



La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

**EVALUACIÓN DEL RENDIMIENTO DE LA TÉCNICA OPS-SAP PARA LA
REDUCCIÓN DE LA PAPR EN ESTÁNDARES DE REDES AD-HOC**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
EN ELECTRÓNICA Y REDES DE INFORMACIÓN**

ALEXANDER JAVIER ACERO ARIAS

alexander.acero@epn.edu.ec

DIRECTOR: ING. MARTHA CECILIA PAREDES PAREDES. PHD

cecilia.paredes@epn.edu.ec

Quito, Agosto 2020

AVAL

Certifico que el presente trabajo fue desarrollado por ALEXANDER JAVIER ACERO ARIAS, bajo mi supervisión.

Ph.D. Martha Cecilia Paredes Paredes
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Alexander Javier Acero Arias, declaro bajo juramento que el trabajo 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.

A través de la presente declaración dejo constancia de que la Escuela Politécnica Nacional podrá hacer uso del presente trabajo según los términos estipulados en la Ley, Reglamentos y Normas vigentes.

ALEXANDER JAVIER ACERO ARIAS

DEDICATORIA

Le dedico a mi querida y amada madre quien siempre me brinda su afecto, paciencia y amor incondicional.

AGRADECIMIENTO

Agradezco a Dios por la vida, a mi mamita por su guía y sabiduría, quien pese a todas las dificultades siempre me apoya y me brinda su amor abnegado, a mi padre por su apoyo, y a mi familia por apoyarme incondicionalmente.

Agradezco a: Byron, David, Alejandra y Andres por su amistad incondicional y apoyo a lo largo de esta aventura que fue la poli, son y siempre serán muy importantes para mí, su amistad, cariño y lealtad son los más grandes regalos que atesoro con mucho cariño.

A todas las personas que me brindaron su amistad y apoyo a lo largo de la carrera: Alex, Victor, Juan, Andrea, Vico, Sandra, Giss, Fabo, Karla... muchas gracias por sus consejos, tiempo y sabiduría.

A mis profesores, grandes profesionales por compartir sus conocimientos y formarme como un profesional a carta cabal.

Finalmente, quiero mostrar mi más profundo sentimiento de agradecimiento a mi directora Dra. Cecilia Paredes quien me brindo su guía, impulso, comprensión y paciencia para poder realizar y culminar con éxito la tesis.

ÍNDICE DE CONTENIDO

| | |
|--|------|
| AVAL..... | I |
| DECLARACIÓN DE AUTORÍA | II |
| DEDICATORIA | III |
| AGRADECIMIENTO | IV |
| ÍNDICE DE CONTENIDO..... | V |
| ÍNDICE DE FIGURAS | X |
| ÍNDICE DE TABLAS | XIII |
| RESUMEN | XIV |
| ABSTRACT..... | XV |
| 1. INTRODUCCIÓN..... | 1 |
| 1.1 OBJETIVOS..... | 2 |
| 1.2 ALCANCE..... | 2 |
| 1.3 MARCO TEÓRICO | 4 |
| 1.3.1. REDES AD-HOC | 4 |
| 1.3.2. ESTANDAR IEEE 802.11a/g/p | 5 |
| 1.3.2.1. IEEE 802.11a..... | 5 |
| 1.3.2.2. IEEE 802.11g..... | 5 |
| 1.3.2.3. IEEE 802.11p..... | 6 |
| 1.3.3. COMPARACIÓN ENTRE LOS ESTÁNDARES IEEE 802.11a/g/p | 6 |
| 1.3.4. CAPA FÍSICA DE LOS ESTÁNDARES IEEE 802.11a/g/p | 6 |
| 1.3.4.1. Subcapa PLME | 7 |
| 1.3.4.2. Subcapa PLCP | 7 |
| 1.3.4.3. Subcapa PMD..... | 8 |
| 1.3.5. AMPLIFICADOR DE ALTA POTENCIA | 19 |
| 1.3.6. PROBLEMA DE LA PAPR ELEVADA EN SISTEMAS OFDM..... | 20 |
| 1.3.6.1. Definición de la PAPR..... | 21 |
| 1.3.6.2. Método OPS-SAP para la reducción de la PAPR..... | 21 |
| 1.3.7. MÉTRICAS PARA LA EVALUACIÓN DE LA PAPR..... | 26 |
| 1.3.7.1. Complementary Cumulative Distribution | 26 |

| | |
|--|----|
| 1.3.7.2. Power Spectral Density (PSD) | 27 |
| 1.3.7.3. Bit Error Rate (BER) | 28 |
| 1.3.8. CANAL INALÁMBRICO | 29 |
| 1.3.8.1. Desvanecimiento | 30 |
| 1.3.8.2. Modelos de Propagación a pequeña escala | 31 |
| 2. METODOLOGÍA..... | 33 |
| 2.1. MATLAB | 33 |
| 2.1.1. APP DESIGNER | 35 |
| 2.1.2. ENTORNO DE TRABAJO DE APP DESIGNER | 35 |
| 2.2. APLICACIÓN | 37 |
| 2.2.1. DISEÑO | 38 |
| 2.2.2. FUNCIONAMIENTO | 39 |
| 2.3. CALLBACK PRINCIPAL | 39 |
| 2.4. TRANSMISOR..... | 40 |
| 2.4.1. FUENTE | 40 |
| 2.4.2. CREACIÓN DEL PREAMBULO..... | 40 |
| 2.4.2.1. Preámbulo Corto..... | 40 |
| 2.4.2.2. Preámbulo Largo | 41 |
| 2.4.3. CREACIÓN DE LA CABECERA | 41 |
| 2.4.4. CREACIÓN DEL CAMPO DATA..... | 41 |
| 2.4.5. ALEATORIZADOR..... | 42 |
| 2.4.6. CODIFICACIÓN..... | 43 |
| 2.4.7. ENTRELAZADO | 44 |
| 2.4.8. MODULACIÓN | 44 |
| 2.4.9. BLOQUE OFDM | 46 |
| 2.4.10. AMPLIFICADOR DE ALTA POTENCIA HPA | 48 |
| 2.5. CANAL INALÁMBRICO | 48 |
| 2.5.1. CANAL RAYLEIGH..... | 49 |
| 2.5.2. CANAL Rician..... | 51 |
| 2.6. RECEPTOR..... | 52 |
| 2.6.1. OBTENCIÓN DEL CAMPO DATA | 52 |
| 2.6.2. BLOQUE OFDM EN RECEPCIÓN | 52 |
| 2.6.3. DEMODULACIÓN DE DATOS..... | 53 |
| 2.6.4. DEENTRELAZADO | 55 |

| | | |
|----------|---|----|
| 2.6.5. | DECODIFICADOR CONVOLUCIONAL | 56 |
| 2.6.6. | DESALEATORIZADOR | 56 |
| 2.7. | REDUCCIÓN DE LA PAPR | 57 |
| 2.7.1. | PAPR..... | 57 |
| 2.7.2. | TÉCNICA OPS-SAP | 58 |
| 2.8. | TRATAMIENTO DE DATOS..... | 61 |
| 2.8.1. | BER | 61 |
| 2.8.2. | CCDF DE LA PAPR..... | 62 |
| 2.8.3. | PSD | 63 |
| 2.9. | PRESENTACIÓN DE RESULTADOS..... | 64 |
| 3. | RESULTADOS Y DISCUSIÓN | 65 |
| 3.1. | PARÁMETROS PARA LOS ESCENARIOS DE SIMULACIÓN | 65 |
| 3.2. | PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE COMUNICACIÓN AD-HOC. | 68 |
| 3.2.1. | FUNCIONAMIENTO DE LA TÉCNICA OPS-SAP | 69 |
| 3.3. | RESULTADOS DEL SISTEMA DE COMUNICACIÓN IEEE 802.11a/g/p..... | 69 |
| 3.3.1. | ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 1/2..... | 71 |
| 3.3.2. | ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 1/2. | 73 |
| 3.3.2.1. | BER vs SNR | 73 |
| 3.3.2.2. | CCDF de la PAPR | 73 |
| 3.3.2.3. | PSD | 73 |
| 3.3.3. | ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 3/4..... | 74 |
| 3.3.4. | ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 3/4. | 76 |
| 3.3.4.1. | BER vs SNR | 76 |
| 3.3.4.2. | CCDF de la PAPR | 76 |
| 3.3.4.3. | PSD | 76 |
| 3.3.5. | ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 1/2..... | 77 |
| 3.3.6. | ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 1/2..... | 79 |
| 3.3.6.1. | BER vs SNR | 79 |

| | |
|--|----|
| 3.3.6.2. CCDF de la PAPR | 79 |
| 3.3.6.3. PSD | 79 |
| 3.3.7. ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 3/4..... | 80 |
| 3.3.8. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 3/4..... | 82 |
| 3.3.8.1. BER vs SNR | 82 |
| 3.3.8.2. CCDF de la PAPR | 82 |
| 3.3.8.3. PSD | 82 |
| 3.3.9. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 1/2..... | 83 |
| 3.3.10. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 16- QAM Y TASA DE CODIFICACIÓN 1/2. | 85 |
| 3.3.10.1. BER vs SNR..... | 85 |
| 3.3.10.2. CCDF de la PAPR..... | 85 |
| 3.3.10.3. PSD..... | 85 |
| 3.3.11. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 3/4..... | 86 |
| 3.3.12. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 16- QAM Y TASA DE CODIFICACIÓN 3/4. | 88 |
| 3.3.12.1. BER vs SNR..... | 88 |
| 3.3.12.2. CCDF de la PAPR..... | 88 |
| 3.3.12.3. PSD..... | 89 |
| 3.3.13. ESQUEMA DE MODULACIÓN 64-QAM Y TASA DE CODIFICACIÓN 2/3..... | 89 |
| 3.3.14. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 64- QAM Y TASA DE CODIFICACIÓN 2/3. | 91 |
| 3.3.14.1. BER vs SNR..... | 91 |
| 3.3.14.2. CCDF de la PAPR..... | 92 |
| 3.3.14.3. PSD..... | 92 |
| 3.3.15. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 3/4..... | 92 |
| 3.3.16. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 64- QAM Y TASA DE CODIFICACIÓN 3/4. | 94 |
| 3.3.16.1. BER vs SNR..... | 94 |
| 3.3.16.2. CCDF de la PAPR..... | 95 |
| 3.3.16.3. PSD..... | 95 |

| | | |
|--------|--|-----|
| 3.4. | ANÁLISIS DE LOS RESULTADOS DEL SISTEMA DE COMUNICACIÓN IEEE 802.11a/g/p..... | 95 |
| 3.4.1. | ANÁLISIS DE LOS RESULTADOS BER VS SNR | 95 |
| 3.4.2. | ANÁLISIS DE LOS RESULTADOS DE LA CCDF DE LA PAPR | 96 |
| 3.4.3. | ANÁLISIS DE LOS RESULTADOS DE LA PSD | 96 |
| 4. | CONCLUSIONES Y RECOMENDACIONES | 98 |
| 4.1 | CONCLUSIONES | 98 |
| 4.2 | RECOMENDACIONES | 100 |
| 5. | REFERENCIAS BIBLIOGRÁFICAS..... | 101 |
| | ANEXOS..... | 106 |
| | ANEXO A Diagramas de Flujo | 107 |
| | ANEXO B Códigos y Funciones..... | 144 |
| | ANEXO C Manual de Usuario | 161 |
| | ORDEN DE EMPASTADO..... | 168 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1.1 Modelo de comunicación OFDM | 3 |
| Figura 1.2 Subcapa PLCP | 7 |
| Figura 1.3 Diagrama del transmisor OFDM. | 9 |
| Figura 1.4 Índices en el proceso de entrelazado | 11 |
| Figura 1.5 Diagrama de constelación de una modulación B-PSK..... | 13 |
| Figura 1.6 Diagrama de constelación de una modulación Q-PSK | 14 |
| Figura 1.7 Diagrama de constelación de una modulación 16-QAM | 14 |
| Figura 1.8 Diagrama de constelación de una modulación 64-QAM. | 15 |
| Figura 1.9 Operación IFFT en una señal OFDM..... | 17 |
| Figura 1.10 Inserción del prefijo cíclico en el símbolo OFDM | 18 |
| Figura 1.11 Predistorsión de los puntos en 16-QAM | 22 |
| Figura 1.12 Diagrama de bloques de la técnica OPS en el dominio de la frecuencia. | 25 |
| Figura 1.13 Diagrama de la técnica OPS-SAP | 26 |
| Figura 1.14 CCDF de la PAPR para un esquema de modulación BPSK y tasa de codificación $r = 1/2$ | 27 |
| Figura 1.15 PSD de una señal OFDM | 28 |
| Figura 1.16 BER vs SNR..... | 29 |
| Figura 2.1 Ícono New con la opción App. | 36 |
| Figura 2.2 Ventana correspondiente a DESIGNER | 36 |
| Figura 2.3 Ventana perteneciente al EDITOR | 37 |
| Figura 2.4 Diseño visual de la aplicación..... | 38 |
| Figura 2.5 Proceso que lleva a cabo el callback principal..... | 40 |
| Figura 3.1 Resultados con los valores de L y α | 67 |
| Figura 3.2 Curvas teóricas y simuladas de los diferentes esquemas de modulación | 68 |
| Figura 3.3 BER vs SNR para una modulación BPSK y tasa de codificación $r =$ $1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHZ). | 71 |
| Figura 3.4 CCDF de la PAPR con modulación BPSK y tasa de codificación $r =$ $1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHZ). | 71 |

| | |
|--|----|
| Figura 3.5 PSD con modulación BPSK tasa de codificación $r = 1/2$ y 20MHz... | 72 |
| Figura 3.6 PSD con modulación BPSK tasa de codificación $r = 1/2$ y 10MHz... | 72 |
| Figura 3.7 BER vs SNR para una modulación BPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 74 |
| Figura 3.8 CCDF de la PAPR con modulación BPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 74 |
| Figura 3.9 PSD con modulación BPSK, tasa de codificación $r = 3/4$ y 20MHz.. | 75 |
| Figura 3.10 PSD con modulación BPSK, tasa de codificación $r = 3/4$ y 10MHz. | 75 |
| Figura 3.11 BER vs SNR para una modulación QPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).. | 77 |
| Figura 3.12 CCDF de la PAPR con modulación QPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 77 |
| Figura 3.13 PSD con modulación QPSK, tasa de codificación $r = 1/2$ y 20MHz | 78 |
| Figura 3.14 PSD con modulación QPSK, tasa de codificación $r = 1/2$ y 10MHz. | 78 |
| Figura 3.15 BER vs SNR para una modulación QPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 80 |
| Figura 3.16 CCDF de la PAPR con modulación QPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 80 |
| Figura 3.17 PSD con modulación QPSK, tasa de codificación $r = 3/4$ y 20 MHz. | 81 |
| Figura 3.18 PSD con modulación QPSK, tasa de codificación $r = 3/4$ y 10 MHz. | 81 |
| Figura 3.19 BER vs SNR para una modulación 16-QAM y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 83 |
| Figura 3.20 CCDF de la PAPR con modulación 16-QAM y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 83 |
| Figura 3.21 PSD con modulación 16-QAM, tasa de codificación $r = 1/2$ y 20MHz. | 84 |
| Figura 3.22 PSD con modulación 16-QAM, tasa de codificación $r = 1/2$ y 10MHz | 84 |
| Figura 3.23 BER vs SNR para una modulación 16-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 86 |

| | |
|---|----|
| Figura 3.24 CCDF de la PAPR con modulación 16-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 87 |
| Figura 3.25 PSD con modulación 16-QAM, tasa de codificación $r = 3/4$ y 20 MHz. | 87 |
| Figura 3.26 PSD con modulación 16-QAM, tasa de codificación $r = 3/4$ y 10 MHz. | 88 |
| Figura 3.27 BER vs SNR para una modulación 64-QAM y tasa de codificación $r = 2/3$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz). | 89 |
| Figura 3.28 CCDF de la PAPR con modulación 64-QAM y tasa de codificación $r = 2/3$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 90 |
| Figura 3.29 PSD con modulación 64-QAM, tasa de codificación $r = 2/3$ y 20 MHz. | 90 |
| Figura 3.30 PSD con modulación 64-QAM, tasa de codificación $r = 2/3$ y 10 MHz. | 91 |
| Figura 3.31 BER vs SNR para una modulación 64-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz). | 92 |
| Figura 3.32 CCDF de la PAPR con modulación 64-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..... | 93 |
| Figura 3.33 PSD con modulación 64-QAM, tasa de codificación $r = 3/4$ y 20 MHz. | 93 |
| Figura 3.34 PSD con modulación 64-QAM, tasa de codificación $r = 3/4$ y 10 MHz. | 94 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1.1 Tasa de codificación y esquema de modulación de los diferentes estándares | 3 |
| Tabla 1.2 Diferencias entre los estándares IEEE 802.11a/g/p. | 6 |
| Tabla 1.3 Esquemas de modulación y tasas de codificación de acuerdo al estándar IEEE 802.11 | 15 |
| Tabla 1.4 Número de bits por cada símbolo en el proceso de demodulación. | 16 |
| Tabla 1.5 Valores de potencia promedio y retardo relativo para el estándar IEEE 802.11p..... | 32 |
| Tabla 2.1 Diferencias entre GUIDE y App Designer | 34 |
| Tabla 2.2 Valores de PDP para el estándar IEEE 802.11a. | 49 |
| Tabla 2.3 Valores de PDP para el estándar IEEE 802.11g. | 49 |
| Tabla 2.4 Valores de PDP para el estándar IEEE 802.11p. | 49 |
| Tabla 2.5 Nomenclatura para los valores de BER. | 62 |
| Tabla 2.6 Comandos para la creación de gráficas | 64 |
| Tabla 3.1 Escenarios de Simulación | 65 |
| Tabla 3.2 Parámetros para cada estándar IEEE 802.11a/g/p. | 66 |
| Tabla 3.3 Parámetros del HPA. | 66 |
| Tabla 3.4 Valores del PDP para cada estándar | 66 |
| Tabla 3.5 Parámetros de la técnica OPS-SAP..... | 67 |
| Tabla 3.6 Valores promedios de la PAPR para cada esquema de modulación.... | 69 |
| Tabla 3.7 Nomenclatura de las señales para las gráficas de BER vs SNR. | 69 |
| Tabla 3.8 Valores de la reducción de la radiación fuera de banda con respecto a la señal original de cada técnica para cada escenario de simulación. | 97 |

RESUMEN

Las redes inalámbricas Ad-Hoc constituyen una parte importante de las redes inalámbricas por su fácil implementación y flexibilidad, ya que su capa física PMD (*Physical Medium Depend*) se fundamenta en la técnica de multiplexación OFDM (*Orthogonal Frequency Division Multiplexing*). Sin embargo, en el lado del transmisor, la señal tiene una alta relación de potencia instantánea respecto a su potencia promedio, esto se conoce como el problema de la PAPR (*Peak-to-Average Power Ratio*). Este fenómeno ocasiona una degradación en la señal OFDM cuando atraviesa por un amplificador de potencia, ya que dicho dispositivo se satura. La saturación ocasiona que la señal rote, se atenúe y se desplace dando como resultado el aumento de la BER (*Bit Error Rate*). El rendimiento de la transmisión puede ser mejorado si se aplica algún mecanismo que permita reducir la PAPR de la señal. Entre las técnicas de reducción de la PAPR más prometedoras encontradas en la literatura se encuentra la técnica denominada OPS-SAP (*Orthogonal Pilot Sequences with Simple Amplitud Predistorsion*), puesto que esta técnica ha proporcionado buenos resultados en otros estudios referentes a comunicaciones inalámbricas.

El presente Trabajo de Titulación analiza el rendimiento de la técnica OPS-SAP en los estándares IEEE 802.11a/g/p, a través de simulaciones en MATLAB. En primer lugar, se realiza una breve descripción de: (1) la capa física OFDM de los estándares IEEE 802.11a/g/p (utilizados en redes Ad-Hoc), (2) el problema de la PAPR y (3) las técnicas de reducción de la PAPR OPS, SAP y OPS-SAP. A continuación. En el capítulo 2 se detalla el diseño e implementación en *App Designer* de Matlab del sistema de comunicación IEEE 802.11a/g/p para la transmisión de información a través de tres diferentes tipos de canales inalámbricos AWGN, Rayleigh y Rician y también con las técnicas de reducción de la PAPR OPS, SAP y OPS-SAP. El capítulo 3 presenta los resultados de las simulaciones para evaluar el rendimiento de estas técnicas con respecto a la señal original. Los resultados se presentan en términos de BER vs SNR (*Signal To Noise Ratio*), CCDF (*Complementary Cumulative Distribution Funcion*) de la PAPR y PSD (*Power Spectral Density*).

Las simulaciones demuestran la ganancia obtenida con la aplicación de la técnica OPS-SAP en redes Ad-Hoc. Dado que, en el mejor de los escenarios, la ganancia de PAPR es de aproximadamente 3.5 dB con una probabilidad de 10^{-3} con respecto a la señal sin técnica de reducción de la PAPR.

PALABRAS CLAVE: AD-HOC, OFDM, OPS-SAP, PAPR, APP DESIGNER.

ABSTRACT

Ad-Hoc wireless networks are an important part of wireless networks due to its easy implementation and flexibility, since it's Physical Medium Depend (PMD) layer is based on the OFDM (Orthogonal Frequency Division Multiplexing) multiplexing technique. However, on the transmitter side, the signal has a high instantaneous power to its average power ratio; this is known as the PAPR (Peak-to-Average Power Ratio) problem. This phenomenon causes a degradation in the OFDM signal when it passes through a power amplifier, since this device becomes saturated. This degradation causes rotate, attenuate, and shifting of the signal. Transmission performance can be improved if some mechanism is applied to reduce the PAPR of the signal. Among the most promising PAPR reduction techniques found in the literature is the technique called OPS-SAP (Orthogonal Pilot Sequences with Simple Amplitude Predistorsion), since this technique has provided good results in other studies regarding wireless communications.

This final career project analyzes the performance of the OPS-SAP technique in the IEEE 802.11a/g/p standards, through simulations in MATLAB. First, a brief description of the next aspects was carried out. (1) The OFDM physical layer of the IEEE 802.11a/g/p standards (used in Ad-Hoc networks), (2) the PAPR problem and (3) the reduction techniques of the PAPR OPS, SAP, and OPS-SAP. Next, Chapter 2 details the design and implementation in MATLAB App Designer of the IEEE 802.11a/g/p communication system for the information transmission through three different types of wireless channels AWGN, Rayleigh, and Rician and with OPS, SAP, and OPS-SAP PAPR reduction Techniques. Chapter 3 presents the results of the simulations and with OPS, SAP, and OPS-SAP PAPR reduction Techniques performance evaluation of the techniques with respect to the original signal. The results are presented in terms of BER vs SNR (*Signal To Noise Ratio*), CCDF (Complementary Cumulative Distribution Function) of the PAPR, and PSD (Power Spectral Density).

The simulations demonstrate the gain obtained with the OPS-SAP technique application in Ad-Hoc networks. Since, in the best of the scenarios, the PAPR gain is approximately 3.5 dB at a probability of 10^{-3} with respect to the signal without the PAPR reduction technique.

KEYWORDS: AD-HOC, OFDM, OPS-SAP, PAPR, APP DESIGNER

1. INTRODUCCIÓN

La tecnología inalámbrica en los últimos años va tomando una importancia relevante y en la actualidad es una parte fundamental de las diferentes redes de datos, tanto así que las redes inalámbricas complementan el funcionamiento de las redes alámbricas. Las redes Ad-Hoc son parte importante de las redes inalámbricas, estas redes no cuentan con una infraestructura existente y son descentralizadas facilitando así la interconectividad en cualquier lugar. Debido a la flexibilidad y adaptabilidad de estas redes, sumado al aumento de equipos portátiles, da como resultado su gran popularidad para trabajar con este tipo de redes.

Existen diferentes tipos de redes Ad-Hoc entre las cuales se tiene las siguientes: redes Ad-Hoc móviles o MANETS (*Mobile Ad-Hoc Networks*), redes inalámbricas malladas y redes de sensores WSN (*Wireless Sensor Networks*). Las redes MANET o malladas se fundamentan en algunos estándares de la norma IEEE 802.11 como IEEE 802.11a, IEEE 802.11g e IEEE 802.11p.

Los estándares antes mencionados fundamentan su capa física PMD en la multiplexación por división de frecuencias ortogonales o OFDM (*Orthogonal Frequency Division Multiplexing*) debido a las grandes ventajas que presenta esta técnica.

Pese a las diferentes ventajas presentes en OFDM, en el lado del transmisor, suelen ocurrir fenómenos que ocasionan distorsiones en la señal. Uno de los fenómenos es aquel que produce una alta relación de potencia instantánea respecto a su potencia promedio, este fenómeno es conocido como el problema de la PAPR (*Peak-to-Average Power Ratio*). Este problema consiste en la aparición de picos altos de potencia instantánea que se generan por la multiplexación, dichos picos introducen degradación en el rendimiento del HPA (*High Power Amplifier*), puesto que este dispositivo se satura, ocasionando que la señal rote, se atenúe y se desplace dando como resultado el aumento de la BER (*Bit Error Rate*) [1].

Para solucionar este problema se han desarrollado varias técnicas de reducción de la PAPR, entre las más destacadas por sus características se tiene OPS (*Orthogonal Pilot Sequence*), SAP (*Simple Amplitud Predistorsion*) y OPS-SAP que es el resultado del trabajo en conjunto de las dos técnicas anteriores [2].

Sin embargo, no se cuenta con documentación de la evaluación del rendimiento de la técnica OPS-SAP en redes Ad-Hoc, por lo que se considera que, esta investigación servirá para que los docentes, investigadores y estudiantes cuenten con un documento que presente resultados de la aplicación de esta técnica en ambientes de redes Ad-Hoc.

El presente trabajo de titulación presenta la implementación del transmisor y receptor según IEEE 802.11a/g/p, que a su vez se comunican a través de un canal inalámbrico con ruido AWGN, desvanecimiento Rician y Rayleigh, se implementara la técnica OPS-SAP dado en [1].

Para determinar el rendimiento de la técnica OPS-SAP se utilizarán las métricas CCDF (*Complementary Cumulative Distribution Funcion*) de la PAPR, PSD (*Power Spectral Density*) y BER vs SNR. Los resultados obtenidos de las diferentes simulaciones serán comparados con y sin la aplicación de la técnica OPS-SAP para determinar la ganancia que se obtiene al implementar dicha técnica.

Finalmente, todo el sistema se implementará en el ambiente de desarrollo perteneciente a *App Designer* que a su vez es parte del ambiente de trabajo de Matlab.

1.1 OBJETIVOS

El objetivo general de este proyecto técnico consiste en:

- Evaluar el rendimiento de la técnica OPS-SAP para la reducción de la PAPR en estándares de redes Ad-hoc.

Los objetivos específicos de este proyecto técnico son:

- Analizar la capa física OFDM del estándar IEEE 802.11a/g/p y la técnica OPS-SAP.
- Diseñar los bloques de la capa física OFDM del estándar IEEE 802.11a/g/p y la técnica OPS-SAP.
- Implementar una aplicación en el entorno de desarrollo perteneciente a App Designer.
- Analizar los resultados obtenidos en términos de curvas de CCDF de la PAPR, BER vs SNR y PSD para cada estándar.

1.2 ALCANCE

El proyecto de titulación entregará: una aplicación implementada en el entorno de desarrollo App Designer de MATLAB en la cual se ingresarán diferentes parámetros y mostrará a través de curvas CCDF de la PAPR, BER vs SNR y PSD la evaluación del rendimiento de la técnica de reducción de la PAPR en las enmiendas para redes Ad-hoc IEEE 802.11a/g/p. Por lo tanto, se implementará: (1) un modelo de comunicación (Transmisor y Receptor) IEEE 802.11 conectados por canales AWGN, Rayleigh y Rician y (2) la técnica de reducción de la PAPR OPS-SAP.

(1) Modelo de comunicación

La simulación consiste en la transmisión de una cadena de bits generados aleatoriamente, agrupados en paquetes de similar tamaño. El modelo de la capa física OFDM seguirá el proceso de formación y recepción de tramas a nivel de capa física descrita en el estándar IEEE 802.11[3] y se conectarán por un canal AWGN, Rician y Rayleigh (se implementarán todos los bloques descritos en la Figura 1.1 , además de otros procesos dentro del estándar como la creación y concatenación de preámbulo y cabecera, estos datos se pasarán en orden a través de un modulador que asigna a grupos de bits dentro del paquete, símbolos de acuerdo al diagrama de constelaciones utilizando, mapeo de símbolos OFDM para la asignación de cada símbolo modulado a la subportadora del símbolo OFDM respectiva, IFFT (*Inverse Fast Fourier Transform*) para pasar el símbolo al dominio del tiempo, la agregación de prefijo cíclico que añade una porción del mismo símbolo OFDM en tiempo al principio del mismo símbolo OFDM para combatir los efectos adversos del multitrayecto.

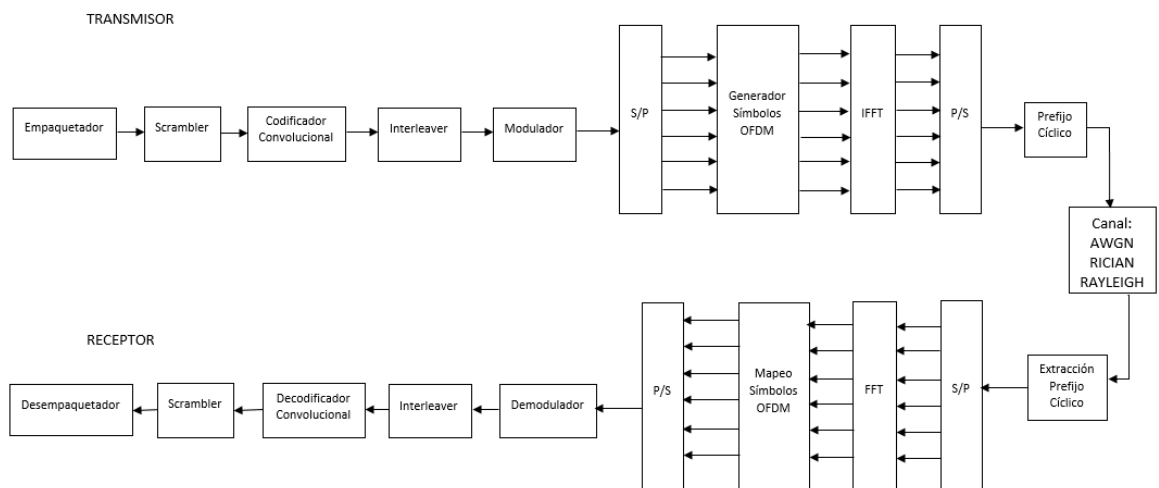


Figura 1.1 Modelo de comunicación OFDM

Las posibles combinaciones de tasa de codificación y esquema de modulación disponibles en el estándar IEEE 802.11 [3], se resumen en la Tabla 1.1.

Tabla 1.1 Tasa de codificación y esquema de modulación de los diferentes estándares

| Esquema de modulación | Tasa de codificación | Tasa de datos [Mbps] IEEE 802.11a | Tasa de datos [Mbps] IEEE 802.11g | Tasa de datos [Mbps] IEEE 802.11p |
|-----------------------|----------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| BPSK | 1/2 | 6 | 6 | 3 |
| BPSK | 3/4 | 9 | 9 | 4.5 |
| QPSK | 1/2 | 12 | 12 | 6 |

| | | | | |
|--------------|-----|----|----|----|
| QPSK | 3/4 | 18 | 18 | 9 |
| 16QAM | 1/2 | 24 | 24 | 12 |
| 16QAM | 3/4 | 36 | 36 | 18 |
| 64QAM | 2/3 | 48 | 48 | 24 |
| 64QAM | 3/4 | 54 | 54 | 27 |

(2) Técnica OPS-SAP

Se implementarán funciones en MATLAB para modelar la técnica OPS-SAP, que utiliza dos técnicas: SAP la cual es un esquema enfocado en la extensión de constelación que a su vez se basa en una métrica [4] y secuencias piloto ortogonales u OPS que es una técnica de edición de la PAPR que ayuda con la estimación de canal [5], dando como resultado que la técnica SAP se encuentre asistida por OPS con el fin de alcanzar mejores resultados ya que se obtiene una buena compensación entre el rendimiento y la complejidad de la transmisión.

La evaluación del rendimiento de OPS-SAP se presentarán a través de curvas de CCDF de la PAPR, BER vs SNR y PSD para cada uno de los estándares IEEE 802.11a/g/p.

Para complementar el trabajo práctico, este trabajo de titulación presentará una breve revisión teórica sobre las redes Ad-hoc, específicamente de la capa física de las enmiendas IEEE 802.11a/g/p, el problema de la PAPR y la técnica OPS-SAP.

1.3 MARCO TEÓRICO

En esta sección se abordarán todos los aspectos asociados al trabajo de titulación. Se iniciará con el estudio y una introducción a las redes Ad-Hoc y se profundizará en la capa física OFDM de los estándares IEEE 802.11a/g/p válidos para redes Ad-Hoc. Posteriormente se explicará el problema de la PAPR y sus principales técnicas de reducción, y se abordará a detalle la técnica de reducción de la OPS-SAP. Se explicarán las métricas de evaluación de una técnica de reducción de la PAPR.

1.3.1. REDES AD-HOC

En los últimos años las redes Ad-Hoc han cobrado mucha popularidad debido a la rápida evolución de la tecnología inalámbrica, la proliferación de equipos portátiles, la alta demanda de conectividad móvil y la interconectividad entre diferentes grupos de trabajo en tiempo real [6].

Las redes Ad-Hoc son un tipo de red inalámbrica descentralizada que no depende de una infraestructura pre-existente, están formadas por un grupo de equipos que se interconectan entre sí, es decir que, cada estación actúa como un nodo independiente que a su vez está

en la posibilidad de participar en el enrutamiento mediante el reenvío de datos hacia otros nodos [7].

La flexibilidad y autonomía que presentan estas redes facilitan la conectividad entre hogares, oficinas y sitios comerciales, además la baja latencia, el bajo consumo de energía y confiabilidad en la conexión, han puesto estas redes como la mejor solución en cuanto a conexión móvil [8].

Dentro de redes Ad-Hoc se tienen: redes Ad-Hoc móviles o MANETS (*Mobile Ad-Hoc Networks*), redes inalámbricas malladas y redes de sensores WSN (*Wireless Sensor Networks*) [6]. Las redes MANET o malladas se fundamentan en algunos estándares de la norma IEEE 802.11 como IEEE 802.11a [9], IEEE 802.11g [10] e IEEE 802.11p [11] que definen el control de acceso al medio o MAC (*Medium Access Control*) y la capa Física o PHY (*Physical Layer*).

Los estándares IEEE 802.11a, IEEE 802.11g e IEEE 802.11p fundamentan su capa física en la multiplexación por división de frecuencias ortogonales o OFDM (*Orthogonal Frequency Division Multiplexing*), debido a que posee algunas ventajas como: la robustez frente al desvanecimiento por trayectos múltiples, alta eficiencia de ancho de banda y una estructura de ecualizador simple [3].

1.3.2. ESTANDAR IEEE 802.11a/g/p

1.3.2.1. IEEE 802.11a

Este estándar opera en la banda de 5 GHz, posee 12 canales de 20 MHz de los cuales 8 son empleados para ambientes indoor y 4 pueden ser utilizados en ambientes indoor u outdoor los cuales no son sobrelapados [9].

El estándar define una técnica de transmisión OFDM, posee 52 subportadoras de las cuales 48 son usadas para datos y 4 como portadoras piloto para la supervisión de los desfases de rutas e ICI (*Inter Carrier Interference*), los canales se enumeran de -26 a 26 y la duración del símbolo OFDM es de 4 μ s [12].

1.3.2.2. IEEE 802.11g

Este estándar se presenta como la evolución del estándar IEEE 802.11b. Posee una velocidad máxima de 54 Mbps siendo las velocidades de 6, 12 y 24 obligatorias [10]. Opera en la banda de 2.4 GHz, este estándar añade una variación a la PHY de velocidad extendida (ERP, *Extended Rate PHY*). Existen varios tipos de ERP, pero en este trabajo se utilizará ERP-OFDM que es el modo principal de este estándar [10].

1.3.2.3. IEEE 802.11p

El estándar IEEE 802.11p define el acceso inalámbrico en ambientes vehiculares [11]. Utiliza una técnica de transmisión multiportadora OFDM, opera en un rango de frecuencias de 5.850 GHz a 5.925 GHz, posee un ancho de banda de 75 MHz con 7 canales cuyo espacio es de 10 MHz y alcanza una tasa de transmisión de hasta 27 Mbps [11]. La duración del símbolo OFDM es de 8us [11].

1.3.3. COMPARACIÓN ENTRE LOS ESTÁNDARES IEEE 802.11a/g/p

Los estándares IEEE 802.11a, IEEE 802.11g e IEEE 802.11p se fundamentan en la arquitectura OFDM de IEEE 802.11 [3]. Por lo tanto, todos los estándares poseen el mismo formato de trama PLCP y realizan el mismo procedimiento en el transmisor como en el receptor. Las diferencias que presentan los estándares se detallan en la Tabla 1.2.

Tabla 1.2 Diferencias entre los estándares IEEE 802.11a/g/p.

| Especificaciones | IEEE 802.11a | IEEE 802.11g | IEEE 802.11p |
|-------------------------------|------------------------------------|------------------------------------|-------------------------------------|
| Subportadoras | 52 | 52 | 52 |
| Duración del símbolo OFDM | 4us | 4us | 8us |
| Tiempo de guarda | 0.8us | 0.8us | 1.6us |
| Periodo FFT | 3.2us | 3.2us | 6.4us |
| Duración de Preámbulo | 16us | 16us | 32us |
| Espaciamiento de Subportadora | 0.3125 MHz | 0.3125 MHz | 0.15625 MHz |
| Ancho de Banda | 20 MHz | 20MHz | 10 MHz |
| Velocidades | 6, 9, 12, 18, 24, 36,48 y 54 Mbps. | 6, 9, 12, 18, 24, 36,48 y 54 Mbps. | 3, 4.5, 6, 9, 12, 18, 24 y 27 Mbps. |

IEEE 802.11p opera con un ancho de banda de 10 MHz, que representa la mitad del ancho de banda en relación a IEEE 802.11a y IEEE 802.11g. Los tres estándares trabajan con 52 subportadoras, pero el espaciamiento entre dichas subportadoras difiere ya que IEEE 802.11p presenta la mitad del espaciamiento con respecto a IEEE 802.11a y IEEE 802.11g.

1.3.4. CAPA FÍSICA DE LOS ESTÁNDARES IEEE 802.11a/g/p

La estructura de la capa física de los tres estándares es la misma y se encuentra constituida por tres subcapas: PLME (*Physical Layer Management Entity*), PLCP (*Physical Layer Convergence Procedure*) y PMD (*Physical Medium Dependent*) [13].

1.3.4.1. Subcapa PLME

La subcapa PLME se encarga de comunicarse con una entidad de administración de la estación y a nivel MAC denominada MLME (*MAC Sublayer Management Entity*) por medio de los puntos de acceso al servicio SAPs (*Service Access Points*) [3] [13] [14].

1.3.4.2. Subcapa PLCP

La subcapa PLCP es utilizada como una interfaz entre la subcapa MAC y la PMD. La subcapa MAC proporciona la trama lógica denominada MPDU (*MAC Protocol Data Unit*) que a nivel de subcapa PLCP se denomina PSDU (*PLCP Service Data Unit*) que a su vez es entregada a la subcapa PMD para su procesamiento [14].

La subcapa PLCP se encarga de colocar el PSDU en un formato adecuado mediante encapsulamiento que contenga información sobre los parámetros de transmisión y de la propia trama que no es más que añadir un preámbulo, cabecera y un trailer al PSDU [3] [13] [14]. La nueva trama formada toma el nombre de PPDU (*PLCP Protocol Data Unit*) que es enviada a la subcapa PMD para su tratamiento y posterior envío a través del medio inalámbrico.

a) Estructura de la Trama PLCP

La trama PLCP cuenta con varios campos Figura 1.2, donde se observa que está formada por un preámbulo, signal y datos como campos principales.

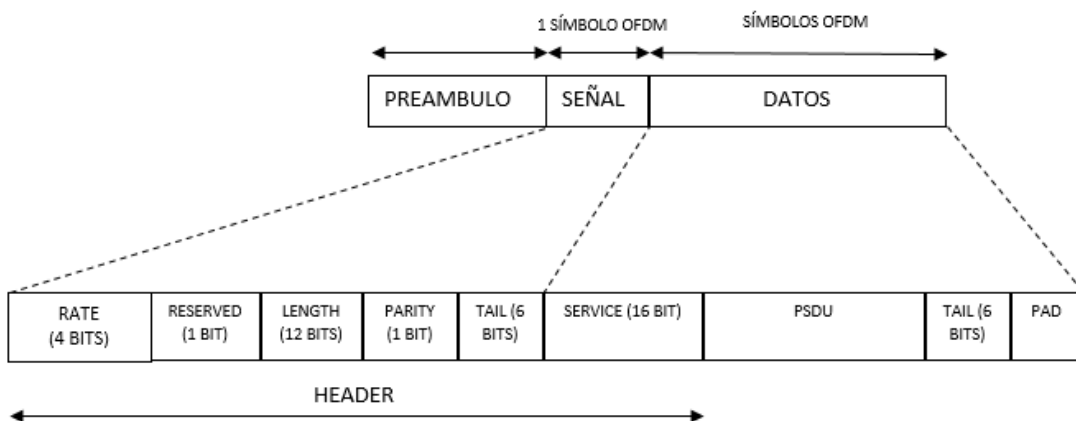


Figura 1.2 Subcapa PLCP

El preámbulo consiste en una serie conocida de 1's y 0's empleado en la sincronización que sirve para iniciar una transmisión, el preámbulo se forma de dos partes: un preámbulo largo y uno corto [3] [13].

La cabecera (*Header*) es transmitida como un símbolo OFDM que es codificado con una tasa de codificación $r = 1/2$ y un esquema de modulación BPSK [3].

La cabecera se encuentra constituida por los siguientes campos:

RATE: Este campo está compuesto por 4 bits que se encargan de indicar la velocidad [15].

LENGTH: Se compone de 12 bits que se encargan de codificar el número de bytes en la trama MAC incrustada. La longitud es procesada mediante un código de circunvolución para protegerlo frente a errores de bit [15].

RESERVED y *PARITY*: Se conforman por un bit cada campo [15].

TAIL se usa para resetear al codificador, además de proporcionar información para el codificador [16]. El campo *SERVICE* se emplea para inicializar el *scrambler* (7 bits) y los 9 bits restantes son reservados para uso futuro [16].

Finalmente, al concatenar el preámbulo, Header junto con el campo *SERVICE*, *PSDU*, *TAIL* y *PAD* forman la PPDU cuyo valor es un entero de símbolos OFDM [3].

Cabe aclarar que para el estándar IEEE 802.11g el preámbulo corto PPDU es obligatorio [10]. Además, trabaja con la capa física ERP-OFDM (*Extended Rate Physical Layer - OFDM*) la cual permite que el estándar se adapte a una frecuencia de 2.4 GHz y tenga las velocidades indicadas en la Tabla 1.1, además se requiere un tiempo de actividad de $6 \mu s$ denominado extensión de señal para asegurar que el proceso de decodificación convolucional llegue a ser completado dentro del tiempo estimado del proceso de transmisión [10].

1.3.4.3. Subcapa PMD

La subcapa PMD es aquella que se encuentra en contacto con el medio externo, la función de esta subcapa consiste en recibir los datos PLCP en un formato correcto y adecuarlas al medio que se encuentre conectado [3], además ofrece a la PLCP el servicio y recepción de tramas en el medio inalámbrico ya que convierte la información (binaria) en señales de radio.

La subcapa PMD se fundamenta en la técnica de transmisión OFDM para todos los estándares, es decir que provee todos los procesos para generar un símbolo OFDM a partir de una cadena de bits [3].

El sistema OFDM de acuerdo al estándar se encuentra estructurado de la siguiente manera: Transmisor, canal (inalámbrico) y receptor Figura 1.2. Las funciones del transmisor son: modular los datos, procesarlos usando la IFFT (*Inverse Fast Fourier*

Transform) para obtener la señal en el dominio del tiempo, agregar un intervalo de guarda (prefijo cíclico), además de ciertos bloques adicionales para luego transmitirla por el canal inalámbrico. La señal llega al receptor, en el cual se realiza el proceso inverso al desarrollado en el transmisor OFDM [17].

La Figura 1.3 presenta un diagrama con los bloques que conforman un transmisor OFDM entre los que se tiene: modulación, conversor serie/paralelo, multiplexación y prefijo cíclico. A continuación, se realiza una breve descripción de cada bloque.

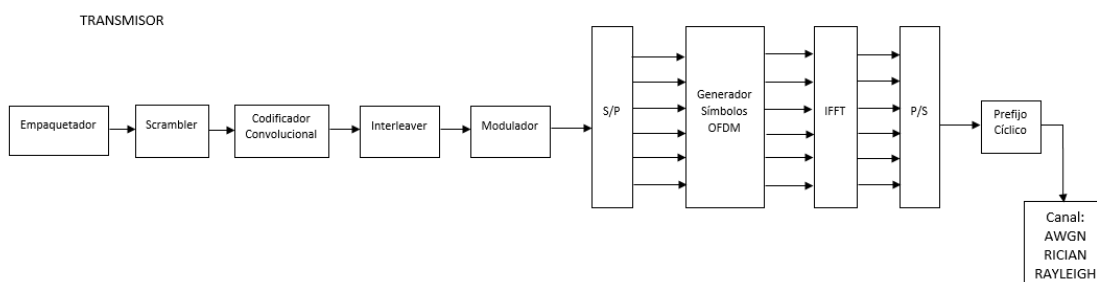


Figura 1.3 Diagrama del transmisor OFDM.

a) Aleatorizador

El objetivo del aleatorizador es asegurar que no se generen secuencias largas de 1's lógicos en los datos binarios, el aleatorizador se establece a través de un polinomio generador (Ecuación 1.1) en este caso tiene la siguiente forma de acuerdo al estándar IEEE 802.11 [3]:

$$S(x) = x^7 + x^4 + 1 \quad (1.1)$$

b) Codificador de canal

Es un sistema de estado finito en el cual por cada e bits de entrada, se entregan h bits de salida cumpliendo con la premisa $h > e$. En este caso, los bits de salida no dependen únicamente de los bits de entrada, sino que, de los e bits anteriores a la salida denominados longitud de código, donde e es la memoria del codificador que corresponde al número de registros en serie más una unidad [18]. La tasa de codificación viene dada por la siguiente expresión matemática (1.2):

$$r = \frac{e}{h} \quad (1.2)$$

La codificación convolutiva está en la capacidad de realizar una corrección robusta de errores en recepción a cambio de introducir redundancia en los datos, el proceso para la codificación según el estándar IEEE 802.11 se describe a continuación [3].

Cada uno de los datos binarios previamente aleatorizados atraviesan el codificador convolucional con los siguientes polinomios generadores binarios (expresiones (1.3), (1.4)).

$$\partial_1 = 1011011 \quad (1.3)$$

$$\partial_2 = 1111001 \quad (1.4)$$

La forma como trabaje el codificador convolucional depende de la afectación o no de cada bit de la longitud del código sobre cada bit de salida, participante o no en la operación XOR entre todos ellos o no, generándose como resultado un determinado bit de salida. Dicho comportamiento se representa por medio de los polinomios generadores.

En consecuencia al comportamiento de los bits de la longitud del código puede afectar a los h bits de salida de forma independiente, existe un polinomio generador por cada bit de salida es decir h polinomios generadores. Teniendo en cuenta que los e bits están en la posibilidad de afectar o no a un bit de salida, cada polinomio generador posee una longitud de e . Además, como consecuencia a que los polinomios generadores expresan si un bit de la longitud del código entra o no a la operación XOR da como resultado un bit de salida ya sea 0's o 1's [3].

Para finalizar, al tener e entradas simultaneas para cada uno de los h bits de salida e correspondientes a los polinomios generadores de longitud E . De esta manera al conocer los polinomios generadores de un codificador convolucional, el mismo queda caracterizado por completo [18].

Las diferentes tasas de codificación que indica el estándar [3] se presentan en la Tabla 1.1. El codificador por defecto posee una tasa de codificación $r = 1/2$, pero para obtener una tasa de codificación mayor con este mismo codificador es necesario utilizar el concepto de puntura. Dicha técnica consiste en omitir ciertos bits a la salida del codificador en la etapa de codificación, mientras tanto en la etapa de decodificación se ingresan bits 0's en estos sitios antes de realizar el proceso [18]. Los patrones de puntura para las tasas de codificación $r = 1/4$ y $r = 2/3$ se presentan en las expresiones (1.5) y (1.6) respectivamente.

$$Punt_{3/4} = [111001] \quad (1.5)$$

$$Punt_{2/3} = [11110] \quad (1.6)$$

Cuando se realiza el proceso de decodificación de preferencia se emplea el algoritmo de Viterbi que se presenta en [18].

c) Entrelazado

La implementación del entrelazado (*Interleaver*) ayuda a contrarrestar los efectos negativos del canal inalámbrico al permutar los bits adyacentes, de esta manera los errores aparecen de forma aislada en el receptor evitando que aparezcan en ráfagas [19]. El proceso de entrelazado según el estándar IEEE 802.11 se describe a continuación [3].

Los datos binarios previamente codificados pasan a través del entrelazador, este proceso consiste en dos permutaciones realizadas sobre bloques de datos con longitud fija de forma independiente. El tamaño de cada uno de los bloques es igual al número de bits que fueron codificados por bloque OFDM (N_{CBPS}). Con la primera permutación se asegura que los bits codificados adyacentes no se coloquen en subportadoras adyacentes, la segunda permutación se encarga de asegurar que los bits codificados adyacentes sean modulados alternadamente en uno o varios bits significativos [3].

Cada permutación se caracteriza por los índices, quienes indican su posición dentro de cada bloque, por lo tanto, se tiene:

- a corresponde al índice de cada elemento antes de la primera permutación.
- b corresponde al índice de cada elemento después de la primera permutación.
- c corresponde al índice de cada elemento después de la segunda permutación.

Por lo tanto, se tiene que $a, b, c = 0, 1, \dots, N_{CBPS} - 1$, como se aprecia en la Figura 1.4.

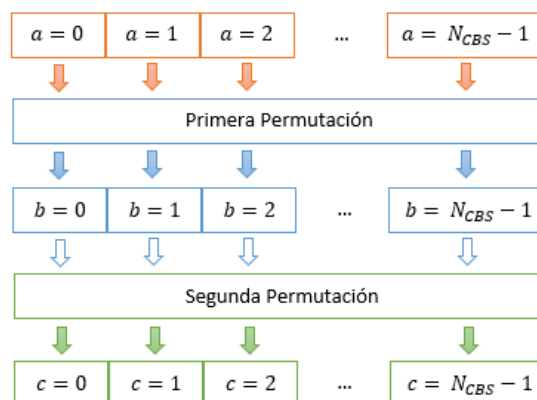


Figura 1.4 Índices en el proceso de entrelazado

Mediante operaciones sobre los índices se realizan las permutaciones para determinar los siguientes índices que intervienen en el proceso. En primer lugar se toma el índice a de cada elemento del bloque y se procede a realizar la primera permutación, obteniendo el índice correspondiente b , a continuación, se realiza la segunda permutación con los índices

b para obtener los índices correspondientes c , quienes indican la posición que deben tomar los elementos dentro del bloque al finalizar el entrelazado [3].

Un parámetro empleado para el entrelazado se indica en la ecuación (1.7).

$$s = \max\left(\frac{N_{CBPS}}{2}, 1\right) \quad (1.7)$$

La primera permutación del entrelazador está dada por la ecuación (1.8) [3].

$$b = \left(\frac{N_{CBPS}}{16}\right)(a \bmod 16) + \text{Floor}\left(\frac{a}{16}\right) \quad (1.8)$$

Donde Floor() corresponde a la función piso o redondeo hacia abajo (toma el entero más próximo hacia abajo) y mod es la operación modulo.

La segunda permutación del entrelazador está dada por la ecuación (1.9) [3].

$$c = s\text{Floor}\left(\frac{b}{s}\right) + \left(b + N_{CBPS} - \text{Floor}\left(16\frac{b}{N_{CBPS}}\right)\right) \bmod s \quad (1.9)$$

Donde s está dado por la ecuación (1.7) y N_{CBPS} corresponde al número de bits codificados por subportadora.

En el lado del receptor el proceso del deentrelazado se realiza de forma diferente, primero se realiza la segunda permutación y luego la primera permutación obteniéndose así los datos originales.

d) Modulación

El objetivo de la modulación digital consiste en introducir los datos de información que van a ser transmitidos en banda base, ya sea en frecuencia, fase, amplitud o una combinación de las mismas en una señal denominada portadora [20].

Los siguientes tipos de modulación son utilizados por el estándar IEEE 802.11 para comunicaciones Wi-Fi [3]:

Phase Shift Keying (PSK)

El esquema de modulación PSK consiste en realizar un cambio en la fase de la portadora en base a la señal en banda base. La variación entre fases toma niveles de π/M [20]. La modulación PSK tiene la cualidad de poseer varios niveles (multinivel) por lo tanto los datos tienen la posibilidad de tomar diferentes niveles y así representar los dígitos binarios, dichos dígitos ya sean 1s o 0s toman M niveles donde $M = 2^m$ y m representa el número de dígitos en cada nivel.

El diagrama de constelación ayuda a representar la señal modulada, en dicho diagrama los símbolos son representados por un punto, los ejes X e Y están encargados de representar los coeficientes en cuadratura que conforman el diagrama de constelación, además utilizan las funciones seno (eje X) y coseno (eje Y). La señal seno toma el nombre de señal I o referencia, mientras tanto la señal coseno se la denomina señal Q y se encuentra desfasada con respecto a la señal I [20].

Un parámetro muy importante en el diagrama de constelación es la distancia mínima, la cual es la distancia entre puntos adyacentes. El valor de la distancia mínima se calcula empleando la formula presentada en la ecuación (1.10) [20].

$$d = 2 * \text{sen}(\pi/M) \quad (1.10)$$

La distancia entre puntos adyacentes se encuentra relacionada de forma directa con los errores en un sistema de modulación digital presente en el receptor, si la distancia es mayor, la inmunidad al ruido crece. En otras palabras “si la distancia mínima en el diagrama de constelación aumenta, la BER (*Bit Error Rate*) disminuye y para una misma BER la Relación Señal a Ruido (SNR) aumenta” [20].

Los esquemas de modulación empleados en sistemas OFDM son: B-PSK, Q-PSK, donde B y Q representan $M = 2$ y $M = 4$ respectivamente.

- Modulación B-PSK

Un esquema de modulación B-PSK está constituido por dos fases: la primera fase corresponde a los 1s lógicos mientras que la segunda corresponde a los 0s lógicos con una separación en ambos casos de π [21]. En la Figura 1.5 se observa el diagrama de constelación correspondiente a una modulación B-PSK [22].

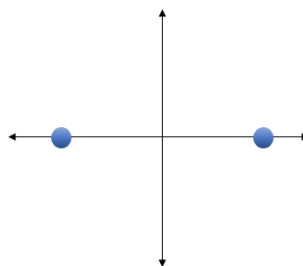


Figura 1.5 Diagrama de constelación de una modulación B-PSK

- Modulación Q-PSK

Una modulación Q-PSK presenta cuatro diferentes fases, las cuales son asignadas a cada combinación de 2 bits y posee una separación de $\pi/2$ [21]. En la Figura 1.6. se observa el diagrama de constelación para la modulación Q-PSK [20].

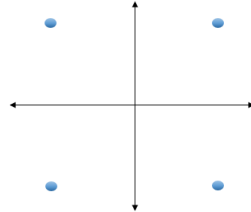


Figura 1.6 Diagrama de constelación de una modulación Q-PSK

Quadrature Amplitude Modulation (QAM)

Este tipo de modulación presenta una variación tanto en amplitud como fase. Las señales generadas en banda base son independientes entre sí [20]. La modulación QAM está en la posibilidad de ser multinivel: Si $M < 4$ el uso del espectro es mucho más eficiente, pero al realizarse modulaciones de mayor nivel como 16-QAM, 64-QAM, etc., la separación entre puntos adyacentes presentes en el diagrama de constelaciones disminuye dando como resultado que la probabilidad de error (PE) se incremente [20].

La distancia mínima entre puntos adyacentes para un diagrama de constelación con modulación M-QAM está representada en la ecuación (1.11), donde $L = \sqrt{M}$ y M es el nivel de modulación [20].

$$d = \frac{\sqrt{2}}{L-1} \quad (1.11)$$

- Modulación 16-QAM

El diagrama de constelación perteneciente a 16-QAM se lo presenta en la Figura 1.7, donde se cumple una regla muy importante la cual indica que los puntos adyacentes varían en un bit dando como resultado la disminución de la PE [23].

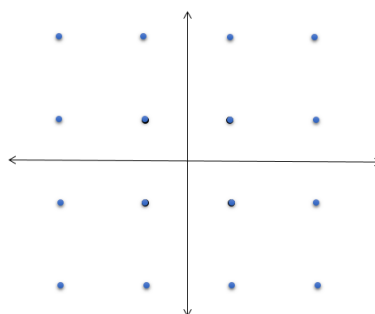


Figura 1.7 Diagrama de constelación de una modulación 16-QAM

- Modulación 64-QAM

En [23] los autores indican que “Una modulación 64-QAM presenta la cualidad de transmitir 6 bits por símbolo proporcionando una gran ventaja respecto a modulaciones de menor nivel, pero a su vez es mucho más susceptible al ruido y errores en recepción”. Los puntos adyacentes se encuentran bastante apegados como se observa en la Figura 1.8, los puntos pertenecientes al diagrama de constelación toman valores entre (1, 3, 5, 7, -1, -3, -5, -7) ya sea para I como para Q [23].

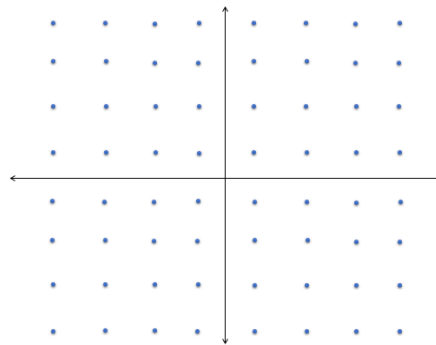


Figura 1.8 Diagrama de constelación de una modulación 64-QAM.

Para cada esquema de modulación le corresponde una tasa de codificación como indica el estándar Tabla (1.3).

Tabla 1.3 Esquemas de modulación y tasas de codificación de acuerdo al estándar IEEE 802.11 [3].

| Esquema de Modulación | Tasa de Codificación |
|-----------------------|----------------------|
| BPSK | $1/2$ |
| BPSK | $3/4$ |
| QPSK | $1/2$ |
| QPSK | $3/4$ |
| 16-QAM | $1/2$ |
| 16-QAM | $3/4$ |
| 64-QAM | $2/3$ |
| 64-QAM | $3/4$ |

El proceso de demodulación se lleva a cabo teniendo en cuenta el número de bits que se debe asignar a cada símbolo dependiendo del esquema de modulación que se tenga según el estándar IEEE 802.11 [3]. La Tabla 1.4, presenta el número de bits que se debe asignar a cada símbolo dependiendo del esquema de demodulación.

