



UNIVERSIDAD NACIONAL DE COLOMBIA

“Análisis de Desempeño de un Sistema MIMO-OFDM con Predicción de Canal”

Catalina Muñoz Morales

Universidad Nacional de Colombia

Facultad de Ingeniería

Departamento de Ingeniería de Sistemas e Industrial

Bogotá DC., Colombia

2013

Análisis de Desempeño de un Sistema MIMO-OFDM con Predicción de Canal

Catalina Muñoz Morales

Trabajo de investigación presentado como requisito parcial para optar al título de:
Magister en Ingeniería - Telecomunicaciones

Director:

Ph.D., Sebastián Eslava G.

Línea de Investigación:

Redes y Sistemas de Telecomunicaciones.

Grupo de Investigación:

Grupo de Microelectrónica de la Universidad Nacional de Colombia (Gmun)

Universidad Nacional de Colombia

Facultad de Ingeniería

Bogotá DC., Colombia

2013

Resumen

Las comunicaciones inalámbricas en que el canal de transmisión inalámbrico se define por efectos de dispersión por movimiento y obstáculos físicos entre transmisor y receptor, son un claro ejemplo de los retos que se enfrentan para lograr una comunicación efectiva mediante un ambiente ruidoso. La demanda de múltiples servicios de telecomunicaciones, como transmisión de voz, video y datos, ha hecho que la capacidad de transmisión y recepción de los sistemas de comunicaciones aumente para lograr grandes tasas de transmisión de datos con baja cantidad de errores recibidos, que hagan la comunicación confiable y utilizando el mínimo de recursos como espectro radioeléctrico y energía (potencia).

Los sistemas de cuarta generación (4G) han llegado en los últimos años, con diversas tecnologías, para cumplir con los requerimientos impuestos; en estos sistemas se utilizan técnicas como el uso de múltiples antenas (*Multiple Input – Multiple Output*, MIMO), modulación (*Orthogonal Frequency Division Multiplexing*, OFDM), codificación, estimación y predicción de canal para adaptar el sistema a las condiciones de este, y así lograr el objetivo de obtener una transmisión confiable.

En este trabajo se evalúan diferentes técnicas lineales y no lineales de predicción del canal inalámbrico, cuyas muestras con distribución Rayleigh han sido generadas y aplicadas sobre un sistema MIMO-OFDM.

El sistema MIMO-OFDM se desarrolló con una descripción a nivel de sistemas usando el lenguaje de *SystemC* para obtener medidas de desempeño del mismo sobre el canal de comunicaciones, al cual se adapta utilizando la relación señal a ruido del canal estimado y predicho. La adaptación al canal o enlace de comunicaciones (*Link Adaptation*) se obtuvo en relación a esquemas de modulación para aumentar o disminuir la tasa de transmisión y de errores de bit (BER – *Bit Error Rate*) de acuerdo con las condiciones de ruido del canal.

El análisis de desempeño del sistema y las técnicas de predicción de canal fue determinado por métricas de cantidad de errores, precisión de la predicción realizada,

tasa de transmisión, latencia, complejidad computacional y utilización de memoria. A partir de allí se hicieron comparaciones de desempeño entre técnicas lineales y no lineales que permitieron establecer la viabilidad de la implementación de estas en un sistema de comunicaciones real.

La investigación fue llevada a cabo usando una descripción del sistema a nivel de capa física cuya evaluación en banda base da la posibilidad de validar el comportamiento del mismo con respecto al procesamiento de señales generadas en forma de trama de datos binaria con distribución uniforme.

Los resultados obtenidos muestran que los algoritmos de predicción de canal generan un aumento de aproximadamente el 7% mínimo en la latencia del sistema MIMO-OFDM y además existe una relación entre el funcionamiento y la complejidad computacional de los algoritmos estudiados.

Palabras clave: MIMO, OFDM, Predicción de canal, SystemC, Modulación adaptativa, Codificación adaptativa.

Abstract

Wireless communications, where the channel is defined by spread effects and physical obstacles between transmitter and receiver, are a clear example of the challenges faced in order to have an effective communication in noisy environments. Demand for multiple communication systems, such as voice, video and data, have made transmission and reception capabilities of nowadays communication systems to increase in order to accomplish high transmission and low reception error rates that makes wireless communications more reliable with the use of minimum radio spectrum and power.

Fourth generation (4G) systems have come in the recent years with diverse technologies, to fit requirements imposed; in this systems multiple techniques such as multiple antennas at transmitter and receiver, modulation, coding channel estimation and prediction are used to adjust the system to channel conditions.

In this work different techniques are evaluated for linear and non-linear prediction of n time spaces of the wireless channel, whose samples have been generated and applied to a MIMO-OFDM system.

The MIMO-OFDM system has been developed on a system – level description with *SystemC* in order to enable performance measures of the communication system adaptation over a noisy wireless channel using Signal to Noise ratio (SNR) measured of the estimated and/or predicted channel. Link adaptation was made around different modulation schemes to increase or decrease transmission data rates and Bit Error Rate (BER) according to channel noise conditions.

Performance analysis for the system and its channel prediction techniques was determined by metrics of number of errors, prediction error, transmission rate, latency, computational complexity and memory usage. Comparison between linear and non-linear prediction techniques in order to establish the viability of implementation of such techniques in real communication systems was also performed as one of the main goals for this work.

Research on this topic was performed using a system level description of the physical layer of the MIMO-OFDM system whose evaluation in base band gives the possibility to validate system behaviour in relation to signal processing of randomly generated bit frames with a uniform distribution.

Results show that channel prediction algorithms increase latency of the MIMO-OFDM system in about 7%, and also that there is a relationship between performance and computational complexity for the studied algorithms.

Keywords: MIMO, OFDM, Channel Prediction, SystemC, Adaptive modulation, Adaptive coding.

Índice

	Pág.
Resumen	V
Lista de figuras	XI
Lista de tablas	XIII
Lista de Símbolos y abreviaturas	15
1. Introducción	16
2. Capa física de Sistema MIMO-OFDM	19
2.1 Sistema Transmisor.....	19
2.1.1 Codificación de canal – <i>Channel Coding</i>	20
2.1.2 Intercalador de Bits	22
2.1.3 Esquemas de Modulación.....	22
2.1.4 <i>Codificación Espacio - Tiempo: Código Alamouti</i>	25
2.1.5 Multiplexación Ortogonal por División de Frecuencia - OFDM	26
2.2 Canal de Comunicación	28
2.3 Sistema Receptor	29
2.3.1 Decodificador MIMO	30
2.3.2 Estimación de Canal	31
2.3.3 Decodificación Convolutiva	32
3. Diseño del Sistema	35
3.1 Configuración de sistema MIMO-OFDM	35
3.1.1 Etapa 1: Randomizador de bits, Intercalador de bits, Codificación Convolutiva y Algoritmo Viterbi, Modulación M-QAM.	38
3.1.2 Etapa 2: Codificación MIMO Alamouti y Decodificación Zero Forcing.....	40
3.1.3 Etapa 3: Modulación y Demodulación <i>OFDM</i>	41
3.1.4 Modelamiento de canal inalámbrico	41
3.2 Diseño SystemC.....	42
3.2.1 Parámetros de diseño SystemC	43
3.2.2 Particionamiento Hardware/Software	47
4. Técnicas de Predicción de Canal	49
4.1 Algoritmos de predicción Lineal.....	49
4.1.1 Predicción Lineal con Modelo Autoregresivo AR(<i>n</i>):	49
4.1.2 Filtro Kalman Con Modelo Autoregresivo AR	50
4.2 Algoritmos de Predicción No Lineal.....	51
4.2.1 Redes Neuronales Con Retrasos de Tiempo	52

4.2.2	Redes Neuronales Recurrentes.....	54
5.	Resultados	56
5.1	Validación del sistema MIMO-OFDM	57
5.1.1	Comportamiento del canal inalámbrico.....	58
5.1.2	Comportamiento del sistema con modulación OFDM	59
5.1.3	Comportamiento del Sistema para Diferentes Esquemas de Modulación.....	60
5.2	Configuración de Algoritmos de predicción de Canal.....	65
5.3	Desempeño de Esquemas de Predicción	67
5.4	Desempeño de sistema de comunicaciones MIMO-OFDM	71
5.4.1	BER Según Medición SNR para Estimación y Predicción de Canal:	71
5.5	Análisis de Desempeño del Sistema Digital	75
5.5.1	Tasa de transmisión de datos:.....	75
5.5.2	Latencia de sistema:	76
5.5.3	Complejidad computacional de cada algoritmo de predicción:.....	77
5.5.4	Utilización de memoria de cada algoritmo de predicción:.....	78
5.6	Comparación final entre esquemas de predicción de canal.....	79
6.	Conclusiones y Trabajo Futuro	81
6.1	Conclusiones	81
6.2	Trabajo Futuro	82
	Bibliografía	83

Lista de figuras

	Pág.
Figura 2-1. Sistema de transmisión básico.....	20
Figura 2-2 Codificador Convolutacional.....	21
Figura 2-3 Máquina de estados finita de un Codificador Convolutacional.....	22
Figura 2-4 Señal FSK Modulada.....	23
Figura 2-5 Señal PSK modulada.	24
Figura 2-6. Diagrama de Constelación para 16QAM Rectangular.	25
Figura 2-7 Estructura de símbolo OFDM.	27
Figura 2-8 Estructura de símbolos piloto. a) <i>Block type</i> , b) <i>Comb type</i> , c) <i>Lattice Type</i> . .	27
Figura 2-9 Múltiples trayectos de una señal en un sistema Transmisor – Receptor.	28
Figura 2-10 Modelo de canal MIMO.	29
Figura 2-11. Sistema de Recepción Básico.....	30
Figura 2-12 Estructura de símbolos piloto para múltiples antenas Tx.....	31
Figura 2-13 Diagrama de decodificación convolutacional para $R=1/2$	33
Figura 2-14. Transmisor y Receptor de Sistema MIMO-OFDM.....	34
Figura 3-1 Estructura Sistema MIMO-OFDM.....	37
Figura 3-2 Estructura <i>Randomiser/Derandomiser</i>	38
Figura 3-3 Estructura <i>Convolutional Coder</i>	39
Figura 3-4 Estructura <i>Interleaver/Deinterleaver</i>	39
Figura 3-5 Estructura a) <i>M-QAM modulator</i> , b) <i>M-QAM demodulator</i>	40
Figura 3-6 Estructura decodificador <i>MIMO</i>	41
Figura 3-7 Estructura <i>OFDM Modulador/Demodulador</i>	41
Figura 3-8 Características de SystemC. [12].....	43
Figura 3-9 Flujo de tareas de sistema con arquitectura MIMO-OFDM.	44
Figura 3-10 Diseño SystemC de sistema MIMO-OFDM.....	47
Figura 4-1 Red neuronal <i>Feedforward</i>	52
Figura 4-2 Diagrama interno de una neurona.....	53
Figura 4-3 Red Neuronal con Retraso de tiempo.	53
Figura 4-4 Red Neuronal Recurrente.....	54
Figura 5-1. Sistema MIMO-OFDM con predicción de canal en el receptor.	57
Figura 5-2 Canal inalámbrico Rayleigh según modelo Clarke.....	58
Figura 5-3 Potencia canal inalámbrico modelo Rayleigh.....	59
Figura 5-4 Distribución de canal inalámbrico.....	59
Figura 5-5 Potencia espectral de señal OFDM antes y después de transmisión.	60
Figura 5-6 Grafica de dispersión de símbolos con $E_b/N_0=0$ dB; a) BPSK, b) 16-QAM, c) 64-QAM, d) 256-QAM.	61
Figura 5-7 Grafica de BER para modulación BPSK con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).....	62

Figura 5-8 Grafica de BER para modulación 16-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).....	62
Figura 5-9 Grafica de BER para modulación 64-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).....	63
Figura 5-10 Grafica de BER para modulación 256-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).....	63
Figura 5-11 Predicción lineal con modelo AR.....	68
Figura 5-12 Predicción con filtro kalman.....	69
Figura 5-13 Predicción con red neuronal con n retrasos en la entrada.....	69
Figura 5-14 Predicción Con Red neuronal recurrente.....	70
Figura 5-15 E_b / N_0 Vs. SNR de algoritmos de predicción.....	72
Figura 5-16 E_b/N_0 Vs. BER de esquemas de algoritmos de predicción.....	73

Lista de tablas

	Pág.
Tabla 2-1 Sub-portadoras en OFDM. [7]	28
Tabla 2-2 Sucesión de estados para codificador convolucional $R=1/2$	33
Tabla 2-3 Posibles salidas según estado estados para codificador convolucional $R=1/2$	33
Tabla 3-1 Parámetros de sistema MIMO-OFDM.	36
Tabla 3-2 Descripción tareas específicas.	45
Tabla 3-3 Rendimiento de funciones con trama de datos = 384, $R = 1/2$, $M = 16$, $N = 25646$	
Tabla 4-1 algoritmo pseudocódigo de predicción lineal con modelo AR(n).	50
Tabla 4-2 Algoritmo Pseudo-Código filtro Kalman.	51
Tabla 5-1 Algoritmo de adaptación de canal para sistema transmisor.	64
Tabla 5-2 Error MSE para Predictor con modelo AR.	66
Tabla 5-3 Error MSE para Filtro Kalman.	66
Tabla 5-4 Error MSE para estructuras de Red Neuronal con Retraso en Entrada.	67
Tabla 5-5 Error MSE para estructuras de Red Neuronal Recurrente.	67
Tabla 5-6 Mediciones de BER y SNR para configuraciones del sistema $fdTs = 0.1$	74
Tabla 5-7 Mediciones de BER y SNR para configuraciones del sistema $fdTs = 0.3$	74
Tabla 5-8 Tasa de transmisión de sistema MIMO-OFDM con predicción de canal. $fdTs = 0.1$ para 100 espacios de tiempo.	76
Tabla 5-9 Tasa de transmisión de sistema MIMO-OFDM con predicción de canal. $fdTs = 0.3$ para 100 espacios de tiempo.	76
Tabla 5-10 Latencia del sistema MIMO-OFDM con canal estimado y predicho.	77
Tabla 5-11 Complejidad computacional para algoritmos de predicción de canal.	77
Tabla 5-12. Utilización de memoria para algoritmos de predicción de canal.	78
Tabla 5-13 comparación algoritmos de predicción en sistema MIM-OFDM.	79

Lista de Símbolos y abreviaturas

MIMO	Multiple Entrada – Multiple Salida, (Multiple Input – Multiple output).
OFDM	Multiplexación por división de frecuencias ortogonales (Orthogonal Frequency Division Multiplexing).
Tx	Transmisor.
Rx	Recepción.
h_{mn}	Trayecto de canal de antena m a antena n .
Z^{-1}	Retraso de un espacio de tiempo de una señal.
BER	Tasa de Error de Bits (Bit Error Rate).
SNR	Relación Señal a Ruido (Signal to Noise Ratio).
M-QAM	Modulación de Amplitud en Cuadratura (Quadrature Amplitude Modulation).
BPSK	Modulación por Desplazamiento de Fase Binario (Binary Phase Shift Keying).
4G	Sistemas móviles de cuarta generación.
PHY	Abreviación de la capa física del modelo OSI.
FFT	Transformada Rápida de Fourier (Fast Fourier Transform).
STBC	Código de bloque Espacio – Tiempo (Space Time Block Code).
AR(n)	Modelo Autoregresivo de orden n .
Eb/No	Relación energía por bit / densidad espectral de potencia de ruido (the energy per bit to noise power spectral density ratio).
ACI	Interferencia de Canal Adyacente (<i>Adjacent-channel interference</i>).

1. Introducción

La demanda de servicios de telecomunicaciones, como transmisión de voz, video y datos, ha hecho que la capacidad de transmisión y recepción de los sistemas de comunicaciones deba aumentar, de tal forma que se logren grandes tasas de transmisión con baja cantidad de errores recibidos, para que la comunicación sea confiable y utilice el mínimo de recursos como espectro radioeléctrico y energía (potencia). La transmisión de datos en sistemas de comunicación inalámbrica enfrenta grandes problemas relacionados con errores en la transmisión, la tasa de transmisión de datos, y rendimiento del sistema, dado el ambiente mediante el cual ocurre la comunicación. En ese sentido, nuevas técnicas de codificación y modulación, han sido desarrolladas con el objetivo de incrementar el rendimiento del sistema para que de esa forma la tasa de envío de información aumente y a la vez disminuya la cantidad de errores ocurridos en la comunicación.

Los sistemas de cuarta generación (4G), sucesores de las tecnologías 2G y 3G, se refieren a sistemas de comunicaciones móviles con estándares LTE (*Long Term Evolution*) [61], Wimax [62] y SAE (*Service Architecture Evolution*) [63]; En estos sistemas existen dos tecnologías que han sido desarrolladas en su capa física, las cuales son: (1) *Multiple Input – Multiple Output* (MIMO) en la cual se utiliza una o más antenas en el transmisor y/o el receptor para incrementar la tasa de transmisión sin que sea necesario incrementar el ancho de banda del sistema [24], y (2) *Orthogonal Frequency División Multiplexing* (OFDM) la cual es una técnica de modulación que asegura altas tasas de datos al convertir un canal de comunicación en múltiples sub-canales ortogonales. MIMO y OFDM combinadas con estimación de canal, permiten incrementar la capacidad del sistema, esto significa transmisiones más confiables y eficientes en el uso de ancho de banda [24] [25].

La estimación de canal se refiere a la adquisición de propiedades del canal, como niveles de dispersión y atenuación, y es utilizada para adquirir información precisa que permita al sistema adaptarse a las condiciones de transmisión; esta información es obtenida mediante el análisis de la señal recibida y enviada de vuelta al transmisor para que sea utilizada por este.

Una estimación perfecta del canal resulta que, además de ser imposible en la realidad dadas las condiciones propias de canal, no es suficiente al momento de hacer una adaptación del sistema al enlace por el cual se transmite, ya que esta no es obtenida a tiempo para ser utilizada en el receptor y transmisor para su adaptación al medio, lo cual es uno de los objetivos principales de un sistema MIMO-OFDM cuyas técnicas pueden ser adaptables en sus parámetros de forma que se alcance una estabilidad en términos de relación señal a ruido y tasa de errores. La adaptación de enlace puede ser lograda de diferentes formas, que en su mayoría, utilizan esquemas de modulación como BPSK, QPSK, y MQAM y técnicas de codificación convolucionales para ajustar la transmisión a las posibles condiciones difíciles al momento de la transmisión; esto se hace con el

objetivo de lograr una potencia de consumo, o la tasa de error de bit específicos. Dado que la adaptación de enlace es lograda mediante el envío de información de vuelta al transmisor sobre las condiciones del canal a través del mismo canal u otro dedicado, se ha encontrado que dicha información llega al transmisor con retrasos y errores que hacen que el sistema no se adapte de forma adecuada o definitivamente no se logre adaptar.

Para solucionar el problema anteriormente mencionado, técnicas de predicción de canal han sido desarrolladas; estas técnicas consisten en la recolección de valores del canal dados a lo largo del tiempo y el procesamiento de estos para predecir n espacios de tiempo de la misma señal para que esta logre llegar al sistema transmisor y pueda ser utilizada para adaptación de enlace.

La investigación alrededor del canal inalámbrico y las técnicas para predecirlo han estado enfocadas inicialmente en encontrar modelos que permita simular las condiciones del canal y evaluar el rendimiento de sistemas de comunicaciones presentados allí [31] [32]. De acuerdo con las características del ambiente, el canal inalámbrico puede ser modelado usando diferentes distribuciones estadísticas; por ejemplo, canales sin línea de vista entre transmisor (Tx) y receptor (Rx) puede ser descrito con una distribución Rayleigh [33], y de la misma manera, un canal con línea de vista se describe con una distribución Rician [64]. Otras características que se toman en cuenta para modelar el canal inalámbrico son el nivel de desvanecimiento, que es la atenuación que afecta la señal transmitida, la dispersión por efecto Doppler y la cantidad de trayectos de canal. Para este trabajo se consideró un canal Rayleigh no selectivo en frecuencia, dado que este presenta algunas de las características más difíciles.

Trabajo previo en predicción de canal se basa en el desarrollo de algoritmos lineales que usan modelos Autoregresivos del canal [26], uso de filtros kalman [27] y algunos más complejos hacen uso de algoritmos no lineales [28][29].

La importancia de esta investigación radica en la necesidad de saber si en un ambiente ruidoso y con altos niveles de interferencia las técnicas de predicción de canal lineales y no lineales tomadas en cuenta son útiles, y si es así, cuál es la mejor técnica de predicción. Este análisis se lleva a cabo utilizando métricas de desempeño del sistema de comunicaciones MIMO-OFDM propuesto en diferentes escenarios, utilizando técnicas de predicción que se desarrollaran como parte del sistema; las técnicas utilizadas se basan en modelo Autorregresivo (AR), filtro kalman, y redes neuronales con retrasos en entrada y recurrente. Las métricas utilizadas están comprendidas dentro de dos perspectivas de análisis: por un lado, se observa el desempeño del sistema con respecto al procesamiento de señales, sobre el cual se mide el error encontrado tanto en la predicción del canal como en la señal transmitida; por otro lado, el análisis se realiza desde el punto de vista del sistema digital, la capacidad de procesamiento y la complejidad de este, para a partir de allí determinar medidas de complejidad computacional, latencia del sistema y acceso a elementos de almacenamiento. Todos estos análisis son obtenidos desde el marco de un sistema MIMO-OFDM diseñado según un estudio bibliográfico e implementado utilizando lenguaje de descripción de sistemas; esto se ha realizado con el objetivo de lograr una simulación realista en la que se

combinan la disposición de elementos hardware y software implementables y funcionales de forma concurrente dentro de un espacio de tiempo determinado.

En este trabajo se realiza la comparación de esquemas lineales y no lineales de predicción de canal inalámbrico sobre un sistema de comunicaciones MIMO-OFDM; inicialmente en la sección 2 se desarrolla un estudio bibliográfico sobre sistemas MIMO-OFDM y sus características a nivel de capa física (PHY), para posteriormente ser implementado y validado a nivel de simulación como es mostrado en la sección 3; para esto se hizo uso de la descripción de alto nivel de sistemas digitales, usando el lenguaje *SystemC*, una librería para C++ que presenta ventajas al permitir co-diseño hardware/software y la posibilidad de hacer una descripción funcional y evaluarla usando métricas de desempeño medibles dentro del sistema. Una vez descrito el sistema, técnicas de predicción de canal, lineales y no lineales, fueron implementadas e integradas al sistema con el objetivo de evaluar este en términos de error de predicción, tasa de errores, tasa de transmisión, latencia, complejidad computacional y utilización de memoria; los resultados y el análisis comparativo entre esquemas de predicción se muestra en la sección 5 para finalmente presentar conclusiones y posibles trabajos futuros en la sección 6.

2. Capa física de Sistema MIMO-OFDM

En telecomunicaciones, *Multiple Input Multiple Output* se refiere al uso de más de una antena en el transmisor y/o en el receptor. Esta tecnología se ha convertido en una de las más importantes para las comunicaciones inalámbricas, ya que permite incrementar de forma significativa la tasa de transmisión de datos sin que sea necesario aumentar el ancho de banda del sistema, y la codificación y decodificación usada también permite disminuir la tasa de error de bits cuando se tiene información del estado del canal de comunicaciones (*Channel State Information - CSI*), que se refiere a la estimación de las características actuales del canal por el cual se transmite tomando en cuenta características como desvanecimiento y multi-trayecto (Estimación de canal).

Desde 1966, *Orthogonal Frequency Division Multiplexing* es uno de las más utilizadas técnicas de modulación [65], ya que permite una transmisión de datos más segura al convertir un canal inalámbrico de condiciones altamente variables, en una serie de subcanales planos y ortogonales entre sí.

MIMO y OFDM combinadas con una estimación de canal apropiada para las características de la transmisión permite incrementar la capacidad de un sistema de comunicaciones inalámbricas, lo que se traduce en una mayor eficiencia en la transmisión de datos y el uso de ancho de banda con respecto a otros sistemas de telecomunicaciones, convirtiéndose en la técnica de comunicación utilizada en sistemas de cuarta generación como Wimax y LTE. [1][2]

Este capítulo presenta la estructura básica de la capa física (PHY) de un sistema de cuarta generación, y se hace una introducción a cada una de las técnicas de procesamiento de datos en transmisión y recepción, utilizada en un sistema MIMO-OFDM.

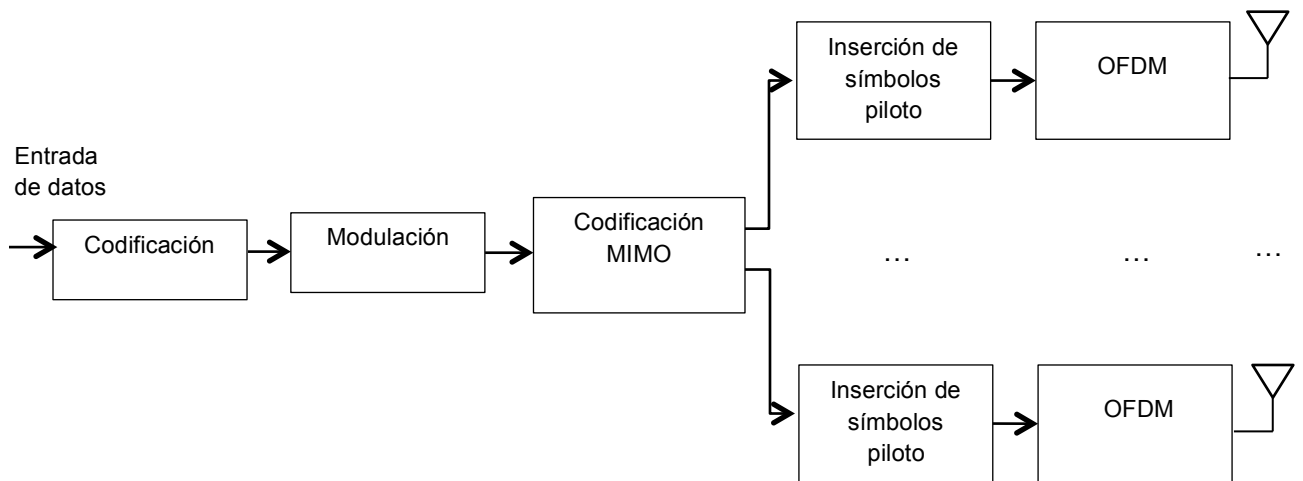
2.1 Sistema Transmisor

Un sistema de comunicaciones básico incluye un módulo de transmisión en el cual se procesa la información para ser enviada a un módulo receptor encontrado a una cierta distancia. El transmisor utiliza técnicas como codificación, modulación y multiplicación,

que hacen la señal lo suficientemente compleja para reducir la tasa de error de bit (BER)¹.

