Análisis del Desempeño de la Codificación Reed-Muller en un Canal con Ruido Blanco Aditivo Gaussiano.

Wilson Felipe Bolaños Arcos

Diego Fernando Díaz Díaz

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Departamento de Telecomunicaciones

Línea de Investigación Gestión Integrada de Redes, Servicios y Arquitectura de Telecomunicaciones

Grupo de Nuevas Tecnologías en Telecomunicaciones

Popayán 2015.

Análisis del Desempeño de la Codificación Reed-Muller en un Canal con Ruido Blanco Aditivo Gaussiano.

Wilson Felipe Bolaños Arcos

Diego Fernando Díaz Díaz

Trabajo de Grado presentado como requisito para obtener el título de Ingeniero en Electrónica y Telecomunicaciones.

Director: M.Sc. Jesús Mauricio Ramírez Viáfara

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Departamento de Telecomunicaciones

Línea de Investigación Gestión Integrada de Redes, Servicios y Arquitectura de Telecomunicaciones

Grupo de Nuevas Tecnologías en Telecomunicaciones

Popayán 2015.

A mi familia por su apoyo incondicional,
en especial a mi madre que ya está en el cielo y a mi padre que aún está conmigo.

A Sofía por darme las fuerzas, cuando más las necesité, para seguir adelante.

Wilson Felipe Bolaños Arcos.

Contenido

Pág.

Lista de Tablas

Pág.

Lista de Figuras

Pág.

Lista de Acrónimos

ARQ	*Automatic Retransmission Query* (Solicitud de Retransmisión Automática).
AWGN	*Additive White Gaussian Noise* (Ruido Blanco Aditivo Gaussiano).
BCH	Bose-Ray-Chaudhuri.
BER	*Bit Error Rate* (Tasa de Error de Bit).
EDAC	*Error Detection And Correction* (Detección y Corrección de Errores).
FEC	*Forward Error Correction* (Corrección de Errores Hacia Adelante).
GF	*Galois Fields* (Campos de Galois).
MATLAB	*Matrix Laboratory* (Laboratorio de Matrices).
M-PAM	*M-ary Pulse Amplitude Modulation* (Modulación por Amplitud de Pulsos M-ario).
RM	Reed-Muller.
SDC	*Silent Data Corruption* (Corrupción de Datos Silenciosa).
SEC	*Single Error Correction* (Corrección de Único Error).

Resumen

En este documento se presenta el análisis del desempeño, en cuanto a probabilidad de error, de la codificación Reed-Muller (RM) en la transmisión sobre un canal con ruido blanco aditivo Gaussiano (AWGN, *Additive White Gaussian Noise*). Se hace un estudio de los campos de cuerpo finito o campos de Galois (GF, *Galois Fields*) y las funciones booleanas, se describen las características generales de los códigos RM para luego mostrar el proceso de codificación RM. También se estudian las técnicas de decodificación lineales, *Hard Decision,* a través del criterio de distancia mínima Hamming, y *Soft Decision* con el criterio de distancia mínima Euclidiana. Se realiza la simulación del sistema de comunicación con codificación RM y se muestran los resultados de las diferentes pruebas realizadas en MATLAB. Por último se realiza el análisis del desempeño de algunos códigos RM utilizando las técnicas de decodificación descritas.

1. Introducción

Los sistemas de comunicaciones digitales ofrecen considerables ventajas en comparación a los sistemas de comunicaciones analógicos, debido a la combinación de diferentes técnicas en cada uno de sus procesos. Dentro de estas ventajas se encuentran la gran capacidad de integración, flexibilidad, escalabilidad, reducción de costos, como también la robustez de la señal transmitida frente a efectos degradantes como el ruido y la interferencia. Otra de las ventajas de gran impacto es la capacidad de detectar y corregir errores, idea que surgió originalmente para dar solución a problemas prácticos de la ingeniería eléctrica y que de igual manera permite mayor confiabilidad y mejor desempeño en los sistemas de comunicación.

La codificación de canal permite detectar y corregir los errores que ocurren durante el proceso de comunicación añadiendo redundancia a la información que se desea transmitir. De esta manera, la codificación puede aprovecharse para alcanzar el nivel de confiabilidad deseado en la información que será entregada al usuario. Alternativamente a la codificación de canal muchos sistemas de comunicaciones tratan el problema mediante el incremento del nivel de potencia de la señal transmitida, esto con el objetivo de lograr que la señal supere las alteraciones que pueda sufrir en la transmisión, así como evitar las implicaciones negativas de la codificación, tales como el incremento del consumo de ancho de banda y complejidad del sistema o el aumento en el tiempo de transmisión [1].

La capacidad de detectar y/o corregir errores provee confiabilidad en la transmisión de información sin utilizar grandes niveles de potencia, lo que genera un ahorro considerable comparado con sistemas en los que no se utiliza la codificación. Cabe aclarar que los costos de potencia en comunicaciones son demasiado altos.

Las técnicas utilizadas en la codificación de canal básicamente transforman el mensaje original en una secuencia que incluye redundancia, con el objetivo de dotarla de un tipo de protección y hacerla menos susceptible a alteraciones y errores que pueda introducir el canal. Esta técnica se conoce como Corrección de Errores Hacia Adelante (FEC, *Forward Error Correction*), y juega un papel importante en la comunicación digital moderna. La otra técnica conocida para el

control de errores es la la Solicitud de Retransmisión Automática (ARQ[1], *Automatic Retransmission Query*).

La codificación de canal se divide en dos grandes grupos: la codificación bloque y la codificación convolucional. La diferencia entre ellas radica en el hecho de que los primeros llevan a cabo el proceso de codificación sobre segmentos de longitud fija, mientras que los segundos lo hacen en forma continua sobre todo el mensaje [1].

La codificación para control de errores hace parte de una rama de las matemáticas aplicadas, llamada teoría de la información (Shannon, 1948). Una aplicación específica corresponde a los códigos RM.

Los códigos RM forman una familia infinita de códigos muy usados en el envío de información a través de largas distancias o a través de canales con alta probabilidad de error. Fue la primera familia de códigos en proporcionar un mecanismo para la obtención de una distancia mínima deseada. Existe una variedad de construcciones para los códigos RM, que ha hecho que sean útiles en muchos desarrollos teóricos así como en componentes de otros sistemas [2].

El presente trabajo introduce los conceptos teóricos básicos del esquema de codificación RM, indicando los bloques funcionales involucrados en el diseño del transmisor y el receptor. Posteriormente, se evalúa el desempeño de algunos códigos RM sobre un canal del tipo AWGN, en cuanto a probabilidad de error; esto a través de una simulación, cuyos resultados se comparan a través de curvas de Tasas de Error de Bit (BER, *Bit Error Rate*) vs Relación Energía de Bit a Densidad de Ruido (Eb/No) con un sistema sin codificación.

Para el desarrollo del proyecto se sigue la guía de referencia del Modelo Lineal Secuencial [3]. Este modelo sugiere un enfoque sistemático y secuencial que se basa en 5 fases:

1. Fase de Recolección de Información y Preparación.
2. Fase de Análisis y Síntesis de la Información.
3. Fase de Diseño e Implementación.
4. Fase de Pruebas y Análisis de Resultados.
5. Fase de Entrega.

[1] ARQ: El receptor es capaz de reconocer o detectar la presencia de un error, pero no puede corregirlo y solicita al transmisor la repetición del mensaje.

Las fases 1 y 2 corresponden a la base conceptual o marco teórico del proyecto, presentado en el capítulo 2, en el cual se socializan los conceptos que caracterizan la codificación de canal RM y los criterios de decodificación de distancia mínima Hamming y distancia mínima Euclidiana; la fase 3 permite diseñar un algoritmo propio de simulación para un sistema de comunicación digital en banda base con codificación RM sobre un canal AWGN, éste se describe en el capítulo 3; la fase 4 compete a la evaluación y análisis de los resultados del desempeño de los códigos RM, y por lo tanto del sistema de comunicación en general, obtenidos por medio de simulación, aquí también se realiza un análisis comparativo del desempeño del decodificador RM con criterio de decodificación de distancia mínima Hamming frente al decodificador RM con criterio de distancia mínima Euclidiana, esto se muestra en el capítulo 4; La fase 5 consiste en la elaboración y entrega del documento final y artículo concernientes al trabajo de grado; finalmente, en el capítulo 5 se presentan las conclusiones más importantes y algunas opciones para trabajos futuros.

2. Generalidades

En este capítulo se introducen algunas generalidades sobre: los sistemas de comunicación digital, la codificación de canal Reed-Muller (en especial su descripción matemática) y la descripción de los procesos de decodificación basados en distancia mínima Euclidiana y distancia mínima Hamming.

2.1 Sistema de Comunicación Digital

Comunicación es el proceso por el cual se transmite información[2] desde un punto llamado fuente o emisor a otro punto llamado destino o receptor. Dicha transmisión se hace a través de un canal y debe realizarse consumiendo la menor cantidad posible de recursos como ancho de banda y potencia de transmisión.