Tabla 1.4 Número de bits por cada símbolo en el proceso de demodulación.

| Esquema de Demodulación | Número de bits por cada símbolo |
|-------------------------|---------------------------------|
| BPSK | 1 |
| QPSK | 2 |
| 16-QAM | 4 |
| 64-QAM | 6 |

e) Bloque OFDM

OFDM es una técnica de transmisión multiportadora MCM (*Multicarrier Modulation*) que se fundamenta en la división del ancho de banda total en diferentes subportadoras que llevan la información de forma simultánea [24].

La técnica de transmisión OFDM se fundamenta en la división del ancho de banda BW (*Bandwidth*) disponible en N subcanales denominados subportadoras, cada una de las cuales realiza una modulación independiente de la información que transmite.

El objetivo principal que persigue OFDM es que las N subportadoras sean ortogonales entre sí, donde la frecuencia de cada subportadora está dada por $f_k = k\Delta f$, con $k = \{0, 1, \dots, N - 1\}$ donde Δf es la separación entre subportadoras consecutivas y está dado por la división entre el ancho de banda total disponible BW y N . Para cumplir con el principio de ortogonalidad se debe cumplir que cada subportadora debe tener una duración de $T = 1/\Delta f$ o múltiplo de T . Las subportadoras se encuentran sobrepuestas con el objetivo de obtener una mayor eficiencia espectral, además la ortogonalidad de las subportadoras garantiza que no exista ICI [21].

Los esquemas MCM transmiten datos de forma simultánea por medio de un mismo canal de banda limitada y elimina completamente los efectos de la ICI (*Inter Carrier Interference*) e ISI (*Inter Symbol Interference*) [24]. Luego de varios años, una modulación y demodulación es posible llevar a cabo mediante la transformada discreta de Fourier DFT (*Discrete Fourier Transform*) [22]. En la actualidad se emplea la transformada rápida de Fourier FFT (*Fast Fourier Transform*) reduciendo así su compleja implementación. Más adelante, investigadores concibieron un nuevo concepto que se lo denominó prefijo cíclico cuyo fin era combatir los efectos tanto de ICI e ISI y de esta manera mantener la ortogonalidad mediante la eliminación de un tiempo de guarda vacío por una extensión cíclica del símbolo OFDM [25].

1) Conversión serie-paralelo

Los bits de información son distribuidos en grupos de m bits paralelos, cuyos grupos son asignados a una subportadora diferente. El número de bits perteneciente a cada grupo es determinado por el tipo de modulación que se esté utilizando.

2) Transformada de Fourier

En [26] los autores indican que “Los sistemas digitales existentes en la actualidad obligaron a que la transformada de Fourier ya no trabaje en tiempo continuo sino en tiempo discreto, dando como resultado que la señal se encuentre discretizada y representada por medio de impulsos. A esta relación se la denomina Transformada Discreta de Fourier (DFT) y viene dada por la ecuación (1.12)”.

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \quad 0 \leq k \leq N - 1 \quad (1.12)$$

Donde $x[n]$ es la señal de N muestras que se encuentran en el dominio del tiempo, $X(k)$ son las muestras del símbolo OFDM en el dominio de la frecuencia, n y k representan los índices en el dominio del tiempo y frecuencia respectivamente.

Para realizar el proceso inverso se emplea la IDFT (*Inverse Discrete Fourier Transform*) que está representada por la ecuación (1.13) [26]:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N} \quad 0 \leq n \leq N - 1 \quad (1.13)$$

La operación de IFFT se encuentra representada en la Figura 1.9.

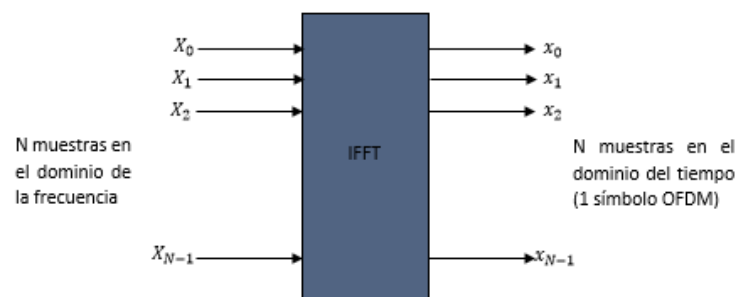


Figura 1.9 Operación IFFT en una señal OFDM

FFT e IFFT son utilizados por los sistemas OFDM, los cuales son una variación de DFT. Estos algoritmos son muy útiles ya que reducen la carga computacional [26].

Intervención de IFFT en OFDM

Una vez realizada la modulación en fase Q-PSK o cuadratura M-QAM, se tienen N números complejos $X(k)$ con $k = \{0, 1, \dots, N - 1\}$ los cuales están conformados por una parte real y

otra imaginaria, al utilizar la IFFT se obtiene la señal en el dominio del tiempo $x[n]$ donde, $n = \{0, 1, \dots, N - 1\}$.

Cada número complejo en $X(k)$ se relaciona con una subportadora $k = \{0, 1, \dots, N - 1\}$, al proceder a normalizar la ecuación anterior, se tiene la expresión matemática (1.14) [27], obteniendo así la señal en $x[n]$ en el dominio del tiempo.

$$x[n] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N} \quad 0 \leq n \leq N - 1 \quad (1.14)$$

Una vez finalizado el proceso en el bloque IFFT se obtendrá el símbolo OFDM $x[n] = \{x[0], x[1], \dots, x[N - 1]\}$, cuya longitud es N [28].

3) Prefijo Cíclico

Cuando se pierde la ortogonalidad de las subportadoras debido al efecto del multitrayecto, aparecen los efectos del ICI e ISI en los sistemas OFDM.

Para combatir este efecto se utilizan las bandas de guarda, pero si dichas bandas eran enviadas vacías no es posible contrarrestar el efecto de ICI, como resultado de lo anterior se optó por la inserción del prefijo cíclico CP (*Cyclic Prefix*) [25].

El prefijo cíclico consiste en copiar las J últimas muestras del símbolo OFDM que se obtienen a la salida del IFFT y ubicarlas al inicio, tal como se observa en la Figura 1.10. [25]. Cuando se tienen 64 muestras esto en el entorno discreto, lo anterior descrito se traduce como tomar las últimas 16 muestras (correspondiente a la cuarta parte) y copiarlas al principio, por lo tanto, se obtiene un arreglo con 80 muestras.

Al realizar este proceso se asegura la periodicidad del nuevo símbolo, de ahí que a este proceso se lo denomina prefijo cíclico.

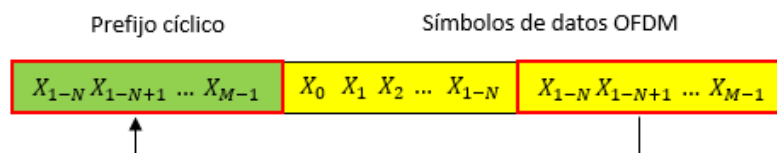


Figura 1.10 Inserción del prefijo cíclico en el símbolo OFDM

Por ejemplo, si a la salida del IFFT se tienen $N = 11$ muestras que pertenecen a un símbolo OFDM cuyo tamaño de PC es $J = 4$, el $x[n] = \{abcdefghijk\}$ (muestras a la salida del IFFT), por lo tanto las muestras que pertenecen al prefijo cíclico son $\{hijk\}$ como resultado el nuevo símbolo OFDM (con datos y prefijo cíclico) será $x_{pc}[n] = \{hijkabcdefghijk\}$ cuyo tamaño aumento a $N + J$ [26].

En [18] y [26] los autores indican “Al introducir el CP se mejora el rendimiento de los sistemas OFDM y se reduce el BER, al contrarrestar los efectos que introduce el canal como interferencia y ruido blanco. Una desventaja que presenta el CP si está en el dominio de la frecuencia es aumentar el ancho de banda, pero esto es insignificante en comparación a las ventajas presentadas”.

1.3.5. AMPLIFICADOR DE ALTA POTENCIA

El trabajo del HPA (*High Power Amplifier*) consiste en amplificar la potencia de la señal, de tal suerte que se logre entregar la potencia máxima a la antena y alcance su máxima eficiencia [29].

Para determinar el HPA más adecuado para un sistema se debe tener en cuenta ciertas características como son: linealidad y eficiencia. La linealidad del amplificador indica que la señal de salida no presente distorsiones respecto a la señal de entrada del amplificador [30]. La eficiencia del amplificador es la relación entre la potencia requerida para amplificar la señal de salida con respecto a la potencia que se pierde en forma de calor [31].

Teniendo en cuenta que un amplificador real es un sistema no lineal, se conoce que en el rango que opera dicho amplificador no presenta una ganancia constante. Esta característica provoca que la señal de salida presente distorsiones, ya sea en amplitud (conversión AM/AM) o en fase (conversión AM/PM) donde AM (*Amplitud Modulation*) y PM (*Phase Modulation*) [32]. Los parámetros de conversión son de suma importancia para la formulación de modelos matemáticos que definen el comportamiento de la operación de los HPA, ya que la salida del amplificador se modela como una función de estos dos parámetros [32].

Un aspecto importante para elegir un HPA adecuado es el punto de saturación del amplificador, este punto está definido por la relación entre IBO (*Input Back-off*) y OBO (*Out Back-off*) cuyas expresiones matemáticas se muestran en (1.15) y (1.16) respectivamente [2].

$$IBO = 10 * \log \frac{P_{sat,in}}{P_{pro,in}} \quad (1.15)$$

$$OBO = 10 * \log \frac{P_{sat,out}}{P_{pro,out}} \quad (1.16)$$

Donde $P_{sat,in}$ y $P_{sat,out}$ son las potencias máximas instantáneas de entrada y salida respectivamente, mientras tanto, $P_{pro,in}$ y $P_{pro,out}$ corresponden a las potencias medias de entrada y salida respectivamente [2].

Tanto IBO como OBO tienen una gran importancia ya que sus valores son tomados como referencia para mantener la señal OFDM con PAPR elevada, dentro de la zona lineal del HAP evitando así que la señal se distorsione al atravesar el amplificador [2].

El amplificador más común en equipos bajo el estándar IEEE 802.11 es el modelo Rapp que produce una transición suave de la envolvente modulada hacia el nivel de saturación [33].

El modelo Rapp considera el siguiente modelo matemático (ecuación 1.17) para la conversión AM/AM [33].

$$G[|x[n]|] = \frac{g \cdot |x[n]|}{\left(1 + \left(\frac{|x[n]|}{x_{sat}}\right)^{2p}\right)^{\frac{1}{2p}}} \quad (1.17)$$

Donde, n son las muestras que conforman la señal, $|x[n]|$ representa la señal que entra al HPA, g representa la ganancia, x_{sat} es el valor de saturación del amplificador, finalmente p representa el parámetro que rige la precisión de la región de saturación del HPA.

La conversión AM/PM considera un modelo matemático cuyo valor es similar a cero ya que sus efectos no son exactamente cero, por lo tanto, el modelo matemático es representado por la ecuación (1.18) [33].

$$\emptyset[|x[n]|] = 0 \quad (1.18)$$

1.3.6. PROBLEMA DE LA PAPR ELEVADA EN SISTEMAS OFDM

Los sistemas OFDM manifiestan un problema debido al valor elevado de la PAPR esto ocurre en el lado del transmisor ya que se puede dar la suma de las subportadoras con la misma fase produciendo así que el nivel de potencia tenga un valor alto. Los picos elevados de potencia pueden causar que el HPA opere de manera errónea, dando como consecuencia que el HPA trabaje en su zona de saturación degradando la señal transmitida [27].

Si la eficiencia energética alcanzada por el sistema OFDM es buena, es posible lograr que la cobertura sea de mayor alcance, dando como resultado que exista un ahorro en el consumo de energía y que los dispositivos tengan un tamaño menor [34].

1.3.6.1. Definición de la PAPR

Una forma de medir la contribución de potencia considerable ocasionada por la suma de subportadoras con la misma fase, es la relación entre la potencia máxima y su potencia promedio conocida como PAPR (*Peak-to-Average Power Ratio*)[35].

Existen varias formas en las que puede ser expresada la PAPR ya sea en banda base o paso banda, así mismo existe la posibilidad de expresarla en el dominio del tiempo continuo o tiempo discreto.

Es así que, la PAPR para un símbolo OFDM en el dominio del tiempo se presenta en la ecuación matemática (1.19) [5].

$$PAPR\{x[n]\} = \frac{\max(|x[n]|)^2}{E[|x[n]|^2]} \quad 0 \leq n \leq N - 1 \quad (1.19)$$

Donde $x[n]$ son las muestras del símbolo OFDM a la salida de la IFFT, $\max(|x[n]|^2)$ hace referencia a la máxima potencia instantánea y $E[|x[n]|^2]$ representa la potencia promedio de la señal $x[n]$ donde $E[]$ corresponde al valor esperado.

1.3.6.2. Método OPS-SAP para la reducción de la PAPR

Una de las soluciones para que la señal OFDM no presente los problemas mencionados anteriormente, es disminuir el valor de la PAPR antes que atraviese por el HPA.

Se han implementado diferentes técnicas para intentar suprimir los picos de potencia presentes en OFDM. Un método de los más simples consiste en recortar intencionalmente los picos de la señal OFDM antes de que ocurra la amplificación [36]. Sin embargo, el recorte ocasiona la reducción de la PAPR, provocando que se genere un proceso no lineal que provocaría una distorsión significativa en banda, dando como resultado la degradación del BER, por consiguiente, se reduce la eficiencia espectral [36].

Existen técnicas las cuales no producen distorsiones en la señal, las mismas se fundamentan en la variación de los datos fuente como es el caso de la técnica OPS-SAP (*Orthogonal Pilot Sequences - Simple Amplitud Predistortion*). Esta técnica resulta de la unión entre las técnicas OPS (*Orthogonal Pilot Sequences*) y SAP (*Simple Amplitud Predistortion*).

Simple Amplitude Predistortion (SAP)

La técnica SAP consiste en desplazar una cierta distancia cada una de las muestras de los símbolos OFDM que más contribuyen a la formación de los picos de la señal. El desplazamiento debe ser realizado teniendo en cuenta que no se produzca una afectación

en la distancia mínima existente entre los puntos de la constelación, dando como resultado no afectar la BER. Es decir, si no se altera la distancia mínima, no ocurrirá ninguna degradación en la BER, pero si se genera el aumento de la energía requerida por cada símbolo.

En la Figura 1.12. indica cómo se realiza el desplazamiento de los puntos de las muestras de un símbolo OFDM para una modulación 16QAM [4], donde los puntos se desplazan en la dirección que indican las flechas.

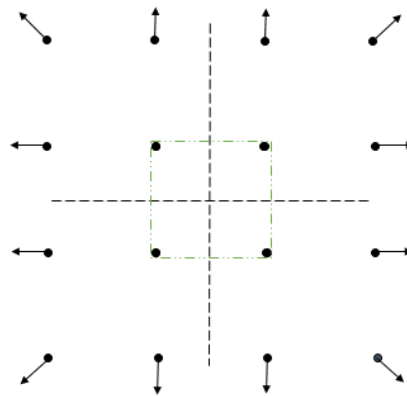


Figura 1.11 Predistorsión de los puntos en 16-QAM

Para determinar el o los símbolos que deben desplazarse (predistorsión) se emplea una métrica, además se amplía con una constante positiva predefinida.

En general, la predistorsión de los símbolos, como también se describe en [4], provoca el aumento de la potencia promedio de la señal transmitida, pero esto puede ser controlado con facilidad limitando el número de símbolos predistorsionados por bloque.

La métrica se encarga de medir cuánto contribuye la muestra del símbolo OFDM en el dominio de la frecuencia a la salida de la IDFT. La métrica perteneciente al símbolo de entrada en el dominio de la frecuencia $X(k)$ está definido en la expresión matemática (1.20) [37].

$$\mu_m = \sum_{n \in S_K} w(n) f(n, k) \quad (1.20)$$

Donde $f(n, k)$ representa una función que provee una medida apropiada para el ángulo de fase entre la muestra de salida $x[n]$ y la contribución del símbolo $X(k)$ a éste, $w(n)$ corresponde a una función de ponderación de $x[n]$. S_K es un conjunto de tamaño K , cuyos elementos son los índices de las muestras de salida con valores que exceden alguno de los valores pertenecientes al umbral predeterminado.

Una opción conveniente para $f(n, k)$ [37] se presenta en la expresión (1.21):

$$f(n, k) = -\cos(\varphi_{nk}) \quad (1.21)$$

Esta función es maximizada cuando los dos números complejos se encuentran en fase opuesta. La función de ponderación $w(n)$ provoca un mayor peso en la expresión de la métrica μ_m y así producir muestras cuyas magnitudes sean más elevadas. Una vez realizado el cálculo de las métricas para cada uno de los símbolos pertenecientes al bloque de entrada, los Y símbolos correspondientes a las métricas positivas de mayor tamaño son predistorsionados en amplitud.

La predistorsión de un símbolo $X(k)$ consiste en transmitir $aX(k)$, donde $a > 1$, en otras palabras, las partes real e imaginaria de los Y símbolos seleccionados se expanden por el mismo factor α .

El seleccionar los parámetros α y Y provoca un fuerte impacto en el rendimiento de la técnica SAP es así que dichos parámetros deben ser optimizados. La mejor opción para realizar este procedimiento es determinar los valores de los parámetros que conducen a una baja potencia pico promedio obtenida de una secuencia grande de símbolos OFDM. Como se indica en [4], los valores de α y Y no requieren cálculos en tiempo real, para implementaciones prácticas, se han considerado valores predeterminados: $\{\alpha= 2, Y = 10\}$, $\{\alpha= 1.3, Y = 40\}$, $\{\alpha= 1.55, Y = 26\}$.

La complejidad más alta del algoritmo se localiza en el cálculo de la métrica, cuya complejidad es proporcional a KN , donde N corresponde al número de subportadoras. La predistorsión de amplitud implica únicamente el escalado de Y símbolos de los datos complejos, que corresponden a una operación simple. Finalmente, una vez que los símbolos de los datos se encuentran predistorsionados, la actualización de las muestras en el dominio del tiempo se puede realizar directamente utilizando las operaciones de seno y coseno en lugar de realizar una IFFT completa [4].

La técnica correspondiente a la predistorsión de símbolos puede ser extendida fácilmente a las constelaciones de señales que presentan modulación QAM de alto nivel de la misma manera que se describe en [4]. Específicamente, los puntos de las esquinas de la constelación se expanden como sucede en 16-QAM, mientras que solo las partes reales o imaginarias de los símbolos laterales se expanden y los puntos internos no presentan ninguna modificación.

Orthogonal Pilot Sequences (OPS)

En sistemas OFDM, los símbolos piloto por lo general se encuentran insertados en la cuadrícula de tiempo-frecuencia 2D para así estimar el canal. El símbolo OFDM es considerado como N subportadoras, donde un subconjunto γ de subportadoras con cardinalidad $N_p = |\gamma|$ llevará símbolos piloto, por lo tanto, los símbolos transmitidos se describen en (1.22) [5]:

$$Z(k) = \begin{cases} P(k), k \in \gamma \\ D(k), k \in \gamma' \end{cases} \quad (1.22)$$

Donde $P(k)$ y $D(k)$ corresponden a los símbolos piloto y de datos respectivamente. La señal transmitida en tiempo discreto $z[n] = p[n] + d[n]$ está en la posibilidad de ser separada en dos partes, como se muestra en la expresión (1.23) [5]:

$$z[n] = \begin{cases} p[n] = \frac{1}{\sqrt{N}} \sum_{k \in \gamma} P(k) e^{\frac{j2\pi kn}{N}} \\ d[n] = \frac{1}{\sqrt{N}} \sum_{k \notin \gamma} D(k) e^{\frac{j2\pi kn}{N}} \end{cases} \quad (1.23)$$

Donde $p[n]$ y $d[n]$ corresponden a las señales en el dominio del tiempo de los símbolos piloto y de datos respectivamente.

La técnica OPS propone el uso de un conjunto predeterminado de M secuencias piloto ortogonales de longitud N_p con ($M \leq N_p$), donde para un símbolo OFDM se elige la secuencia piloto que proporcione la PAPR más baja [37]. Dichas secuencias ortogonales son obtenidas por medio de una matriz *Hadamard* de orden l , la cual es una matriz $l \times l$ formada por 1's y -1's con la premisa de que sus filas son ortogonales.

Los N_p símbolos piloto disponibles para cada símbolo OFDM son recopilados en una secuencia de longitud N denotada como P . Cada secuencia piloto de éste conjunto finito contiene los símbolos piloto en el dominio de la frecuencia en las posiciones piloto, mientras que los ceros se insertan en los restantes. Así el elemento k -ésimo de esta secuencia viene dado en (1.24) [5]:

$$[P]_k = \begin{cases} P(k), k \in \gamma \\ 0, k \notin \gamma \end{cases} \quad (1.24)$$

Dichas secuencias piloto son ortogonales entre ellas, cumpliéndose la condición de ortogonalidad dada en (1.25) [5]:

$$(P_m, P_n) = 0 \quad n \neq m \quad m, n = \{1, \dots, M\} \quad (1.25)$$

Donde $(.,.)$ denota el producto escalar.

En el transmisor, el esquema OPS original propuesto en [5] lleva a cabo M operaciones IDFT en cada símbolo OFDM, como se muestra en la Figura 1.12 [1-2]. Pero, la implementación en el dominio de frecuencia provoca una carga computacional bastante alta debido a las M operaciones adicionales.

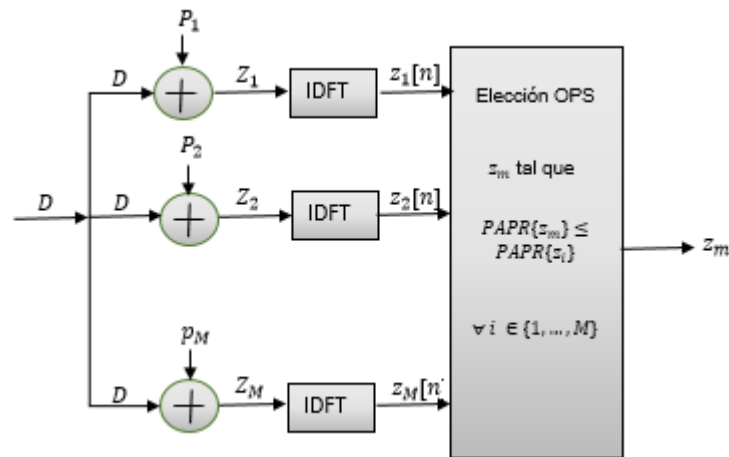


Figura 1.12 Diagrama de bloques de la técnica OPS en el dominio de la frecuencia.

OPS-SAP

Hay ciertos algoritmos que se originaron con el objetivo de ayudar a reducir del problema de la PAPR elevada, pero requieren de cierta complejidad computacional adicional. También, existen técnicas que presentan mejores resultados en la reducción de la PAPR como es el caso de las técnicas OPS y SAP, de esta manera se aborda la reducción de la PAPR en un sistema OFDM mediante secuencias piloto ortogonales combinadas con SAP, dando como resultado la técnica OPS-SAP.

En [1] se ha estudiado esta técnica sobre redes IEEE 802.11p donde se indica que es adecuada para este tipo de redes debido a que: no existe una reducción de la velocidad de transmisión ya que no requiere el envío de información adicional o realizar algún procedimiento en el lado del receptor, ni estimación del canal. Por otra parte, no se produce ninguna degradación en la BER, además se realiza la pre-distorsión de amplitud en OPS-

SAP requiriendo menos energía por símbolo complejo con respecto a SAP, también alcanza una alta eficiencia y baja complejidad computacional (ahorro de energía en el procesamiento) en la reducción de la PAPR en comparación con otros esquemas [1].

La técnica OPS-SAP cumple con el procedimiento mostrado en la Figura 1.13. [1] [2]. En primer lugar, se aplica la técnica OPS de baja complejidad propuesta en [1] [2], es decir, los pilotos ortogonales son insertados adecuadamente, para obtener la PAPR más baja del conjunto disponible. A continuación, se calcula la PAPR y, si dicho valor posee un valor más grande que un umbral predeterminado B (este parámetro puede tomarse como el valor para el que se obtiene la mayor disminución en la PAPR promedio) [1] [2], se procede a aplicar la técnica SAP. En este caso, el algoritmo SAP es aplicado ya sea sobre símbolos piloto o de datos indistintamente, permitiendo así la predistorsión de amplitud de cualquier símbolo complejo [1] [2].

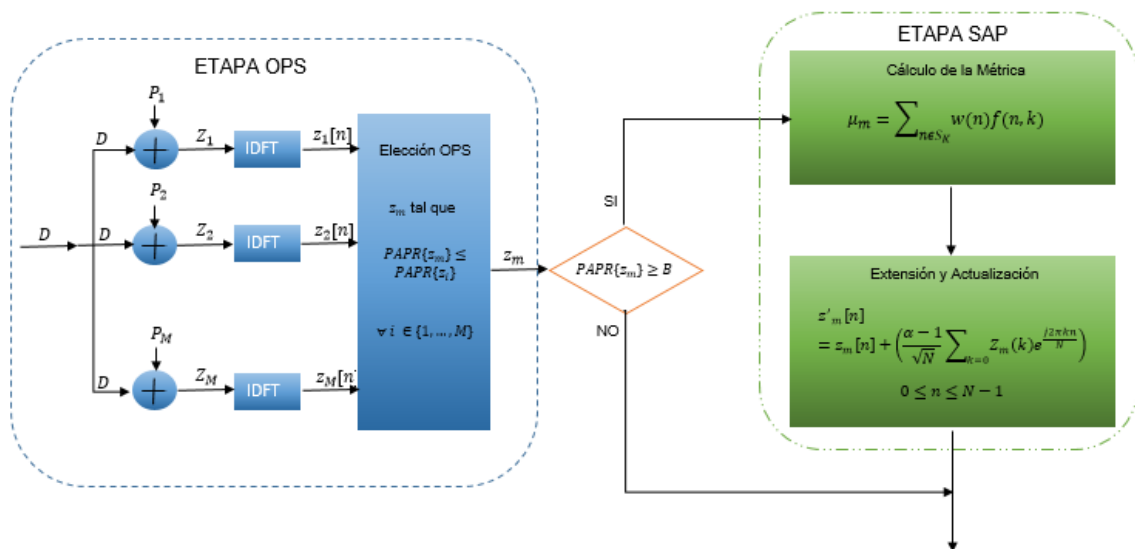


Figura 1.13 Diagrama de la técnica OPS-SAP

1.3.7. MÉTRICAS PARA LA EVALUACIÓN DE LA PAPR

Para evaluar los efectos producidos por la elevada PAPR en señales OFDM, es necesario definir ciertos parámetros de cuantificación.

1.3.7.1. Complementary Cumulative Distribution

La PAPR es una variable aleatoria y la forma de evaluar el rendimiento es mediante la función de distribución acumulativa complementaria (CCDF), que determina la probabilidad de que la PAPR de un determinado símbolo OFDM (donde N son muestras independientes) sea mayor a un umbral fijo (W_0). La CCDF de la PAPR es representada por la siguiente expresión (1.26) [36].

$$CCDF(W) = \Pr(W > W_0) = 1 - (1 - e^{-W_0})^N \quad (1.26)$$

La Figura 1.15 muestra las gráficas de la CCDF de la PAPR para las señales tratadas con OPS, SAP, OPS-SAP y la señal original (sin técnica) para un esquema de modulación BPSK y tasa de codificación igual a $r = 1/2$. Dónde se aprecia claramente que la señal original posee la PAPR más elevada; es así que se evalúa la reducción de la PAPR por parte de las diferentes técnicas respecto a la original.

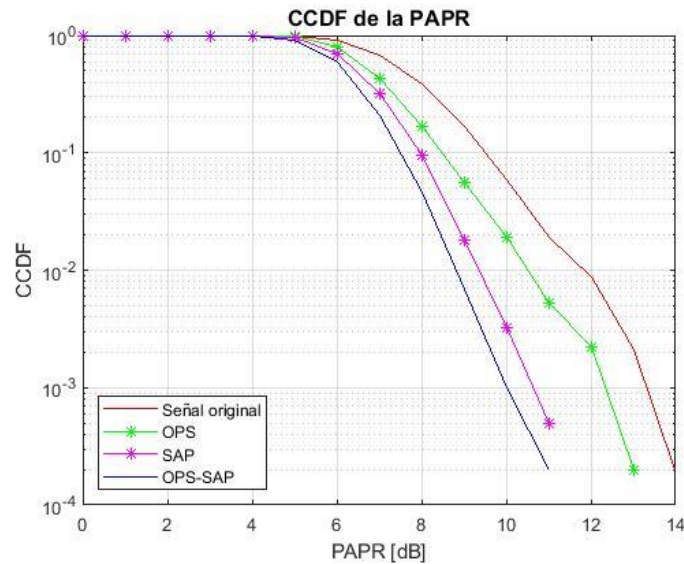


Figura 1.14 CCDF de la PAPR para un esquema de modulación BPSK y tasa de codificación $r = 1/2$.

1.3.7.2. Power Spectral Density (PSD)

La PSD de una señal describe la potencia presente en la señal en función de la frecuencia. El objetivo de la PSD es evaluar las distorsiones introducidas por parte del HPA cuando una señal OFDM con PAPR elevada es amplificada.

La Figura 1.16. muestra que la señal original (antes de pasar por el HPA) presenta menos distorsiones con respecto a la señal que atraviesa el HPA. Cuando se trata la señal con las diferentes técnicas para la reducción de la PAPR y atraviesa por el HPA la señal se asemeja bastante a la señal original antes de ingresar al HPA.

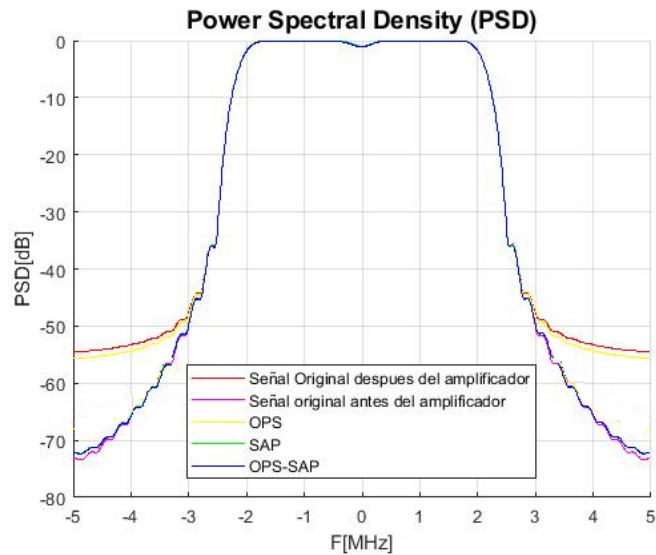


Figura 1.15 PSD de una señal OFDM

1.3.7.3. Bit Error Rate (BER)

La BER corresponde a la cantidad de bits errados respecto al total recibidos para un mismo valor de SNR (*Signal To Noise Ratio*), en otras palabras, es la relación entre bits errados (Bit_{err}) y el total de bits recibidos (Bit_{total}). La BER puede ser expresada de forma adimensional (veces) o mediante decibelios como se indica en la ecuación (1.27) [3].

$$BER = 10 \log_{10} \left(\frac{Bit_{err}}{Bit_{total}} \right) \quad (1.27)$$

La Figura 1.17, muestra las gráficas de BER vs SNR para las velocidades de 6 y 9 Mbps pertenecientes al estándar IEEE 802.11p, dónde las señales con la técnica OPS-SAP presentan una ganancia en la SNR con respecto a la señal original (sin técnica), comprobando que no existe una degradación en la BER y que la PAPR disminuyó.

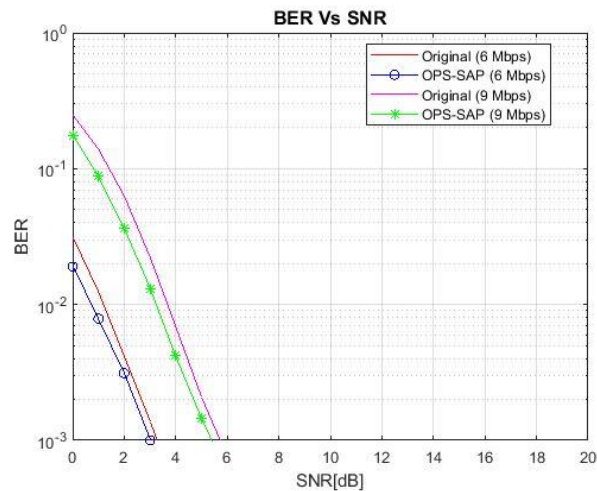


Figura 1.16 BER vs SNR

1.3.8. CANAL INALÁMBRICO

El canal inalámbrico a diferencia del canal alámbrico posee ciertas características que lo vuelven dinámico e impredecible, incrementando la dificultad para realizar el análisis del sistema de comunicación inalámbrico.

Existen tres factores que deben ser tomados en cuenta para el estudio del comportamiento del canal inalámbrico, el primero es el desvanecimiento, el segundo corresponde a la propagación de la señal por el canal y el tercero es la interferencia existente entre el emisor y receptor [38].

a) Interferencia

La interferencia en el canal inalámbrico es ocasionada por fenómenos físicos tales como: absorción, reflexión, difracción, etc. produciendo en el receptor un estado de incertidumbre ya que existe el desconocimiento del estado de los datos que se encuentran en el receptor.

b) Propagación de la señal por un canal inalámbrico

Los efectos que aparecen en la propagación de la señal por el canal inalámbrico corresponden principalmente a la atenuación, multitrayecto y ruido.

La atenuación hace referencia a la reducción en la intensidad de la señal provocando que la señal se distorsione [39].

El multitrayecto provoca que diferentes versiones de la señal lleguen al receptor produciéndose variaciones en la intensidad de la señal, distorsiones y desvanecimiento a pequeña escala.

El ruido corresponde aquellas señales no deseadas que se introducen en algún momento y/o lugar entre el emisor y receptor, provocando que se altere la señal original dando como resultado que el desempeño del sistema de comunicaciones se reduzca [40]. El ruido térmico es el más común en comunicaciones inalámbricas.

El origen del ruido térmico ocurre por el movimiento de los electrones en el medio conductor, este fenómeno se extiende por todo el rango de frecuencias siguiendo una distribución de Gauss [41]. Dicho ruido se distribuye uniformemente por el ancho de banda empleado en los sistemas de comunicación de ahí que es conocido como ruido blanco [40].

1.3.8.1. Desvanecimiento

El desvanecimiento de la señal aparece cuando ocurren variaciones en la amplitud ya sea en el dominio de la frecuencia o del tiempo. El desvanecimiento es considerado como una fuente de degradación de la señal caracterizado como una perturbación de la señal no aditiva en el canal inalámbrico [38].

El origen del desvanecimiento ocurre por varias causas entre las que se tienen las siguientes:

- Desvanecimiento por multitrayectoria: Se produce porque la señal viaja por diferentes trayectorias.
- Desvanecimiento por sombra: Se produce por los diferentes obstáculos que afectan la propagación de las ondas transmitidas.

El desvanecimiento es clasificado en dos tipos: desvanecimiento a gran escala y pequeña escala.

a) Desvanecimiento a gran escala

Este tipo de desvanecimiento aparece cuando el móvil se desplaza por grandes distancias, por lo tanto, el desvanecimiento es provocado por la pérdida en la potencia de la señal en función de la distancia conocido como *Path Loss* y por la sombra perteneciente a objetos de gran tamaño conocido como *Shadowing* [38].

b) Desvanecimiento de pequeña escala

El desvanecimiento de pequeña escala se caracteriza por la rápida variación de la potencia de la señal cuando la estación móvil se desplaza pequeñas distancias. Dicho fenómeno aparece por la interferencia constructiva o destructiva debido al multitrayecto ya que las diferentes versiones de la señal llegan al receptor con diferentes fases. El desvanecimiento

de pequeña escala es atribuido a la multitrayectoria, velocidad del móvil, velocidad de los objetos circundantes y el ancho de banda de la señal [38].

Para representar el canal inalámbrico con desvanecimiento a pequeña escala existen varias formas, sin embargo, el más utilizado es PDP (*Power Delay Profile*).

1.3.8.2. Modelos de Propagación a pequeña escala

- Desvanecimiento Rayleigh

El desvanecimiento Rayleigh corresponde a un modelo estadístico que describe el efecto que se produce en una señal de radio que sigue una distribución Rayleigh. Ocurre en ambientes en los cuales no existe línea de vista entre las antenas del transmisor y receptor [43].

La función de densidad de probabilidad Rayleigh se describe en la ecuación (1.28) [43].

$$PDF_{ray} = \begin{cases} \frac{v}{\vartheta^2} \exp\left(-\frac{v^2}{2\vartheta^2}\right) & 0 \leq v \leq \infty \\ 0 & v < 0 \end{cases} \quad (1.28)$$

Dónde ϑ es el valor RMS del voltaje de la señal recibida antes de la detección de la envolvente, $\frac{v^2}{2}$ es la potencia instantánea y ϑ^2 corresponde a la potencia promedio en el dominio del tiempo de la señal en recepción.

- Desvanecimiento Rician

Este tipo de desvanecimiento describe la señal de radio que llega al receptor por varios caminos (multitrayecto), pero existe una componente de la señal que predomina sobre las otras, dicha señal viaja en línea recta, por lo tanto, existe línea de vista entre el transmisor y receptor [43].

La función de densidad de probabilidad Rician se presenta en la ecuación (1.29) [43].

$$PDF_{rician} = \begin{cases} \frac{v}{\vartheta^2} \exp\left(-\frac{v^2+A^2}{2\vartheta^2}\right) L_0\left(A \cdot \frac{v}{\vartheta^2}\right) & 0 \leq A, 0 \leq v \\ 0 & v < 0 \end{cases} \quad (1.29)$$

Donde A es la amplitud pico de la señal dominante, v es la amplitud de la envolvente de la señal recibida, $L_0\left(A \cdot \frac{v}{\vartheta^2}\right)$ representa la función de *Bessel* de primer tipo y orden cero, $\frac{v^2}{2}$ corresponde a la potencia instantánea y ϑ es la desviación estándar de la potencia local.

- Power Delay Profile (PDP)

El perfil de retardo de potencia hace referencia a la distribución de la potencia de la señal recibida a través de un canal multitrayecto en función de los diferentes retardos de propagación [38].

Los valores que determinan el PDP son suministrados por tablas, cuya información es obtenida de valores empíricos por tal razón no es una representación precisa del entorno real [42].

La Tabla 1.5 muestra cómo se representa el PDP para el estándar IEEE 802.11p.

Tabla 1.5 Valores de potencia promedio y retardo relativo para el estándar IEEE 802.11p [48].

| Parámetro | Trayecto 1 | Trayecto 2 | Trayecto 3 | Trayecto 4 | Unidades |
|-----------|------------|------------|------------|------------|----------|
| Potencia | 0 | -2 | -5 | -7 | dB |
| Retardo | 0 | 200 | 433 | 700 | ns |

El PDP define tres parámetros de dispersión de tiempo los cuales son: Dispersión del retardo promedio (*Mean Excess Delay*), dispersión del retardo RMS (*RMS Delay Spread*) y máxima dispersión del retardo (*Excess Delay Spread*) [43], para sistemas OFDM se establece que la máxima dispersión de retardo es el valor del intervalo de guarda es decir para IEEE 802.11a e IEEE 802.11g es 0.8us y para IEEE 802.11p es 1.6us [43].

2. METODOLOGÍA

En este capítulo se describe el proceso para el diseño, implementación y funcionamiento de la aplicación para la evaluación del rendimiento de la técnica OPS-SAP para la reducción de la PAPR en redes Ad-Hoc en el entorno de desarrollo de *App Designer* propiedad de Matlab.

En primer lugar, se desarrolla la aplicación en su entorno visual, definiendo claramente los parámetros de ingreso por parte del usuario, así como la presentación de las gráficas con los resultados obtenidos.

A continuación, se implementa la estructura de la trama PDU cuyo formato se encuentra descrito en la Figura 1.2 [3]. Una vez definido el formato se implementa tanto el transmisor como receptor según lo indica el estándar IEEE 802.11 [3]. El comportamiento del canal inalámbrico considerado en esta aplicación es con ruido AWGN, desvanecimiento Rician y Rayleigh. Las técnicas de reducción de la PAPR son implementadas según se indica en [1] [4] [5]. Finalmente se diseñan e implementan las funciones para presentar los resultados en función de curvas de BER vs SNR, CCDF de la PAPR y PSD.

Para terminar, el funcionamiento de la aplicación.

2.1. MATLAB

Matlab es un entorno de trabajo cuya forma de programación es orientada a objetos, cuenta con sus propias librerías, *toolbox* y lenguaje de programación. La forma más conocida de programación en Matlab consiste en scripts, pero también brinda la posibilidad de construir aplicaciones gráficas ya sea con GUIDE (*Graphic User Interface*) o en el entorno de desarrollo de aplicaciones denominado *App Designer*.

GUIDE existe hace varios años, mientras que a partir del año 2016 se introdujo *App Designer* [44]. *App Designer* es el reemplazo en el futuro de GUIDE, la documentación perteneciente a Matlab sugiere el uso de *App Designer* para la creación de aplicaciones ya sean web o de escritorio, puesto que presenta mejores componentes, interfaz más moderna y además brinda la posibilidad de pasar las aplicaciones desarrolladas en *GUIDE* a *App Designer* sin mucho esfuerzo de por medio [45].

App Designer y GUIDE presentan diferentes estructuras de código como, por ejemplo: sintaxis para la invocación de código, así como el ingreso y manipulación de los

componentes de la interfaz y la forma de compartir información (datos). Las diferencias presentes se resumen en la Tabla 2.1.

Tabla 2.1 Diferencias entre GUIDE y App Designer [46].

| Diferencia | GUIDE | APP DESIGNER |
|---|--|---|
| Uso de figuras y gráficos | GUIDE llama a la función para crear la ventana de la aplicación. <code>figure</code> GUIDE llama a la función para crear ejes para mostrar trazados. <code>Axes</code> Todas las funciones gráficas son compatibles. No es necesario especificar los ejes de destino. | El diseñador de aplicaciones llama a la función para crear la ventana de aplicación. <code>uifigure</code> El diseñador de aplicaciones llama a la función para crear ejes para mostrar trazados. <code>uiaxes</code> La mayoría de las funciones gráficas son compatibles. Debe especificar los ejes de destino. |
| Uso de componentes | GUIDE crea la mayoría de los componentes con la función. <code>Uicontrol</code> , existen menos componentes disponibles. | El diseñador de aplicaciones crea cada componente de interfaz de usuario con su propia función dedicada. Existen más componentes disponibles. <code>.TreeGaugeTabGroupDatePicker</code> |
| Acceso a propiedades de componente | GUIDE utiliza "get" para acceder a las propiedades de los componentes, para especificar un componente utiliza. <code>setgethandles</code> Por ejemplo, <code>name = get(handles.Fig, 'Name')</code> | El diseñador de aplicaciones admite y fomenta el uso de la notación de puntos para acceder a las propiedades de los componentes y los utiliza para especificar un componente. <code>setgetapp</code> Por ejemplo, <code>name = app.UIFigure.Name</code> |
| Administración de código de aplicación | El código se define como una función principal que puede llamar a las funciones locales. Todo el código es editable. | El código se define como una clase. Solo se pueden editar las devoluciones de llamada, las funciones auxiliares y las propiedades personalizadas. |
| Escribir callbacks | Los argumentos de entrada y devolución de llamada requeridos son: <code>.handlesObjecteventdata</code> Por ejemplo, <code>myCallback(hObject,eventdata,handles)</code> | Los argumentos de entrada y devolución de llamada requeridos son: <code>.appevent</code> Por ejemplo, <code>myCallback(app,event)</code> |
| Compartir datos | Para almacenar y compartir datos de entrada, devoluciones de llamada y funciones se utiliza la estructura de la siguiente forma: <code>.UserDatahandlesguidatasetappdatagetappdata</code> . Por ejemplo: <code>Handles.currSelection = selection;</code> <code>guidata(hObject,handles);</code> | Para almacenar y compartir datos entre devoluciones de llamada y funciones, se utiliza propiedades personalizadas para crear variables. Por ejemplo, <code>app.currSelection = selection</code> |
| Ventajas adicionales | | El entorno para editar la aplicación es robusto y proporciona un flujo de trabajo. Lo más importante se tiene la posibilidad de crear y compartir una aplicación de escritorio o web independientemente. |

2.1.1. APP DESIGNER

App Designer brinda la posibilidad de crear aplicaciones profesionales sin la necesidad de ser un desarrollador de software profesional.

El ambiente de desarrollo es bastante amigable con el usuario ya que solo se deben arrastrar los componentes para utilizarlos en la interfaz gráfica y utilizar las guías de alineación para conseguir un diseño preciso, además existen opciones de redistribución automática para que la aplicación se adapte a diferentes tamaños sin requerir una programación lógica de redimensionamiento compleja [47].

El editor integrado permite programar rápidamente el comportamiento de la aplicación, dando la oportunidad de detectar problemas de codificación mediante *Code Analyzer*, es decir a medida que se codifica aparecen mensajes de advertencia que indican la existencia de errores evitando posibles eventualidades [47].

La librería de componentes brinda la creación de aplicaciones que no poseen solo componentes estándar como botones, casillas de verificación y listas desplegadas, sino elementos de control como: medidores, indicadores luminosos, controles y conmutadores que brindan la posibilidad de replicar las acciones de paneles de instrumentación. Existen componentes de contenedor como es el caso de pestañas y paneles para organizar de mejor manera la interfaz de usuario [47].

Las interacciones entre los diferentes componentes de la aplicación son personalizadas mediante *callbacks* que se ejecutan cuando el usuario interactúa con la aplicación. Existe la posibilidad de utilizar gráficos en 2D y 3D, así como tablas para la presentación de resultados de manera interactiva [47].

Las aplicaciones trabajan de forma independiente mediante *Matlab Compiler* dando la posibilidad de ser compartidas con otros usuarios. Finalmente, las aplicaciones pueden ser creadas como aplicaciones web y compartirlas a través de una URL exclusiva, dando la posibilidad a los usuarios de ingresar directamente a las aplicaciones a través de un navegador sin instalar software adicional [47].

2.1.2. ENTORNO DE TRABAJO DE APP DESIGNER

Para crear una aplicación con *App Designer* se requiere iniciar sesión en el entorno de *Matlab*. Una vez iniciado *Matlab* para crear una aplicación se elige la pestaña HOME a continuación se selecciona el ícono NEW, de todas las opciones que se despliegan se elige *App* Figura 2.1, otra forma de acceder es por medio del terminal, una vez en el terminal se digita “*appdesigner*” directamente.

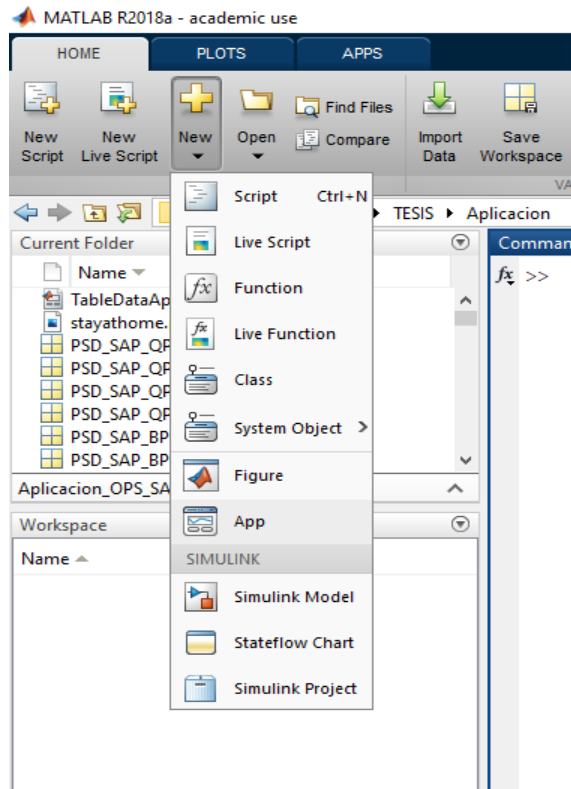


Figura 2.1 Ícono New con la opción App.

El entorno de trabajo de App Designer posee varias herramientas entre las cuales existen dos principales; la primera abarca todo lo referente al diseño de la parte visual de la aplicación y se ubica en la ventana *DESIGNER* Figura 2.2, mientras tanto, la segunda es el *EDITOR* Figura 2.3, dónde se realiza todo lo referente a la programación de la aplicación.

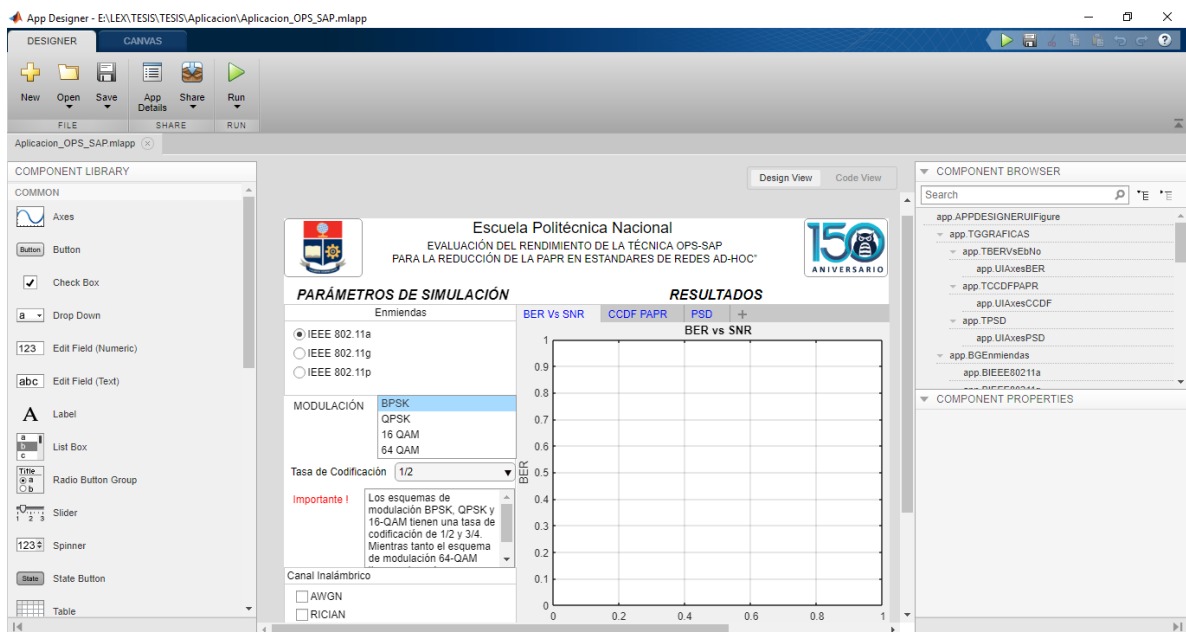


Figura 2.2 Ventana correspondiente a DESIGNER

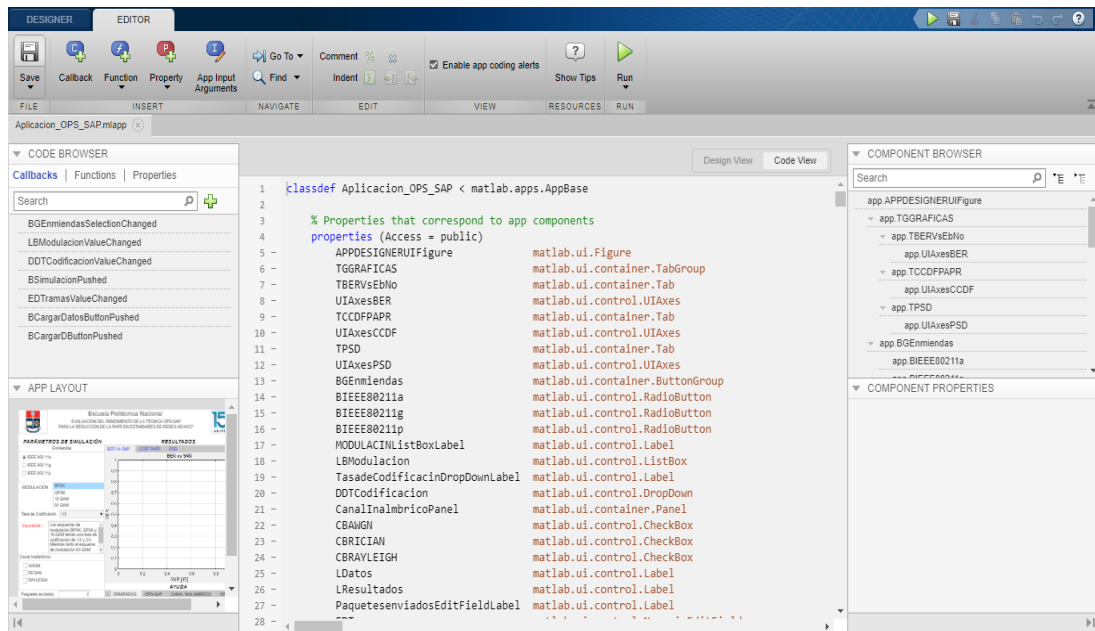


Figura 2.3 Ventana perteneciente al EDITOR

Todo lo referente a la ventana *DESIGNER* se encuentra en *Design View* que proporciona una amplia biblioteca de componentes para el diseño de la interfaz de usuario, dando como resultado la creación de una variedad de características interactivas. Los cambios que se realizan en la parte visual (*Design View*) de la aplicación se reflejan automáticamente en *Code View* que corresponde a la ventana del *EDITOR*.

La ventana que pertenece a *Desing View*, en la parte izquierda se encuentra la ventana *Component Library* donde se encuentran todos los objetos para diseñar la aplicación. En la parte superior derecha se encuentra el *Component Browser* donde se encuentran los objetos que forman parte de la aplicación, cada objeto posee su propio identificador el cual puede ser modificado según se requiera. El *Inspector Component Browser* permite personalizar la apariencia de algún componente previamente seleccionado.

Por otra parte, todo lo que corresponde al *EDITOR* se encuentra en la ventana correspondiente a *Code View*, en este lugar existen diferentes ventanas que ayudan en el proceso de codificación. En la parte izquierda superior de *Code View* se ubica el *Code Browser*, dónde se encuentran todos los *callbacks* y funciones que se crearon para llevar a cabo el funcionamiento de la aplicación.

2.2. APLICACIÓN

La aplicación desarrollada para evaluar el rendimiento de la técnica OPS-SAP para la reducción de la PAPR en redes Ad-Hoc, cuenta con tres zonas específicas, la primera pertenece al ingreso de datos por parte del usuario para cada escenario de simulación, la

segunda es la presentación de resultados y la tercera zona corresponde a la ayuda brindada por el programador con el objetivo que el usuario conozca sobre los procedimientos que se realizan e intervienen en la simulación.

2.2.1. DISEÑO

La parte visual de la aplicación cuenta con varios elementos como son: Botones, casillas, campos de texto y listas desplegables; estos elementos se utilizarán para el ingreso de datos por parte del usuario. Para mostrar resultados se utiliza *Graphics* que sirve para mostrar todo tipo de gráficas. Finalmente se cuenta con paneles y pestañas para distribuir de mejor manera los objetos que forman parte de la aplicación. El diseño de la parte visual de la aplicación se muestra en la Figura 2.4, dónde se identifican las tres zonas fundamentales de la aplicación, la zona de selección de parámetros, la correspondiente a la presentación de resultados y la zona para brindar información al usuario.

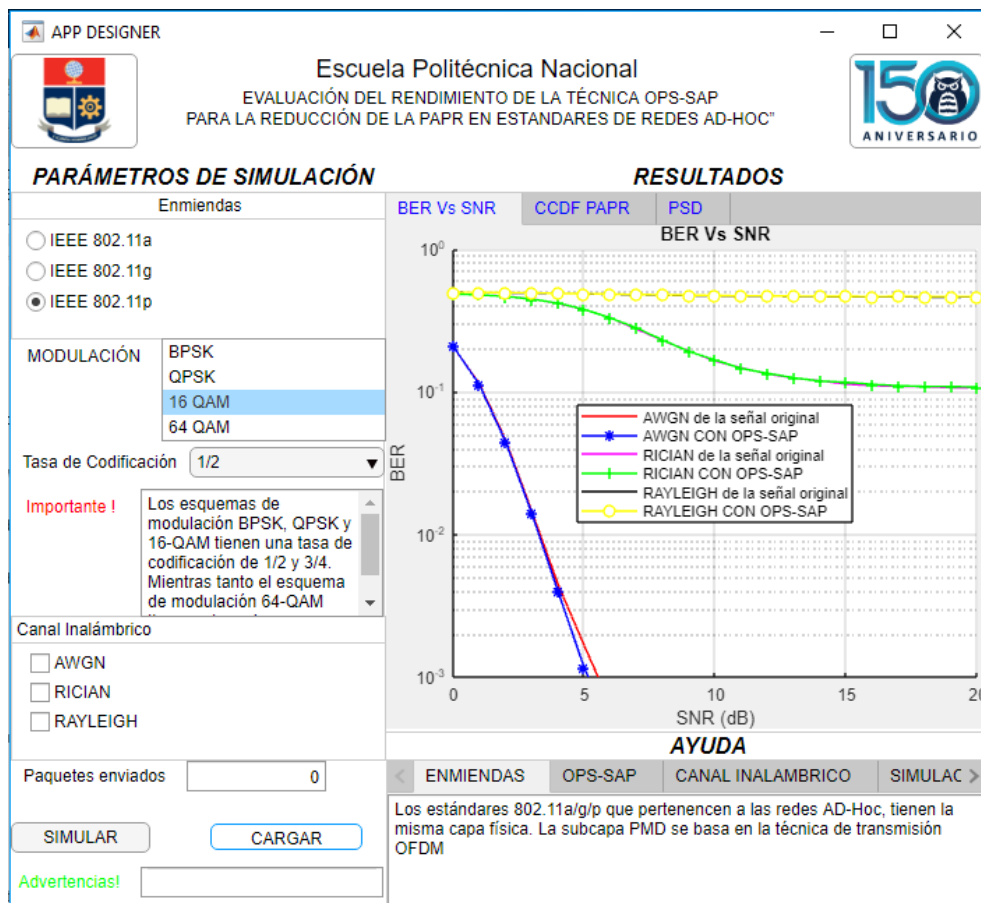


Figura 2.4 Diseño visual de la aplicación.

2.2.2. FUNCIONAMIENTO

El funcionamiento de la aplicación se realiza de forma modular, el proceso de envío y recepción de datos se alojado en el *callback* perteneciente al botón “SIMULAR” en este lugar convergen todas las funciones necesarias para desarrollar el proceso mencionado.