En un sistema MIMO-OFDM, adicional al procesamiento de datos, que se realiza mediante técnicas de modulación y codificación, es de gran importancia tomar en consideración el canal por el cual se envían los datos; esta información puede ser obtenida estudiando la señal que llega al receptor. Para este propósito, el transmisor una serie de símbolos piloto que se integran a la trama de datos que conforma la señal enviada por el transmisor. La figura 2.1 muestra un sistema de transmisión básico.

Figura 2-1. Sistema de transmisión básico.



2.1.1 Codificación de canal – *Channel Coding*

La codificación de canal o corrección de errores hacia adelante (*Forward Error Correction* - FEC) es una técnica de codificación utilizada con el objetivo de controlar la cantidad de errores ocurridos durante una transmisión a través de un canal ruidoso. Típicamente, un codificador FEC genera una cierta redundancia en los datos, de tal forma que sea posible saber si ha ocurrido un error en la trama enviada para que sea corregida en el receptor.

La redundancia añadida durante la etapa de codificación está dada por algoritmos determinados dependiendo de la complejidad necesaria. Existen dos grandes categorías de métodos de codificación [34]: Códigos de bloque (*Block Codes*) y Códigos convolucionales (*Convolutional Codes*). También puede ser encontrada en la literatura una serie de códigos híbridos en los que se juntan las mejores características de los métodos anteriormente mencionados. Dada la complejidad y robustez de los códigos

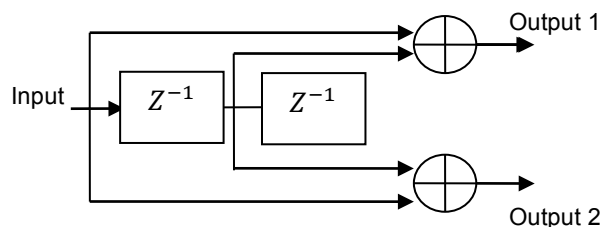
¹ La Tasa de Error de Bit o BER indica la cantidad de errores obtenidos durante la transmisión de una trama de datos binaria. Está definida como $BER = \frac{\text{No.de Errores}}{\text{Numero de Bits transmitidos}}$.

convolucionales simples, estos son altamente aplicados a comunicaciones móviles y satelitales. [30]

Los códigos de bloque codifican la señal de entrada en paquetes de tamaño fijo como los códigos hamming [35].

En la codificación convolucional, la salida actual depende no solo de la entrada actual, sino también de un número n de entradas anteriores, conservadas de la forma como se muestra en la figura 2.2.

Figura 2-2 Codificador Convolucional.



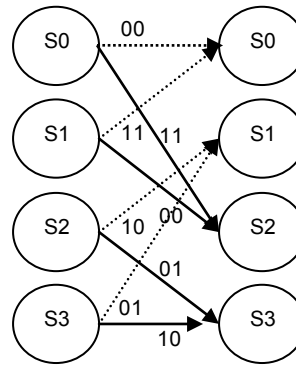
Típicamente, un Codificador Convolucional puede ser caracterizado por tres parámetros:

- Tasa (*Rate*): es la cantidad de bits de salida dada la cantidad de bits de entrada.
- Restricción de Longitud (*Constraint Length - K*): es el número de retrasos en el codificador. En otras palabras, se puede entender como la cantidad de bits de entrada que se conservan para generar una salida.
- Generador Polinomial (*Generator Polynomial*): representa las conexiones entre los retrasos y los elementos de entrada que van a formar una salida. Como ejemplo, considérese el polinomio en la ecuación (2.1) descrito en la figura 2.2. donde 7_8 realiza la suma en Modulo 2 (*Module-2 add*) de la entrada actual, la entrada previa y la entrada anterior a la anterior para formar $output_1$, y de la misma forma, 5_8 realiza la suma de la entrada actual y la entrada anterior a la anterior para formar $output_2$:

$$[7,5]_8 = [111,101]_2 \quad (2.1)$$

Un codificador convolucional también puede ser visto como una máquina de estados finita (*Finite State Machine - FSM*) a lo largo de un número determinado de espacios de tiempo, donde el número de estados corresponde a 2^n , siendo n el número de salidas del codificador. La figura 2.3 muestra la máquina de estados de un codificador con tasa $\frac{1}{2}$. [3][4]

Figura 2-3 Máquina de estados finita de un Codificador Convolutivo.



2.1.2 Intercalador de Bits

Un intercalador de bits es utilizado en sistemas de comunicación para reorganizar una trama de bits a ser enviada de tal forma que si ocurre una pérdida de datos en un espacio de tiempo dado, esta no afecte todo un símbolo. Los intercaladores de bits son comúnmente utilizados junto con esquemas de codificación FEC, dado que esto permite recuperar de forma completa los datos en ambientes de transmisión variables donde dicha transmisión se ve afectada por ruido e interferencia.

Algunos métodos de intercalado que han sido desarrollados y estandarizados son [5] [6]: (1) Intercalado Rectangular, donde los bits son reorganizados por filas formando una matriz y enviados por columnas, y (2) Intercalado aleatorio donde los bits son reordenados usando permutación pseudo-aleatoria.

2.1.3 Esquemas de Modulación

La modulación es una serie de técnicas que consiste en variar parámetros de una señal digital o analógica para transportarla usando una señal portadora. La modulación analógica puede hacerse variando la frecuencia, amplitud o fase de la señal portadora, pero la modulación digital ha sido la técnica utilizada en sistemas modernos porque ofrece algunas ventajas sobre las técnicas análogas; algoritmos robustos de corrección de errores y codificación, como el presentado en la sección 2.1.1. y altas velocidades de transmisión son algunas de las mejoras en el procesamiento de la información que se da como resultado del envío de esta en la forma de símbolos formados por bits.

Existen numerosos esquemas de modulación digital que se utilizan dependiendo del propósito y el tipo de información a ser enviado. Algunas de las restricciones a tomar en

cuenta al momento de elegir un esquema de modulación específico son: velocidad de transmisión, eficiencia espectral² y error esperado.

Los esquemas de modulación digital utilizados comúnmente están basados en frecuencia, fase y amplitud o combinaciones de estos:

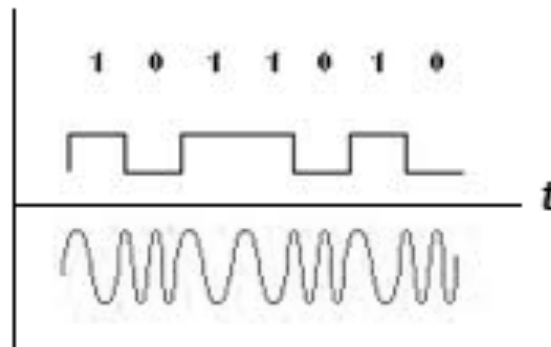
- **MFSK (*Frequency Shift Keying*):**

En este tipo de modulación la señal está dada por la ecuación (2.2):

$$s_i(t) = A \cos[2\pi f_c t + 2\pi \alpha_i \Delta f_c t + \varphi_i], 0 \leq t < T_s \quad (2.2)$$

Donde f_c representa la frecuencia de portadora, $\alpha_i = 2^i - 1 - M$, donde M indica el tamaño del alfabeto de símbolos binarios y la mínima separación entre portadoras está dado por $2\pi \alpha_i \Delta f_c$. La figura 2.4 muestra la relación entre la trama de datos binaria y la señal modulada.

Figura 2-4 Señal FSK Modulada.



- **MPSK:**

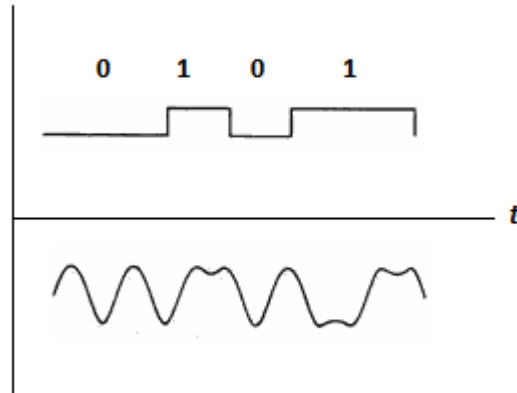
En este tipo de modulación la señal a enviar es formada por cambios de fase de acuerdo a una trama de bits como se muestra en la figura 2.5. La señal modulada se da por la ecuación (2.3)

$$s_i(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \vartheta) \quad (2.3)$$

² La eficiencia espectral se refiere a la medida de aprovechamiento de la banda de frecuencias utilizada para transmitir datos. Está definida como $E = R/B$, donde R es la tasa de transmisión y B es el ancho de banda utilizado.

Donde E_b representa la energía por bit, T_b es el tamaño del alfabeto de símbolos y $\vartheta = \frac{2\pi(i-1)}{M}$ indica las diferentes fases que puede tomar la señal.

Figura 2-5 Señal PSK modulada.



- **MQAM:**

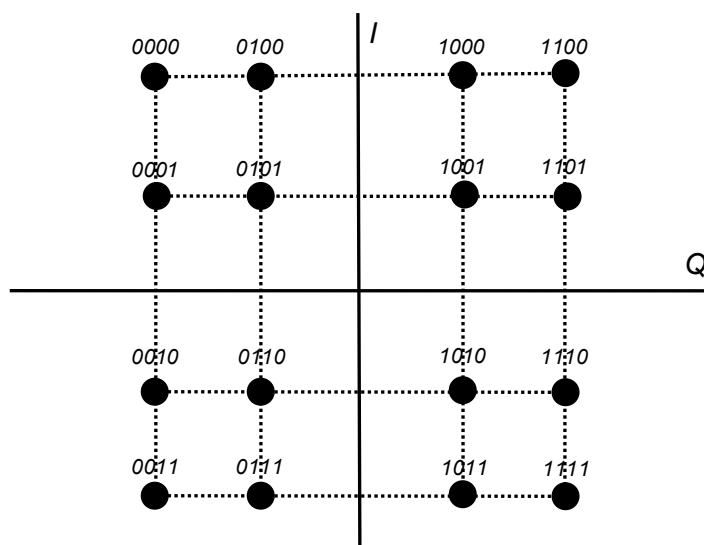
Para este tipo de modulación la codificación de los símbolos que contienen el mensaje es realizada de acuerdo a cambios de amplitud y fase al mismo tiempo, y por lo tanto puede ser considerada como la forma de modulación más eficiente en términos de espectro; por esta razón es utilizado extensamente en sistemas de comunicaciones. La señal modulada se da por la ecuación (2.4), donde $I(t)$ y $Q(t)$ son los componentes de fase y cuadratura respectivamente.

$$s_i(t) = I(t) \cos(2\pi fct) - Q(t) \sin(2\pi fct) \quad (2.4)$$

Para lograr eficiencia espectral en QAM se utilizan mapas de constelaciones organizados de forma específica que permiten codificar $\log_2 M$ bits en un mismo símbolo. Como ejemplo obsérvese la figura 2.6 se muestra el diagrama de constelación rectangular para 16QAM, que se forma por la ecuación (2.5)

$$\propto MQAM = \{\pm(2M - 1) + (2M - 1)\} \quad (2.5)$$

Figura 2-6. Diagrama de Constelación para 16QAM Rectangular.



2.1.4 Codificación Espacio - Tiempo: Código Alamouti

La diversidad de espacio es una técnica en la que la información es transmitida usando diversos caminos, lo cual es logrado mediante el uso de dos o más antenas en el transmisor y/o receptor, a lo que se le da el nombre de MIMO (*Multiple Input – Multiple output*). Esta técnica ha tomado gran importancia ya que permite una tasa de transmisión más alta sin que sea necesario aumentar el ancho de banda.

La diversidad de tiempo, al igual que la de espacio, se utiliza en comunicaciones digitales para combatir los errores ocurridos en canales de transmisión variables en el tiempo y consiste en enviar datos en diferentes espacios de tiempo de forma repetitiva para generar redundancia.

Alrededor de la diversidad de espacio y tiempo han sido desarrolladas técnicas de codificación en las que la señal es enviada de forma redundante para contrarrestar los efectos causados por fenómenos de desvanecimiento e interferencias del canal de comunicaciones; estas técnicas pueden ser divididas en dos categorías principalmente que son [36]: *Space – Time Trellis Codes* (STTC) en la cual se utiliza un código Trellis – que es un tipo de codificación convolucional - sobre una trama de datos, y *Space – Time Block Codes* (STBC) en la que la codificación es realizada en un pequeño bloque de la señal como es el caso de la codificación Alamouti que es la más utilizada y simple de su tipo; otro tipo de codificación es el denominado *Bell Laboratories Layered Space-Time* (BLAST) que además aumenta la capacidad de envío de datos [37].

Los códigos Alamouti fueron inventados por Siavash Alamouti [6] y se utilizaron inicialmente para el caso de dos antenas transmisoras y una antena receptora (2 X 1),

pero posteriormente han sido extendidas al caso 2 X 2. El objetivo es organizar y codificar los datos de tal forma que estos puedan ser recuperados en el receptor utilizando las componentes de la señal captadas por cada antena, eliminando así las interferencias causadas por el canal. La codificación para el caso MIMO 2 X 2 es realizada de acuerdo a la ecuación (2.6). Donde X_1 y X_2 son los bits a transmitir, T_{x_1} y T_{x_2} son la antena uno y dos respectivamente y t_n indican espacios de tiempo.

$$\begin{array}{cc}
 & \begin{array}{c} T_{x_1} \\ T_{x_2} \end{array} \\
 \begin{array}{c} t_1 \\ t_2 \end{array} & \begin{array}{|c|c|} \hline X_1 & X_2 \\ \hline -X_2^* & X_1^* \\ \hline \end{array}
 \end{array} \quad (2.6)$$

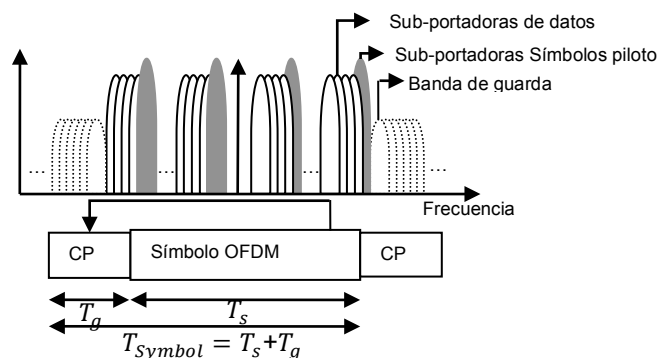
2.1.5 Multiplexación Ortogonal por División de Frecuencia - OFDM

Una vez los datos son codificados para ser transmitidos por múltiples antenas es necesario convertirlos a una señal paralela para que estos sean modulados en múltiples portadoras ortogonales³. La modulación OFDM consiste en codificar la señal de tal forma que esta sea representada por diversos símbolos ortogonales que pueden ser transmitidos de forma simultánea aumentando la tasa de transmisión del sistema. La importancia de la ortogonalidad radica en que aunque las señales son enviadas al mismo tiempo, la interferencia que podrían causarse es eliminada, haciendo que no sea necesario incluir un intervalo de guarda entre portadoras.

El símbolo OFDM se constituye por una serie de símbolos que contienen la información, unos símbolos piloto que son insertados de forma específica para efectos de estimación de canal y una banda de guarda que elimina la interferencia de canal adyacente (del inglés *Adjacent-channel interference - ACI*) causada por potencias de señales fuera de banda. Además, se añade a la señal un intervalo de guarda en forma de prefijo cíclico (*Cyclic Prefix - CP*), sufijo cíclico (*Suffix Prefix - SP*), o rellenado de ceros (*Zero Padding - ZP*) que contribuye a eliminar la interferencia inter-símbolo (*Intersymbol Interference - ISI*) que es la interferencia generada por un símbolo OFDM a símbolos subsiguientes. La figura 2.7 muestra la estructura de un símbolo OFDM; allí T_s representa el periodo del símbolo OFDM y T_g el periodo del intervalo de guarda (CP)

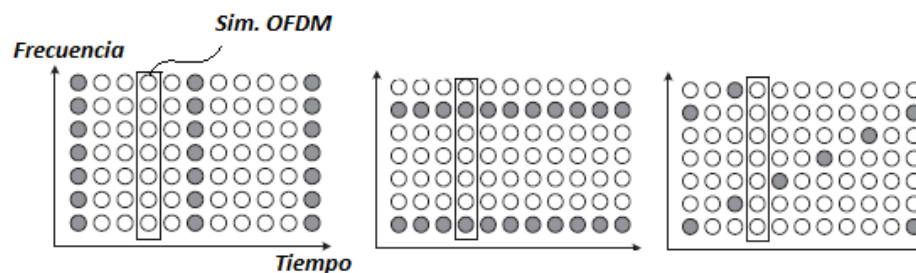
³ Ortogonalidad en matemáticas representa la perpendicularidad geométrica en un espacio vectorial $x \perp y$.

Figura 2-7 Estructura de símbolo OFDM.



- **Símbolos piloto:**

Los símbolos piloto pueden ser insertados de forma que sea posible, a partir de ellos, estimar todas las características del canal de comunicaciones. Existen tres tipos de estructuras definidas que son, *Block type* en el que los símbolos son insertados en todas las sub-portadoras de un símbolo OFDM en espacios de tiempo de forma periódica, *Comb type* en el que son ubicados en sub-portadoras separadas entre sí por n sub-portadoras, y *Lattice type* en el que los símbolos piloto son arreglados por sub-portadoras de un símbolo pero también por espacios de tiempo como se muestra en la figura 2.8.

Figura 2-8 Estructura de símbolos piloto. a) *Block type*, b) *Comb type*, c) *Lattice Type*.

a)

b)

c)

- **Intervalo de Guarda:**

Con el objetivo de incrementar la tasa de transmisión disminuyendo interferencias en la señal, se utiliza el intervalo de guarda, cuya longitud depende de parámetros de la transmisión como retrasos de propagación y reflexión de la señal. Entre más largo es el intervalo de guarda, hay mayor inmunidad a interferencias por retraso de la señal. Este intervalo es una extensión del símbolo OFDM en un factor $T_g = NT_s$ donde N es el tamaño de la extensión dada típicamente en valores de $\frac{1}{4}$, $\frac{1}{8}$, y $\frac{1}{32}$ como se muestra en la figura 2.8.

- **OFDM y transformada de Fourier:**

La ortogonalidad en la señal a transmitir es lograda eficientemente implementando algoritmos de transformada de Fourier en la etapa receptora y su inversa en la etapa

transmisora. El algoritmo de transformada rápida de Fourier o *FFT* es utilizado generalmente para lograr este objetivo, y el tamaño de esta depende del ancho de banda, entre otros factores propios del sistema de comunicaciones. Valores típicos son 256 y 512.

La tabla 2.1 muestra Valores de los parámetros mencionados anteriormente para una transmisión OFDM dependiendo del tamaño de la *fft*.

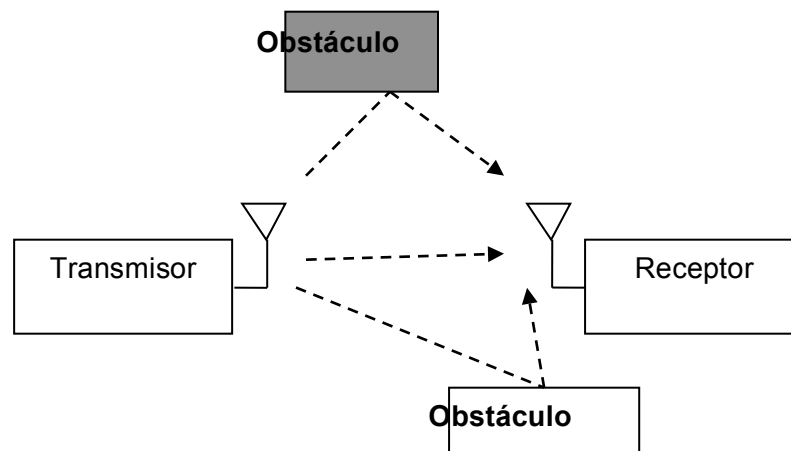
Tabla 2-1 Sub-portadoras en OFDM. [7]

FFT	N Datos	N Símbolos piloto	N Intervalo de guarda	N Prefijo cíclico
256	192	8	1/4	56
512	360	60	1/8	92

2.2 Canal de Comunicación

El canal de comunicaciones inalámbrico hace que se presenten una serie de desafíos relacionados con la propagación de la señal transmitida en un medio con interferencias y ruidos, que aumenta la tasa de errores del sistema. Por esto, y con el objetivo de prevenir altas tasas de error, es necesario conocer de forma clara el canal de comunicaciones sobre el cual se transmite, para lo cual existen modelos que especifican parámetros como retrasos, dispersión por efecto Doppler, multi-trayecto, desvanecimiento que indica el nivel de atenuación y niveles de ruido de acuerdo al ambiente, como se muestra en la figura 2.9.

Figura 2-9 Múltiples trayectos de una señal en un sistema Transmisor – Receptor



Para caracterizar el canal de comunicaciones, además del número de trayectos de señales que llegan al receptor, se toma en cuenta si existe o no línea de vista entre transmisor y receptor. En ese sentido, dos modelos principales de canal de

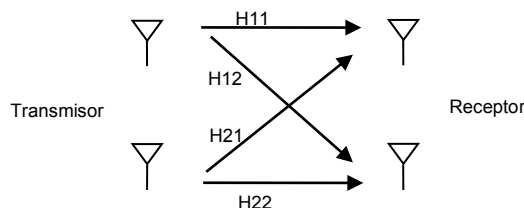
comunicaciones pueden ser identificados [57]: Rayleigh Fading (no existe línea de vista) y Rician Fading (existe línea de vista), que se expresan según la ecuación (2.7).

$$y_i = \sum_n S_{i-\alpha_n} g_n \quad (2.7)$$

Donde α_n representa el retraso de cada trayecto, y g_n la ganancia de trayecto, las cuales son modeladas como procesos gaussianos complejos con varianza unitaria y media cero. El factor de desvanecimiento del canal se puede presentar como selectivo en frecuencia y en tiempo, y rápido o lento y representa la atenuación que afecta la señal durante la propagación de la señal.

En sistemas MIMO-OFDM existe la ventaja de que el canal es modelado como de desvanecimiento plano, que significa que no afecta la señal de forma selectiva en frecuencia, gracias a la separación del ancho de banda en múltiples sub-portadoras. El modelo de canal MIMO es el mostrado en la figura 2.10, donde H11, H12, H21, H22 son ganancias de trayecto.

Figura 2-10 Modelo de canal MIMO.

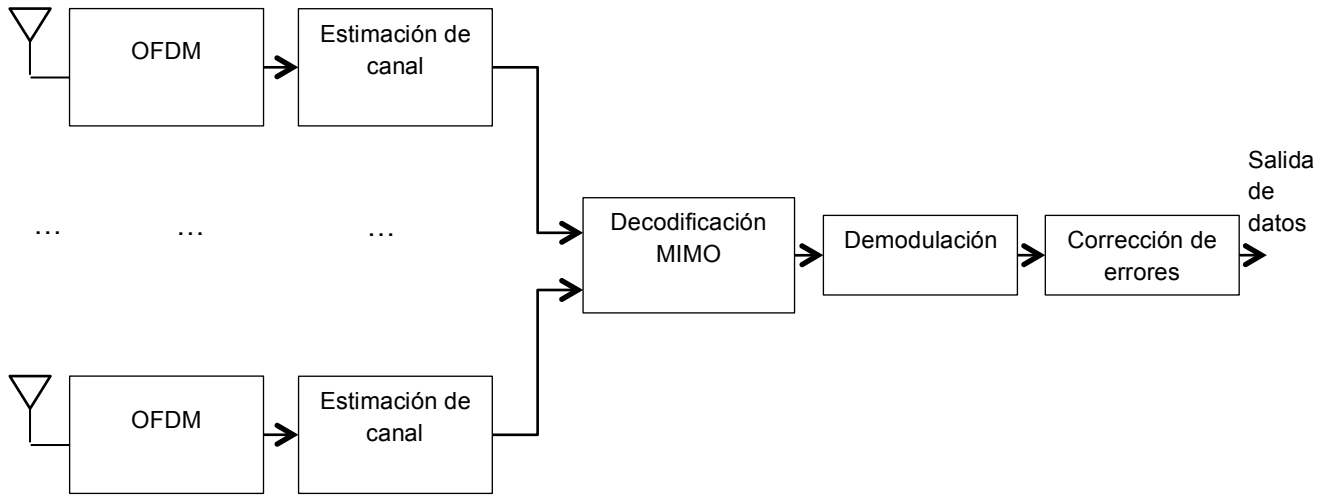


2.3 Sistema Receptor

En el receptor de un sistema de comunicaciones se realizan operaciones inversas al transmisor que permiten recuperar la información originalmente enviada. Se utilizan técnicas de decodificación y demodulación, además de operaciones de estimación de canal y realimentación de información de estado de canal.

En esta sección se mostraran algunas técnicas de decodificación MIMO, algoritmos de corrección de errores y técnicas de demodulación OFDM, utilizadas en típicamente un sistema MIMO-OFDM, de acuerdo a la figura a continuación donde se presenta la estructura básica de un sistema de recepción.

Figura 2-11. Sistema de Recepción Básico.



2.3.1 Decodificador MIMO

Los símbolos enviados por las antenas en el transmisor y recibidas en el receptor deben ser decodificadas de acuerdo a la forma como fueron codificadas. La decodificación de códigos Alamouti es realizada dependiendo de la cantidad de antenas en el transmisor (n) y en el receptor (m), y existen métodos de estimación de símbolos como *Zero Forcing (ZF)*, *Minimum Mean Square Error (MMSE)* y *Maximum Likelihood* [38] [39]. Para hacer la estimación y decodificación estas técnicas necesitan un conocimiento de las ganancias del canal de comunicaciones, como se muestra en la ecuación (2.8), donde la señal recibida se da durante dos espacios de tiempo para el caso MIMO de 2 X 2, asumiendo una codificación Alamouti como la descrita en la ecuación (2.6).

$$\begin{aligned} \begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix} &= \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1^1 \\ n_2^1 \end{bmatrix} \text{ espacio de tiempo } t_1 \\ \begin{bmatrix} y_1^2 \\ y_2^2 \end{bmatrix} &= \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} * \begin{bmatrix} -x_2^* \\ x_1^* \end{bmatrix} + \begin{bmatrix} n_1^2 \\ n_2^2 \end{bmatrix} \text{ espacio de tiempo } t_2 \end{aligned} \quad (2.8)$$

Donde y_n^m indica el símbolo recibido por la antena n en el espacio de tiempo, y h_{ij} son las ganancias de trayectoria del canal.