Un sistema de comunicación es el conjunto de mecanismos que aseguran la conexión entre el emisor y el receptor [4]. Se definen tres componentes principales:

Transmisor: Se encarga de convertir el mensaje proveniente de la fuente, en una señal[3] adecuada para transmitirse a través del canal de comunicación.

Canal de comunicación: Transporta la señal desde el transmisor hasta el receptor.

Receptor: Se encarga de tomar la señal del canal y de reconstruir el mensaje original a partir de ésta.

En la figura 2.1 se muestra el modelo básico de un sistema de comunicación.

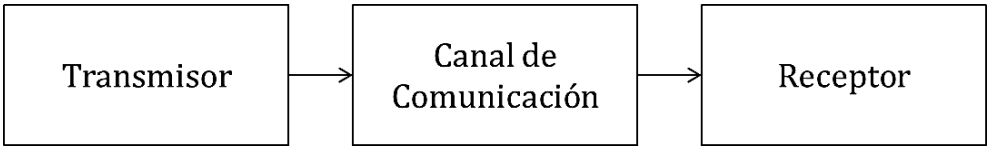


Figura 2.1 Modelo lineal de un sistema de comunicación.

[2] Información: Señales inteligentes comunicadas o recibidas[25].
[3] Señal: Cantidad física que varía con respecto a una o más variables independientes. Estas contienen información de la naturaleza o comportamiento de algún fenómeno [25].

En los sistemas de comunicación digital, los mensajes pertenecen a un conjunto finito y discreto de valores, siendo menos sensibles al ruido que los sistemas de comunicación analógica [1].

2.1.1 Elementos de un Sistema de Comunicación Digital

Un sistema de comunicación digital está compuesto por los bloques funcionales mostrados en la figura 2.2 [5]. A continuación se describe el rol que desempeña cada uno de ellos:

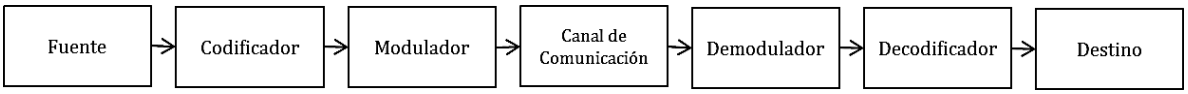


Figura 2.2 Diagrama en bloques de un sistema genérico de comunicación digital.

Fuente: Genera la información a transmitir en su forma más simple, sin ninguna operación externa para fines de transmisión. Las fuentes son vistas como flujos de números aleatorios que se rigen por alguna distribución de probabilidad. Su significado es conocido en el destino del sistema.

Codificador: Se pueden aplicar dos tipos de codificación al mensaje original: la codificación de fuente, que comprime los datos mediante la eliminación de redundancia en el mensaje original; y la codificación de canal, que agrega información redundante de manera controlada con el propósito de detectar y/o corregir errores en el receptor.

Modulador: Convierte la salida del codificador de canal en una señal adecuada para la transmisión sobre un canal que introducirá, entre otros fenómenos, distorsión y ruido. Muchos canales requieren que las señales sean enviadas como formas de onda continuas en el tiempo, tales como campos electromagnéticos, voltajes, corrientes, pulsos de luz, etc., según la naturaleza física del canal y las restricciones impuestas por el entorno donde se lleva a cabo la comunicación. De esa manera, el modulador proporciona la representación apropiada conforme al canal [6].

Canal de comunicación: Es el medio físico sobre el que se transmite la información desde la fuente hasta el destino. La característica esencial es que la señal transmitida sufre efectos degradantes entre los que se destacan: la atenuación, el ruido, la distorsión y la interferencia [1].

Demodulador: Recibe la señal proveniente del canal y la convierte en una secuencia de símbolos. Normalmente, esto implica muchas funciones, tales como traslación en frecuencia, sincronización de portadora, estimación de temporización de símbolos, sincronización de trama, filtraje adaptado y muestreo seguido por una etapa de detección en la que se toman las decisiones acerca de los símbolos transmitidos [6].

Decodificador: El objetivo de este bloque es realizar las operaciones inversas al codificador. La decodificación de canal aprovecha la redundancia introducida por su contraparte en el transmisor para corregir los errores que puedan haberse introducido en el tránsito por el canal, mientras que la decodificación de fuente descifra y reestablece el mensaje original a través de un algoritmo de descompresión.

Destino: Es el punto final de la comunicación. Tiene la capacidad de entender y procesar la información digital.

2.2 Códigos Reed-Muller

Los códigos RM son una familia infinita de códigos bloque lineales binarios utilizados para la detección y corrección de errores en las comunicaciones de larga distancia. Fueron tratados inicialmente como códigos binarios debido a que operan sobre bits individuales, pero en la actualidad existen generalizaciones relacionadas con códigos multi-nivel. Los códigos RM fueron diseñados por los ingenieros Irving S. Reed y David E. Muller en 1954 como una contribución a la teoría de la información, encargada del estudio de la transmisión y posterior recepción de datos a través de canales ruidosos [2].

Los códigos RM poseen muchas propiedades interesantes que vale la pena examinar, entre ellas, la capacidad de construir códigos más extensos en forma recursiva a partir de códigos elementales. Desafortunadamente, a medida que la longitud de las palabras código aumenta, los códigos RM se hacen cada vez menos robustos; pero esto no impide que se utilicen como bloques de construcción para otros códigos. La variedad de construcciones que existen es una de las principales características de los códigos RM y esto los hace útiles en muchos desarrollos teóricos [6].

A pesar de que en la práctica los códigos RM han sido desplazados en gran parte por los códigos Reed-Solomon, tienen un algoritmo rápido de decodificación, basado en máxima verosimilitud que sigue siendo muy atractivo. Las palabras código RM pueden ser decodificadas fácilmente debido a que la distancia mínima del código es suficientemente grande [6].

2.3 Aplicaciones de los Códigos Reed-Muller

Los códigos RM se encuentran entre los códigos más antiguos conocidos y se han encontrado numerosas aplicaciones entre las que se destacan:

- *Localización de nodos maliciosos:* Los códigos RM son utilizados para implementar seguridad en sistemas de comunicaciones inalámbricos. La localización de nodos maliciosos en una red inalámbrica puede ser posible utilizando un código de detección de errores como RM. Los códigos RM son aplicados a las rutas en la red y la decodificación en el nodo destino revela el nodo malicioso. En el uso de este método intervienen parámetros como la probabilidad de que un nodo sea malicioso, así como del número de nodos en la red [7].
- *Esquemas criptográficos seguros:* Estos esquemas criptográficos que utilizan los códigos RM extienden el tamaño de la llave pública, lo que permite incrementar la seguridad. La codificación RM es usada para construir la clave secreta, la cual es una matriz generadora. Aunque la codificación RM extiende el tamaño de la clave secreta, esto no representa un problema, ya que el algoritmo de decodificación RM es conocido por su eficiencia y velocidad de decodificación [8].
- *Sistemas de transmisión de datos:* Es la primera aplicación conocida de los códigos RM. Se usó para codificar imágenes transmitidas en el espacio. Las primeras fotografías del planeta Marte fueron tomadas por la serie de naves Mariner en los años 60 y principios de los 70, en los cuales se utilizó el código RM de primer orden y longitud 32 para obtener fotografías de buena calidad a blanco y negro. Posteriormente, el código RM permitió el incremento en la calidad de las imágenes, incluso se logró codificar imágenes a color [9].
- *Memorias de almacenamiento:* Comúnmente se usan códigos que pueden corregir un error en una palabra de memoria. Dichos códigos se conocen como códigos de Corrección de Único Error (SEC*, Single Error Correction*),

además de la corrección de errores simple, se detectan errores que no se pueden corregir para evitar la corrupción de datos silenciosa (SDC, *Silent Data Corruption*). Los códigos RM corrigen errores simples y proporcionan la detección de errores múltiples [10].

Los códigos RM están estrechamente ligados con las variables booleanas y pueden describirse a través de polinomios sobre el Campo de Galois[4] o Campo Finito $GF(2)$ [11]. Por tal motivo, es importante introducir algunos conceptos del álgebra computacional que son necesarios para el desarrollo.

2.4 Campos de Galois

Galois enfocó sus estudios hacia la teoría de grupos. Un grupo está formado por un conjunto de elementos abstractos o símbolos que se manipulan por medio de una ley u operación de composición interna claramente definida dentro del grupo. Dicha operación debe satisfacer las propiedades asociativa, conmutativa y clausurativa, así como poseer el elemento identidad y el elemento inverso [12].

Un campo de Galois es un grupo conformado por un número finito de elementos que cumple con las condiciones expuestas en el párrafo anterior a través de dos operaciones básicas binarias, a saber, la adición y la multiplicación. Se denota con $GF(q) = GF(p^m)$, donde q es el tamaño del campo, p un número primo llamado base o módulo del campo finito $(p \geq 2)$ y m la longitud en bits del símbolo $(m \geq 1)$.

El campo de Galois $GF(2)$, llamado también campo binario, es el campo $GF(p^m)$ con $p = 2$ y $m = 1$, el cual tiene como elementos los dígitos binarios 0 y 1. Las operaciones binarias de adición y multiplicación se definen en las tablas 1 y 2. Los elementos identidad para la adición y la multiplicación son 0 y 1 respectivamente.