Para iniciar el proceso es necesario que se ingresen los diferentes parámetros que intervienen en la transmisión, una vez ingresados los parámetros se presiona el botón “SIMULAR”. En cuanto termina el proceso de transmisión-recepción se presentan los resultados.

Las funciones necesarias para cada bloque del transmisor, canal inalámbrico, receptor y técnica para la reducción de la PAPR convergen en el botón “SIMULAR”.

2.3. CALLBACK PRINCIPAL

El *callback* principal pertenece al botón “SIMULAR”, formado por todas las funciones necesarias para obtener el funcionamiento del transmisor y receptor, además en este lugar también se encuentran las funciones necesarias para mostrar los resultados de la simulación.

Las tareas específicas que se realizan son:

Receptar los valores de los parámetros de transmisión, los cuales son ingresados por el usuario como son: estándar, tipo de modulación, tasa de codificación, canal inalámbrico y número de tramas a ser enviadas.

Llamar a cada una de las funciones que conforman el transmisor, receptor y canal, así como aquellas que almacenan valores e inicializan variables.

Simular el envío y recepción de las tramas dependiendo de los parámetros ingresados previamente para un mismo rango de SNR.

Guardar la información necesaria para presentar los resultados.

Presentar los resultados en términos de curvas de: BER vs SNR, CCDF de la PAPR y PSD.

El proceso que realiza el *callback* principal se presenta en la Figura 2.5.

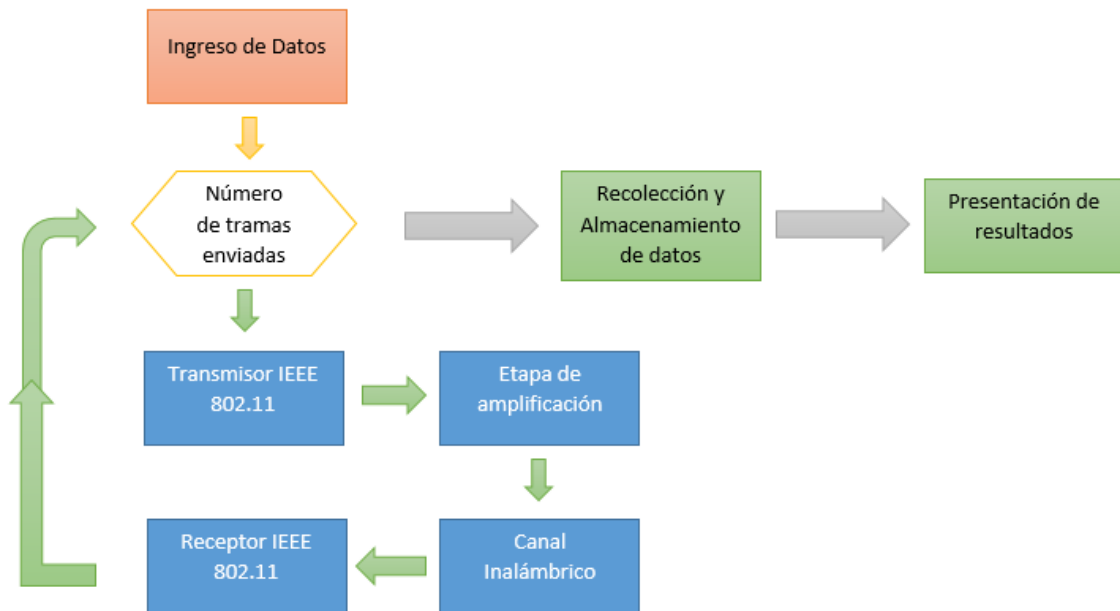


Figura 2.5 Proceso que lleva a cabo el *callback* principal.

2.4. TRANSMISOR

El transmisor genera la trama PPDU PLCP y somete dicha trama a varios procesos propios de la subcapa PLCP y PMD la cual siguen procesos que contemplan: aleatorización, codificación, entrelazado, modulación, construcción del símbolo OFDM dónde se aplica la técnica de reducción de la PAPR OPS-SAP. Finalmente, la información se encuentra lista para ser enviada por el canal inalámbrico.

2.4.1. FUENTE

La fuente se encarga de generar el número de bits suficientes para el preámbulo, cabecera y DATA.

2.4.2. CREACIÓN DEL PREAMBULO

El preámbulo se encuentra formado a su vez por un preámbulo corto y uno largo como se indica en [3]. La sintaxis para generar el preámbulo se presenta a continuación.

$$\text{PreambuloTotal}=[\text{Pcorto PLargo}]$$

2.4.2.1. Preámbulo Corto

Se genera un arreglo conformado por 53 bits los cuales son multiplicados por un factor de normalización, a dichos bits se les aplica la IFFT. Finalmente, el preámbulo corto se forma por 10 secuencias cortas que son el primer $\frac{1}{4}$ del arreglo descrito en el estándar IEEE 802.11 [3]. La función encargada de formar el preámbulo corto es `CrearPreambuloCorto()` y la sintaxis es la siguiente:

`Pcorto=CrearPreambuloCorto(app)`

Dónde, `Pcorto` es el preámbulo corto en tiempo de la trama PDU PLCP.

El argumento `app` es propio del entorno de desarrollo de App Designer, todas las funciones implementadas en este entorno deben tener este argumento de entrada.

El diagrama de flujo que describe a la función anterior se encuentra en la Figura A.1 del anexo A y el código se encuentra en la sección B.1 del anexo B.

2.4.2.2. Preámbulo Largo

Se genera un arreglo conformado por 53 bits los cuales pasan por IFFT, se genera un prefijo cíclico cuya longitud es de 16 bits. Finalmente, el preámbulo largo se encuentra formado por dos secuencias largas cuya longitud es la mitad del arreglo más el prefijo cíclico tal como se describe en el estándar IEEE 802.11 [3].

La función encargada de formar el preámbulo largo es `CrearPremabuloLargo()` y a continuación se presenta su sintaxis:

`PLargo=CrearPreambuloLargo(app)`

Dónde, la salida es el preámbulo largo en tiempo de la trama PDU PLCP (`PLargo`)

El diagrama de flujo que describe a la función anterior se encuentra en la Figura A.2 del anexo A y el código se encuentra en la sección B.2 del anexo B.

2.4.3. CREACIÓN DE LA CABECERA

La cabecera de la trama PDU PLCP está constituida por 24 bits y pasa por varios procesos que están representados por funciones específicas como; codificación (`codificarString()`), entrelazado (`entrelazarString()`), modulado (`modularString()`), y bloque OFDM (`bloqueOFDM()`). Al finalizar el proceso se obtiene un símbolo OFDM, cabe señalar que la cabecera es codificada únicamente con una tasa de codificación $r = 1/2$ y modulación BPSK como lo indica el estándar IEEE 802.11 [3].

El arreglo de bits que representa la cabecera PDU PLCP es "SIGNAL".

El diagrama de flujo que describe la creación de la cabecera "SIGNAL" se encuentra en la Figura A.3 del anexo A y el código se localiza en la sección B.3 del anexo B.

2.4.4. CREACIÓN DEL CAMPO DATA

El campo data de la trama PDU PLCP está constituido por cuatro subcampos: *SERVICE*, *TAIL*, *PSDU* y *PAD*; las tramas siempre forman un número entero de símbolos OFDM, por

lo tanto, los bits que constituyen este campo deben atravesar por diferentes procesos para ser símbolos OFDM, que serán transmitidos a través del canal inalámbrico.

Las funciones por las cuales pasan los bits son: aleatorización (`scrambleString()`), codificación (`codificarString()`), entrelazado (`entrelazarString()`), modulado (`modularString()`) y bloque OFDM (`bloqueOFDM()`). El procedimiento se realiza para todas las tasas de codificación y esquemas de modulación como se indica en el estándar IEEE 802.11 [3].

El arreglo de bits que representa el campo data PDU PLCP toma el nombre de "DATA".

El diagrama de flujo que describe la creación del campo DATA se encuentra en la Figura A.4 del anexo A y el código se encuentra en la sección B.4 del anexo B.

2.4.5. ALEATORIZADOR

Este bloque se encarga de realizar la aleatorización de los bits pertenecientes al campo DATA, en la cabecera de la trama PDU PLCP no se realiza este procedimiento como lo indica [3]. La función encargada del proceso de aleatorización es `scrambleString()` cuya sintaxis es la siguiente:

$$SDatos = \text{scrambleString}(\text{app}, \text{datos}, \text{estadoInicial})$$

Dónde a la entrada de la función se tienen los Bits de datos (`datos`) y el estado inicial (`estadoInicial`).

Obteniendo a la salida los bits aleatorizados (`SDatos`).

El diagrama de flujo que describe al aleatorizador se encuentra en la Figura A.5 del anexo A y el código se encuentra en la sección B.5 del anexo B.

Para la implementación de la función `scrambleString()` se utilizó dos comandos especiales:

$$C = \text{xor}(A, B)$$

El comando "xor" corresponde al operador lógico or exclusiva cuyos parámetros de entrada A y B pueden ser: escalares, vectores y matrices. Los parámetros de entrada deben ser del mismo tamaño o de un tamaño compatible entre ellos.

$$Y = \text{circshift}(A, K, \text{dim})$$

El comando "circshift" realiza un desplazamiento de forma circular. Dónde, los valores de la matriz A se desplazan por las posiciones de K a lo largo de la dimensión dim, los valores de K y dim deben ser escalares.

2.4.6. CODIFICACIÓN

Este bloque se encarga de realizar la codificación de los bits del campo DATA y cabecera de la trama PDU PLCP. El campo DATA es codificado con las siguientes tasas de codificación: $r = 1/2, 3/4$ y $2/3$, mientras que la cabecera únicamente es codificada con una tasa de codificación de $r = 1/2$. La función que realiza la codificación de canal es `codificarString()` y a continuación se presenta su sintaxis:

$$\text{CDatos} = \text{codificarString}(\text{app}, \text{datos}, R)$$

Los bits de datos (`datos`) y tasa de codificación (`R`) son los parámetros de entrada. Los bits codificados (`CDatos`) se obtiene al finalizar el proceso de codificación.

El diagrama de flujo que describe al codificador se encuentra en la Figura A.6 del anexo A y el código se encuentra en la sección B.6 del anexo B.

La implementación de la función `codificarString()` requiere de tres comandos para llevar a cabo su funcionamiento, los cuales se detallan a continuación:

Para obtener la estructura de trellis se utiliza el comando `poly2trellis`

$$\text{TRE} = \text{poly2trellis}(k, g)$$

El cual retorna la descripción de la estructura de trellis. Dónde `k` es el valor correspondiente al retraso de las secuencias de entrada al codificador y `g` especifica los polinomios generadores.

$$\text{PUNT} = \text{fliplr}(A)$$

El comando `fliplr` retorna un vector con la misma longitud de `A`, solo que sus elementos se encuentran invertidos en su orden.

Para obtener los bits codificados con codificación convolucional se emplea el siguiente comando:

$$\text{CD} = \text{convenc}(\text{datos}, \text{trellis}, \text{punct})$$

El comando `convenc` realiza la codificación convolucional cuyos argumentos de entrada son: bits a ser codificados (`datos`), la estructura de trellis (`trellis`) y un patrón de puntura (`punct`).

2.4.7. ENTRELAZADO

Este bloque se encarga de realizar el entrelazado de los bits teniendo en cuenta el número de bits codificados por símbolo (NCBPS) para el procesamiento grupo a grupo. Cada grupo es sometido a una primera y segunda permutación como lo indica IEEE 802.11 [3].

La función encargada de realizar el proceso de entrelazado es `entrelazarString()`, su formato es el siguiente:

$$EDatos = \text{entrelazarString}(\text{app}, \text{datos}, \text{NCBPS})$$

La entrada de esta función contempla: Bits de datos (`datos`) y número de bits codificados por símbolo (NCBPS). Mientras que, el resultado a la salida de la función son los bits entrelazados (`EDatos`).

El diagrama de flujo que describe al entrelazado se encuentra en la Figura A.7 del anexo A y el código se encuentra en la sección B.7 del anexo B.

La implementación de la función anterior requiere del comando denominado “floor” cuya estructura se detalla a continuación:

$$Y = \text{floor}(X)$$

Dónde, floor redondea cada elemento de X al entero más cercano menor o igual a ese elemento.

2.4.8. MODULACIÓN

En este bloque se realiza la modulación de los datos teniendo en cuenta el esquema de modulación M , cabe mencionar que el campo DATA admite modulaciones BPSK, QPSK, 16-QAM y 64-QAM, mientras que la cabecera de la trama PPDU PLCP admite únicamente el esquema de modulación BPSK.

Una vez ejecutada la función `modularString()`, se obtienen los datos modulados. La sintaxis de la función es la siguiente:

$$MDatos = \text{modularString}(\text{app}, \text{datos}, M)$$

Dónde los argumentos de entrada son los bits de datos (`datos`) y el esquema de modulación (M). Obteniendo a la salida los símbolos complejos (`MDatos`).

El diagrama de flujo que describe a la modulación se encuentra en la Figura A.8 del anexo A y el código se encuentra en la sección B.8 del anexo B.

a) Modulación BPSK

Este esquema de modulación asigna a cada bit de entrada un símbolo complejo de acuerdo al diagrama de constelación mostrado en la Figura 1.5. La función encargada de realizar una modulación BPSK es `modularBPSK()` y su formato es el siguiente:

$$S_{\text{Modulado}} = \text{modulacionBPSK}(\text{app}, \text{datos})$$

Donde, los bits de datos (`datos`) son los parámetros de entrada y los símbolos complejos con modulación BPSK en frecuencia (`SModulado`) corresponden el resultado de la función.

El diagrama de flujo que describe la modulación BPSK se encuentra en la Figura A.9 del anexo A y el código se encuentra en la sección B.9 del anexo B.

b) Modulación QPSK

El esquema de modulación QPSK asigna a cada dos bits de entrada un símbolo complejo como se indica en el diagrama de constelación presentado en la Figura 1.6. La función encargada de realizar una modulación QPSK es `modularQPSK()` y la sintaxis es la siguiente:

$$S_{\text{Modulado}} = \text{modulacionQPSK}(\text{app}, \text{datos})$$

Dónde, los bits de datos (`datos`) son los parámetros de entrada y los símbolos complejos con modulación QPSK en frecuencia (`SModulado`) corresponden el resultado de la función.

El diagrama de flujo que describe a la modulación QPSK se encuentra en la Figura A.10 del anexo A y el código se encuentra en la sección B.10 del anexo B.

c) Modulación 16-QAM

Este esquema de modulación permite asignar a cada cuatro bits de entrada un símbolo complejo tal como se presenta en el diagrama de constelación Figura 1.7. La función encargada de realizar la modulación 16-QAM es `modular16QAM()` y su sintaxis es:

$$S_{\text{Modulado}} = \text{modulacion16QAM}(\text{app}, \text{datos})$$

Dónde, los bits de datos (`datos`) son los parámetros de entrada y los símbolos complejos con modulación 16-QAM en frecuencia (`SModulado`) corresponden al resultado de la función.

El diagrama de flujo que describe a la modulación 16-QAM se encuentra en la Figura A.11 del anexo A y el código se encuentra en la sección B.11 del anexo B.

d) Modulación 64-QAM

El esquema de modulación 64-QAM asigna a cada seis bits de entrada un símbolo complejo como se indica en la Figura 1.8. La función encargada de realizar la modulación 64-QAM es `modular64QAM()` y su formato es:

$$S_{\text{Modulado}} = \text{modulacion64QAM}(\text{app}, \text{datos})$$

Dónde los bits de datos (`datos`) son los parámetros de entrada y los símbolos complejos con modulación 64-QAM en frecuencia (`SModulado`) corresponden al resultado de la función.

El diagrama de flujo que describe al esquema de modulación 64-QAM se encuentra en la Figura A.12 del anexo A y el código se encuentra en la sección B.12 del anexo B.

2.4.9. BLOQUE OFDM

En este bloque se detalla el procedimiento para obtener un símbolo OFDM en el dominio de la frecuencia, en el dominio del tiempo y finalmente añadir el prefijo cíclico.

A) Generación de los pilotos

Se genera las subportadoras pilotos como lo indica IEEE 802.11 [3] a través de la siguiente función:

$$\text{Polaridad} = \text{generarPolaridadPiloto}(\text{app})$$

Dónde la salida de la función corresponde a los pilotos (`Polaridad`).

El diagrama de flujo que describe a la función anterior se encuentra en la Figura A.13 del anexo A y el código se encuentra en la sección B.13 del anexo B.

B) Formación del símbolo OFDM

Se genera un arreglo con 52 muestras a las cuales se les asignan los datos de entrada (48 símbolos complejos) y los pilotos con su respectiva polaridad (4 pilotos).

El arreglo de bits que representan el símbolo OFDM con las 52 muestras es “`SimboloF`”:

El diagrama de flujo que describe el arreglo de bits se encuentra en la Figura A.14 del anexo A y el código se encuentra en la sección B.14 del anexo B.

Se agrega las subportadoras nulas (12 incluyendo la subportadora DC). El arreglo de bits que representa el símbolo OFDM con las 64 muestras es “`SimboloFArr`”:

El diagrama de flujo que describe a la formación del símbolo OFDM con 64 muestras se encuentra en la Figura A.14 del anexo A y el código se encuentra en la sección B.15 del anexo B.

Finalmente, se realiza la IFFT del símbolo OFDM en el dominio de la frecuencia y se concatena el último $\frac{1}{4}$ del mismo que corresponde al prefijo cíclico como lo indica IEEE 802.11 [3]. Dando como resultado un arreglo de muestras en el dominio del tiempo que serán enviadas por el canal inalámbrico.

El arreglo que representa el símbolo OFDM en el dominio del tiempo es "DATA"

El diagrama de flujo que describe la creación del campo DATA se encuentra en la Figura A.4 del anexo A y el código se encuentra en la sección B.4 del anexo B.

Adicionalmente se pueden obtener símbolos OFDM en el dominio del tiempo modificados con la técnica OPS, SAP y OPS-SAP.

El arreglo que representa el símbolo OFDM en el dominio del tiempo con la técnica OPS es "DATA_OPS".

El diagrama de flujo que describe la creación del campo DATA_OPS se encuentra en la Figura A.15 del anexo A

El arreglo que representa el símbolo OFDM en el dominio del tiempo con la técnica SAP es "DATA_SAP".

El diagrama de flujo que describe la creación del campo DATA-SAP se encuentra en la Figura A.16 del anexo A.

El arreglo que representa el símbolo OFDM en el dominio del tiempo con la técnica OPS-SAP es "DATA_OPS_SAP".

El diagrama de flujo que describe la creación del campo DATA_OPS_SAP se encuentra en la Figura A.17 del anexo A y el código se encuentra en la sección B.16 del anexo B.

Una vez obtenido el campo DATA se forma la trama PPDU PLCP (CABECERA SIGNAL DATA) que será enviada a través del canal inalámbrico.

El arreglo que representa la trama PPDU PLCP en el dominio del tiempo es "TRAMA", mientras que, el arreglo que representa la trama PPDU PLCP en el dominio del tiempo con técnica OPS-SAP es "TRAMA_OPS_SAP".

Por otra parte, el arreglo que representa la trama PPDU PLCP en el dominio del tiempo con técnica SAP es "TRAMA_SAP" y el arreglo que representa la trama PPDU PLCP en el dominio del tiempo con técnica OPS es "TRAMA_OPS".

2.4.10. AMPLIFICADOR DE ALTA POTENCIA HPA

Una vez creada la trama PDU PLCP esta atraviesa por el amplificador de potencia que fue diseñado en base al modelo matemático Rapp [33], Finalmente a la salida del amplificador se tiene la trama PDU PLCP amplificada, la cual se obtiene mediante la función `amplificadorHPA()` cuya sintaxis es:

$$[\text{Samplificada}] = \text{amplificadorHPA}[\text{app}, \text{Trama}, \text{IBO}]$$

Los parámetros de entrada son: el arreglo de bits de la trama PDU PLCP en el dominio del tiempo (Trama) y el valor de input back off en [dB] (IBO).

Como resultado a la salida de la función se tiene el arreglo de bits de la trama PDU PLCP amplificada en el dominio del tiempo (Samplificada).

El diagrama de flujo que describe al amplificador se encuentra en la Figura A.18 del anexo A y el código se encuentra en la sección B.17 del anexo B.

Cabe señalar que las tramas que fueron tratadas con las técnicas OPS, SAP y OPS-SAP también pueden pasar a través del amplificador dando como resultado: `Samplificado OPS`, `Samplificado SAP` y `Samplificado OPS_SAP` respectivamente.

2.5. CANAL INALÁMBRICO

La transmisión de la información se realiza a través de un canal inalámbrico que posee desvanecimiento a pequeña escala; este tipo de canal presenta multitrayecto y dispersión de retardo, por esta razón el canal cuenta con ciertas características dependiendo del canal a modelar; se tienen tres tipos de canales:

Canal con ruido AWGN

Canal con desvanecimiento Rayleigh

Canal con desvanecimiento Rician

Para el canal con desvanecimiento Rayleigh y Rician se utiliza el perfil de retardo de potencia (PDP) el cual define la intensidad de la señal recibida a través del canal multitrayecto. El PDP es representado por dos valores (arreglos de datos); el primero corresponde a las potencias promedio normalizadas de cada uno de los rayos y el segundo representa el retardo de cada uno de los rayos.

Los valores de PDP varían de acuerdo a cada estándar IEEE 802.11a/g/p, los valores de PDP se toman como referencia de [48].

Para cada estándar se presentan los valores de PDP que sirven tanto para modelar el canal con desvanecimiento Rayleigh como con desvanecimiento Rician.

Tabla 2.2 Valores de PDP para el estándar IEEE 802.11a [49].

| Parámetro | Trayecto 1 | Trayecto 2 | Trayecto 3 | Trayecto 4 | Unidades |
|-----------|------------|------------|------------|------------|----------|
| Potencia | 0 | -1 | -3.9 | -6 | dB |
| Retardo | 0 | 2 | 7 | 12 | Us |

Tabla 2.3 Valores de PDP para el estándar IEEE 802.11g [49].

| Parámetro | Trayecto 1 | Trayecto 2 | Trayecto 3 | Trayecto 4 | Unidades |
|-----------|------------|------------|------------|------------|----------|
| Potencia | 0 | -2.4 | -5.5 | -7.5 | dB |
| Retardo | 0 | 4.8 | 10 | 15 | Us |

Tabla 2.4 Valores de PDP para el estándar IEEE 802.11p [48].

| Parámetro | Trayecto 1 | Trayecto 2 | Trayecto 3 | Trayecto 4 | Unidades |
|-----------|------------|------------|------------|------------|----------|
| Potencia | 0 | -2 | -5 | -7 | dB |
| Retardo | 0 | 4 | 9 | 14 | Us |

2.5.1. CANAL RAYLEIGH

El canal Rayleigh se implementa teniendo en cuenta cuatro rayos con su respectiva potencia y retardo. Para la construcción del canal Rayleigh se crea un objeto, dicho objeto es creado empleando el Toolbox “comm.RayleighChannel” perteneciente a Matlab.

El toolbox “comm.RayleighChannel” posee varios parámetros que intervienen en su modelación, los cuales se describen a continuación:

SampleRate: Corresponde a la frecuencia de muestreo de la señal.

PathDelays: Hace referencia al vector de retardos de cada uno de los rayos, su valor se expresa en segundos.

AveragePathGain: Este parámetro es el vector de potencias de cada una de las señales expresado en dB.

MaximumDopplerShif: Representa el desplazamiento *Doppler*.

Una vez creado el objeto con sus respectivos parámetros, se aplica a la secuencia de bits (Trama PPDU PLCP) que se transmiten. Los bits una vez que pasan a través del canal con desvanecimiento Rayleigh se les agrega ruido mediante el comando awgn.

```
out = awgn(in,snr,signalpower)
```

El comando `awgn` agrega ruido blanco Gaussiano a la señal de entrada (`in`). Si se requiere que la potencia de entrada sea medida, en el lugar de “`signalpower`” se escribe “`measured`”

Los pasos para que la señal atraviese por el canal con desvanecimiento Rayleigh se detallan a continuación:

Se crea el objeto con desvanecimiento Rayleigh por donde atraviesa la señal.

```
rayleighChan = comm.RayleighChannel()
```

A continuación, la señal atraviesa por el objeto denominado “`rayleighChan`” de la siguiente forma:

```
Senal_Rayleigh=(step(rayleighChan,TRAMA.))';
```

Dónde, la señal de datos a la salida del transmisor (`TRAMA`) es la señal que va pasar por el canal. La señal de datos una vez que pasa por el canal Rayleigh es “`Senal_Rayleigh`”.

Una vez que la señal pasa por el canal con desvanecimiento Rayleigh es necesario que se agregue ruido a dicha señal, por lo tanto, se realiza este procedimiento con la ayuda del comando “`awgn`” descrito anteriormente. La sintaxis para obtener la señal de datos con ruido se muestra a continuación.

```
DatosRX_Rayleigh = awgn(Senal_Rayleigh,SNR(i),'measured');
```

Dónde la señal de datos con desvanecimiento Rayleigh es “`Senal_Rayleigh`” y “`SNR(i)`” corresponde al valor de la SNR.

La señal a la salida del canal con desvanecimiento Rayleigh y ruido AWGN es “`DatosRX_Rayleigh`”.

El código que describe la implementación del canal con desvanecimiento Rayleigh y el paso de la señal por dicho canal junto con la agregación de ruido AWGN se encuentra en la sección B.18 del anexo B.

Finalmente, cuando se quiere obtener la señal de datos a través de un canal Rayleigh con la técnica OPS, SAP o OPS-SAP se realiza el mismo procedimiento, únicamente las señales que atraviesan por el canal deben ser tratadas con dichas técnicas.

El código para obtener la señal con técnica OPS-SAP a través del canal con desvanecimiento Rayleigh y ruido AWGN se encuentra en la sección B.19 del anexo B.

2.5.2. CANAL RICIAN

El canal Rician es implementado teniendo en cuenta cuatro rayos con su respectiva potencia y retardo. Para la construcción del canal Rician se crea un objeto con desvanecimiento Rician, para llevar a cabo este proceso se emplea el Toolbox “comm.RicianChannel” perteneciente a Matlab.

El toolbox “comm.RicianChannel” posee varios parámetros que intervienen en su creación los mismos que se describen a continuación.

SampleRate: Corresponde a la frecuencia de muestreo de la señal.

PathDelays: Es el vector de retardos de cada uno de los rayos en segundos.

AveragePathGain: Este valor representa al vector de potencias de cada una de las señales expresado en [dB].

kFactor: Factor presente en el desvanecimiento Rician, si este valor tiende a cero se dice que es un canal con desvanecimiento Rayleigh, mientras que si el factor K tiende al $+\infty$ se dice que es un canal con ruido AWGN.

MaximumDopplerShif: Representa el desplazamiento *Doppler*.

Una vez creado el objeto con sus respectivos parámetros, la secuencia de bits (Trama PDU PLCP) procedente del transmisor pasa por este objeto dando como resultado que la señal tenga desvanecimiento Rician. Para finalizar, a dicha señal se debe agregar ruido mediante el comando “awgn” de Matlab. Los pasos para realizar el proceso anterior se detallan a continuación:

Se crea el objeto con desvanecimiento Rician.

```
ricianChan = comm.RicianChannel()
```

A continuación, la señal atraviesa por dicho objeto, la sintaxis para llevar a cabo este proceso se muestra a continuación:

```
Senal_Rician = (step(ricianChan,TRAMA.')).';
```

Dónde, la señal de datos a la salida del transmisor en el dominio del tiempo es TRAMA y la señal de datos con desvanecimiento Rician es “Senal_Rician”.

Una vez que se tiene la señal con desvanecimiento Rician es necesario que a dicha señal se agregue ruido, por lo tanto, se utiliza el comando “awgn”. La sintaxis para agregar ruido a la señal se presenta a continuación.

`DatosRX_Rician = awgn(Senal_Rician,SNR(i),'measured');`

Dónde, la señal de datos con desvanecimiento Rician es “Senal_Rician” y el valor de la SNR es “SNR(I)”.

La señal con desvanecimiento Rician y ruido AWGN es “DatosRX_Rician”.

El código que describe el paso de la señal por un canal con desvanecimiento Rician y ruido AWGN se encuentra en la sección B.20 del anexo B.

Finalmente, cuando se requiere obtener la señal de datos a través de un canal Rician aplicada la técnica OPS, SAP o OPS-SAP se debe seguir el mismo proceso únicamente las señales que atraviesan por del canal Rician son aquellas que fueron tratadas con dichas técnicas.

El código para que la señal con técnica OPS-SAP atraviese por un canal con desvanecimiento Rician y tenga ruido AWGN se encuentra en la sección B.21 del anexo B.

2.6. RECEPTOR

Una vez que la información atraviesa por el canal inalámbrico y llega al receptor que es el destino final de la información. El receptor se encarga de realizar diferentes procesos para obtener la información que se encuentra en el campo DATA de cada una de las tramas PPDU PLCP recibidas, a continuación, se detallan cada uno de los procesos.

2.6.1. OBTENCIÓN DEL CAMPO DATA

En este bloque se extrae de la trama (PPDU PLCP) los campos que pertenecen a la cabecera y al preámbulo, de esta manera se obtiene únicamente el campo DATA. Para llevar a cabo este proceso se utiliza la función `extraerDATA()` y su sintaxis se presenta a continuación:

`DATAT = extraerDATA(app, TRAMA)`

Dónde, como parámetro de entrada se tiene la trama PPDU PLCP (TRAMA) y como parámetro de salida se tiene el arreglo de bits del campo DATA (DATAT).

El diagrama de flujo que describe la función anterior se encuentra en la Figura A.19 del anexo A y el código se encuentra en la sección B.22 del anexo B.

2.6.2. BLOQUE OFDM EN RECEPCIÓN

En este bloque se procesa cada símbolo OFDM recibido, es decir a cada símbolo se le extrae el prefijo cíclico, a continuación, se realiza el proceso FFT para pasar el símbolo OFDM al dominio de la frecuencia, luego se extraen las muestras de datos (48 muestras),

para finalizar estas muestras (símbolos en frecuencia) son colocadas en un arreglo y posteriormente enviadas para su respectivo procesamiento en los siguientes bloques.

a) Obtención de los símbolos OFDM

El campo DATA se encuentra constituido por un número entero de símbolos OFDM por lo tanto es necesario separar estos símbolos, para llevar a cabo este proceso se utiliza la función `bloqueOFDMrecepcion()` y su formato es el siguiente:

$$M\text{DATA} = \text{bloqueOFDMrecepcion}(\text{app}, \text{DATA})$$

Dónde, (DATA) es el arreglo con los símbolos OFDM pertenecientes al campo DATA de la TRAMA PPDU PLCP.

MDATA corresponde al arreglo de los símbolos en frecuencia, los cuales son obtenidos uno a la vez en el proceso señalado en el inciso b.

El diagrama de flujo que describe el proceso anterior se encuentra en la Figura A.20 del anexo A y el código se encuentra en la sección B.23 del anexo B.

b) Obtención de los datos en el dominio de la frecuencia

Una vez que cada símbolo OFDM es separado, es necesario obtener únicamente las muestras que pertenecen a los datos.

Primero se extrae el prefijo cíclico, a continuación, se aplica la FFT para que las muestras se encuentren en el dominio de la frecuencia, luego se extrae las subportadoras piloto para finalmente tener las 48 muestras que corresponden a los datos. Este proceso es llevado a cabo por la función `extraerSimboloOFDM()` cuya sintaxis es:

$$D\text{simbolos} = \text{extraerSimboloOFDM}(\text{app}, \text{SymbOFDM}, \text{polPiloto})$$

Dónde, los símbolos OFDM en el dominio del tiempo (SymbOFDM) y las subportadoras pilotos (polPiloto) son los argumentos de entrada. Las 48 muestras (símbolos en frecuencia) (Dsimbolos) son el resultado que devuelve como salida la función.

El diagrama de flujo que describe el proceso anterior se encuentra en la Figura A.21 del anexo A y el código se encuentra en la sección B.24 del anexo B.

2.6.3. DEMODULACIÓN DE DATOS

En este bloque se realiza la demodulación de los datos teniendo en cuenta el esquema de modulación que viene dado por M . Una vez que se lleva a cabo la función

demodularString() se obtienen los bits demodulados, su formato se presenta a continuación:

$$IDatos = \text{demodularString}(\text{app}, \text{MDatos}, \text{M})$$

Los argumentos de entrada son los símbolos en el dominio de la frecuencia (MDatos), y el esquema de modulación (M). El argumento de salida son los bits de datos demodulados (IDatos).

El diagrama de flujo que describe a esta función se encuentra en la Figura A.22 del anexo A y el código se encuentra en la sección B.25 del anexo B.

a) Demodulación BPSK

Este esquema de demodulación permite asignar a cada símbolo complejo un bit de acuerdo al diagrama de constelación mostrado en la Figura 1.5, una vez que se emplea la función demodularBPSK() se obtiene el arreglo con los bits demodulados. La sintaxis es la siguiente:

$$IDatos = \text{demodulacionBPSK}(\text{app}, \text{MDatos})$$

Dónde, los símbolos con modulación BPSK (MDatos) son el argumento de entrada y los bits de datos demodulados (IDatos) son la respuesta a la ejecución de la función.

El diagrama de flujo que describe al esquema de demodulación BPSK se encuentra en la Figura A.23 del anexo A y el código se encuentra en la sección B.26 del anexo B.

b) Demodulación QPSK

El esquema de demodulación QPSK permite asignar a cada símbolo complejo dos bits como se indica en la Figura 1.6, una vez que se utiliza la función demodularQPSK(), se obtiene el arreglo de bits demodulados y su formato es el siguiente:

$$IDatos = \text{demodulacionQPSK}(\text{app}, \text{MDatos})$$

Los argumentos de entrada son los símbolos con modulación QPSK (MDatos) y los bits de datos demodulados (IDatos) corresponde la respuesta de la función.

El diagrama de flujo que describe la demodulación QPSK se encuentra en la Figura A.24 del anexo A y el código se encuentra en la sección B.27 del anexo B.

c) Demodulación 16-QAM

Este tipo de demodulación permite asignar a cada símbolo complejo cuatro bits como indica el diagrama de constelación presente en la Figura 1.7, Una vez empleada la función

demodular16QAM() se obtiene el arreglo de bits demodulados bajo un esquema de demodulación 16-QAM. La sintaxis de la función es:

```
IDatos = demodulacion16QAM (app, MDatos)
```

Los argumentos de entrada son los símbolos con modulación 16QAM (MDatos) y los bits de datos demodulados (IDatos) son el resultado de la función.

El diagrama de flujo que describe el esquema de demodulación 16-QAM se encuentra en la Figura A.25 del anexo A y el código se encuentra en la sección B.28 del anexo B.

d) Demodulación 64-QAM

Este tipo de demodulación permite asignar a cada símbolo complejo seis bits como se muestra en la Figura 1.8, Finalmente se obtiene el arreglo de bits demodulados utilizando la función demodulacion64QAM() y su formato se presenta a continuación:

```
IDatos = demodulacion64QAM (app, MDatos)
```

Los argumentos de entrada son los símbolos con modulación 64QAM (MDatos) y los bits de datos demodulados (IDatos) corresponden la salida de la función.

El diagrama de flujo que describe la demodulación 64-QAM se encuentra en la Figura A.26 del anexo A y el código se localiza en la sección B.29 del anexo B.

2.6.4. DEENTRELAZADO

Este bloque se encarga de realizar el deentrelazado de los bits, teniendo en cuenta el número de bits codificados por símbolo (NCBPS) que intervienen en el procesamiento grupo a grupo. Cada grupo es sometido a una segunda y primera permutación inversa como lo indica IEEE 802.11 [3]. Al finalizar se obtiene los bits deentrelazados por medio de la función deentrelazarString() cuya sintaxis es:

```
CDatos = deentrelazarString(app, IBits, NCBPS)
```

Dónde, el arreglo de Bits de datos (IBits) y el número de bits codificados por símbolo (NCBPS) constituyen los parámetros de entrada, mientras que, el arreglo de bits deentrelazados (CDATA) es el parámetro de salida.

El diagrama de flujo que describe el proceso de deentrelazado se encuentra en la Figura A.27 del anexo A y el código se localiza en la sección B.30 del anexo B.

2.6.5. DECODIFICADOR CONVOLUCIONAL

Este bloque se encarga de realizar la decodificación de los bits. El codificador realiza la decodificación de bits de acuerdo al algoritmo de *Viterbi*. Finalmente se obtienen los bits decodificados siguiendo el diagrama de *trellis* generado y el patrón de puntura seleccionado mediante la función `decodificarString()` cuyo formato es el siguiente:

```
SDatos = decodificarString(app, CDatos, R, M)
```

El arreglo de bits (CDatos), la tasa de codificación (R) y esquema de modulación (M) corresponden a los argumentos de entrada de la función. El arreglo de bits decodificados (SDatos) constituyen el resultado de la función.

El diagrama de flujo que describe la función anterior se encuentra en la Figura A.28 del anexo A y el código se encuentra en la sección B.31 del anexo B.

Para el proceso de decodificación se utiliza el algoritmo de *viterbi*, por lo tanto, se emplea el comando `vitdec` cuya sintaxis se describe a continuación:

```
decodedout = vitdec(codedin,trellis,tbdepth,opmode,dectype,puncpat)
```

El comando “`vitec`” realiza la decodificación de los datos de acuerdo al algoritmo de *viterbi*. Los parámetros a excepción de los datos a codificar (`codedin`) determinan el enrejado para establecer las condiciones del algoritmo.

El argumento de entrada “`opmode`” que se utiliza en el decodificador es “`trunc`” que indica que el codificador se remonta desde el mejor estado.

El argumento “`dectype`” utilizado en el decodificador es “`hard`” que significa que el decodificador espera valores binarios.

2.6.6. DESALEATORIZADOR

Este bloque se encarga de realizar la desaleatorización de los bits. Para el proceso se toma en cuenta el estado inicial como lo indica IEEE 802.11 [3]. Finalmente se obtiene un arreglo de bits desaleatorizados utilizando la función `scrambleString()` y su formato es el siguiente:

```
ODatos = scrambleString(app, datos, estadoInicial)
```

Dónde, el arreglo de Bits decodificados (`datos`) y el estado inicial (`estadoInicial`) son los argumentos de entrada, mientras tanto el arreglo de Bits desaleatorizados (`ODatos`) es el argumento de salida.

El diagrama de flujo que describe a la función anterior encuentra en la Figura A.5 del anexo A y el código se localiza en la sección B.5 del anexo B.

Del grupo de bits desaleatorizados se retira los bits pertenecientes al campo *Sevice*, *Tail* y *Pad*, dando como resultado únicamente los bits pertenecientes a la PSDU.

Para finalizar, cabe señalar que los bits obtenidos que pertenecen al PSDU de cada una de las tramas PDU PLCP recibidas de la señal original (sin técnica) y modificadas con la técnica OPS, SAP y OPS-SAP se presentan a continuación.

El arreglo de Bits de la señal original es “PSDU_RX”

El diagrama de flujo que describe el arreglo de bits PSDU_RX se encuentra en la Figura A.29 del anexo A y el código se encuentra en la sección B.32 del anexo B.

Por otra parte, el arreglo de bits que pertenece a la señal con la técnica OPS está dado por “PSDU_RX_OPS” y el arreglo de bits de la señal con técnica SAP es “PSDU_RX_SAP”

Los diagramas de flujo que describen los arreglos de bits PSDU_RX_OPS y PSDU_RX_SAP se encuentran en la Figura A.30 y A.31 del anexo A respectivamente.

Para finalizar, el arreglo de bits de la señal con la técnica OPS-SAP está dado por PSDU_RX_OPS_SAP.

El diagrama de flujo que describe el arreglo de bits PSDU_RX_OPS_SAP se encuentran en la Figura A.32 del anexo A y el código se localiza en la sección B.33 del anexo B.

2.7. REDUCCIÓN DE LA PAPR

En esta sección se describe la implementación de las técnicas utilizadas para la reducción de la PAPR (OPS, SAP y OPS-SAP), además se describe la implementación y el procedimiento para realizar el cálculo de la PAPR. Cabe indicar que este procedimiento se lleva a cabo antes que se agregue el prefijo cíclico es decir cuando el símbolo OFDM está formado por 64 muestras.

2.7.1. PAPR

Determinar el valor correspondiente a la PAPR es uno de los objetivos principales, puesto que dicho valor identifica si el símbolo OFDM que es transmitido posee un elevado valor de la PAPR.

En el bloque denominado PAPR se realiza el cálculo del valor de la PAPR de la señal original, con técnica OPS, SAP y OPS-SAP. El procedimiento para determinar el valor de la PAPR ocurre cuando se está formando el símbolo OFDM, exactamente cuándo el símbolo OFDM cuenta con 64 muestras. Una vez que se tiene el símbolo OFDM se realiza

la evaluación de la PAPR por medio de la función calcularPAPR() y su sintaxis es la siguiente:

$$[\text{PAPR_dB, AVG, PEAK}] = \text{calcularPAPR}(\text{app, SimboloOFDM})$$

Las muestras pertenecientes al símbolo OFDM que se encuentran en el dominio del tiempo (SimboloOFDM) constituyen el parámetro de entrada, mientras que el arreglo con el valor de la PAPR en dB (PAPR_dB), valor de la PAPR necesario para futuros cálculos (AVG), y el valor máximo de la potencia del símbolo OFDM (PEAK) son los parámetros de salida.

El diagrama de flujo que describe al cálculo de la PAPR se encuentra en la Figura A.33 del anexo A y el código se encuentra en la sección B.34 del anexo B.

Cabe señalar que el arreglo con el valor de la PAPR para los símbolos tratados con las técnicas OPS, SAP y OPS-SAP se presentan a continuación:

El Arreglo con el valor de la PAPR con la técnica OPS es "PAPR OPS", mientras tanto, el arreglo con el valor de la PAPR con la técnica SAP es "PAPR SAP", para finalizar, el arreglo con el valor de la PAPR con la técnica OPS-SAP es "PAPR OPS SAP".

2.7.2. TÉCNICA OPS-SAP

La arquitectura OPS-SAP se encuentra formada por dos técnicas OPS y SAP ambas técnicas trabajan independientemente o en conjunto dependiendo de los resultados que se quieren obtener.

El bloque OPS-SAP como tal es implementado con la ayuda de tres funciones, una de ellas OPS() corresponde a la técnica OPS y las otras dos restantes (metrica() y predistorsion()) corresponden a la técnica SAP.

a) Bloque OPS

El bloque OPS se encarga de reemplazar las subportadoras piloto por una secuencia ortogonal, la cual cumple con el requisito de tener el menor valor de la PAPR.

Para llevar a cabo este proceso, en el bloque OPS se crea una matriz 4x4 (Hadamard) cuyas filas representan una secuencia ortogonal distinta una de la otra. Cada secuencia es probada, para ello se calcula la PAPR de cada una de ellas y así se determina cuál de ellas presenta la menor PAPR.

La secuencia que presente la menor PAPR es seleccionada y reemplaza el valor de las subportadoras piloto en el símbolo OFDM. Finalmente se obtiene el símbolo OFDM en el dominio de la frecuencia aplicado la técnica OPS.

La función encargada de realizar este procedimiento es OPS() y su sintaxis se presenta a continuación:

$$[\text{par, par_avg, s_ops, pilots_selected}] = \text{OPS}(m, N_p, N, s_mi, \text{idx_p})$$

Dónde, el número de subportadoras piloto (N_p) y (m), el número de muestras que forman el símbolo OFDM (N), el símbolo OFDM en el dominio de la frecuencia (s_mi) y el vector con los índices de las subportadoras piloto (idx_p) son los argumentos de entrada.

La secuencia con el valor mínimo de la PAPR (par), el valor medio de la PAPR de las secuencias ortogonales (par_avg), el símbolo OFDM aplicado la técnica OPS (s_ops), y el arreglo con la secuencia ortogonal seleccionada (pilots_selected) son los argumentos de salida.

El diagrama de flujo que describe la obtención del símbolo OFDM en el dominio de la frecuencia aplicado la técnica OPS se encuentra en la Figura A.35 del anexo A y el código se encuentra en la sección B.36 del anexo B.

b) Bloque SAP

El bloque SAP se encarga de realizar la predistorsión de las muestras que presentan una métrica mayor con respecto a un umbral predeterminado.

Para realizar este proceso, en el bloque SAP, se realizan dos procedimientos;

El primer proceso corresponde al cálculo de la métrica de cada uno de las muestras del símbolo OFDM para ello se utiliza la función $\text{metrica}()$.

Para identificar cuáles muestras del símbolo OFDM deben ser extendidas (predistorsionadas) se aplica la técnica SAP, esta técnica define una métrica para la identificación de las muestras que están contribuyendo en la formación de los picos de energía a la salida de la IFFT. La función $\text{metrica}()$, se encarga de identificar a través de su índice las subportadoras que deben ser extendidas.

La sintaxis de la función que realiza el cálculo de la métrica se presenta a continuación:

$$[\text{me_s, id_s}] = \text{metrica}(\text{simboloTiempo}, \text{Avg}, \text{simboloFrecuencia}, N)$$

Dónde, el arreglo con las muestras del símbolo OFDM en el dominio del tiempo (simboloTiempo), el arreglo con las muestras del símbolo OFDM en el dominio de la frecuencia (simboloFrecuencia), el valor de la PAPR (Avg) y el número de muestras (subportadoras) que conforman el símbolo OFDM (N) son los argumentos de entrada. Los

índices de las muestras que aportan con los picos de energía (id_s) corresponden el argumento de salida.

El diagrama de flujo que describe al cálculo de la métrica se encuentra en la Figura A.34 del anexo A y el código se localiza en la sección B.35 del anexo B.

Cabe señalar que los arreglos pertenecientes a las muestras de los símbolos OFDM tanto en el dominio del tiempo como de la frecuencia para aquellos símbolos OFDM aplicada la técnica OPS y que van a pasar por el proceso para el cálculo de la métrica son “simboloTiempo OPS” y “simboloFrecuencia OPS” respectivamente

Una vez que se realizó el procedimiento anterior se receptan los índices que corresponden a las muestras que sobrepasaron el umbral, es decir aquellas que poseen una métrica elevada. Las muestras pertenecientes a los índices son predistorcionados según el tipo de muestra como se indica en la Figura 1.12. Finalmente se obtiene el símbolo OFDM en el dominio de la frecuencia con las muestras predistorcionadas es decir el símbolo OFDM fue tratado con la técnica SAP.

La función encargada de realizar este proceso es predistorcion() y su sintaxis se presenta a continuación:

$$a_mod = predistorcion(m, indices, a, alfa, L)$$

Dónde, el tipo de modulación (m), los índices procedentes de la función metrica() (índices), el subconjunto (L), el factor de escala constante (alfa), y el símbolo OFDM cuyas muestras fueron aplicadas la métrica (a) son los argumentos de entrada.

Mientras tanto, el símbolo OFDM con las muestras predistorcionadas (a_mod), es decir el símbolo OFDM con la técnica SAP es el argumento de salida.

El diagrama de flujo que describe la obtención del símbolo OFDM en el dominio de la frecuencia aplicado la técnica SAP se encuentra en la Figura A.36 del anexo A y el código se encuentra en la sección B.37 del anexo B.

c) Bloque OPS-SAP

El bloque OPS-SAP corresponde a la unión de los bloques OPS y SAP respectivamente, en otras palabras, cada símbolo OFDM cuyas subportadoras piloto fueron modificadas con la técnica OPS, y sus muestras pertenecientes a las subportadoras de datos fueron predistorcionadas de acuerdo a una métrica, constituyen el símbolo OFDM en el dominio de la frecuencia aplicado la técnica OPS-SAP.

El orden de ejecución de las funciones es el siguiente:

Primero se lleva a cabo la etapa perteneciente a la técnica OPS representada por la función OPS().

$$s_ops = OPS(m, N_p, N, s_mi, idx_p)$$

Una vez aplicada la función anterior se obtiene el símbolo OFDM cuyas subportadoras piloto fueron reemplazadas por una secuencia ortogonal (s_ops).

A continuación, dicho símbolo (s_ops) es sometido al cálculo de la métrica respectiva, con el propósito de obtener los índices de las muestras que contribuyen a los picos altos de energía por medio de la función metrica().

$$[métrica, ids] = metrica(s_opsTiempo, Avg, s_ops, N)$$

Al ejecutar la función anterior se obtiene un arreglo con los índices de las muestras que poseen una elevada métrica (ids).

Las muestras pertenecientes a los índices obtenidos anteriormente son predistorcionados por medio de la función predistorsion()

$$a_mod = predistorsion(m, ids, a, alfa, L)$$

Finalmente se obtiene el símbolo OFDM modificado con la técnica OPS-SAP (a_mod).

2.8. TRATAMIENTO DE DATOS

Una vez implementadas las funciones para el transmisor, canal inalámbrico, receptor, técnica OPS-SAP y el cálculo de la PAPR se requiere almacenar dichos valores para luego procesarlos y presentarlos.

2.8.1. BER

El cálculo de este parámetro es prescindible para determinar la relación entre los bits errados que llegaron al receptor, respecto a los bits enviados por parte del transmisor.

Este parámetro se calcula en base a un mismo SNR. El valor de la BER es representado mediante un arreglo de valores, dependiendo de los parámetros o factores que afectaron la transmisión de la información. Dichos factores incluyen si la transmisión se realizó a través de un canal AWGN, con desvanecimiento Rayleigh o Rician, además si los datos fueron tratados con la técnica OPS, SAP u OPS-SAP.

Los diferentes valores correspondientes a la BER son almacenados de acuerdo a la siguiente nomenclatura Tabla 2.5.

Tabla 2.5 Nomenclatura para los valores de BER.

| Nomenclatura | Descripción |
|------------------------------|---|
| BER_AWGN_ORIGINAL | Arreglo de datos, cuyos datos se les agrego ruido, sin técnica de reducción de la PAPR |
| BER_RAYLEIGH_ORIGINAL | Arreglo de datos, cuyos datos atravesaron por un canal con ruido y desvanecimiento Rayleigh, sin técnica de reducción de la PAPR. |
| BER_RICIAN_ORIGINAL | Arreglo de datos, cuyos datos atravesaron por un canal con ruido y desvanecimiento Rician, sin técnica de reducción de la PAPR. |

Si la señal fue modificada con alguna técnica de reducción de la PAPR en el lugar de “ORIGINAL” corresponde el nombre de la técnica (OPS, SAP u OPS-SAP).

2.8.2. CCDF DE LA PAPR

La evaluación del rendimiento de la PAPR cuando se aplica la técnica SAP, OPS u OPS-SAP a los datos transmitidos se realiza mediante métricas de evaluación como son: CCDF de la PAPR y PSD.

Teniendo en cuenta que los valores de la PAPR son variables, es necesario que dichos valores sean evaluados desde un enfoque estadístico. Por tal razón se emplea la CCDF.

La recopilación de los datos de la CCDF de la PAPR cuando se aplican las técnicas OPS, SAP u OPS-SAP se compara con los valores originales (sin técnica) mediante curvas. De esta forma se evidencia si existe o no una reducción en el valor de la PAPR.

Este procedimiento se implementa por medio de diferentes funciones que se detallan a continuación:

Primero se determina si la probabilidad de la PAPR perteneciente a un determinado símbolo OFDM es mayor a un umbral predefinido. De esta manera se obtienen los valores de la CCDF de la PAPR de la señal original y de aquellas que fueron modificadas con la técnica OPS, SAP u OPS-SAP. Para realizar este proceso se implementó la función `calcularCCDF()` y cuya sintaxis se presenta a continuación:

$$[\text{CCDF}] = \text{calcularCCDF}(\text{app}, \text{PAPR}, \text{Umbral}, \text{numTramas})$$

El arreglo con los diferentes valores de PAPR (PAPR), el arreglo con el valor del umbral (Umbral), y número de tramas PDU PLCP enviadas (numTramas) son los argumentos de entrada.

Mientras que el arreglo con los valores de la CCDF de la PAPR de la señal ingresada (CCDF) es el resultado que brinda la ejecución de la función.

El diagrama de flujo que describe la obtención de los valores de CCDF de la PAPR se encuentra en la Figura A.37 del anexo A y el código se encuentra en la sección B.38 del anexo B.

Cabe mencionar que los valores pertenecientes a la PAPR modificados con las técnicas OPS, SAP u OPS-SAP proporcionan los datos suficientes para determinar la CCDF de la PAPR para cada técnica respectivamente.

2.8.3. PSD

Otra forma para evaluar el rendimiento de la reducción de la PAPR es la PSD. Los valores para el cálculo de la PSD se obtienen en el lado del transmisor, después que la señal atraviesa la etapa de amplificación.

Una vez que se tiene el arreglo de bits amplificados se realiza el cálculo de la PSD.

La función encargada de realizar este proceso es la siguiente:

| |
|---|
| <code>w = hann(floor(max(size(S_AMP))/na));</code> |
| <code>[DatosSpecIn(:,j)] = pwelch(S_AMP,w,[],spftSize,BW);</code> |
| <code>PSD =sum(DatosSpecIn,2)./nloop;</code> |

Dónde, la señal amplificada (S_AMP), el tamaño de la ventana (w), Ancho de banda (BW), y vector con los valores iniciales (spftSize) corresponden a los argumentos de entrada, mientras que los datos pertenecientes a la PSD (DatosSpecIn) son el argumento de salida.

El diagrama de flujo que describe la obtención de los valores de la PSD se encuentra en la figura A.38 del anexo A, y el código se encuentra en la sección B.39 del anexo B.

Existen varios comandos empleados en el tratamiento de las muestras y la obtención de la PSD, los comandos utilizados se describen a continuación:

$$w = \text{hann}(L)$$

El comando “hann” retorna una ventana de Hann simétrica de puntos L.

$$[pxx,f] = \text{pwelch}(x,\text{window},\text{noverlap},f,\text{fs})$$

El comando “pwelch” devuelve la estimación de densidad espectral de potencia (PSD). Donde, la señal se divide en segmentos acorde a la longitud de la ventana (w), “noverlap”

corresponde al número de muestras superpuestas, además f es el vector con las frecuencias específicas y f_s corresponde a la frecuencia de muestreo.

Una vez realizado el proceso anterior se tienen los valores pertenecientes a la PSD de la señal original. Además, para obtener el valor de la PSD para las señales con las técnicas OPS, SAP y OPS-SAP, se realiza el mismo procedimiento teniendo en cuenta que la señal amplificada debe corresponder aquella que fue modificada con alguna de las técnicas mencionadas.

2.9. PRESENTACIÓN DE RESULTADOS

Los resultados son presentados a través de curvas de BER vs SNR, CCDF de la PAPR y PSD. Las curvas que corresponden a transmisiones por canales con ruido AWGN, desvanecimiento Rician y Rayleigh se presentan en un solo gráfico que cuenta con las curvas de la BER vs SNR con ruido AWGN, desvanecimiento Rayleigh, y Rician, para la señal original y aquella con técnica OPS-SAP.

Los resultados de la CCDF de la PAPR se presentan a través de curvas que representan las señales de datos original (sin técnica de reducción) y aquellas modificadas con las técnicas OPS, SAP y OPS-SAP.

Finalmente, los resultados de la PSD se presentan a través de curvas que representan la señal original (sin técnica) y las señales que fueron modificadas con las técnicas OPS, SAP y OPS-SAP. Los comandos empleados para realizar los gráficos se detallan a continuación en la Tabla 2.6.

Tabla 2.6 Comandos para la creación de gráficas

| Comando | Descripción |
|-----------------|---|
| Semilogy | Realiza un gráfico cuya componente en Y se encuentra en escala semilogarítmica. |
| Plot | Realiza un gráfico. |
| Title | Se adhiere un título a la gráfica que se está construyendo. |
| Legend | Aplica una leyenda en la gráfica para distinguir las diferentes curvas con facilidad. |
| Label | Agrega una leyenda a los ejes. |
| Grid on | Agrega una cuadratura a los gráficos para que se pueda visualizar mejor. |

El código correspondiente a la creación de los gráficos se encuentra en la sección B.40, B.41 y B.42 del anexo B.

3. RESULTADOS Y DISCUSIÓN

En este capítulo se presentan los resultados obtenidos en la simulación referente a la transmisión de información en un sistema de comunicación IEEE 802.11a/g/p a través de canales inalámbricos con ruido AWGN, desvanecimiento Rician y Rayleigh. Los resultados se presentan a través de curvas de BER vs SNR, CCDF de la PAPR y PSD para cada escenario descrito en la Tabla 3.1. Cada escenario corresponde a las velocidades permitidas por los estándares IEEE 802.11a/g/p ya sea para un ancho de banda de 20 MHz y 10 MHz.

Tabla 3.1 Escenarios de Simulación

| Esquema de Modulación | Tasa de Codificación | Ancho de Banda |
|-----------------------|----------------------|----------------|
| BPSK | $r = 1/2$ | 10 y 20 MHz |
| BPSK | $r = 3/4$ | 10 y 20 MHz |
| QPSK | $r = 1/2$ | 10 y 20 MHz |
| QPSK | $r = 3/4$ | 10 y 20 MHz |
| 16-QAM | $r = 1/2$ | 10 y 20 MHz |
| 16-QAM | $r = 3/4$ | 10 y 20 MHz |
| 64-QAM | $r = 2/3$ | 10 y 20 MHz |
| 64-QAM | $r = 3/4$ | 10 y 20 MHz |

El análisis de los resultados se enfoca en la evaluación del rendimiento de las señales aplicadas la técnica OPS-SAP con respecto a la señal original (sin técnica de reducción de la PAPR) para las curvas de BER vs SNR. Para las gráficas de la CCDF de la PAPR y PSD el análisis también se lo realiza respecto a las técnicas SAP y OPS.

3.1. PARÁMETROS PARA LOS ESCENARIOS DE SIMULACIÓN

Para desarrollar las diferentes simulaciones se utiliza la interfaz desarrollada en App Designer, la misma que cuenta con tres zonas específicas; la primera corresponde a los parámetros de entrada, la segunda es la presentación de resultados y la tercera corresponde a la información acerca de los procesos que lleva a cabo la aplicación.

La aplicación se encuentra estructurada principalmente por una parte visual y otra correspondiente a la programación. La parte de la programación se ubica en la ventana *EDITOR* dónde a través de funciones se lleva a cabo los diferentes procesos. Mientras que la parte visual se ubica en la ventana *DESIGNER* dónde se encuentran todos los componentes visuales ya sea para recepción de datos o para visualización de resultados.

En la aplicación se consideró los parámetros de cada estándar IEEE 802.11a/g/p señalados en la Tabla 3.2.

Tabla 3.2 Parámetros para cada estándar IEEE 802.11a/g/p.

| Parámetro | Descripción | Valor |
|-----------|---|--|
| M | Esquema de modulación | 2 (BPSK), 4 (QPSK), 16 (16-QAM), 64 (64-QAM) |
| r | Tasa de codificación | 1/2, 3/4 y 2/3 |
| m | Número de bits codificados por cada subportadora | 1 (BPSK), 2 (QPSK), 4 (16-QAM), 6 (64-QAM) |
| V | Velocidad de transmisión [Mbps] para un AB = 10 MHz | 3, 4.5, 6, 9, 12, 18, 24, 27 |
| | Velocidad de transmisión [Mbps] para un AB = 20 MHz | 6, 9, 12, 18, 24, 36, 48, 54 |
| IG | Longitud el prefijo cíclico | ¼ de la longitud del símbolo OFDM |
| AB | Ancho de banda | 10 MHz y 20 MHz |

Mientras que, en la etapa de amplificación se utilizó un amplificador de potencia HPA modelo Rapp [38]. El diseño de este modelo cuenta con los siguientes parámetros Tabla 3.3.

