vitDecoder(rx_dem); %Decodificacion FEC
```

Segmento de código 2.27. Demodulación digital y decodificación FEC.

2.2.6. CÁLCULO DE BITS ERRADOS

La señal pasa a través de todos los procesos que fueron descritos anteriormente, el transmisor SEFDM entrega la señal al receptor después de pasar por el canal AWGN. Sin embargo, para realizar el análisis en detalle de los bits salientes del bloque receptor SEFDM, se debe realizar el cálculo de los bits errados.

Para ello, se ocupa el comando *bitter*, el cual realiza una comparación entre dos parámetros de entrada de igual dimensión, en este caso, para obtener la tasa de error de bits se compara los bits originalmente transmitidos en la variable *bits_tx_s* y los bits receptados a la salida del codificador FEC *rx_bits_dec*. El comando *bitter* entrega el número de bits errados, los cuales son almacenados en la variable *n_error* y en la variable *ratio* se almacena el resultado de la relación entre el número de bits errados *n_error* con el número total de bits. Obtenido el número de bits errados y su relación, se genera la matriz de errores *BER_T* con dimensiones $j \times i$ formada por el valor dentro de la variable *ratio*. Este cálculo se puede ver en el segmento de código 2.28.


```

%Número de Errores
[n_error, ratio]=biterr(bit_tx_s(1:end-
traceBack), rx_bit_dec(traceBack+1:end));

%Matriz de tasa de errores
BER_T(j,i)=ratio;
end
end

```

Segmento de código 2.28. Cálculo de bits errados.

Después, a la salida de lazos for que inician el sistema SEFDM, se realiza un promedio de la tasa de errores de la matriz BER_T . Este resultado es el que entrega la función principal *SEFDM_awgn.m* para que, en el script principal *SEFDM_varios_alfa_awgn* se pueda realizar las gráficas BER vs E_b/N_0 correspondientes para el análisis entre los dos escenarios de simulación: uno con estimación perfecta y otro afectado por CFO con alguna de las técnicas de estimación CFO descritas anteriormente. A continuación, se muestra el segmento de código 2.29, donde se calcula el promedio de la matriz de errores.

```

%Promedio de tasa de Errores
BER=mean(BER_T);

```

Segmento de código 2.29. Promedio de la matriz de tasa de errores.

2.3. GRÁFICAS DE RESULTADOS EN SCRIPT PRINCIPAL

Para la obtención de gráficas, se utiliza el script principal *SEFDM_varios_alfa_awgn.m* además de dos valores diferentes de α . En primer lugar, se realizan las gráficas de estimación perfecta, donde el valor de CFO es 0, ya que, no existe influencia de CFO y se ingresa el valor de 3, en referencia a la Tabla 2.4, porque la estimación perfecta tiene ese valor de parámetro de entrada dentro de la función *SEFDM_awgn.m*.

Después, se obtiene los parámetros para graficar la técnica de estimación CFO indicada en la variable *estimacion* dentro de los parámetros de simulación del script principal con valor de CFO ingresado en la variable *CFO*. Los datos para las gráficas se obtienen con el segmento de código 2.30.

```

% OBTENCIÓN DE LOS VECTORES BER PARA DISTINTOS VALORES DE ALFA
%Uso de la función SEFDM_awgn

%Gráficas con Estimación PERFECTA
BER(1,:)=SEFDM_awgn(N,m,alfa(1),Ntrials,EbNo,N_sym,3,0); %Alfa=N/N+0
grafica1=BER(1,:);
BER(2,:)=SEFDM_awgn(N,m,alfa(2),Ntrials,EbNo,N_sym,3,0); %Alfa=N/N+2
grafica2=BER(2,:);
%Gráficas con Técnica de estimacion a ELEECCION
BER(1,:)=SEFDM_awgn(N,m,alfa(1),Ntrials,EbNo,N_sym,estimacion,CFO);
%Alfa=N/N+0
grafica3=BER(1,:);
BER(2,:)=SEFDM_awgn(N,m,alfa(2),Ntrials,EbNo,N_sym,estimacion,CFO);
%Alfa=N/N+2
grafica4=BER(2,:);

```

Segmento de código 2.30. Obtención de gráficas estimación perfecta y estimación con técnica de estimación CFO.