Tabla 2.1 Adición binaria.

$0 \oplus 0$	0
$0 \oplus 1$	1
$1 \oplus 0$	1
$1 \oplus 1$	0

[4]Estos campos son llamados así en honor al matemático francés *Évariste Galois*.

Tabla 2.2 Multiplicación binaria.

$0 \cdot 0$	0
$0 \cdot 1$	0
$1 \cdot 0$	0
$1 \cdot 1$	1

2.5 Funciones Booleanas

Las funciones booleanas se denominan así en honor al matemático británico George Boole. Gracias a su álgebra es posible manipular las operaciones lógicas de la aritmética computacional moderna.

Una función de Boole es una función cuyo dominio son las palabras conformadas por los valores binarios 0 ó 1 ("falso" o "verdadero", respectivamente), y cuyo codominio son justamente los valores 0 y 1. Una función de Boole de m variables, $f\ (x_1, x_2, \ldots, x_m)$, es un mapeo del espacio vectorial Z_2^m al campo finito Z_2, es decir, de las m-tuplas binarias $(x_1, x_2, \ldots, x_m)$ a los números binarios $(0, 1)$. Formalmente, $f: Z_2^m \rightarrow Z_2$, donde $Z_2 = \{0,1\}$ y m es un entero no negativo [6].

El número de funciones booleanas distintas que se pueden obtener con m variables coincide con el número de secuencias binarias distintas de longitud 2^m que surgen al combinar linealmente las m variables. El número de funciones booleanas es 2^{2^m}. Así, si B_m simboliza al conjunto de todas las funciones booleanas de m variables, se tiene que su cardinalidad es $|B_m| = 2^{2^m}$[6].

El conjunto B_m tiene definida una suma y una multiplicación en el álgebra de Boole como se muestra a continuación:

Dadas $f, g \in B_m$ y $t \in Z_2$, las operaciones son:

Suma:

$$(f \oplus g)(x_1, x_2, \ldots, x_m) = f\ (x_1, x_2, \ldots, x_m) \oplus g\ (x_1, x_2, \ldots, x_m). \tag{2.1}$$

Multiplicación:

$$(f \cdot g)(x_1, x_2, \ldots, x_m) = f\ (x_1, x_2, \ldots, x_m) \cdot g(x_1, x_2, \ldots, x_m). \tag{2.2}$$

$$(t \cdot f)(x_1, x_2, \ldots, x_m) = t \cdot f\ (x_1, x_2, \ldots, x_m). \tag{2.3}$$

La ecuación 2.1 hace referencia a la suma módulo 2 entre cada elemento de la función f y cada elemento de la función g, cada función es booleana y contiene m variables. Las ecuaciones 2.2 y 2.3 corresponden a la multiplicación entre cada elemento de la función f y cada elemento de la función g, siendo f y g funciones booleanas de m variables.

El conjunto B_m es un Z_2- espacio vectorial y también es un anillo conmutativo[5] con uno. La suma y la multiplicación entre funciones booleanas se establecen de acuerdo a operaciones módulo 2. Cabe aclarar que las operaciones de suma y multiplicación deben regirse por las propiedades clausurativa, conmutativa, asociativa, distributiva, elemento inverso y elemento identidad.

El peso Hamming de una función booleana f se define como el número de variables diferentes de cero en dicha función, como se indica en la siguiente ecuación:

$$w(f) = \sum_{i=1}^{m} f(x_i), \tag{2.4}$$

y la distancia Hamming entre dos funciones booleanas f y g como:

$$d(f, g) = w(f \oplus g). \tag{2.5}$$

Existen varias maneras de representar a cada una de estas funciones, entre ellas se destacan: el álgebra booleana, la tabla de verdad, la forma normal algebraica, la forma normal numérica, el espectro de Fourier, etc. Cada una de dichas representaciones posee ventajas y desventajas desde el punto de vista analítico [13]. Las representaciones de interés para el estudio de los códigos RM se definen a continuación.

2.5.1 Representación Vectorial

Las funciones booleanas se suelen representar mediante una tabla de verdad (o vector característico). Una tabla de verdad contiene todos los valores posibles de una función booleana dependiendo del valor de sus variables. El número de combinaciones posibles para una función de m variables es 2^m. Una función

[5] Un anillo conmutativo es un anillo $(R, +, \cdot)$ en el que la operación de multiplicación $\cdot$ es conmutativa; es decir, si $a, b \in R$, entonces se tiene que $a \cdot b = b \cdot a$.

booleana sólo tiene asociada una única tabla de verdad. La tabla de verdad o vector característico $V_f \in Z_2^{2^m}$ de $f \in B_m$ es:

$$V_f = \left(f(x_0), f(x_1), \dots, f(x_{2^m-1})\right). \quad (2.6)$$

Esta representación establece un isomorfismo[6] entre los espacios vectoriales Z_2 y $Z_2^{2^m}$. De aquí que la dimensión de B_m es 2^m. El peso de la función booleana f se puede calcular directamente de su tabla de verdad [6]. Por ejemplo, para $m = 3$ variables, se puede considerar la siguiente función booleana:

Tabla 2.3 Tabla de verdad de una función booleana.

Entero Binario	0	1	2	3	4	5	6	7
x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
$f(x_1, x_2, x_3)$	0	1	1	0	1	0	0	1

Donde $f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3$.

De la tabla se puede inferir que:

- Las columnas de la tabla de verdad son enumeradas de 0 a $2^m - 1$ con x_3 como la variable con los bits menos significativos.

- Dado que $m = 3$ y teniendo en cuenta que toda función booleana es una palabra de longitud 2^m, ésta puede ser representada simplemente por el vector característico:

$$V_f = (0,1,1,0,1,0,0,1)$$

- Así, el peso de f es $w(f) = 4$.

2.5.2 Representación Polinomial

Una de las principales cualidades de la representación polinomial es que permite definir el grado algebraico de la función booleana. A partir de esta definición es

[6] El concepto matemático de isomorfismo (del griego iso-morfos: Igual forma) pretende captar la idea de tener la misma estructura.

posible definir de manera simple los códigos bloque lineales RM. Toda función booleana f admite una representación polinomial en el espacio conocida como forma normal algebraica [13].

Para entender mejor el concepto se deben definir los monomios y polinomios de Boole.

Un monomio de Boole en las variables $x_1, x_2, \ldots, x_m$ es un elemento p del Campo de Galois $GF(2)$ que tiene la forma:

$$p = x_1^{r_1} x_2^{r_2} \ldots x_m^{r_m}. \tag{2.7}$$

Donde r_i es el grado de la variable x_i. El monomio 1 es $p = x_1^0 x_2^0 \ldots x_m^0$.

El monomio p puede ser reducido aplicando las reglas:

$$x_i x_j = x_j x_i$$

$$x_i x_i = x_i^2.$$

El grado de un monomio de Boole es el número de variables que contiene el monomio en forma reducida.

Un polinomio de Boole es una combinación lineal de monomios de Boole con coeficientes en Z_2. Dicho polinomio se encuentra en forma reducida si cada monomio que lo compone también lo está. El grado de un polinomio de Boole corresponde al mayor grado entre los monomios que lo conforman.

Por ejemplo, la función $f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3$ es la representación polinomial de la función f del ejemplo anterior. La representación corresponde a una combinación lineal de los monomios (x_1, x_2, x_3). Sin embargo, no es sencillo obtener una representación polinomial única de una función booleana, por ejemplo, el polinomio $p(x_1, x_2, x_3) = x_1^2 \oplus x_2 \oplus x_3$ también constituye una representación polinomial de la función booleana de la tabla 3.

El conjunto de todas las funciones booleanas B_m tiene las siguientes bases:

$$\{x_0, x_1, x_2, \ldots, x_{m-1}, x_m, \ldots, x_1 x_2, x_1 x_3, \ldots, x_1 x_m, \ldots, x_1 x_2 x_3 \ldots x_m\}.$$

Siendo x_0 el vector compuesto por 2^m 1's. Entonces una función booleana puede representarse como una combinación lineal de las bases del conjunto B_m:

$$f = a_0x_0 + \ a_1x_1 + a_2x_2 + \ \ldots + a_mx_m + a_{12}x_1x_2 + a_{12\ldots m}x_1x_2 \ldots x_m. \qquad (2.8)$$

Donde $\{a_{0,}a_1, \ldots, a_m, \ldots, a_{12\ldots m}\}$ son coeficientes en Z_2.

Por ejemplo si $m = 3$, las funciones booleanas tienen la siguiente forma:

$$f = a_0 1 + \ a_1x_1 + a_2x_2 + a_3x_3 + a_{12}x_1x_2 + a_{13}x_1x_3 + a_{23}x_2x_3 + a_{123}x_1x_2x_3$$

2.6 Definición del Código Reed-Muller

Un código binario $RM(r, m)$ de orden r y longitud 2^m está conformado por todas las combinaciones lineales posibles entre vectores asociados a monomios de m variables, con grado menor o igual r [12].