Tabla 3.3 Parámetros del HPA.

| Parámetro | Descripción | Valor |
|--------------------|------------------|-------|
| HPABKOFF | IBO del HPA | 10 |
| Ganancia | Ganancia del HPA | 1 |
| SuavidadHPA | Suavidad el HPA | 2 |

Los parámetros para establecer el comportamiento tanto del canal Rician como Rayleigh dependen de los valores de PDP (*Power Delay Profile*), dichos valores cambian de acuerdo a cada estándar. Los valores pertenecientes a la PDP se indican en la Tabla 3.4.

Tabla 3.4 Valores del PDP para cada estándar

| Estándar | Potencia de los Rayos [dB] | Retardo de los Rayos [us] |
|---------------------|----------------------------|---------------------------|
| IEEE 802.11a | [0 -1 -3.9 -6] | [0 2 7 12] |
| IEEE 802.11g | [0 -2.4 -5.5 -7.5] | [0 4.8 10 15] |
| IEEE 802.11p | [0 -2 -5 -7] | [0 4 9 14] |

En cuanto a los parámetros que intervienen en la técnica OPS-SAP, cabe mencionar, que la primera etapa que pertenece a la técnica OPS, se estableció una matriz *Walsh - Hadamard* de tamaño 4 para la sustitución de las señales piloto. La segunda etapa

correspondiente a SAP, se eligió el tamaño del subconjunto $L = 26$ que representa el valor de símbolos a predistorcionar y el valor de factor de escala constante $\alpha = 1.55$ ya que presentan los mejores resultados como se indica en la Figura 3.1. Los parámetros que utiliza la aplicación que pertenecen a la técnica OPS-SAP se describen en la Tabla 3.5.

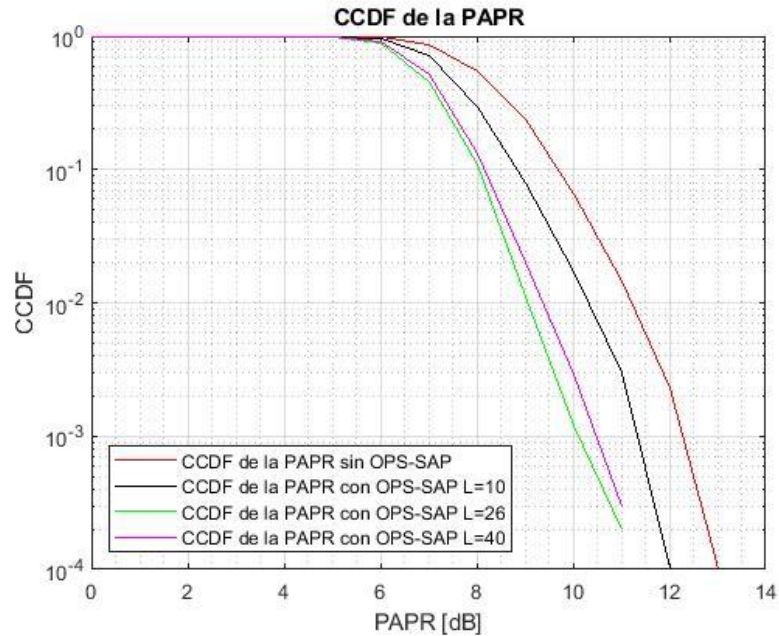


Figura 3.1 Resultados con los valores de L y α

Tabla 3.5 Parámetros de la técnica OPS-SAP

| Parámetro | Descripción | Valor |
|-----------|----------------------------|-------|
| N_p | Matriz Walsh-Hadamard | 4 |
| L | Tamaño del subconjunto | 26 |
| α | Factor de escala constante | 1.55 |

Finalmente, la simulación desarrollada en la aplicación concluye cuando se envían y reciben 10000 tramas PDU PLCP para obtener un amplio volumen de valores y obtener curvas con trazos bien definidos.

Cabe mencionar que la manipulación y ejecución de la aplicación se detallan en el anexo C donde se encuentra un manual de usuario correspondiente a la aplicación desarrollada en *App Designer*.

3.2. PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE COMUNICACIÓN AD-HOC.

Una vez implementado el sistema de comunicación con sus diferentes componentes, es necesario comprobar el funcionamiento correcto del sistema, para ello se realiza el envío de información a través de un canal inalámbrico con ruido AWGN. Una vez finalizada la transmisión se comparan las curvas de BER vs SNR obtenidas mediante la simulación con respecto a las teóricas.

Los resultados a ser comparados corresponden al sistema de comunicaciones regido por el estándar IEEE 802.11p. Las curvas obtenidas pertenecen a los esquemas de modulación: BPSK, QPSK, 16-QAM y 64-QAM. Cabe mencionar que en este escenario de simulación no se tomó en cuenta al codificador convolucional.

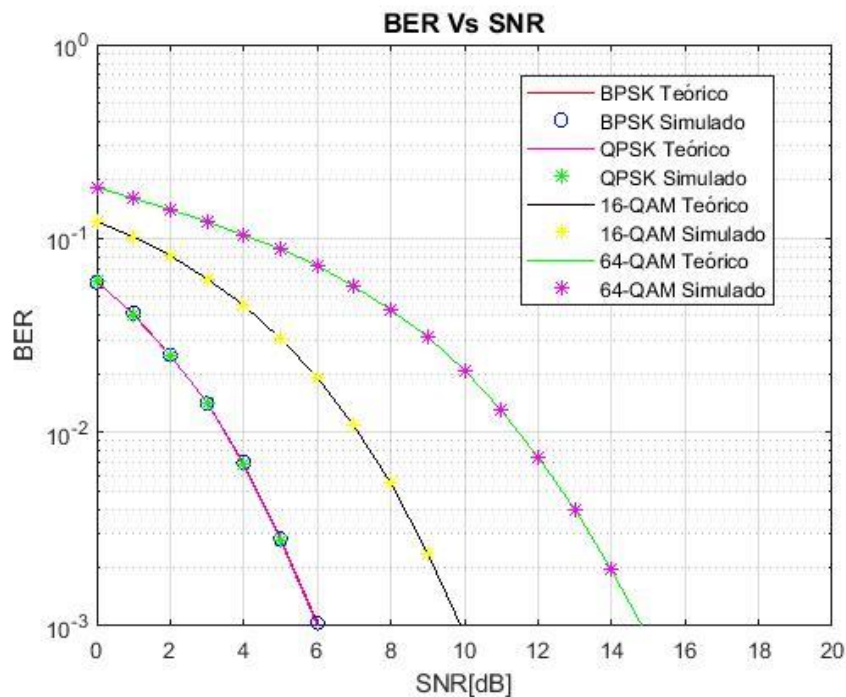


Figura 3.2 Curvas teóricas y simuladas de los diferentes esquemas de modulación

La Figura 3.2 muestra las curvas teóricas (líneas continuas) que pertenecen a cada esquema de modulación y las curvas obtenidas mediante la simulación (líneas mixtas con asteriscos). Las curvas obtenidas por medio de la simulación son bastante similares a las teóricas para todos los esquemas de modulación, de esta forma se verifica el correcto funcionamiento del sistema implementado.

3.2.1. FUNCIONAMIENTO DE LA TÉCNICA OPS-SAP

Una parte importante de la simulación corresponde a la implementación de la técnica OPS-SAP, para comprobar su funcionamiento es necesario obtener los resultados de la métrica con la cual se evalúa el rendimiento de dicha técnica, por tal razón se obtienen los valores de la PAPR de la señal original, con técnica OPS, SAP y OPS-SAP.

Los valores en la Tabla 3.6 pertenecen a los valores promedios de la PAPR para cada técnica obtenidos en los diferentes esquemas de modulación. Al revisar estos valores se verifica la correcta implementación de las técnicas de reducción de la PAPR, ya que, el valor de la PAPR de la señal original tiene el mayor valor.

Tabla 3.6 Valores promedios de la PAPR para cada esquema de modulación.

| Esquema de modulación | Original [dB] | OPS [dB] | SAP [dB] | OPS-SAP [dB] |
|-----------------------|---------------|----------|----------|--------------|
| BPSK | 7.73 | 6.96 | 6.58 | 6.26 |
| QPSK | 8.22 | 7.58 | 7.22 | 6.96 |
| 16-QAM | 8.22 | 7.56 | 8.22 | 7.56 |
| 64-QAM | 8.23 | 7.57 | 8.23 | 7.57 |

3.3. RESULTADOS DEL SISTEMA DE COMUNICACIÓN IEEE 802.11a/g/p.

A) BER vs SNR

Los resultados de la BER vs SNR, para un canal inalámbrico con ruido AWGN, desvanecimiento Rayleigh o Rician. Las gráficas son presentadas en escala semilogarítmica (eje Y), las señales originales son aquellas continuas y las señales que fueron modificadas con la técnica OPS-SAP son líneas mixtas, en la Tabla 3.7 se detallan las características de cada línea asociada a cada señal.

Tabla 3.7 Nomenclatura de las señales para las gráficas de BER vs SNR.

| Tipo de señal | Estilo | Color | Leyenda |
|--|----------|----------|--------------------------|
| Señal original con ruido AWGN | Continua | Roja | AWG de la señal original |
| Señal original con desvanecimiento Rician para el Estándar IEEE 802.11a | Continua | Amarilla | Rician original(a) |
| Señal original con desvanecimiento Rician para el Estándar IEEE 802.11g | Continua | Rosada | Rician original(g) |
| Señal original con desvanecimiento Rician para el Estándar IEEE 802.11p | Continua | Cian | Rician original(p) |
| Señal original con desvanecimiento Rayleig para el Estándar IEEE 802.11a | Continua | Negra | Rayleigh original(a) |

| | | | |
|---|----------|----------|----------------------|
| Señal original con desvanecimiento Rayleigh para el Estándar IEEE 802.11g | Continua | Roja | Rayleigh original(g) |
| Señal original con desvanecimiento Rayleigh para el Estándar IEEE 802.11 ^a | Continua | Verde | Rayleigh original(p) |
| Señal OPS-SAP con ruido AWGN | Mixta | Azul | AWG OPS-SAP |
| Señal OPS-SAP con desvanecimiento Rician para el Estándar IEEE 802.11a | Mixta | Negra | Rician OPS-SAP(a) |
| Señal OPS-SAP con desvanecimiento Rician para el Estándar IEEE 802.11g | Mixta | Azul | Rician OPS-SAP(g) |
| Señal OPS-SAP con desvanecimiento Rician para el Estándar IEEE 802.11p | Mixta | Verde | Rician OPS-SAP(p) |
| Señal OPS-SAP con desvanecimiento Rayleigh para el Estándar IEEE 802.11a | Mixta | Amarilla | Rayleigh OPS-SAP(a) |
| Señal OPS-SAP con desvanecimiento Rayleigh para el Estándar IEEE 802.11g | Mixta | Azul | Rayleigh OPS-SAP(g) |
| Señal OPS-SAP con desvanecimiento Rayleigh para el Estándar IEEE 802.11 ^a | Mixta | Rosada | Rayleigh OPS-SAP(p) |

El análisis de los resultados se realiza entre la señal original (sin técnica) y la señal con la técnica OPS-SAP.

B) CCDF de la PAPR

Para el análisis de los resultados se implementan las curvas de CCDF de la señal original (sin técnica de reducción), señal con las técnicas OPS, SAP y OPS-SAP.

La curva que representa el valor de la CCDF de la PAPR para la señal original es una línea continua de color rojo, por otra parte, la señal con la técnica OPS es una línea mixta con asteriscos de color verde, la señal con la técnica SAP es una línea mixta con asteriscos de color rosado y la señal con la técnica OPS-SAP es una línea continua de color azul.

C) PSD

Las gráficas cuentan con la PSD de la señal original (línea continua color roja), PSD de la señal con técnica OPS (línea continua color amarilla), PSD de la señal con técnica SAP (línea continua color verde) y la PSD de la señal con técnica OPS-SAP (línea continua de color azul).

3.3.1. ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 1/2.

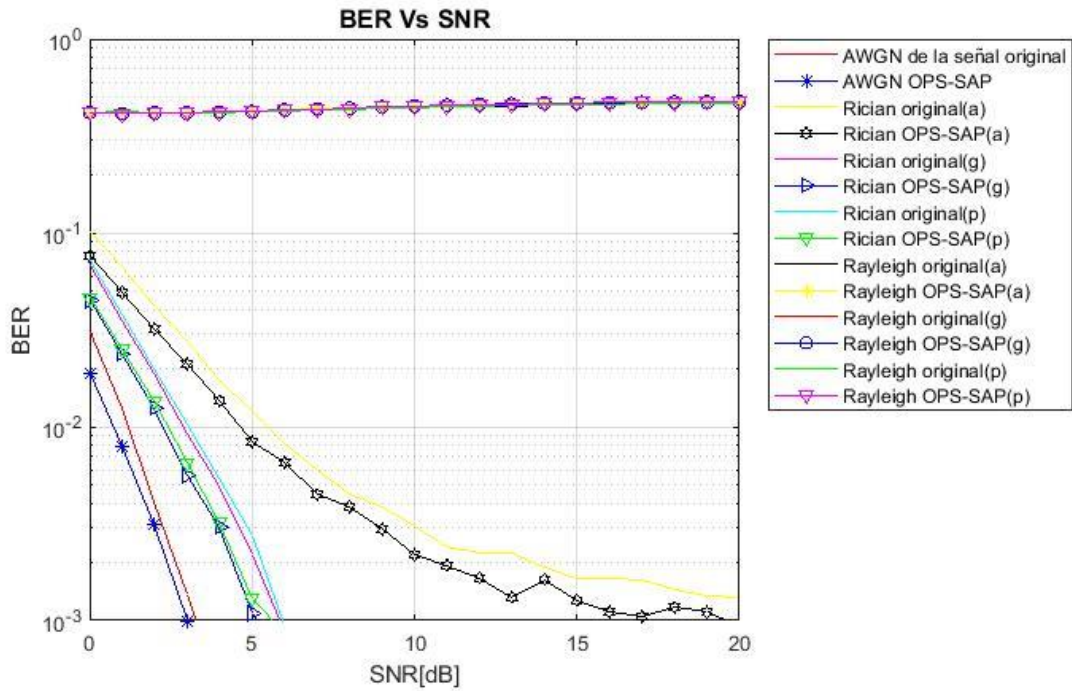


Figura 3.3 BER vs SNR para una modulación BPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

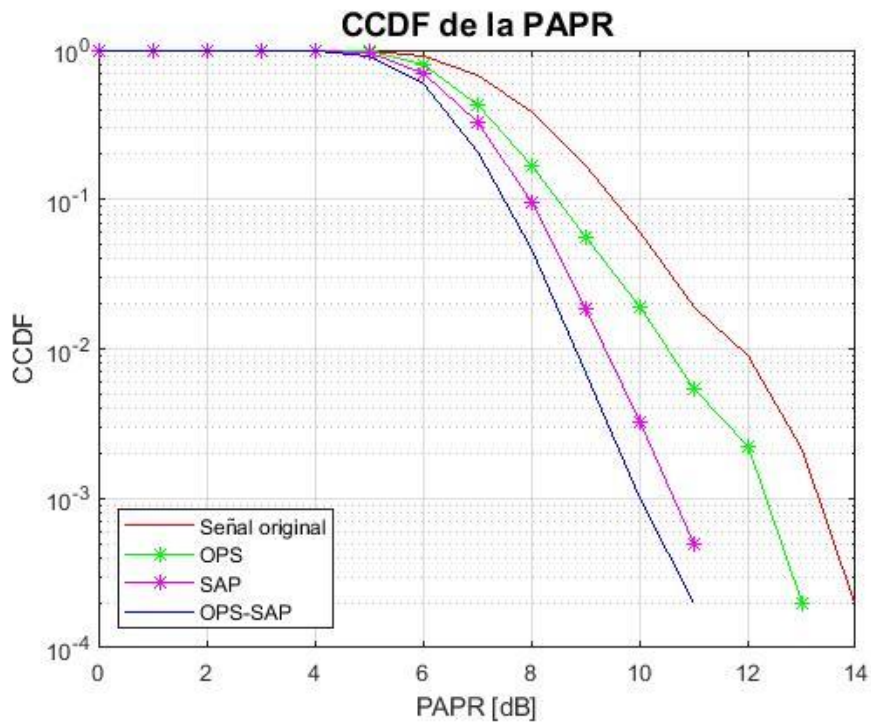


Figura 3.4 CCDF de la PAPR con modulación BPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

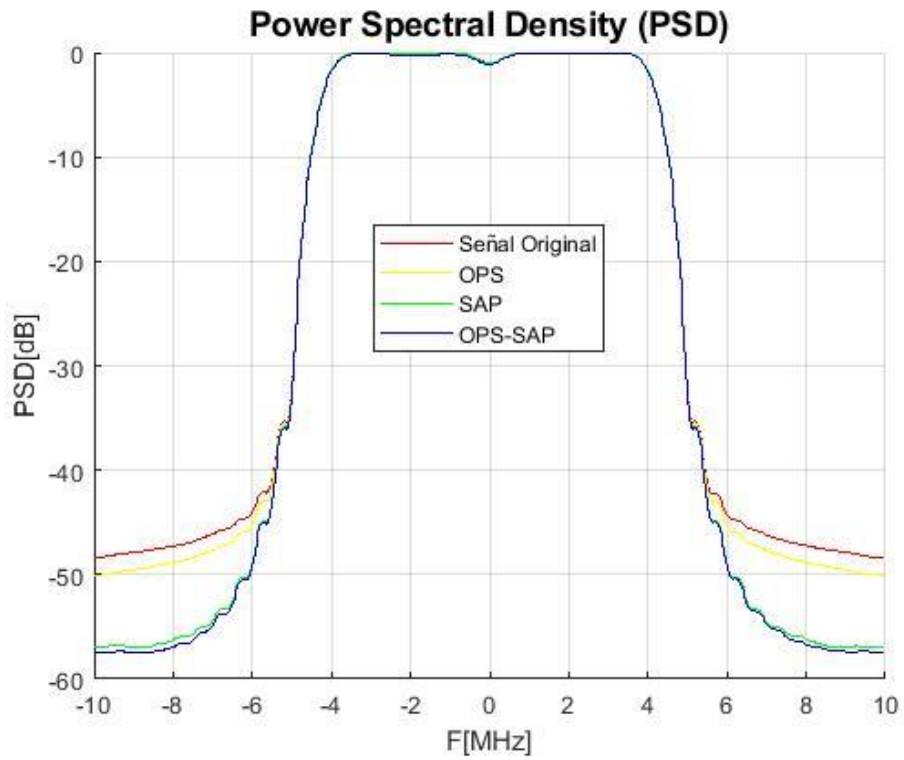


Figura 3.5 PSD con modulación BPSK tasa de codificación $r = 1/2$ y 20MHz

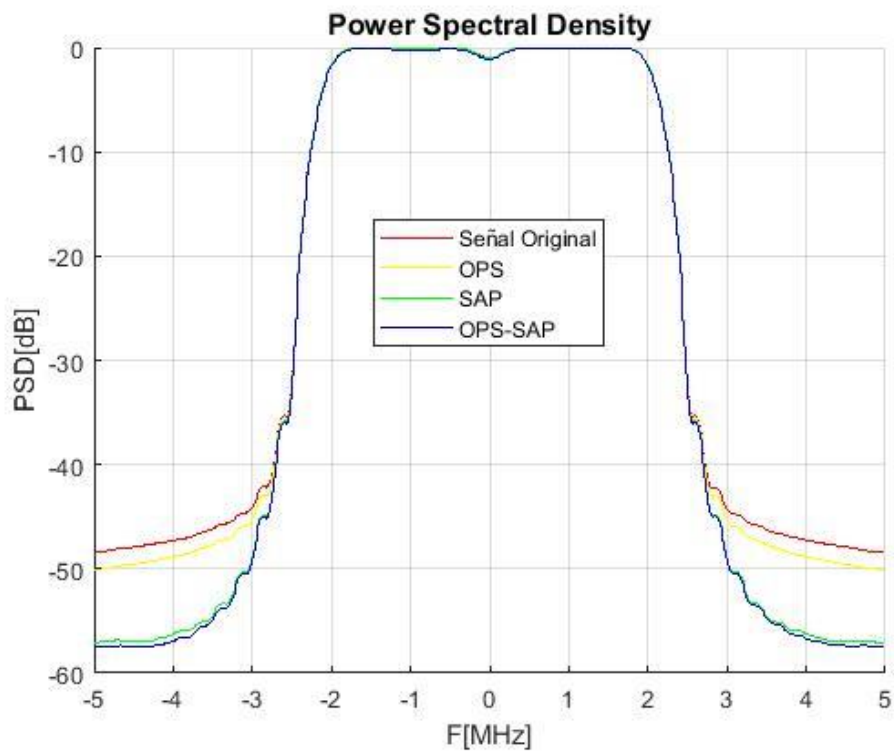


Figura 3.6 PSD con modulación BPSK tasa de codificación $r = 1/2$ y 10MHz

3.3.2. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 1/2.

3.3.2.1. BER vs SNR

Se observa en la Figura 3.3 que la curva perteneciente a la técnica OPS-SAP se aleja hacia la izquierda con respecto a la curva original, indicando una ganancia en la SNR. Para un valor de $BER = 10^{-3}$, la técnica OPS-SAP obtiene una ganancia alrededor de 0.5 [dB] respecto a la señal original.

Para el estándar IEEE 802.11a sobre un canal Rician, en la Figura 3.3 se observa que para un valor de $BER = 10^{-2}$, la técnica OPS-SAP alcanza una ganancia alrededor de 0.9 [dB] respecto a la señal original. Mientras tanto, para los estándares IEEE 802.11g/p y un valor de $BER = 10^{-3}$, la técnica OPS-SAP obtiene una ganancia alrededor de 0.7 [dB] respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, en la Figura 3.3 se observa que el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original. Presenta una mayor degradación en la BER con respecto a una transmisión por un canal con desvanecimiento Rician

3.3.2.2. CCDF de la PAPR

En la Figura 3.4 se observa que las curvas pertenecientes a las técnicas OPS, SAP y OPS-SAP se alejan hacia la izquierda con respecto a la curva original, indicando una reducción de la PAPR. Para una probabilidad de 10^{-3} , la técnica OPS obtiene una reducción de 1 [dB], SAP 2.7 [dB] y OPS-SAP de 3.3 [dB] con respecto a la señal original.

3.3.2.3. PSD

En las Figuras 3.5 y 3.6 se presentan los resultados para un ancho de banda de 20 y 10 MHz respectivamente. En las figuras se observa que la radiación fuera de banda perteneciente a las señales con las técnicas OPS, SAP y OPS-SAP se alejan hacia abajo con respecto a la señal original, indicando una reducción de la radiación fuera de banda. La técnica OPS obtiene una reducción de 2 [dB], SAP 9 [dB] y OPS-SAP de 10 [dB] respecto a la señal original.

3.3.3. ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 3/4.

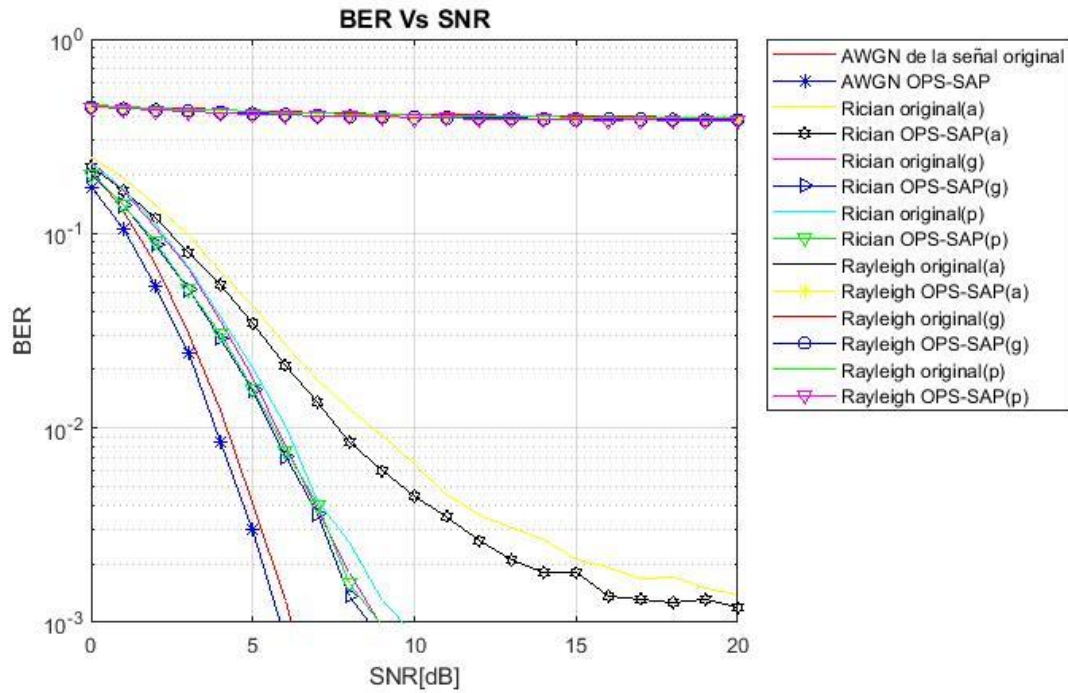


Figura 3.7 BER vs SNR para una modulación BPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

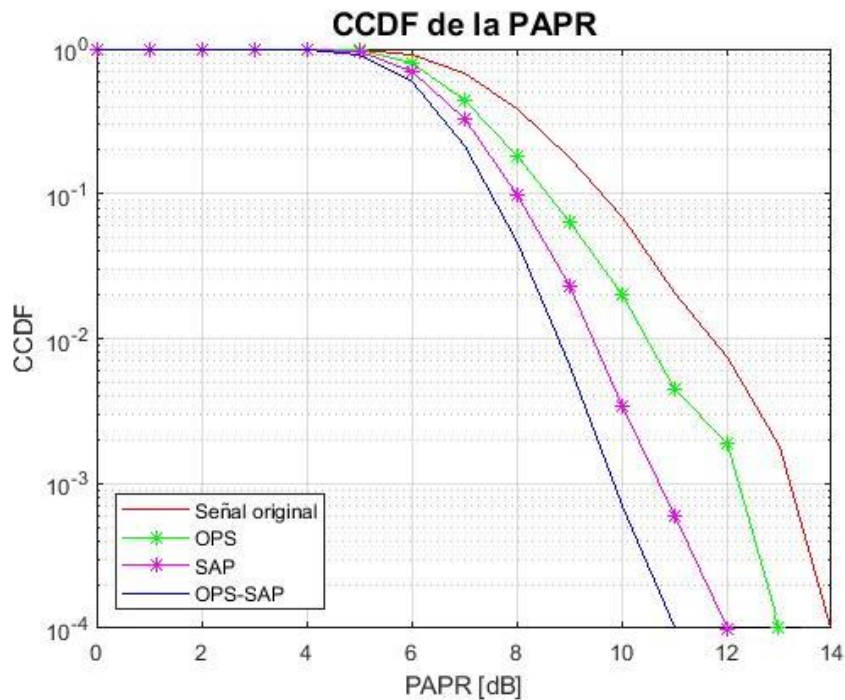


Figura 3.8 CCDF de la PAPR con modulación BPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

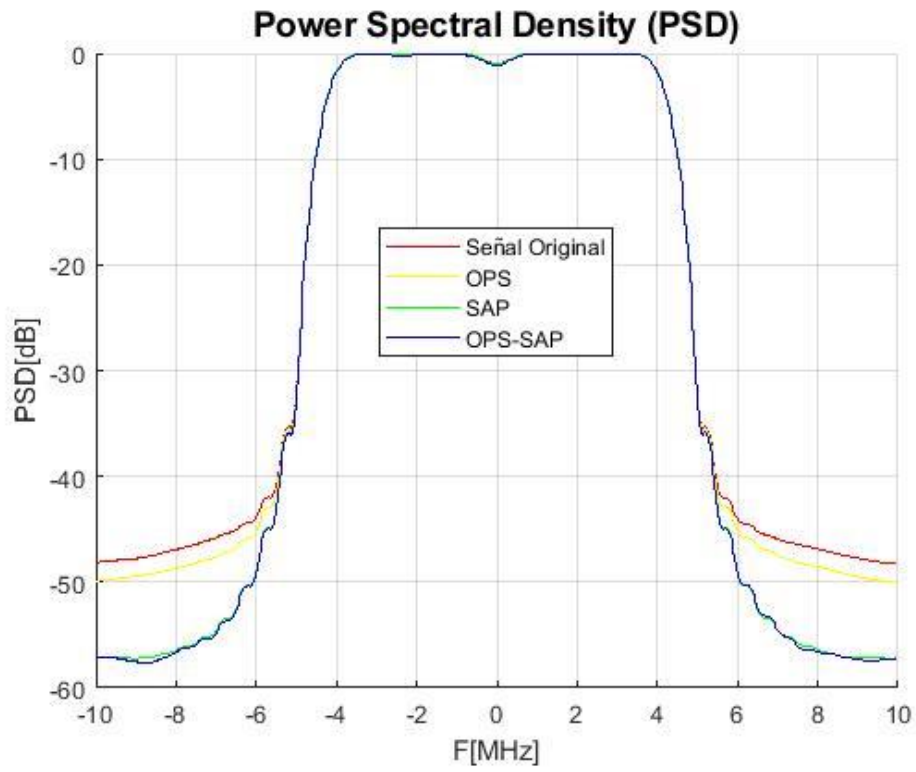


Figura 3.9 PSD con modulación BPSK, tasa de codificación $r = 3/4$ y 20MHz.

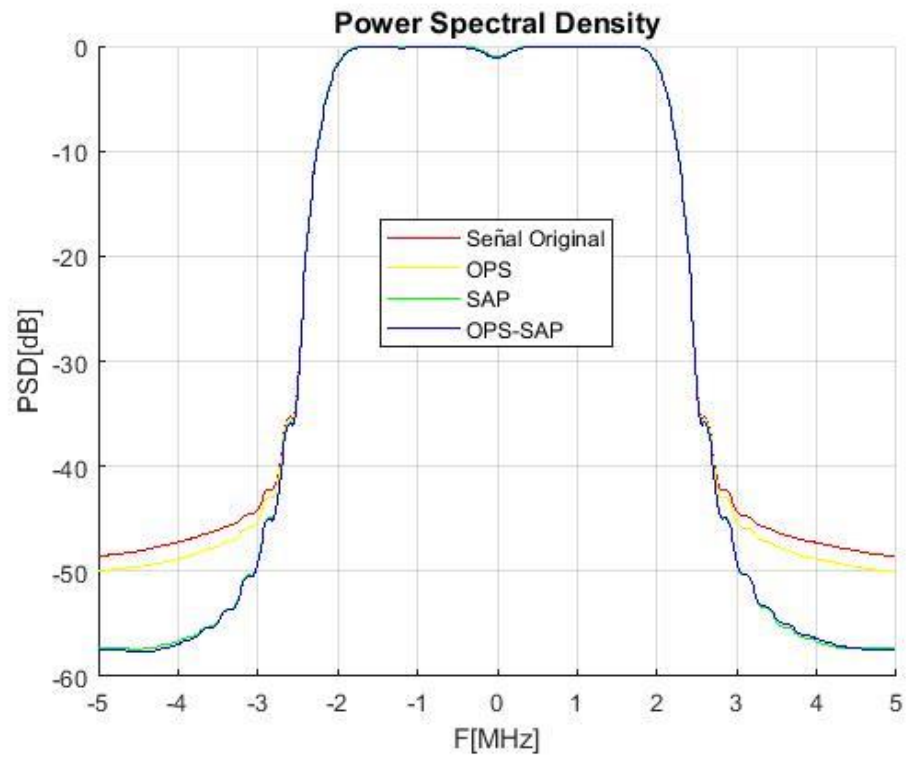


Figura 3.10 PSD con modulación BPSK, tasa de codificación $r = 3/4$ y 10MHz.

3.3.4. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN BPSK Y TASA DE CODIFICACIÓN 3/4.

3.3.4.1. BER vs SNR

En la Figura 3.7 se observa que para un valor de $BER = 10^{-3}$, la técnica OPS-SAP tiene una ganancia en la SNR aproximadamente de 0.5 [dB] respecto a la señal original.

Para el estándar IEEE 802.11a sobre un canal Rician, en la Figura 3.7 se observa que para un valor de $BER = 10^{-2}$, la técnica OPS-SAP alcanza una ganancia alrededor de 1 [dB] respecto a la señal original. Mientras tanto, para los estándares IEEE 802.11g/p y un valor de $BER = 10^{-3}$, la técnica OPS-SAP obtiene una ganancia alrededor de 0.6 [dB] respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, en la Figura 3.7 se observa que el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original. La degradación de la BER es mayor en relación a una transmisión por un canal con desvanecimiento Rician.

3.3.4.2. CCDF de la PAPR

En la Figura 3.8 se observa que para una probabilidad de 10^{-3} , la reducción de la PAPR que obtiene OPS es aproximadamente 1 [dB], SAP 2.4 [dB] y OPS-SAP 3.3 [dB] con respecto a la señal original.

3.3.4.3. PSD

En las Figuras 3.9 y 3.10 se encuentran los resultados para un ancho de banda de 20 y 10 MHz respectivamente. En las figuras se observa que la radiación fuera de banda perteneciente a las señales con las técnicas OPS, SAP y OPS-SAP se alejan hacia abajo con respecto a la señal original, indicando una reducción de la radiación fuera de banda. La técnica OPS obtiene una reducción de la radiación fuera de banda de 2 [dB], SAP 8 [dB] y OPS-SAP de 9 [dB] respecto a la señal original.

3.3.5. ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 1/2.

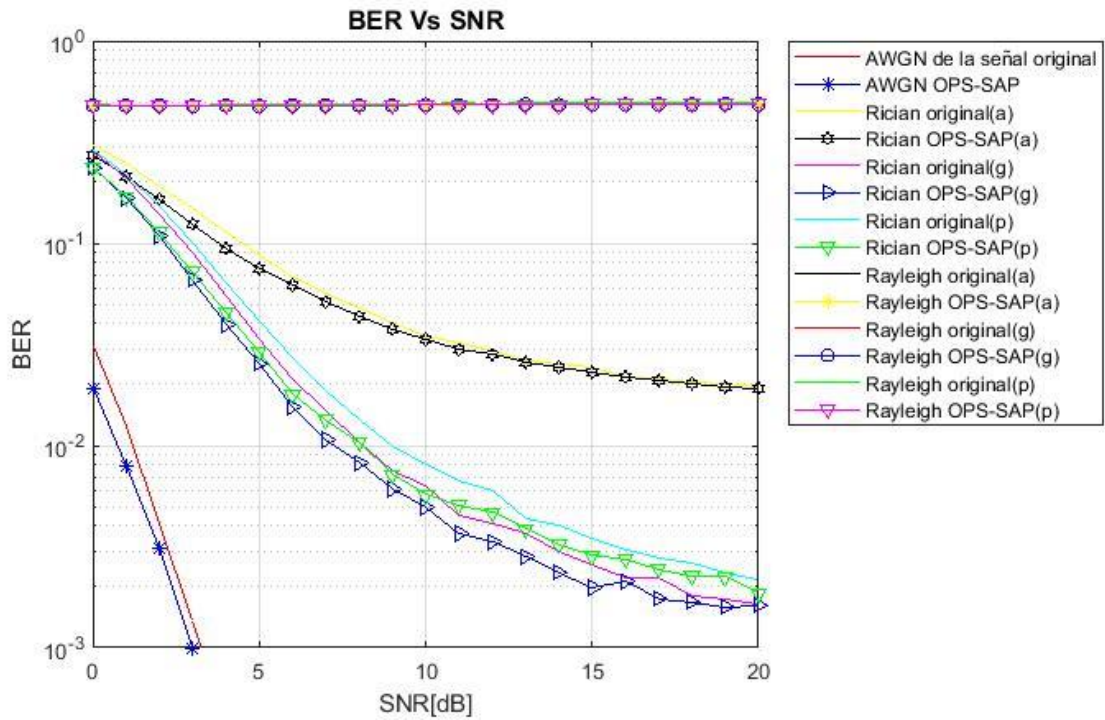


Figura 3.11 BER vs SNR para una modulación QPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz)..

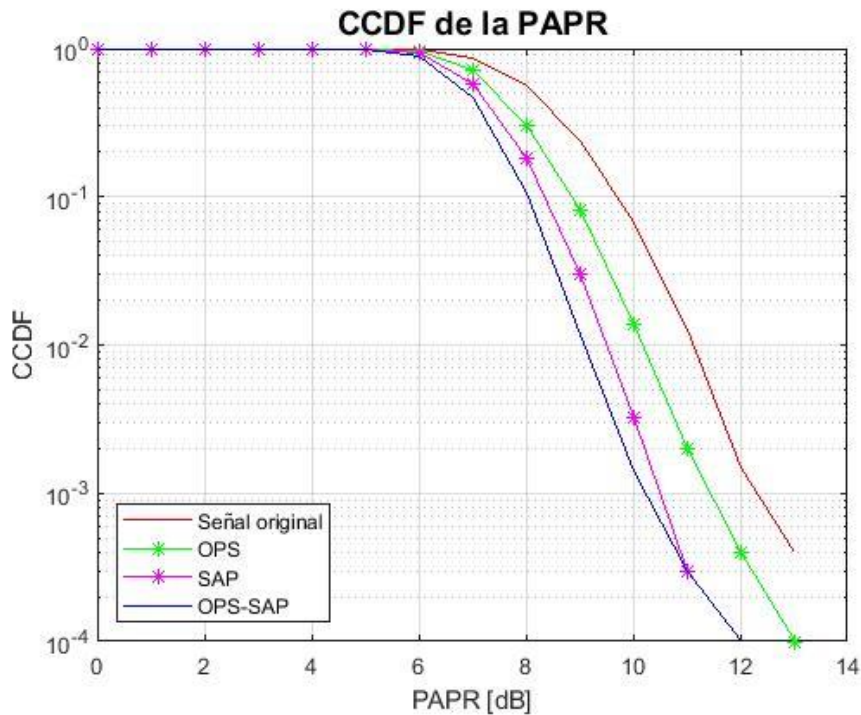


Figura 3.12 CCDF de la PAPR con modulación QPSK y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

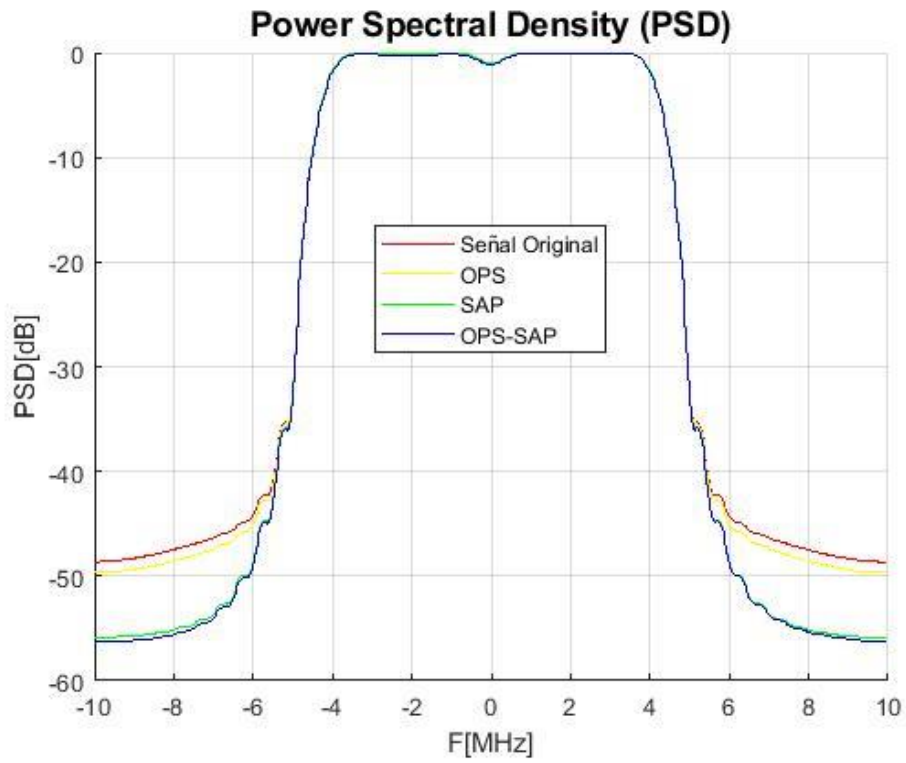


Figura 3.13 PSD con modulación QPSK, tasa de codificación $r = 1/2$ y 20MHz

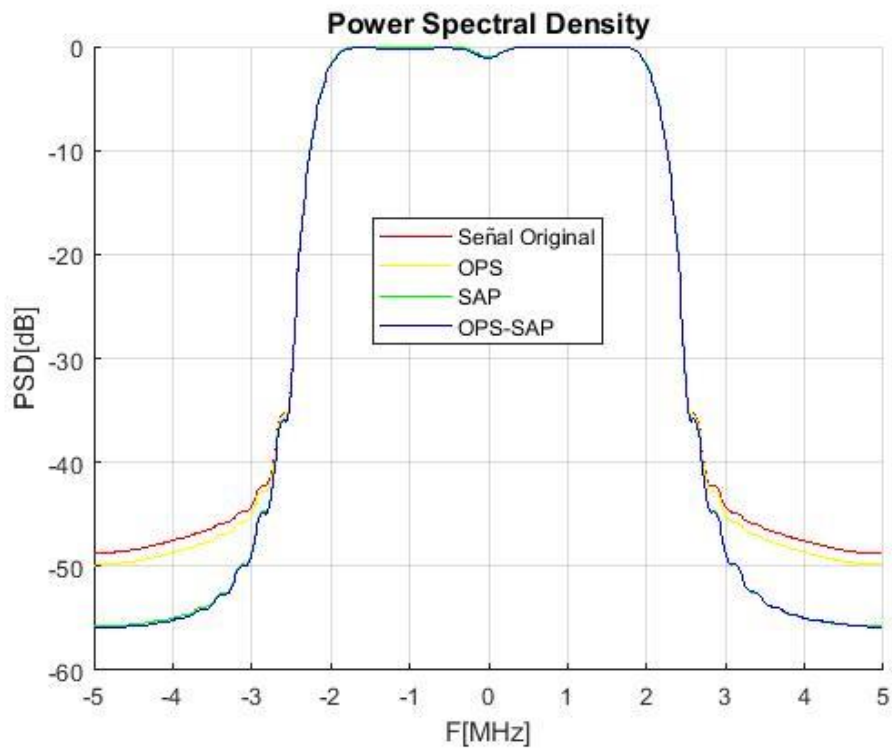


Figura 3.14 PSD con modulación QPSK, tasa de codificación $r = 1/2$ y 10MHz.

3.3.6. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 1/2.

3.3.6.1. BER vs SNR

En la Figura 3.11 se observa que para un valor de $BER = 10^{-3}$, la técnica OPS-SAP obtiene una ganancia alrededor de 0.5 [dB] en la SNR respecto a la señal original.

En la Figura 3.11 se observa que para el estándar IEEE 802.11a sobre el canal Rician y para un valor de $BER = 10^{-1}$, la técnica OPS-SAP obtiene una ganancia en la SNR alrededor de 0.7 [dB] respecto a la señal original. Por otra parte, en la Figura 3.11 se observa que los estándares IEEE 802.11g/p sobre un canal Rician y para un valor de $BER = 10^{-2}$, la técnica OPS-SAP obtiene una ganancia en la SNR aproximadamente de 0.9 [dB] respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, en la Figura 3.11 se observa que el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original.

3.3.6.2. CCDF de la PAPR

Para un esquema de modulación QPSK existe una reducción menor de la PAPR con respecto a la que presenta el esquema de modulación BPSK. En la Figura 3.12 se observa que para una probabilidad de 10^{-3} , la técnica OPS obtiene una reducción de la PAPR aproximadamente de 0.8 [dB], SAP 1.7 [dB] y OPS-SAP de 2.2 [dB] con respecto a la señal original.

3.3.6.3. PSD

Para un esquema de modulación QPSK la tendencia que siguen las señales en cuanto a la radiación fuera de banda es similar a la mostrada en el esquema de modulación BPSK. En las Figuras 3.13 y 3.14 se observan los resultados para un ancho de banda de 20 y 10 MHz respectivamente, donde la técnica OPS tiene una reducción de la radiación fuera de banda aproximadamente de 2 [dB], SAP 6 [dB] y OPS-SAP de 8 [dB] respecto a la señal original.

3.3.7. ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 3/4.

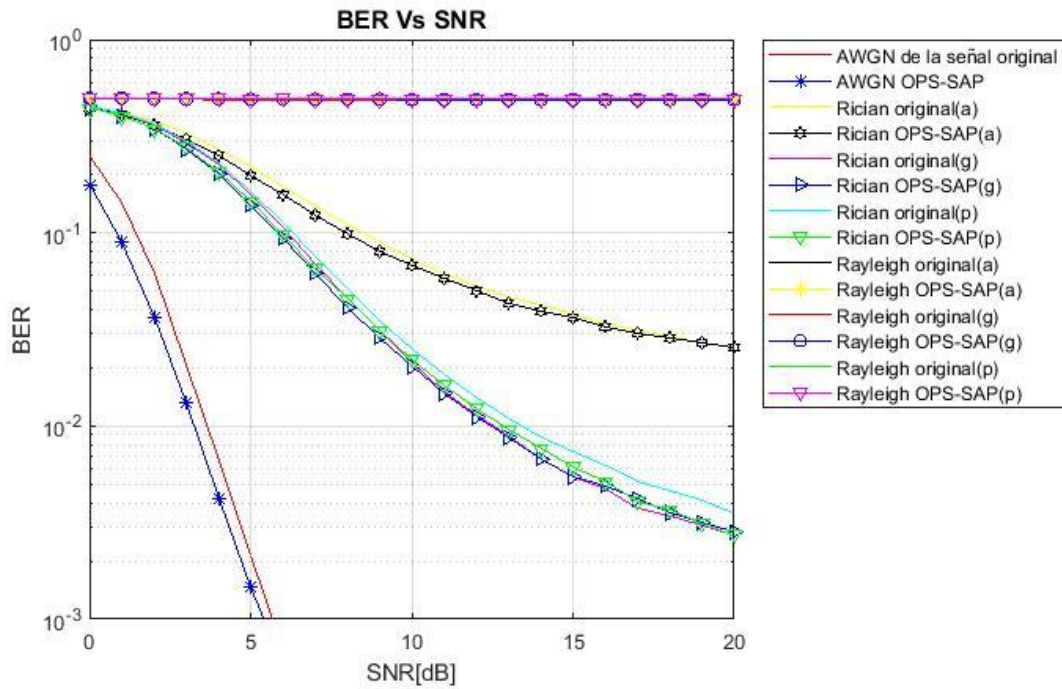


Figura 3.15 BER vs SNR para una modulación QPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

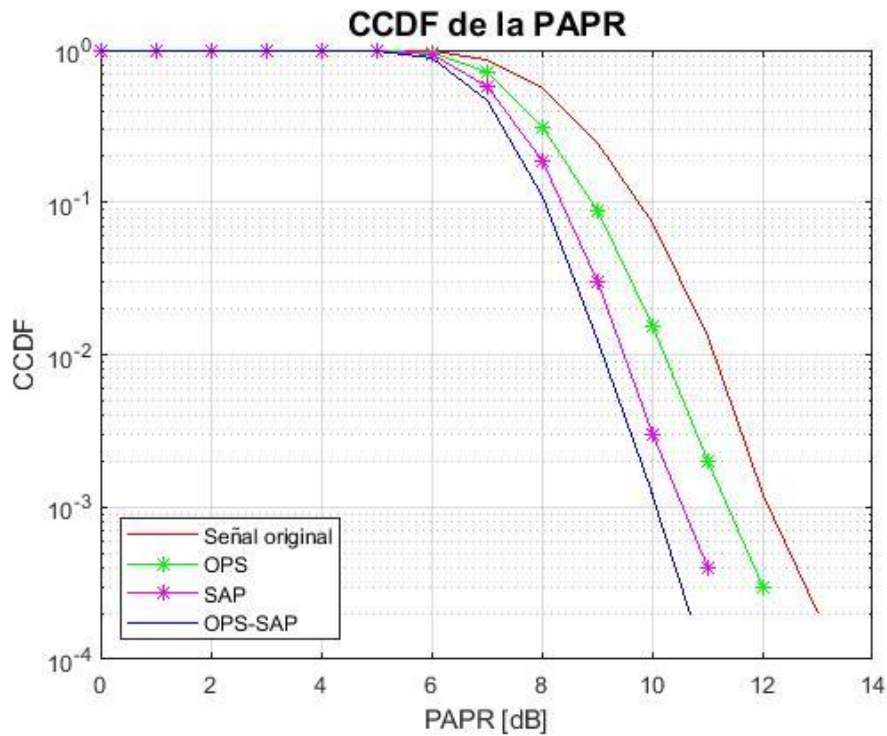


Figura 3.16 CCDF de la PAPR con modulación QPSK y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

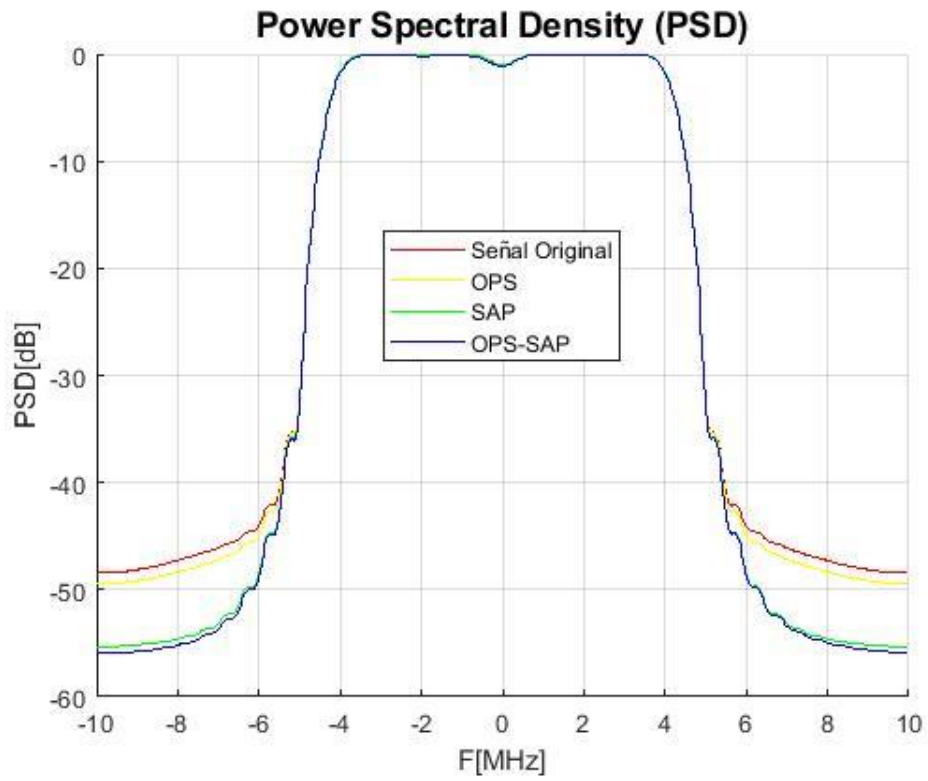


Figura 3.17 PSD con modulación QPSK, tasa de codificación $r = 3/4$ y 20 MHz.

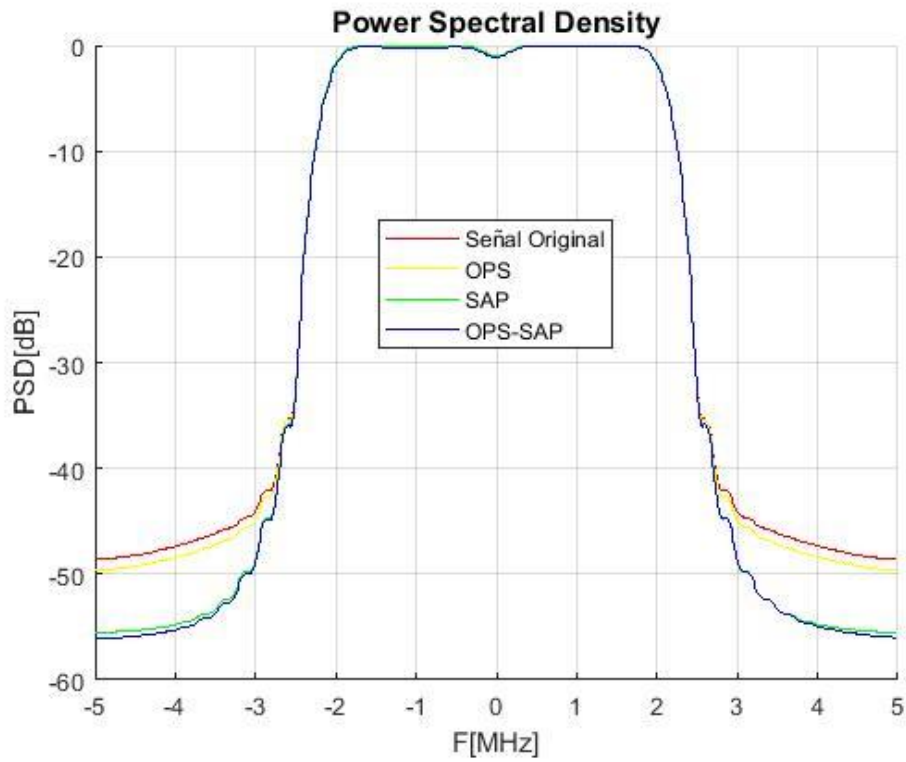


Figura 3.18 PSD con modulación QPSK, tasa de codificación $r = 3/4$ y 10 MHz.

3.3.8. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN QPSK Y TASA DE CODIFICACIÓN 3/4.

3.3.8.1. BER vs SNR

En la Figura 3.15 se observa que para un valor de $BER = 10^{-3}$, la técnica OPS-SAP tiene una ganancia en la SNR aproximadamente de 0.4 [dB] respecto a la señal original.

En la Figura 3.15 se observa que los estándares IEEE 802.11g/p sobre un canal Rician obtienen una ganancia en la SNR menor con relación al valor presentado en la Figura 3.11 para un mismo valor de BER. Para un valor de $BER = 10^{-2}$, la técnica OPS-SAP obtiene una ganancia en la SNR alrededor de 0.4 [dB] respecto a la señal original. Mientras que para el estándar IEEE 802.11a sobre un canal Rician para un valor de $BER = 10^{-1}$ la técnica OPS-SAP obtiene una ganancia de 0.3 [dB] respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, en la Figura 3.15 se observa que el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original.

3.3.8.2. CCDF de la PAPR

En la Figura 3.16 se observa que para una probabilidad de 10^{-3} , la técnica OPS obtiene una reducción de la PAPR aproximadamente de 0.7 [dB], SAP 1.7 [dB] y OPS-SAP de 2.1 [dB] con respecto a la señal original.

3.3.8.3. PSD

En las Figuras 3.17 y 3.18 se encuentran los resultados para un ancho de banda de 20 y 10 MHz respectivamente. En las figuras se puede observar que la técnica OPS obtiene una reducción de la radiación fuera de banda aproximadamente de 2 [dB], SAP 6 [dB] y OPS-SAP de 7 [dB] respecto a la señal original.

3.3.9. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 1/2.