Combinando las ecuaciones anteriores se obtiene:

$$\begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1^1 \\ n_2^1 \\ n_1^{2*} \\ n_2^{2*} \end{bmatrix} \quad (2.9)$$

Y abreviando:

$$Y = H * X + N \quad (2.10)$$

De esa manera, para estimar x_1 y x_2 es necesario calcular la inversa de la matriz H . La matriz pseudo-inversa de H está definida así:

$$H^{inv} = (H^H H)^{-1} H^H \quad (2.11)$$

Por lo tanto, para obtener los símbolos \hat{x}_1 y \hat{x}_2 es necesario realizar la operación mostrada en (2.12).

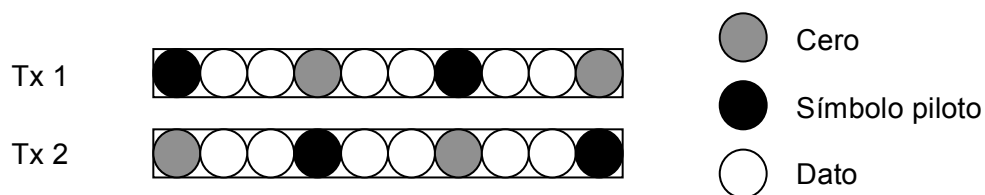
$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = (H^H H)^{-1} H^H \begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} \quad (2.12)$$

2.3.2 Estimación de Canal

El conocimiento de información del canal de comunicaciones en sistemas MIMO-OFDM es importante, ya que permite decodificar los símbolos recibidos por las antenas receptoras. Esto significa que entre más exacta sea la estimación de canal, menor será la desviación y posibles errores en la decodificación de símbolos.

Los símbolos piloto insertados en la señal transmitida son utilizados entonces para obtener de allí las ganancias de trayectoria de la señal captadas por cada antena receptora. Una de las estructuras generalmente utilizadas en sistemas MIMO-OFDM se muestra en la figura 2.1.

Figura 2-12 Estructura de símbolos piloto para múltiples antenas Tx.



En esta estructura los símbolos piloto están ordenados de forma periódica, e intercalados con símbolos de valor cero, y esta periodicidad depende de la rapidez con que cambia el canal de comunicaciones. Los símbolos de color gris que representan un cero en la señal son colocados para cancelar una de las ganancias de trayectoria del canal, de acuerdo a lo expresado en la ecuación (2.13), tomando en cuenta la figura 2.10 del modelo de canal MIMO. En el primer espacio de tiempo la señal y_1 recibida en la antena 1, asumiendo que x_1^1 es un símbolo piloto y x_2^1 es un cero:

$$y_1^1 = h_{11} * x_1^1 + h_{12} * x_2^1 + n_1^1 \quad (2.13)$$

Y para la antena 2 en el primer espacio de tiempo asumiendo x_1^1 como cero y x_2^1 un símbolo piloto:

$$y_2^1 = h_{22} * x_2^1 + h_{21} * x_1^1 + n_2^1 \quad (2.14)$$

De la misma forma, en el segundo espacio de tiempo asumiendo el caso contrario al primer espacio de tiempo:

$$y_1^2 = h_{11} * x_1^2 + h_{12} * x_2^2 + n_1^2 \quad (2.15)$$

$$y_2^2 = h_{22} * x_2^2 + h_{21} * x_1^2 + n_2^2 \quad (2.16)$$

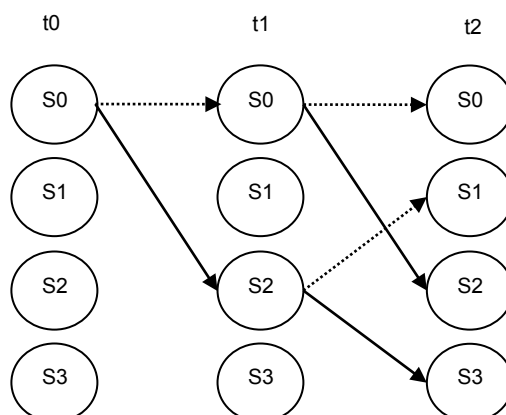
Lo cual significa que en dos espacios de tiempo en los que se reciben símbolos piloto, conocidos por el receptor, es posible calcular x_2 y x_1 realizando una simple división de los símbolos recibidos y los símbolos piloto [8]. Esta técnica es conocida como *Least Square* (LS), y ha sido ampliamente utilizada por su simplicidad [40] [41]. Sin embargo, como se puede notar, el ruido no es tomado en cuenta al momento de calcular las ganancias de trayectoria del canal, lo que significa que entre mayor es el nivel de ruido presente en el ambiente, más imprecisa es la estimación del canal. Otras técnicas de estimación de canal más robustas incluyen la implementación de filtros de análisis de desviación de error [9] [10].

2.3.3 Decodificación Convolutiva

La Decodificación de códigos convolucionales puede ser altamente compleja dependiendo de los parámetros de codificación como el número de retrasos de entrada utilizados. El algoritmo de Viterbi, Propuesto por Andrew Viterbi en 1967 [11] es uno de los más utilizados para valores pequeños del parámetro k (*Constraint Length*) mencionado en la sección 2.1.1. Este algoritmo hace uso de tablas de relación entre entradas, salidas y estados.

La figura 2.13 muestra la relación de cada estado con las posibles salidas de acuerdo al estado siguiente en n espacios de tiempo.

Figura 2-13 Diagrama de decodificación convolucional para R=1/2.



Inicialmente el decodificador se posiciona en el estado cero S0, al cual pueden seguirle únicamente los estados S0 y S2, de acuerdo a la tabla de sucesión de estados indicada por el codificador (ver tabla 2.2). De forma sucesiva, el objetivo es hallar el camino de estados para los cuales existe el menor error acumulado, para lo que es necesario hacer un cálculo de “distancia” entre el símbolo recibido y todos los posibles valores de símbolos que hay; esta distancia puede ser computada usando la medida de *distancia hamming*⁴, la cual se calcula realizando comparaciones bit a bit. Para este caso, por ejemplo, los posibles símbolos recibidos son 00, 01, 10, y 11, dado que la tasa de codificación (R) es de $\frac{1}{2}$. Se sabe, dada la tabla de posibles salidas según estado (ver tabla 2.3), que para el estado S0 los posibles símbolos recibidos son 00 y 11; supóngase que en el espacio de tiempo t0 se recibió el símbolo 01, esto indicaría que la distancia *hamming* en el trayecto s0 – s0 es de 1, al igual que en el trayecto s0 – s2.

Tabla 2-2 Sucesión de estados para codificador convolucional R=1/2.

Estado actual	Estado Siguiendo para	
	Entrada = 0	Entrada = 1
S0	00	10
S1	00	10
S2	01	11
S3	02	11

Tabla 2-3 Posibles salidas según estado estados para codificador convolucional R=1/2.

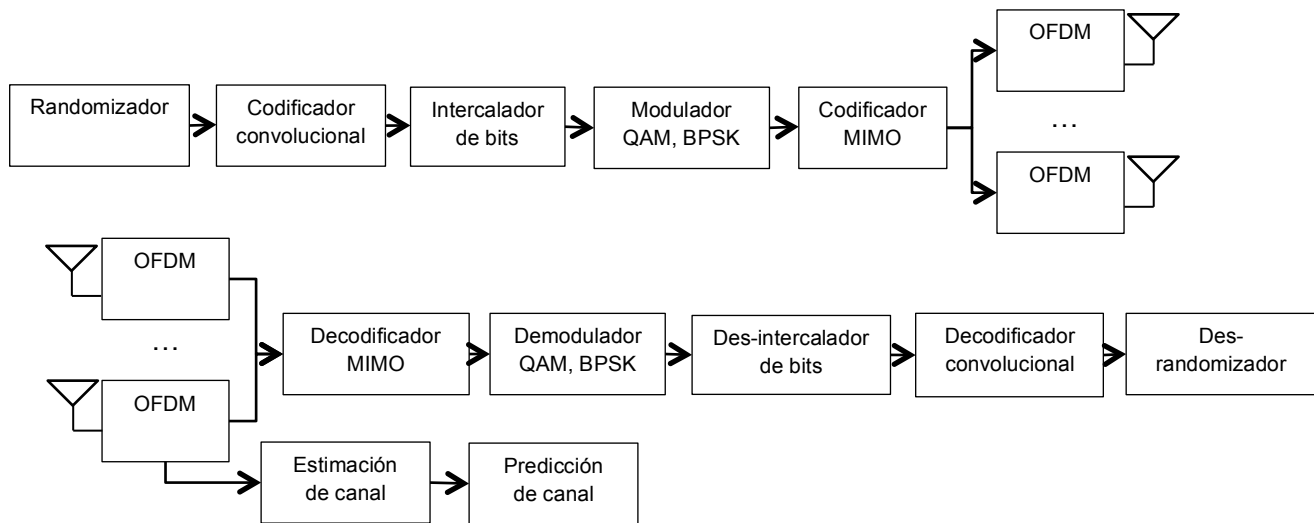
Estado actual	Salida para	
	Entrada = 0	Entrada = 1
S0	00	11
S1	11	00
S2	10	01
S3	01	10

⁴ La Distancia Hamming se refiere a la diferencia entre una palabra de código válida y otra medida en bits; por ejemplo, la distancia hamming entre 1111000 y 1101100 es 2.

Una vez el algoritmo ha computado el error acumulado para cada trayecto (*Accumulated Error Metric - AEM*) en toda la trama de bits recibida, se selecciona el trayecto con menor valor AEM y se recorre este decodificando los símbolos de acuerdo a la tabla 2.2.

Tomando en cuenta la descripción teórica de las técnicas alrededor de MIMO y OFDM, se puede establecer el diagrama de bloques del sistema de comunicaciones. La figura 2.14 muestra la descripción estructural de un sistema con los parámetros propios de MIMO-OFDM, donde se incluye un bloque de predicción de canal, el cual es utilizado para realizar adaptación al canal de comunicaciones de forma precisa; este bloque será explicado de forma amplia en la sección 4.

Figura 2-14. Transmisor y Receptor de Sistema MIMO-OFDM



3. Diseño del Sistema

Esta sección describe el proceso de diseño de un sistema de comunicaciones MIMO-OFDM seleccionando parámetros de codificación y modulación de acuerdo al modelo de la capa física como se muestra en la figura 3.1. En esta sección se identifican unos parámetros fijos de funcionamiento del sistema MIMO-OFDM, como son los esquemas de modulación, la tasa de codificación convolucional y el canal de comunicaciones entre otros, y posteriormente se presenta el problema de diseño SoC para el cual se establecieron las tareas de procesamiento y se separaron de forma modular para identificar aquellas con mayor cantidad de operaciones; finalmente se tomó una decisión de particionamiento Hardware/Software.

Se busca realizar la descripción del sistema MIMO-OFDM como un *System on Chip* (SoC). Un SoC es la integración de elementos de computadoras y elementos electrónicos dispuestos en un mismo chip con propósito específico, en el que además se pueden encontrar componentes de procesamiento de señales de radio frecuencia. Esta tecnología es ampliamente utilizada en la actualidad dado que permite realizar un diseño en el que se incorpore Hardware y Software con el objetivo de desarrollar sistemas embebidos con mayor capacidad de procesamiento en un menor tiempo.

Para la descripción del SoC se realiza primero un particionamiento hardware/software. Esta es una tarea en la que se decide cuales tareas o funciones del sistema serán implementadas en hardware y cuales en software. El particionamiento se realizó tomando en cuenta restricciones y características propias tanto de un sistema de comunicaciones inalámbrico, como del modelo de canal de transmisión; Finalmente en esta sección se presenta la arquitectura final del sistema diseñado con una estructura modular.

3.1 Configuración de sistema MIMO-OFDM

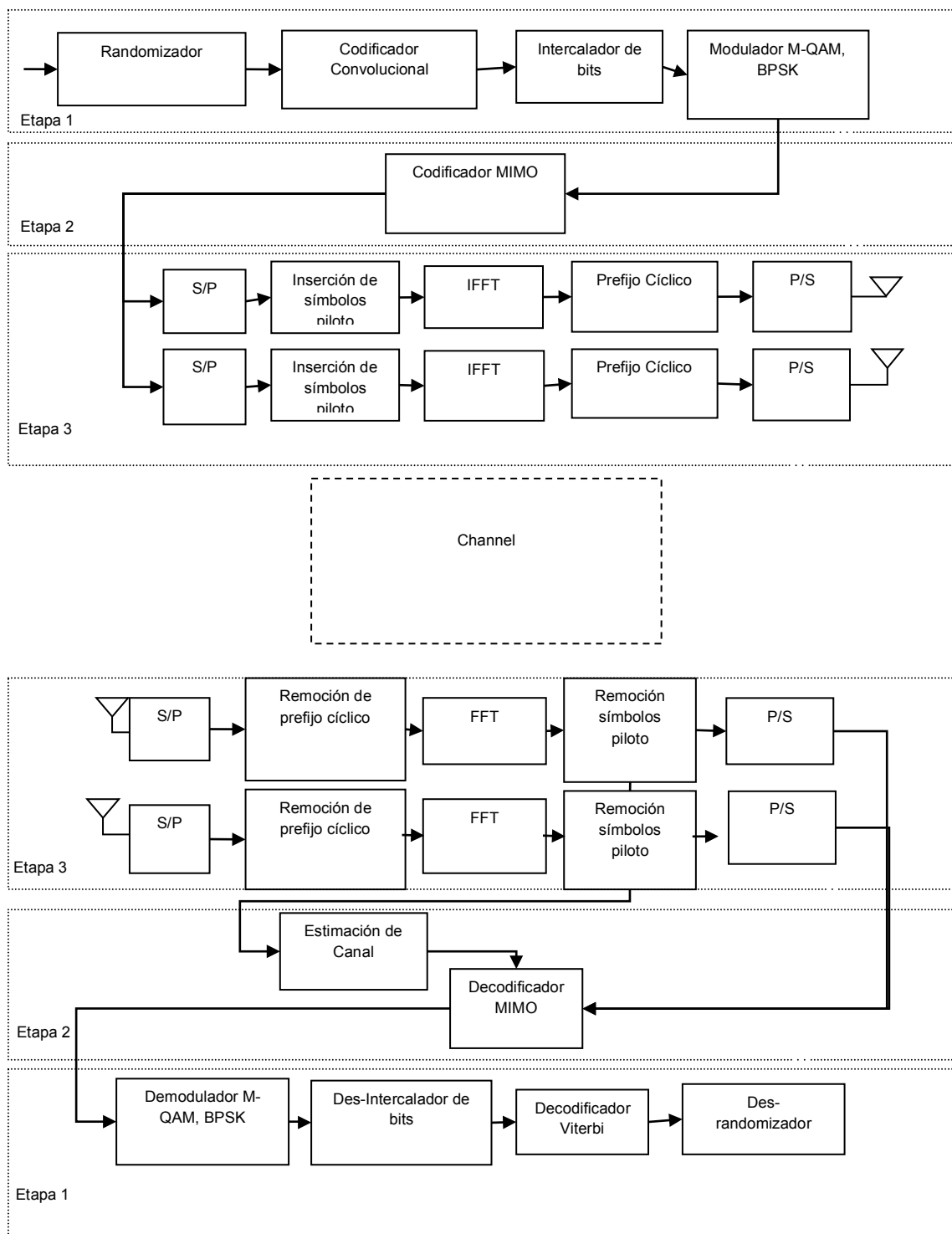
Tomando como base la sección anterior y los sistemas descritos en [7] y [58] se diseñó un sistema de comunicaciones compuesto por un módulo transmisor, un módulo receptor y un módulo que simula el canal de comunicaciones inalámbrico. Las características del sistema son descritas en la tabla 3.1 y corresponden a los parámetros de un sistema con esquemas de modulación adaptativo.

Tabla 3-1 Parámetros de sistema MIMO-OFDM.

Parámetros	Especificaciones
Modulación	BPSK, 16QAM, 64QAM, 256QAM
Tasa de codificación convolucional	$R = 1/2$
Tamaño de FFT	256, 512
Antenas Tx	2
Antenas Rx	2
Codificación MIMO	STBC Alamouti
Canal de comunicaciones	Flat Rayleigh fading

Como ejemplo de sistemas de comunicaciones 4G, Wimax es un estándar IEEE (802.16) para redes inalámbricas de amplio rango. Wimax considera el uso de la tecnología MIMO-OFDM para la capa física (PHY) y por lo tanto es el punto de inicio para este trabajo [13].

Figura 3-1 Estructura Sistema MIMO-OFDM.



El sistema MIMO-OFDM diseñado, que se basa en el descrito en la figura 2.14 de la sección 2.3.3., se organiza en tres etapas, tanto para el modulo transmisor y el receptor:

Para el transmisor:

1. la primera etapa están los módulos: *Randomizador*, *intercalador de bits*, *codificador convolucional*, y modulador *M-QAM*;
2. en la segunda etapa el módulo de codificación *STBC Alamouti* que ordena los bits para ser transmitidos en dos antenas y
3. la tercera etapa corresponde al módulo *OFDM* en el que se hace operación de FFT, inserción de símbolos piloto (*Pilot Insertion*) e intervalos de guarda (*Cyclic Prefix*).

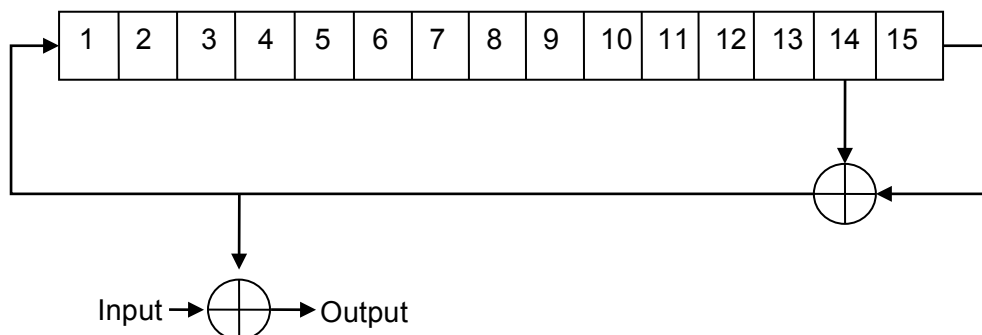
Para el receptor son realizadas operaciones inversas y son implementadas técnicas de decodificación para recuperar los datos originales,

1. la tercera etapa se encarga de demodulación *OFDM*;
2. en la segunda etapa se hace decodificación *MIMO* usando estimación *Zero Forcing* y
3. en la primera etapa están los módulos de *Demodulador M-QAM*, *Desintercalador de bits*, *decodificador Viterbi* para decodificación convolucional y *de-randomizador*.

3.1.1 Etapa 1: Randomizador de bits, Intercalador de bits, Codificación Convolucional y Algoritmo Viterbi, Modulación M-QAM.

- La figura 3.2 muestra la estructura del bloque *Randomiser* o Randomizador de bits, cuya estructura hardware es utilizada en ambos extremos del transmisor y receptor. Como podemos ver en la figura este módulo está compuesto por 15 bits de retraso que combinados con la entrada de datos binaria, mediante suma de módulo-2, forma una salida pseudo-aleatoria según el polinomio $1 + x^{14} + x^{15}$. Cada bit de la secuencia es inicializado en cada transmisión con la trama 100101010000000.

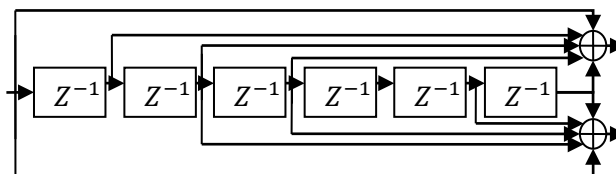
Figura 3-2 Estructura *Randomiser/Derandomiser*.



- El codificador convolucional usado en este trabajo se compone de 6 retrasos y dos salidas de la suma en modulo-2. El polinomio que forma la estructura del codificador se muestra en la ecuación (3.1); la estructura final del módulo se muestra en la figura 3.3.

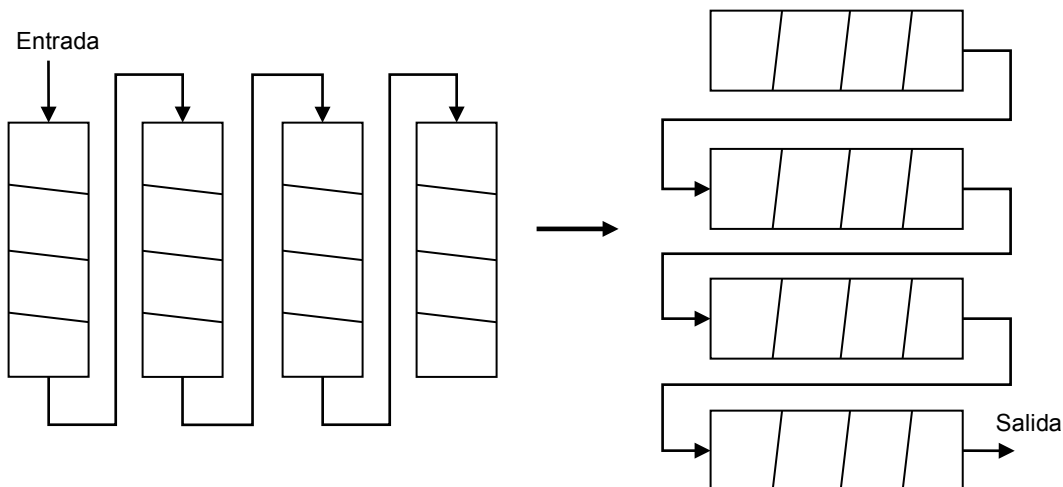
$$[171,133]_8 = [1111001,1011011]_2 \quad (3.1)$$

Figura 3-3 Estructura *Convolutional Coder*.



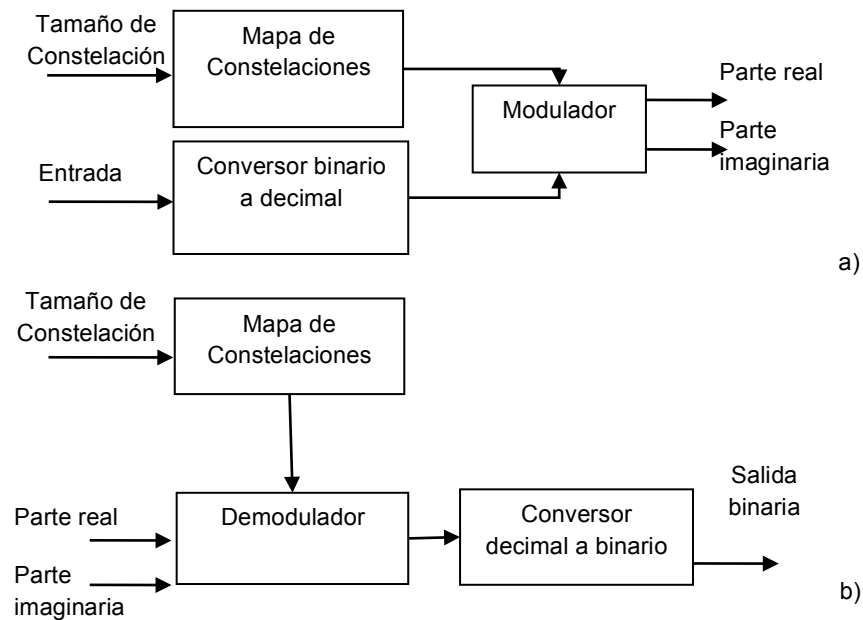
- La estructura del intercalador de bits utilizado en este trabajo se muestra en la figura 3.4. Allí se toman las tramas binarias de salida del codificador convolucional y se intercalan una a una formando una matriz de tamaño variable de acuerdo al tamaño de la trama binaria y al esquema de modulación. Después de llenar la matriz por filas, se vacía por columnas para formar la salida.

Figura 3-4 Estructura *Interleaver/Deinterleaver*.



- El módulo siguiente al de intercalado de bits es el modulador, cuya estructura se muestra en la figura 3.5 (a). En este trabajo utilizamos dos tipos de modulación, BPSK y M-QAM con $M = 16, 64$ y 256 . En este módulo se convierte la trama binaria en una decimal según el esquema de modulación seleccionado y posteriormente se modula con la constelación correspondiente. En el caso de la modulación BPSK únicamente se obtiene una salida real. El demodulador realiza la operación contraria al modulador; su estructura se muestra en la figura 3.5 (b).

Figura 3-5 Estructura a) *M-QAM modulator*, b) *M-QAM demodulator*.



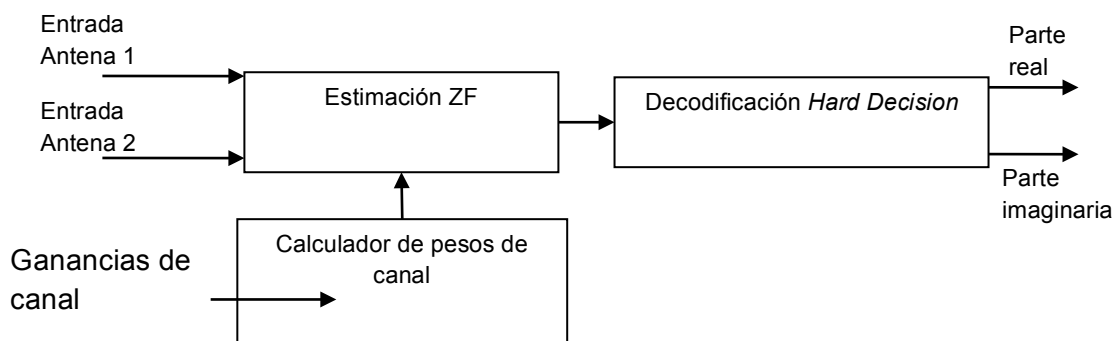
- El módulo final de la primera etapa del sistema de comunicaciones es el de decodificación convolucional ubicado en el receptor. En este módulo se hizo uso del algoritmo de viterbi para realizar la corrección de errores en bits. su estructura es fija, y dada por el polinomio de la ecuación (2.1) del codificador convolucional.