Dicho de otra manera, el código RM puede ser visto como el conjunto de todos los polinomios de Boole posibles (según los valores de r y m), ya que cada uno de dichos polinomios es simplemente una combinación lineal de monomios de Boole, con coeficientes en $GF(2)$.

2.6.1 Características Generales del Código Reed-Muller

Los códigos RM se especifican mediante dos parámetros enteros, a saber: $m \geq 1$ y $0 \leq r \leq m$. Para cada posible elección de r y m existe un código bloque lineal binario de la forma $(n, k, d_{mín})$, donde n representa la longitud de las palabras código, k la cantidad de bits de información en cada palabra código y $d_{mín}$ la distancia mínima Hamming del código. Es conveniente aclarar que los códigos RM no son de naturaleza cíclica [2].

Los parámetros $n, k, d_{mín}$ de un código $RM(r, m)$ se pueden obtener a partir de r y m de la siguiente manera:

$$n = 2^m, \qquad (2.9)$$

$$k(r, m) = \sum_{j=0}^{r} \binom{m}{j}, \qquad (2.10)$$

donde $\binom{m}{j} = \frac{m!}{j!(m-j)!}$.

$$d_{mín}(r,m) = 2^{m-r}. \tag{2.11}$$

De esa forma, el código $RM(r,m)$ es un código bloque lineal $(2^m, k, 2^{m-r})$.

Una de las características más llamativas de esta familia de códigos es la manera como se puede obtener explícitamente la distancia mínima, como se indica en la ecuación 2.11.

La redundancia en un código RM, denotada por la letra h, es la diferencia entre la longitud del bloque codificado n y la longitud de información binaria k, esto es,

$$h = n - k. \tag{2.12}$$

El proceso de codificación bloque lineal, en general, consiste en adicionar símbolos de redundancia al bloque de información. En el proceso inverso, realizado por el decodificador, la palabra recibida del canal de comunicaciones se analiza para detectar y corregir los errores presentes en el mensaje, logrando disminuir la BER.

La tasa de codificación, o eficiencia espectral, de un código bloque lineal, T, es una cantidad adimensional definida como:

$$T = \frac{k}{n} = \frac{k}{2^m}. \tag{2.13}$$

Todo código$(n, k, d_{mín})$ puede detectar hasta $d_{mín} - 1$ y corregir hasta $\left\lfloor \frac{d_{mín}-1}{2} \right\rfloor$ errores en cada palabra código de n bits. En ese sentido, la capacidad de corrección de error de un código $RM(r,m)$ denotada con la letra t es:

$$t = \left\lfloor \frac{2^{m-r} - 1}{2} \right\rfloor. \tag{2.14}$$

Los códigos que comparten los mismos valores de algunos de los parámetros descritos anteriormente, comparten al mismo tiempo ciertas particularidades. La tabla 2.4 muestra la clasificación de los códigos RM según los parámetros $n, k, d_{mín}$. Cada tipo de códigos descrito en la tabla 2.4 comparten características comunes con respecto a su función, la proyección geométrica de los símbolos en el plano o la semejanza con relación a otros códigos de bloque lineales.

Tabla 2.4 Códigos Reed-Muller de longitudes de hasta 32.

						$RM(m,m)$ $(2^m, 2^m, 1)$	**Códigos universales**
					$RM(5,5)$ $(32,32,1)$		
				$RM(4,4)$ $(16,16,1)$		$RM(m-1,m)$ $(2^m, 2^m-1,2)$	**Códigos SPC**
			$RM(3,3)$ $(8,8,1)$		$RM(4,5)$ $(32,31,2)$		
		$RM(2,2)$ $(4,4,1)$		$RM(3,4)$ $(16,15,2)$		$RM(m-2,m)$ $(2^m, 2^m-m-1,4)$	**Códigos Hamming ext.**
	$RM(1,1)$ $(2,2,1)$		$RM(2,3)$ $(8,7,2)$		$RM(3,5)$ $(32,26,4)$		
$RM(0,0)$ $(1,1,1)$		$RM(1,2)$ $(4,3,2)$		$RM(2,4)$ $(16,11,4)$			
	$RM(0,1)$ $(2,1,2)$		$RM(1,3)$ $(8,4,4)$		$RM(2,5)$ $(32,16,8)$		**Códigos auto-duales**
$RM(-1,0)$ $(1,0,\infty)$		$RM(0,2)$ $(4,1,4)$		$RM(1,4)$ $(16,5,8)$			
	$RM(-1,1)$ $(2,0,\infty)$		$RM(0,3)$ $(8,1,8)$		$RM(1,5)$ $(32,6,16)$		
		$RM(-1,2)$ $(4,0,\infty)$		$RM(0,4)$ $(16,1,16)$		$RM(1,m)$ $(2^m, m+1, 2^{m-1})$	**Códigos biortogonales**
			$RM(-1,3)$ $(8,0,\infty)$		$RM(0,5)$ $(32,1,32)$		
				$RM(-1,4)$ $(16,0,\infty)$		$RM(0,m)$ $(2^m, 1, 2^m)$	**Códigos de repetición**
					$RM(-1,5)$ $(32,0,\infty)$		
						$RM(-1,m)$ $(2^m, 0, \infty)$	**Códigos triviales**

2.7 Proceso de Codificación Reed-Muller

La forma en la que se construye una palabra código en un codificador RM es la siguiente [14]:

$$C = M \cdot G_{RM(r,m)}. \tag{2.15}$$

Donde C es un vector que contiene los elementos $[c_1, c_2, \dots, c_n]$ y corresponde a la palabra código, resultado de la multiplicación matricial entre el mensaje M con elementos $[m_1, m_2, \dots, m_k]$ y la matriz generadora del código RM denotada $G_{RM(r,m)}$.

2.7.1 Matriz Generadora

Considerando el mapeo[7] entre monomios booleanas del conjunto B_m y vectores del Campo de Galois binario definido como sigue [14]:

$\Psi: B_m \rightarrow GF(2)$:

$$\Psi(0) = \underbrace{00\ldots0}_{2^m}$$
$$\Psi(1) = \underbrace{11\ldots1}_{2^m}$$
$$\Psi(x_1) = \underbrace{11\ldots1}_{2^{m-1}}\ \underbrace{00\ldots0}_{2^{m-1}}$$
$$\Psi(x_2) = \underbrace{11\ldots1}_{2^{m-2}}\ \underbrace{00\ldots0}_{2^{m-2}}\ \underbrace{11\ldots1}_{2^{m-2}}\ \underbrace{00\ldots0}_{2^{m-2}}$$
$$\Psi(x_3) = \underbrace{11\ldots1}_{2^{m-3}}\ \underbrace{00\ldots0}_{2^{m-3}}\ \underbrace{11\ldots1}_{2^{m-3}}\ \underbrace{00\ldots0}_{2^{m-3}}\ \underbrace{11\ldots1}_{2^{m-3}}\ \underbrace{00\ldots0}_{2^{m-3}}\ \underbrace{11\ldots1}_{2^{m-3}}\ \underbrace{00\ldots0}_{2^{m-3}}$$
$$\vdots$$
$$\Psi(x_i) = \underbrace{11\ldots1}_{2^{m-i}}\ \underbrace{00\ldots0}_{2^{m-i}}$$
$$\vdots$$
$$\Psi(x_m) = \underbrace{101010\ldots}_{2^{m-m}}$$

La matriz generadora para los códigos $RM(r,m)$ está organizada de la siguiente manera:

- La primera fila corresponde al vector que representa al monomio 1.
- Las siguientes m filas corresponden a los vectores que representan a los monomios de grado 1, estos monomios son: $x_1, x_2, \ldots, x_m$.
- Las filas que completan el arreglo son las correspondientes a los vectores que resultan de combinar entre sí a los monomios de grado 1 hasta obtener monomios a lo sumo de grado r.

Entonces la matriz generadora del código $RM(r,m)$ de tamaño $k \times n$ tiene la siguiente forma:

[7] Mapeo: Trazado de un conjunto de elementos de cierto tipo de datos a otro tipo diferente de conjunto.

$$G_{RM(r,m)} = \begin{bmatrix} \Psi(1) \\ \Psi(x_1) \\ \Psi(x_2) \\ . \\ . \\ . \\ \Psi(x_m) \\ \Psi(x_1)\Psi(x_2) \\ \Psi(x_1)\Psi(x_3) \\ . \\ . \\ . \\ \Psi(x_{m-1})\Psi(x_m) \\ \Psi(x_1)\Psi(x_2)\Psi(x_3) \\ . \\ . \\ . \\ \Psi(x_{m-r+1})\Psi(x_{m-r+2}) \dots \Psi(x_m) \end{bmatrix}_{k\times n} \tag{2.16}$$

A continuación se expone un ejemplo donde se ilustra el mecanismo de obtención de la matriz generadora para un código $RM(2,4)$.