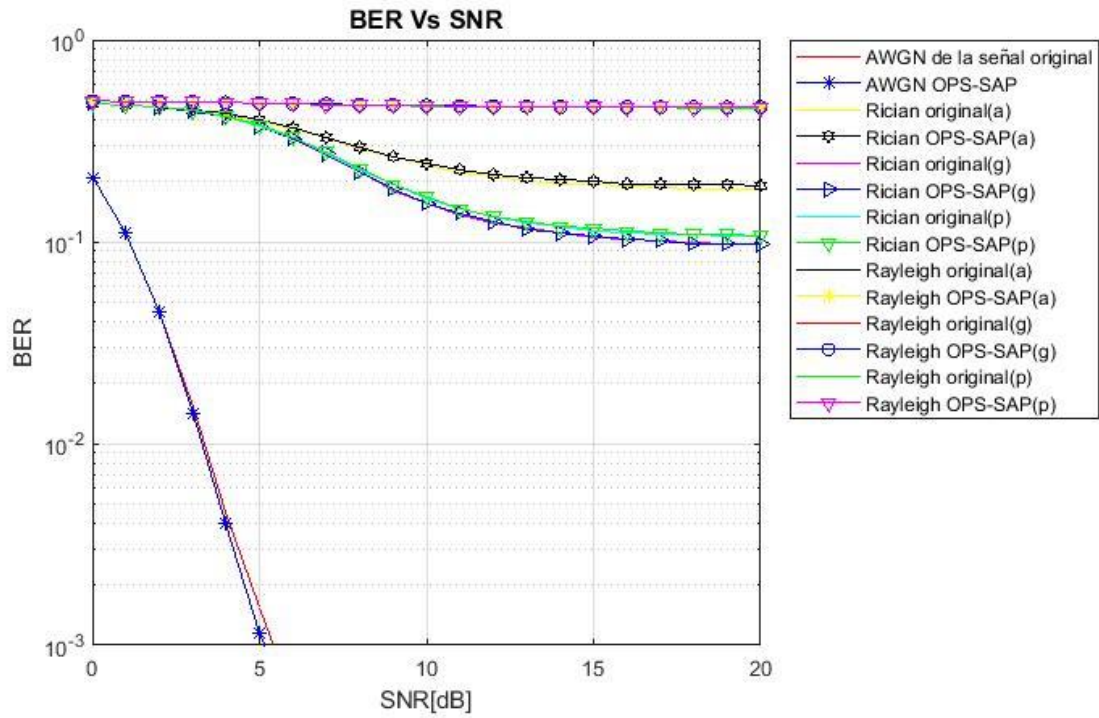


Figura 3.19 BER vs SNR para una modulación 16-QAM y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

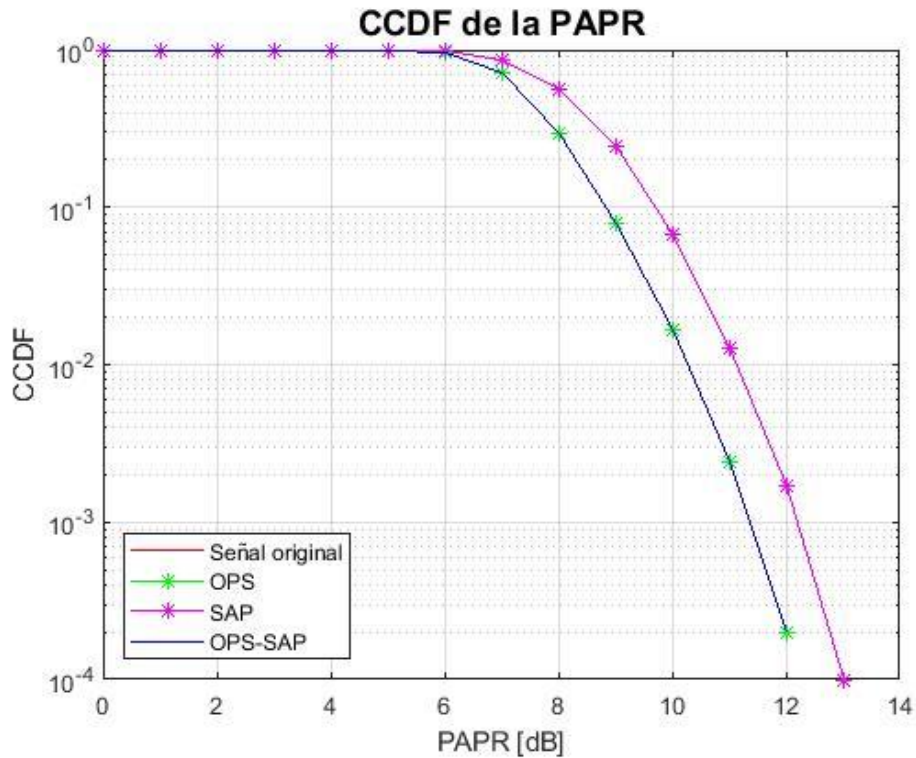


Figura 3.20 CCDF de la PAPR con modulación 16-QAM y tasa de codificación $r = 1/2$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

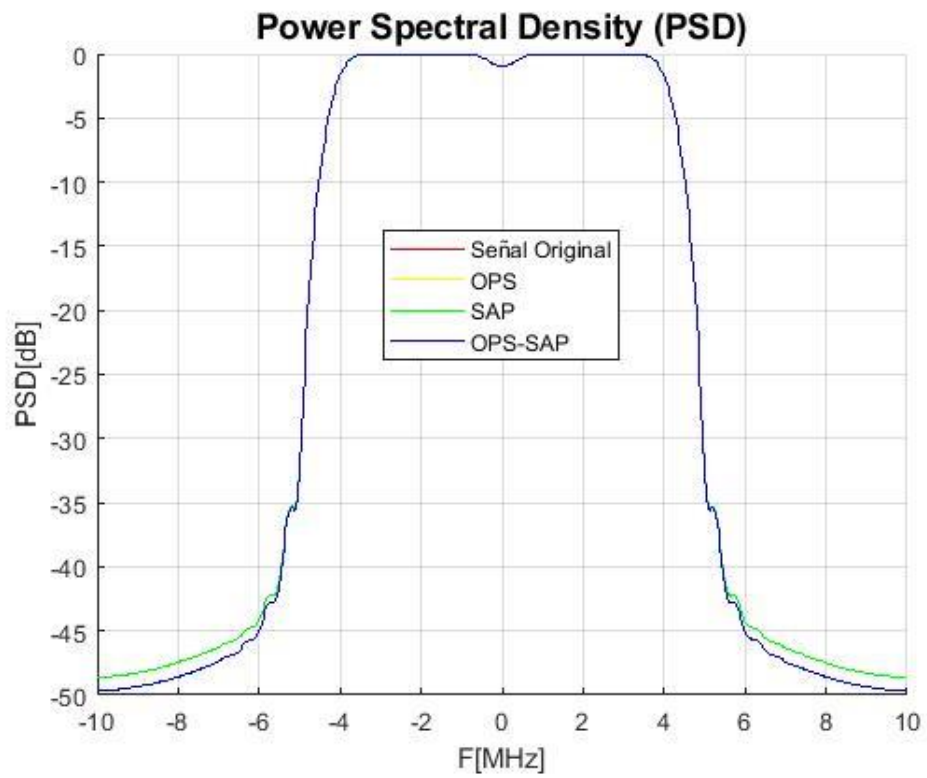


Figura 3.21 PSD con modulación 16-QAM, tasa de codificación $r = 1/2$ y 20MHz.

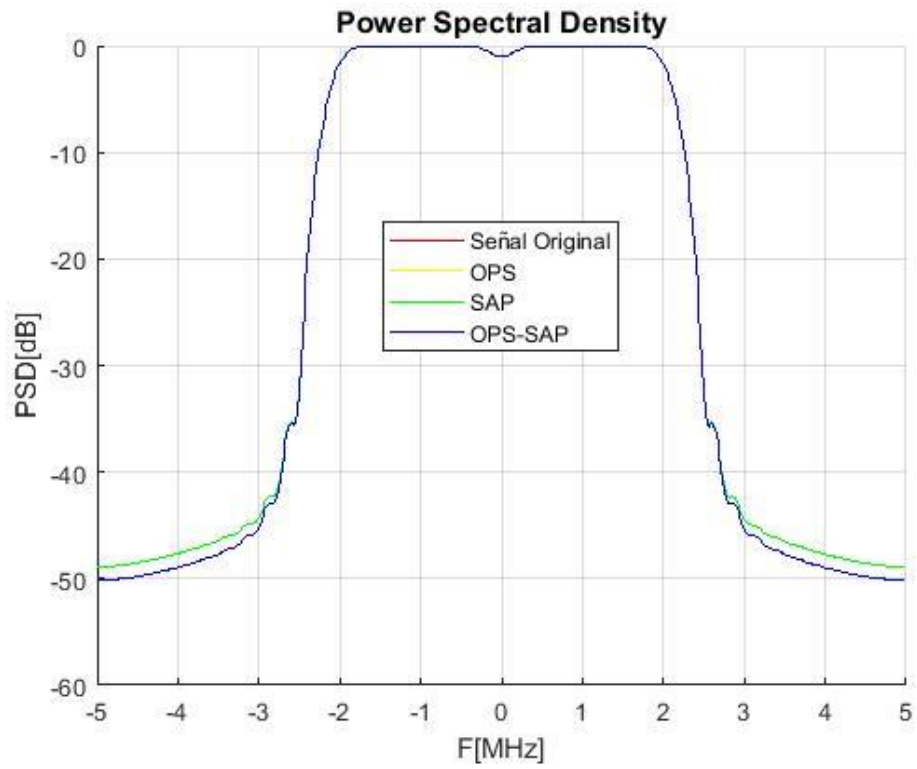


Figura 3.22 PSD con modulación 16-QAM, tasa de codificación $r = 1/2$ y 10MHz

3.3.10. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 1/2.

3.3.10.1. BER vs SNR

En la Figura 3.19 se observa que la ganancia en la SNR disminuye con respecto al valor que presenta el esquema de modulación QPSK (Figura 3.15) para un mismo valor de BER. La Figura 3.19 muestra que para un valor de $BER = 10^{-3}$, la técnica OPS-SAP presenta una ganancia alrededor de 0.3 [dB] en la SNR respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rician, en la Figura 3.19 se observa que el valor de la SNR de la señal con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal Original.

El comportamiento que presentan los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, es similar al que se tiene sobre un canal Rician, en la Figura 3.19 se observa que el valor de la SNR de las señales con la técnica OPS-SAP es similar a la señal original, sin embargo, la degradación de la BER en un canal Rician es menor al que presenta un canal Rayleigh.

3.3.10.2. CCDF de la PAPR

Para un esquema de modulación 16-QAM se observa que la reducción de la PAPR con respecto a un esquema de modulación QPSK es menor. En la Figura 3.20 se observa que para una probabilidad de 10^{-3} , la técnica OPS presenta una reducción de la PAPR aproximadamente de 1.1 [dB] al igual que OPS-SAP con respecto a la señal original. SAP no presenta una reducción de la PAPR respecto a la señal original.

3.3.10.3. PSD

En un esquema de modulación 16-QAM. la reducción de la radiación fuera de banda es significativamente menor con relación a la presentada en QPSK. En las Figuras 3.21 y 3.22 se observan los resultados para un ancho de banda de 20 y 10 MHz respectivamente. Las figuras muestran que las señales pertenecientes a la técnica OPS y OPS-SAP presentan una reducción similar de la radiación fuera de banda cuyo valor es aproximadamente 2 [dB] respecto a la señal original. La señal con técnica SAP no presenta una reducción de la radiación fuera de banda.

3.3.11. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 3/4.

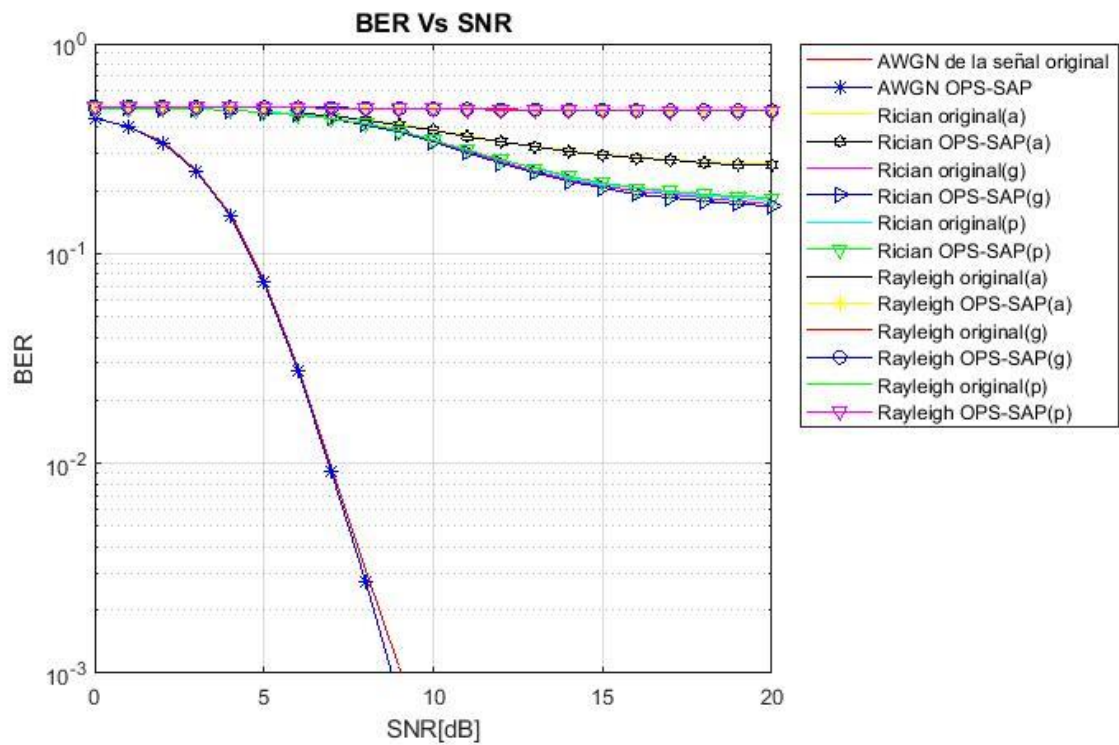


Figura 3.23 BER vs SNR para una modulación 16-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

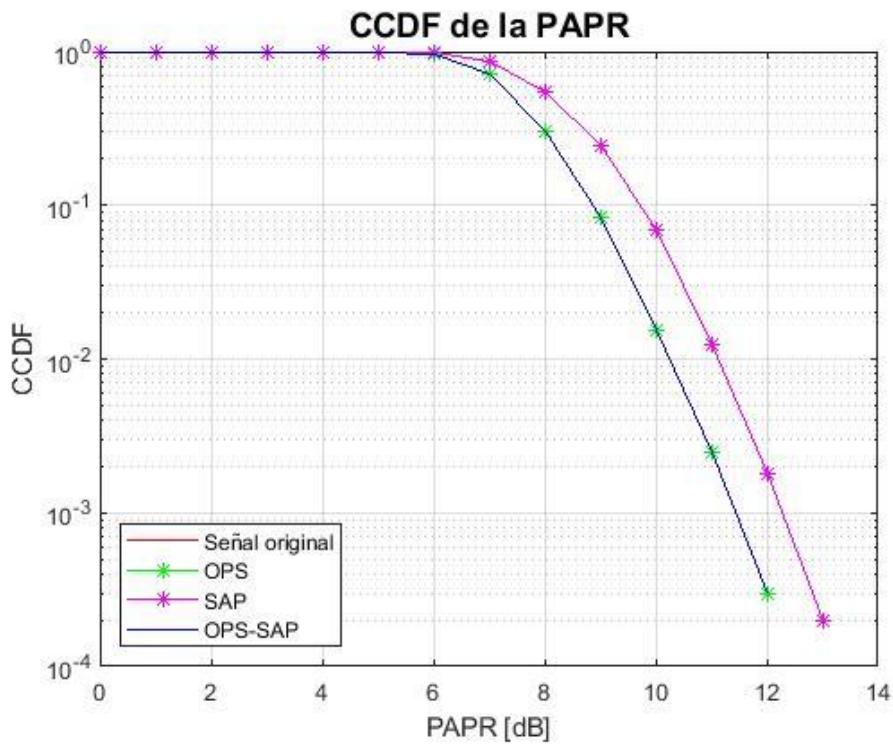


Figura 3.24 CCDF de la PAPR con modulación 16-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

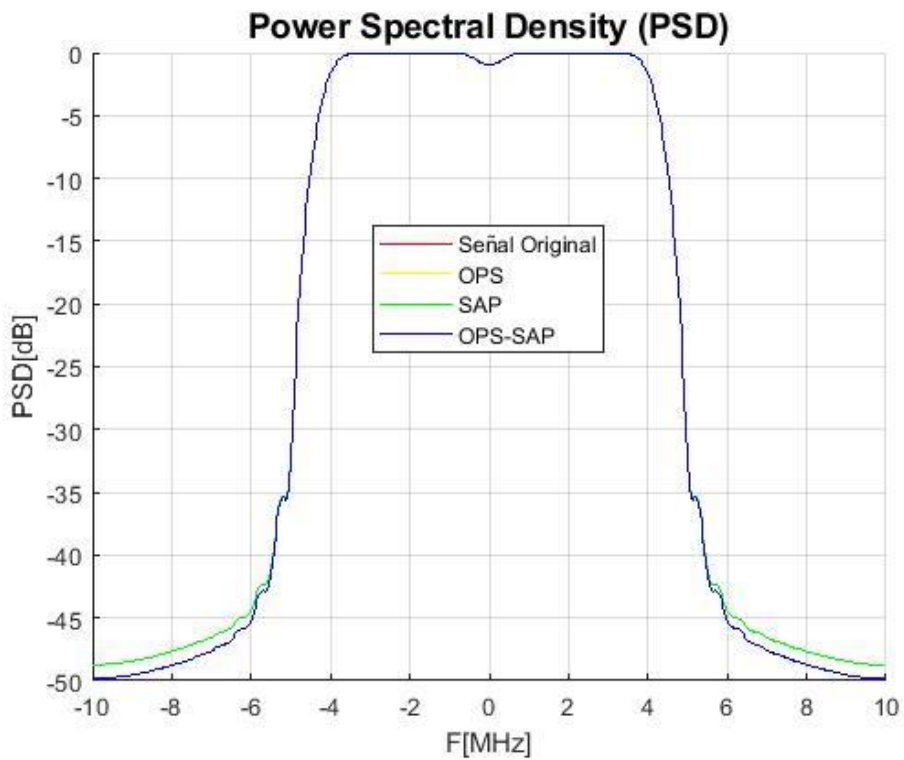


Figura 3.25 PSD con modulación 16-QAM, tasa de codificación $r = 3/4$ y 20 MHz.

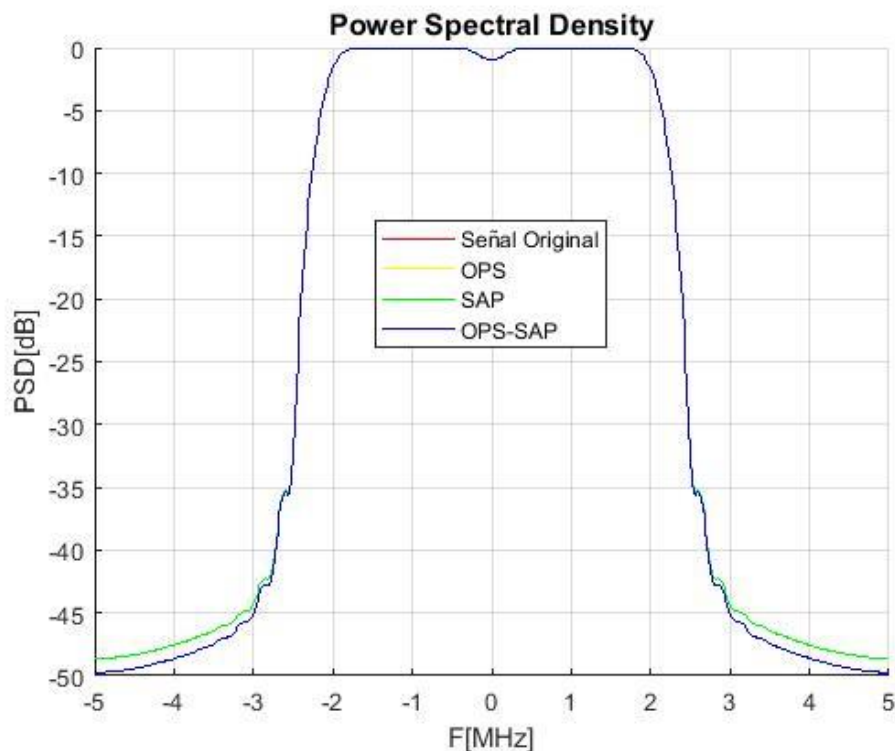


Figura 3.26 PSD con modulación 16-QAM, tasa de codificación $r = 3/4$ y 10 MHz.

3.3.12. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 3/4.

3.3.12.1. BER vs SNR

En la Figura 3.23 se observa que para un valor de $BER = 10^{-3}$, la técnica OPS-SAP presenta una ganancia alrededor de 0.3 [dB] en la SNR respecto a la señal original.

Para los estándares IEEE 802.11a/g/p sobre un canal Rician, en la Figura 3.23 se observa que la señal con la técnica OPS-SAP presenta un valor en la SNR similar o ligeramente menor con respecto a la señal Original.

El comportamiento que presentan los estándares IEEE 802.11a/g/p sobre un canal Rayleigh se observa en la Figura 3.23, donde el valor de la SNR de las señales con la técnica OPS-SAP es similar a la señal original, sin embargo, la degradación de la BER en un canal Rician es menor al que presenta un canal Rayleigh.

3.3.12.2. CCDF de la PAPR

En la Figura 3.24 se aprecia que para una probabilidad de 10^{-3} la técnica OPS presenta una reducción de la PAPR alrededor de 1.2 [dB] al igual que OPS-SAP con respecto a la señal original. Por otra parte, la técnica SAP no presenta una reducción de la PAPR con respecto a la señal original.

3.3.12.3. PSD

En las Figuras 3.25 y 3.26 se observan los resultados para un ancho de banda de 20 y 10 MHz respectivamente. Las figuras muestran que las señales pertenecientes a la técnica OPS y OPS-SAP presentan una reducción similar de la radiación fuera de banda cuyo valor es aproximadamente 2 [dB] respecto a la señal original. La señal con técnica SAP no presenta una reducción de la radiación fuera de banda.

3.3.13. ESQUEMA DE MODULACIÓN 64-QAM Y TASA DE CODIFICACIÓN 2/3.

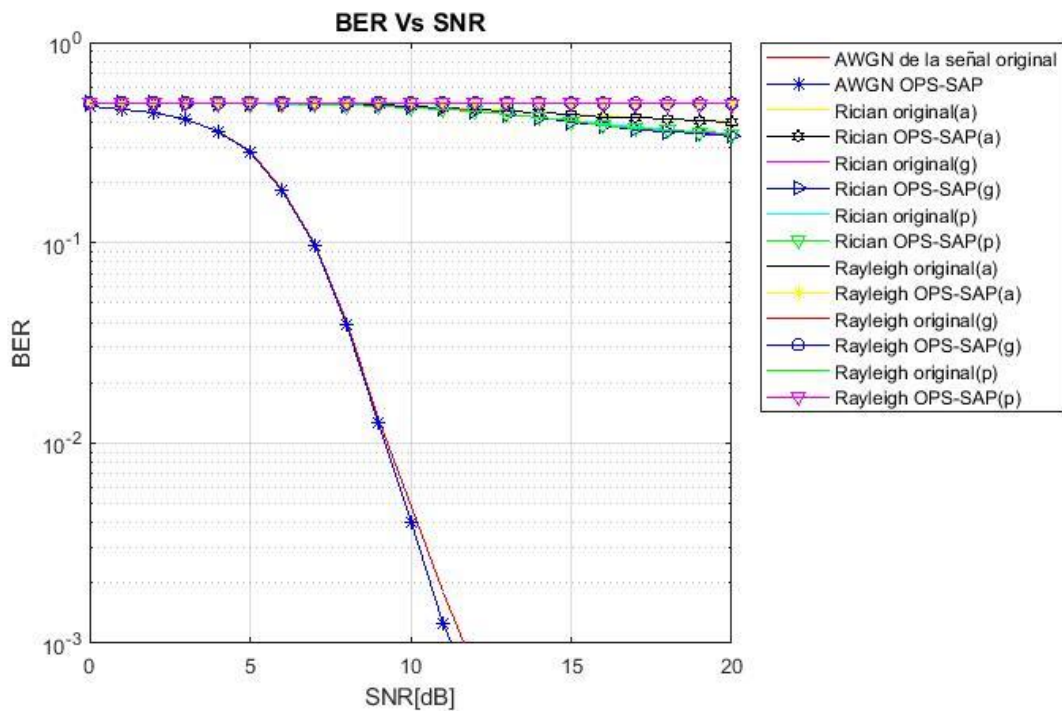


Figura 3.27 BER vs SNR para una modulación 64-QAM y tasa de codificación $r = 2/3$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

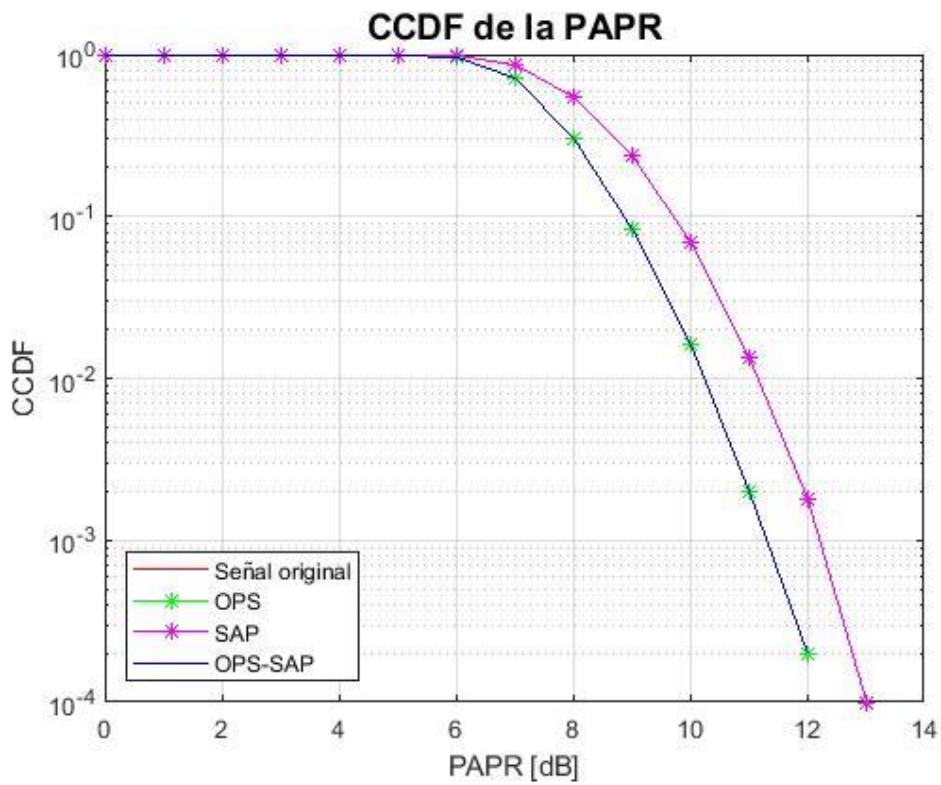


Figura 3.28 CCDF de la PAPR con modulación 64-QAM y tasa de codificación $r = 2/3$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

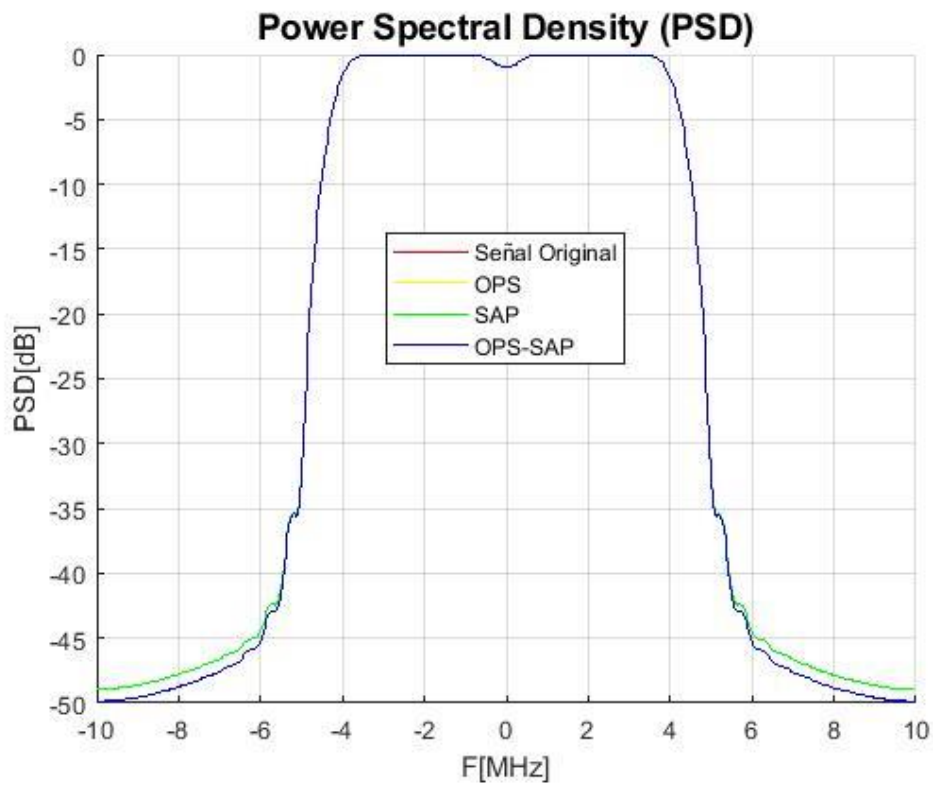


Figura 3.29 PSD con modulación 64-QAM, tasa de codificación $r = 2/3$ y 20 MHz.

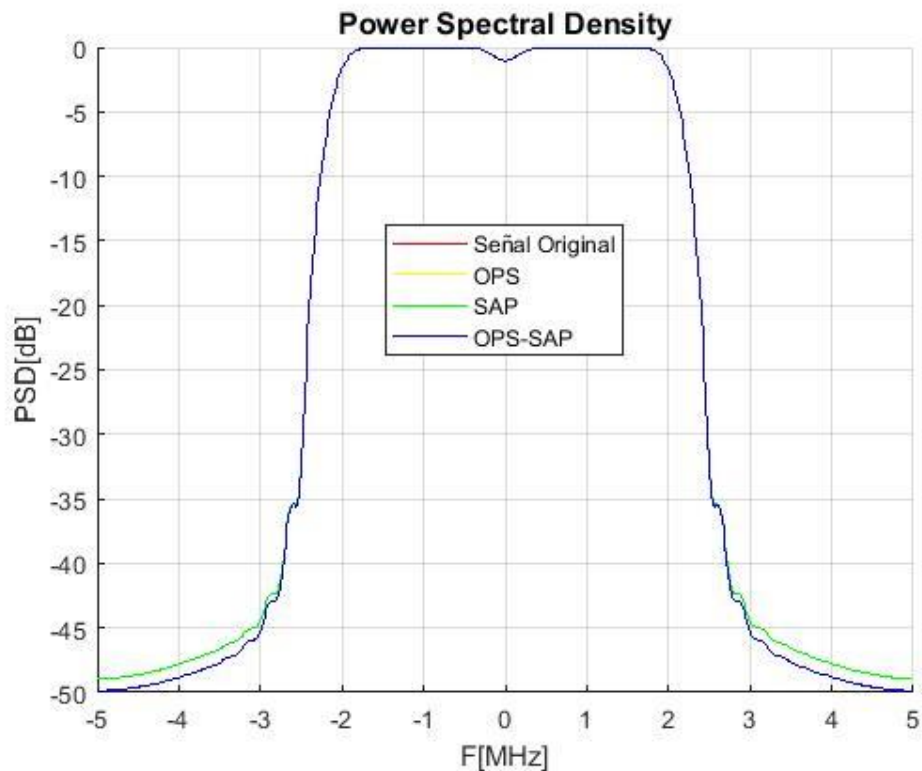


Figura 3.30 PSD con modulación 64-QAM, tasa de codificación $r = 2/3$ y 10 MHz.

3.3.14. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 64-QAM Y TASA DE CODIFICACIÓN 2/3.

3.3.14.1. BER vs SNR

En la Figura 3.27 se observa que para un valor de $BER = 10^{-3}$ la técnica OPS-SAP presenta una ganancia alrededor de 0.3 [dB] en la SNR respecto a la señal original.

En los estándares IEEE 802.11a/g/p sobre un canal Rician, el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal Original esto se observa en la Figura 3.27.

En los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original, este resultado se observa en la Figura 3.27.

El comportamiento que presentan los estándares IEEE 802.11a/g/p sobre un canal Rayleigh se observa en la Figura 3.27, dónde el valor de la SNR de las señales con la técnica OPS-SAP es similar a la señal original.

3.3.14.2. CCDF de la PAPR

Para un esquema de modulación 64-QAM las curvas de la CCDF de la PAPR siguen un patrón similar al presentado en el esquema de modulación 16-QAM, puesto que se observa el mismo comportamiento de la técnica SAP, la cual no presenta una reducción de la PAPR respecto a la señal original. En la Figura 3.28 se observan los resultados, para una probabilidad de 10^{-3} la técnica OPS presenta una reducción de la PAPR aproximadamente de 1.1 [dB] al igual que OPS-SAP con respecto a la señal original.

3.3.14.3. PSD

Para un esquema de modulación 64-QAM la reducción de la radiación fuera de banda sigue una tendencia similar al presentada en el esquema de modulación 16-QAM, es decir la reducción que presenta las técnicas OPS y OPS-SAP es similar, mientras tanto, SAP no presenta ninguna reducción de la radiación fuera de banda con respecto a la señal original.

En las Figuras 3.29 y 3.30 se observan los resultados para un ancho de banda de 20 y 10 MHz respectivamente, En las figuras se observa que las señales pertenecientes a la técnica OPS y OPS-SAP presentan una reducción de la radiación fuera de banda de 2 [dB] respecto a la señal original.

3.3.15. ESQUEMA DE MODULACIÓN 16-QAM Y TASA DE CODIFICACIÓN 3/4.

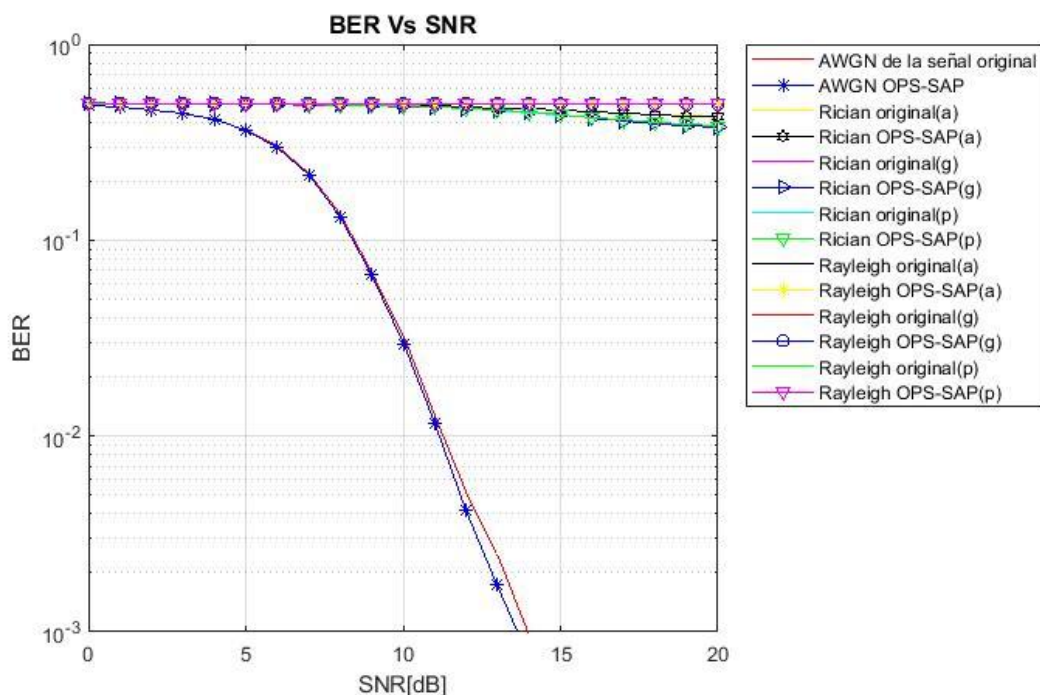


Figura 3.31 BER vs SNR para una modulación 64-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

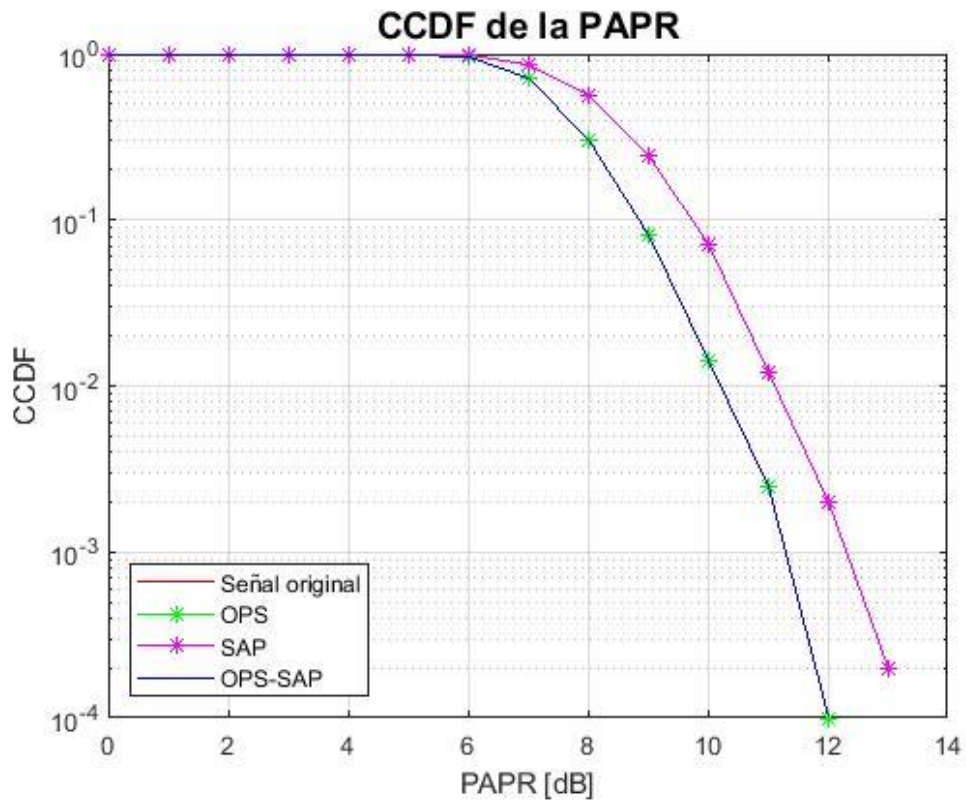


Figura 3.32 CCDF de la PAPR con modulación 64-QAM y tasa de codificación $r = 3/4$ para los estándares IEEE 802.11a/g/p (AB=10 y 20 MHz).

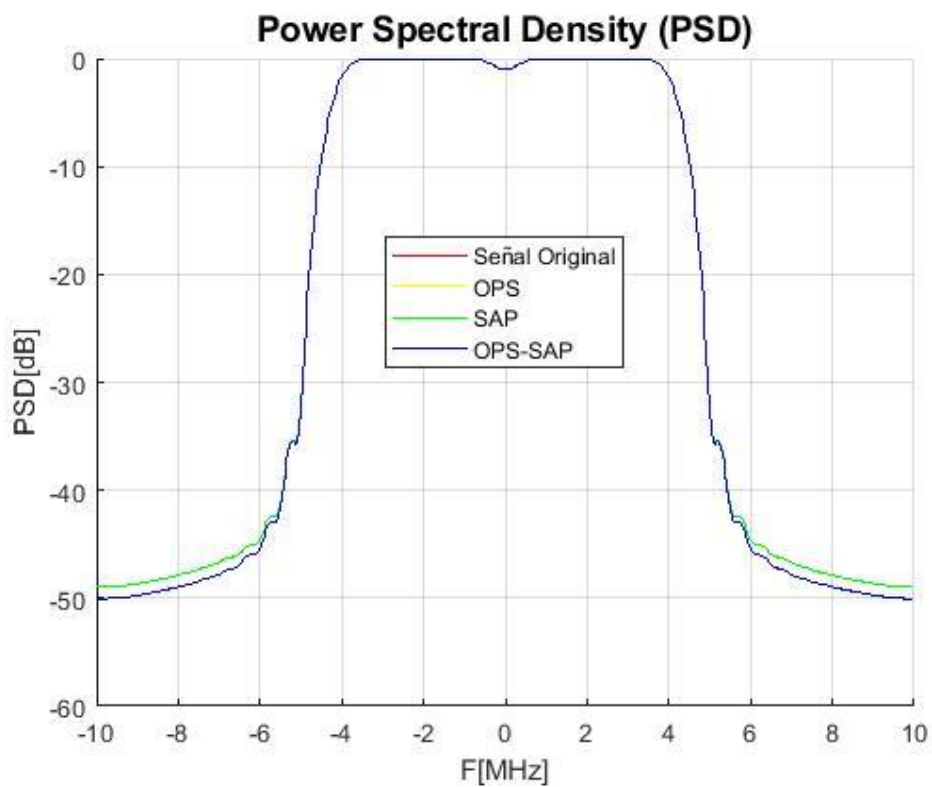


Figura 3.33 PSD con modulación 64-QAM, tasa de codificación $r = 3/4$ y 20 MHz.

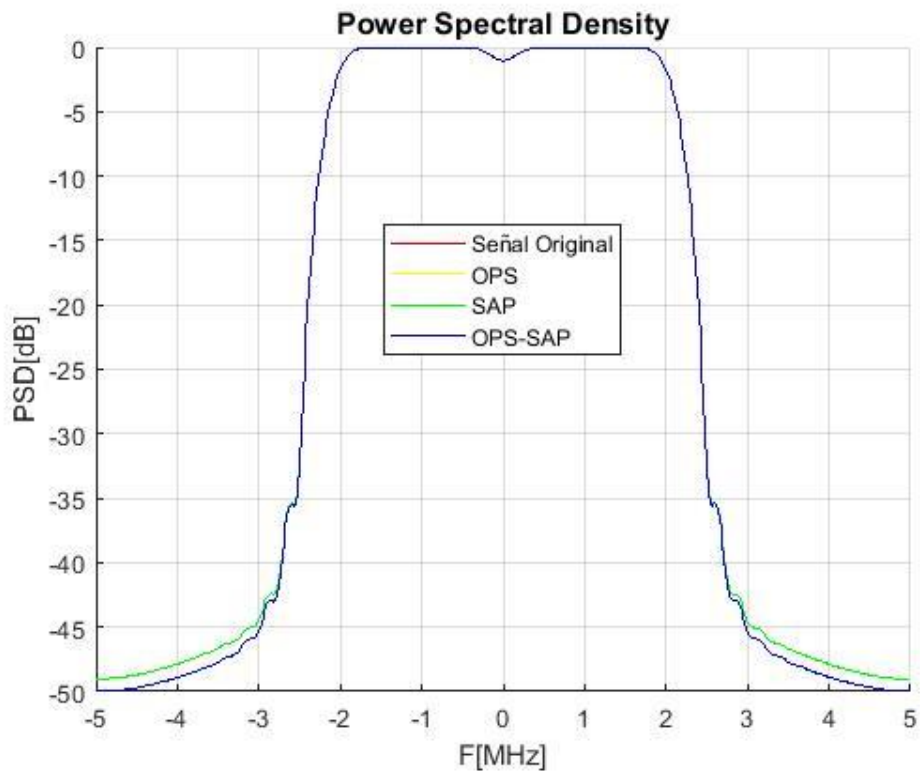


Figura 3.34 PSD con modulación 64-QAM, tasa de codificación $r = 3/4$ y 10 MHz.

3.3.16. ANÁLISIS DE RESULTADOS DEL ESQUEMA DE MODULACIÓN 64-QAM Y TASA DE CODIFICACIÓN 3/4.

3.3.16.1. BER vs SNR

En la Figura 3.31 se observa que para un valor de $BER = 10^{-3}$ la técnica OPS-SAP presenta una ganancia alrededor de 0.3 [dB] en la SNR respecto a la señal original.

En los estándares IEEE 802.11a/g/p sobre un canal Rician, el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal Original esto se observa en la Figura 3.31.

En los estándares IEEE 802.11a/g/p sobre un canal Rayleigh, el valor de la SNR de las señales con la técnica OPS-SAP es similar o ligeramente menor con respecto a la señal original, este resultado se observa en la Figura 3.31.

El comportamiento que presentan los estándares IEEE 802.11a/g/p sobre un canal Rayleigh se observa en la Figura 3.31, donde el valor de la SNR de las señales con la técnica OPS-SAP es similar al valor de la señal original.

3.3.16.2. CCDF de la PAPR

En la Figura 3.32 se observa que para una probabilidad de 10^{-3} , la técnica OPS obtiene una reducción de la PAPR aproximadamente de 0.9 [dB], al igual que OPS-SAP respecto a la señal original. La técnica SAP no presenta una reducción de la PAPR respecto a la señal original.

3.3.16.3. PSD

En las Figuras 3.33 y 3.34 se observan los resultados para un ancho de banda de 20 y 10 MHz respectivamente, En las figuras se observa que la técnica OPS-SAP al igual que OPS obtienen una reducción de la radiación fuera de banda de 1.5 [dB] respecto a la señal original.

3.4. ANÁLISIS DE LOS RESULTADOS DEL SISTEMA DE COMUNICACIÓN IEEE 802.11a/g/p

A continuación, se analizan los resultados para cada escenario de simulación

3.4.1. ANÁLISIS DE LOS RESULTADOS BER VS SNR

Una vez obtenidos los resultados pertenecientes a un canal con ruido AWGN se afirma que, para los esquemas de modulación BPSK y QPSK la señal con técnica OPS-SAP presenta una ganancia en la SNR aproximadamente de 0.5 [dB] con respecto a la señal original, mientras tanto, para los esquemas de modulación 16-QAM y 64-QAM la señal con técnica OPS-SAP presenta una ganancia en la SNR aproximadamente de 0.3 [dB] respecto a la señal original, para un valor de $BER = 10^{-3}$.

Los resultados pertenecientes a un canal con ruido AWGN y desvanecimiento Rician, para los estándares IEEE 802.11g/p se afirma que la ganancia en el valor de la SNR de la señal con la técnica OPS-SAP para un valor de $BER = 10^{-3}$ es alrededor de 1 [dB] con respecto a la señal original, esto ocurre en el esquema de modulación BPSK que representa el mejor resultado.

El valor de la SNR para un canal con ruido AWGN con desvanecimiento Rayleigh presenta un comportamiento similar en los tres estándares IEEE 802.11a/g/p pese a que poseen diferentes valores de PDP, el valor de la SNR de la señal con la técnica OPS-SAP es menor o similar en comparación a la señal original, nunca dicho valor es superior.

Los resultados que presentan las señales que atraviesan por un canal inalámbrico con desvanecimiento Rician presentan una degradación menor de la BER con respecto a las

señales que atraviesan por un canal con desvanecimiento Rayleigh, puesto que Rician a diferencia de Rayleigh posee línea de vista.

3.4.2. ANÁLISIS DE LOS RESULTADOS DE LA CCDF DE LA PAPR

La técnica OPS presenta una menor reducción de la PAPR con respecto a las otras técnicas en esquemas de modulación BPSK y QPSK, mientras que para esquemas de modulación como 16-QAM y 64-QAM la reducción de la PAPR es mayor a la alcanzada por SAP y similar a la que se obtiene con la técnica OPS-SAP.

La técnica SAP presenta una reducción significativa de la PAPR únicamente en los esquemas de modulación BPSK y QPSK, ya que para los esquemas de modulación 16-QAM y 64-QAM su aporte a la reducción de la PAPR es nula. Esto ocurre porque únicamente los puntos ubicados en las esquinas del diagrama de constelación son predistorcionados y cuyo número es inferior en comparación al número total de puntos disponibles.

Finalmente, la técnica OPS-SAP presenta los mejores resultados para la reducción de la PAPR para todos los escenarios de simulación en comparación con OPS y SAP. El resultado de mayor éxito se presenta para el esquema de modulación BPSK con una reducción superior a los 3 [dB], seguido por QPSK que obtiene una reducción alrededor de 2 [dB], mientras que, para un esquema de modulación 16-QAM la reducción es superior a 1 [dB] y para un esquema de modulación 64-QAM se obtiene una reducción alrededor de 1 [dB] para una probabilidad de 10^{-3} .

3.4.3. ANÁLISIS DE LOS RESULTADOS DE LA PSD

La reducción de la radiación fuera de banda es mayor en los esquemas de modulación BPSK y QPSK cuyo valor es superior a los 7 [dB], mientras tanto para los esquemas de modulación 16-QAM y 64-QAM la reducción es menor y ronda un valor de 2 [dB]. Este resultado ocurre porque la reducción de la PAPR es mayor en los esquemas de modulación BPSK y QPSK.

El valor de la reducción de la radiación fuera de banda para la técnica OPS es similar para todos los esquemas de modulación esto se debe a que el número de subportadoras piloto permanece constante para todos los esquemas de modulación.

El resumen con los valores de la reducción de la radiación fuera de banda para cada escenario de simulación se detalla en la Tabla 3.8.

Tabla 3.8 Valores de la reducción de la radiación fuera de banda con respecto a la señal original de cada técnica para cada escenario de simulación.

| Esquema de Modulación | Tasa de codificación | OPS [dB] | SAP [dB] | OPS-SAP [dB] |
|------------------------------|-----------------------------|-----------------|-----------------|---------------------|
| BPSK | $r = 1/2$ | 2 | 9 | 10 |
| BPSK | $r = 3/4$ | 2 | 8 | 9 |
| QPSK | $r = 1/2$ | 2 | 6 | 8 |
| QPSK | $r = 3/4$ | 2 | 6 | 7 |
| 16-QAM | $r = 1/2$ | 2 | 0 | 2 |
| 16-QAM | $r = 3/4$ | 2 | 0 | 2 |
| 64-QAM | $r = 2/3$ | 2 | 0 | 2 |
| 64-QAM | $r = 3/4$ | 2 | 0 | 2 |

4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

En el presente trabajo de titulación se evaluó el rendimiento de la técnica OPS-SAP para la reducción de la PAPR en estándares Ad-Hoc. La transmisión se realizó a través de canales inalámbricos con ruido AWGN, desvanecimiento Rician y Rayleigh. La evaluación del rendimiento se presentó mediante el análisis de curvas de BER vs SNR, CCDF de la PAPR y PSD. La implementación y simulación de la capa física OFDM de los estándares IEEE 802.11a/g/p, así como de las técnicas para la reducción de la PAPR se realizó con éxito en el entorno de desarrollo de *App Designer* perteneciente a Matlab. Luego de la realización de las simulaciones y la obtención de los resultados se obtuvieron las siguientes conclusiones:

- Al aplicar la técnica OPS-SAP se confirma que no existe una degradación de la BER, ya que al distorsionar los puntos pertenecientes al diagrama de constelación no se excede la distancia mínima que existen entre ellos y se emplea menos energía por símbolo como indican las gráficas de BER vs SNR que se encuentran en el capítulo tres. Para todos los estándares IEEE 802.11a/g/p se evidencia que los esquemas de modulación BPSK y QPSK presenta una ganancia en la SNR aproximadamente de 0.5 [dB] con respecto a la señal original debido a que todos los puntos pertenecientes al diagrama de constelación son distorsionados a diferencia de 16-QAM y 64-QAM en los cuales solo algunos puntos son distorsionados.
- Para los estándares IEEE 802.11g e IEEE 802.11p cuya transmisión se realiza a través de un canal con desvanecimiento Rician y ruido AWGN, la técnica OPS-SAP presenta una ganancia en la SNR superior a la que se tiene en una transmisión por un canal AWGN para un valor de BER^{-3} porque el sistema de transmisión OFDM es robusto frente al fenómeno de desvanecimiento Rician. Estos resultados ocurren en el esquema de modulación BPSK como se observa en las gráficas de BER vs SNR. La ganancia de la SNR es alrededor de 0.7 [dB] con respecto a la señal original.
- La técnica OPS-SAP presenta una reducción alrededor de 3 [dB] de la PAPR para el esquema de modulación BPSK y tasa de codificación $r = 1/2$. Este resultado se evidencia al realizar el análisis de las gráficas de la CCDF

de la PAPR presentadas en el capítulo 3 y representa el resultado de mayor éxito porque en este esquema de modulación el número de puntos pertenecientes al diagrama de constelación que son distorsionados es mayor en relación a los otros esquemas, de esta forma la técnica OPS-SAP actúa de forma más eficiente.

- La reducción de la PAPR que se obtiene al aplicar la técnica OPS-SAP baja su eficiencia cuando se tienen esquemas de modulación 16-QAM y 64-QAM. Esto ocurre porque el número de símbolos pertenecientes al diagrama de constelación que pueden ser extendidos (predistorsionados) en estos esquemas de modulación es inferior con relación al número de símbolos totales.
- La técnica OPS presenta una reducción constante en el valor de la PAPR con respecto a la señal original porque el número de subportadoras piloto se mantiene constante y cuyo valor es alrededor de 1 [dB]. Mientras tanto la técnica SAP presenta diferentes valores en cuanto a la reducción de la PAPR con respecto a la señal original, el mejor resultado es alrededor de 2.7 [dB] y esto ocurre en un esquema de modulación BPSK.
- Cuando se utiliza un amplificador HPA en el lado del transmisor OFDM se producen distorsiones fuera de la banda de frecuencia. Dichas distorsiones se reducen considerablemente cuando se aplica la técnica OPS-SAP en un esquema de modulación BPSK y QPSK. Para los esquemas de modulación 16-QAM y 64-QAM el valor de la reducción de las distorsiones se reduce debido a que la eficiencia en la reducción de la PAPR es menor como se observó en el capítulo 3.
- El uso del entorno de desarrollo de *App Designer* para la implementación del sistema de transmisión en ambientes Ad-Hoc con la técnica OPS-SAP y la evaluación de dicha técnica para la reducción de la PAPR ha quedado demostrado, dando la oportunidad para que estudiantes, profesores y demás personas cuenten con resultados detallados para facilitar la realización de futuras investigaciones.

4.2 RECOMENDACIONES

En este trabajo de titulación se obtuvieron buenos resultados en la evaluación de la técnica OPS-SAP para la transmisión de información a través de canales con ruido AWGN, desvanecimiento Rician y Rayleigh, sin embargo, se recomienda el estudio de la transmisión de información por un canal inalámbrico con desvanecimiento Rayleigh que consiga mejores resultados.

Es recomendable usar una versión de Matlab 2018-B o superior para utilizar sin problemas la aplicación implementada en *App Designer*.

Se recomienda, realizar la simulación en equipos que cuenten con procesadores de gran capacidad, ya que para obtener las diferentes gráficas con trazos bien definidos se requiere de un número elevado de simulaciones (mayor a las 10000)

Para trabajos futuros implementados en *App Designer* se recomienda realizar una aplicación Web.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] M. C. Paredes Paredes and M. J. Fernández-Getino García, "Energy efficient peak power reduction in OFDM with amplitude predistortion aided by orthogonal pilots," IEEE Transaction Consumer Electronics, vol. 59, no. 1, pp. 45-53, 2013.
- [2] M. C. Paredes, "Algoritmos para la reducción de los picos de potencia en los sistemas OFDM", Tesis Doctoral, Escuela Politécnica Superior, Dpto. Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid, Leganés, España, diciembre 2014.
- [3] IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007), pages 1–2793, 2012
- [4] B. S. Krongold and D. L. Jones, "PAR Reduction in OFDM via Active Constellation Extension," IEEE Trans. on Broadcasting, vol. 49, no. 3, pp. 258-268, Sept. 2003.
- [5] M. J. Fernández Getino, O. Edfors, and J. M. Páez-Borrillo, "Peak Power Reduction for OFDM Systems with Orthogonal Pilot Sequences," IEEE Trans, on Wireless Comm., vol. 5, no. 1, pp. 47-51, Jan. 2006.
- [6] J. Wu y I. Stojmenovic, «Ad Hoc Networks,» pp. 29-31, 2004.
- [7] S. u. Rehman, Vehicular Ad-Hoc Networks (VANETs) - An Overview, Sydney, 2013.
- [8] L. Han, «Wireless Ad-hoc Networks,» 2004.
- [9] H. Yomo, C. Huan, P. Kyritsi, T. Duc, S. Chakraborty y R. Prasad, «PHY and MAC Performance Evaluation of IEEE 802.11a WLAN over Fading Channels,» IETE Journal of Research, vol. 51, pp. 1-10, 2015.
- [10] E. Chaparro, "METODOLOGÍA DE DISEÑO DE RED WIFI EN ENTORNOS INDUSTRIALES," Universidad de Sevilla, Abril, 2008. [En línea]. Available: <http://bibing.us.es/proyectos/abreproy/11592/fichero/Memoria%252FANEXO+B.pdf> [Último acceso: 05 noviembre 2019]

- [11] H. Yomo, C. Huan, P. Kyritsi, T. Duc, S. Chakraborty y R. Prasad, «PHY and MAC Performance Evaluation of IEEE 802.11a WLAN over Fading Channels,» IETE Journal of Research, vol. 51, pp. 1-10, 2015.
- [12] E. Herrera, A. Diaz y C. Calafate, «Desarrollando el estándar IEEE 802.11n, un paso adelante en WLAN,» CiComp07, pp. 1-9, 2007.
- [13] IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments. IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009), pages 1–51, 2010. [Ultimo acceso: 05 noviembre 2019]
- [14] Devin Akin and Jim Geier. Certified Wireless Analysis Professional™ Official Study Guide. McGraw-Hill/Osborne, Estados Unidos, 2004.
- [15] M. Gast, 802.11 Wireless Networks, O'Reilly, 2005.
- [16] IEEE, IEEE Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks-- Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE, 2016.
- [17] L.A. Quilumba Toapanta, “Simulación de la técnica Constellation Extension de reducción de la PAPR de los sistemas OFDM,» Escuela Politécnica Nacional, 2017.
- [18] Shu Lin and Daniel J Costello. Error Control Coding. 01 2004.
- [19] Y. Q. Shi, Xi Min Zhang, Zhi-Cheng Ni, and N. Ansari. Interleaving for combating bursts of errors. IEEE Circuits and Systems Magazine, 4(1):29–42, 2004.
- [20] F. Valles, "Estudio y Diseño de la Amplificación del Backbone para la Metro Ethernet de Ecuonline S.A. mediante radioenlaces en la banda 5,8GHz con Modulación OFDM, para el trayecto Quito-Cayambe-Otavalo-Ibarra," Escuela Politécnica Nacional, 2009.

- [21] C. Vargas, W. López y C. Rocha, "Sistemas de Comunicación Inalámbrica MIMO-OFDM," Grupo de Investigación en Comunicaciones del Departamento de Ingeniería Eléctrica de la Universidad Federal de Santa Catarina, vol. 3, pp. 763768, 2007.
- [22] S. Weinstein y P. Ebert, "Data Transmission by Frequency-Division Multiplexing Using the Discrete Fourier Transform," IEEE Transaction on Communication Technology, vol. 19, nº 5, pp. 628-634, 1971.
- [23] D. Paredes, "Análisis y Simulación de una Sistema de Transmisión Digital Utilizando MATLAB con Simulink," Escuela Politécnica Nacional, 1999.
- [24] R. W. Chang, "Orthogonal Frequency Multiplex Data Transmission System". US Patente 3, 1970.
- [25] A. Peled y A. Ruiz, "Frequency Domain Data Transmission Using Reduced Computational Complexity Algorithms," IEEE International Conference on Acoustics, vol. 5, pp. 964-967, 1980.
- [26] P. Calderón, "Análisis del Desempeño de la Técnica OFDM sobre Canales Dispersos," Escuela Politécnica del Ejército, 2007.
- [27] M. Paredes y M. J. Fernández Getino, "The Problem of Peak-to-Average Power Ratio in OFDM Systems," Revista Digital Científico Tecnológica, vol. 1, nº 2, p. 6, 2012.
- [28] J. Peñafiel, "Análisis de le Tecnología Li-Fi: Comunicaciones por Luz Visible como Punto de Acceso a Internet, una Alternativa a la Transmisión de Datos en las Comunicaciones Inalámbricas," Universidad Politécnica Salesiana, Febrero, 2015.
- [29] K. M. Gharaibeh, "Nonlinear distortion in wireless systems modeling and simulation with MATLAB". Chichester, United Kingdom: IEEE Press, John Wiley & Sons, Ltd., 2012.
- [30] F. Sarzosa, "ANÁLISIS DE LOS EFECTOS OCASIONADOS POR UN HPA EN UNA TRANSMISIÓN OFDM CON PICOS DE POTENCIA ELEVADA MEDIANTE SIMULACIONES EN MATLAB", Tesis de ingeniería, Quito, Julio, 2017.