3.1.2 Etapa 2: Codificación MIMO Alamouti y Decodificación Zero Forcing

La estructura del codificador Alamouti está dada para un sistema MIMO de 2 x 2 antenas en transmisor y receptor. Su comportamiento está dado por la ecuación (2.6) del apartado anterior.

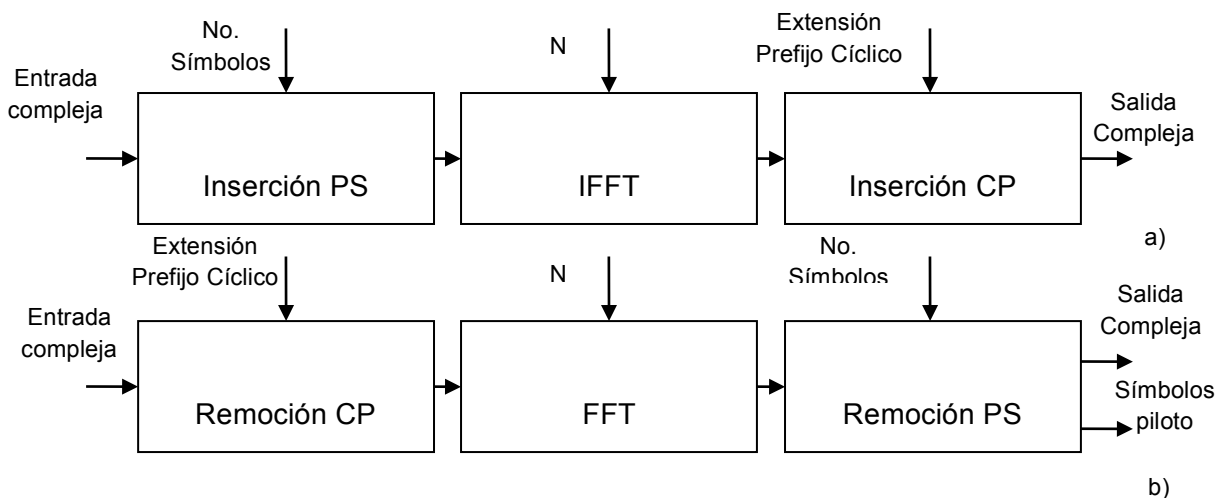
La figura 3.6 muestra el módulo decodificador MIMO donde se realiza la operación contraria al codificador Alamouti en el receptor. Este módulo se compone de un estimado *Zero Forcing* dado por las ecuaciones (2.8) a (2.12), un calculador de pesos para cada trayecto del canal H y un decodificador tipo *Hard Decision*⁵ que toma la trama de datos después de estimada y observa su cercanía a cada uno de los símbolos de la constelación dada por el esquema de modulación utilizado.

⁵ La decodificación *Hard decision* es un tipo de algoritmo en el que la base de datos para decodificar es fija. Este tipo de decodificación está en contraste con la denominada *Soft Decision* en la cual se toma información extra sobre la señal para decodificarla.

Figura 3-6 Estructura decodificador *MIMO*.

3.1.3 Etapa 3: Modulación y Demodulación *OFDM*

La estructura de los módulos de modulación y demodulación *OFDM* se muestra en la figura 3.7. Los parámetros de modulación están dados de acuerdo a la tabla 3.1, en la que se muestran dos tamaños de prefijo cíclico de 1/4 y 1/8; para este trabajo se utilizó el tamaño 1/4, con tamaño de *fft* igual a 256 como parámetros fijos, y la inserción de símbolos piloto se hizo de tipo *Comb* como se muestra en la figura 2.8 b). Como en el sistema *MIMO-OFDM* tenemos dos antenas en el transmisor y dos en el receptor es necesario tener dos moduladores y dos demoduladores.

Figura 3-7 Estructura *OFDM* Modulador/Demodulador.

3.1.4 Modelamiento de canal inalámbrico

El modelamiento del canal de comunicaciones representa una de las tareas más importantes cuando se trata de simular un sistema inalámbrico, pues a partir de allí se

observará y analizará su funcionamiento al transmitir en un medio ruidoso o con características no ideales.

En la literatura se encuentran disponibles diversas formas para simular un canal inalámbrico con parámetros como frecuencia Doppler (fd) y periodo de muestreo específicos; modelos estadísticos como los propuestos por William C. Jake [21] y R. H. Clarke [23] son los popularmente utilizados. La frecuencia Doppler y el periodo de muestreo juegan un papel importante en la descripción de canal, ya que se dice que un canal es de desvanecimiento rápido cuando el periodo de muestreo de la señal T_s es mayor al tiempo de coherencia $T_c = 1/fd$ y es de desvanecimiento lento cuando $T_s \ll T_c$. Visto de otra forma, un canal de desvanecimiento rápido es aquel cuyas ganancias cambian durante la transmisión de un símbolo.

El canal Rayleigh que se muestra a continuación está basado en [22], y consiste en una sumatoria de señales sinusoidales con fase y frecuencia Doppler aleatorias. Las ecuaciones (3.2) y (3.3) indican la formación del proceso Rayleigh con $h_i(t)$ y $h_q(t)$ las partes real e imaginaria de este.

$$h_i(t) = \sqrt{\frac{2}{M}} \sum_{n=1}^M \cos(w_d t \cos \alpha_n + \varphi_n) \quad (3.2)$$

$$h_q(t) = \sqrt{\frac{2}{M}} \sum_{n=1}^M \cos(w_d t \sin \alpha_n + \beta_n) \quad (3.3)$$

Con $\alpha_n = \frac{2\pi n - \pi + \theta}{4M}$ y M el número de trayectos del canal.

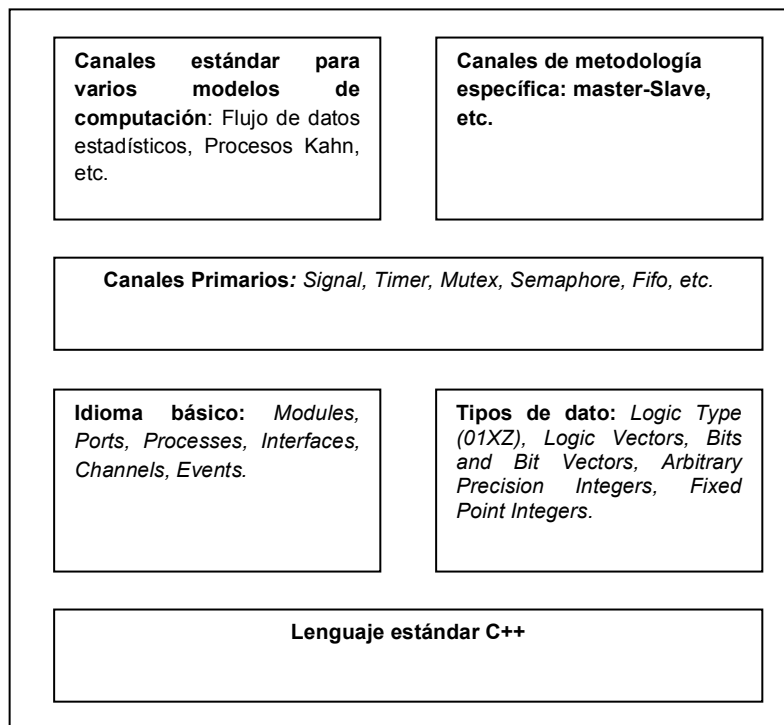
φ_n , β_n y θ son estadísticamente independientes y uniformemente distribuidos en $[-\pi, \pi)$ para todo n .

3.2 Diseño SystemC

SystemC [66] es una librería para C++ usada en el diseño a nivel de sistemas, y aprobada como estándar por la asociación de estándares IEEE como IEEE 1666-2005. Con esta librería es posible implementar, usando lenguaje C++, procesos que funcionan de forma concurrente y simular su comportamiento para propósitos de verificación.

La sintaxis utilizada en SystemC es propia de un diseño a nivel de sistemas, en el que existen elementos como módulos, puertos y procesos, además de señales para interconexión y canales de comunicación complejos como *FIFOs* y *buffers* entre otros. La figura 3.8 muestra las características principales de este lenguaje que permite diferentes niveles de abstracción.

Figura 3-8 Características de SystemC. [12]



El diseño SoC de sistemas de comunicaciones ha sido objeto de estudio en los últimos años [42]; la importancia de esto radica en el hecho de que cada vez se requiere un mejor desempeño de dichos sistemas, dado la creciente demanda de diferentes servicios de transmisión con numerosos usuarios, lo que hace que sea necesario implementar tecnologías que incrementen el rendimiento sin aumentar la latencia del sistema. Algunas de las características de codificación y modulación propias de los sistemas de comunicaciones actuales presentan una alta complejidad, lo que se traduce en implementaciones robustas; por lo tanto, se convierte en una necesidad el uso de herramientas que permitan integrar en un mismo marco diferentes estrategias de diseño para su validación, por lo que SystemC resulta ser una herramienta apropiada para esta tarea.

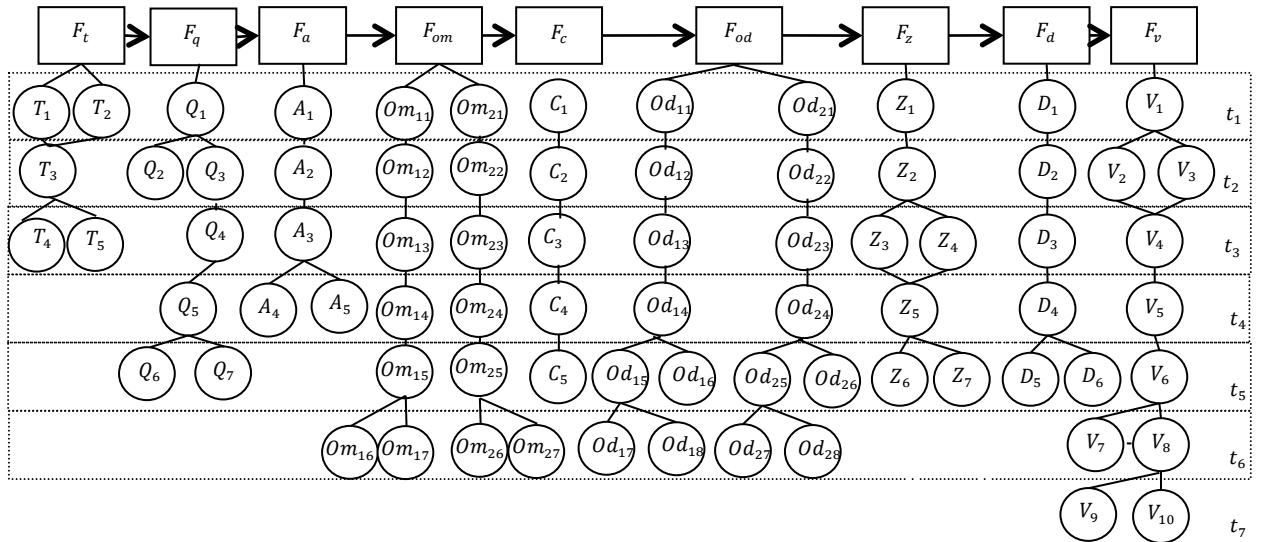
3.2.1 Parámetros de diseño SystemC

Con el objetivo de definir una arquitectura apropiada del sistema de comunicaciones en términos de su implementación, es necesario desarrollar criterios de medición del procesamiento realizado por este. En ese sentido se separó la arquitectura del sistema en forma modular de acuerdo a tres etapas de procesamiento, y se tomó en cuenta la cantidad de operaciones realizadas por cada uno de los módulos para tener un parámetro que permitiera tomar una decisión. También se estableció el nivel de

conurrencia del sistema de acuerdo a un flujo de tareas establecido para t_n espacios de tiempo.

La figura 3.9 muestra el proceso de separación de arquitectura realizado en el que se obtuvieron nueve funciones paralelas principales tomadas del diseño realizado en la sección 3.1: Codificación convolucional (F_t) y decodificación por algoritmo Viterbi (F_v), modulación QAM (F_q) y demodulación (F_d), codificación MIMO STBC (F_a) y estimación *Zero Forcing* (F_z), modulación OFDM (F_{om}) y demodulación (F_{od}), y simulación de canal de comunicaciones (F_c), que en este caso es usado únicamente para efectos de validación del sistema. Observando detalladamente en cada función del sistema otras sub-funciones más específicas son identificadas. De acuerdo con las funciones de procesamiento extraídas del sistema, los elementos más esenciales y robustos son la FFT para modulación OFDM y la decodificación y corrección de errores.

Figura 3-9 Flujo de tareas de sistema con arquitectura MIMO-OFDM.



La tabla 3.2 muestra la descripción de cada una de las tareas mostradas en la figura 3.9, ejecutadas en el espacio de tiempo t_n en el que debe ser ejecutada. $F_t, F_v, F_q, F_d, F_a, F_z, F_{om}, F_{od}$ y F_c , son las 9 funciones principales; para el modulo transmisor, F_t (aleatorización, codificación convolucional e intercalado), y F_q (modulación QAM) para la etapa 1, F_a (codificación Alamouti) para la etapa 2 y F_o (inserción de símbolos piloto, transformada rápida de Fourier inversa e inserción de prefijo cíclico) para la etapa 3 según la figura 3.1. Para el modulo receptor, F_{od} (remoción de prefijo cíclico, transformada de Fourier inversa y remoción de símbolos piloto) para la etapa 3, F_z (estimación ZF) para la etapa 2 y finalmente, F_d (demodulación QAM) y F_v (desintercalado, algoritmo Viterbi y des-aleatorización) para la etapa 1.

$T_n, Q_n, A_n, O_{1n}, O_{2n}$ representan las tareas específicas para cada función principal $F_t, F_v, F_q, F_d, F_a, F_z, F_{om}, F_{od}$ y F_c .

Tabla 3-2 Descripción tareas específicas.

Transmisor		Receptor	
Tarea Especifica	Función	Tarea Especifica	Función
$T_1, Q_1, A_1, Om_{11}, Om_{21}, Od_{11},$ Od_{21}, Z_1, D_1, V_1	Solicitud de memoria e inicialización de vectores	Od_{14}, Od_{24}	Operaciones de FFT para antenas 1 y 2 (a1 y a2).
T_2	Conversión de polinomio para codificación convolucional	Od_{15}, Od_{25}	Remoción de símbolos piloto para a1 y a2.
T_3	Desplazamiento de bits y adición modulo-2.	Od_{16}, Od_{26}	Dstrucción de parámetros de FFT.
$T_4, Q_6, A_4, Om_{16}, Om_{26},$ $Od_{17}, Od_{27}, Z_6, D_5$	Almacenamiento de valores en FIFOs	Z_2	Arreglo de datos de a1 y a2
$T_5, Q_7, A_5, O_{17}, Om_{27}, Od_{18},$ $Od_{28}, Z_7, D_6, V_{10}$	Liberación de memoria	Z_3	Operaciones de matrices inversa y hermitiana.
Q_2	Inicialización de constelación para modulación QAM	Z_4	Operaciones de conjugación
Q_3	Agrupación de bits por tamaño de símbolo	Z_5	Decodificación <i>Hard Decision</i>
Q_4	Conversión binario a decimal	D_2	Inicialización de constelación QAM para demodulación
Q_5	Reorganización de datos modulados.	D_3	Demodulación de símbolos complejos
A_2	Operaciones de conjugación y multiplicación para codificación espacial	D_4	Conversión decimal a binario
A_3	Almacenamiento de datos para antenas 1 y 2	V_2	Organización de datos para des-intercalado
Om_{12}, Om_{22}	Inserción de símbolos piloto	V_3	Inicialización de tablas de estado y salida para algoritmo Viterbi
Om_{13}, Om_{23}	Creación de parámetros de IFFT	V_4	Medición de distancia <i>hamming</i>
Om_{14}, Om_{24}	Ejecución de operación IFFT	V_5	Almacenamiento de medida de error acumulado
Om_{15}, Om_{25}	Adición de prefijo cíclico	V_6	Selección de camino de estados
Om_{18}, Om_{28}	Dstrucción de parámetros de IFFT	V_7	Conteo de errores
Od_{12}, Od_{22}	Remoción de prefijo cíclico	V_8	Decodificación de bits
Od_{13}, Od_{23}	Creación de parámetros para FFT.	V_9	Operación de des-aleatorización

Posterior a la identificación de tareas y su concurrencia se estableció un perfil de procesamiento contando la cantidad de operaciones de multiplicación-acumulación (*Multiply-accumulate operations -MA*) con el objetivo de establecer una medida de rendimiento para cada operación del sistema. Estos valores fueron establecidos usando herramientas de *profiling* de Visual Studio [24]. La evaluación de rendimiento fue

realizada con parámetros fijos de tamaño de datos, FFT (N), tasa de codificación (R), y esquema de modulación (M), seleccionados de acuerdo a la cantidad de datos necesarios para formar un símbolo OFDM de tamaño 256.

Tabla 3-3 Rendimiento de funciones con trama de datos = 384, R = 1/2, M = 16, N = 256

Función	Operaciones MA
Codificación Convolutiva	7507
Intercalador de bits	685
Constricción de constelación	1404
Conversión binaria a decimal	1464
Modulación	288
Codificación Alamouti	864
Operación de conjugada	1
Inserción de símbolos piloto	950
IFFT	Fma* = 128
Adición CP	768
Remoción CP	768
FFT	Fma* = 128
Remoción símbolos piloto	600
Arreglo de pesos Zero Forcing	Fma* = 34
Decodificación <i>hard</i> decisión MIMO	384
Conversión decimal a binaria	1156
Demodulación	1538
Algoritmo Viterbi	103296
Desintercalador	6349
Randomización	5760

*Fma = operación de MA para punto flotante.

Los resultados mostrados en la tabla 3.3 indican que las funciones que más operaciones de multiplicación-acumulación identificadas fueron: Codificación convolutiva, algoritmo Viterbi, y construcción de mapa de constelación QAM que consiste en accesos a memoria. Además se encontró que el tiempo más alto de ejecución fue dado por las operaciones de FFT e IFFT.

A partir de la información obtenida se propuso un particionamiento junto con una reconstrucción de los algoritmos para incrementar el rendimiento del sistema:

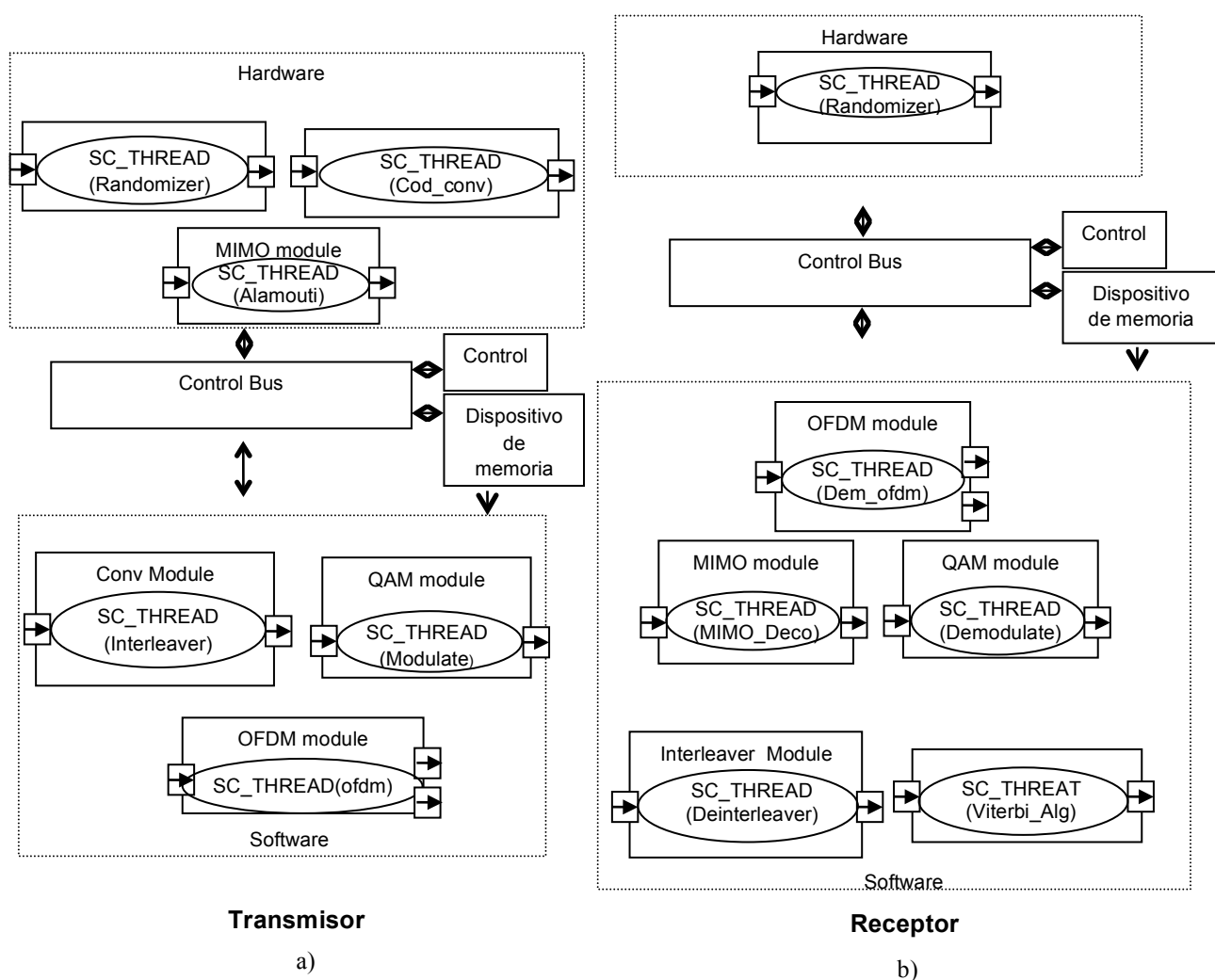
1. La solicitud y liberación de memoria en el sistema solo será hecho una vez con valores fijos en la inicialización del sistema.
2. La codificación convolutiva será realizada en hardware ya que su arquitectura es sencilla de implementar a un costo de tiempo de ejecución bajo.
3. Las operaciones de intercalado y des-intercalado serán realizadas de forma matemática.
4. Las matrices de modulación serán almacenadas de forma permanente en bloques de memoria con el objetivo de disminuir el tiempo de procesamiento de este módulo.

5. La librería *fftw* será usada ya que presenta el menor tiempo de procesamiento comparada con otros algoritmos de su tipo para aritmética de punto flotante.

3.2.2 Particionamiento Hardware/Software

La última etapa de descripción consiste en hacer un diseño modular hardware y software; el sistema MIMO-OFDM fue construido usando FIFOs para interconectar cada módulo y señales de control a todo el sistema. Un *sc_fifo* es un canal primitivo de SystemC encontrado en la case *sc_fifo*. La figura 3.10 muestra el diagrama del sistema resultante del problema de diseño y particionamiento hw/sw expuesto en esta sección.

Figura 3-10 Diseño SystemC de sistema MIMO-OFDM



4. Técnicas de Predicción de Canal

En esta sección se comparan algunas de las más utilizadas técnicas de predicción de canal aplicables a canales de desvanecimiento plano con múltiples trayectorias de señal. Para efectos de comparación estas técnicas fueron separadas y escogidas de acuerdo a su linealidad o no linealidad; en este sentido, fueron escogidas las técnicas de predicción lineal con modelo AR (autoregresivo) y filtro Kalman, como técnicas lineales, y redes neuronales artificiales (ANN – *Artificial Neural Network*) con y sin realimentación de la salida como técnicas no lineales. Los algoritmos diseñados e implementados en lenguaje C++ fueron integrados al sistema de comunicaciones MIMO-OFDM descrito en la sección 3 de tal forma que se hiciera posible una comparación a nivel de capa física de dicho sistema en el tiempo.

4.1 Algoritmos de predicción Lineal

Variadas técnicas de predicción lineal han sido propuestas en [16] [17] y [18] aplicables a sistemas MIMO-OFDM con modelo de canal de desvanecimiento plano Rayleigh. En el desarrollo de estas técnicas se muestra que es posible aproximar el canal de comunicaciones a un modelo lineal, para facilitar de esta forma el procesamiento de la señal.

Cabe notar que, dadas las características del canal, durante la aproximación de este a un modelo lineal es posible que se pierda generalidad en la representación de la señal, lo que podría traducirse en un nivel bajo de rendimiento o una mala predicción.

4.1.1 Predicción Lineal con Modelo Autoregresivo AR(n):

Los modelos estocásticos Autoregresivos son generalmente utilizados para describir series de tiempo en forma lineal [43] [44], los cuales se basan en la premisa de que una señal variable en el tiempo puede ser descrita por n valores anteriores de esta misma, mediante una sumatoria de los mismos como lo indica la ecuación (4.1). Estos modelos son finalmente filtros de respuesta infinita al impulso (IIR – *Infinite Impuse Response*).

$$X_t = \sum_{i=1}^n \phi_i X_{t-i} + \varepsilon_t \quad (4.1)$$

Dónde: n representa el orden del modelo, ϕ_i indica los parámetros o pesos del modelo y ε_t es el error asociado al mismo.

De la ecuación (4.1) puede fácilmente observarse que para representar cualquier señal mediante un modelo AR(n) es necesario tener un orden n adecuado pero también saber los valores de ϕ_i . El orden puede ser identificado de forma experimental, y para estimar los pesos ϕ_i adecuados es posible recurrir, por ejemplo, al set de ecuaciones de Yule –

Walker, que hacen uso de la matriz de autocovarianza o de autocorrelación de la señal a modelar, como lo muestra la ecuación (4.2a) y (4.2b) [14].

$$\mathbf{R}_{xx} \mathbf{a} = -\mathbf{v} \quad (4.2a)$$

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_i \end{bmatrix} = \begin{bmatrix} \gamma_0 & \gamma_{-1} & \gamma_{-2} \\ \gamma_1 & \gamma_0 & \gamma_{-1} \\ \vdots & \vdots & \vdots \\ \gamma_{n-1} & \gamma_{n-2} & \gamma_0 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_i \end{bmatrix} \quad (4.2b)$$

Con \mathbf{R}_{xx} como matriz de autocorrelación y \mathbf{a} como vector de pesos ϕ_i .