Sean $r = 2$ y $m = 4$, el código RM de segundo orden tiene los siguientes parámetros:

$$n = 2^m = 2^4 = 16$$

$$k(2,4) = \sum_{j=0}^{r} \binom{m}{j} = \sum_{j=0}^{2} \binom{4}{j} = \binom{4}{0} + \binom{4}{1} + \binom{4}{2} = 1 + 4 + 6 = 11$$

$$d_{mín}(2,4) = 2^{m-r} = 2^{4-2} = 4$$

De esa forma, el código $RM(2,4)$ es un código lineal $(16,11,4)$; con capacidad de corregir 1 error y detectar 3 errores por palabra código. Dicho código también se lo conoce como código Hamming extendido (obtenido al agregar un bit de paridad adicional al código Hamming $(15,11)$)

Según la definición de la matriz generadora, los $k = 11$ vectores de $n = 16$ bits de longitud que componen la matriz son los mostrados en la tabla 5:

Tabla 2.5 Vectores de la matriz generadora del código RM(2,4)

$\Psi(1)$	1111111111111111
$\Psi(x_1)$	0000000011111111
$\Psi(x_2)$	0000111100001111
$\Psi(x_3)$	0011001100110011
$\Psi(x_4)$	0101010101010101
$\Psi(x_1)\Psi(x_2)$[8]	0000000000001111
$\Psi(x_1)\Psi(x_3)$	0000000000110011
$\Psi(x_1)\Psi(x_4)$	0000000001010101
$\Psi(x_2)\Psi(x_3)$	0000001100000011
$\Psi(x_2)\Psi(x_4)$	0000010100000101
$\Psi(x_3)\Psi(x_4)$	0001000100010001

Entonces el código $RM(2,4)$ tiene la siguiente matriz generadora:

$$G_{RM(2,4)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{k \times n}$$

2.7.2 Cálculo de una palabra código

Supóngase que se desea codificar la palabra mensaje $M = [0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0]$ con un código $RM(2,4)$. Utilizando la ecuación (2.15), la multiplicación vectorial con la matriz generadora da como resultado la palabra código:

$$C = M \cdot G_{RM(r,m)}$$

[8] La operación $\Psi(x_i)\Psi(x_j)$ se realiza mediante la multiplicación binaria sobre el campo de Galois $GF(2)$, definida en la sección 2.4.

$$C = [0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0]_{1\times k} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{k\times n}$$

$$C = [0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]_{1\times n}$$

El vector resultante de la multiplicación matricial, corresponde a la palabra código de longitud $n = 16$ bits.

De acuerdo con [15], no hay un orden específico de preferencia para disponer los elementos y las variables booleanas, por lo tanto la matriz generadora $G_{RM(r,m)}$ no es única.

2.8 Recursividad de los Códigos Reed-Muller

Como se mencionó anteriormente, los códigos RM más grandes pueden ser construidos recursivamente utilizando códigos RM más pequeños. Formalmente el código $RM(r,m)$ de longitud 2^m se puede generar a partir de los códigos $RM(r,m-1)$ y $RM(r-1,m-1)$ de longitud 2^{m-1}. Las palabras código de los códigos inmediatamente anteriores están asociadas con las funciones booleanas en $m-1$ variables, con grado menor o igual que r y menor o igual que $r-1$ respectivamente. En la ecuación 2.17 se define la recursividad de los códigos RM [6]:

$$RM(r,m) = \{(\boldsymbol{u}, \boldsymbol{u}+\boldsymbol{v}) : \boldsymbol{u} \in RM(r,m-1) \text{ y } \boldsymbol{v} \in RM(r-1,m-1)\}. \quad (2.17)$$

Los vectores $\boldsymbol{u}$ y $\boldsymbol{v}$ pertenecen a los códigos $RM(r,m-1)$ y $RM(r-1,m-1)$ respectivamente; por lo tanto, las combinaciones lineales $(\boldsymbol{u}, \boldsymbol{u}+\boldsymbol{v})$ deben ser palabras código en $RM(r,m)$.

La distancia mínima del código $RM(r,m)$ corresponde al valor mínimo entre dos veces la distancia mínima del código $RM(r,m-1)$ y la distancia mínima del código $RM(r-1,m-1)$, esto es:

$$d_{mín}(r,m) = mín\,\{2d_{mín}(r-1,m-1), d_{mín}(r,m-1)\}$$

$$= mín\{2(2^{m-r-1}), 2^{m-r}\}$$

$$d_{mín}(r,m) = 2^{m-r}. \tag{2.18}$$

La matriz generadora construida recursivamente es:

$$G_{RM(r,m)} = \begin{bmatrix} G_{RM(r,m-1)} & G_{RM(r,m-1)} \\ \mathbf{0} & G_{RM(r-1,m-1)} \end{bmatrix}. \tag{2.19}$$

Donde $\mathbf{0}$ es una matriz de ceros de dimensión $k(r-1, m-1) \times 2^{m-1}$.

La matriz 2.19 muestra que en efecto los códigos RM pueden construirse recursivamente a partir de códigos RM más pequeños mediante una secuencia de construcciones $(\boldsymbol{u}, \boldsymbol{u}+\boldsymbol{v})$. Por ejemplo el código $RM(r,m)$ también puede construirse mediante los códigos $RM(r,m-2)$, $RM(r-1,m-2)$ y $RM(r-2,m-2)$, todos de longitud 2^{m-2}. La matriz generadora en términos de estos códigos es:

$$G_{RM(r,m)} = \begin{bmatrix} G_{RM(r,m-2)} & G_{RM(r,m-2)} & G_{RM(r,m-2)} & G_{RM(r,m-2)} \\ \mathbf{0} & G_{RM(r-1,m-2)} & \mathbf{0} & G_{RM(r-1,m-2)} \\ \mathbf{0} & \mathbf{0} & G_{RM(r-1,m-2)} & G_{RM(r-1,m-2)} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & G_{RM(r-2,m-2)} \end{bmatrix}. \tag{2.20}$$

Esta estructura también permite considerar mecanismos de decodificación basados en decisión suave multi-etapa para aquellos códigos RM con baja probabilidad de error, ya que dichos mecanismos disminuyen la complejidad de la decodificación [16].

De la construcción del código, se observa que el código $RM(r-1,m)$ es un subcódigo del código $RM(r,m)$. Según lo mencionado, se tiene la siguiente cadena de inclusión [2]:

$$RM(0,m) \subset RM(1,m) \subset \cdots \subset RM(r,m). \tag{2.21}$$

Los códigos RM de igual longitud y orden menor que r están incluidos en el código $RM(r,m)$. Por su parte, los códigos $RM(0,m)$ son códigos de repetición. Adicionalmente, el código dual de un código $RM(r,m)$ es un código $RM(m-r-1,m)$ [2].

2.9 Proceso de Decodificación Reed-Muller

La decodificación es la operación inversa a la codificación en la cual se detectan y/o corrigen los errores introducidos por el canal en la palabra codificada C. Las técnicas de decodificación RM son similares a las de códigos bloque lineales convencionales. En este trabajo de grado se analizan dos técnicas de decodificación: decodificación por síndrome (*hard decision*) y la decodificación por distancia Euclidiana (*soft decision*).

2.9.1 Decodificación *Hard Decision*

El primer paso en la decodificación *hard decision* consiste en comparar cada muestra de la señal recibida con un umbral de decisión, esto es, si la muestra está por encima del umbral se decide por un "1", de lo contrario, por un "0". Después de obtener la palabra código de n bits, se verifica si ésta corresponde a una palabra del alfabeto del código. En caso afirmativo, se la deja tal cual y en caso negativo, se calcula la distancia Hamming entre dicha palabra y cada uno de los elementos del alfabeto. Al finalizar el cálculo, se sustituye la palabra recibida por aquella del alfabeto que se encuentre a la menor distancia Hamming. Si existen 2 o más palabras del alfabeto a la misma distancia mínima Hamming, el decodificador elegirá una de ellas al azar [6].

La palabra decodificada, C', correspondiente a la palabra recibida, R, se puede obtener como:

$$C' = \underset{B \in \mathcal{A}}{\operatorname{argmin}}[w(R \oplus B)], \tag{2.22}$$

donde B es una palabra que pertenece al alfabeto del código denotado por $\mathcal{A}$ y $w(R \oplus B)$ es el peso Hamming de la suma módulo 2 entre R y B.

Este algoritmo es válido para cualquier código bloque lineal y su capacidad de detección y corrección de errores depende directamente de la distancia entre las palabras código del alfabeto.

Existe un método general de decodificación para códigos bloque lineales creado por Slepian, denominado *decodificación por síndrome*, el cual permite agilizar el proceso de decodificación *hard decision*. En la sección 2.8.2 se describe su funcionamiento.

2.9.2 Método de Decodificación por Síndrome

Cuando se envía una palabra código C a través de un canal ruidoso, en el extremo receptor se recibe una palabra código R, siendo ésta una versión "ruidosa" de C. R es la palabra codificada originalmente (C) alterada por un patrón de error E como se presenta en la siguiente ecuación [2]:

$$R = C \oplus E. \tag{2.23}$$

El patrón de error E es un vector de peso mínimo y alta probabilidad de ocurrencia, el cual representa la "diferencia (en álgebra módulo 2)" entre la palabra código enviada C y la palabra código recibida R, por lo tanto[2]:

$$E = R \oplus C, \tag{2.24}$$

donde el vector E es una n-tupla constituida por los elementos $[e_1, e_2, \dots, e_n]$, siendo $e_i = 1$ si existe un error en la posición i y $e_i = 0$ si no existe error.