- [31] R. Gilmore and L. Besser, "Practical RF Circuit Design for Modern Wireless Systems", Volume II: Active Circuits and Systems. Norwood, USA: Artech House, INC., 2003.
- [32] G. G. Gamez, "Measurements for modelling of wideband nonlinear power amplifiers for wireless communications", Master's Thesis, Helsinki University of Technology, Espoo, Finland, September 2004.
- [33] P. Jantunen, "Modelling of Nonlinear Power Amplifiers for Wireless Communications", Master's Thesis, Helsinki University of Technology, Espoo, Finland, March 2004.
- [34] M. C. Paredes, D. Mejía, D. F. Puga, "Analysis of quantization metrics for PAPR evaluation in OFDM systems", Escuela Politécnica Nacional, Noviembre, 2016.
- [35] T. Jiang y Y. Wu, "An Overview: Peak-to-Average Power Ratio Reduction Techniques for OFDM Signals," IEEE Transactions on Broadcasting, vol. 54, nº 2, pp. 257-267, Junio, 2008.
- [36] H. Ochiai and H. Imai, "On the Distribution of the Peak-to-Average Power Ratio in OFDM signal," IEEE Trans. on Comunications, vol. 49, no.23, pp. 282-289, Feb. 2001.
- [37] S. Sezginer, and H. Sari, "Metric-Based Symbol Predistortion Techniques for Peak Power Reduction in OFDM Systems," IEEE Trans. on Wireless Comm., vol. 6, no. 7, pp. 2622-2629, Jul. 2007.
- [38] Y. Cho, J. Kim, W. Yang, Ch. Kang, MIMO-OFDM Wireless Communications with MATLAB, 5ta ed., Asia: Jhon Wiley & Sons, 2010.
- [39] V. S. Bagad, "Parameters Affecting Microwave Link" de Wireless Communication, 1ra ed., India, Technical Publications Pune, 2009.
- [40] W. Stallings, Data and Computer Communications, 10ma ed., New Jersey: Prentice Hall, 2007.
- [41] M. Thompson Ph.D., Intuitive Analog Circuit Design, 2da ed., Massachusetts: Elsevier, 2014.

- [42] Mathuranathan, «GaussianWaves,» 9 de julio 2014. [En línea]. Available: <https://www.gaussianwaves.com/2014/07/power-delay-profile/>. [Último acceso: 10 diciembre 2019].
- [43] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle, USA: Prentice Hall, 2001.
- [44] D. Rodríguez, “Creación de aplicaciones en Matlab con App Designer,” 5 Dic 2018. [En línea]. Available: <https://www.analyticslane.com/2018/12/05/creacion-de-aplicaciones-en-matlab-con-app-designer/>. [Último acceso: 15 enero 2019].
- [45] D. Rodríguez, “Migración de GUIDE a App Designer,” 5 Jun 2019. [En línea]. Available: <https://www.analyticslane.com/2019/06/05/migracion-de-guide-a-app-designer/> [Último acceso: 15 enero 2019].
- [46] MATLAB, “MathWorks,” 14 Oct 2018. [En línea]. Available: https://la.mathworks.com/help/matlab/creating_guis/differences-between-app-designer-and-guide.html [Último acceso: 15 enero 2019].
- [47] MATLAB, “MathWorks,” 18 Dic 2018. [En línea]. Available: <https://la.mathworks.com/products/matlab/app-designer.html> [Último acceso: 31 de enero 2019].
- [48] Kan, Malik, “IEEE 802.11 Regulatory SC,” *DSRC Tiger Team – IEEE Mentor*, 2014.
- [49] T.S. Rappaport, *Wireless Communications, Principles and Practice*. Upper Saddle River, N.J.: Prentice Hall, 1999.

ANEXOS

ANEXO A. Diagramas de flujo

ANEXO B. Códigos y Funciones







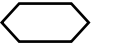
ANEXO C. Manual de Usuario

ANEXO A Diagramas de Flujo

En este anexo se adjunta los diagramas de flujo para cada una de las funciones y procesos descritos en el capítulo 2.

Todos los diagramas de flujo están diseñados tomando en cuenta la nomenclatura en la Tabla A.1.

Tabla A.1: Simbología de los diagramas de flujo

| Símbolo | Descripción |
|--|------------------------|
|  | Inicio del diagrama |
|  | Final del diagrama |
|  | Entrada de Datos |
|  | Salida de Datos |
|  | Procesamiento de Datos |
|  | Decisor de Datos |
|  | Contador de Datos |

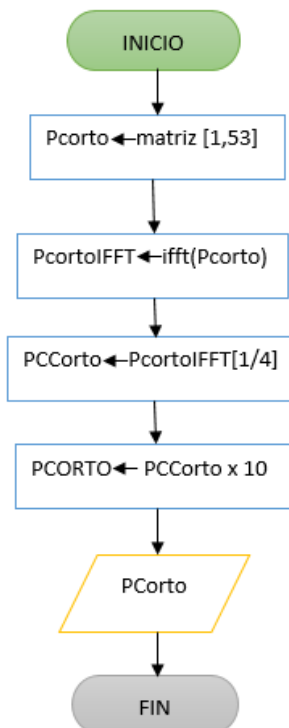


Figura A.1: Diagrama de flujo de la función CrearPreambuloCorto() sección 2.4.2.1.

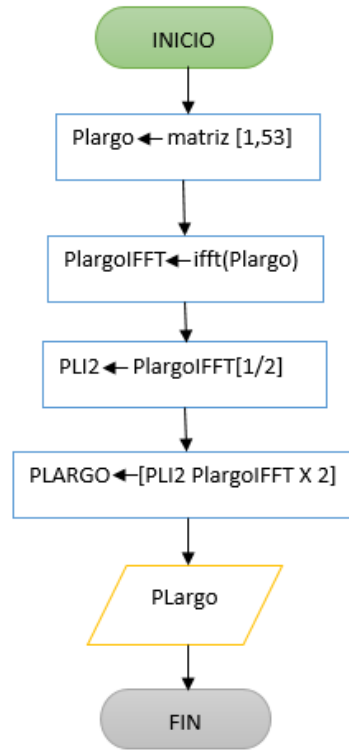


Figura A.2: Diagrama de flujo de la función CrearPreambuloLargo() sección 2.4.2.2.

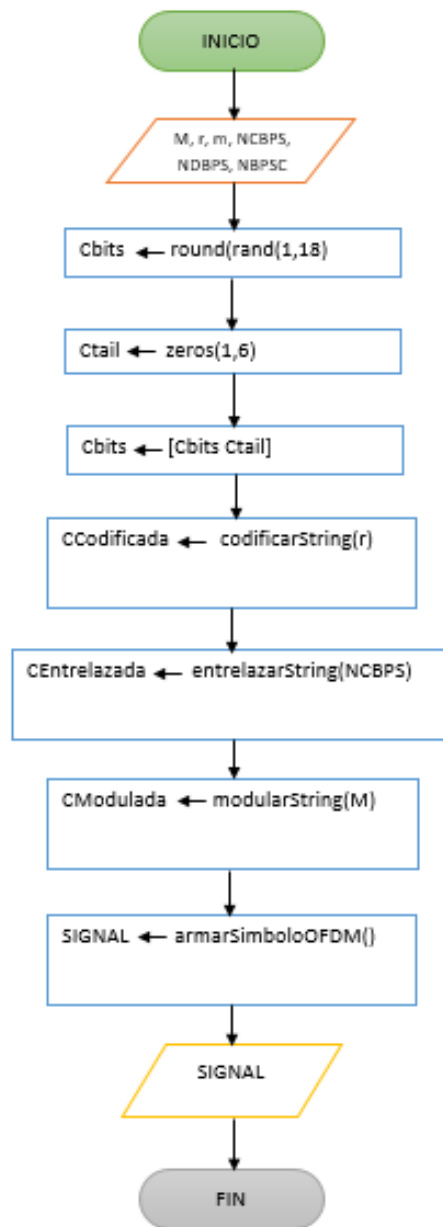


Figura A.3: Diagrama de flujo del campo SIGNAL sección 2.4.3.

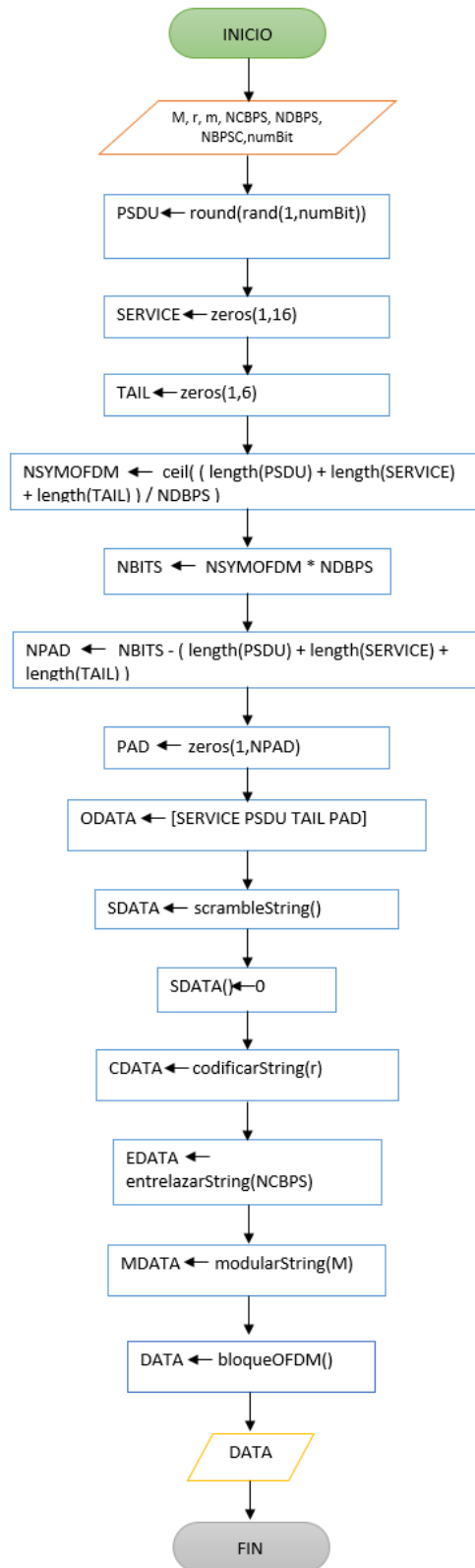


Figura A.4: Diagrama de flujo del campo DATA sección 2.4.4.

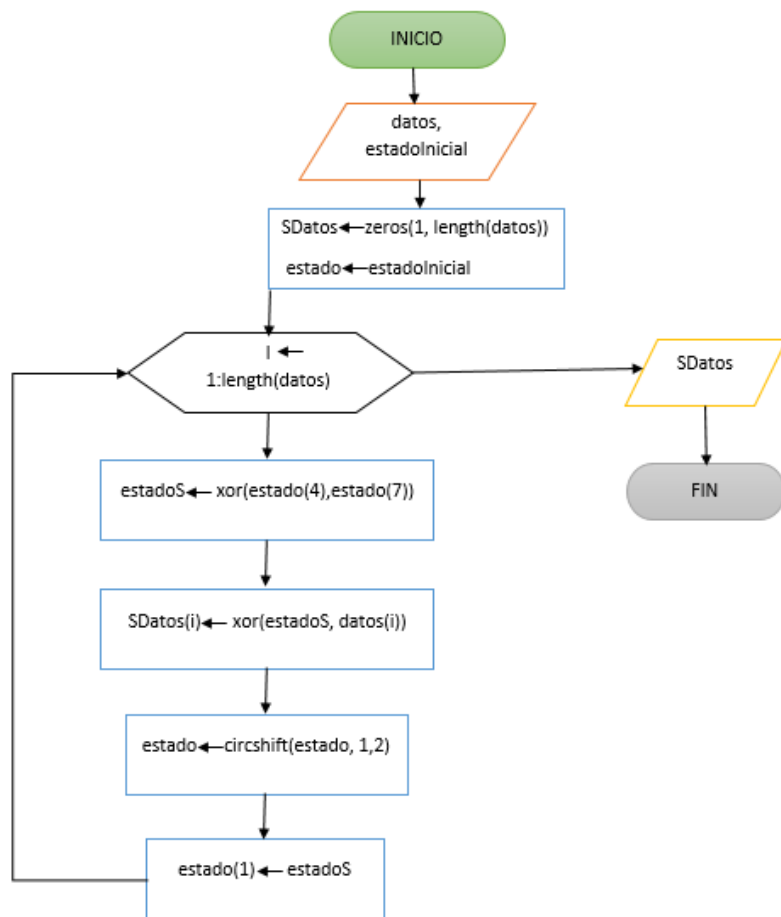


Figura A.5: Diagrama de flujo de la función scrambleString() sección 2.4.5.

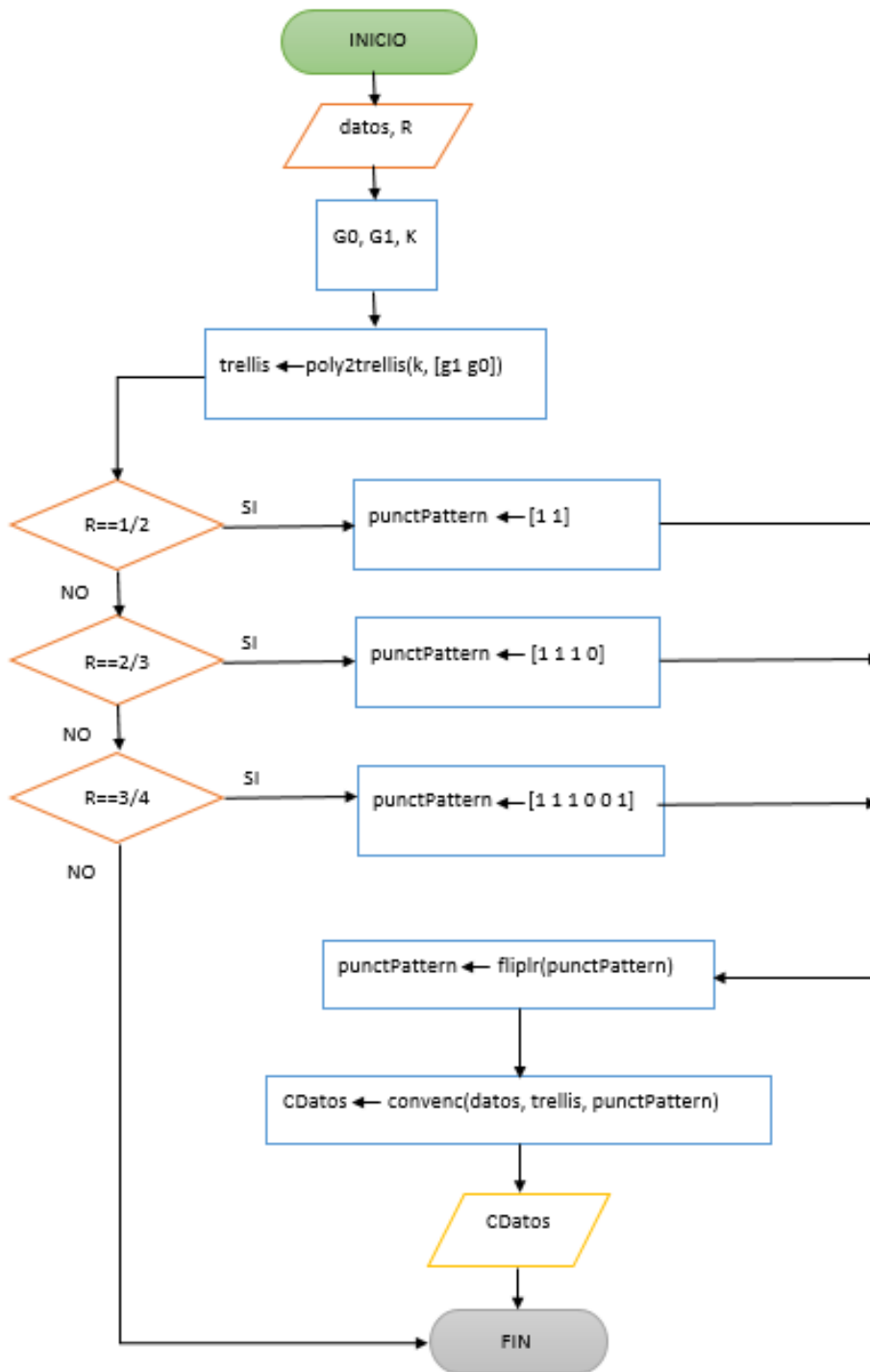


Figura A.6: Diagrama de flujo de la función codificarString() sección 2.4.6.

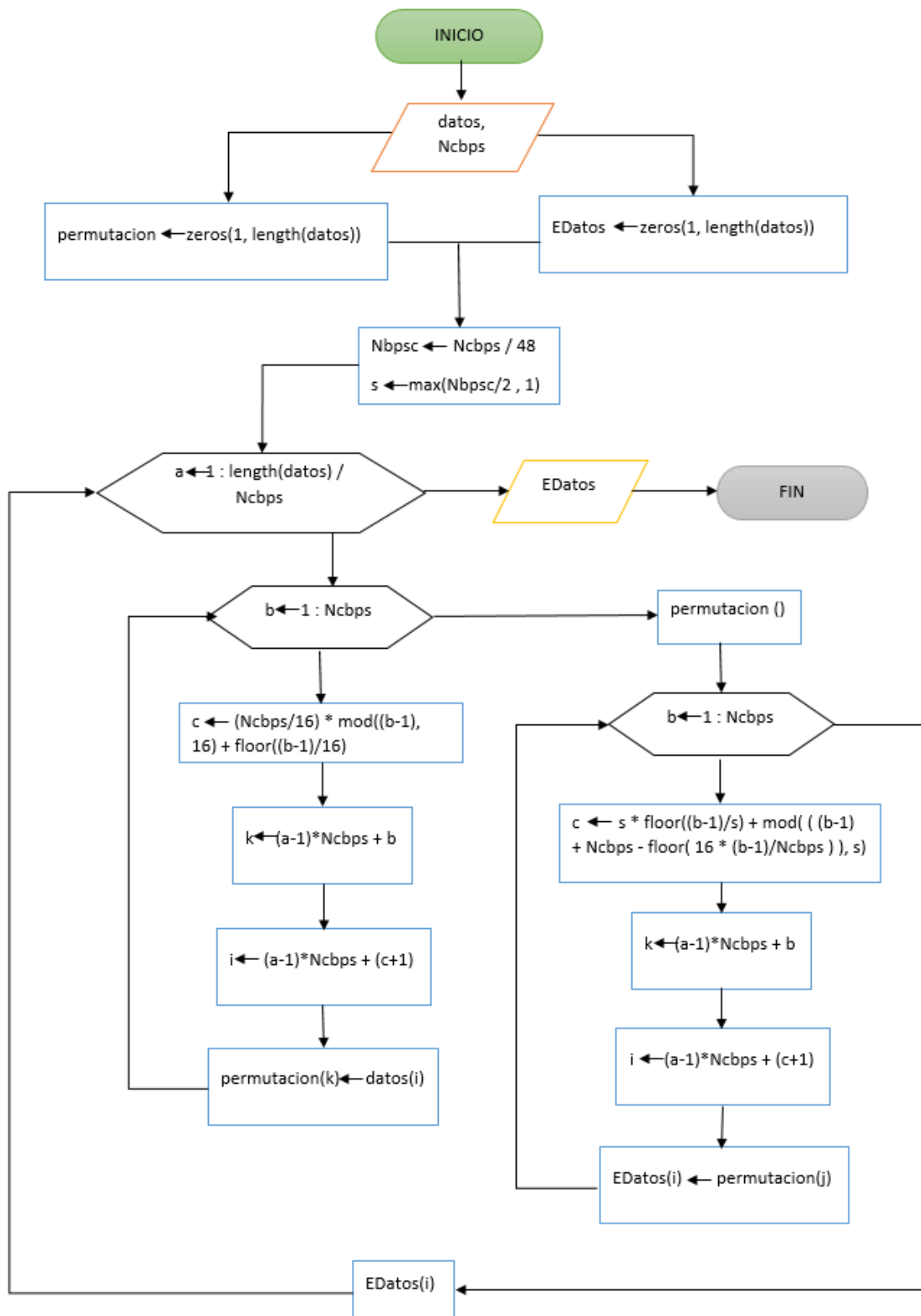


Figura A.7: Diagrama de flujo de la función entrelazarString() sección 2.4.7.

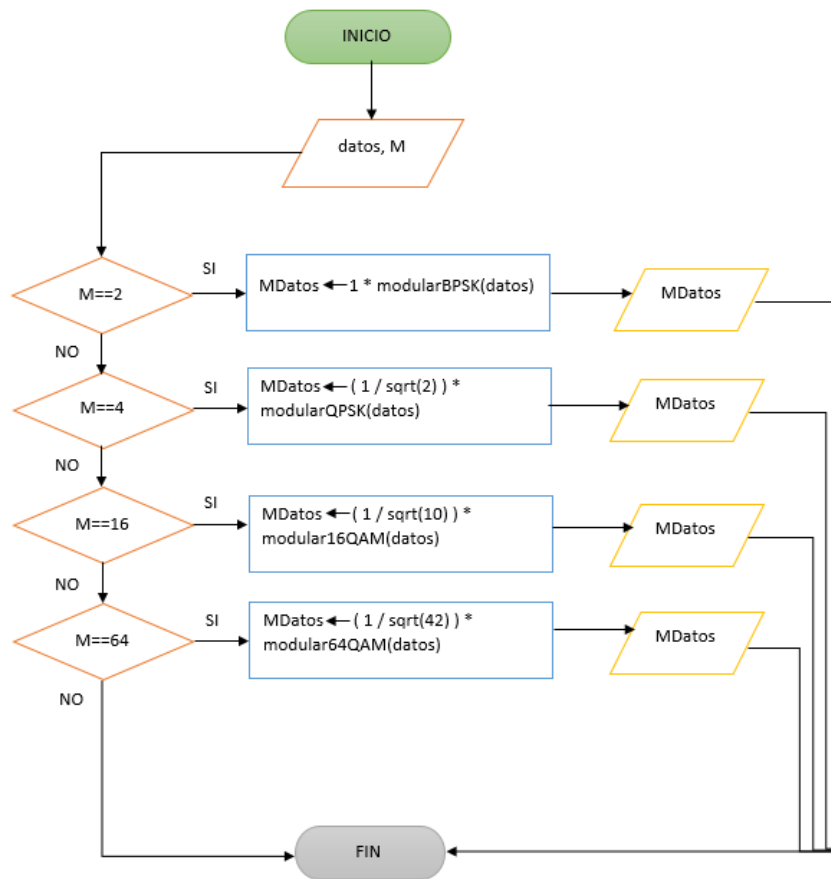


Figura A.8: Diagrama de flujo de la función modularString() sección 2.4.8.

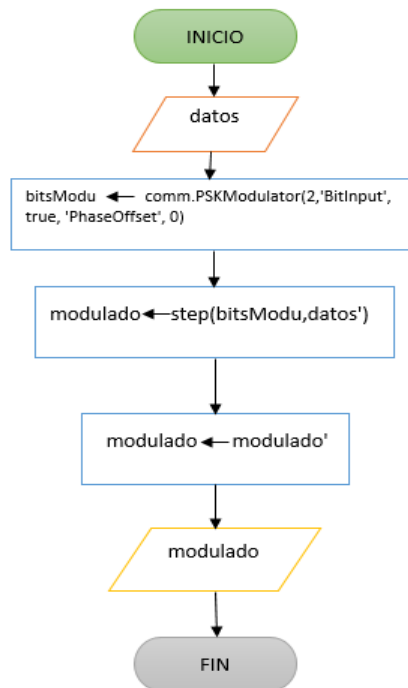


Figura A.9: Diagrama de flujo de la función modulacionBPSK() sección 2.4.8 a.

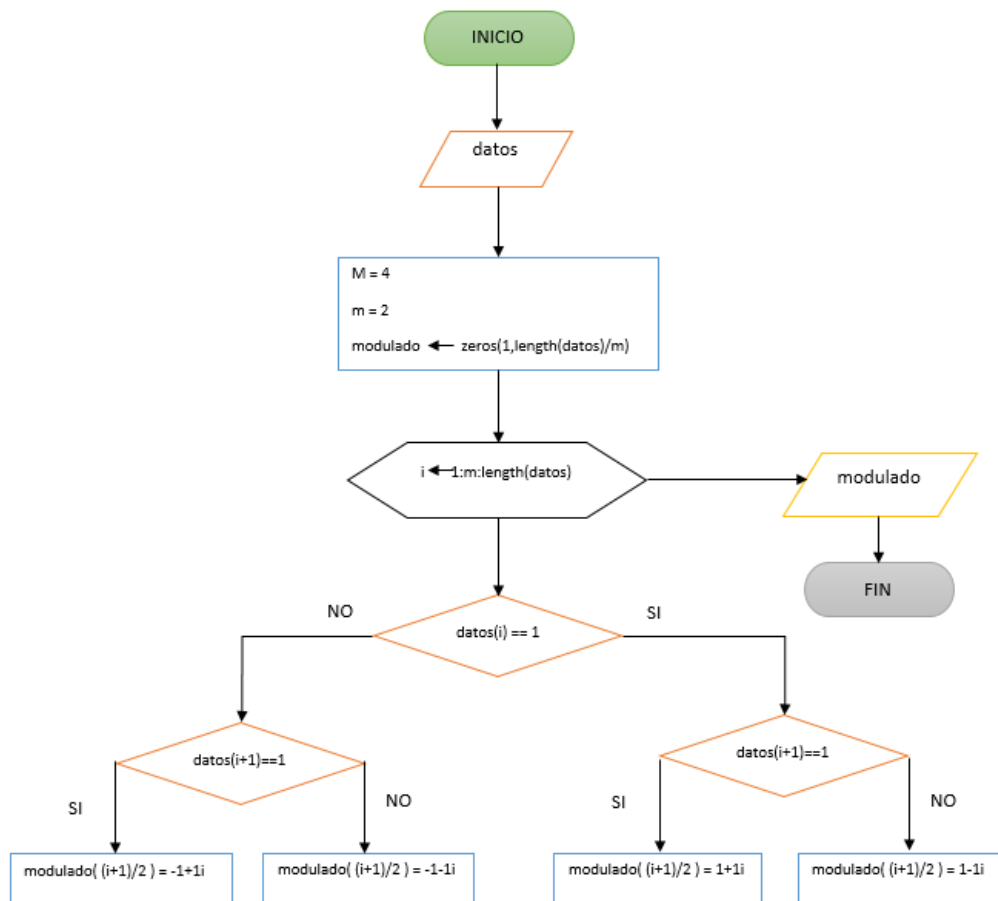


Figura A.10: Diagrama de flujo de la función modulaciónQPSK() sección 2.4.8 b.

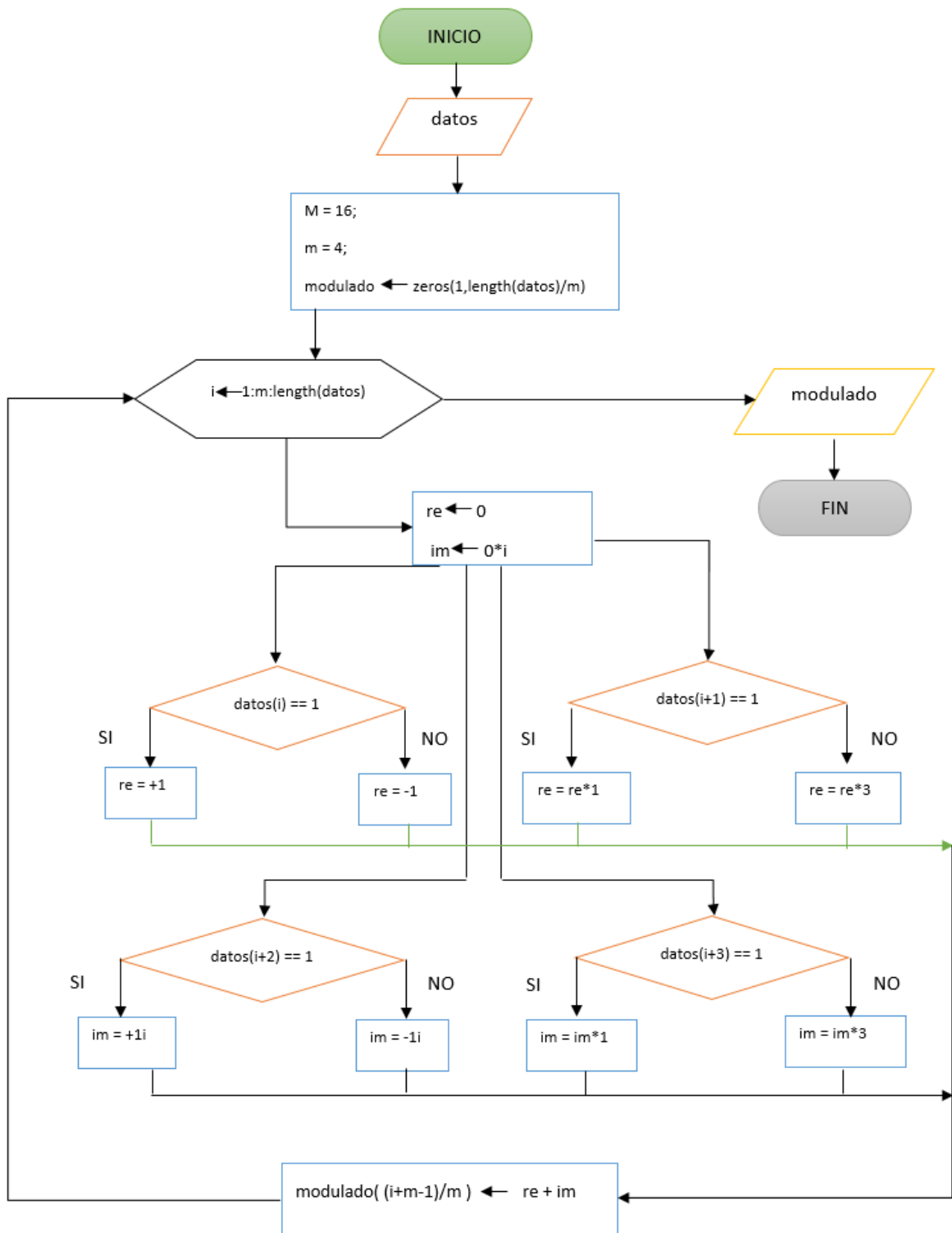


Figura A.11: Diagrama de flujo de la función modulación16QAM() sección 2.4.8 c.

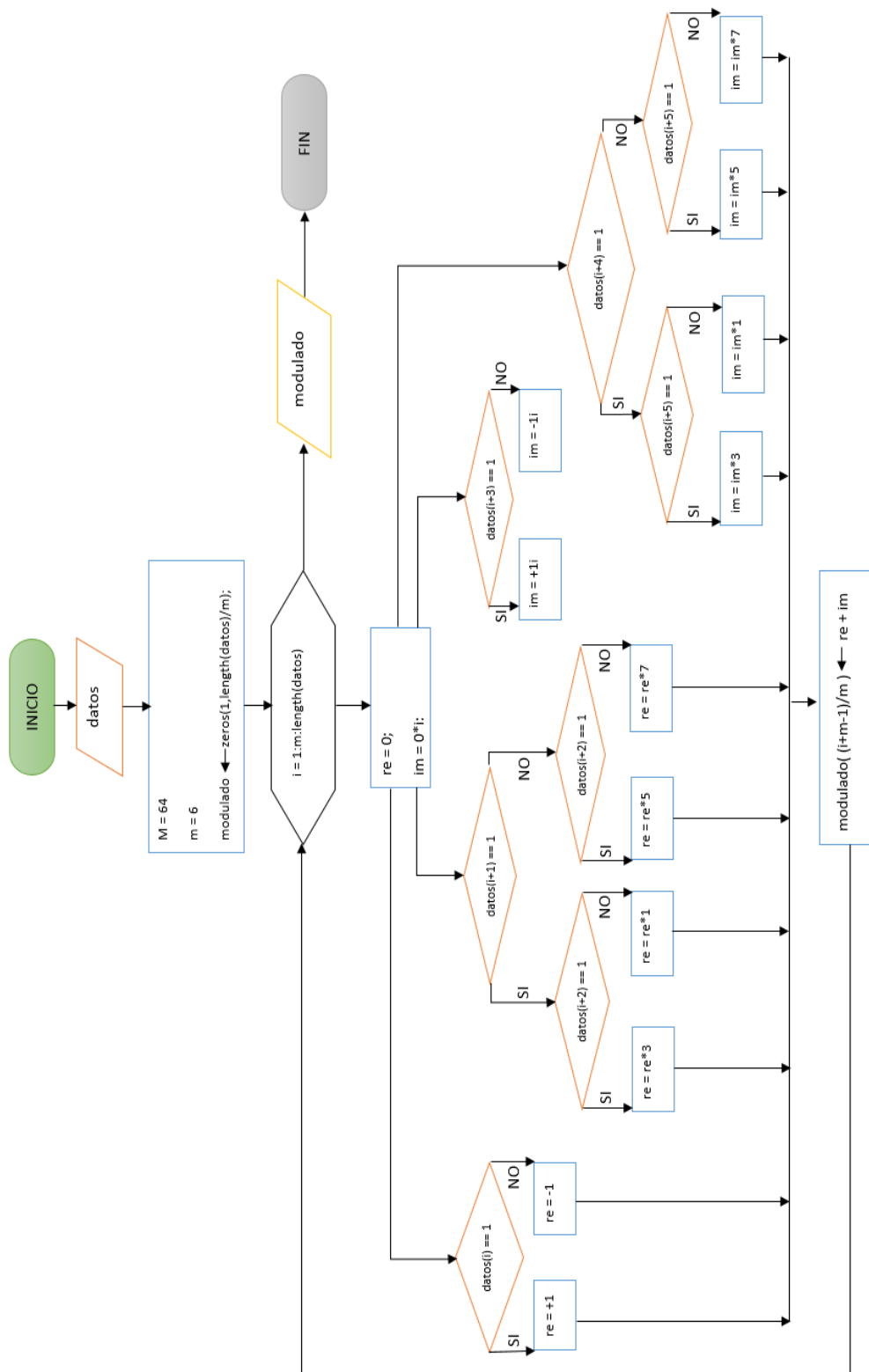


Figura A.12: Diagrama de flujo de la función modulación64QAM() sección 2.4.8 d.

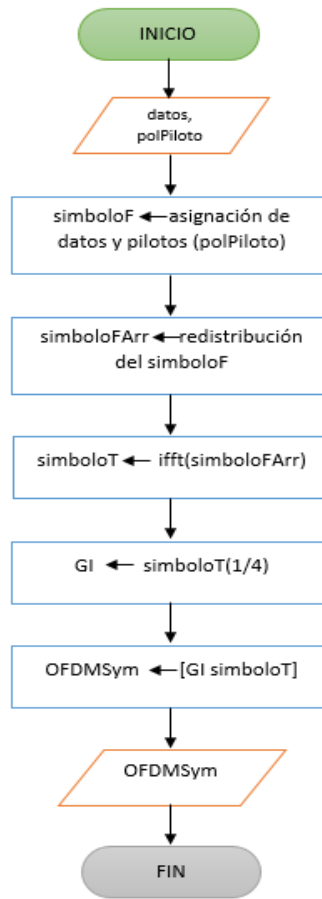


Figura A.14: Diagrama de flujo de la función Armar Símbolo OFDM sección 2.4.9 b.

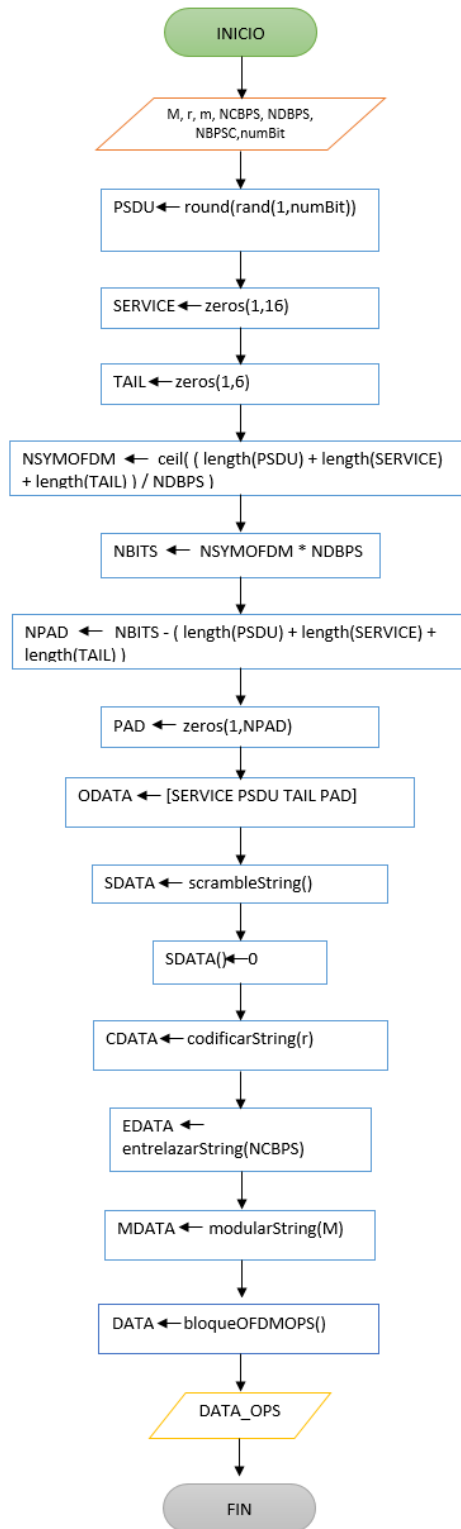


Figura A.15: Diagrama de flujo de la creación del campo DATA_OPS sección 2.4.9.

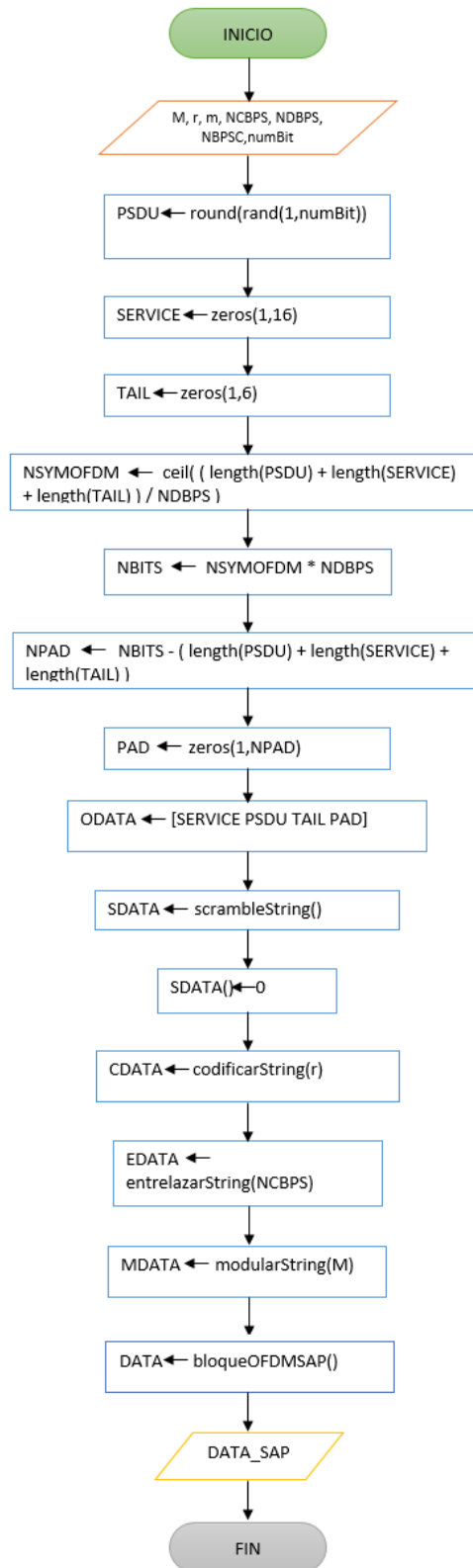


Figura A.16: Diagrama de flujo de la creación del campo DATA_SAP sección 2.4.9.

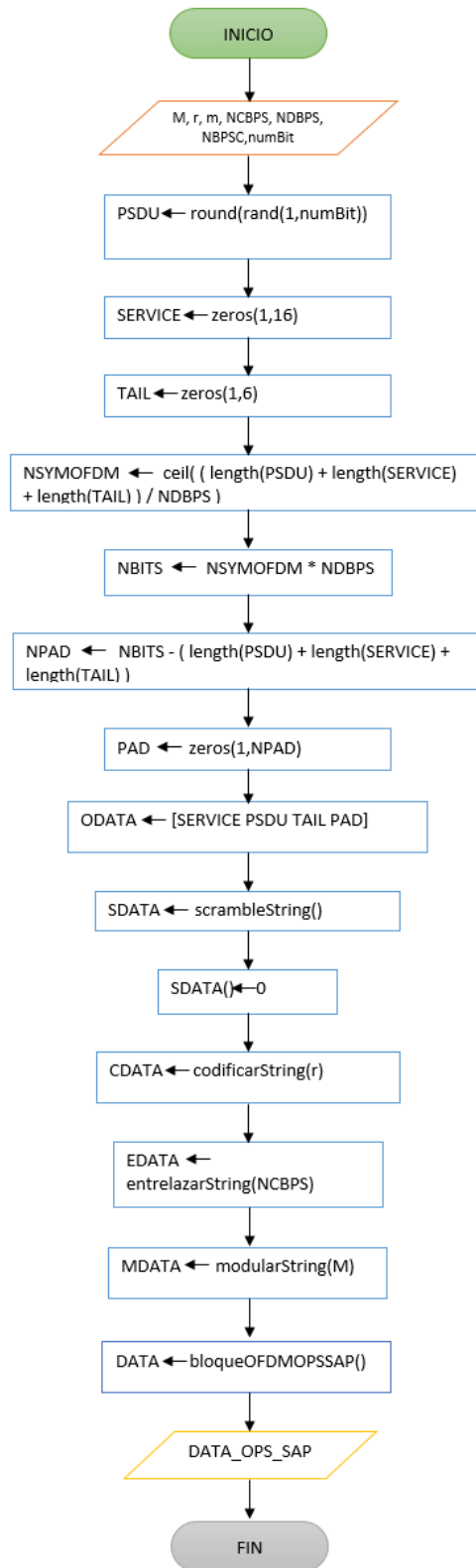


Figura A.17 Diagrama de flujo de la función DATA con técnica OPS-SAP sección 2.4.9.

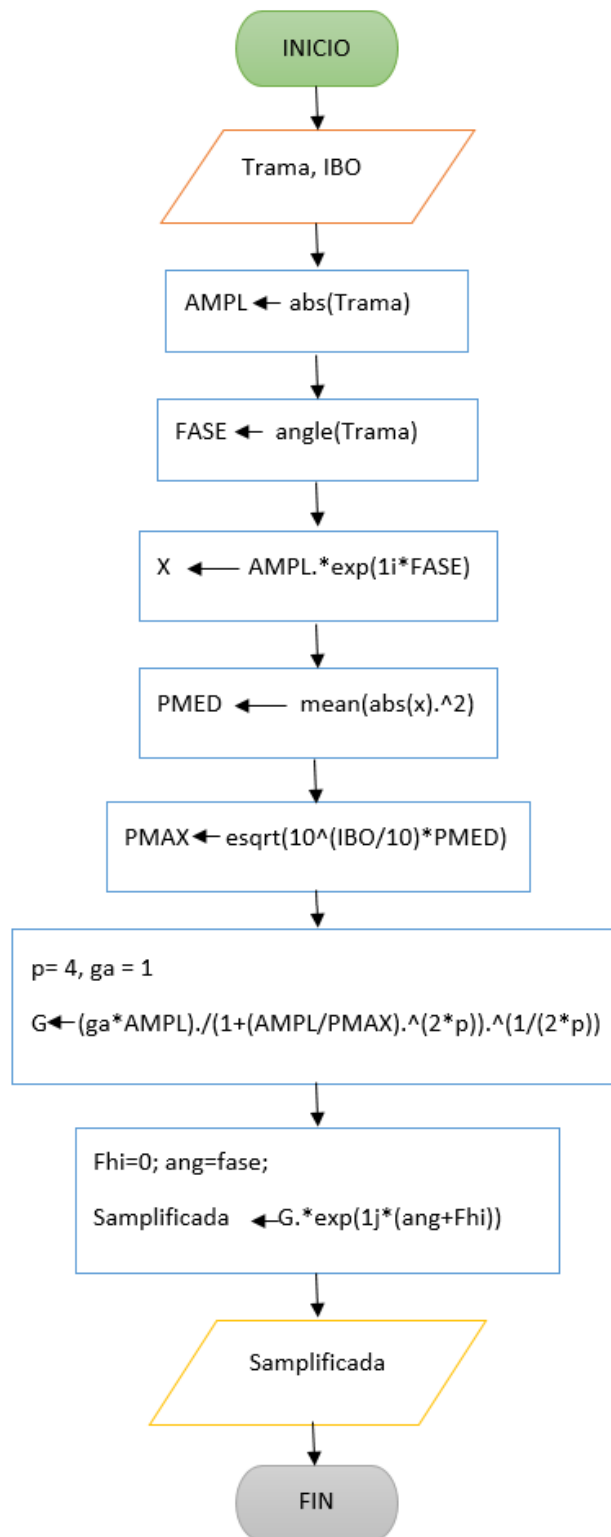


Figura A.18: Diagrama de flujo de la función `amplificadorHPA()` sección 2.4.10.

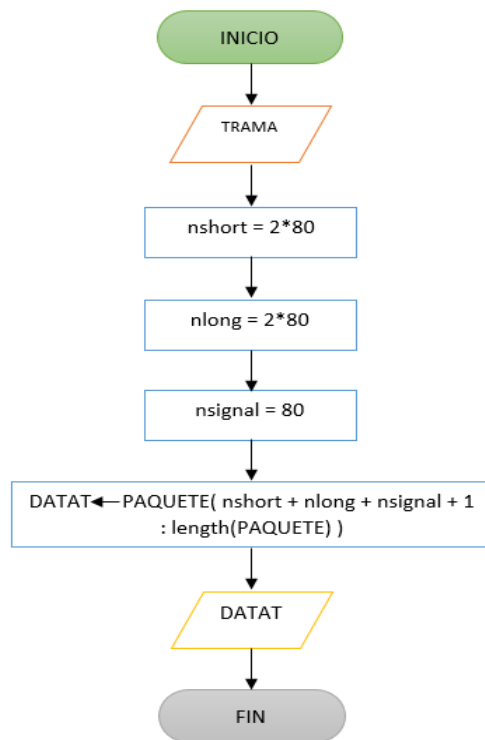


Figura A.19: Diagrama de flujo de la función extraerData() sección 2.6.2 a.

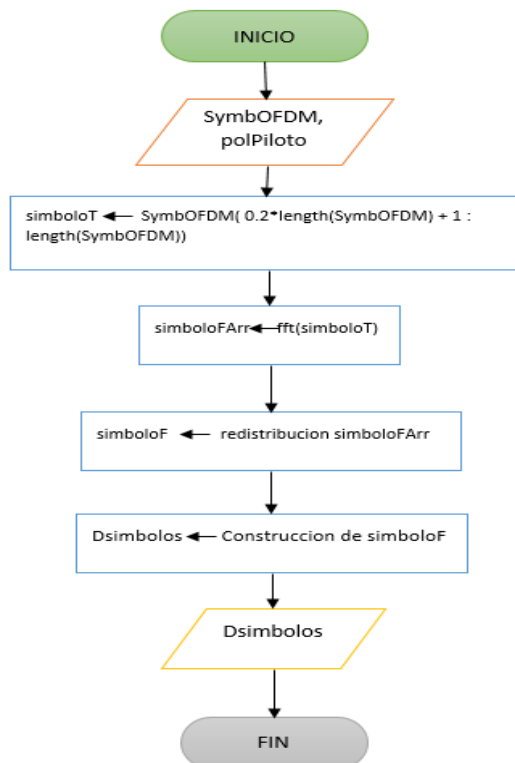


Figura A.20 Diagrama de flujo de la función extraerSimboloOFDM() sección 2.6.2 b.

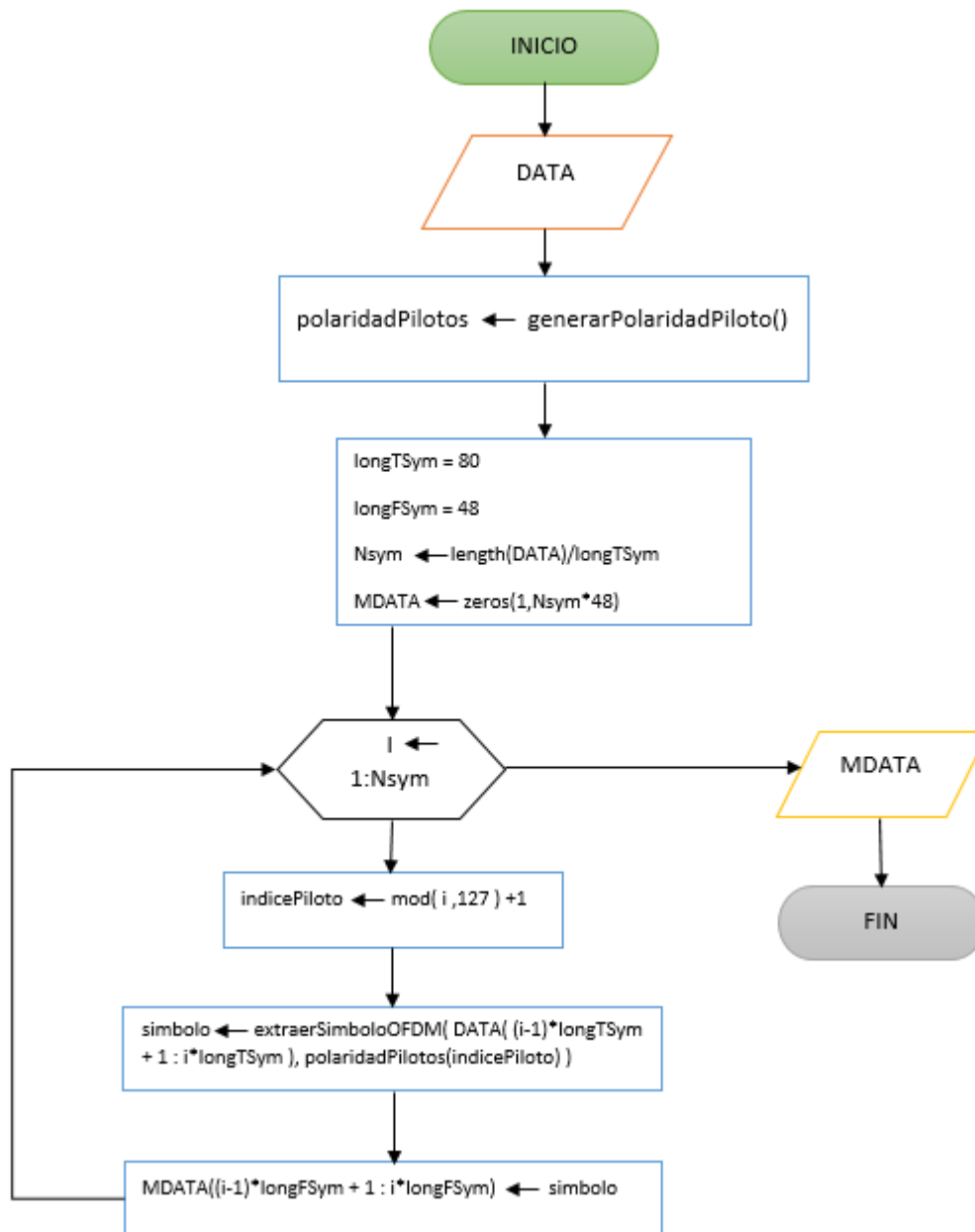


Figura A.21: Diagrama de flujo de la función bloqueOFDMrecepcion() sección 2.6.1.

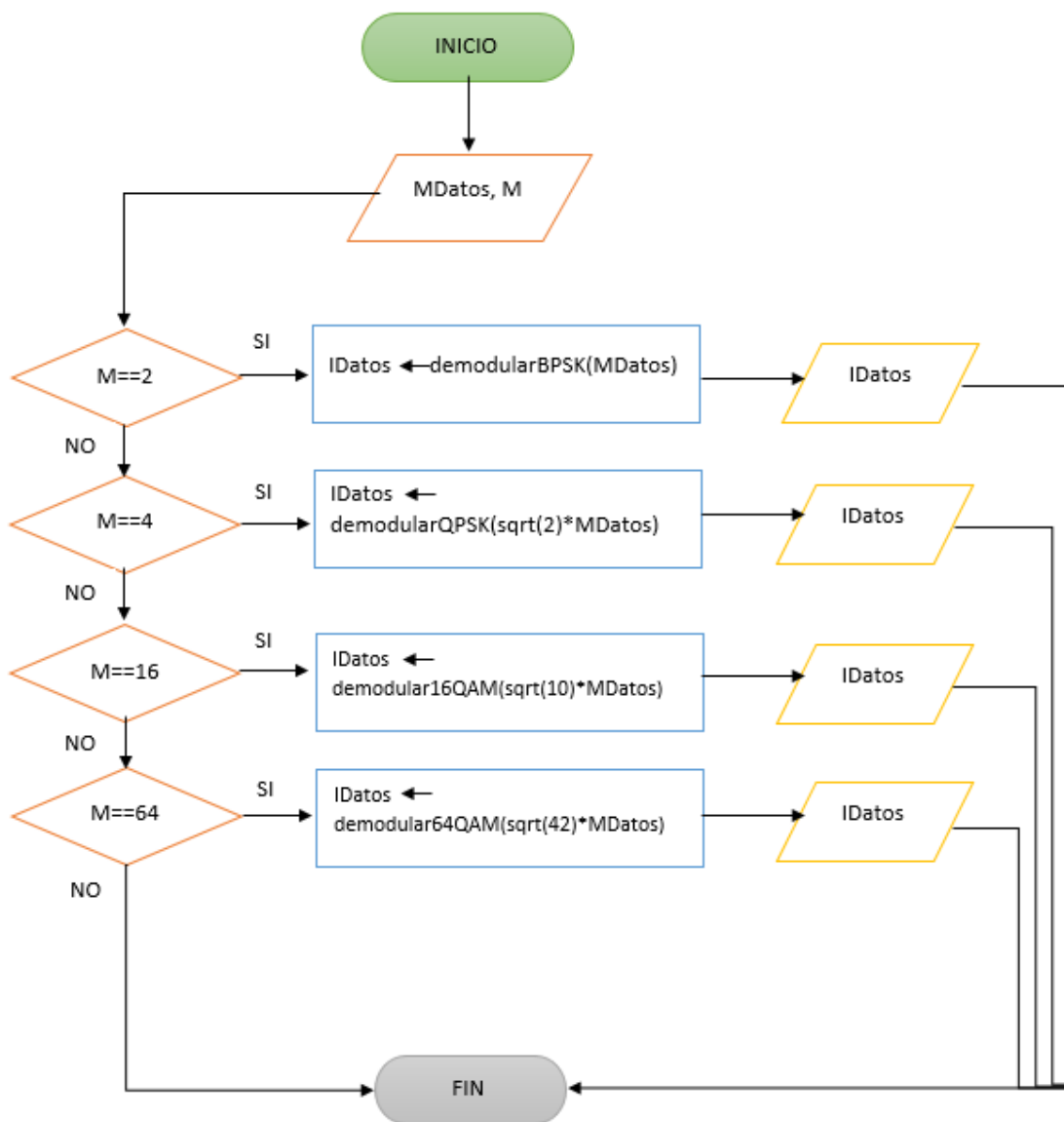


Figura A.22: Diagrama de flujo de la función demodularString() sección 2.6.3.

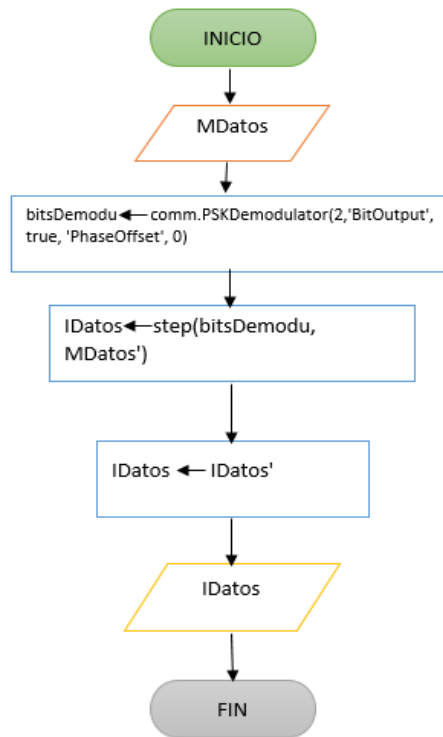


Figura A.23: Diagrama de flujo de la función demodulacionBPSK() sección 2.6.3 a.

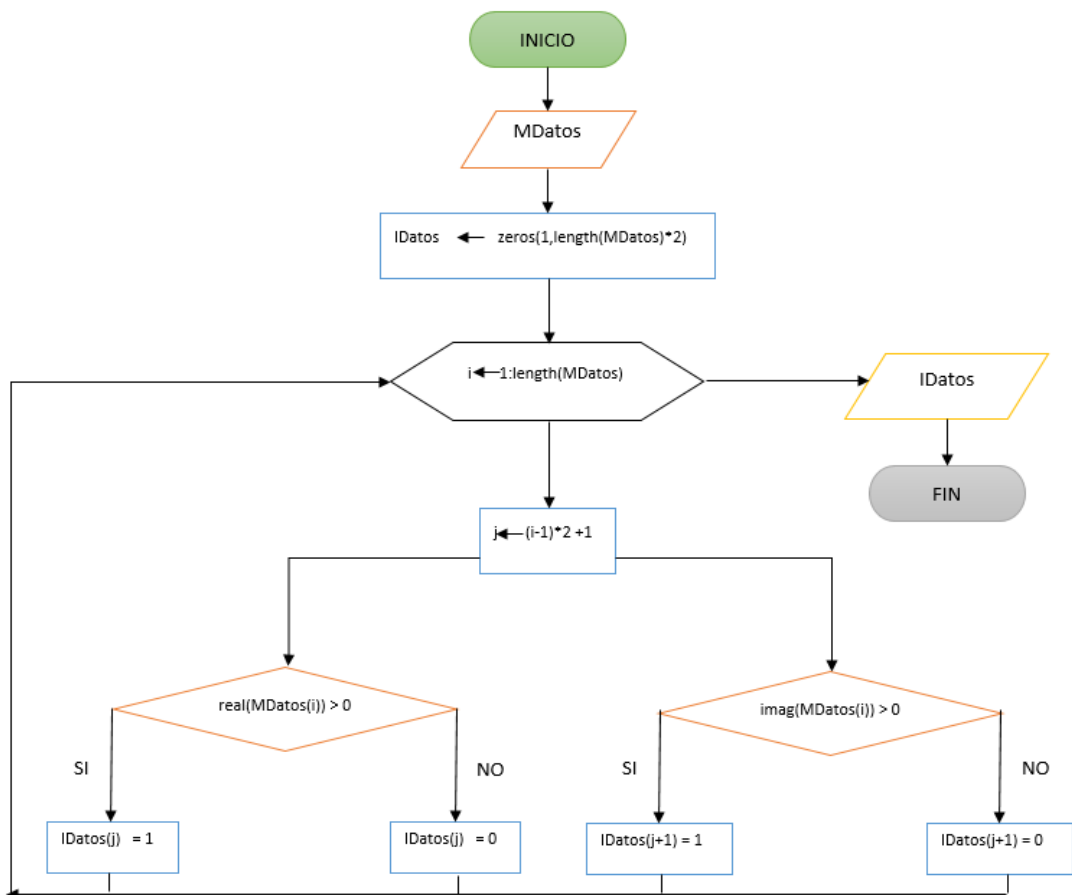


Figura A.24: Diagrama de flujo de la función demodulacionQPSK() sección 2.6.3 b.