Estas ecuaciones pueden ser resueltas de forma eficiente utilizando la recursión de Levinson – Durbin [45] [46] que es un procedimiento matemático recursivo que permite calcular la solución a ecuaciones en las cuales hay una matriz toeplitz, que en este caso correspondería a la matriz de autocorrelación o autocovarianza de la señal a modelar (\mathbf{R}_{xx}) de tal forma que se garantiza una única solución para $\mathbf{a} = -\mathbf{v}\mathbf{R}_{xx}^{-1}$.

Una vez se sabe que es posible modelar el canal de comunicaciones usando una aproximación lineal del mismo. Para este trabajo se consideró un modelo AR con orden variable y se resolvieron las ecuaciones de Yule – Walker (ecuaciones 3.2a y 3.2b) de forma iterativa a lo largo del tiempo, usando la matriz de autocorrelación del canal para hacer la predicción del canal (\mathbf{H}_n) a un espacio de tiempo (\mathbf{H}_{n+1}).

En la tabla 4.1 se presenta el esquema en pseudocódigo del algoritmo de predicción lineal, para cada trayecto h_n^{ij} un canal MIMO con configuración 2 X 2 con desvanecimiento rápido en el tiempo y cuyos trayectos son no correlacionados:

Tabla 4-1 algoritmo pseudocódigo de predicción lineal con modelo AR(n).

```

Para ( n espacios de tiempo ) Hacer
{// Obtener Predicción de n + 1
  // Recursión Levinson – Durbin
  Coefij = -vRxx;
  Mientras ( j < Orden AR ) Hacer
  {
    hn+1ij =- Coefjij * hn-jij;
  }
}

```

4.1.2 Filtro Kalman Con Modelo Autoregresivo AR

El filtro Kalman, descrito inicialmente en el año 1960 [15] por Rudolf E. Kalmán, es un filtro digital recursivo ampliamente utilizado en la industria del control, especialmente para sistemas de navegación, dada su gran capacidad para realizar predicción de señales tomando en cuenta el ruido presente en la medición y el modelo que describe a dicha señal.

En este trabajo se diseñó el filtro kalman describiendo la señal con un modelo AR de orden variable de la misma forma como se realizó en el apartado anterior.

EL filtro Kalman está dispuesto en dos etapas, una de predicción y otra de actualización, donde la primera consiste en obtener una medida actual de la señal usando la matriz de transición \mathbf{A} que relaciona n con $n - 1$, así: $X_{n|n-1} = \mathbf{A} X_{n-1|n-1}$; y la segunda corrige la medida de la señal en cuestión usando una medición ruidosa Z_n de esta $Y_n = Z_n - \mathbf{H}_n X_{n|n-1}$.

La tabla 4.2 muestra el algoritmo iterativo que calcula el canal un espacio de tiempo hacia adelante [19].

Tabla 4-2 Algoritmo Pseudo-Código filtro Kalman.

<i>Para (n espacios de tiempo) Hacer</i>	
<i>{// Obtener Predicción de $\tilde{h}_k^{ij}(n+1)$</i>	
$\hat{h}_k^{ij}(n n-1) = \mathbf{A}_n h_k^{ij}(n-1 n-1);$ $P_k^{ij}(n n-1) = \mathbf{A}_n P_k^{ij}(n-1 n-1) \mathbf{A}_n^H + \mathbf{V}(n);$	Predicción
$K_k^{ij}(n) = P_k^{ij}(n n-1) \mathbf{C}^H * [\mathbf{C} P_k^{ij}(n n-1) \mathbf{C}^H + \mathbf{W}(n)];$ $\hat{h}_k^{ij}(n n) = \hat{h}_k^{ij}(n n-1) + K_k^{ij}(n) [Z_k^{ij}(n) - \mathbf{C} \hat{h}_k^{ij}(n-1 n-1)];$ $P_k^{ij}(n n) = [I_u - K_k^{ij}(n) \mathbf{C}] P_k^{ij}(n n-1);$ $\tilde{h}_k^{ij}(n+1) = \mathbf{A} \hat{h}_k^{ij}(n n);$	Actualización
<i>}</i>	

Donde P_k^{ij} Es la matriz de correlación, \mathbf{C} es la matriz de medición de la señal, y $\mathbf{V}(n)$ y $\mathbf{W}(n)$ son el ruido del modelo y el ruido de la medición Z_k^{ij} respectivamente.

Los valores iniciales de P_k^{ij} y h_k^{ij} son $0_{u,1}$ y I_u , donde u representa el orden del modelo A.

4.2 Algoritmos de Predicción No Lineal

Las técnicas de predicción no lineal son tomadas en cuenta en este trabajo y estudiadas en esta sección ya que representan una solución efectiva al problema de generalidad presente entre las técnicas lineales, dado que mediante estas puede tratarse características del canal como la no estacionariedad y la rápida variación con resultados precisos. Sin embargo, el aumento en la carga computacional puede ser representativo,

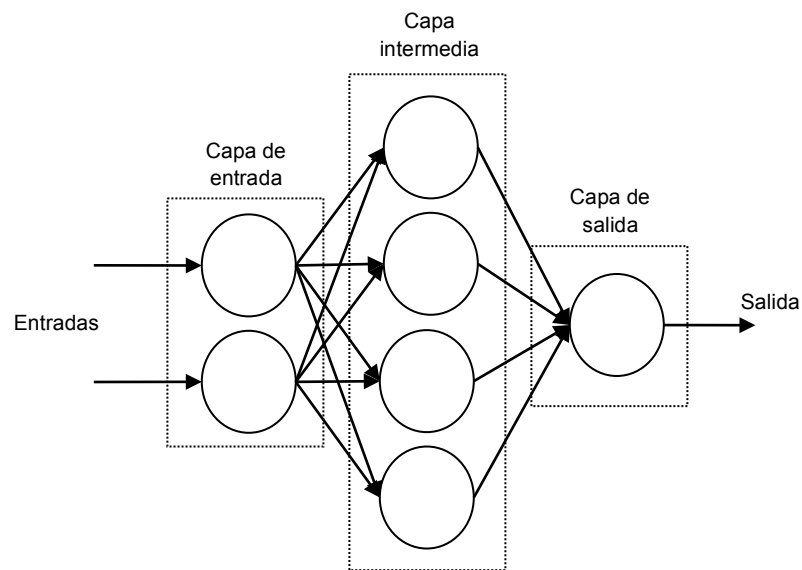
lo que resulta en un compromiso entre la precisión del algoritmo de predicción y la cantidad de operaciones realizadas. [20]

4.2.1 Redes Neuronales Con Retrasos de Tiempo

Las redes neuronales son herramientas de procesamiento no lineal para sets de datos con altos niveles de complejidad estadística. Estas redes son utilizadas en control adaptativo y aplicaciones sobre las cuales se tiene una amplia cantidad de información con la cual se entrena la red.

Las ANN (Artificial Neural Network) se componen, como las redes biológicas, de pequeñas unidades de procesamiento, o neuronas, dispuesta en una o varias capas. La figura 4.1 muestra la arquitectura de una red neuronal simple de tipo *Feedforward*, que se compone de una capa de entrada con dos neuronas, una capa intermedia con cuatro neuronas y una capa de salida con una neurona:

Figura 4-1 Red neuronal *Feedforward*.



Existen diversas configuraciones de redes neuronales, en las cuales se modifica la cantidad de neuronas, la cantidad de capas y la cantidad de entradas y salidas de acuerdo a la necesidad de la aplicación y la disposición de recursos para el procesamiento.

Una de las configuraciones que se tomó en cuenta para este trabajo es la de una Red Neuronal con retraso en el tiempo, que tiene la característica de tener como entrada valores retrasados en el tiempo de la señal misma, para de esta forma obtener un modelo de la señal basada en valores anteriores de esta, como se muestra en la figura 4.3.

Toda red neuronal debe disponer de una función de entrenamiento que también varía de acuerdo a la aplicación. Estas funciones cumplen una de las tareas más importantes y es la de encontrar los pesos de cada neurona w_n , (ver figura 4.2), de tal forma que la red tenga el comportamiento apropiado. Algunas de las funciones de entrenamiento más comunes son Algoritmos evolutivos y otros de optimización.

Figura 4-2 Diagrama interno de una neurona.

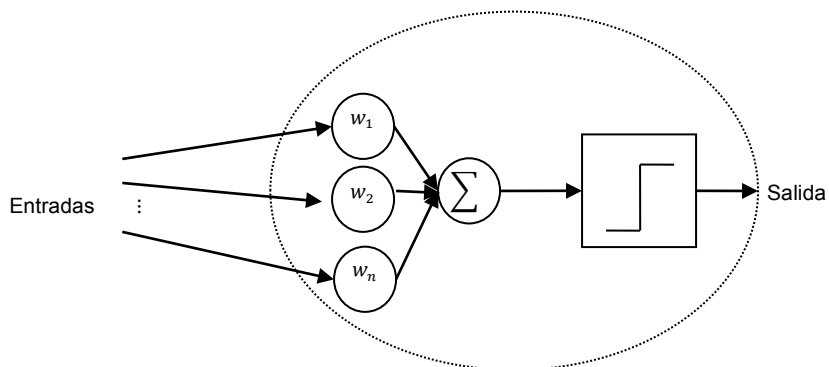
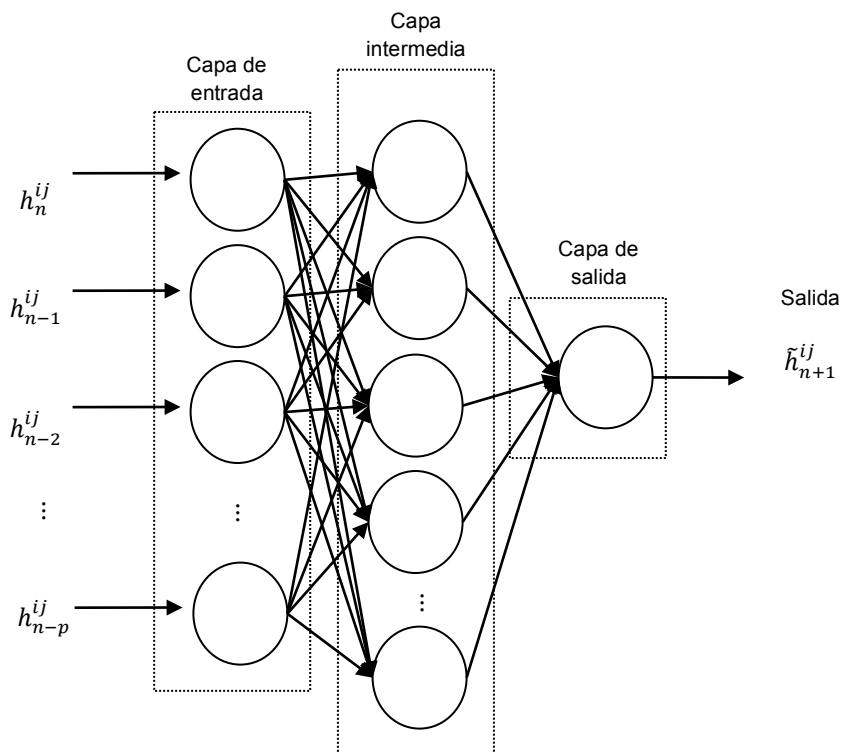


Figura 4-3 Red Neuronal con Retraso de tiempo.

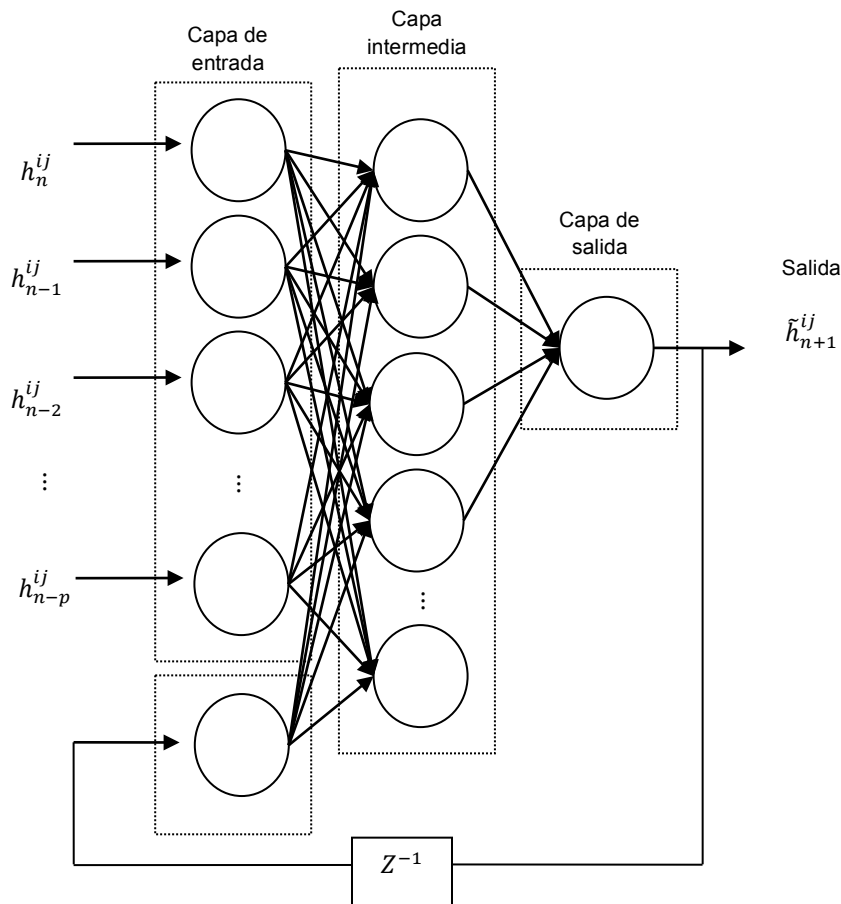


4.2.2 Redes Neuronales Recurrentes

Al igual que las mostradas en el apartado anterior, las redes neuronales recurrentes consisten en una agrupación de neuronas, en un número de capas determinado y entrenadas por una función que encuentra los pesos w_n óptimos para cada neurona.

Las redes neuronales recurrentes utilizan la salida de la red como una entrada retroalimentada, con el objetivo de que la red aprenda de la información estadística que la salida provee. En este caso se probaron tres funciones de entrenamiento para una red con p entradas que corresponden a p valores anteriores del canal y una salida realimentada como se muestra en la figura 4.4.

Figura 4-4 Red Neuronal Recurrente.



Existen descritas en la literatura diversas funciones de entrenamiento, algunas más complejas que otras. Esta es una de las etapas de la operación de una red neuronal más importantes ya que a partir de allí se obtienen los pesos de cada neurona en las diferentes capas [47].

La función de entrenamiento se encarga de tomar los datos de entrenamiento y encontrar la relación entre ellos, para así obtener el mejor modelo operacional de la red. Algunas

de las funciones de entrenamiento de la red están basadas en algoritmos de optimización aleatorios como evolutivos[48], *gradient descent* [49], y *Levenberg – Marquad* [50].

Se realizaron pruebas de entrenamiento de red con algoritmos evolutivos, *gradient descent* y *Levenberg – Marquad*; finalmente, dado que presentaba el menor tiempo de de entrenamiento, se utilizó la función *Levenberg – Marquad* para entrenar las redes con un set de datos de 5000 ganancias de canal Rayleigh.

La función de optimización *Levenberg – Marquad* utilizada sigue el problema de minimización *Least Square* de la función $S(\beta)$ de la ecuación (4.3).

$$S(\beta) = \sum_{i=1}^m [y_i - f(x_i, \beta)] \quad (4.3)$$

Donde (x_i, y_i) son parte del set de m variables dependientes entre sí, y β es el peso de optimización para el modelo $f(x_i, \beta)$.

A partir del estudio de algoritmos lineales y no lineales presentado en esta sección es posible realizar una implementación software que pueda ser integrada al sistema MIMO-OFDM descrito en la sección 3.2 de tal manera que se realice una comparación de dichos algoritmos y se establezca un recurso para la adaptación al canal de comunicaciones.

5. Resultados

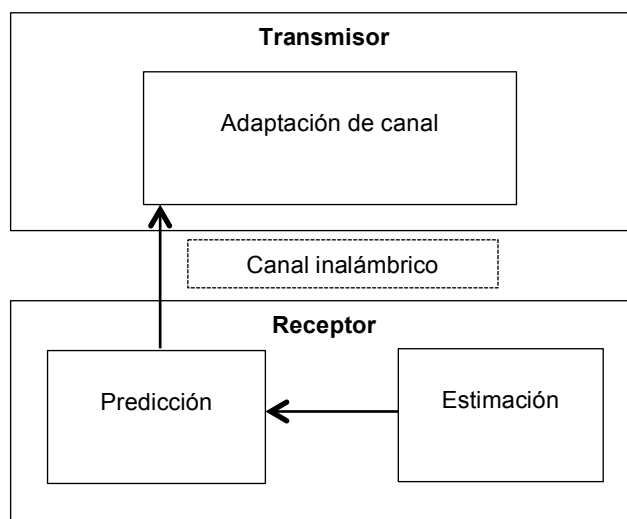
En esta sección se hace inicialmente una validación del sistema de comunicaciones, mostrando su comportamiento de los esquemas de modulación utilizados, la generación de ganancias de canal y la estimación de canal realizada, además del establecimiento de un algoritmo de adaptación del sistema al canal inalámbrico de acuerdo al comportamiento de esquemas de modulación BPSK, 16-QAM, 64-QAM y 256-QAM.

El posterior análisis del sistema se realizó desde dos puntos de vista:

- 1) El primero de ellos está relacionado con el funcionamiento del sistema de comunicaciones, para el cual se tuvo en cuenta la cantidad de errores de transmisión por el canal de desvanecimiento y la relación señal a ruido del sistema; se tomaron medidas de error de estimación para niveles de ruido con el objetivo de analizar la adaptación del enlace de comunicaciones con adaptación según el canal estimado y según la predicción obtenida por los algoritmos implementados.
- 2) El segundo punto de vista está relacionado con el análisis a nivel de sistema digital; en este caso se tuvo en cuenta el diseño *Hardware / Software* para realizar medidas de latencia, tasa de transmisión, complejidad computacional y utilización de memoria para el sistema con estimación de canal y con algoritmos de predicción.

La figura 5.1 muestra el diagrama del sistema sobre el cual fueron realizadas las simulaciones de los algoritmos de predicción de canal; este sistema, cuya estructura fue descrita en la sección 3, cuenta con un módulo adición en el cual se realiza la predicción con base en la información estimada del canal en el receptor. La predicción es enviada a través de un canal dedicado al módulo de transmisión para que este se adapte de acuerdo a la medición de SNR.

Figura 5-1. Sistema MIMO-OFDM con predicción de canal en el receptor.



Para entender los resultados obtenidos se debe tener en cuenta que el canal Rayleigh usado en este trabajo da la ganancia o atenuación de los símbolos transmitidos, y esa ganancia debe ser estimada para decodificar los datos recibidos a través del arreglo de antenas MIMO; Esa estimación se hace a través de símbolos piloto y está sujeta, por el tipo de estimación realizada, al ruido gaussiano añadido en forma aditiva a la transmisión recibida por cada antena. A medida que aumenta el ruido la estimación se hace más errónea impidiendo una decodificación correcta. Adicional a esto, el canal estimado debe ser predicho para establecer la adaptación al enlace de comunicaciones y poder disminuir los errores transmitidos, pero esa predicción también está sujeta al ruido gaussiano aditivo. Por este motivo es que las mediciones de SNR son importantes para establecer el funcionamiento de los algoritmos implementados.

Los resultados mostrados en esta sección fueron obtenidos mediante la simulación del sistema MIMO-OFDM mostrado en la sección 3.2.

5.1 Validación del sistema MIMO-OFDM

Como ha sido mostrado en la sección 3 (tabla 3.1), el sistema de comunicaciones MIMO-OFDM descrito en este trabajo tiene una característica importante relacionada con la adaptación del enlace de comunicaciones a las variaciones del canal inalámbrico, como es la adaptación al canal de transmisión mediante esquemas de modulación M-QAM (con $M = 16, 64$ ó 256) y BPSK a utilizar de acuerdo al valor de SNR medido del canal estimado o predicho.

Con el objetivo de validar el sistema de transmisión, la simulación del canal inalámbrico y el sistema de recepción en base a los esquemas de modulación, el canal inalámbrico y la

modulación OFDM, se realizaron pruebas de generación de señales binarias aleatorias con distribución uniforme a las cuales se les aplicó el procesamiento de señales en términos de codificación, modulación y adición de ruido y ganancia de canal de acuerdo a parámetros específicos de frecuencia Doppler y trayectos de la señal. Los resultados del comportamiento del canal, mostrado por el sistema MIMO-OFDM desarrollado fueron comparados con los descritos en [51] [52] [53] [54] [55] y [56], demostrando que el sistema funciona correctamente dentro de los parámetros de modulación y codificación establecidos.

5.1.1 Comportamiento del canal inalámbrico

El canal inalámbrico fue generado usando el modelo Clarke's descrito en la sección 3.1.4, que utiliza la sumatoria de señales seno y coseno con frecuencia variable en el tiempo. La figura 5.2 muestra 1000 muestras o *samples* de la señal generada con valores absolutos de amplitud entre 0 y 2.6 y un valor de $f_d T_s = 0.1$, que representan un trayecto del canal MIMO; la potencia en dB para esta señal en particular varía entre los 10 y los -30 dBs como se muestra en la figura 5.3. La distribución de las muestras generadas con el modelo Clarke's es como se muestra en el histograma de la figura 5.4 que corresponde a una distribución Rayleigh.

Figura 5-2 Canal inalámbrico Rayleigh según modelo Clarke.

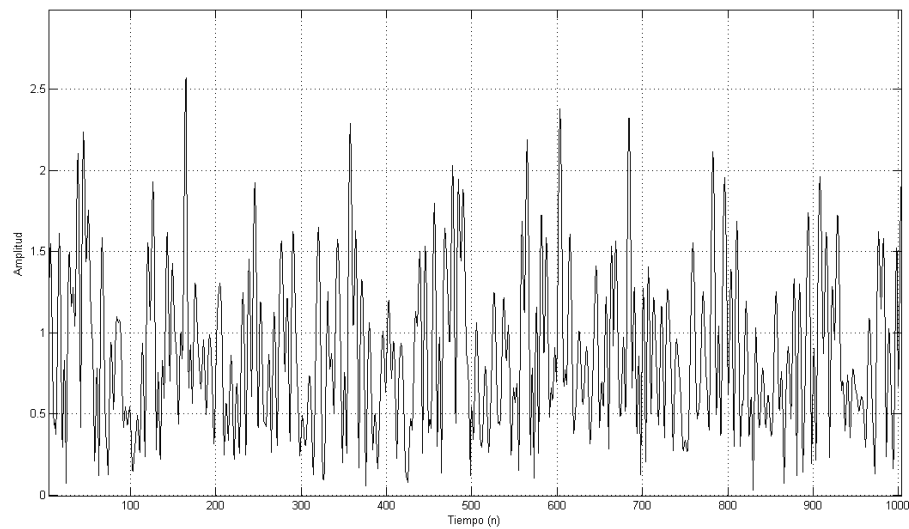


Figura 5-3 Potencia canal inalámbrico modelo Rayleigh.

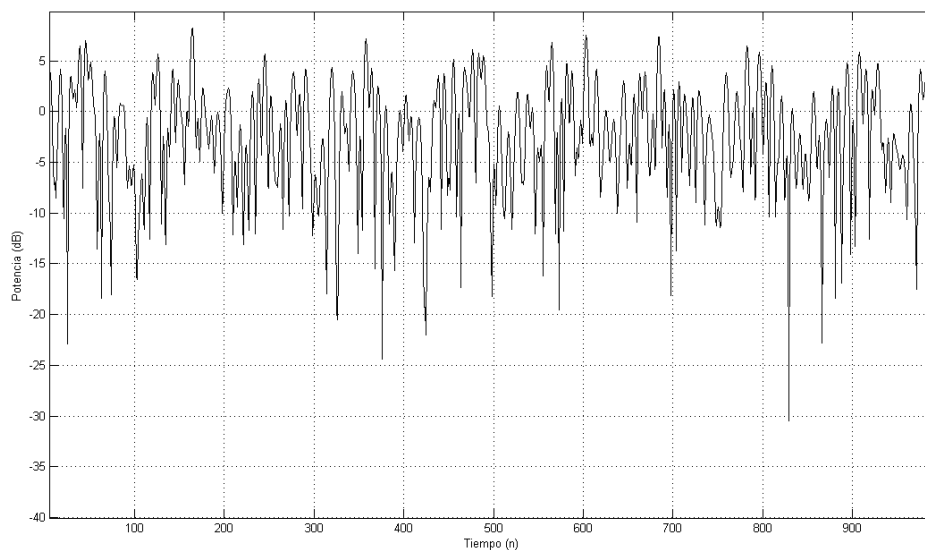
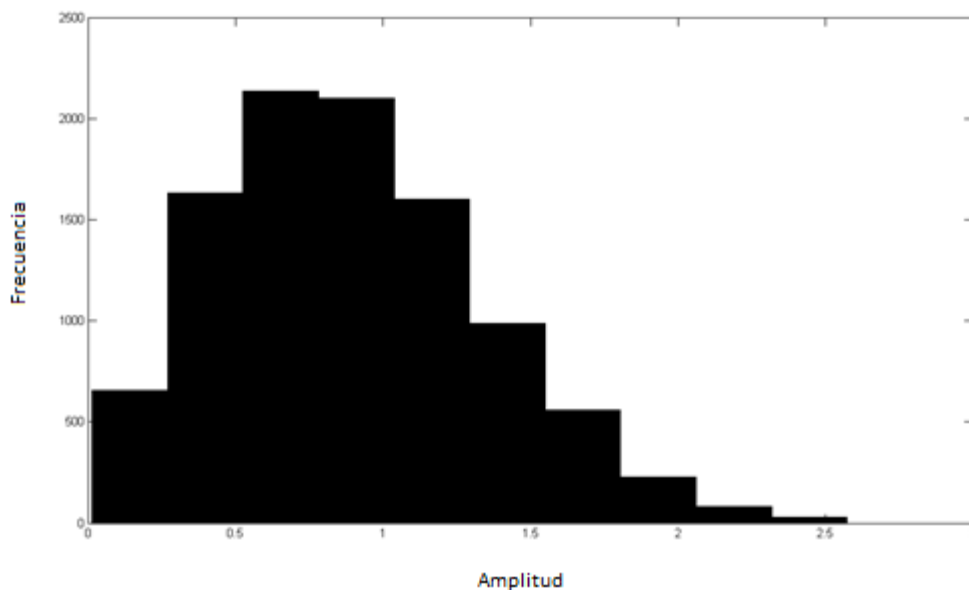
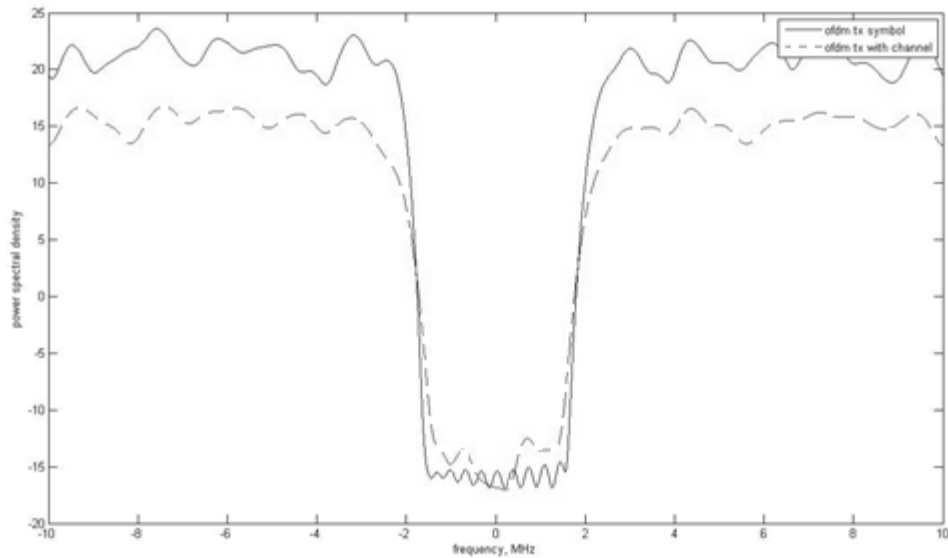


Figura 5-4 Distribución de canal inalámbrico.