Obtenidos los parámetros para las gráficas, se procede a graficar cada escenario de simulación para realizar la comparación. La gráfica BER vs Eb/No muestra la relación del BER en cada escenario de simulación, dependiendo del orden de modulación que se indica en los parámetros de simulación del script principal. En la gráfica BER vs Eb/No se tiene 4 curvas pertenecientes a los escenarios de simulación diferenciados con un estilo de línea en las curvas. La gráfica final se observa en el segmento de código 2.31.

```

%GRAFICA BER vs Eb/No EN SEFDM
%Comparacion grafica estimacion perfecta con estimacion a eleccion
figure;
semilogy(EbNo,grafica1,'--','Color','m','LineWidth',1.2); grid on;
hold on;
semilogy(EbNo,grafica2,'-d','Color','k','LineWidth',1.2); grid on;
semilogy(EbNo,grafica3,'--','Color','c','LineWidth',1.2); grid on;
semilogy(EbNo,grafica4,'-d','Color','r','LineWidth',1.2); grid on;
axis([0 30 10^-4 1]);
title({'BER vs EbNo';modulaciones}); %Título gráfica general

```

Segmento de código 2.31. Gráfica BER vs Eb/No de los diferentes escenarios de simulación.

3. RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo, se presentan los resultados obtenidos de la simulación del sistema SEFDM con estimación CFO empleando un canal AWGN. Los resultados se presentan en gráficas BER vs Eb/No con variaciones en: la técnica de modulación, valor de CFO, valores en el factor de compresión de ancho de banda y técnica de estimación CFO. Con las variaciones en los parámetros de simulación presentes en las gráficas, se muestra la influencia que tiene la estimación CFO en el rendimiento del sistema SEFDM al exponerlo a distintos escenarios: uno con técnica de estimación perfecta y otro con una técnica específica de estimación CFO (CP o Preámbulo).

3.1. ESCENARIOS DE SIMULACIÓN

Dentro de los escenarios de simulación, se ocupa varios ejemplos en la presentación de los resultados en las gráficas BER vs Eb/No. En las gráficas, se modifican algunos parámetros de simulación, como se observa en la Tabla 3.1, para realizar una comparación de eficiencia y rendimiento entre los resultados obtenidos de la simulación de las técnicas utilizadas de estimación CFO exponiéndolas a distintos escenarios de simulación.

Tabla 3.1. Parámetros comunes de configuración en los escenarios de simulación.

Parámetro	Valor	Descripción
N	16	Número de subportadoras
N_{trials}	20000	Número de iteraciones para cada valor de SNR
N_{sym}	5	Número de símbolos SEFDM por paquete
tc	0.5	Tasa de codificación
Ng	4	Tamaño de prefijo cíclico
$pream$	40	Tamaño del preámbulo con forma según Figura 2.2

3.1.1. PRUEBAS DE SIMULACIÓN EN LAS 3 TÉCNICAS DE ESTIMACIÓN CON VARIOS VALORES DE α Y UN VALOR DE CFO.

En la Figura 3.1, se tiene la simulación de las 3 técnicas de estimación con valores distintos de α : 0.89, 0.84, 0.8. Como se observa en la figura, el rendimiento de la técnica de estimación con preámbulo (línea morada) es menor en comparación con la estimación ocupando CP (línea azul), ya que, se obtiene valores un poco más altos del BER en la

estimación con preámbulo. Sin embargo, el rendimiento de estas dos técnicas es menor en comparación con la estimación perfecta (línea roja) porque en esta no se tiene una afectación de CFO que degrade el BER.

También se observa en la Figura 3.1 que, al aumentar la compresión de ancho de banda, es decir, el valor de α disminuye, existe una degradación en el BER, ya que, el valor del BER aumenta en cada una de las técnicas de estimación. Este comportamiento es característico del sistema SEFDM y se observa en la Tabla 3.2 donde, para el análisis se toma el valor de $\frac{Eb}{No} = 10 \text{ dB}$ en las curvas de estimación CFO con preámbulo y se ve como el BER aumenta a medida que el α disminuye.

Tabla 3.2. Valores del BER en curvas de estimación con preámbulo con un valor de $Eb/No = 10 \text{ dB}$.