Si se Supone un código binario $(n, k, d_{mín})$ con matriz de verificación de paridad H y $R \in F_2^n$, el cálculo del síndrome de R se define por la siguiente ecuación [2]:

$$s(R) = RH^T = CH^T \oplus EH^T,$$

$$s(R) = RH^T = EH^T, \tag{2.25}$$

dado que $CH^T = 0$ para toda palabra de n bits que pertenece al código. Lo anterior demuestra que el síndrome sólo depende del patrón de error.

Previamente, para lograr el cálculo del síndrome es estrictamente necesaria la obtención de la matriz de verificación de paridad H. El cálculo de la matriz de verificación de paridad es posible gracias a una característica de los códigos bloque llamada dualidad la cual indica que la matriz generadora de un código RM es la matriz de verificación de paridad de otro código RM [2], como se presenta en la siguiente igualdad:

$$H_{RM(r,m)} = G_{RM(m-r-1,m)}, \tag{2.26}$$

El proceso de *decodificación por síndrome* se desarrolla con el llamado arreglo Slepiano para $\mathcal{A}$ [2]:

$$\begin{array}{ccccc} \mathbf{0} & C_1 & C_2 & \dots & C_r \\ E_1 & C_1+E_1 & C_2+E_1 & \dots & C_r+E_1 \\ E_2 & C_1+E_2 & C_2+E_2 & \dots & C_r+E_2. \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ E_s & C_1+E_s & C_2+E_s & \dots & C_r+E_s \end{array} \tag{2.27}$$

Donde $r = 2^k - 1$ y $s = 2^{n-k} - 1$. El arreglo se construye en los siguientes pasos:

1. Se ubican las 2^k palabras código posibles C_i en la primera fila con el vector nulo en la primera posición.
2. En la segunda fila, a cada palabra C_i se le suma el patrón de error E_i con la más alta probabilidad de ocurrencia y menor peso Hamming, tal que no se encuentre en el arreglo previamente.
3. El proceso descrito en el numeral 2 se repite para $2^{n-k} - 1$ patrones de error.

Los patrones de error de mayor probabilidad de ocurrencia son los que conforman la primera columna del arreglo.

Al analizar el arreglo Slepiano es posible deducir que todas las palabras de una misma fila comparten el mismo valor de síndrome. Según el síndrome de la palabra código recibida R, se selecciona el patrón de error E, ubicado en la primera posición de la fila a la que dicho síndrome pertenece.

La nueva palabra código recibida se calcula mediante la siguiente ecuación:

$$C' = R \oplus E \tag{2.28}$$

En resumen, la *decodificación por síndrome* se constituye de tres pasos fundamentales [2]:

- Calcular el síndrome de la palabra recibida, $s(R)$.
- Asociar dicho síndrome con el patrón de error respectivo E_i .
- Evaluar la ecuación $C' = R \oplus E$, donde C' es ahora la palabra código corregida.

Para aclarar lo mencionado, en el anexo C se lleva a cabo el proceso de decodificación por síndrome de un mensaje codificado con un código RM que se envía por un canal que introduce ruido.

2.9.3 Decodificación *Soft Decision*

El proceso de decodificación s*oft decision* se basa en el cálculo de la distancia Euclidiana. En primer lugar, las muestras de la señal recibida (previas al decisor) forman una n-tupla de números reales, X, la cual se compara con la imagen Euclidiana de cada una de las palabras código del alfabeto $\mathcal{A}$. Luego, la palabra código decodificada será aquel elemento de $\mathcal{A}$ que se encuentre a la menor distancia Euclidiana de X [17], esto es:

$$C' = \underset{B \in \mathcal{A}}{\operatorname{argmin}} \left\{ \left\| X - \left(B - \frac{1}{2} \right) \right\|^2 \right\}, \tag{2.29}$$

donde el operador $\|\cdot\|$ denota norma Euclidiana y $B - \frac{1}{2}$ representa la imagen Euclidiana de la palabra código B (asumiendo un esquema de modulación binario como 2-PAM ó BPSK con símbolos $\frac{1}{2}$ y $-\frac{1}{2}$).

En otras palabras si $Y = [y_1, y_2, \dots, y_n]$ representa la imagen Euclidiana de una palabra código $C \in \mathcal{A}$ y $X = [x_1, x_2, \dots, x_n]$ es la n-tupla formada por las muestras recibidas, se debe buscar aquella palabra código del alfabeto que minimice la cantidad [18]:

$$d_{XY} = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}. \tag{2.30}$$

Los decodificadores *soft decision* utilizan toda la información de la señal recibida (niveles de voltaje), mientras los decodificadores *hard decision* no utilizan toda la información disponible en tal señal. El uso de *soft decision* mejora el desempeño en el receptor, incluso, provee la capacidad de corregir errores cuando el esquema *hard decision* no lo permite ($t = 0$). Evidentemente en el esquema *soft decision* la capacidad de corrección no depende directamente de la distancia Hamming.

La decodificación *soft decision* consta de los siguientes pasos fundamentales:

- Cálculo de la distancia mínima Euclidiana entre las secuencias de muestras recibidas y cada palabra código del alfabeto.
- Elegir como palabra decodificada aquella que se encuentre a la menor distancia Euclidiana de la secuencia de muestras recibidas.

El algoritmo de decodificación *soft decision* proporciona mejores prestaciones en términos de probabilidad de error. La desventaja es el aumento de complejidad del decodificador, que debe trabajar con información adicional [19].

2.9.4 Probabilidad de un Error de Decodificación

Un error de decodificación es el nombre dado a una operación errada de Detección y Corrección de Errores (*EDAC*, Error Detection And Correction) donde el mensaje de información recuperado es un mensaje incorrecto. La probabilidad de un error de decodificación es la medida de la frecuencia relativa promedio con que se presenta dicho error. Cabe aclarar que un error en la decodificación de una palabra no es lo mismo que un error en la detección de un bit individual, de hecho, algunos códigos tienen la propiedad de recuperar correctamente la secuencia mensaje a pesar de la ocurrencia de errores en algunos bits.

2.9.4.1 Probabilidad de un Error de Decodificación *Hard Decision*

Los errores de decodificación se dan cuando el número de bits erróneos o incorrectos en una palabra código recibida supera el valor del parámetro t, es decir la capacidad de corrección del código.

En el apéndice A se obtiene la probabilidad de error de bit con el método de decodificación *hard decision*, del cual resulta la expresión:

$$\Pr\{\mathcal{E}b\} = \frac{2^k/2}{2^k - 1} \sum_{i=t+1}^{n} \binom{n}{i} p^i (1-p)^{n-i}, \tag{2.31}$$

donde $\mathcal{E}b$ es el evento “error en la detección de un bit de información” y p la probabilidad de error en un bit. Esta probabilidad depende del esquema de modulación utilizado. Teniendo en cuenta que la energía de bit se ve afectada por la tasa de codificación y suponiendo un esquema de modulación binario como 2-PAM, la probabilidad de error está dada por la siguiente ecuación:

$$p = Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}}\right). \tag{2.32}$$

El parámetro Eb/No depende estrictamente del canal de comunicación. En apartados posteriores se analizará en detalle.

2.9.4.2 Probabilidad de un Error de Decodificación *Soft Decision*

La probabilidad de error de bit con decodificación *soft decision* se obtiene en el apéndice B, y da como resultado la siguiente ecuación:

$$\Pr\{\mathcal{E}b\} = \frac{2^k - 1}{2}\, Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}d_H}\right). \tag{2.33}$$

2.9.5 Ganancia de Codificación

La ganancia de codificación es un parámetro que determina el número de decibeles en que es posible reducir en la relación Eb/No para alcanzar una misma probabilidad de error respecto a un esquema de transmisión no codificado. Esta cantidad, por tanto, mide la mejora proporcionada al usar un determinado código [20]. El efecto de este parámetro se verá reflejado gráficamente en las curvas de desempeño de los códigos RM al ser comparadas con las curvas de desempeño de un sistema sin codificación. La ganancia de codificación, G, se puede obtener de la siguiente manera:

$$G(dB) = \left(\frac{Eb}{No}\right)_U (dB) - \left(\frac{Eb}{No}\right)_C (dB), \tag{2.34}$$

donde $\left(\frac{Eb}{No}\right)_U$ es la relación energía de bit a densidad de ruido requerida sin codificación y $\left(\frac{Eb}{No}\right)_C$ es la relación energía de bit a densidad de ruido requerida con codificación.

3. Desarrollo de la Simulación de un Sistema con Codificación Reed-Muller

3.1 Introducción

Para el desarrollo de la simulación del sistema de comunicación digital en cuestión, se hace una adaptación de la metodología planteada en la guía para Simulación de Sistemas de Telecomunicaciones descrita por Astaiza en [21]. Con ello se busca diseñar un algoritmo propio de simulación, que permita realizar los procesos de codificación y decodificación asociados con la codificación de canal RM.

3.2 Metodología de Simulación

La metodología de Simulación de Sistemas de Telecomunicaciones planteada en [21] consta de las fases que se resumen en la Figura 3.1.