5.1.2 Comportamiento del sistema con modulación OFDM

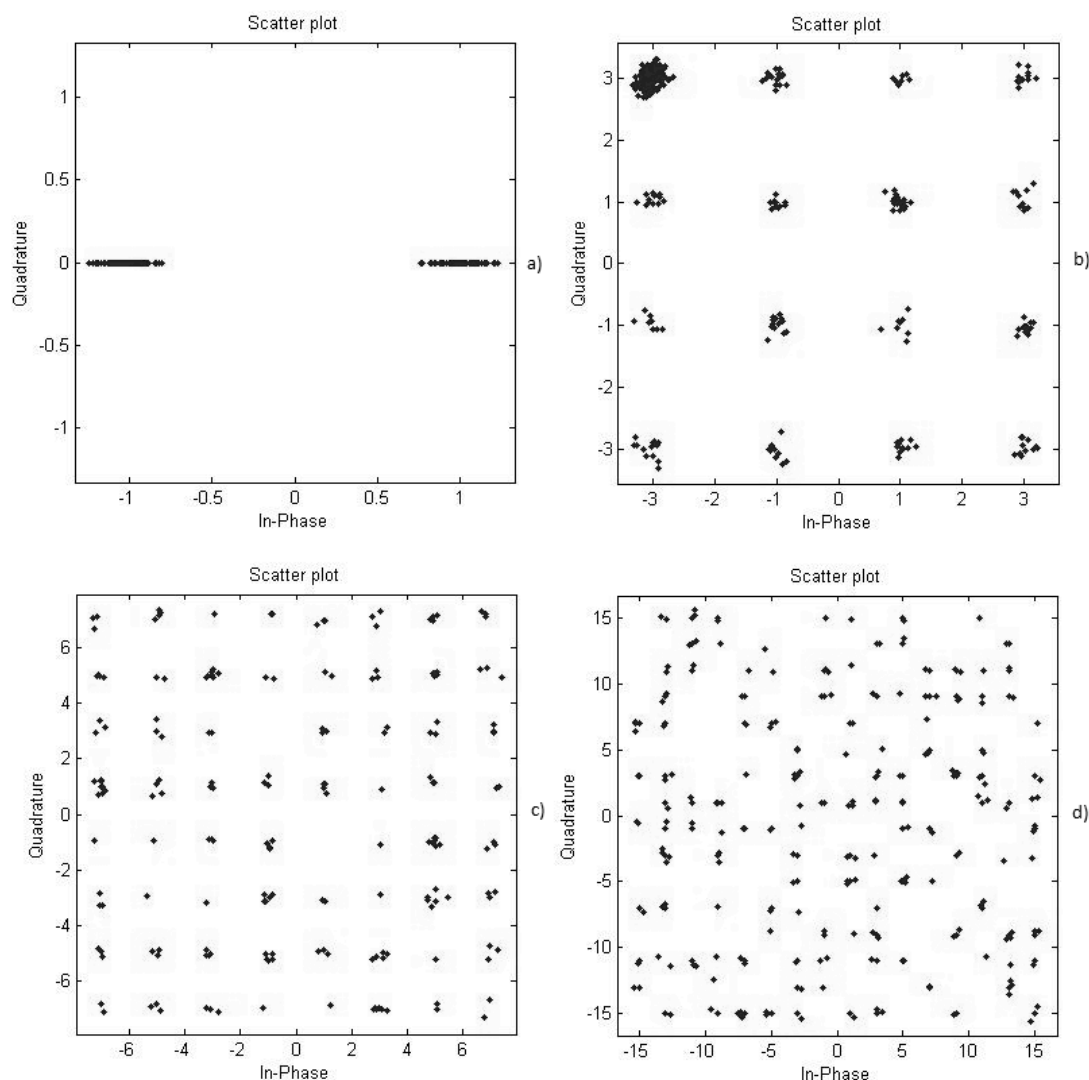
La figura 5.5 muestra cómo, al combinar la señal transmitida con las ganancias de canal inalámbrico generado en la sección 5.1.1 y ruido gaussiano aditivo, la señal codificada con modulación OFDM (señal sólida) pierde ortogonalidad y disminuye su potencia espectral al ser transmitida por el canal inalámbrico (señal punteada), lo que hace que al decodificarla en el receptor se presenten errores.

Figura 5-5 Potencia espectral de señal OFDM antes y después de transmisión.

5.1.3 Comportamiento del Sistema para Diferentes Esquemas de Modulación

Como ejemplo de la degradación dada por el ruido a la señal transmitida, la figura 5.6 muestra como una señal BPSK, 16-QAM, 64-QAM y 256QAM, con símbolos correspondientes a las constelaciones definidas, se ve afectada por el ruido haciendo que los símbolos se dispersen; los valores de ruido tiene un valor de E_b/N_0 igual a 0. Este resultado es de gran importancia al momento de establecer la capacidad de adaptación del sistema al canal de comunicaciones, ya que como se puede ver en la figura, al aumentar el número de símbolos de la constelación la dispersión de estos es tal, que se hace más difícil demodular sin errores.

Figura 5-6 Grafica de dispersión de símbolos con $E_b/N_0=0$ dB; a) BPSK, b) 16-QAM, c) 64-QAM, d) 256-QAM.



El canal inalámbrico tiene 2 componentes, uno de estos es la ganancia que está dada por las trayectorias de canal MIMO (2X2), y el otro dado por el ruido gaussiano aditivo. De acuerdo a esto, las figuras 5.7, 5.8, 5.9 y 5.10 muestran la degradación en dB de la transmisión de la señal para modulación BPSK, 16-QAM, 64-QAM, y 256-QAM después de ser decodificadas, primero asumiendo conocimiento perfecto del canal de comunicaciones (señal de línea punteada) y segundo haciendo una estimación de canal (señal de línea sólida) con la técnica *Least Square* de la forma mostrada en el apartado 2.3.2. La señal azul muestra una degradación significativa al aumentar la potencia del ruido gaussiano; sin embargo, la degradación más importante se observa al hacer la estimación de canal dado que esta estimación resulta más y más ruidosa al aumentar la potencia de ruido gaussiano, lo que afecta directamente la decodificación de la señal resultando en errores de transmisión.

Figura 5-7 Grafica de BER para modulación BPSK con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).

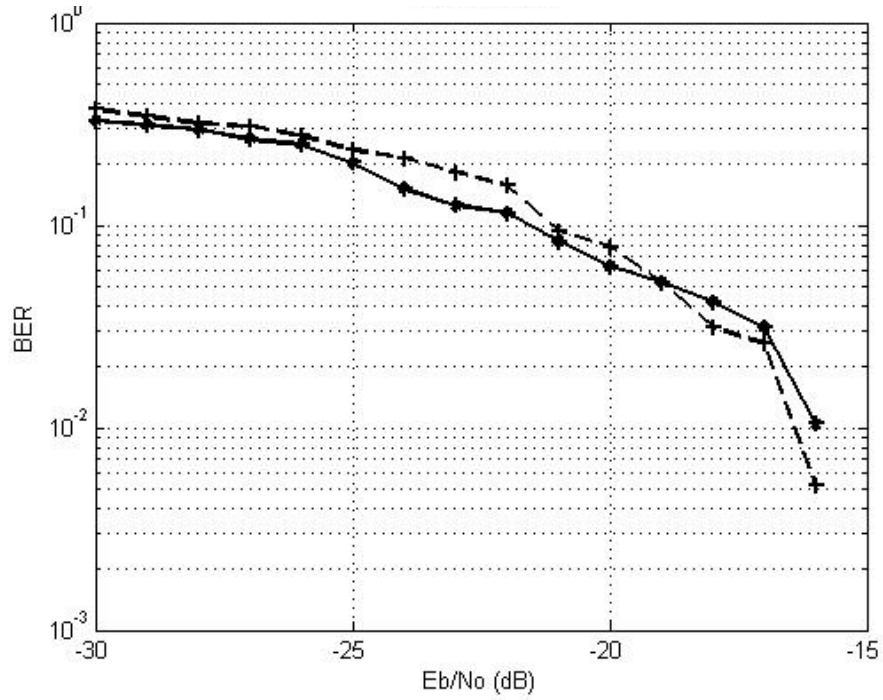


Figura 5-8 Grafica de BER para modulación 16-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).

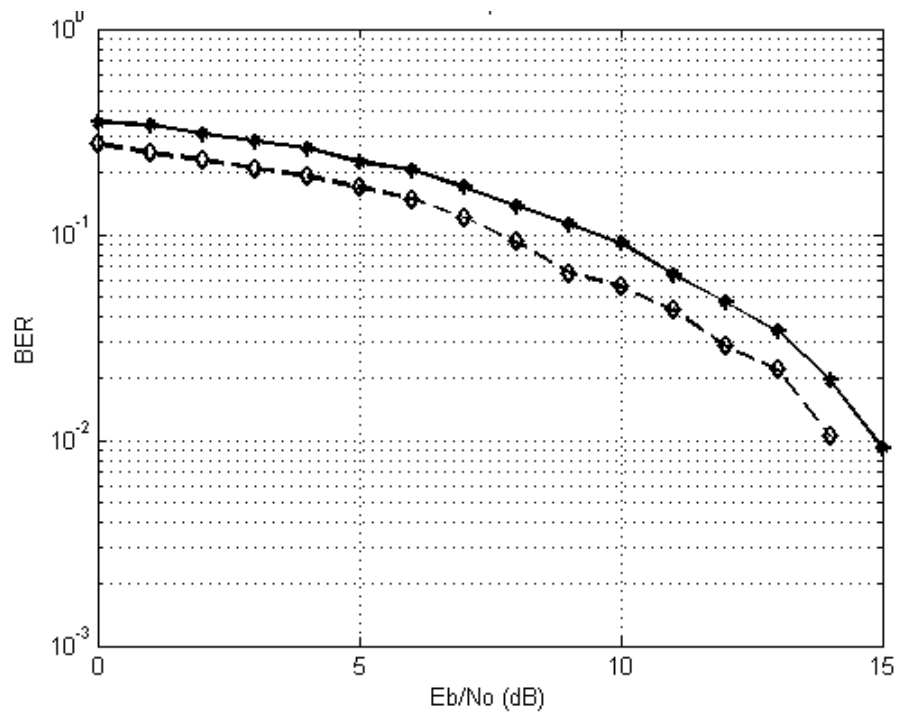


Figura 5-9 Grafica de BER para modulación 64-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).

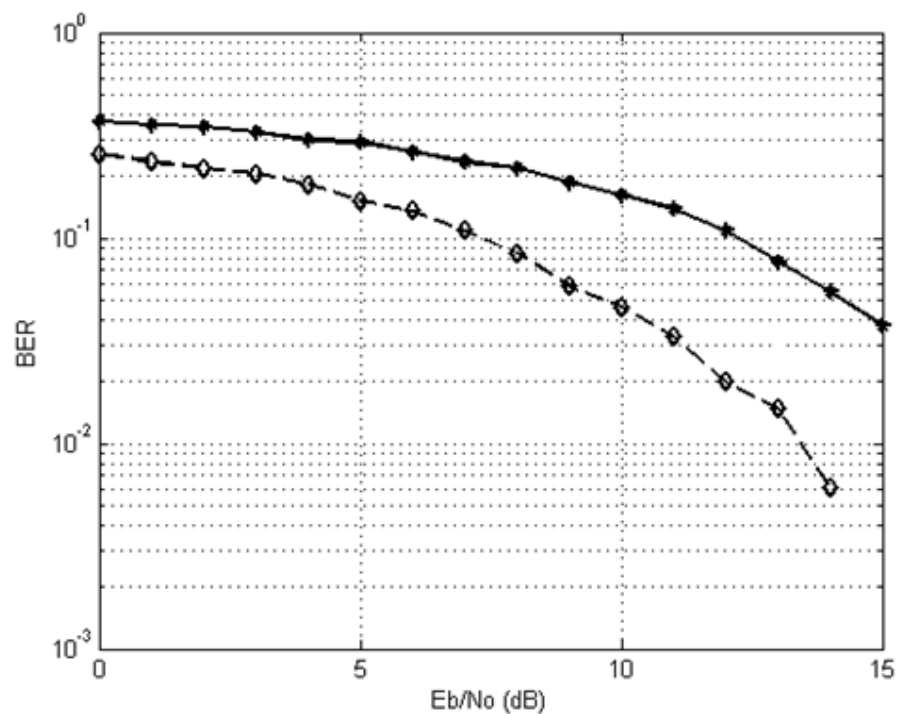
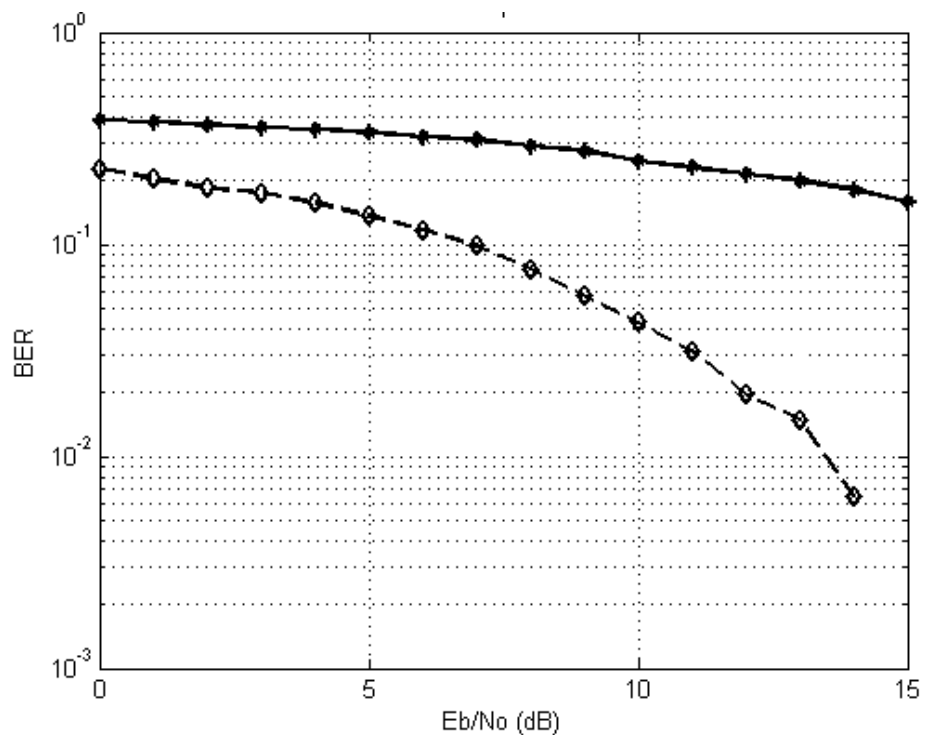


Figura 5-10 Grafica de BER para modulación 256-QAM con asumiendo canal perfecto (línea punteada) y con estimación de canal (línea solida).



Al comparar las figuras se puede notar que la degradación se hace más importante al aumentar el valor M de modulación M-QAM por lo cual se hace evidente que, a pesar de que el esquema 256-QAM tiene una mayor tasa de transmisión es mucho más sensible al ruido. En cambio, para el caso de modulación con esquema BPSK no ocurre un cambio mayor entre la medida de BER asumiendo canal perfecto o con estimación de este, debido a que la codificación Alamouti aplicada para MIMO 2 x 2 tiene un funcionamiento adecuado en símbolos modulados con parte real e imaginaria, que no es el caso de la modulación BPSK. Sin embargo, se asumió ésta como la forma de modulación más apropiada para niveles altos de ruido ya que presenta el mayor espacio entre símbolos de la constelación, lo que disminuye la posible cantidad de errores al demodular en el receptor.

Tomando en cuenta la información obtenida con respecto a la transmisión de datos y los esquemas de modulación se diseñó un algoritmo de adaptación al canal de transmisión de tal forma que se encontrará una relación favorable entre la tasa de transmisión y tasa de errores de bit. La tabla 5.1 muestra el algoritmo implementado para obtener una tasa de error de bit (BER) media de 0.05. En este caso, y tomando en cuenta que el canal gaussiano aditivo fue modelado con niveles de E_b/N_0 entre 0 y 25 dBs también utilizó la modulación BPSK, ya que esta es la más robusto de los esquemas de modulación, pero su tasa de transmisión de datos es la más baja (solo un bit por símbolo).

De esta forma, se estableció que para valores de SNR altos estimados según el canal, la tasa de transmisión es mayor, y para valores pequeños de SNR la tasa de transmisión será menor pero más confiable.

Tabla 5-1 Algoritmo de adaptación de canal para sistema transmisor.

```

si (SNR_dB >= 21)
{
    M = 256;
}
sino ( si (SNR_dB < 21 && SNR_dB >= 15) )
{
    M = 64;
}
sino ( si (SNR_dB < 15 && SNR_dB >= 10) )
{
    M = 16;
}
sino ( si (SNR < 5) )
{
    M = 2;
}

```

En esta sección se pudo comprobar que el sistema MIMO-OFDM descrito en *SystemC* como se mostró en la sección 3.2 funciona de forma correcta con tasas de error de bit entre 0.01 y 0.1 para los diferentes esquemas de modulación con estimación de canal, teniendo en cuenta la degradación de la señal a causa del canal de comunicaciones Rayleigh simulado. Adicional a esto, se logró establecer un algoritmo de adaptación del

sistema a las condiciones del canal inalámbrico de acuerdo a los esquemas de modulación usados.

5.2 Configuración de Algoritmos de predicción de Canal

El análisis comparativo entre algoritmos de predicción de canal inicia con el desarrollo y configuración de estos a partir de parámetros modificables, como son el orden de modelo AR para los lineales y la cantidad de capas de entrada y salida para los no lineales, que permita hallar para cada uno de los algoritmos el funcionamiento más apropiado, esto es, la predicción con menor error o más precisa. Se estableció una medida de error para cuantificar el desempeño de los algoritmos con diferentes configuraciones; esta medida es llamada *Mean Square Error* o error cuadrático medio (MSE), y resulta de gran ayuda al momento de cuantificar la diferencia entre un valor estimado y el verdadero valor objeto de estimación, descrito como: $MSE = \frac{1}{n} \sum_{i=0}^n (\hat{y}_i - y_i)^2$. Se realizaron ejecuciones con diversas configuraciones durante 200 espacios de tiempo y se calcularon valores de MSE con un modelo de canal específico con los siguientes parámetros de canal inalámbrico:

- 1) El parámetro de canal relacionado con la frecuencia de dispersión Doppler (f_d) $f_d T_s = 0.1$.
- 2) La potencia de ruido aditivo $E_b/N_0 = 0$.

Las tablas 5.2, 5.3, 5.4, y 5.5 muestran algunas de las configuraciones con resultados más significativos sobre las evaluadas para los algoritmos de predicción, cuyos parámetros fueron variados con el objetivo de encontrar la configuración más apropiada en términos de desempeño.

El algoritmo de predicción lineal con modelo autorregresivo (AR) tiene como parámetro de configuración variable el orden del modelo AR; este algoritmo presenta uno de los errores MSE más altos entre todos los evaluados, y aunque es una de las formas más sencillas de predicción, allí no se toma en cuenta la no linealidad ni los niveles de ruido de la señal lo cual afecta su capacidad de predicción. En este caso observamos una disminución importante en el error al aumentar el orden del modelo tomado en cuenta; esto es porque al tomar mayor cantidad de muestras anteriores del canal, se puede obtener una mayor información estadística sobre este; sin embargo se llega a un punto estable en el que a pesar de que se aumente el orden del modelo el error únicamente disminuye en 0.0033 con respecto al orden anterior, como se puede ver para los valores de error del orden 20 y 25. Mostrados en la tabla 5.2 En este caso tomamos el orden 20 como la configuración más apropiada para este algoritmo.

Tabla 5-2 Error MSE para Predictor con modelo AR.

	Error MSE
orden = 5	0.2636
orden = 10	0.2139
orden = 15	0.1986
orden = 20	0.1897
orden = 25	0.1864

En el filtro kalman el parámetro de configuración variable es el orden de modelo tomado en cuenta para la formación de la matriz A , al igual que para el algoritmo con modelo AR. Es necesario tomar en cuenta que, dado que el canal varía en el tiempo de forma rápida tanto la matriz A , como la covarianza del error ruido y del error del modelo mismo debe ser recalculada cada n espacios de tiempo aumentando la complejidad del algoritmo, por lo que se tomaron en cuenta modelos de orden cinco, diez, quince, veinte y veinticinco, que pueden representar el modelo de forma adecuada, como se vio para el algoritmo con modelo AR. De acuerdo a las ecuaciones del capítulo 4.1.2 donde se muestra el algoritmo implementado, se puede notar que al aumentar el orden de dicho modelo aumenta el tamaño de las matrices X, P, C y Mn , por lo que la complejidad aumenta sin que necesariamente aumente la precisión de la predicción de éste de forma significativa con una disminución del error de solo 0.0020 aproximadamente entre un orden y otro a partir del orden = 10, como se puede ver en la tabla 5.3. Para este caso, y a pesar de que presento un error intermedio entre las configuraciones evaluadas, se tomó el modelo de orden diez para equilibrar de alguna forma la relación complejidad – desempeño.

Tabla 5-3 Error MSE para Filtro Kalman.

	Error MSE
orden = 5	0.1822
orden = 10	0.1734
orden = 15	0.1731
orden = 20	0.1619
orden = 25	0.1598

La tabla 5.4 muestra los resultados del cálculo MSE para la predicción basada en red neuronal con n retrasos en entrada, donde se puede observar que el mejor comportamiento lo obtuvo la red neuronal con 20 retrasos de entrada y una capa intermedia de cuarenta neuronas para un valor de MSE = 0.0875. Podemos notar que utilizar una cantidad pequeña de neuronas produce un error alto, y además, que no necesariamente el aumento en el número de neuronas significa una mejora en el comportamiento de la red. Tenemos entonces que la relación número de neuronas – número de entradas no es lineal sino que puede ser una cuestión de ensayo y error, hasta encontrar la mejor relación posible; sin embargo, en este trabajo se siguió una

relación $N_{neuronas} = 2 * N_{entradas}$ hasta llegar a la solución encontrada, para una configuración de red con error igual a 0.0875.

Tabla 5-4 Error MSE para estructuras de Red Neuronal con Retraso en Entrada.

	Error MSE
nInputDelays = 30 intermediateLayerSize = 60	0.1497
nInputDelays = 20 intermediateLayerSize = 40	0.0875
nInputDelays = 10 intermediateLayerSize = 20	0.1041
nInputDelays = 5 intermediateLayerSize = 10	0.0896
nInputDelays = 2 intermediateLayerSize = 6	0.0908

En el caso de la red neuronal recurrente cuyos resultados se muestran en la tabla 5.5, observamos un comportamiento similar al de la red mostrada anteriormente, con una disminución en el error obtenido en configuraciones similares. Esto es debido a la realimentación de la salida que tiene la red neuronal recurrente, la cual permite recopilar información estadística importante, que hace que se obtengan resultados adecuados, con una menor cantidad de entradas y neuronas en la capa intermedia. Para la red neuronal recurrente encontramos que la configuración adecuada es de diez retrasos en la entrada y veinte neuronas en la capa intermedia, para un error obtenido de 0.0811. Se puede notar que tanto la disminución en la cantidad de neuronas en la capa intermedia como la cantidad de retrasos en la entrada es bastante significativa con respecto a la red neuronal con retrasos en la entrada.

Tabla 5-5 Error MSE para estructuras de Red Neuronal Recurrente.

	Error MSE
nInputDelays = 30 intermediateLayerSize = 60	0.1434
nInputDelays = 20 intermediateLayerSize = 40	0.1195
nInputDelays = 10 intermediateLayerSize = 20	0.0811
nInputDelays = 5 intermediateLayerSize = 10	0.0842
nInputDelays = 2 intermediateLayerSize = 6	0.1757

5.3 Desempeño de Esquemas de Predicción

Con el objetivo de evaluar el comportamiento del sistema dentro de un ambiente aleatorio tomamos mediciones de la predicción de canal en el sistema de comunicaciones MIMO-

OFDM con niveles variables de ruido gaussiano aditivo y frecuencia Doppler de $f_d T_s = 0.1$ y $E_b/N_0 = 0$. Los resultados obtenidos corresponden a la predicción realizada por los algoritmos con las configuraciones escogidas en la sección 5.2, y se muestran en las figuras 5.11 (predicción lineal con modelo AR), 5.12 (predicción con filtro Kalman), 5.13 (Predicción no lineal con red neuronal con retrasos en la entrada) y 5.14 (Predicción no lineal con red neuronal recurrente).