Valores de α	Valor aproximado del BER
0.89	0.01
0.84	0.0158
0.80	0.1

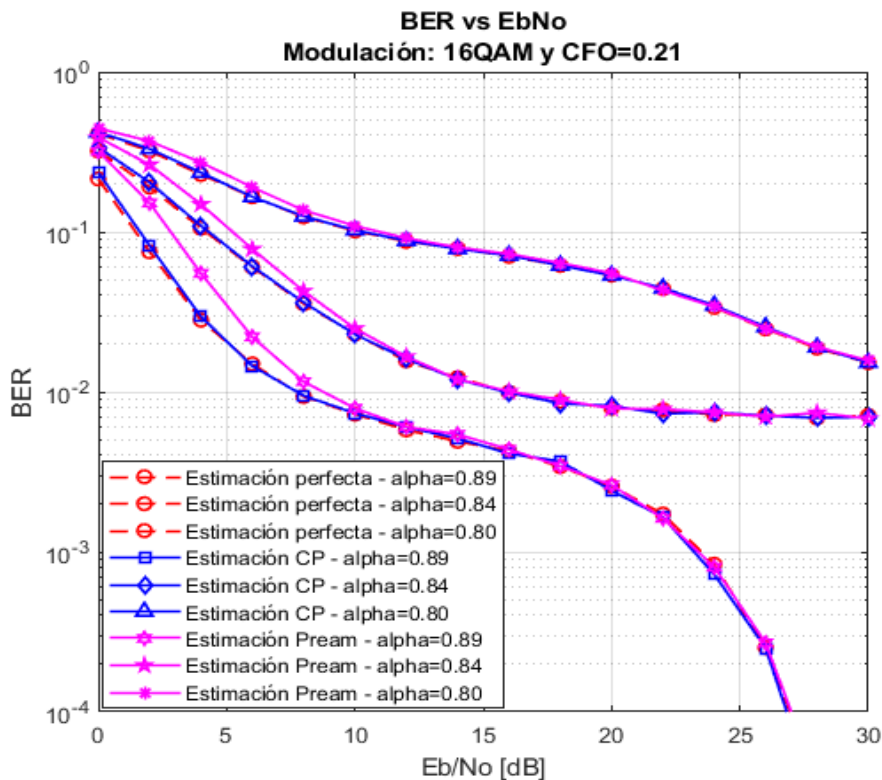


Figura 3.1. BER vs Eb/No 3 técnicas de estimación, distintos valores de α y $CFO = 0.21$ con modulación 16QAM

3.1.2. PRUEBAS DE SIMULACIÓN EN LAS 3 TÉCNICAS DE ESTIMACIÓN CON UN VALOR DE α Y CFO.

Con estas pruebas se busca comparar los resultados de las 3 técnicas de estimación, como se observa en la Figura 3.2, el escenario de simulación ocupa un $\alpha = 0.62$ y $CFO = 0.1$ con modulación QPSK. Sin embargo, para este tipo de modulación, se modifica el intervalo de Eb/No para obtener una gráfica adecuada para este tipo de modulación que es más robusta.

En la Figura 3.2, se observa que al aumentar el valor de Eb/No, en las 3 técnicas de estimación disminuye el valor del BER y el rendimiento de estas va asemejándose hasta el punto de que las 3 curvas coinciden. Sin embargo, en los primeros valores de Eb/No, la técnica de estimación con preámbulo presenta menor rendimiento, ya que, posee un valor del BER mayor en relación con las otras 2 técnicas de estimación.

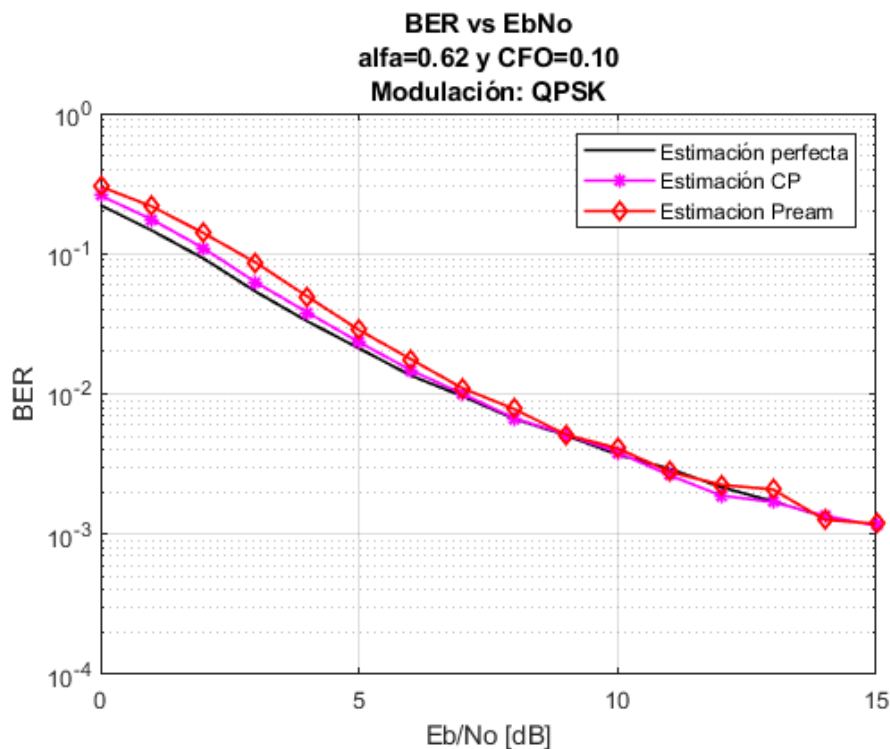


Figura 3.2. BER vs Eb/No 3 técnicas de estimación, $\alpha = 0.62$ y $CFO = 0.1$ con modulación QPSK.