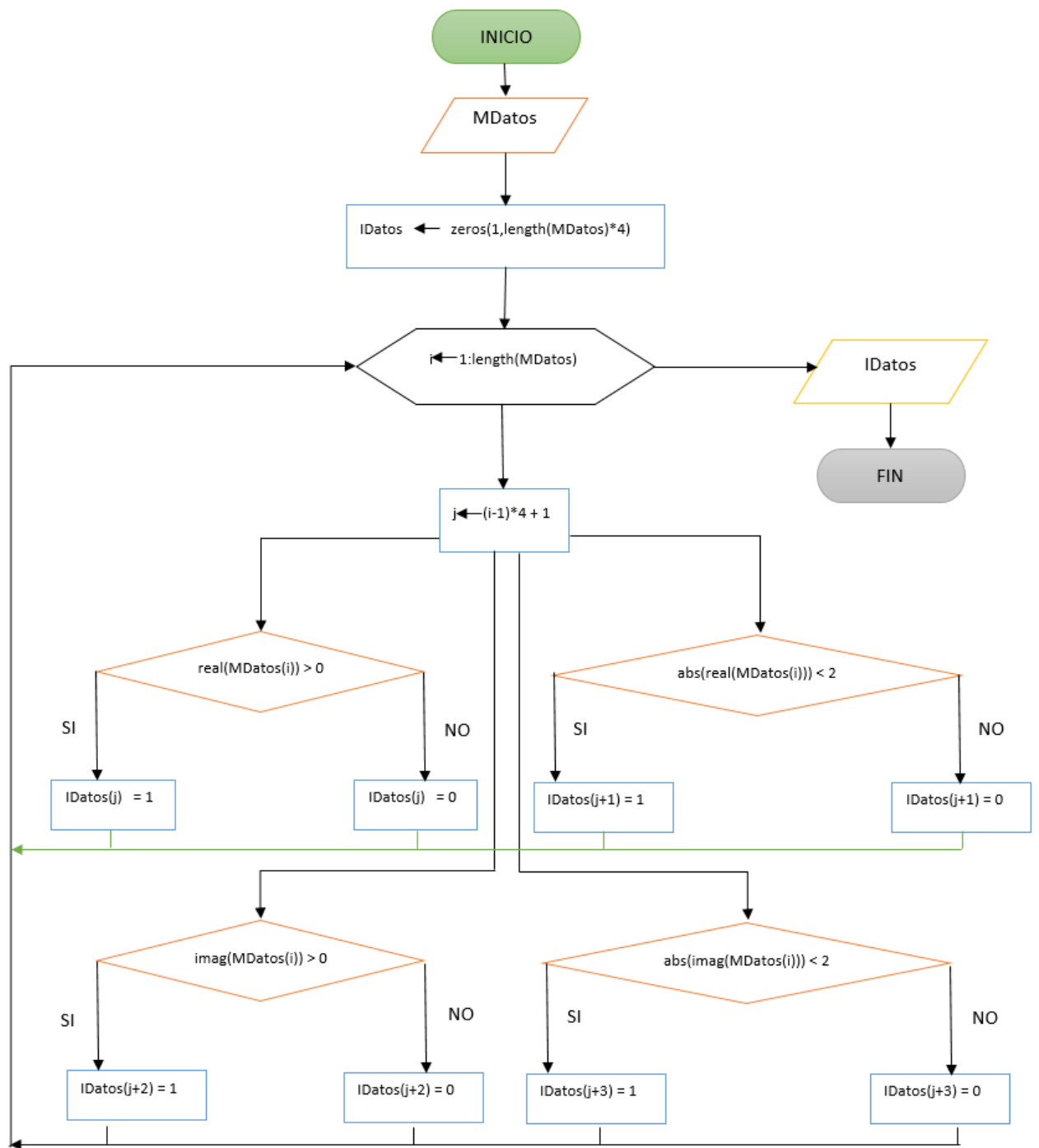


Figura A.25: Diagrama de flujo de la función demodulacion16QAM() sección 2.6.3 c.

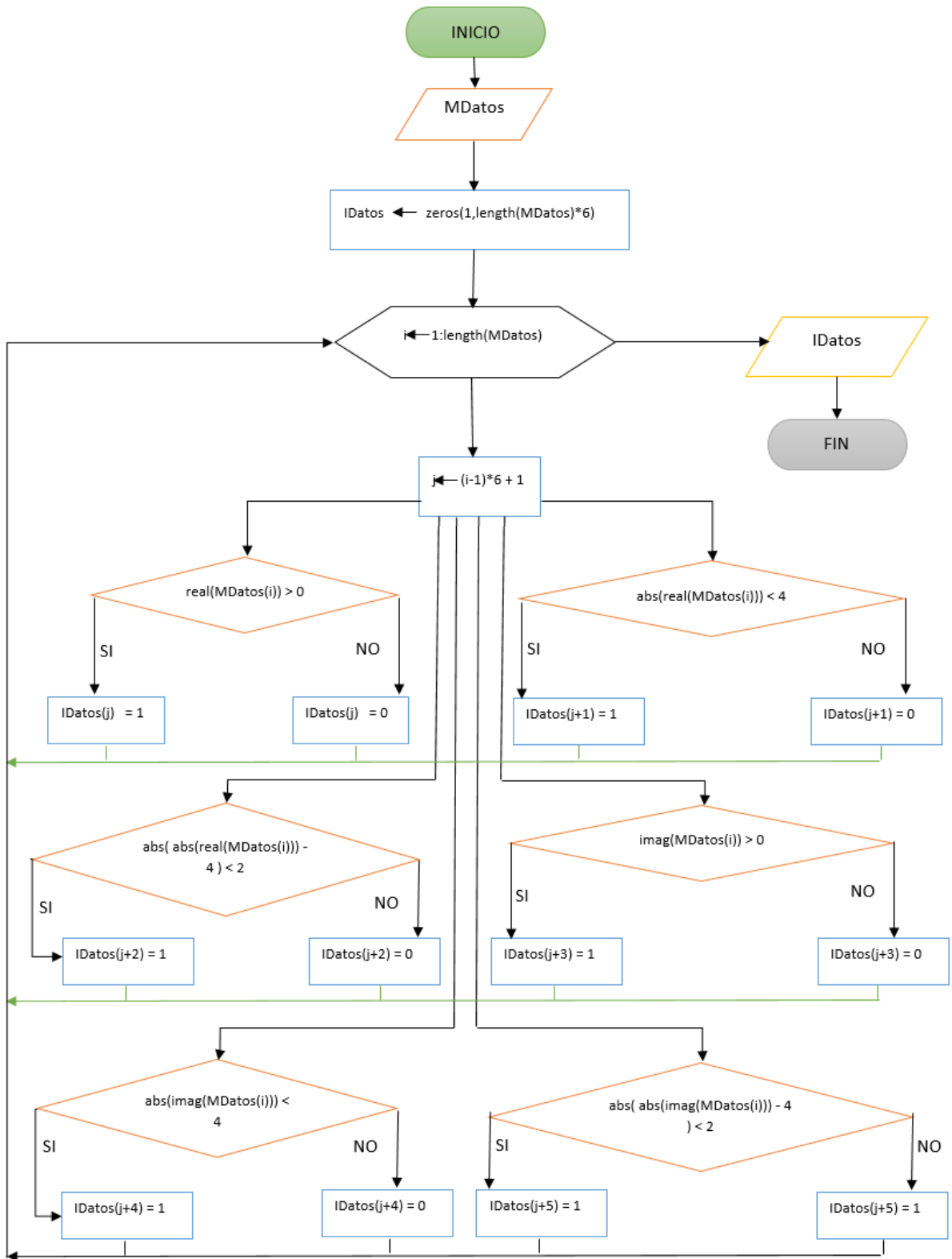


Figura A.26: Diagrama de flujo de la función demodulacion64QAM() sección 2.6.3 d.

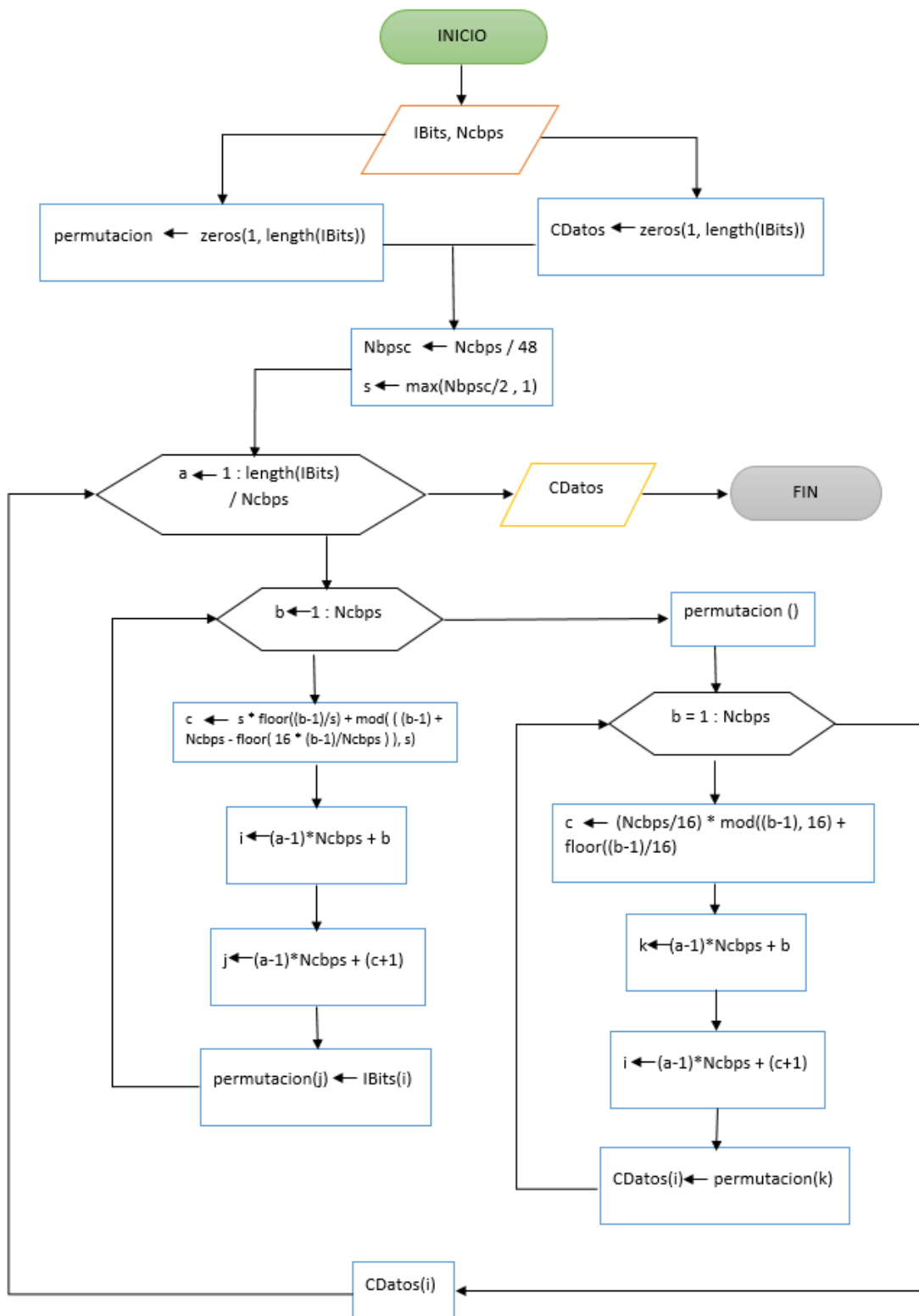


Figura A.27: Diagrama de flujo de la función deentrelazarString() sección 2.6.4.

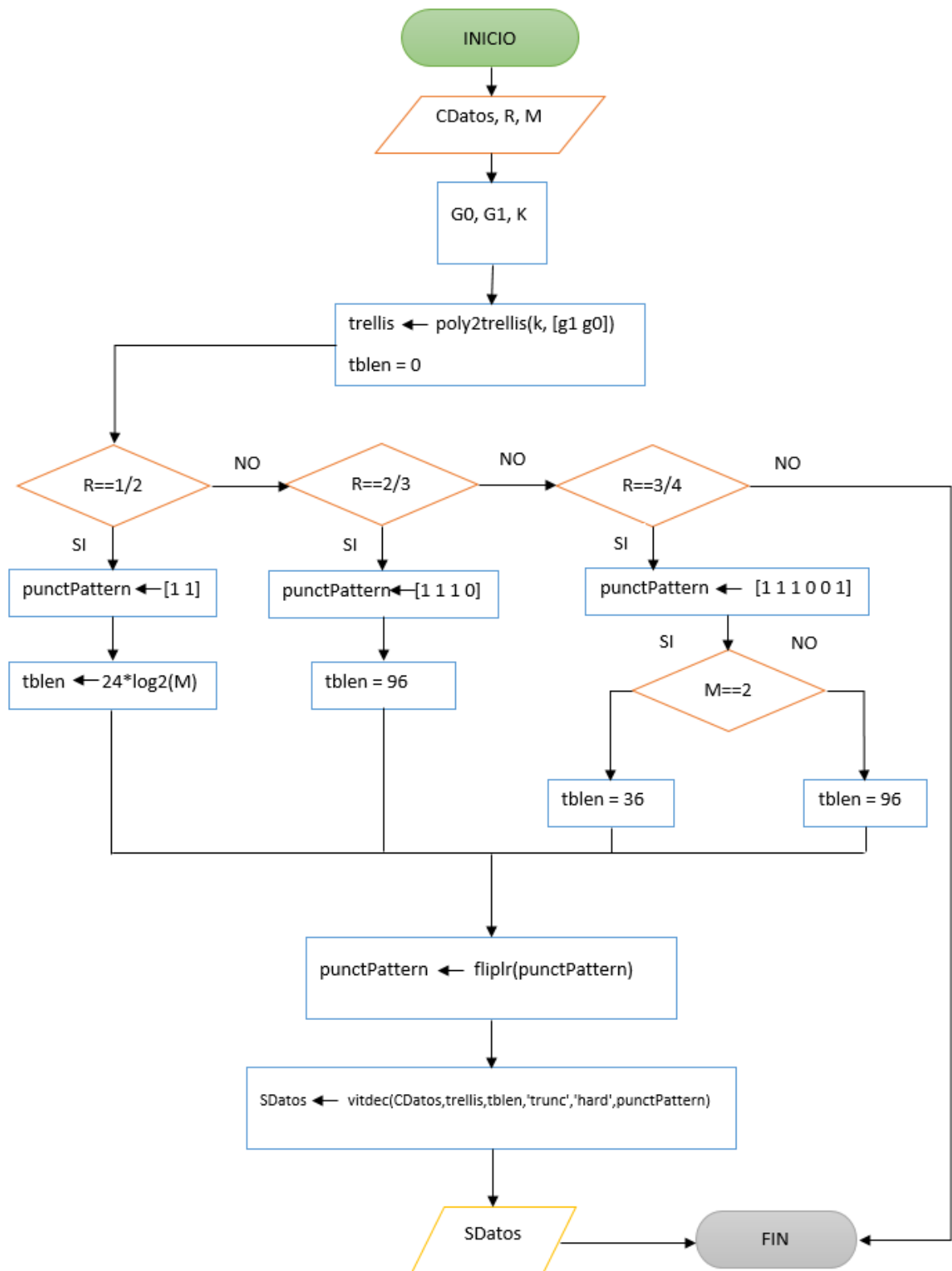


Figura A.28: Diagrama de flujo de la función decodificarString() sección 2.6.5.

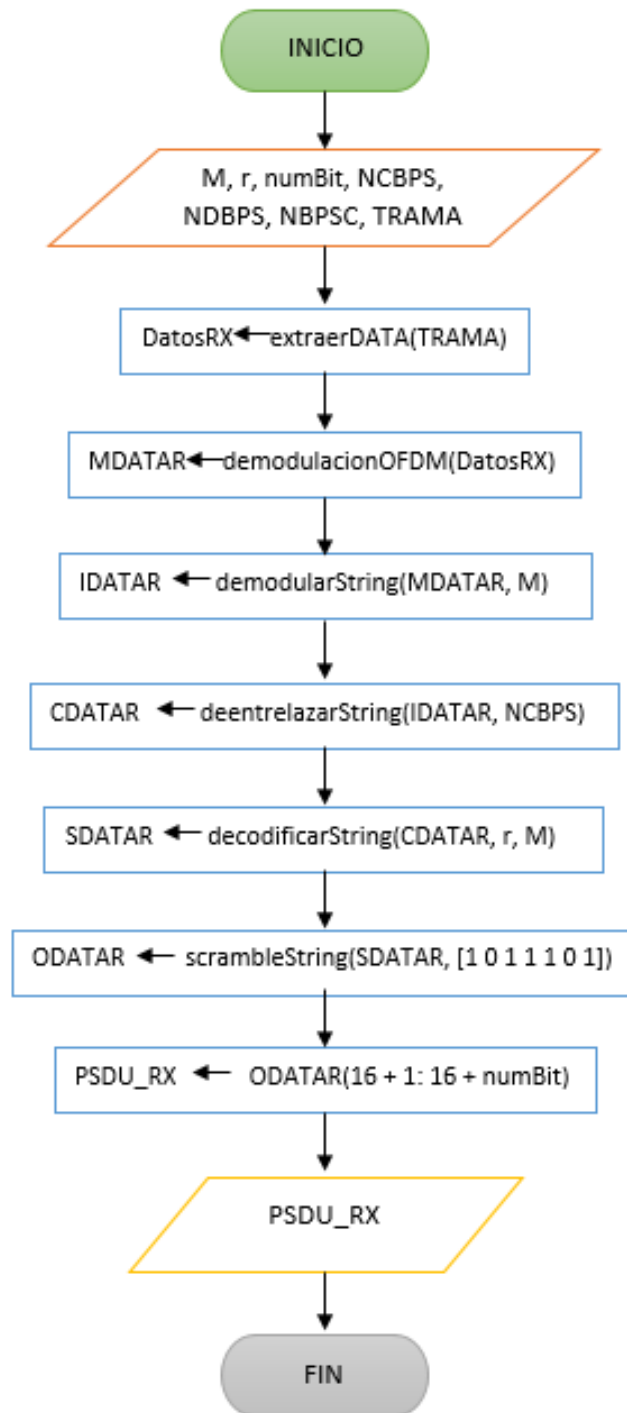


Figura A.29: Diagrama de flujo de la función Receptor OFDM “PSDU_RX” sección 2.6.6.

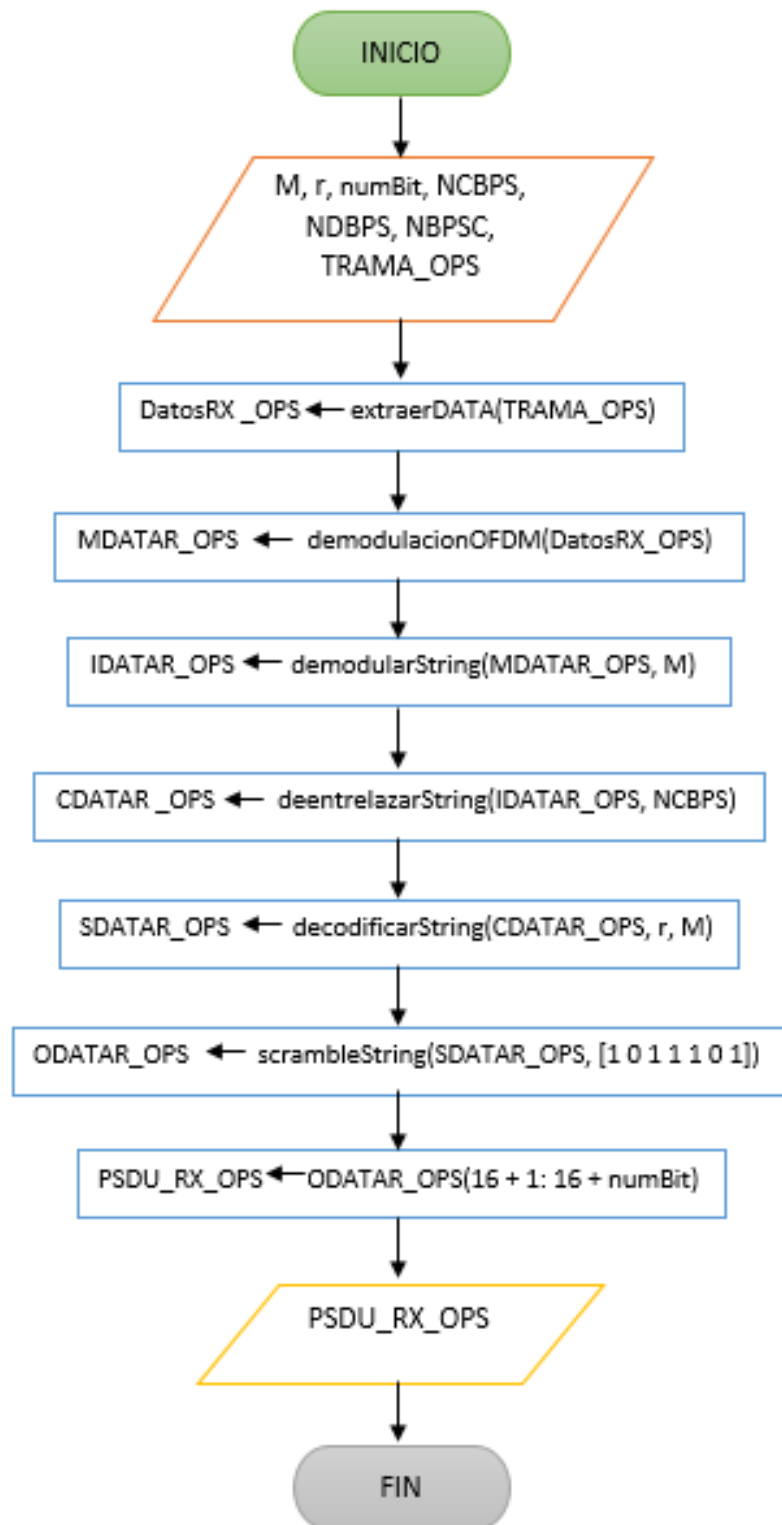


Figura A.30: Diagrama de flujo de la función Receptor OFDM con técnica OPS “PSDU_RX_OPS” sección 2.6.6.

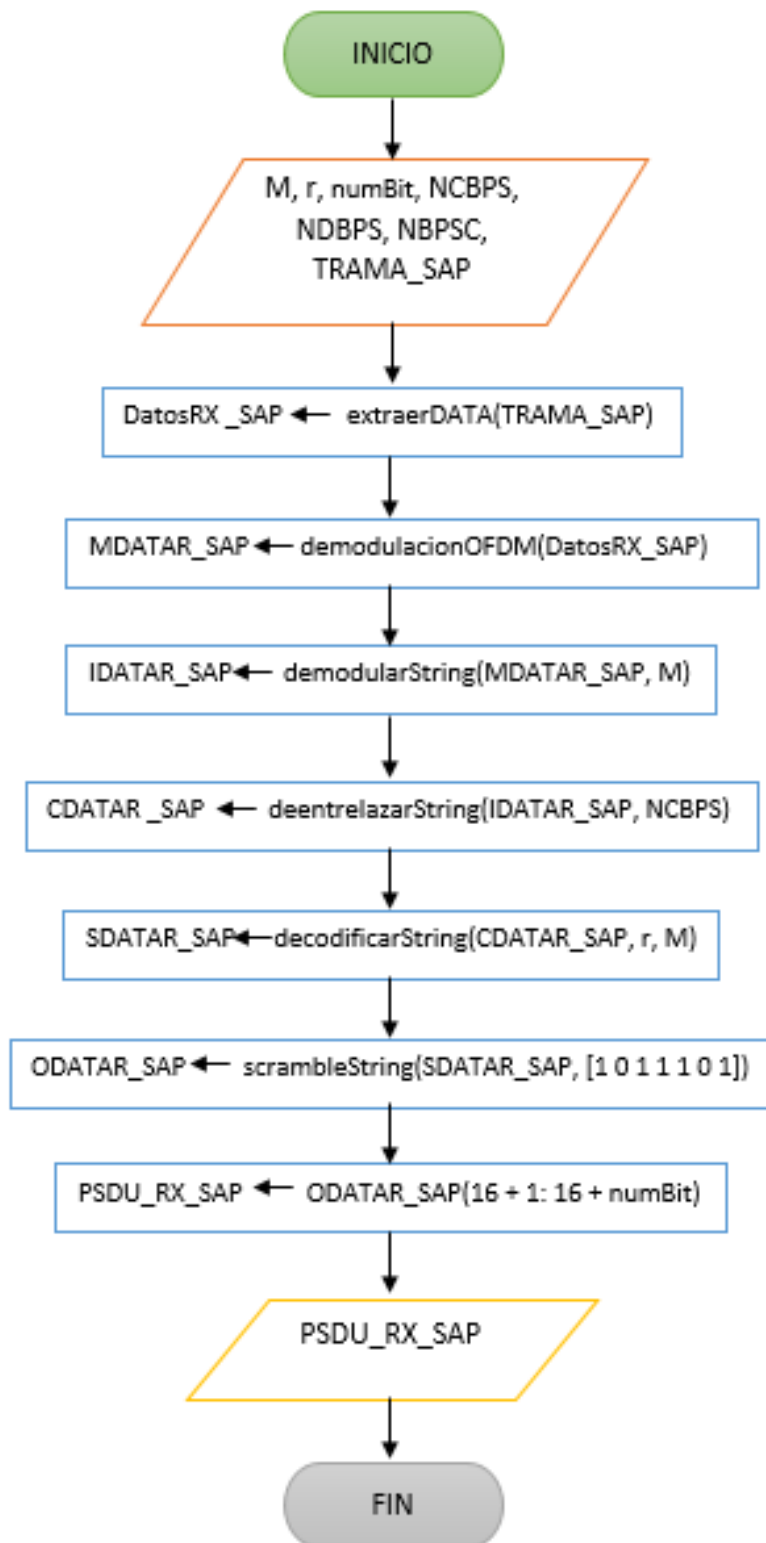


Figura A.31: Diagrama de flujo de la función Receptor OFDM con técnica SAP “PSDU_RX_SAP” sección 2.6.6.

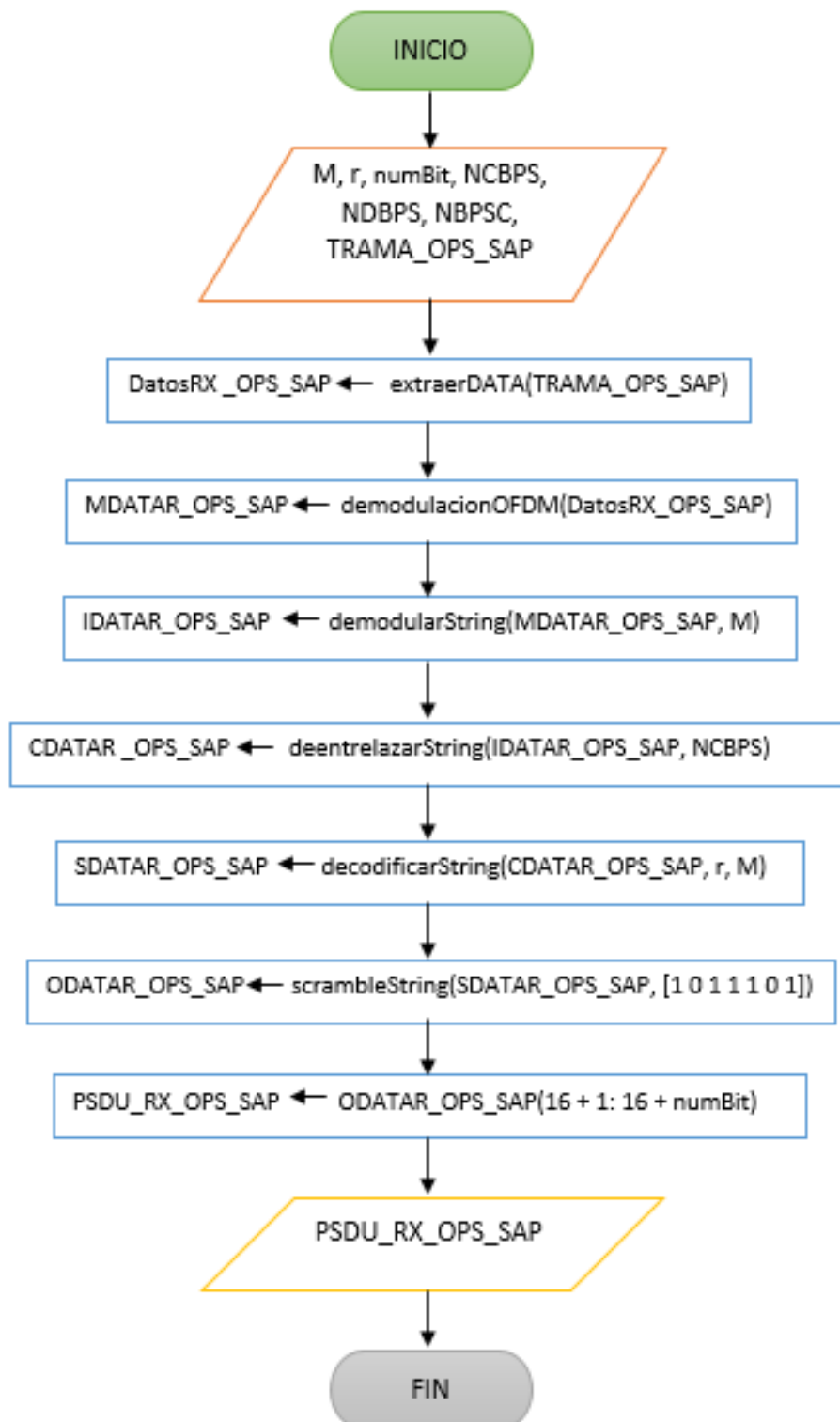


Figura A.32: Diagrama de flujo de la función Receptor OFDM con técnica OPS-SAP “PSDU_RX_OPS_SAP” sección 2.6.6.

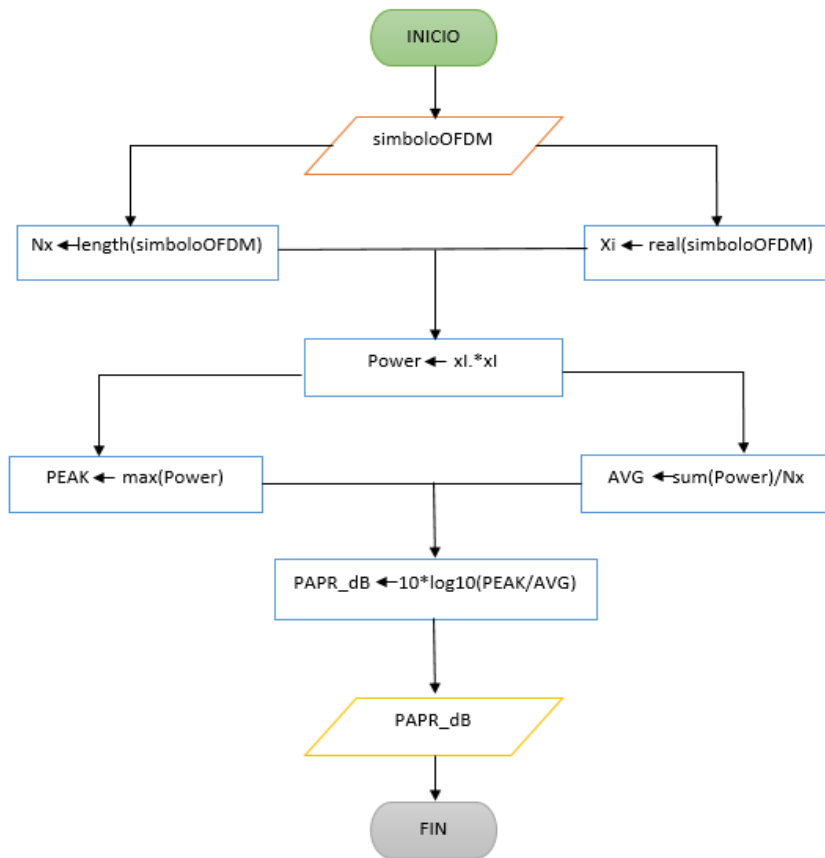


Figura A.33 Diagrama de flujo de la función calcularPAPR() sección 2.7.1.

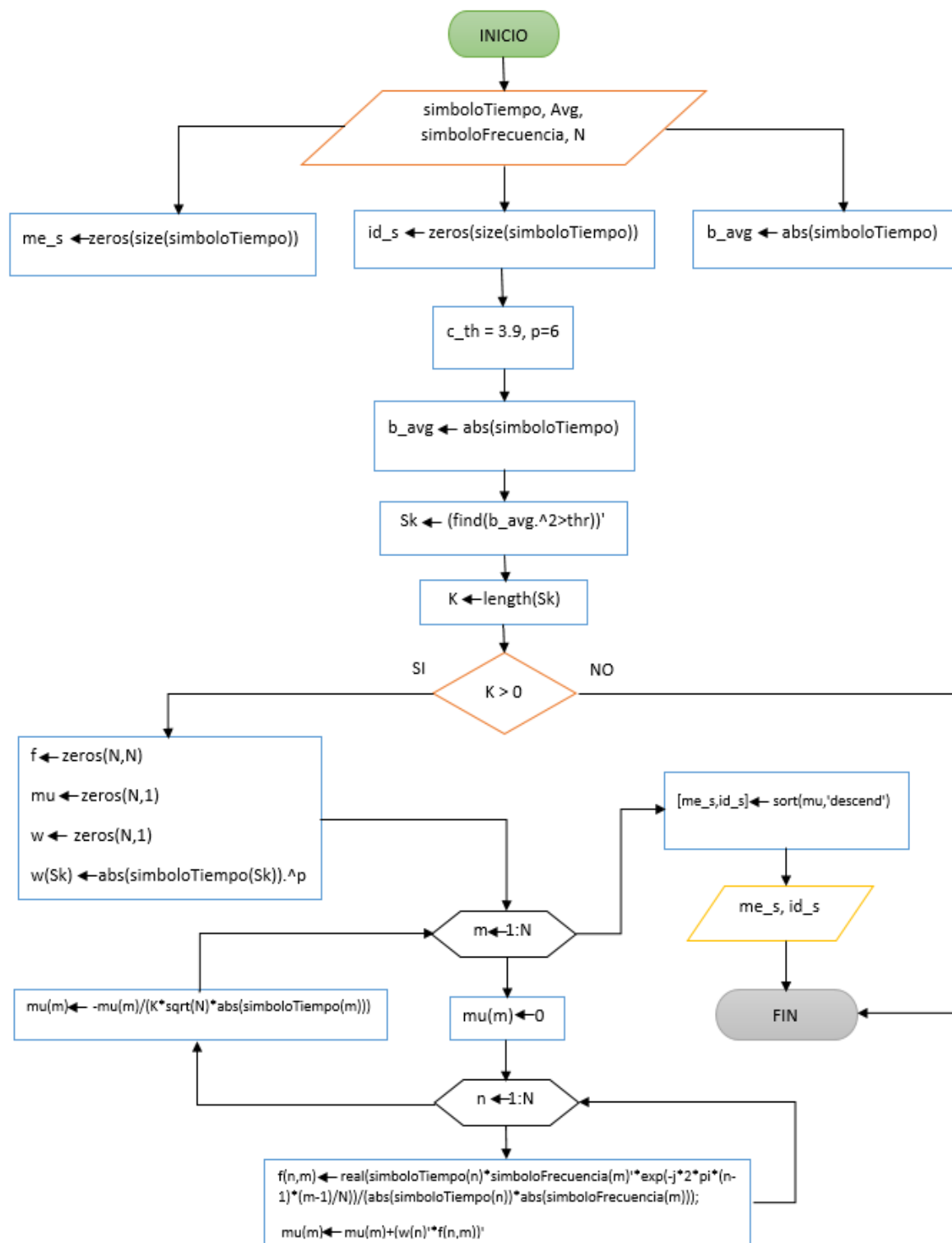


Figura A.34: Diagrama de flujo de la función `metrica()` sección 2.7.2.

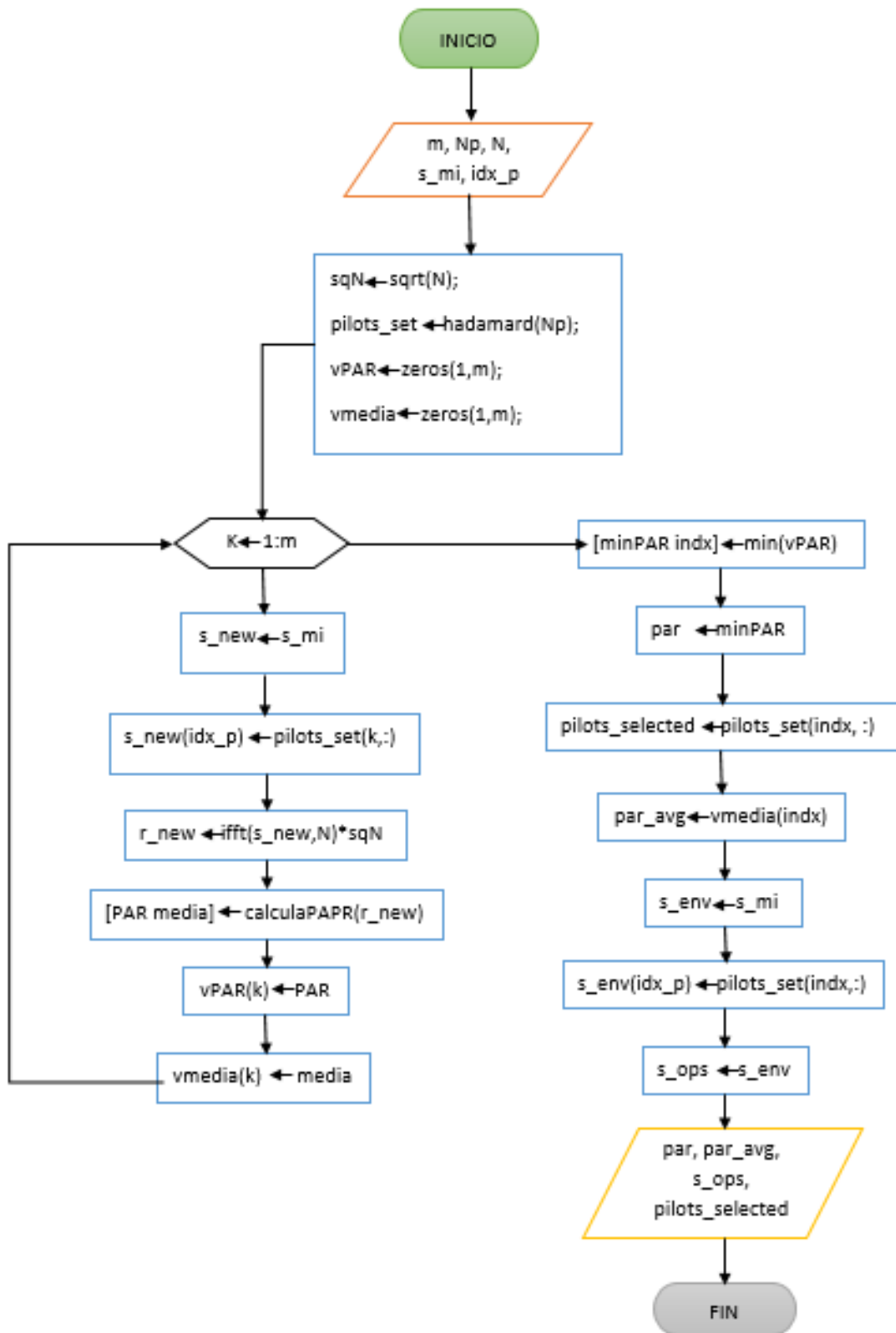


Figura A.35: Diagrama de flujo de la función OPS() sección 2.7.2 a.

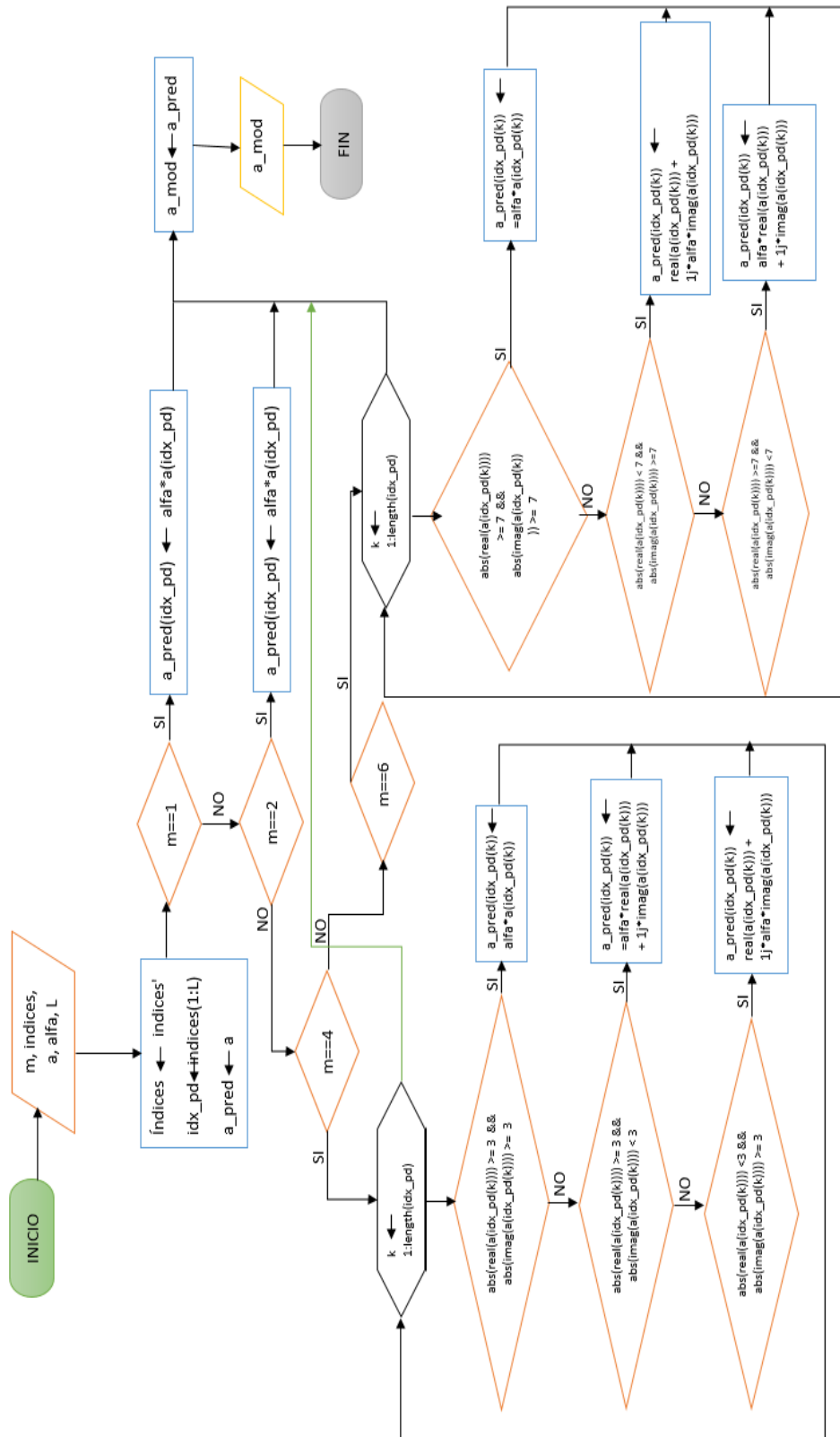


Figura A.36: Diagrama de flujo de la función predistorsion() sección 2.7.2 b.

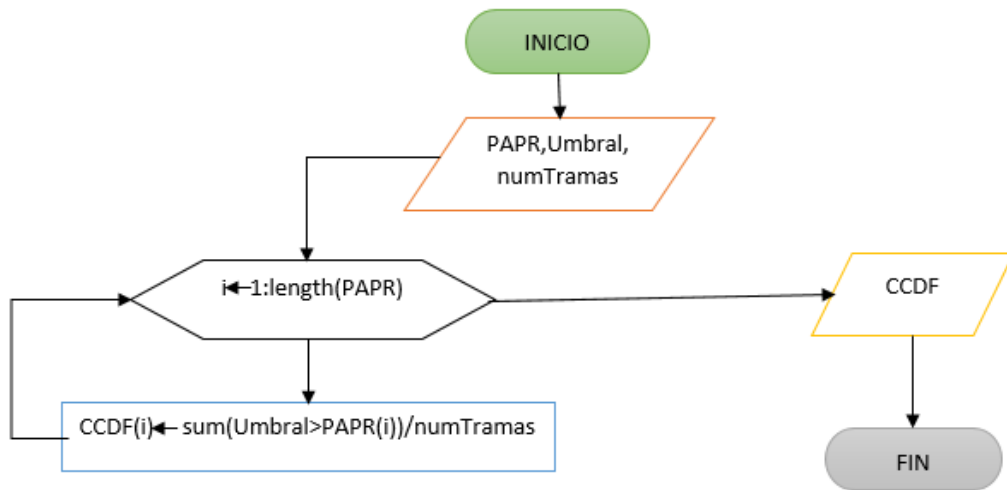


Figura A.37: Diagrama de flujo de la función calcularCCDF() sección 2.8.2.

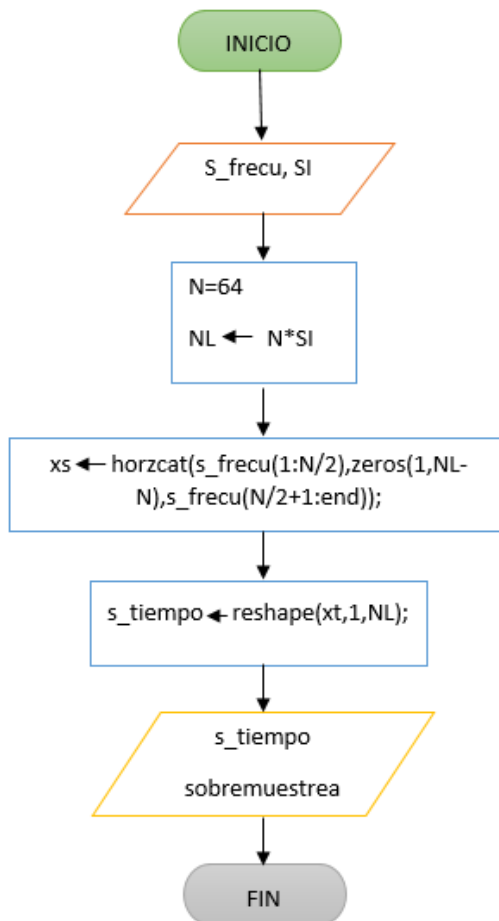


Figura A.38: Diagrama de flujo de la función PSD sección 2.8.3.

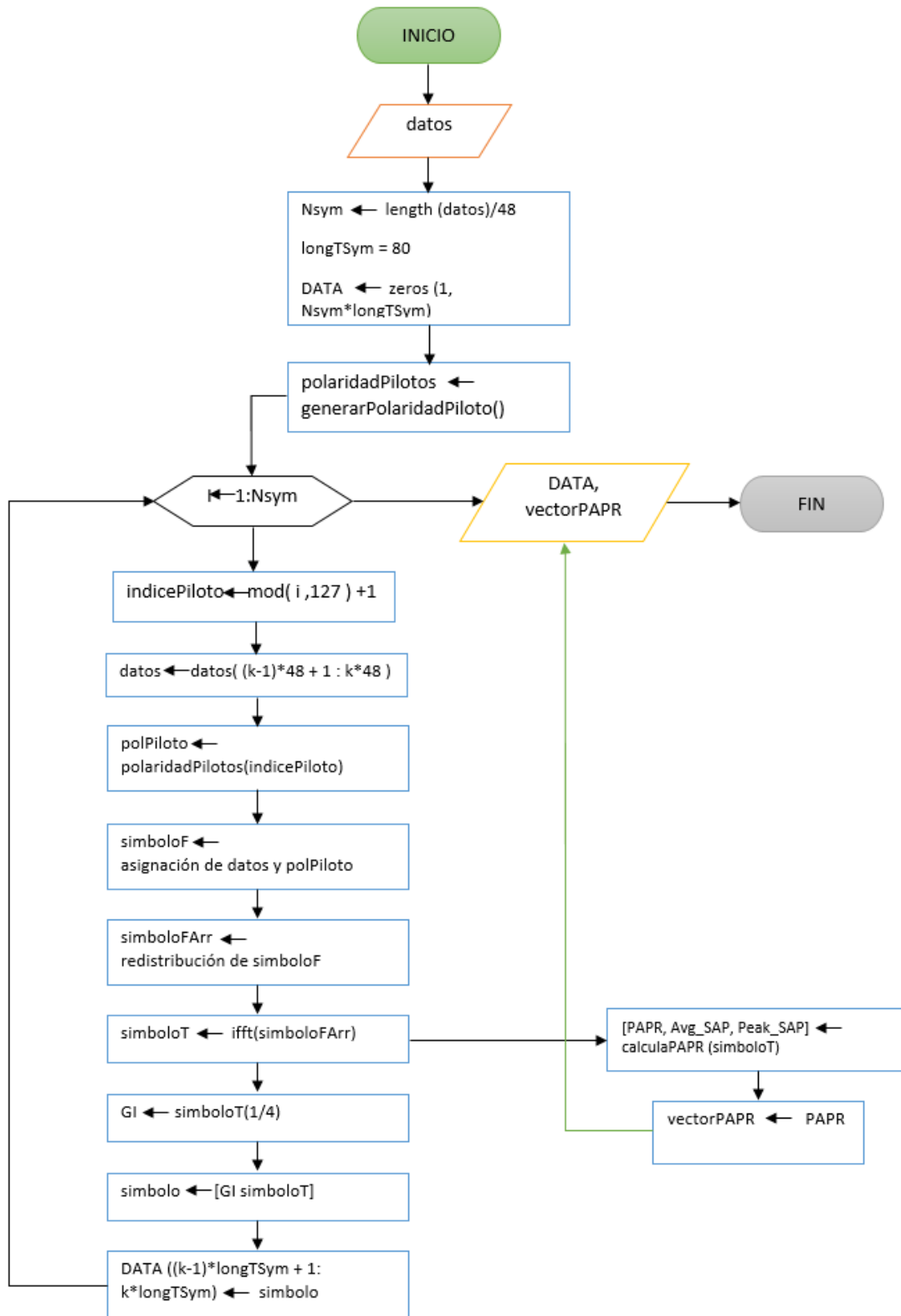


Figura A.39: Diagrama de flujo de la función Símbolo OFDM sección 2.7.2.

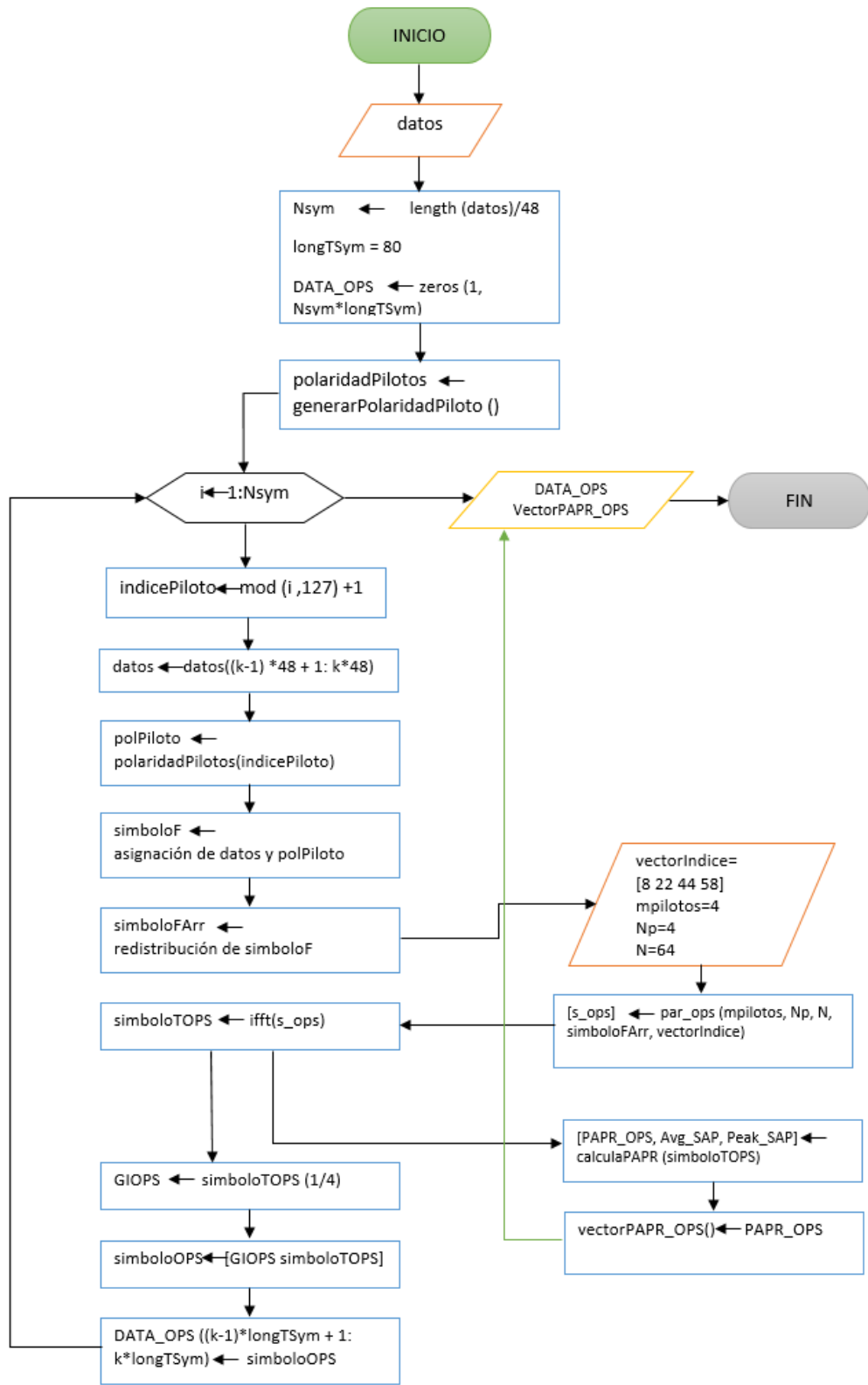


Figura A.40: Diagrama de flujo de la función Símbolo OFDM con técnica OPS sección 2.7.2.

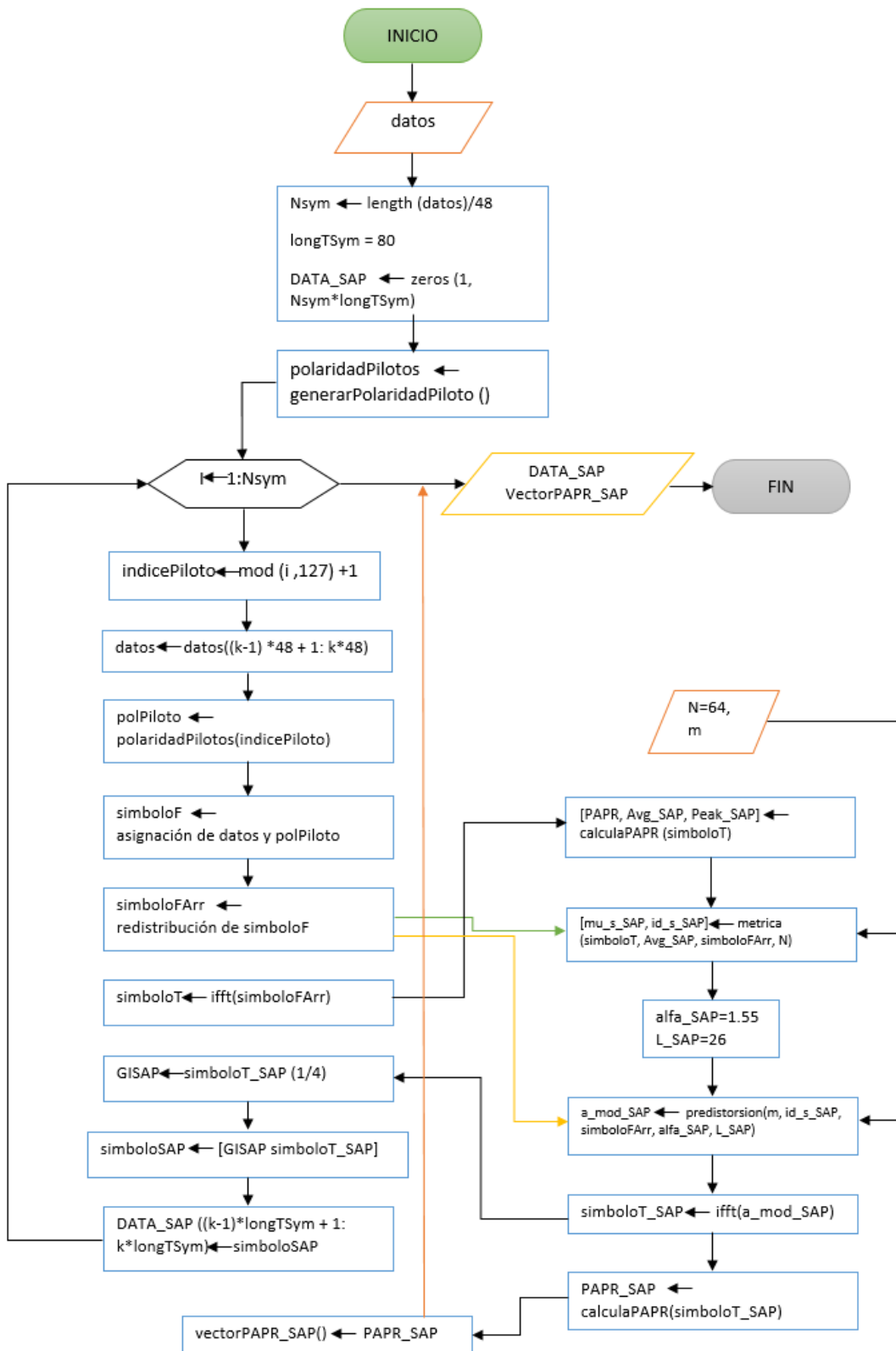


Figura A.41: Diagrama de flujo de la función Símbolo OFDM con técnica SAP sección 2.7.2.