Dado que la predicción del canal se realiza principalmente en base a muestras anteriores del mismo, existe un tiempo de ajuste diferente para cada algoritmo; en el caso de los algoritmos lineales, con modelo AR y filtro kalman, depende del orden de modelo por lo que se necesitan al menos 20 espacios de tiempo y 10 espacios de tiempo respectivamente. En el caso de los algoritmos no lineales tomados en este trabajo, el tiempo de ajuste depende del número de retrasos de entrada que es 20 para la red neuronal con retrasos de entrada y 10 para la red neuronal recurrente.

Figura 5-11 Predicción lineal con modelo AR.

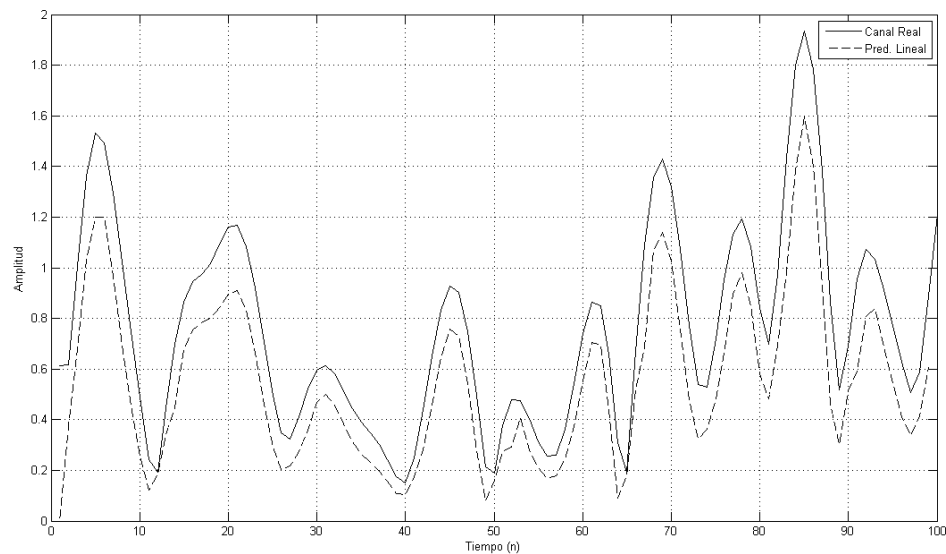


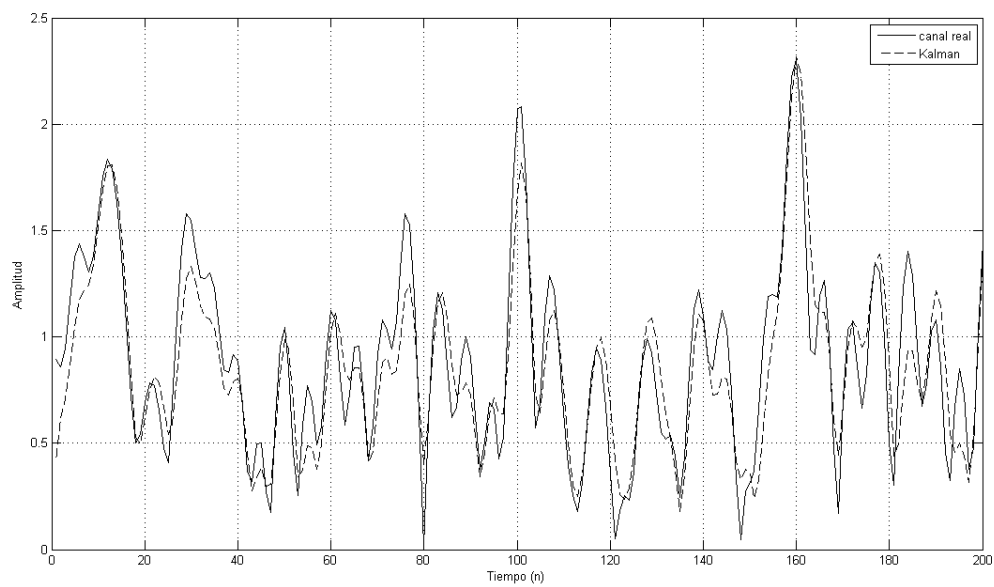
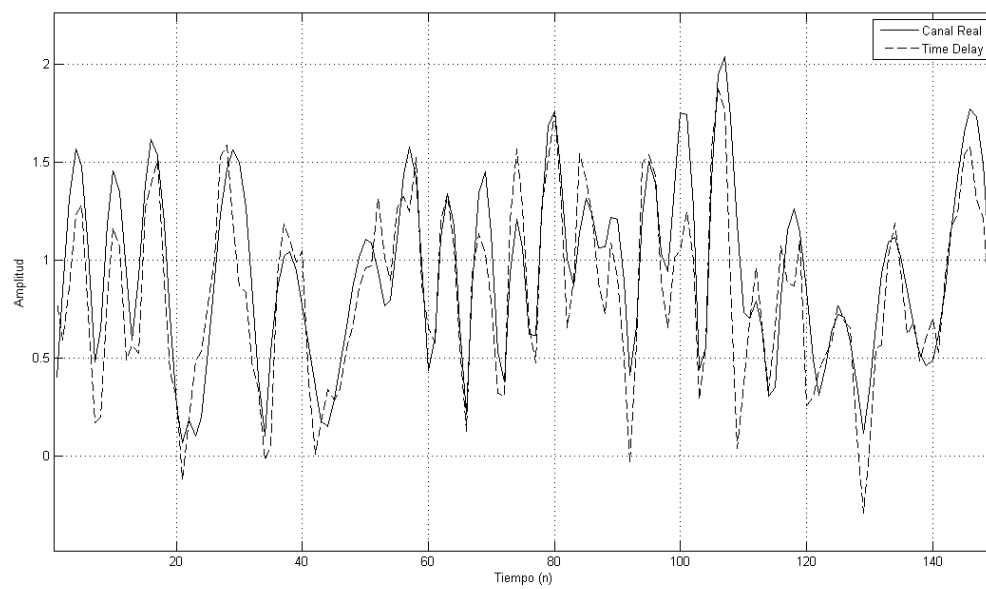
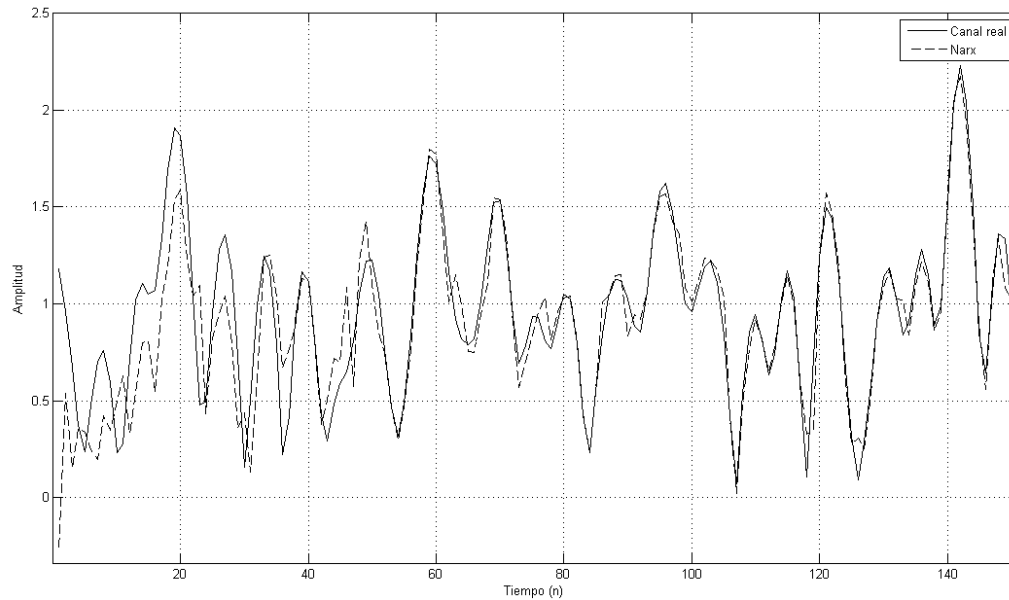
Figura 5-12 Predicción con filtro kalman.**Figura 5-13 Predicción con red neuronal con n retrasos en la entrada.**

Figura 5-14 Predicción Con Red neuronal recurrente.

De las tablas 5.2 a 5.5 y las figuras 5.11 a 5.14 se puede concluir que los algoritmos no lineales, presentan el mejor desempeño dadas las condiciones variables del canal y la adición de ruido. Esto puede ser porque, dadas las características no lineales del canal, la mejor forma de predecirlo es utilizando técnicas no lineales, que permitan recopilar la mayor cantidad de información estadística de este y la procesen de tal forma que los resultados sean lo suficientemente buenos para obtener una mejora en la adaptación del sistema.

Dado que el canal cambia de forma rápida, los parámetros de procesamiento de los algoritmos se desactualizan rápidamente, lo que hace que se observe una disminución en el desempeño de estos. Este problema puede solucionarse cambiando los modelos base de cada algoritmo en tiempo real así: en el caso del filtro kalman un cambio en el modelo A, y en el caso de las redes neuronales un nuevo entrenamiento con valores actualizados de canal.

Los niveles de ruido gaussiano aditivo también afectan la precisión en la predicción; por ejemplo, el modelo AR no toma en cuenta estadísticas de ruido por lo que puede tender a ser más sensible a este, y la predicción basada en redes neuronales usa datos previos del canal para entrenar la red por lo que si diversos niveles de ruido no son tomados en cuenta la predicción no será correcta.

Los algoritmos de predicción presentados en esta sección permiten hacer un análisis comparativo entre diferentes configuraciones de estos y además permiten constatar que existe la posibilidad de realizar una predicción de canal con diversos de niveles de precisión que están relacionados con la complejidad del algoritmo mismo.

5.4 Desempeño de sistema de comunicaciones MIMO-OFDM

En esta sección se analiza el sistema desde el punto de vista de procesamiento de señal en un sistema de comunicaciones, para lo que se toma en cuenta medidas de tasa de error de bit (BER) y relación señal a ruido (SNR) usadas comúnmente como medida de desempeño en sistemas de transmisión y recepción.

La tasa de error de bit o BER indica en número de bits erróneos recibidos durante la transmisión de una trama de datos binarios, obtenidos a causa de ruido e interferencia.

La medida de BER está dada de la siguiente forma: $BER = \frac{\text{Numero de bits errados}}{\text{Numero de bits transmitidos}}$.

Por otro lado, la relación señal a ruido o SNR es una medida usada para comparar los niveles de potencia de ruido con la potencia de la señal transmitida, y es expresada generalmente en decibeles. La medida de SNR está dada generalmente de la forma: $SNR = 10 \log_{10} \frac{\text{Potencia de la señal}}{\text{Potencia del ruido}} \text{ dB}$; sin embargo, existe una forma alternativa de calcular la medida instantánea de SNR usando la ecuación (5.1) que define el coeficiente de variación como la tasa de desviación estándar (σ) con respecto a la media de la señal transmitida (μ), y es la medida base usada en este trabajo [25]:

$$SNR = \frac{\mu}{\sigma} \quad (5.1)$$

5.4.1 BER Según Medición SNR para Estimación y Predicción de Canal:

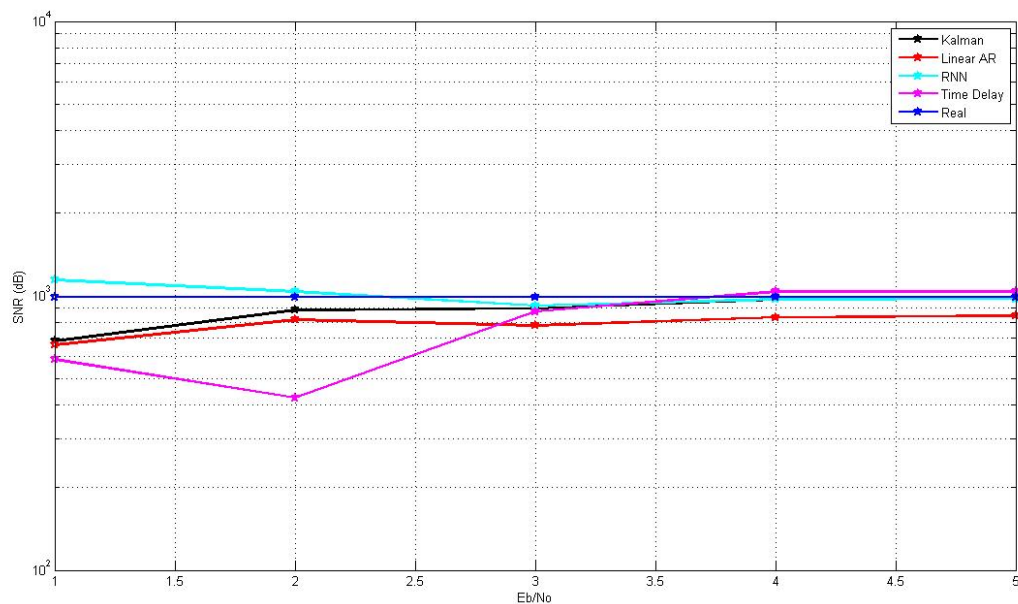
Con el objetivo de comparar el sistema en las configuraciones de los algoritmos de predicción propuestas en la sección 5.2, se realizaron mediciones de relación señal a ruido SNR y tasa de error en la transmisión en términos de error de bits BER durante 100 espacios de tiempo para dar un rango amplio de procesamiento del algoritmo de adaptación. Los resultados son mostrados en la tabla 5.6 para un valor $f_d T_s = 0.1$, donde se dan los valores obtenidos de acuerdo con las configuraciones: sin predicción, predicción lineal con modelo AR, predicción con filtro kalman, predicción con red neuronal recurrente, predicción con red neuronal con retrasos de entrada. La tabla 5.7 muestra igualmente medidas de SNR y BER para los algoritmos de predicción; en este caso se tomó un valor de $f_d T_s = 0.3$ que permite evaluar la capacidad de adaptación de los algoritmos a las variaciones de dispersión Doppler (f_d) del canal de comunicaciones.

Siguiendo los cálculos para SNR dados en [59] [60] para canales MIMO con múltiples trayectos, la figura 5.14 muestra la desviación entre el valor de SNR medido sobre la predicción realizada por cada algoritmo y el valor real, al aumentar E_b/N_0 del ruido gaussiano aditivo. Allí se puede observar que el error aumenta a medida que el nivel de ruido aumenta y esto se traduce en un funcionamiento erróneo de la adaptación de canal.

Los algoritmos lineales tuvieron un comportamiento similar al de la red neuronal recurrente manteniendo valores constantes aunque no precisos. El algoritmo con modelo AR adaptó el modelo de canal al ruido aditivo y dio un valor de SNR siempre inferior al real; el filtro kalman presento el mejor comportamiento al mostrar error en la medición de SNR a partir de $E_b/N_0 = 3$.

Con respecto a los algoritmos no lineales se pueden ver comportamientos diferentes a pesar de que sus estructuras y lógica son similares entre sí: la red neuronal con retrasos de entrada muestra el peor comportamiento de los evaluados, afectándose significativamente a partir de $E_b/N_0 = 2$; en cambio la red neuronal recurrente mantuvo un valor de SNR superior a la medida original, pero consistente en el tiempo.

Figura 5-15 E_b / N_0 Vs. SNR de algoritmos de predicción.

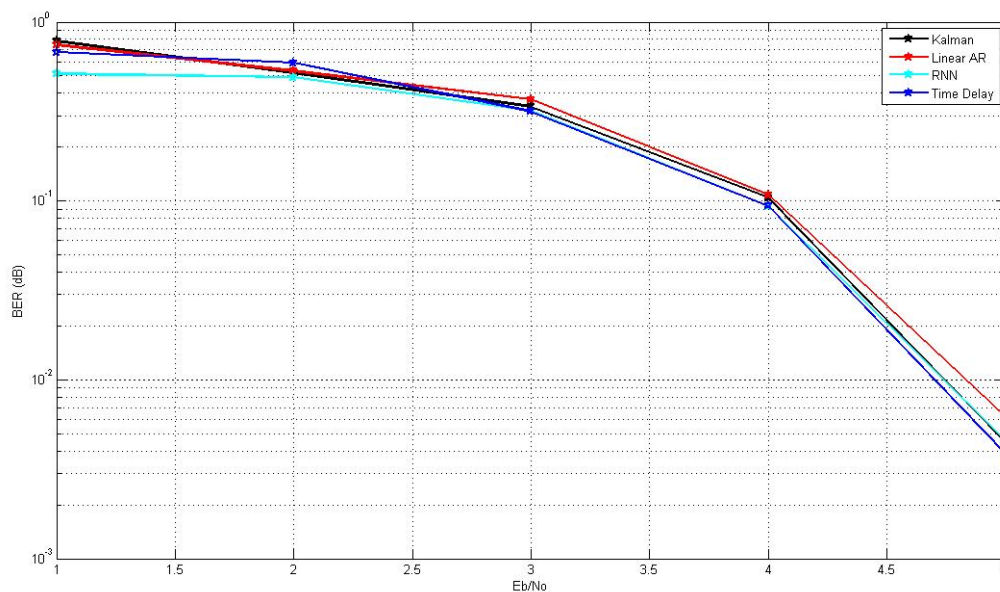


Podemos notar de la figura 5.15 que a pesar de que el algoritmo lineal con modelo AR tiene una complejidad menor a la de los demás algoritmos la respuesta en el tiempo es consistente, lo que indica un buen nivel de adaptación a cambios en los parámetros de ruido. Este no es el caso para el algoritmo de filtro Kalman que, a pesar de su capacidad para filtrar el ruido introducido en la estimación del canal, al variar parámetros de ruido o las características del canal mismo, el modelo es desactualizado y su desempeño decrece.

La figura 5.16 muestra la medición de BER para cada uno de los esquemas de predicción lineal y no lineal para diferentes valores de E_b/N_0 del ruido gaussiano aditivo. Allí se puede observar que para un mayor nivel de error el algoritmo lineal con modelo AR presenta un nivel de error mucho mayor a los demás algoritmos; en cambio, la menor

tasa de errores fue presentada por el algoritmo no lineal basado en red neuronal con entrada retrasada.

Figura 5-16 Eb/No Vs. BER de esquemas de algoritmos de predicción.



En las tablas 5.6 y 5.7 se puede observar que la tasa de errores en la transmisión medida para los algoritmos no lineales disminuye con respecto al caso de la no predicción siempre y cuando las características del canal sean determinadas correctamente. En la tabla 5.6 se tomaron mediciones de SNR y BER sobre el canal correspondiente al conjunto de datos utilizado para entrenar las redes neuronales y se encontró un mejoramiento en el rendimiento del sistema; sin embargo, al cambiar las características del canal de forma drástica, como es dando un mayor valor de la frecuencia Doppler, se encontró que las redes neuronales presentaron la mayor cantidad de errores de transmisión con un aumento de cerca del 38% con valores de 0.0890 hasta 0.1236 para la red neuronal con retraso en la entrada y 0.0889 hasta 0.1516 para la red neuronal recurrente respectivamente; este resultado es incluso mayor que el dado por los algoritmos lineales que presentaron un aumento de cerca de 2.5%, con valores de 0.0965 hasta 0.0995 para el modelo AR y de 0.0921 hasta 0.0987 para el filtro kalman.

Se puede decir también que los algoritmos lineales y no lineales tienden a presentar error en la predicción, ya sea una predicción mayor o menor al valor real, al aumentar la frecuencia Doppler, lo que se traduce también en una mayor tasa de errores o menor tasa de transmisión de datos, cuyo caso crítico sería una modulación BPSK fija, para lo cual no tendría sentido el algoritmo de adaptación de enlace sobre todo porque de cualquier forma no existe un esquema de modulación totalmente inmune al ruido.

Tabla 5-6 Mediciones de BER y SNR para configuraciones del sistema $f_d T_s = 0.1$.

	Sin predicción	predicción lineal con modelo AR	predicción con filtro kalman	predicción con red neuronal con retrasos de entrada	predicción con red neuronal recurrente
SNR medio (dB)	9.66	8.45	8.50	9.23	10.56
BER medio	0.1021	0.0965	0.0921	0.0890	0.0889

Tabla 5-7 Mediciones de BER y SNR para configuraciones del sistema $f_d T_s = 0.3$.

	Sin predicción	predicción lineal con modelo AR	predicción con filtro kalman	predicción con red neuronal con retrasos de entrada	predicción con red neuronal recurrente
SNR medio (dB)	9.51	9.15	9.89	6.89	3.80
BER medio	0.0975	0.0995	0.0987	0.1236	0.1576

Los efectos de una predicción imprecisa resultan en una mayor cantidad de errores en la transmisión o una menor tasa de transmisión de datos. A medida que el nivel de ruido, y por lo tanto el error en la medición de SNR aumenta, la adaptación debería tender a la modulación BPSK pues esta presenta la mayor protección a ruido. Sin embargo, la red neuronal con retraso en el tiempo tiende a adaptar el sistema entre las modulaciones 64 y 256 QAM porque la predicción indica que el ruido disminuye; en cambio, la red neuronal recurrente tiende a mantener la adaptación de link en modulaciones BPSK y 16-QAM porque sus valores de SNR siempre superiores a la medida real. Por otro lado, el filtro kalman y el algoritmo con modelo AR no muestra una tendencia de adaptación y su transmisión de datos es constante, aunque con mayor cantidad de errores.

El algoritmo con modelo AR tiene un mayor nivel de afectación al ruido porque este no se determina durante la predicción; cómo podemos ver en la tabla 5.6 la tasa de error de bit medida en el receptor es de 0.0965 y la figura 5.15 demuestra una mayor tasa de error con respecto a los demás algoritmos, en cambio el filtro kalman toma en cuenta la varianza de ruido lo cual le da una mayor resistencia a este; sin embargo, esta se desactualiza constantemente por lo que podría ser recalculada cada n espacios de tiempo dependiendo de las características del ruido para que el algoritmo se ajuste apropiadamente.

Por otro lado los algoritmos no lineales presentados en este trabajo, que corresponden a las redes neuronales en diferentes estructuras tienen la habilidad de adaptarse adecuadamente a cualquier característica del canal siempre y cuando sean entrenadas con un set de datos amplio, que permita recopilar información estadística sobre el ruido.

Como se vio en las tablas 5.6 y 5.7 se puede concluir que la predicción puede no ser precisas o ser completamente errónea dando un aumento en la tasa de errores recibidos, incluso mayor que para el sistema sin predicción de canal. Además, la adaptación del sistema depende directamente de la medición de SNR, y como se pudo ver para valores altos de ruido gaussiano aditivo y cambios drásticos en el modelo del canal, esta medida resulta errónea haciendo que el sistema se adapte de tal forma que se presente mayor cantidad de errores o una menor tasa de transmisión (figura 5.15).

5.5 Análisis de Desempeño del Sistema Digital

El segundo punto de vista para evaluar el desempeño del sistema MIMO-OFDM en sus diferentes configuraciones, y sobre todo los algoritmos de predicción y su efecto en el mismo sistema, está relacionado con el análisis a nivel de sistema digital; en este caso se tuvo en cuenta las características propias del diseño *Hardware / Software*, como uso de dispositivos de memoria, concurrencia del sistema y operaciones de los algoritmos, para realizar medidas de latencia y tasa de transmisión para el sistema sin estimación de canal, con estimación y con algoritmos de predicción. También se evaluaron los algoritmos de predicción en términos de complejidad computacional y utilización de memoria.

5.5.1 Tasa de transmisión de datos:

La tasa de transmisión de bits fue evaluada durante 100 espacios de tiempo y para diferentes condiciones del canal de comunicaciones, objeto de predicción, de forma similar a la sección 5.4.1. La tasa de transmisión fue medida para 100 espacios de tiempo con las características de canal $f_d T_s = 0.1$, $f_d T_s = 0.3$ y un nivel de ruido gaussiano aditivo $E_b/N_0 = 0$.

La tasa máxima de transmisión del sistema es de 153 600 bits (teórica), que corresponde a una transmisión con el esquema de modulación con mayor cantidad de bits por símbolo, ósea 256-QAM, durante los 100 espacios de tiempo durante los cuales se realizó el cálculo. Sin embargo, esta tasa de transmisión nunca se alcanza dado que los niveles de ruido lo no permiten, en cambio el sistema debe adaptarse a otros esquemas de modulación, por lo cual la tasa de transmisión máxima lograda fue de 137.856 con la red neuronal recurrente.

Los resultados medidos de tasa de transmisión de datos del sistema con cada algoritmo se muestran en la tabla 5.8 donde se observa que la mayor tasa de transmisión está dada por el algoritmo no lineal implementado con red neuronal recurrente seguido por el algoritmo lineal de filtro Kalman; esto puede ser comparado con la tasa de errores de transmisión medida en la sección 5.4.1, que resultó ser mayor para los algoritmos lineales.

En cambios significativos de canal como en la dispersión Doppler, con $f_d T_s = 0.3$, la mayor tasa de transmisión la presento el algoritmo lineal con modelo AR seguido del filtro Kalman; por el contrario los algoritmos no lineales – las redes neuronales en diferentes configuraciones – mostraron una tasa de transmisión de cerca de la mitad de la dada por los algoritmos lineales; además, como se muestra en la tabla 5.7, la tasa de error de bit es mucho mayor en los algoritmos no lineales.

Tabla 5-8 Tasa de transmisión de sistema MIMO-OFDM con predicción de canal. $f_d T_s = 0.1$ para 100 espacios de tiempo.

Algoritmo	Tasa de transmisión de bits
Predicción lineal con modelo AR	96 192
Filtro Kalman	110 784
Red neuronal con retrasos en entrada	108 096
Red neuronal recurrente	137 856

Tabla 5-9 Tasa de transmisión de sistema MIMO-OFDM con predicción de canal. $f_d T_s = 0.3$ para 100 espacios de tiempo.

Algoritmo	Tasa de transmisión de bits
Predicción lineal con modelo AR	131 520
Filtro Kalman	107 904
Red neuronal con retrasos en entrada	50 880
Red neuronal recurrente	62 208

5.5.2 Latencia de sistema:

Con el objetivo de establecer la latencia del sistema se midió el tiempo que este toma en codificar y modular la señal en el transmisor, enviar por el canal de comunicaciones, decodificar y demodular en el receptor, predecir el canal de comunicaciones y finalmente obtener información de este para adaptar el sistema a las condiciones dadas. Estas medidas de latencia fueron tomadas para el sistema con y sin algoritmos de predicción.