Además, en la Figura 3.3 se presenta otro ejemplo de simulación de las 3 técnicas de estimación para realizar la comparación exponiéndolas a ciertos parámetros de simulación. En este caso, se tiene: $\alpha = 0.89$ y $CFO = 0.21$ con modulación 64QAM. Con esta figura, se demuestra que la simulación de las estimaciones CFO funcionan correctamente, ya que, al igual que en la Figura 3.2, las curvas de estas estimaciones van acercándose a la curva

de estimación perfecta a medida que avanza el valor de E_b/N_0 . Se muestra en la Tabla 3.3, como los valores del BER se asemejan en cada una de las técnicas de estimación a medida que avanza las curvas.

Tabla 3.3. Valores del BER en curvas de las técnicas de estimación con valor de $\alpha = 0.89$

Valor de E_b/N_0 (dB)	Valor aproximado del BER en estimación CP	Valores del BER en estimación preámbulo	Valores del BER en estimación perfecta
5	0.12589	0.14125	0.12882
15	0.1	0.1	0.1
30	0.001	0.001	0.001

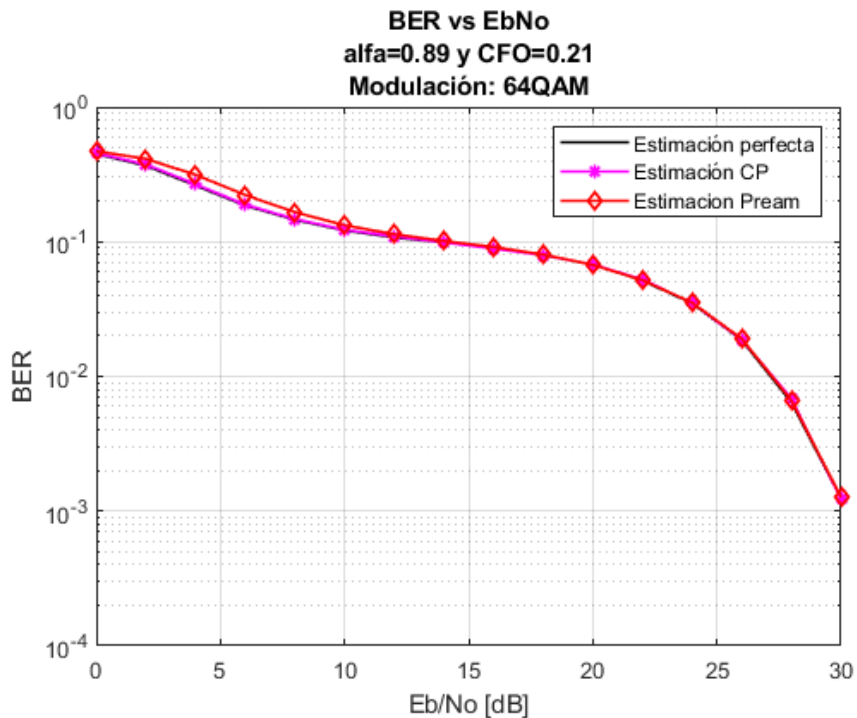


Figura 3.3. BER vs E_b/N_0 3 técnicas de estimación, $\alpha = 0.89$ y $CFO = 0.21$ con modulación 64QAM.