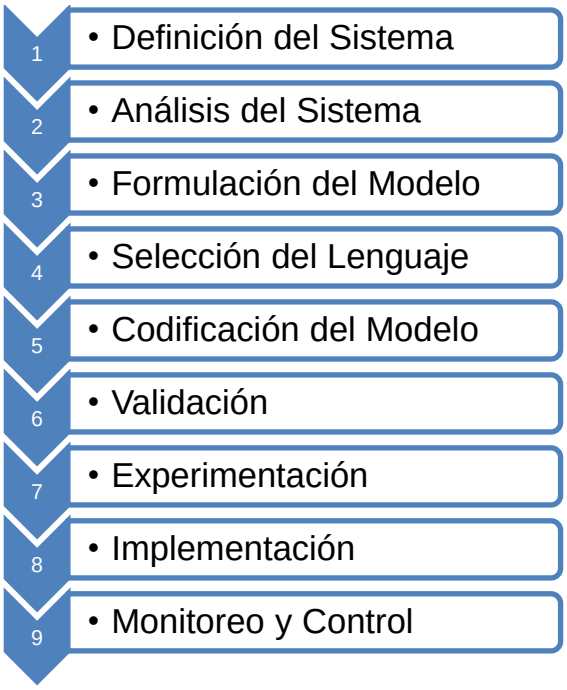


Figura 3.1 Metodología para la creación y desarrollo de una simulación.

3.2.1 Definición del Sistema

Se desarrollará un sistema de comunicación digital en banda base con modulación 2-PAM, que utilice la codificación de canal RM, para la transmisión sobre un canal AWGN, empleando en recepción los métodos de decodificación *hard decision* y *soft decision*.

Cada uno de los módulos del sistema se explica siguiendo las fases 2 y 3 de la metodología de Simulación de Sistemas de Telecomunicaciones.

3.2.2 Análisis del Sistema

En esta fase se describen las interacciones lógicas que existen entre las variables de decisión inmersas en el sistema de comunicación. Las variables de decisión interactúan en función de las variables no controlables de tal manera que se optimice el desempeño del sistema, en cuanto a BER, a medida que aumenta la relación Eb/No.

Variables de decisión:

- Esquema de modulación.
- Parámetros de la codificación RM:
  - r que es el grado de la codificación.
  - m que define la longitud del código.
- Número de bits a transmitir k.
- Número de bits codificados n.
- Distancia mínima Hamming entre palabras codificadas d_H.
- Modelo de canal de comunicación.
- Método de decodificación.
- Eb/No.

Variables no controlables:

- BER.

El esquema de modulación adoptado es 2-PAM; los códigos RM utilizados para el proceso de codificación son RM(0,3), RM(1,3), RM(2,3) y RM(2,4); se considera un canal AWGN con distribución de probabilidad Gaussiana de media nula y varianza σ^2; y por último, los métodos de decodificación *hard decisión* y *soft decisión*.

Cualquier objeto dentro del sistema se denomina entidad y puede ser estática o dinámica. Si la entidad es dinámica se denota como una transacción y su principal característica es su movimiento a través de las entidades estáticas del sistema. Las entidades poseen propiedades llamadas atributos que las caracterizan. Las entidades dinámicas y estáticas del sistema se muestran en las tablas 3.1 y 3.2 respectivamente.

Tabla 3.1 Entidades dinámicas.

Entidad Dinámica	**Atributos**
Fuente Digital	✓ Número de bits de información
Codificador de Canal RM	✓ Grado ✓ Longitud de palabra código ✓ Matriz generadora
Decodificador	✓ Técnica: *hard decision, soft decision*

Tabla 3.2 Entidades estáticas.

Entidad Estática	**Atributos**
Modulación	✓ Esquema ✓ Orden
Canal de Comunicación	✓ Modelo de Canal: AWGN

3.2.3 Formulación del Modelo

En esta fase se muestra el diagrama en bloques del sistema de comunicación diseñado para la simulación, se describe el algoritmo lógico que define las

interacciones entre las variables del sistema, y por último se explican los diagramas de flujo correspondientes a los procesos de transmisión y recepción.

3.2.3.1 Diagrama en bloques del sistema de comunicación digital en banda base con codificación Reed-Muller

La figura 3.2 muestra el diagrama en bloques del sistema de comunicación digital en banda base con codificación RM sobre un canal AWGN. A continuación se describe la salida de cada uno de los bloques, la cual constituye la entrada del siguiente.

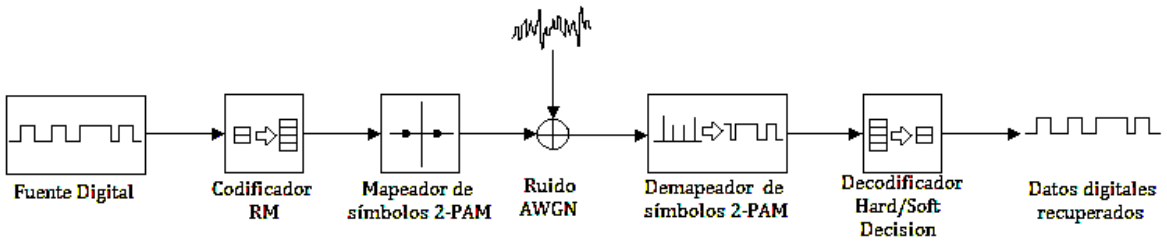


Figura 3.2 Diagrama en bloques del sistema de comunicación digital en banda base con codificación RM.

Es importante aclarar que en el sistema de comunicaciones a evaluar no se tiene en cuenta ningún tipo de filtraje, sincronización de portadora o estimación de temporización de símbolos, pues se asume que dichas operaciones han sido realizadas de manera ideal. En otras palabras, el único efecto degradante que se considera en las simulaciones es el ruido blanco aditivo Gaussiano (AWGN).

Fuente Digital: Es un generador aleatorio de números enteros equiprobables pertenecientes al conjunto $\{0,1,\dots,2^k-1\}$, donde k depende del código RM que se utilice. En este mismo bloque, cada número entero es convertido a formato binario para representar la secuencia M de k bits que pasa al bloque de codificación.

Codificador RM: Se encarga de convertir una secuencia de k bits en una de n bits (agregando redundancia a los datos). En principio se consideran bits estadísticamente independientes entre sí. El codificador utilizado es un codificador de canal que genera un código $RM(r,m)$. La entrada y salida del bloque son señales binarias (secuencias de bits) que representan el bloque sin codificar M y el bloque codificado C respectivamente.

Mapeador de símbolos 2-PAM: En la transmisión de datos digitales en banda base[9] es común usar la modulación por amplitud de pulsos (M-PAM, M-ary Pulse Amplitude Modulation), debido a que utiliza eficientemente los recursos como potencia y ancho de banda. En el presente trabajo se hace uso de este tipo de modulación para realizar un mapeo binario 2-PAM a las palabras codificadas.

Mediante el mapeo, los bits de la palabra codificada se convierten en una secuencia de símbolos S. Un símbolo, para el caso de la modulación 2-PAM, es la representación de un bit en el espacio Euclidiano.

La regla de mapeo definida para 2-PAM es la siguiente:

$$(0 \rightarrow -\alpha)\ y\ (1 \rightarrow \alpha).$$

$$d = 2\alpha, \tag{3.1}$$

donde $d = 2\alpha$ es la distancia Euclidiana entre los dos símbolos de la constelación 2-PAM. La energía promedio de símbolo está dada por:

$$Es = \sum_{a_j \in A} \|a_j\|^2 \Pr(X = a_j), \tag{3.2}$$

donde a_j corresponde a la imagen Euclidiana de la palabra código. El número de palabras código posibles pertenecientes al código RM es 2^k, y por ser los símbolos equiprobables, la probabilidad de símbolo es igual a $1/2^k$.

La ecuación (3.2) se puede reescribir en términos de los parámetros de un código bloque lineal así:

$$Es = 2^k \frac{1}{2^k} n\alpha^2,$$

$$= n\alpha^2. \tag{3.3}$$

Luego la energía media transmitida por bit se define como:

$$Eb = \frac{Es}{k},$$

$$= \frac{n}{k}\alpha^2. \tag{3.4}$$

[9] La señal modulada es transmitida en su frecuencia original, es decir sin translación de frecuencia.

El parámetro Eb influye directamente en la calidad del sistema de comunicaciones, a través de la probabilidad de error, como se puede ver en apartados posteriores.

Canal AWGN: Este modelo de canal es muy usado en análisis de sistemas de comunicación. El ruido es un proceso aleatorio con densidad espectral de potencia constante sobre todas las frecuencias (ruido blanco), el cual se agrega a la señal original. Su distribución de probabilidad es Gaussiana con media nula y varianza σ^2 [4]. El vector aleatorio X resultante de contaminar la secuencia S con un vector de ruido Gaussiano N, está dado por:

$$X = S + N \tag{3.5}$$

Las ecuaciones (3.6) y (3.7) presentan respectivamente las expresiones para la función de densidad de probabilidad y la varianza del proceso de ruido AGWN:

$$f_N(u) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(\frac{u^2}{2\sigma^2}\right)}. \tag{3.6}$$

$$\sigma^2 = \frac{No}{2}. \tag{3.7}$$

Donde No es la densidad espectral unilateral de potencia de ruido, la cual se define matemáticamente como $No = K\,T_N$, siendo K la constante de Boltzmann ($K = 1{,}38\,x\,10 - 23\ [J/°K]$) y T_N la temperatura de ruido del receptor $[°K]$.