Las medidas obtenidas en este apartado fueron realizadas en condiciones estables y con los mismos parámetros de procesamiento para un espacio de tiempo en el que se transmite un símbolo OFDM por cada antena de transmisión. Cabe recordar que las antenas en este caso no aumentan la tasa de transmisión, sino que gracias a la codificación Alamouti realizada en el transmisor, las antenas ayudan a disminuir los errores presentados en la transmisión. En este sistema se necesitan dos espacios de tiempo para transmitir dos símbolos modulados.

De acuerdo con la latencia obtenida para el sistema se encontró que los algoritmos de predicción aumentan el tiempo de procesamiento en un 7% por espacio de tiempo aproximadamente. El algoritmo no lineal con las características específicas de una red neuronal recurrente como la estudiada en este trabajo hace aumentar la latencia del sistema en un 12% aproximadamente. Sorpresivamente el filtro Kalman, que está calificado como lineal muestra una latencia similar a los algoritmos no lineales

aumentando el tiempo de procesamiento del sistema en un 10% aproximadamente. Los resultados obtenidos de la medición de latencia del sistema se muestran en la tabla 5.10.

Tabla 5-10 Latencia del sistema MIMO-OFDM con canal estimado y predicho.

Algoritmo	Latencia del sistema (segundos)
Sin predicción	0.7150
Predicción lineal con modelo AR	0.7650
Filtro Kalman	0.7750
Red neuronal con retrasos en entrada	0.7700
Red neuronal recurrente	0.7950

5.5.3 Complejidad computacional de cada algoritmo de predicción:

La complejidad computacional de cada algoritmo, calculada en esta sección, está relacionada con las operaciones entre matrices realizadas y la cantidad de estas.

La complejidad computacional está dada en notación $O()$ indicada de acuerdo a los parámetros variables de cada algoritmo: los lineales en función del orden del modelo y los no lineales, por consistir en redes neuronales en función de la cantidad de entradas retrasadas y el tamaño de la capa intermedia de procesamiento. La tabla 5.11 muestra las características dadas con respecto a la configuración seleccionada en la sección 5.3 contando la naturaleza de las operaciones y la cantidad de estas requeridas para realizar la predicción en un espacio de tiempo.

Tabla 5-11 Complejidad computacional para algoritmos de predicción de canal.

Algoritmo	Naturaleza de las operaciones	Complejidad computacional
Predicción lineal con Modelo AR(n), orden AR = 20	Operación de matrices (A, B), multiplicación ($A_{m \times n} * B_{n \times p} \rightarrow O(mnp)$), suma ($A_{m \times n} * B_{m \times n} \rightarrow O(mn)$)	22 operaciones de multiplicación y 22 de suma.
Filtro kalman, orden AR = 10	Operación de matrices (A, B), multiplicación ($A_{m \times n} * B_{n \times p} \rightarrow O(mnp)$), suma ($A_{m \times n} * B_{m \times n} \rightarrow O(mn)$)	12 operaciones de multiplicación y 5 operaciones de suma.
Red neuronal con retrasos en entrada, Nentradas = 20, Ncapas = 40, Nsalidas = 1	Operación de matrices (A, B), multiplicación ($A_{m \times n} * B_{n \times p} \rightarrow O(mnp)$), suma ($A_{m \times n} * B_{m \times n} \rightarrow O(mn)$) Carga de archivos de tamaño $n \rightarrow O(n \log n)$	Una operación de multiplicación y una de una de suma en la capa intermedia. Una operación de multiplicación y una de suma en la capa de salida. Carga de 4 archivos.
Red neuronal recurrente, Nentradas = 10, Ncapas = 20, Nsalidas = 1	Operación de matrices (A, B), multiplicación ($A_{m \times n} * B_{n \times p} \rightarrow O(mnp)$), suma ($A_{m \times n} * B_{m \times n} \rightarrow O(mn)$)	Dos operaciones de multiplicación y una de suma en la capa intermedia. Una operación de multiplicación y una de suma en la capa de

	Carga de archivos de tamaño $n \rightarrow O(n \log n)$	salida. Carga de 5 archivos
--	---	-----------------------------

En la tabla anterior $WL1$ y $BL1$ son la matriz de pesos de la capa intermedia para las entradas retrasadas y los valores de *bias* para cada neurona de la capa intermedia respectivamente, WO y BO son la matriz de pesos y los valores de *bias* de la capa de salida, y $WO1$ es la matriz de pesos de la capa intermedia para la señal de salida realimentada a esta. A y C son matrices del modelo usado en los algoritmos lineales.

5.5.4 Utilización de memoria de cada algoritmo de predicción:

Para analizar el desempeño del sistema también se tomó en cuenta los requerimientos de almacenamiento de información que en este caso corresponde a matrices de pesos o modelos de los cuales parte cada algoritmo para realizar el procesamiento.

La tabla 5.12 muestra la cantidad de espacios de memoria requeridos para realizar la predicción con cada uno de los algoritmos analizados. El algoritmo lineal con modelo AR requiere únicamente de 20 espacios de memoria para almacenar 20 ganancias de canal previas. A este le sigue el filtro Kalman que requiere 100 espacios de memoria correspondientes al almacenamiento de la matriz A que contiene el modelo del canal de orden 10, y 10 espacios para almacenamiento para la matriz C que es la de medición de la señal. La red neuronal recurrente requiere 441 espacios de memoria, muchos menos que la red con retrasos, gracias a la realimentación de salida. La cantidad de memoria utilizada por la red con retrasos en el tiempo fue la mayor, con un tamaño de 1641 espacios de memoria para almacenar los pesos y el *bias* de cada neuronal de la red. Los valores almacenados y el procesamiento de los algoritmos están dados en notación de punto flotante.

Tabla 5-12. Utilización de memoria para algoritmos de predicción de canal

Algoritmo	Utilización de memoria
Predicción lineal con Modelo AR(n), orden AR = 20	$A_{orden\ AR*1}$
Filtro kalman, orden AR = 10	$A_{orden\ AR} , C_{orden\ AR*1}$
Red neuronal con retrasos en entrada, Nentradas = 20, Ncapas = 40, Nsalidas = 1	$WL1_{Nentradas*Ncapas} ,$ $WO_{Ncapas*Nsalidas} ,$ $BL1_{Nentradas*Ncapas} ,$ $BO_{Nsalidas*1}$
Red neuronal recurrente, Nentradas = 10, Ncapas = 20, Nsalidas = 1	$WL1_{Nentradas*Ncapas} ,$ $WO1_{Nsalidas*Ncapas} ,$ $WO_{Ncapas*Nsalidas} ,$ $BL1_{Nentradas*Ncapas} ,$ $BO_{Nsalidas*1}$

5.6 Comparación final entre esquemas de predicción de canal

Finalmente, solo falta identificar el mejor esquema de predicción de canal, tomando en cuenta las mediciones de desempeño tomadas al sistema en las configuraciones dadas por los algoritmos de predicción implementados. Para esto, se formula la siguiente pregunta: ¿Cuál de los algoritmos de predicción es mejor para su implementación en sistemas reales?

Como método de decisión se diseñó un sistema de puntajes en el cual, para cada métrica de desempeño del sistema se le dio a cada algoritmo un puntaje de 1 a 4 en el que 1 significa el mejor comportamiento y 4 el peor comportamiento; 2 y 3 son valores intermedios dados a los algoritmos que presentaron, de acuerdo a cada métrica, un comportamiento entre medio y bajo.

Tabla 5-13 comparación algoritmos de predicción en sistema MIMO-OFDM.

	Sistema MIMO_OFDM con Algoritmo de predicción lineal con modelo AR	Sistema MIMO_OFDM con Algoritmo de predicción lineal Kalman	Sistema MIMO_OFDM con Algoritmo de predicción no lineal basado en Red Neuronal con retraso en entrada	Sistema MIMO_OFDM con Algoritmo de predicción no lineal basado en Red Neuronal Recurrente
Error en predicción (MSE)	4	3	2	1
Latencia	1	2	3	4
Utilización de memoria	1	2	4	3
Complejidad computacional	1	2	4	3
Tasa de error de bits (BER)	4	3	1	2
Desviación SNR	3	2	4	1
Tasa de transmisión	4	2	3	1
TOTAL PUNTAJE	18	16	21	15

La tabla 5.13 muestra el resultado dado al aplicar el sistema de puntaje con respecto a cada una de las medidas tomadas al sistema MIMO-OFDM de la figura 5.1. De allí se puede observar que en términos generales el mejor comportamiento lo presentó el sistema MIMO-OFDM con predicción de canal no lineal basado en Red Neuronal Recurrente, seguido muy de cerca por el sistema con predicción lineal con filtro kalman. El peor comportamiento lo mostró el sistema con predicción no lineal basado en Red Neuronal con Entrada Retrasada.

6. Conclusiones y Trabajo Futuro

6.1 Conclusiones

En este trabajo se realizó un proceso de diseño y descripción de un sistema MIMO-OFDM con predicción de canal, sobre el cual se evaluaron algoritmos lineales (Algoritmo basado en modelo AR y filtro kalman) y no lineales (Red neuronal con retrasos en entrada y red neuronal recurrente) de predicción usando métricas de desempeño para el sistema como latencia, tasa de errores, tasa de transmisión, complejidad computacional y utilización de memoria; a partir del análisis de desempeño del sistema se concluye que:

El resultado de la comparación entre algoritmos de predicción de canal mostro que el sistema MIMO-OFDM con predicción no lineal basado en red neuronal recurrente presento una mejora en el comportamiento del sistema en términos de relación señal a ruido y tasa de error de bits y además el mejor comportamiento entre los algoritmos evaluados con respecto al mismo sistema sin predicción de canal con una disminución del 20% en la tasa de errores.

Existe una relación entre la precisión de la predicción obtenida y la complejidad operacional Los algoritmos lineales presentaron una menor precisión en la predicción, lo que resulto en una menor tasa de transmisión, pero también mostraron la menor complejidad computacional. Al contrario, los algoritmos no lineales mostraron una mejor habilidad para predecir las condiciones del canal, pero esto requirió una mayor cantidad de recursos computacionales y de memoria; además, la latencia del sistema se incrementó en valores entre 7% y 12%, un valor significativamente alto, y no en todos los casos resulto en una mejor tasa de transmisión de datos o menor error en dicha transmisión.

Se encontró que el procesamiento de todos los algoritmos consistía en operaciones entre matrices; el problema de cada algoritmo consiste en encontrar los parámetros de diseño adecuados. Allí se puede observar que para el caso de los algoritmos lineales el orden del modelo AR era de gran importancia para el funcionamiento de estos, y para el filtro Kalman el reto se encuentra en adoptar una varianza de error adecuada que debe ser recalculada cada n espacios de tiempo para lograr un funcionamiento adecuado dependiendo del modelo de canal dado. Los algoritmos no lineales analizados en cambio, tienen su reto en el entrenamiento de estos, la cantidad de datos utilizada para este objetivo y la cantidad de entradas realimentadas que representan una aún mayor complejidad computacional.

Los datos de entrenamiento de los algoritmos no lineales juegan un factor importante pues a partir de allí se puede establecer un buen o mal funcionamiento del algoritmo; se

hace necesario pues tener una gran cantidad de datos de canal, posiblemente generados a partir de un modelo establecido; esto representa un costo computacional alto y un tiempo de entrenamiento elevado, y es poco probable que sea posible realizar un entrenamiento online.

El sistema pudo ser evaluado desde diferentes perspectivas de funcionamiento estableciendo métricas apropiadas como son la latencia y la tasa de transmisión. Esto se logró gracias a un diseño previo realizado e implementado sobre una plataforma que permite integrar diversos aspectos del diseño funcional de un sistema digital.

La estimación de canal juega un papel importante dentro de la adaptación al enlace, pues a partir de allí se tomarán decisiones que afectan directamente la tasa de transmisión del sistema y la sensibilidad a errores presentados.

En términos generales del análisis de desempeño del sistema, y de acuerdo al sistema de puntajes establecido para responder a la pregunta de cuál de los esquemas presenta el mejor comportamiento para su implementación, se concluye que la predicción basada en red neuronal recurrente tiene las características de precisión de predicción, latencia de sistema y carga computacional para mejorar las características de un sistema MIMO-OFDM con adaptación a canal inalámbrico.

6.2 Trabajo Futuro

Los trabajos futuros relacionados con la investigación realizada son:

- (1) el análisis del sistema no solo a nivel de capa física, sino también en un escenario multi-usuario para sistema MIMO-OFDM, donde la transmisión de datos tenga en cuenta el tráfico dado por una cantidad mayor a uno de usuarios, donde el acceso a la red es un factor importante,
- (2) la implementación de diferentes técnicas de estimación de canal, que eliminen parcial o totalmente el ruido de la estimación, permitiendo una mejor adaptación del sistema,
- (3) la implementación de técnicas de codificación de señal para adaptación a enlace, y finalmente
- (4) el análisis del sistema bajo otros modelos de canal de comunicaciones.

Bibliografía

- [1] Hongwei Yang, "A Road to Future Broadband Wireless Access: MIMO-OFDM-Based Air Interface", IEEE Communications Magazine, pp. 53-60, January 2005.
- [2] Tellado, J. , "A Fourth-Generation MIMO-OFDM Broadband Wireless System: Design, Performance, and Field Trial Results", Communications Magazine, IEEE , Vol.40 , Issue: 9, 2002.
- [3] Kai-Kit Wong, Lai, S.-K. ; Cheng, R.S.-K. ; Letaief, K.B., "Adaptive Spatial-Subcarrier Trellis Coded MQAM and Power Optimization for OFDM Transmission", Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st, Vol. 3, 2000.
- [4] Sai Kit Lai, Cheng, R.S. ; Letaief, K.B. ; Murch, R.D., "Adaptive Trellis Coded MQAM and Power Optimization for OFDM Transmission", Vehicular Technology Conference, 1999 IEEE 49th, Vol. 1, pp. 290 - 294, 1999.
- [5] Yuan Li, Ying-Chang Liang, Sumei Sun, Rui Zhang, "Adaptive Trellis and Bit-Interleaved Coded Modulation for Ordered MIMO-OFDM Channels", Vol. 1, pp. 226 – 230, 2005.
- [6] S.M. Alamouti (October 1998). "A simple transmit diversity technique for wireless communications". IEEE Journal on Selected Areas in Communications 16 (8): 1451–1458.
- [7] Ali H. Al-Qahtani, Rabah W. Aldhaheeri, "Performance Study of Adaptive Modulation and Coding Schemes in WiMAX OFDM-based System", Proceedings of the 2009 IEEE 9th Malaysia International Conference on Communications, 2009.
- [8] Han Zhang, Jiming Chen ; Youxi Tang ; Shaoqian Li, "Analysis of Pilot-Symbol Aided Channel Estimation for MIMO-OFDM Systems", Communications, Circuits and Systems, 2004. ICCAS 2004. 2004 International Conference on, vol. 1, pp. 299 – 303, 2004.
- [9] Auer, G., "Analysis of Pilot-Symbol Aided Channel Estimation for OFDM Systems with Multiple Transmit Antennas", Communications, 2004 IEEE International Conference on , Vol. 6, pp. 3221 – 3225, 2004.
- [10] Hosseinneshad, M., Salahi, A., "Performance Comparison of Pilot-symbol Aided Channel Estimation Methods for the IEEE802.16d WMAN Standard", Communications

and Networking in China, 2008. ChinaCom 2008. Third International Conference on, pp. 1379 – 1383, 2008.

[11] Viterbi AJ (April 1967). "*Error bounds for convolutional codes and an asymptotically optimum decoding algorithm*". IEEE Transactions on Information Theory 13, pp. 260–269.

[12] Grötke, T., Liao, S., Martin, G., Swan, S., "*System Design with SystemC*", Kuwel Academic Publishers, 2002.

[13] Lajos Hanzo, Yosef (Jos) Akhtman and Li Wang, "*MIMO-OFDM for LTE, Wi-Fi and Wimax: Coherent versus Non-coherent and Cooperative Turbo Transceivers*", John Wiley & Sons Ltd, 2010.

[14] Kareem E. Baddour, d Norman C. Beaulieu, "*Autoregressive Modeling for Fading Channel Simulation*", IEEE Transactions On Wireless Communications, VOL. 4, NO. 4, JULY 2005.

[15] R.E. Kalman, "*A New Approach to Linear Filtering and Prediction Problems*", Transactions of the ASME–Journal of Basic Engineering, 1960.

[16] Jwo-Yuh Wu and Wen-Ming Lee, "*Optimal Linear Channel Prediction for LTE-A Uplink Under Channel Estimation Errors*", IEEE, 2013.

[17] Steve Gifford, Chad Bergstrom and Scott Chuprun, "*Adaptive And Linear Prediction Channel Tracking Algorithms For Mobileofdm-Mimo Applications*", MILCOM 2005. IEEE, 2005.

[18] Trojanowski, A. , Wojciechowski, J., "*Algorithms of Linear Prediction for Non-Stationary Rayleigh Channel*", Personal, Indoor and Mobile Radio Communications, IEEE 17th International Symposium, 2006.

[19] Yau Hee Kho, Desmond P. Taylor, "*MIMO Channel Estimation and Tracking Based on Polynomial Prediction With Application to Equalization*", IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 57, NO. 3, MAY 2008.

[20] Shogo Soejimay, Yuta Ida, Chang-Jun Ahn, Tatsuya Omori, and Kenya Hashimoto, "*Non-linear Channel Compensation Method for OFDM under Fast Fading Channel*", IEEE TENCON, 2011.

[21] W.C.Jakes, "*Microwave mobile communications*", Wiley-IEEE Press, mayo 1994.

- [22] Yahong R. Zheng and Chengshan Xiao, "Improved Models for the Generation of Multiple Uncorrelated Rayleigh Fading Waveforms", IEEE COMMUNICATIONS LETTERS, VOL. 6, NO. 6, JUNE 2002.
- [23] R. H. Clarke, "A statistical theory of mobile-radio reception" Bell Syst. Tech. J., pp. 957–1000, July–Aug. 1968.
- [24] Analyzing Application Performance by Using Profiling Tools", <http://msdn.microsoft.com/en-us/library/z9z62c29.aspx>
- [25] Redlich, Oded and Ezri, Doron and Wulich, Dov , "SNR Estimation in Maximum Likelihood Decoded Spatial Multiplexing", CoRR, 2009.
- [26] Raymond Chan, "Channel Prediction For Adaptive Modulation In Wireless Communications", pp. 56-59, Virginia Polytechnic, 2003, July 16.
- [27] Changkee Min, Namseok Chang, Jongsub Cha, Joonhyuk Kang, "MIMO-OFDM Downlink Channel Prediction for IEEE802.16e Systems Using Kalman Filter", Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE, pp. 942 – 946.
- [28] Chris Potter, Ganesh K. Venayagamoorthy, Kurt Kosbar, "RNN based MIMO channel prediction", Signal Processing 90, pp.440–450.
- [29] G. Routray and P. Kanungo, "Rayleigh Fading MIMO Channel Prediction Using RNN with Genetic Algorithm", CIIT 2011, CCIS 250, pp. 21–29, 2011.
- [30] Perrins, Erik, "FEC Systems for Aeronautical Telemetry, Aerospace and Electronic Systems", IEEE Transactions on, Volume: 49 , Issue: 4 , Publication Year: 2013 , Page(s): 2340 - 2352
- [31] Zhang Xi-jun and He Ji-ai and Jia Ke-jun and Lin Ying, "Research and Simulation of Propagation Characteristic in Wireless Channel", Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on, pp. 1-3, 2009.
- [32] Leung, M. and Leung, J. and Baron, G.S. and Sarris, C.D., "A fast time-domain wireless channel simulation tool for radio-wave propagation courses", Antennas and Propagation Society International Symposium 2006, IEEE, pp. 1277-1280, 2006.
- [33] Sklar, B., "Rayleigh fading channels in mobile digital communication systems .I. Characterization", Communications Magazine, IEEE, Vol. 35, pp. 90-100, 1997.
- [34] Vijay k. Bhargava, "Forward Error Correction Schemes for Digital Communications", IEEE Communications Magazine, 1983.

- [35] R. W. Hamming, "*Error Detecting and Error Correcting codes*", The Bell System Technical Journal, Vol. 29, No. 2, 1950.
- [36] Karen Su, "*Space-Time Coding: From Fundamentals To The Future*", Cambridge University Engineering Department, University of Cambridge, 2003.
- [37] Gerard. J. Foschini, "*Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multi-Element Antennas*". Bell Laboratories Technical Journal: 41–59, 1996.
- [38] Amin Mobasher, Mahmoud Taherzadeh, Renata Sotirov, and Amir K. Khandani, "*A Near Maximum Likelihood Decoding Algorithm for MIMO Systems Based on Semi-Definite Programming*", IEEE Trans. On Info. Theory, 2007.
- [39] Yi Jiang Mahesh K. Varanasi Jian Li, "*Performance Analysis of ZF and MMSE Equalizers for MIMO Systems: An In-Depth Study of the High SNR Regime*", Globecom, 2005.
- [40] Sebastian Caban, Christian Mehlhöhrer, Arpad L. Scholtz, and Markus Rupp, "*Indoor MIMO Transmissions with Alamouti Space -Time Block Codes*", WITSP, 2005.
- [41] John R. Barry, Edward A. Lee, David G. Messerschmitt, "*Digital Communication*", Third Edition, pp. 461-536, September 30, 2003.
- [42] Dr. David J Greaves, "*System on Chip Design and Modelling*", University of Cambridge Computer Laboratory, Lecture Notes, part 2, 2011.
- [43] Stephen J. Roberts and Will D. Penny, "*Variational Bayes for Generalized Autoregressive Models*", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 50, NO. 9, SEPTEMBER 2002.
- [44] Singer, A.C. and Wornell, Gregory W. and Oppenheim, A.V., "*Codebook prediction: a nonlinear signal modeling paradigm*", Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on, 325-328 vol.5, 1996.
- [45] J. Durbin, "*The Fitting of Time-Series Models*", Revue de l'Institut International de Statistique / Review of the International Statistical Institute, Vol. 28, No. 3., pp. 233-244, 1960.
- [46] Levinson N, "*The Wiener R M S (Root Mean Square) error criterion in filter design and prediction*", Appendix B. N. Wiener's book reprint, 1949.
- [47] Raúl Rojas, "*Neural Networks: A Systematic Introduction*", Springer – Verlag, Berlin, 1996.

- [48] McInerney, M. and Dhawan, Atam P., "Use of genetic algorithms with backpropagation in training of feedforward neural networks", Neural Networks, 1993., IEEE International Conference on, pp. 203-208 vol.1, 1993.
- [49] Hinton, G. E. and Salakhutdinov, R. R., "Reducing the Dimensionality of Data with Neural Networks", Science, Num. 5786, Vol. 313, pp. 504-507, 2006.
- [50] Bahram G. Kermani, Susan S. Schiffman, H. Troy Nagle, "Sensors and Actuators B: Chemical", Volume 110, Issue 1, Pp.13-22, 2005.
- [51] Sampath, H. and Talwar, S. and Tellado, J. and Erceg, V. and Paulraj, A., "A fourth-generation MIMO-OFDM broadband wireless system: design, performance, and field trial results", Communications Magazine, IEEE, Vol. 40, Num. 9, pp. 143-149, 2002.
- [52] Zhongshan Wu, "Mimo-Ofdm Communication Systems: Channel Estimation And Wireless location", Dissertation The Department of Electrical and Computer Engineering, Louisiana State University, 2001.
- [53] Poongodi, C. and Ramya, P. and Shanmugam, A., "BER analysis of MIMO OFDM system using M-QAM over Rayleigh fading channel", Communication and Computational Intelligence (INCOCCI), 2010 International Conference on, pp. 284-288, 2010.
- [54] Ogale, A. and Chaoudhary, S. and Patil, A.J., "Performance evaluation of MIMO-OFDM system using Matlab #x00AE; Simulink with real time image input", Wireless and Optical Communications Networks (WOCN), 2013 Tenth International Conference on, pp. 1-5, 2013.
- [55] Jun Zhang and Qing Zhao, "Simulation and analysis of MIMO-OFDM system based on simulink", Communications, Circuits and Systems (ICCCAS), 2010 International Conference on, pp. 19-22, 2010.
- [56] Guillermo Acosta, "OFDM Simulation Using Matlab", Smart Antenna Research Laboratory, 2000.
- [57] Prabhu, G.S. and Mohana Shankar, P., "Simulation of flat fading using MATLAB for classroom instruction", Education, IEEE Transactions on, Vol. 45, Num. 1, pp. 19-25, 2002.
- [58] Al-Qahtani, A.H. and Aldhaheeri, R.W., "Performance study of adaptive modulation and coding schemes in WiMAX OFDM-based system", Communications (MICC), 2009 IEEE 9th Malaysia International Conference on, pp. 836-839, 2009.
- [59] Bellili, F. and Stephenne, A. and Affes, S., "SNR estimation of QAM-modulated transmissions over time-varying SIMO channels", Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on, pp. 199-203, 2008.

- [60] Wiesel, A. and Goldberg, J. and Messer-Yaron, H., "*SNR estimation in time-varying fading channels*", Communications, IEEE Transactions on, Vol. 54, Num. 5, pp. 841-848, 2006.
- [61] Thuong Nguyen Canh and Van Duc Nguyen and Phuong Dang and Luong Pham Van and Thu Nga Nguyen and Patzold, M., "*A performance study of LTE MIMO-OFDM systems using the extended one-ring MIMO channel model*", Advanced Technologies for Communications (ATC), 2012 International Conference on, pp. 263-268, 2012.
- [62] Delestre, F. and Yichuang Sun, "*A channel estimation method for MIMO-OFDM Mobile WiMax systems*", Wireless Communication Systems (ISWCS), 2010 7th International Symposium on, pp. 956-960, 2010.
- [63] Lajos L. Hanzo, Yosef Akhtman, Li Wang, Ming Jiang, "*MIMO-OFDM for LTE, WIFI and WIMAX: Coherent versus Non-Coherent and Cooperative Turbo-Transceivers*", John Wiley & Sons, 1 November, 2010.
- [64] Chengshan Xiao and Zheng, Yahong R. and Beaulieu, N.C., "*Statistical simulation models for Rayleigh and Rician fading*", Communications, 2003. ICC '03. IEEE International Conference on, Vol. 5, pp. 3524-3529, 2003.
- [65] Chang, R. W., "*Synthesis of band-limited orthogonal signals for multi-channel data transmission*", Bell System Technical Journal, vol. 45, issue 10, pp. 1775-1796, 1966.
- [66] Accellera Systems Initiative, Supported IEEE Activities, IEEE 1666 SystemC, www.accellera.org.