3.1.3. PRUEBAS DE SIMULACIÓN EN LAS 2 TÉCNICAS DE ESTIMACIÓN CFO CON VARIOS VALORES DE α Y UN VALOR DE CFO.

En las anteriores pruebas de simulación, se relaciona las 3 técnicas de estimación. Sin embargo, como se observa en la Figura 3.4, se busca analizar las dos técnicas de estimación CFO para ver cuál presenta mejores resultados frente a los parámetros de simulación que se especifican con valores de α : 0.94, 0.89, 0.84.

En la Figura 3.4, se observa que las curvas de estimación CP (línea verde) presentan mejores resultados al comparar valores del BER con respecto a las curvas de estimación preámbulo (línea azul). Además, al comparar las dos estimaciones con un valor de $\alpha = 0.94$, se tiene una mayor diferencia en el rendimiento de la estimación CP en comparación con la estimación con preámbulo. Al analizar los otros valores de α en las curvas de estimaciones CFO, se observa que no existe tanta diferencia en los valores del BER. Como se muestra en la Tabla 3.4, los valores del BER no presentan mayor diferencia para las curvas de las estimaciones CFO en valores de α más bajos, mostrando así en la Figura 3.4 cómo las curvas de estas estimaciones se asemejan más con los valores de α de 0.89 y 0.84.

Tabla 3.4. Valores del BER en curvas de estimación CP y preámbulo con valores α y $E_b/N_0 = 25 \text{ dB}$.

Valor de α	Valor aproximado del BER en estimación CP	Valore aproximado del BER en estimación preámbulo
0.89	0.000199	0.000251
0.84	0.00398	0.00398

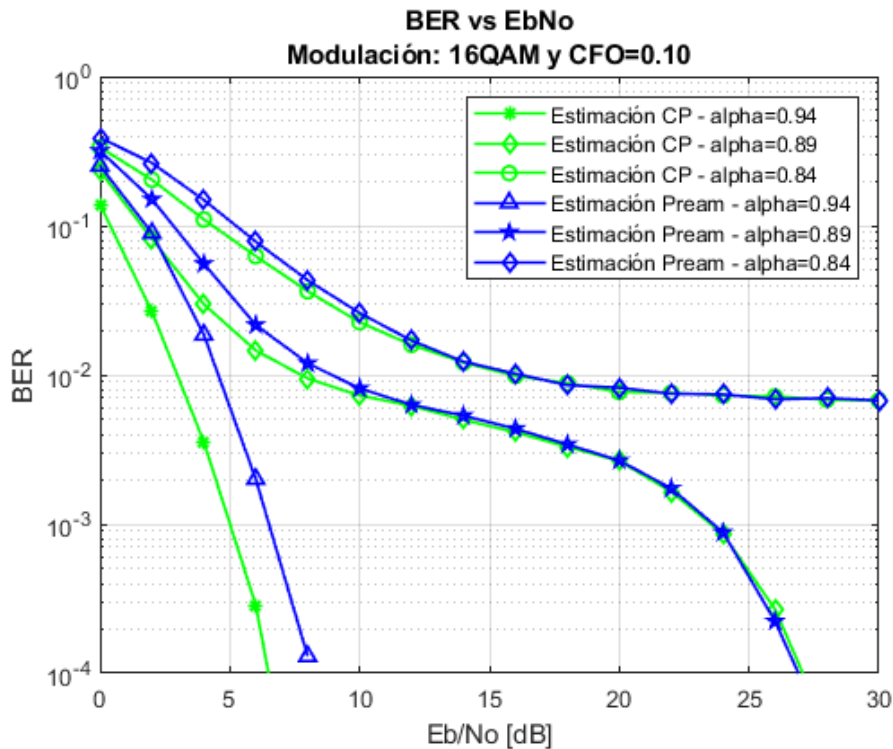


Figura 3.4. BER vs E_b/N_0 2 técnicas de estimación, $CFO=0.1$ con varios valores de α con modulación 16QAM.

3.1.4. PRUEBAS DE SIMULACIÓN EN LAS 2 TÉCNICAS DE ESTIMACIÓN CFO CON UN VALOR DE α Y DISTINTOS VALORES DE CFO.

En el escenario final de simulación, se demuestra que la aplicación de las técnicas de estimación CFO funciona correctamente, incluso al simular con varios valores de CFO. En la Figura 3.5 se simula la técnica de estimación CFO con preámbulo con un $\alpha = 0.84$ donde, se observa que para cada valor de CFO las curvas se asemejan y al aumentar el valor de E_b/N_0 , el valor del BER disminuye.