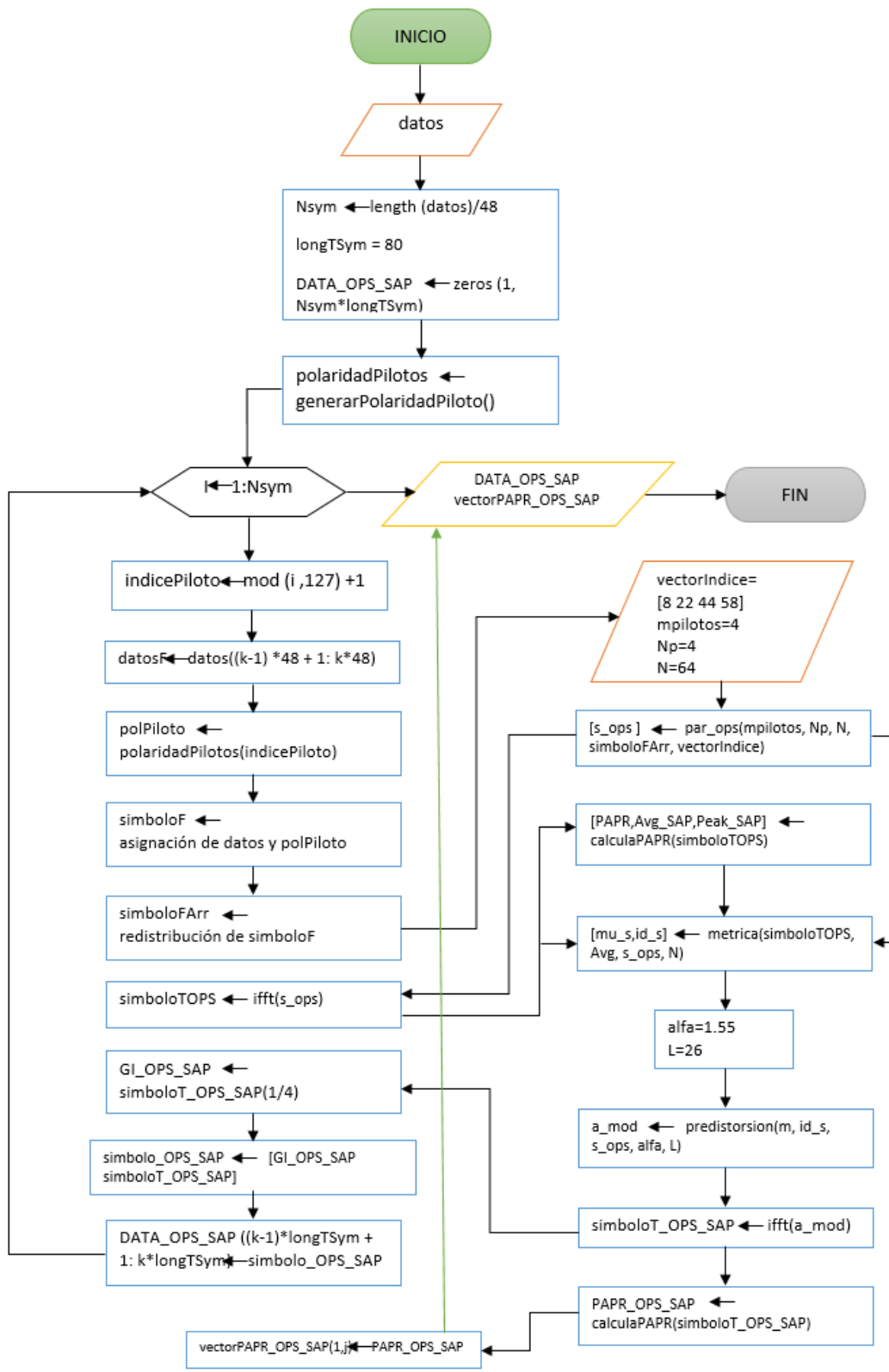


Figura A.42: Diagrama de flujo de la función símbolo OFDM OPS-SAP sección 2.7.2.

ANEXO B Códigos y Funciones

Este Anexo contiene todos los códigos desarrollados que pertenecen a cada función definida en el capítulo 2.

```
%Función encargada de realizar la creación del preámbulo corto.
function Pcorto = CrearPreambuloCorto(app)

    % Proceso de formación de preámbulo corto

    preambuloC = zeros(1,53);
    % Asignación del valor de las portadoras.

    preambuloC(1,3) = 1+1i;
    preambuloC(1,7) = -1-1i;
    preambuloC(1,11) = 1+1i;
    preambuloC(1,15) = -1-1i;
    preambuloC(1,19) = -1-1i;
    preambuloC(1,23) = 1+1i;
    preambuloC(1,31) = -1-1i;
    preambuloC(1,35) = -1-1i;
    preambuloC(1,39) = 1+1i;
    preambuloC(1,43) = 1+1i;
    preambuloC(1,47) = 1+1i;
    preambuloC(1,51) = 1+1i;

    % Proceso de normalización de la energía.
    preambuloC = sqrt(13/6) * preambuloC;

    % Proceso de discretización.
    preambuloCArr = zeros(1,64);
    preambuloCArr(1) = 0;
    preambuloCArr(2:27) = preambuloC(28:53);
    preambuloCArr(28:38) = 0;
    preambuloCArr(39:64) = preambuloC(1:26);
    % -----

    preambuloCifft = ifft(preambuloCArr);
    preambuloCCut = preambuloCifft(1:length(preambuloCifft)/4);
    Pcorto = zeros(1, 64 * 2 * 1.25);
    for i=1:10
        Pcorto(1,[(i-1)*(64 / 4) + 1 : i*(64 / 4) ]) = preambuloCCut;
    end
```

Seccion B.1: Código de la función CrearPreambuloCorto() sección 2.4.2.1.

```

%Función encargada de realizar la creación del preámbulo largo.
function Plargo = CrearPreambuloLargo(app)

    % Proceso de formación de preámbulo largo

    preambuloL = ones(1,53);

    % Asignación del valor de las portadoras.
    preambuloL(1,3)= -1;
    preambuloL(1,4)= -1;
    preambuloL(1,7)= -1;
    preambuloL(1,9)= -1;
    preambuloL(1,16)= -1;
    preambuloL(1,17)= -1;
    preambuloL(1,20)=-1;
    preambuloL(1,22)=-1;
    preambuloL(1,27)=0;
    preambuloL(1,29)=-1;
    preambuloL(1,30)=-1;
    preambuloL(1,33)=-1;
    preambuloL(1,35)=-1;
    preambuloL(1,37)=-1;
    preambuloL(1,38)=-1;
    preambuloL(1,39)=-1;
    preambuloL(1,40)=-1;
    preambuloL(1,41)=-1;
    preambuloL(1,44)=-1;
    preambuloL(1,45)=-1;
    preambuloL(1,47)=-1;
    preambuloL(1,49)=-1;

    % Proceso de discretización.
    preambuloLArr = zeros(1,64);
    preambuloLArr(1) = 0;
    preambuloLArr(2:27) = preambuloL(28:53);
    preambuloLArr(28:38) = 0;
    preambuloLArr(39:64) = preambuloL(1:26);
    % -----

    preambuloLifft = ifft(preambuloLArr);
    GI2 = preambuloLifft( length(preambuloLifft)/2 + 1 : length(preambuloLifft) );
    Plargo = [GI2 preambuloLifft preambuloLifft];    % Se procede a armar el preámbulo

```

Seccion B.2: Código de la función CrearPreámbuloLargo() sección 2.4.2.2.

```

%Creación del campo Signal o header
cabeceraBits = round(rand(1,18));
tail = zeros(1,6);
cabeceraBits = [cabeceraBits tail];
cabeceraCodificada = codificarString(app,cabeceraBits, 1/2);
cabeceraEntrelazada = entrelazarString(app,cabeceraCodificada, 48);
cabeceraModulada = modularString(app,cabeceraEntrelazada, 2);
SIGNAL = armarSimboloOFDM(app,cabeceraModulada, 1);

```

Seccion B.3: Código de la función del campo SIGNAL sección 2.4.3.

```

%Creación del campo DATA.
PSDU = round(rand(1,numBit));
ODATA = PSDU;
SDATA = scrambleString(app,ODATA, [1 0 1 1 1 0 1]);
CDATA = codificarString(app,SDATA, r);
IDATA = entrelazarString(app,CDATA, NCBPS);
MDATA = modularString(app,IDATA, M);
% Todos los símbolos OFDM generados son concatenados
DATA((k-1)*longTSym + 1 : k*longTSym) = simbolo;
DATA_OPS_SAP((k-1)*longTSym + 1 : k*longTSym) = simbolo_OPS_SAP;

```

Seccion B.4: Código de la función crear el campo DATA sección 2.4.4.

```

% Función en la cual se realiza el proceso de aleatorización.
function SDatos = scrambleString(app,datos,estadoInicial)
    SDatos = zeros(1, length(datos));
    estado = estadoInicial;
    for i=1:length(datos)
        estadoS = xor(estado(4),estado(7));
        SDatos(i) = xor(estadoS, datos(i));
        estado = circshift(estado, 1,2);
        estado(1) = estadoS;
    end
end

```

Seccion B.5: Código de la función scrangleString() sección 2.4.5.

```

% Función en la cual se realiza el proceso de codificación.
function CDatos = codificarString(app,datos,R)
    g0 = 133;
    g1 = 171;
    k = 7;
    trellis = poly2trellis(k, [g1 g0]);

    switch R
        case 1/2
            punctPattern = [1 1];
        case 2/3
            punctPattern = [1 1 1 0];
        case 3/4
            punctPattern = [1 1 1 0 0 1];
        otherwise
            error('TASA DE CODIFICACIÓN NO VÁLIDO');
    end

    punctPattern = fliplr(punctPattern);

    CDatos = convenc(datos, trellis, punctPattern);
end

```

Seccion B.6: Código de la función codificarString() sección 2.4.6.

```

%Función encargada de realizar el proceso de entrelazado (Barajador).
function EDatos = entrelazarString(app,datos,Ncbps)

    permutacion = zeros(1, length(datos)); % Creación del vector para la permutación
    EDatos = zeros(1, length(datos));
    Nbpsc = Ncbps / 48; % Número de bits (codificados) por sub portadora
    s = max(Nbpsc/2 , 1);

    % Se procede a dividir el string de bits en cadenas de un número múltiplo de Ncbps

    for a = 1 : length(datos) / Ncbps
        % Se realiza la primera permutación
        for b = 1 : Ncbps
            c = (Ncbps/16) * mod((b-1), 16) + floor((b-1)/16); % Regla de la permutación
            k = (a-1)*Ncbps + b; % Índice original real
            i = (a-1)*Ncbps + (c+1); % Índice nuevo real
            permutacion(k) = datos(i); % Permutación
        end

        % Se realiza la segunda permutación
        for b = 1 : Ncbps
            c = s * floor((b-1)/s) + mod( ( (b-1) + Ncbps - floor( 16 * (b-1)/Ncbps ) ), s);
            i = (a-1)*Ncbps + b;
            j = (a-1)*Ncbps + (c+1);
            EDatos(i) = permutacion(j);
        end
    end
end
end
end

```

Seccion B.7: Código de la función entrelazarString() sección 2.4.7.

```

%Función encargada de realizar el proceso de modulación.
function MDatos = modularString(app,datos,M)
    % Los términos multiplicativos son factores de normalización.
    switch M
        case 2
            MDatos = 1 * modularBPSK(app,datos);
        case 4
            MDatos = ( 1 / sqrt(2) ) * modularQPSK(app,datos);
        case 16
            MDatos = ( 1 / sqrt(10) ) * modular16QAM(app,datos);
        case 64
            MDatos = ( 1 / sqrt(42) ) * modular64QAM(app,datos);
        otherwise
            error('NÚMERO DE NIVELES INCORRECTO - MODULACIÓN')
    end
end
end
end

```

Seccion B.8: Código de la función modularString sección 2.4.8.

```

%Función para poder realizar una modulación BPSK.
function modulado = modulacionBPSK(app,datos)
    bitsModu = comm.PSKModulator(2,'BitInput', true, 'PhaseOffset', 0);
    modulado = step(bitsModu,datos');
    modulado = modulado';
end

```

Seccion B.9: Código de la función modulacionBPSK sección 2.4.8 a.

```

%Función para poder realizar una modulación QPSK.
function modulado = modulacionQPSK(app,datos)
    M = 4;
    m = 2;

    modulado = zeros(1,length(datos)/m);

    for i = 1:m:length(datos)
        if datos(i) == 1
            if datos(i+1)==1    %11
                modulado( (i+1)/2 ) = 1+1i;
            else                %10
                modulado( (i+1)/2 ) = 1-1i;
            end
        else
            if datos(i+1)==1    %01
                modulado( (i+1)/2 ) = -1+1i;
            else                %00
                modulado( (i+1)/2 ) = -1-1i;
            end
        end
    end
end
end
end

```

Seccion B.10: Código de la función modulacionQPSK() sección 2.4.8 b.

```

%Función para poder realizar una modulación 16QAM.
function modulado = modulacion16QAM(app,datos)

    M = 16;
    m = 4;
    modulado = zeros(1,length(datos)/m);
    for i = 1:m:length(datos)
        re = 0;
        im = 0*i;

        if datos(i) == 1
            re = +1;
        else
            re = -1;
        end

        if datos(i+1) == 1
            re = re*1;
        else
            re = re*3;
        end

        if datos(i+2) == 1
            im = +1i;
        else
            im = -1i;
        end

        if datos(i+3) == 1
            im = im*1;
        else
            im = im*3;
        end

        modulado( (i+m-1)/m ) = re + im;
    end
end
end
end

```

Seccion B.11: Código de la función modulacion16QAM() sección 2.4.8 c.

```

%Función para poder realizar una modulación 64QAM.
function modulado = modulacion64QAM(app,datos)

M = 64;
m = 6;
modulado = zeros(1,length(datos)/m);

for i = 1:m:length(datos)
    re = 0;
    im = 0*i;

    if datos(i) == 1
        re = +1;
    else
        re = -1;
    end

    if datos(i+1) == 1
        if datos(i+2) == 1
            re = re*3;
        else
            re = re*1;
        end
    else
        if datos(i+2) == 1
            re = re*5;
        else
            re = re*7;
        end
    end

    if datos(i+3) == 1
        im = +1i;
    else
        im = -1i;
    end

    if datos(i+4) == 1
        if datos(i+5) == 1
            im = im*3;
        else
            im = im*1;
        end
    else
        if datos(i+5) == 1
            im = im*5;
        else
            im = im*7;
        end
    end
    modulado( (i+m-1)/m ) = re + im;
end
end

```

Seccion B.12: Código de la función modulacion64QAM() sección 2.4.8 d.

```

%Función para poder realizar el proceso de generar la polaridad de las subportadoras piloto..
function polaridad = generarPolaridadPiloto(app)

    polaridad = [1,1,1,1,-1,-1,-1,1, -1,-1,-1,-1, 1,1,-1,1, -1,-1,1,1, -1,1,1,-1, 1,1,1,1,

end

```

Seccion B.13: Código de la función generarPolaridadPiloto() 2.4.9 a.

```

% Se arma el símbolo OFDM uno a la vez.
indicePiloto = mod( k ,127 ) +1;
datosF=MDATA( (k-1)*48 + 1 : k*48 );
polPiloto=polaridadPilotos(indicePiloto);

% Generación de un arreglo con 53 muestras
simboloF = zeros(1,53);
simboloF(1,1:5) = datosF(1:5);
simboloF(1,6) = 1*polPiloto; % piloto
simboloF(1,7:19) = datosF(6:18);
simboloF(1,20) = 1*polPiloto; % piloto
simboloF(1,21:26) = datosF(19:24);
simboloF(1,27) = 0; % dc
simboloF(1,28:33) = datosF(25:30);
simboloF(1,34) = 1*polPiloto; % piloto
simboloF(1,35:47) = datosF(31:43);
simboloF(1,48) = -1*polPiloto; % piloto
simboloF(1,49:53) = datosF(44:48);

```

Seccion B.14: Código de la función para obtener el símboloF sección 2.4.9 b.

```

% Proceso de redistribución del símbolo OFDM (discretización).
simboloFArr = zeros(1,64);
simboloFArr(1) = 0;
simboloFArr(2:27) = simboloF(28:53);
simboloFArr(28:38) = 0;
simboloFArr(39:64) = simboloF(1:26);
% -----

```

Seccion B.15: Código de la función para obtener el símboloFArr sección 2.4.9 b.

```

simboloT = ifft(simboloFArr); % Implementación de la transformada inversa de furier
                                % para pasar el símbolo OFDM al
                                % dominio del tiempo continuo.

% Agregación del prefijo cíclico
GI = simboloT( 3*length(simboloT)/4 + 1 : length(simboloT) ); % El último 1/4

% Concatenación del prefijo cíclico junto con el símbolo OFDM.
simbolo = [GI simboloT];

% Todos los símbolos OFDM generados son concatenados
DATA((k-1)*longTSym + 1 : k*longTSym) = simbolo;

```

Seccion B.4.1: Código de la función para obtener el arreglo DATA sección 2.4.4.

```

function s_amplificada = amplificador(app,s_tiempo,IBO)

    % Datos del HPA TIPO RAPP
    p=4; %Factor de Precisión
    ga=1; %Ganancia del amplificador
    IBOdB=IBO; %Valor del IBO
    amp=abs(s_tiempo); %Amplitud de la señal
    fase=angle(s_tiempo); %Fase de la señal
    x=amp.*exp(1i*fase); % Ecuación que describe el HPA
    Pme=mean(abs(x).^2); %Potencia promedio de la señal de entrada
    Pmaxe=sqrt(10^(IBOdB/10)*Pme); %Nivel de saturación
    %Conversión AM/AM
    G=(ga*amp)./(1+(amp/Pmaxe).^(2*p)).^(1/(2*p));
    %Conversión AM/PM
    Fhi=0;
    Theta=fase;
    s_amplificada=G.*exp(1j*(Theta+Fhi));

end

```

Seccion B.17: Código de la función amplificador() sección 2.4.10.

```

rayleighChan=comm.RayleighChannel(...
    'SampleRate',20e6,...
    'PathDelays',[0 2*1e-6 7*1e-6 12*1e-6],...
    'AveragePathGains',[0 -10 -39 -60],...
    'MaximumDopplerShift',0);
% CANAL RAYLEIGH
simbolo_OFDM_Rayleigh=(step(rayleighChan,TRAMA.')); % Paso de la señal por el canal Rayleigh.
DatosRX_Rayleigh=awgn(simbolo_OFDM_Rayleigh,SNR(i),'measured');

```

Seccion B.18: Código de la función para la obtención de los datos a través de un canal Rayleigh sección 2.5.1.

```

rayleighChan=comm.RayleighChannel(...
    'SampleRate',20e6,...
    'PathDelays',[0 2*1e-6 7*1e-6 12*1e-6],...
    'AveragePathGains',[0 -10 -39 -60],...
    'MaximumDopplerShift',0);
simbolo_OFDM_Rayleigh_OPS_SAP=(step(rayleighChan,TRAMA_OPS_SAP.'));
DatosRX_Rayleigh_OPS_SAP_PAQUETE=awgn(simbolo_OFDM_Rayleigh_OPS_SAP,SNR(i),'measured');

```

Seccion B.19: Código de la función para la obtención de los datos con técnica OPS-SAP a través de un canal Rayleigh sección 2.5.1.

```

ricianChan=comm.RicianChannel(...
    'PathDelays',[0 2*1e-6 7*1e-6 12*1e-6],...
    'AveragePathGains',[0 -10 -39 -60],...
    'KFactor',10);
% CANAL RICIAN
simbolo_OFDM_Rician=(step(ricianChan,TRAMA.')); % Paso de la señal por el canal Rician.
DatosRX_Rician=awgn(simbolo_OFDM_Rician, SNR(i), 'measured');

```

Seccion B.20: Código de la función para la obtención de los datos a través de un canal Rician sección 2.5.2.


```

ricianChan=comm.RicianChannel(...
    'PathDelays',[0 2*1e-6 7*1e-6 12*1e-6],...
    'AveragePathGains',[0 -10 -39 -60],...
    'KFactor',10);
% CANAL RICIAN
simbolo_OFDM_Rician_OPS_SAP=(step(ricianChan,TRAMA_OPS_SAP.')).';
DatosRX_Rician_OPS_SAP_PAQUETE=awgn(simbolo_OFDM_Rician_OPS_SAP, SNR(i), 'measured');

```

Figura B.21: Código de la función para la obtención de los datos con técnica OPS-SAP a través de un canal Rician sección 2.5.2.

```

%Función encargada de extraer el campo DATA de la trama PDU PLCP.
function DATAT = extraerData(app,TRAMA)

    % Tshort -> 2 símbolos OFDM
    nshort = 2*80;
    % Tlong -> 2 símbolos OFDM
    nlong = 2*80;
    % T SIGNAL -> 1 símbolo OFDM
    nsignal = 80;

    DATAT = TRAMA( nshort + nlong + nsignal + 1 : length(TRAMA) );

end

```

Seccion B.22: Código de la función extraerDATA() sección 2.6.2 a.

```

%Función encargada de realizar el proceso de extraer el símbolo OFDM uno a la vez.
function Dsimbolos =extraerSimboloOFDM(app,SymboOFDM, polPiloto)

    % Obtención de los bits en frecuencia de un símbolo OFDM en tiempo
    % Primero extraigo el prefijo cíclico
    % El PC es el 20% del símbolo (1.6us de 8us)

    simboloT = SymboOFDM( 0.2*length(SymboOFDM) + 1 : length(SymboOFDM));

    % Obtengo el símbolo en tiempo

    simboloFarr = fft(simboloT);

    % Proceso de discretización (inverso).
    simboloF = zeros(1,53);
    simboloF(1:26) = simboloFarr(39:64);
    simboloF(27) = 0;
    simboloF(28:53) = simboloFarr(2:27);
    % -----
    % Se procede a extraer los datos de las subportadoras y la dc
    Dsimbolos = zeros(1,48);
    Dsimbolos(1:5) = simboloF(1:5);
    Dsimbolos(6:18) = simboloF(7:19);
    Dsimbolos(19:24) = simboloF(21:26);
    Dsimbolos(25:30) = simboloF(28:33);
    Dsimbolos(31:43) = simboloF(35:47);
    Dsimbolos(44:48) = simboloF(49:53);

end

```

Figura B.23: Código de la función extraerSimboloOFDM() sección 2.6.2 b.

```

%Función encargada de realizar el proceso de demodulación.
function MDATA = bloqueOFDMrecepcion(app,DATA)

% Obtiene un string de bits correspondiente a un número de símbolos
% OFDM en tiempo

polaridadPilotos = generarPolaridadPiloto(app);           % Al contar, comenzar desde 2, hace mod 127 y sumar 1
                                                         % mod( i ,127 ) +1
longTSym = 80;                                         % Número de muestras en tiempo de un símbolo (64 + 1/4 GI)
longFSym = 48;                                         % Número de muestras en frecuencia de datos de un símbolo
Nsym = length(DATA)/longTSym;                          % Número de símbolos (con PC) en tiempo

MDATA = zeros(1,Nsym*48);                               % Número de símbolos en frecuencia (48 por símbolo)
for i=1:Nsym
    indicePiloto = mod( i ,127 ) +1;
    simbolo = extraerSimboloOFDM(app, DATA( (i-1)*longTSym + 1 : i*longTSym ), polaridadPilotos(indicePiloto) );
    MDATA((i-1)*longFSym + 1 : i*longFSym) = simbolo;   % Se concatena los símbolos OFDM.
end
end

```

Figura B.24: Código de la función bloqueOFDMrecepcion() sección 2.6.1.

```

%Función encargada de realizar el proceso de demodulación.
function IDatos = demodularString(app,MDatos, M)

switch M
    case 2
        IDatos = demodularBPSK(app,MDatos);
    case 4
        IDatos = demodularQPSK(app,sqrt(2)*MDatos);
    case 16
        IDatos = demodular16QAM(app,sqrt(10)*MDatos);
    case 64
        IDatos = demodular64QAM(app,sqrt(42)*MDatos);
    otherwise
        errordlg('NÚMERO DE NIVELES INCORRECTO - MODULACIÓN')
end
end

```

Seccion B.25: Código de la función demodularString() sección 2.6.3.

```

%Función encargada de realizar el proceso de demodulación BPSK.
function IDatos = demodulacionBPSK(app,MDatos)

bitsDemodu = comm.PSKDemodulator(2,'BitOutput', true, 'PhaseOffset', 0);
IDatos = step(bitsDemodu, MDatos');
IDatos = IDatos';

end

```

Seccion B.26: Código de la función demodulacionBPSK() sección 2.6.3 a.

%Función encargada de realizar el proceso de demodulación QPSK.

```
function IDatos = demodulacionQPSK(app,MDatos)
```

```
    IDatos = zeros(1,length(MDatos)*2);
```

```
    for i = 1:length(MDatos)
```

```
        j = (i-1)*2 +1;
```

```
        if real(MDatos(i)) > 0
```

```
            IDatos(j) = 1;
```

```
        else
```

```
            IDatos(j) = 0;
```

```
        end
```

```
        if imag(MDatos(i)) > 0
```

```
            IDatos(j+1) = 1;
```

```
        else
```

```
            IDatos(j+1) = 0;
```

```
        end
```

```
    end
```

```
end
```

Seccion B.27: Código de la función demodulacionQPSK() sección 2.6.3 b.

%Función encargada de realizar el proceso de demodulación 16QAM.

```
function IDatos = demodulacion16QAM(app,MDatos)
```

```
    IDatos = zeros(1,length(MDatos)*4);
```

```
    for i=1:length(MDatos)
```

```
        j = (i-1)*4 + 1;
```

```
        if real(MDatos(i)) > 0
```

```
            IDatos(j) = 1;
```

```
        else
```

```
            IDatos(j) = 0;
```

```
        end
```

```
        if abs(real(MDatos(i))) < 2
```

```
            IDatos(j+1) = 1;
```

```
        else
```

```
            IDatos(j+1) = 0;
```

```
        end
```

```
        if imag(MDatos(i)) > 0
```

```
            IDatos(j+2) = 1;
```

```
        else
```

```
            IDatos(j+2) = 0;
```

```
        end
```

```
        if abs(imag(MDatos(i))) < 2
```

```
            IDatos(j+3) = 1;
```

```
        else
```

```
            IDatos(j+3) = 0;
```

```
        end
```

```
    end
```

```
end
```

Seccion B.28: Código de la función demodulacion16QAM() sección 2.6.3 c.

%Función encargada de realizar el proceso de demodulación 64QAM.

```
function IDatos = demodulacion64QAM(app,MDatos)

IDatos = zeros(1,length(MDatos)*6);

for i=1:length(MDatos)
    j = (i-1)*6 + 1;
    if real(MDatos(i)) > 0
        IDatos(j) = 1;
    else
        IDatos(j) = 0;
    end

    if abs(real(MDatos(i))) < 4
        IDatos(j+1) = 1;
    else
        IDatos(j+1) = 0;
    end

    if abs( abs(real(MDatos(i))) - 4 ) < 2
        IDatos(j+2) = 1;
    else
        IDatos(j+2) = 0;
    end

    if imag(MDatos(i)) > 0
        IDatos(j+3) = 1;
    else
        IDatos(j+3) = 0;
    end

    if abs(imag(MDatos(i))) < 4
        IDatos(j+4) = 1;
    else
        IDatos(j+4) = 0;
    end

    if abs( abs(imag(MDatos(i))) - 4 ) < 2
        IDatos(j+5) = 1;
    else
        IDatos(j+5) = 0;
    end
end

end
```

Seccion B.29: Código de la función demodulacion64QAM() sección 2.6.3 d.

```

%Función encargada de realizar el proceso de deentrelazado.
function CDatos = deentrelazarString(app,IBits, Ncbps)

    % El proceso es similar a entrelazar, pero se cambia los índices el momento
    % de asignar.

    permutacion = zeros(1, length(IBits)); % Se crea el vector para la permutación
    CDatos = zeros(1, length(IBits));
    Nbpsc = Ncbps / 48; % Número de bits (codificados) por sub portadora
    s = max(Nbpsc/2 , 1);

    % Se procede a dividir el string de bits en cadenas de un número múltiplo de Ncbps

    for a = 1 : length(IBits) / Ncbps
        % Se realiza la primera permutación
        for b = 1 : Ncbps
            c = s * floor((b-1)/s) + mod( ( (b-1) + Ncbps - floor( 16 * (b-1)/Ncbps ) ), s);
            i = (a-1)*Ncbps + b;
            j = (a-1)*Ncbps + (c+1);
            permutacion(j) = IBits(i);
        end

        % Se realiza la segunda permutación
        for b = 1 : Ncbps
            c = (Ncbps/16) * mod((b-1), 16) + floor((b-1)/16); % Regla de la permutación
            k = (a-1)*Ncbps + b; % Índice original real
            i = (a-1)*Ncbps + (c+1); % Índice nuevo real
            CDatos(i) = permutacion(k); % Permutación
        end
    end

end
end
end

```

Seccion B.30: Código de la función deentrelazarString() sección 2.6.4.

```

%Función encargada de realizar el proceso de decodificación.
function SDatos = decodificarString(app,CDatos, R, M)

    g0 = 133;
    g1 = 171;
    k = 7;
    trellis = poly2trellis(k, [g1 g0]);
    tblen = 0;

    switch R
        case 1/2
            punctPattern = [1 1];
            tblen = 24*log2(M);
        case 2/3
            punctPattern = [1 1 1 0];
            tblen = 96;
        case 3/4
            punctPattern = [1 1 1 0 0 1];
    end

    switch M
        case 2
            tblen = 36;
        case 4
            tblen = 96;
        case 16
            tblen = 96;
        case 64
            tblen = 96;
    end

    otherwise
        error('RATE DE CODIFICACIÓN NO VÁLIDO');
    end

    punctPattern = fliplr(punctPattern);
    SDatos = vitdec(CDatos,trellis,tblen, 'trunc', 'hard',punctPattern);

end
end
end

```

Seccion B.31: Código de la función decodificarString() sección 2.6.5.

```

%Recepcion de los datos canal AWGN
DatosRX = extraerData(app,DatosRX_PAQUETE);
MDATAR = demodulacionOFDM(app,DatosRX);
IDATAR = demodularString(app,MDATAR, M);
CDATAR = deentrelazarString(app,IDATAR, NCBPS);
SDATAR = decodificarString(app,CDATAR, r, M);
ODATAR = scrambleString(app,SDATAR, [1 0 1 1 1 0 1]);
PSDU_RX = ODATAR;
err(1,i)=err(i)+sum(PSDU_RX(1:numBit)~=PSDU);

```

Seccion B.32: Código que representa la obtención del campo DATA en recepción sección 2.6.6.

```

%Recepcion de los datos canal AWGN con OPS-SAP
DatosRX_OPS_SAP = extraerData(app,DatosRX_OPS_SAP_PAQUETE);
MDATAR_OPS_SAP = demodulacionOFDM(app,DatosRX_OPS_SAP);
IDATAR_OPS_SAP = demodularString(app,MDATAR_OPS_SAP, M);
CDATAR_OPS_SAP = deentrelazarString(app,IDATAR_OPS_SAP, NCBPS);
SDATAR_OPS_SAP = decodificarString(app,CDATAR_OPS_SAP, r, M);
ODATAR_OPS_SAP = scrambleString(app,SDATAR_OPS_SAP, [1 0 1 1 1 0 1]);
PSDU_RX_OPS_SAP = ODATAR_OPS_SAP;

```

Seccion B.33: Código que representa la obtención del campo DATA en recepción con la técnica OPS-SAP sección 2.6.6.

```

%Función encargada de realizar el cálculo de la PAPR.
function [PAPR_dB,AVG,PEAK] = calculaPAPR(app,simboloOFDM)

% Detailed explanation goes here
Nx=length(simboloOFDM);
xI=real(simboloOFDM);
%xQ=imag(simboloOFDM);

Power=xI.*xI; %+ xQ.*xQ;
PEAK=max(Power);
AVG=sum(Power)/Nx;
PAPR_dB=10*log10(PEAK/AVG);

end

```

Seccion B.34: Código de la función calculaPAPR() sección 2.7.1.

```

%Función encargada de realizar el cálculo de la métrica.
function [me_s,id_s] = metrica(app, simboloTiempo, Avg, simboloFrecuencia, N)

%Parámetros
me_s=zeros(size(simboloTiempo));
id_s=zeros(size(simboloTiempo));

c_th = 3.9;           %muestras que superen 3,9dB de la potencia pico promedio se aplica la tecnica
p=6;                 %norma
thr=10^(0.1*c_th)*Avg; %Umbral
b_avg=abs(simboloTiempo);

Sk=(find(b_avg.^2>thr)); %Indices de portadoras cuya potencia media supere umbral
K=length(Sk);          %Tamaño del conjunto Sk -->
                       %sus elementos son los índices de las muestras de
                       %salida con magnitudes que exceden el umbral

if K > 0

f=zeros(N,N);        %Funcion de coste
mu=zeros(N,1);       %Métrica
w=zeros(N,1);        %Funcion de pesos
w(Sk)=abs(simboloTiempo(Sk)).^p;

%Aplicación de técnica en las portadoras indicadas
for m =1:N
mu(m)=0;
for n = 1:N
f(n,m) = real(simboloTiempo(n)*simboloFrecuencia(m)*exp(-j*2*pi*(n-1)*(m-1)/N))/(abs(simboloTiempo(n))*abs(simboloFrecuencia(m)));
mu(m) = mu(m)+(w(n)*f(n,m));
end
mu(m)=-mu(m)/(K*sqrt(N)*abs(simboloTiempo(m)));
end

%Ordenamos las muestras en orden descendente
[me_s,id_s] = sort(mu,'descend');

end

end

```

Seccion B.35: Código de la función metrica() sección 2.7.2 b.

%Función encargada de realizar la técnica OPS.

```

function [par, par_avg, s_ops, pilots_selected ] = OPS(app,m, Np, N, s_mi, idx_p)

sqN=sqrt(N);
pilots_set=hadamard(Np);
vPAR=zeros(1,m);
vmedia=zeros(1,m);
for k=1:m
%insert the pilots and get the new PAR
s_new=s_mi;
s_new(idx_p)=pilots_set(k,:);
r_new=ifft(s_new,N)*sqN;
[PAR media]=calculaPAPR(app,r_new);
vPAR(k)=PAR;
vmedia(k)= media;
end
%select the one with lowest PAR and transmit it
[minPAR indx]=min(vPAR);
par = minPAR;
pilots_selected=pilots_set(indx, :);
% par_max =vmax(indx);
par_avg=vmedia(indx);
% señal enviada en frecuencia (a_env)
s_env=s_mi;
s_env(idx_p)=pilots_set(indx,:);
s_ops = s_env;

end

```

Seccion B.36: Código de la función OPS() sección 2.7.2 a.

```

%Función encargada de realizar la predistorsión de las muestras del símbolo OFDM.
function a_mod = predistorsion(app,m, indices, a, alfa, L)

    indices=indices';
    idx_pd = indices(1:L);
    %Escalamos la constelación un factor alfa, a las L muestras
    %seleccionadas
    a_pred = a;
    if m==1
        a_pred(idx_pd) = alfa*a(idx_pd);
    elseif m==2
        a_pred(idx_pd) = alfa*a(idx_pd);
    elseif m==4
        for k =1:length(idx_pd)
            if abs(real(a(idx_pd(k)))) >= 3 && abs(imag(a(idx_pd(k)))) >= 3
                a_pred(idx_pd(k)) = alfa*a(idx_pd(k));
            elseif (abs(real(a(idx_pd(k)))) >= 3 && abs(imag(a(idx_pd(k)))) < 3)
                a_pred(idx_pd(k)) = alfa*real(a(idx_pd(k))) + 1j*imag(a(idx_pd(k)));
            elseif (abs(real(a(idx_pd(k)))) <3 && abs(imag(a(idx_pd(k)))) >= 3)
                a_pred(idx_pd(k)) = real(a(idx_pd(k))) + 1j*alfa*imag(a(idx_pd(k)));
            end
        end
    elseif m==6
        for k =1:length(idx_pd)
            if abs(real(a(idx_pd(k)))) >= 7 && abs(imag(a(idx_pd(k)))) >= 7
                a_pred(idx_pd(k)) = alfa*a(idx_pd(k));
            elseif (abs(real(a(idx_pd(k)))) < 7 && abs(imag(a(idx_pd(k)))) >=7)
                a_pred(idx_pd(k)) = real(a(idx_pd(k))) + 1j*alfa*imag(a(idx_pd(k)));
            elseif (abs(real(a(idx_pd(k)))) >=7 && abs(imag(a(idx_pd(k)))) <7)
                a_pred(idx_pd(k)) = alfa*real(a(idx_pd(k))) + 1j*imag(a(idx_pd(k)));
            end
        end
    end
    a_mod=a_pred;

end

```

Seccion B.37: Código de la función predistorsion() sección 2.7.2 b.

```

app.UIAxesBER.XGrid='on';
app.UIAxesBER.YGrid='on';
app.UIAxesBER.Title.String='BER Vs SNR';
app.UIAxesBER.XLabel.String='SNR (dB)';
app.UIAxesBER.YLabel.String='BER';
semilogy(app.UIAxesBER,EBNO,BER,'r');
app.UIAxesBER.NextPlot='add';
semilogy(app.UIAxesBER,EBNO,BER_OPS_SAP,'b-*');
app.UIAxesBER.YLim=[0.001 1];
app.UIAxesBER.XLim=[0 20];
legend(app.UIAxesBER,'AWGN de la señal original','AWGN CON OPS-SAP','Location','best');

```

Seccion B.40: Código empleado para la obtención de las gráficas resultantes de BER vs SNR sección 2.8.1.


```

cla(app.UIAxesCCDF,"reset");
app.UIAxesCCDF.XGrid='on';
app.UIAxesCCDF.YGrid='on';
app.UIAxesCCDF.Title.String='CCDF de la PAPR';
app.UIAxesCCDF.XLabel.String='PAPR (dB)';
app.UIAxesCCDF.YLabel.String='CCDF';
semilogy(app.UIAxesCCDF,vectorUmbral,CCDF,'r');
app.UIAxesCCDF.NextPlot='add';
semilogy(app.UIAxesCCDF,vectorUmbral,CCDF_OPS,'g-*');
app.UIAxesCCDF.NextPlot='add';
semilogy(app.UIAxesCCDF,vectorUmbral,CCDF_SAP,'m-*');
app.UIAxesCCDF.NextPlot='add';
semilogy(app.UIAxesCCDF,vectorUmbral,CCDF_OPS_SAP,'b');
legend(app.UIAxesCCDF,'Señal Original','OPS','SAP','OPS-SAP','Location','southwest');

```

Figura B.41. Código empleado para la obtención de las gráficas resultantes de la CCDF de la PAPR sección 2.8.2.

```

cla(app.UIAxesPSD,"reset");
app.UIAxesPSD.XGrid='on';
app.UIAxesPSD.YGrid='on';
app.UIAxesPSD.Title.String='PSD del símbolo OFDM';
app.UIAxesPSD.XLabel.String='Frecuencia [Hz]';
app.UIAxesPSD.YLabel.String='PSD [dB]';
plot(app.UIAxesPSD,Freq,10*log10(fftshift(txSpec_pn/max(txSpec_pn))), 'r');
app.UIAxesPSD.NextPlot='add';
plot(app.UIAxesPSD,Freq,10*log10(fftshift(txSpec_pn_OPS/max(txSpec_pn_OPS))), 'y')
app.UIAxesPSD.NextPlot='add';
plot(app.UIAxesPSD,Freq,10*log10(fftshift(txSpec_pn_SAP/max(txSpec_pn_SAP))), 'g')
app.UIAxesPSD.NextPlot='add';
plot(app.UIAxesPSD,Freq,10*log10(fftshift(txSpec_pn_OPS_SAP/max(txSpec_pn_OPS_SAP))), 'b')
legend (app.UIAxesPSD, 'Señal Original', 'OPS', 'SAP', 'OPS-SAP', 'Location', 'best');

```

Figura B.42. Código empleado para la obtención de las gráficas resultantes de la PSD sección 2.8.3.

ANEXO C Manual de Usuario

Manual de Usuario de la Aplicación

Ejecución de la Aplicación

1) Proceso para inicializar la Aplicación

Para ejecutar la aplicación se inicializa el programa Matlab, a continuación, se elige la pestaña HOME, luego se selecciona el ícono OPEN y se ubica la carpeta que contiene la aplicación.

Una vez seleccionada la carpeta que contiene la aplicación desarrollada, en la ventana *Current Folder* (Figura C.1) se elige el archivo correspondiente a la aplicación (archivo con extensión .mlapp).

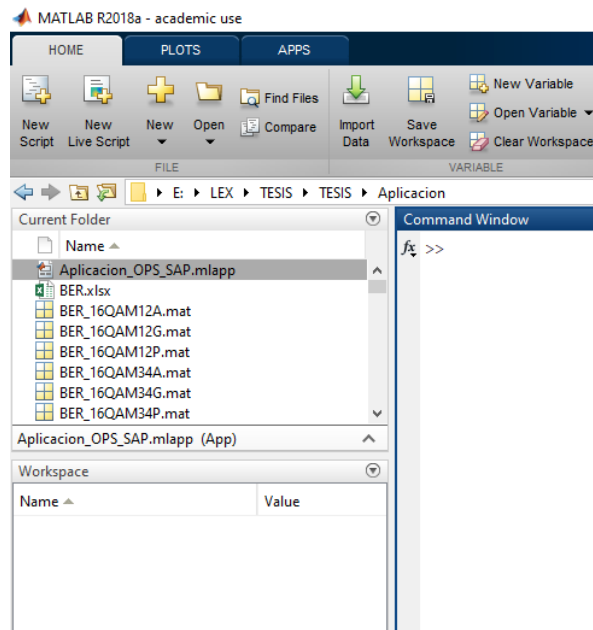


Figura C.1 Archivo con extensión .mlapp ubicado en el *Current Folder*.

Una vez ubicado el archivo perteneciente a la aplicación se ejecuta, para ello existen dos formas:

- La primera, se da doble clic directamente sobre el archivo perteneciente a la aplicación.
- La otra forma es dar clic derecho sobre el archivo de la aplicación y seleccionar Run (Figura C.2).

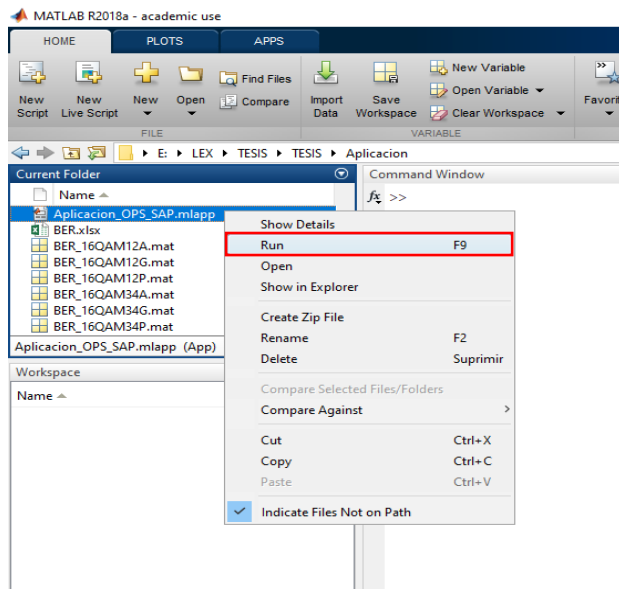


Figura C.2 Segunda forma de ejecución de la aplicación

2) Ejecución de la Aplicación

Cuando la aplicación se encuentra activa, aparece la ventana principal que está formada por tres secciones. Cada sección es diseñada con un propósito específico, a continuación, se describe cada sección:

Ingreso de Datos: En esta sección se ingresan los argumentos necesarios para llevar a cabo la simulación; entre los argumentos de entrada se encuentran los siguientes: estándar, esquema de modulación, tasa de codificación, canal inalámbrico, número de tramas PPDU PLCP que se van enviar y finalmente el botón encargado de ejecutar la simulación (SIMULAR), toda esta sección se observa en la Figura C.3.

PARÁMETROS DE SIMULACIÓN

Enmiendas

IEEE 802.11a
 IEEE 802.11g
 IEEE 802.11p

MODULACIÓN: BPSK, QPSK, 16 QAM, 64 QAM

Tasa de Codificación: 1/2

Importante ! Los esquemas de modulación BPSK, QPSK y 16-QAM tienen una tasa de codificación de 1/2 y 3/4. Mientras tanto el esquema de modulación 64-QAM

Canal Inalámbrico: AWGN, Rician, RAYLEIGH

Paquetes enviados: 0

Advertencias!

Figura C.3 Ingreso de Datos

Presentación de Resultados: Esta sección se utiliza para la presentación de los resultados obtenidos al finalizar la simulación, los resultados son presentados a través de diferentes curvas debidamente identificadas. Para la visualización de las distintas curvas basta con seleccionar las pestañas ubicadas en la parte superior. La sección de resultados se visualiza en la Figura C.4.

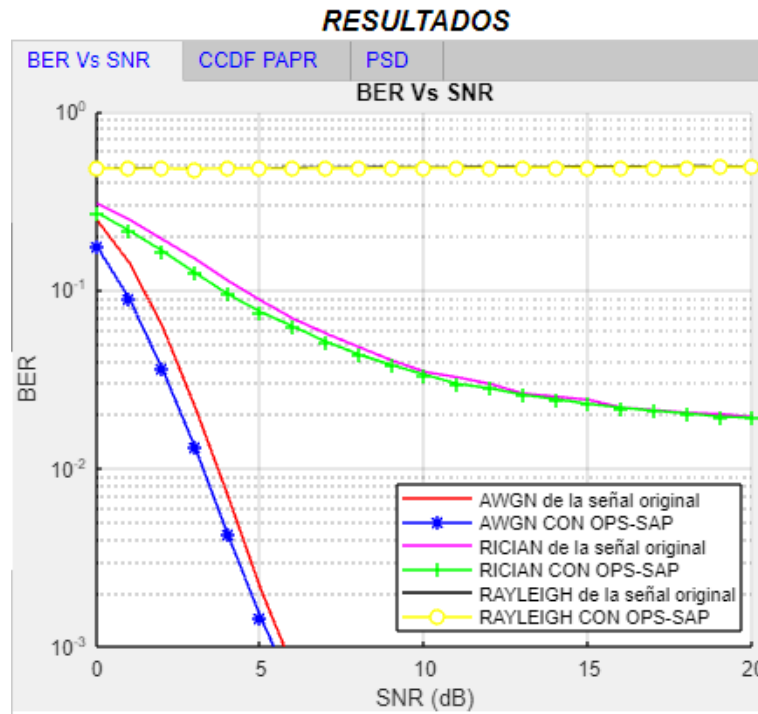


Figura C.4 Sección de Resultados.

Información de los procesos: Esta sección de la aplicación se emplea para describir los procesos, técnicas y parámetros que se utilizan para llevar a cabo la simulación con el propósito de informar al usuario para que comprenda de mejor manera el entorno de la simulación, toda esta sección se observa en la Figura C.5.

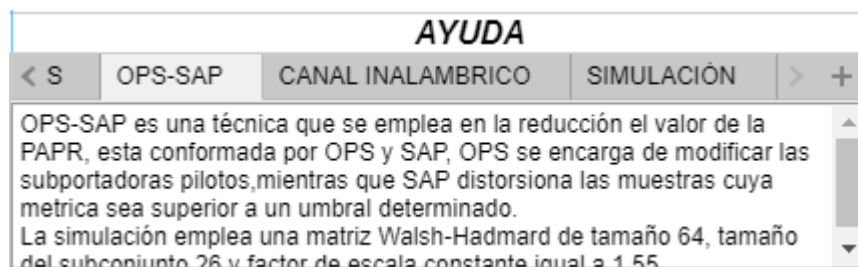


Figura C.5. Sección de información al usuario.

Una vez identificadas cada una de las secciones que conforman la aplicación, a continuación, se indica el procedimiento para la ejecución correcta de la simulación.

- 1) Se elige el estándar.
- 2) Se escoge el esquema de modulación junto con la tasa de codificación, para cada esquema de modulación le corresponde su respectiva tasa de codificación como se indica en la Tabla C.1.

Tabla C.1 Esquemas de Modulación y tasas de codificación

| Esquema de modulación | Tasa de codificación |
|-----------------------|----------------------|
| BPSK | $r = 1/2$ |
| BPSK | $r = 3/4$ |
| QPSK | $r = 1/2$ |
| QPSK | $r = 3/4$ |
| 16QAM | $r = 1/2$ |
| 16QAM | $r = 3/4$ |
| 64QAM | $r = 2/3$ |
| 64QAM | $r = 3/4$ |

- 3) El siguiente paso es la elección del canal inalámbrico ya sea con ruido AWGN, desvanecimiento Rician o Rayleigh. Se puede seleccionar indistintamente cualquiera de las tres opciones y combinarlas de diferentes formas. En el caso de no seleccionarse ninguna opción la simulación por defecto se realizará sobre un canal inalámbrico con ruido AWGN.

Los canales con desvanecimiento Rayleigh y Rician son modelados en base a los valores de PDP (*Power Delay Profile*) correspondientes a cada estándar, estos valores se encuentran definidos acorde a cada estándar.

- 4) A continuación, se ingresa el número de tramas PPDU PLCP que se van a enviar por el sistema de comunicación Ad-Hoc, el número ingresado debe ser entero. Un número adecuado debe ser superior a 10000 tramas para obtener curvas con trazos bien definidos, pero este número de tramas toma un tiempo considerable (alrededor de 60 minutos) para finalizar la simulación, sin embargo, realizar un envío de 1000 tramas brinda resultados con curvas bastante buenas y que son obtenidas en mucho menor tiempo (alrededor de 10 minutos).
- 5) Finalmente, se presiona el botón "SIMULAR" y la simulación se ejecuta.
- 6) Una vez que finaliza la simulación se presentan los resultados a través de curvas. Las curvas que se presentan son: BER vs SNR, CCDF de la PAPR y PSD.

Las gráficas de la BER vs SNR contienen las curvas de la señal original (sin técnica) y las curvas de la señal con la técnica OPS-SAP.

Las gráficas de la CCDF de la PAPR presentan las curvas de la señal original (sin técnica), con la técnica OPS, SAP y OPS-SAP.

Las gráficas de la PSD presentan las curvas de la señal original (sin técnica), con la técnica OPS, SAP y OPS-SAP.

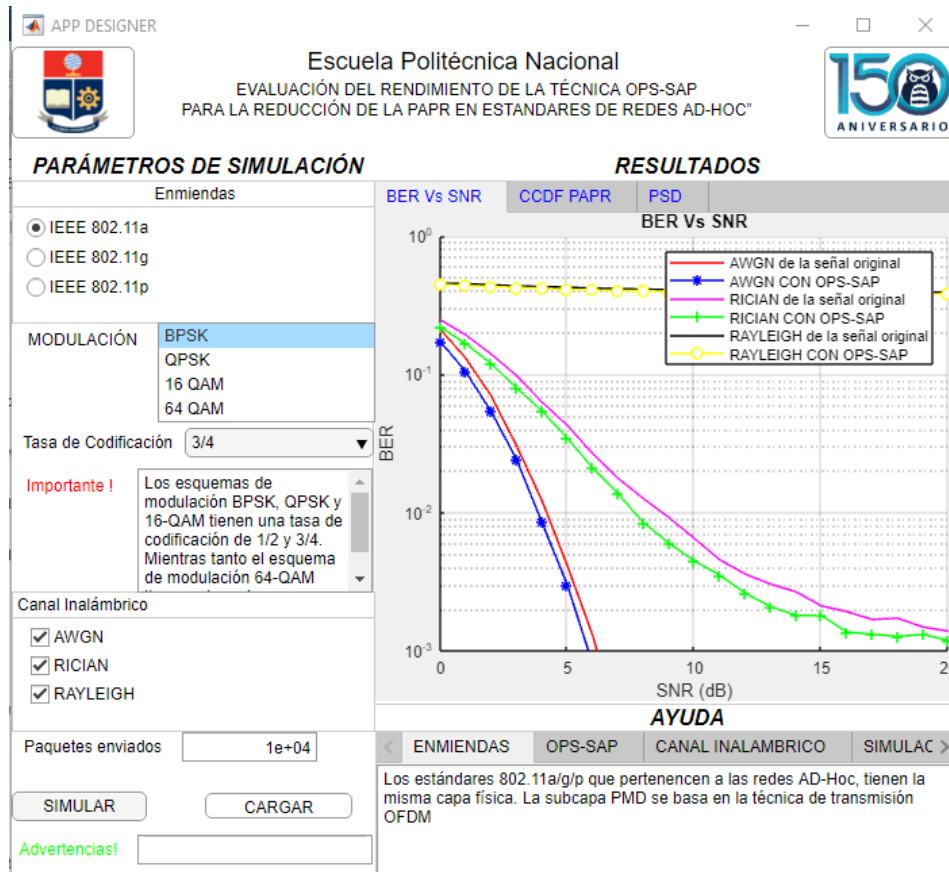


Figura C.6 Simulación realizada exitosamente.

3) Acceso a datos previamente almacenados

Para reducir el tiempo en la obtención de los resultados, se creó un botón adicional denominado "CARGAR" este botón presenta los resultados previamente almacenados con el propósito de visualizar las curvas bien definidas sin tener que esperar mucho tiempo, cabe señalar que estas curvas fueron simuladas con todos los parámetros preestablecidos.

Para observar los resultados se realiza el siguiente procedimiento:

- En primer lugar, se elige el estándar.

- A continuación, el esquema de modulación y tasa de codificación deben ser seleccionados.
- La simulación por defecto muestra los resultados para canales con ruido AWGN, desvanecimiento Rayleigh y Rician. Los resultados obtenidos corresponden al resultado del envío de 10000 tramas PDU PLCP.
- Una vez seleccionados todos los parámetros se presiona el botón “CARGAR”.
- Los resultados se van a presentar en la sección de resultados y van a contener todas las curvas mencionadas en el inciso 6 de la sección anterior.

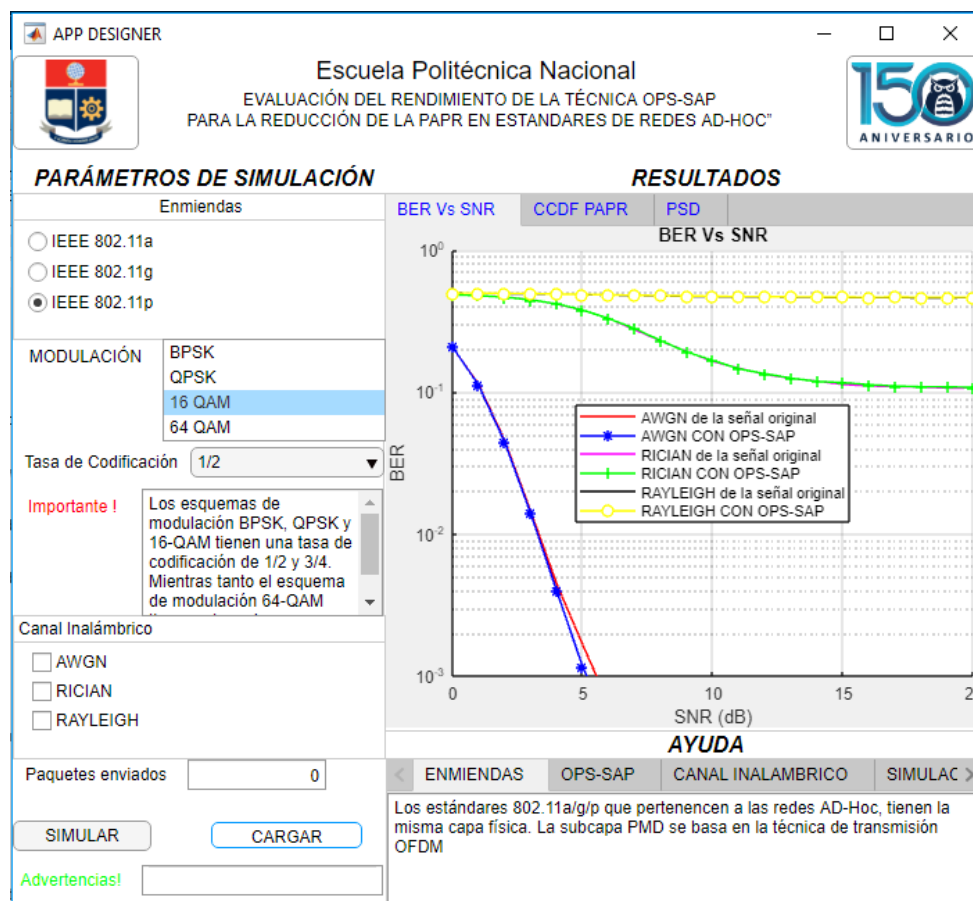


Figura C.7 Presentación de resultados con datos previamente almacenados.

4) Acceso al código de la aplicación

Para ingresar al código de la aplicación es necesario realizar los siguientes pasos:

- Abrir el programa Matlab, seleccionar el ícono OPEN. luego buscar la carpeta que contiene la aplicación.
- Una vez seleccionada la carpeta que contiene la aplicación desarrollada, en la ventana *Current Folder* se ubica el archivo correspondiente a la aplicación (archivo con extensión .mlapp).

- Ubicado el archivo perteneciente a la aplicación se procede a abrirlo, para ello se da clic derecho sobre el archivo de la aplicación y se elige OPEN (Figura C.8).
- En el momento que se abre la aplicación aparecen las ventanas pertenecientes al *Desing View* y *Code View*. En la ventana perteneciente a *Desing View* se observan todos los objetos que se utilizaron para el diseño (parte visual) de la aplicación. En la ventana perteneciente a *Code View* se accede a la codificación realizada para llevar a cabo el funcionamiento de la aplicación. Para visualizar los Callback's y funciones implementadas es necesario ubicarse en la ventana del Code Browser. El Callback principal encargado de toda la lógica de la aplicación se encuentra en el botón simular (BSimulacionPushed).

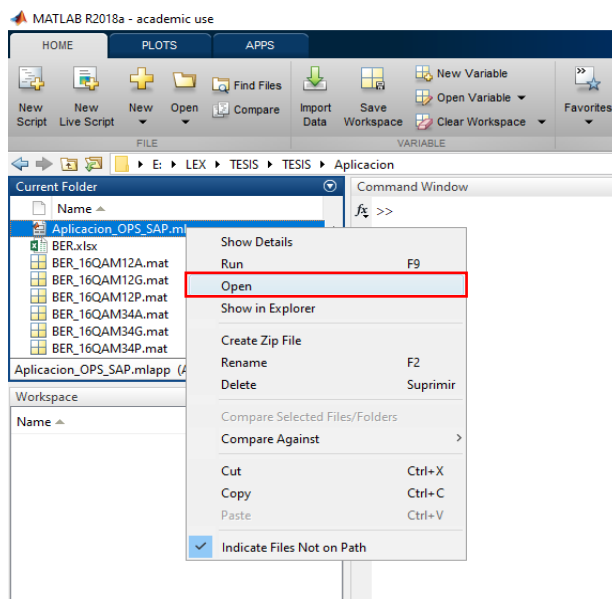


Figura C.8 Proceso para abrir la aplicación y visualizar el código del mismo.

- Para finalizar, los archivos que se encuentran junto con el archivo de la aplicación en la carpeta forman parte de la aplicación y son de ayuda para la ejecución de la misma.

ORDEN DE EMPASTADO