La misma situación se observa en la Figura 3.6, utilizando la técnica de estimación con CP donde, las curvas de cada valor de CFO son prácticamente iguales a lo largo de los valores de E_b/N_0 , sin importar que se haya cambiado el orden de modulación y valor del α .

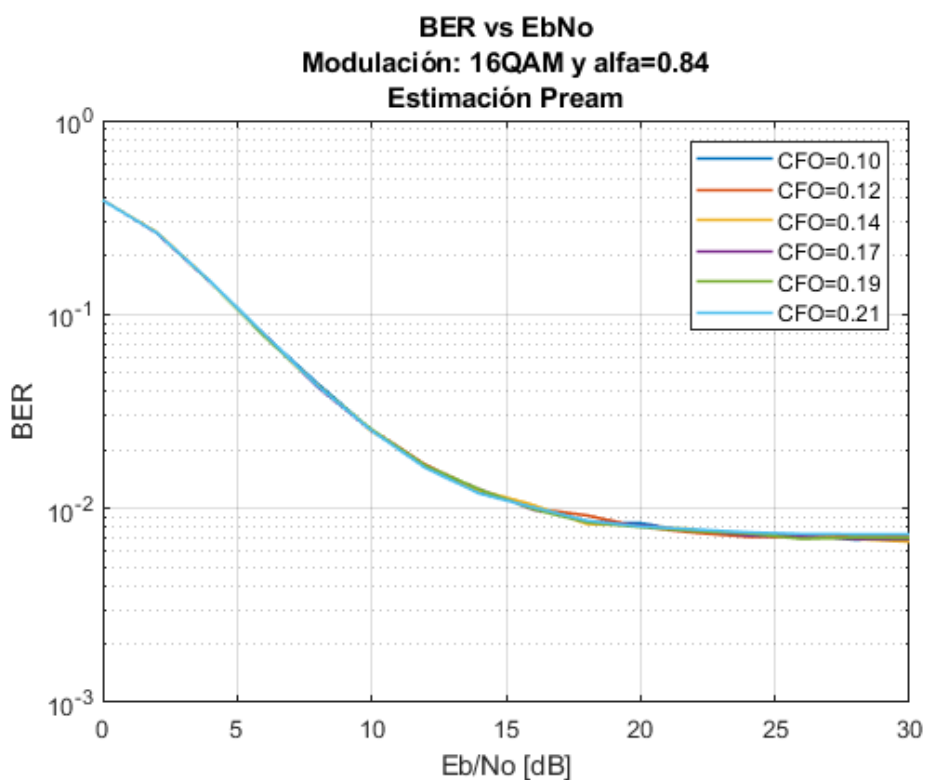


Figura 3.5. BER vs E_b/N_0 técnica de estimación con preámbulo, $\alpha = 0.84$ y varios valores de CFO con modulación 16QAM.

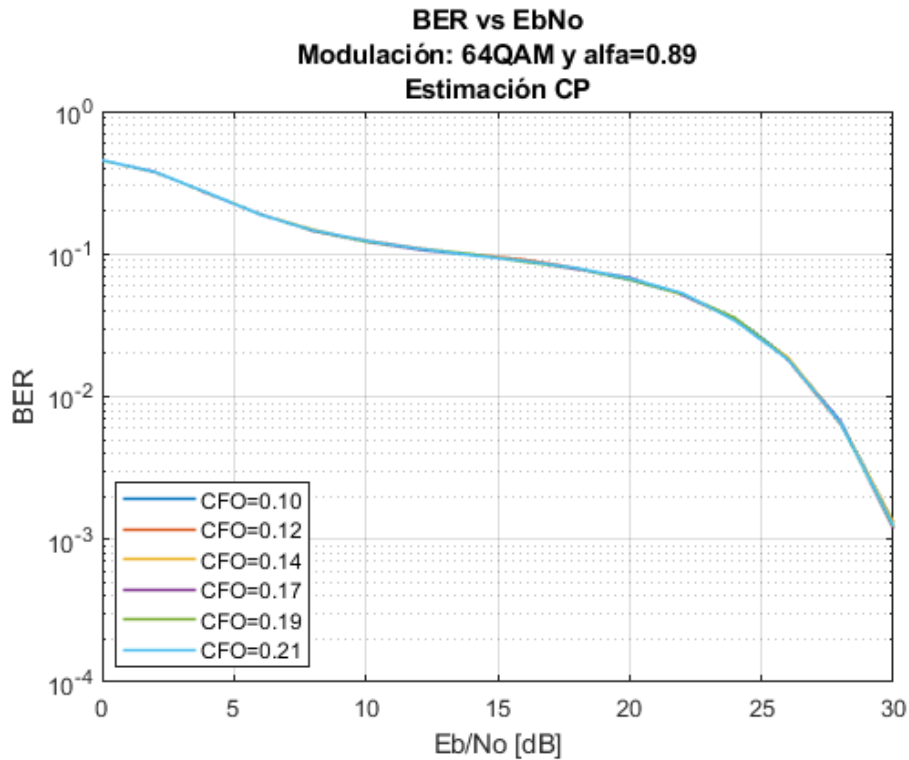


Figura 3.6. BER vs Eb/No técnica de estimación con CP, $\alpha = 0.89$ y varios valores de CFO con modulación 64QAM.

3.2. CONCLUSIONES

Al desarrollar el presente proyecto, se simuló un sistema SEFDM con estimación CFO a través de un canal AWGN. Como valores fijos, se tiene 16 subportadoras y 5 símbolos SEFDM. Para la simulación de estimación CFO, se trabajó con dos técnicas de estimación en el dominio del tiempo (CP y preámbulo), para mejorar cada estimación se realizó un respectivo promedio: en la técnica de estimación con CP se promedió el resultado de los 5 símbolos SEFDM y en la técnica con preámbulo se promedió el resultado de las dos partes del mismo. Sin embargo, para presentar una mejora en las curvas de las gráficas BER vs Eb/No, se realizó simulaciones con 20000 iteraciones, ya que, con iteraciones menor a 10000 las curvas no eran bien definidas. Además, en las simulaciones con modulación QPSK, se modificó el intervalo de Eb/No, ya que, las curvas presentadas en esta técnica no llegaban a más 15 dB.

Los escenarios de simulación se desarrollaron con variaciones de parámetros como la técnica de modulación, factor de compresión de ancho de banda y valor de CFO para el análisis de rendimiento de las técnicas. En cada uno de los escenarios de simulación, se

observó que el rendimiento de la estimación con preámbulo es menor en comparación con la estimación con CP en los valores iniciales de E_b/N_0 (entre 0 dB y 7 dB), ya que, al avanzar la curva, las 2 técnicas van asemejándose hasta el punto que coinciden en un mismo valor del BER. Esto demuestra la efectividad de la aplicación de las técnicas de estimación CFO en el sistema SEFDM porque las curvas de las estimaciones se aproximan a la curva de estimación perfecta ($CFO = 0$), sin importar que valor de CFO y α que se ingrese.

Por otra parte, se observó la característica del sistema SEFDM el cual, al aumentar la compresión de ancho de banda, es decir, el α disminuye, existe una degradación en el BER, ya que, en los escenarios de simulación con las 3 técnicas de estimación con valores de α menores, existía un valor mayor del BER, el cual iba bajando cuando se tenía un valor mayor de α . Este comportamiento depende de la modulación que se utilice, ya que, los valores del BER a un determinado α variaban dependiendo de la modulación.

En las simulaciones donde, se varió solo el valor de CFO para ambas técnicas de estimación CFO, se encontró que la diferencia es mínima para cada curva con un valor específico de CFO. El valor de BER va mejorando respecto avanza el valor de E_b/N_0 , pero, no existe mayor diferencia al ocupar un valor de CFO en cada técnica, ya que, las curvas son iguales a lo largo de la simulación.

El análisis final de los resultados de cada escenario de simulación con las respectivas variaciones en los parámetros, muestra que existe un mejor rendimiento de la técnica de estimación con CP en comparación con la técnica de estimación con preámbulo, pero, esto solo en valores pequeños de E_b/N_0 , ya que, el rendimiento de estas dos técnicas va mejorando a la par que los valores de E_b/N_0 aumentan hasta el punto que las dos curvas prácticamente son iguales y presentan un valor de BER adecuado.

3.3. RECOMENDACIONES

Tener un número mínimo de iteraciones para la simulación, ya que, al ocupar pocas iteraciones las curvas no tendrán una buena presentación en el resultado final. Sin embargo, al ingresar un número mayor de iteraciones, por ejemplo, mayor a 25000 la simulación se tardará un tiempo considerable.

Establecer un determinado valor de α aceptable, dependiendo la técnica de modulación que se ocupa, para evitar que las curvas muestren una simulación errada con valores de BER altos. Además, verificar el intervalo correcto de E_b/N_0 para que las curvas abarquen en su totalidad el intervalo dentro de la simulación.

Como idea base para un futuro trabajo de titulación, consiste en la simulación de un sistema SEFDM usando un canal Rayleigh y empleando técnicas de estimación CFO en dominio de la frecuencia donde, se ocupe diferentes valores de α y CFO, ya que, el presente proyecto ocupa un canal AWGN con técnicas de estimación CFO en el dominio del tiempo.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] T. Xu y I. Darwazeh, "Transmission Experiment of Bandwidth Compressed Carrier Aggregation in a Realistic Fading Channel," *IEEE Transactions on Vehicular Technology*, vol. 66, nº 5, pp. 4087-4097, 2017.
- [2] A. Rashich, A. Kislitsyn, D. Fadeev y T. N. Nguyen, "FFT-Based Trellis Receiver for SEFDM Signals," *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6, 2016.
- [3] D. Vasilyev y A. Rashich, "SEFDM-signals Euclidean Distance Analysis," *IEEE International Conference on Electrical Engineering and Photonics (EExPolytech)*, 2018.
- [4] S. Mohseni y M. Matin, "Study of the sensitivity of the OFDM wireless systems to the carrier frequency offset (CFO)," *International Journal of Distributed and Parallel Systems (IJDPS)*, vol. 4, 2013.
- [5] P. K. Nishad y P. Singh, "Carrier frequency offset estimation in OFDM systems," *IEEE Conference on Information & Communication Technologies*, pp. 885-889, 2013.
- [6] I. Darwazeh, H. Ghannam y T. Xu, "The First 15 Years of SEFDM: A Brief Survey," *11th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*, pp. 1-7, 2018.
- [7] W. Ozan, "Non-Orthogonal Signal and System Design for Wireless Communications," Ph.D. dissertation, Dept. Electron. Elct. Eng., University College London, 2020.
- [8] K. B. Toromoreno, Simulación de técnicas de sincronización en sistemas OFDM, Quito: Escuela Politécnica Nacional, 2020.
- [9] S. Pupiales , Análisis de ganancia de ancho de banda y degradación de la BER en un sistema SEFDM con respecto a OFDM usando MATLAB, Quito: Escuela Politécnica Nacional, 2020.
- [10] G. A. Logacho, Simulación en Matlab de los detectores ZF, MMSE e ID para un sistema SEFDM utilizando un canal con desvanecimiento, Quito: Escuela Politécnica Nacional, 2021.

- [11] T. Xu y I. Darwazeh , “Spectrally Efficient FDM: Spectrum Saving Technique for 5G?,” *1st International Conference on 5G for Ubiquitous Connectivity*, pp. 273-278, 2014.
- [12] M. R. Perrett, P. N. Whatmough, S. Isam y I. Darwazeh, “VLSI Architecture for a Reconfigurable Spectrally Efficient FDM Baseband Transmitter,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, n° 5, pp. 1107-1118, 2012.
- [13] M. Jia, Z. Yin, Q. Guo, G. Liu y X. Gu, “Downlink Design for Spectrum Efficient IoT Network,” *IEEE Internet of Things Journal*, vol. 5, n° 5, pp. 3397-3404, 2018.
- [14] R. C. Grammenos, S. Isam y I. Darwazeh, “FPGA design of a truncated SVD based receiver for the detection of SEFDM signals,” *IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 2085-2090, 2011.
- [15] T. Xu y . I. Darwazeh, “Bandwidth compressed carrier aggregation,” *IEEE International Conference on Communication Workshop (ICCW)*, pp. 1107-1112, 2015.
- [16] H. Ghannam y I. Darwazeh, “SEFDM: Spectral Efficiency Upper Bound and Interference Distribution,” *11th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP)*, pp. 1-6, 2018.
- [17] Y. S. Cho, J. Kim, W. Y. Yang y C. G. Kang, *MIMO-OFDM Wireless Communications with MATLAB*, Asia: John Wiley & Sons, 2010.

5. ANEXOS

ANEXO I

Script principal y funciones empleadas para el desarrollo de la simulación en Matlab (Anexo digital).