La desviación estándar de ruido, σ, correspondiente a un valor especifico de la relación energía de bit a densidad de ruido, Eb/No, se obtiene aplicando las ecuaciones (3.4) y (3.7):

$$\frac{Eb}{No} = \frac{Es/k}{2\sigma^2},$$

$$\sigma^2 = \frac{Es/k}{2Eb/No},$$

$$\sigma = \sqrt{\frac{\frac{n}{k}\alpha^2}{2Eb/No}}. \tag{3.8}$$

La relación energía de bit a densidad espectral de potencia de ruido, Eb/No, es una cantidad adimensional. En forma logarítmica se define como:

$$\frac{Eb}{No}_{[dB]} = 10\ \log\frac{Eb}{No}. \tag{3.9}$$

Cabe resaltar que la transmisión de un símbolo no está influenciada por la transmisión de los símbolos precedentes, es decir, el canal es *sin memoria*. El error en la transmisión de un símbolo no afecta a la transmisión de los siguientes símbolos.

De-mapeador de símbolos 2-PAM: En recepción se deben realizar las operaciones inversas a las realizadas en transmisión. Para modelar el receptor del sistema de comunicación, se toma un de-mapeador 2-PAM cuya función es tratar de recuperar la señal codificada a partir de los símbolos recibidos. La salida de este bloque es una secuencia de dígitos binarios que corresponden a una versión ruidosa R del vector C.

Decodificador *hard/soft decision*: Se utilizan separadamente las dos técnicas de decodificación: *hard decision,* utilizando el método de *decodificación por síndrome*, y *soft decision* que en su modo más simple selecciona la palabra recibida por medio del cálculo de distancias Euclidianas.

Datos originales recuperados: Una vez la señal ha pasado por el decodificador, se obtiene la señal de información recuperada y, con ella, es posible evaluar el desempeño del sistema de comunicaciones. La tasa de Error de Bit (BER) permite evaluar la calidad de un sistema de transmisión digital, mediante la relación entre el número de bits errados en recepción y el número total de bits transmitidos, esto es:

$$BER = \frac{Número\ de\ bits\ errados}{Número\ de\ bits\ transmitidos}. \tag{3.10}$$

El nivel de la BER que garantiza un grado de calidad mínimo es relativo al tipo de servicio ofrecido por el sistema de comunicación, por ejemplo, para transmisión de voz suele estar alrededor de 10^{-3}, mientras para transmisión de datos se debe mantener por debajo de 10^{-6}. En general, la BER es un parámetro que decrece con el aumento de la potencia de transmisión, por lo tanto el diseño de un sistema de comunicación es en últimas un proceso de negociación entre el grado de

calidad ofrecido y el consumo de recursos que implica la prestación de dicho servicio (potencia y ancho de banda) [22].

3.2.3.2 Diagrama de flujo del transmisor

El transmisor del sistema la componen 3 de los bloques funcionales definidos: la fuente digital, el codificador RM y el mapeador de símbolos 2-PAM. La fuente genera números enteros aleatorios entre 0 y $2^k - 1$ y los convierte en números binarios para generar vectores M de longitud k; el codificador RM genera vectores binarios C de longitud n, a través de la multiplicación entre los vectores M y la matriz generadora de los códigos RM, $G_{RM(r,m)}$, de tamaño $k \times n$; y por último el bloque mapeador de símbolos 2-PAM convierte los vectores binarios C en vectores reales S de números - α ó α también de tamaño n. La salida del bloque transmisor es el vector S, éste vector se envía al bloque del canal de comunicación. La figura 3.3 muestra el diagrama de flujo correspondiente al procesamiento de la señal en transmisión, donde:

M es el vector binario de información.

C es el vector binario codificado.

S es el vector real mapeado.

α es la constante con la que se representa la regla de mapeo.

$G_{RM(r,m)}$ es la matriz generadora de los códigos RM.

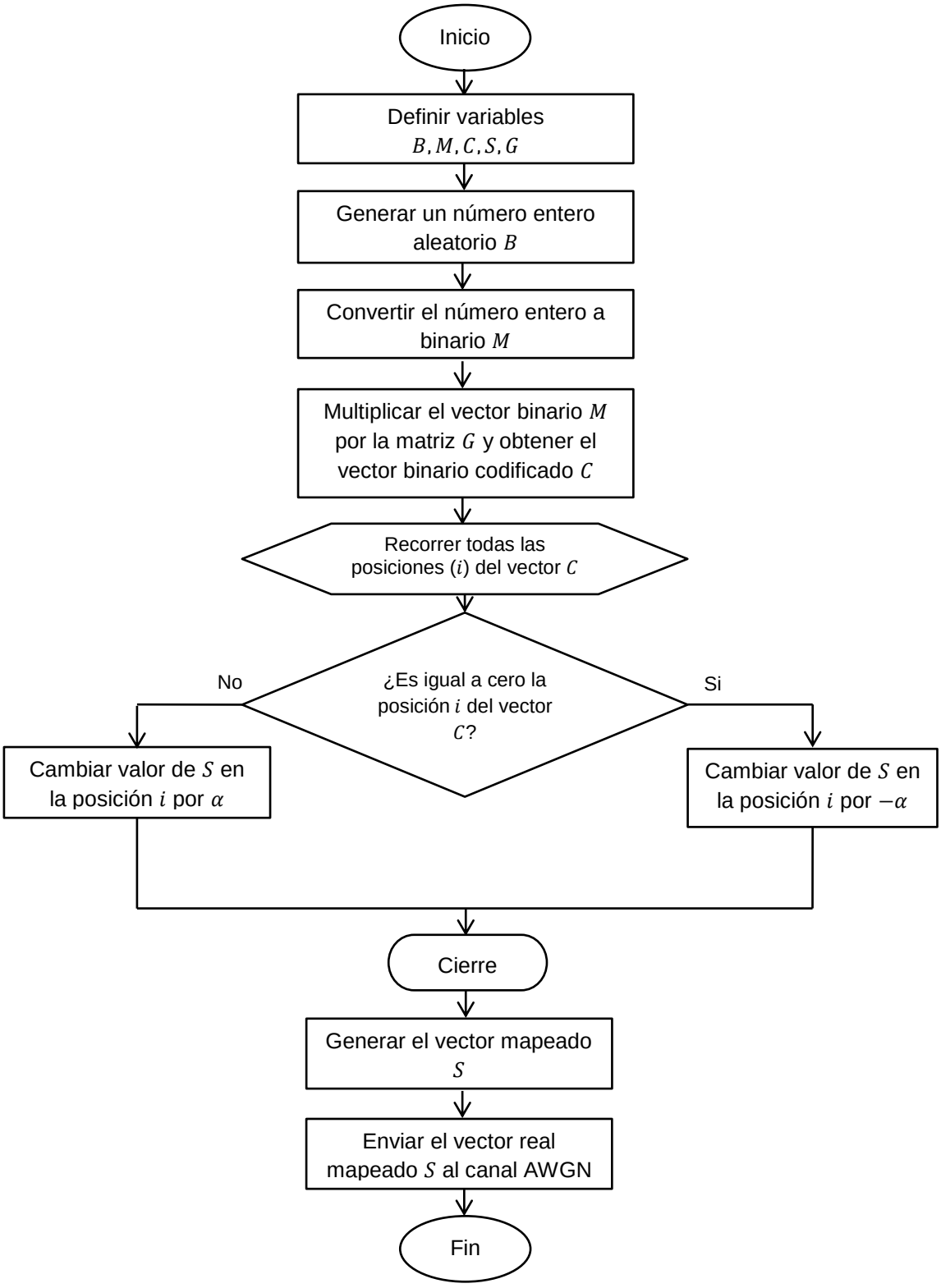


Figura 3.3 Diagrama de flujo del transmisor.

3.2.3.3 Diagrama de flujo del canal de comunicación

El canal de comunicación del sistema lo conforma el bloque de canal AWGN. La entrada de este bloque es el vector mapeado real S entregado por el bloque transmisor, y la salida es un vector real aleatorio X que resulta de sumar el vector real S recibido con un vector real aleatorio N que tiene el mismo tamaño. Cada componente del vector N es una elección aleatoria mediante una distribución normal que se multiplica por un factor correspondiente a la desviación estándar de la distribución, éste factor depende de la relación Eb/No. En la figura 3.4 se muestra el diagrama de flujo del canal de comunicación.

S es el vector real mapeado recibido del transmisor.

Eb/No es la relación Energía de Bit a Densidad Espectral de Potencia de Ruido.

σ es la desviación estándar.

V es el vector real de distribución normal.

N es el vector real aleatorio de ruido.

X es el vector real resultante a la salida del canal.

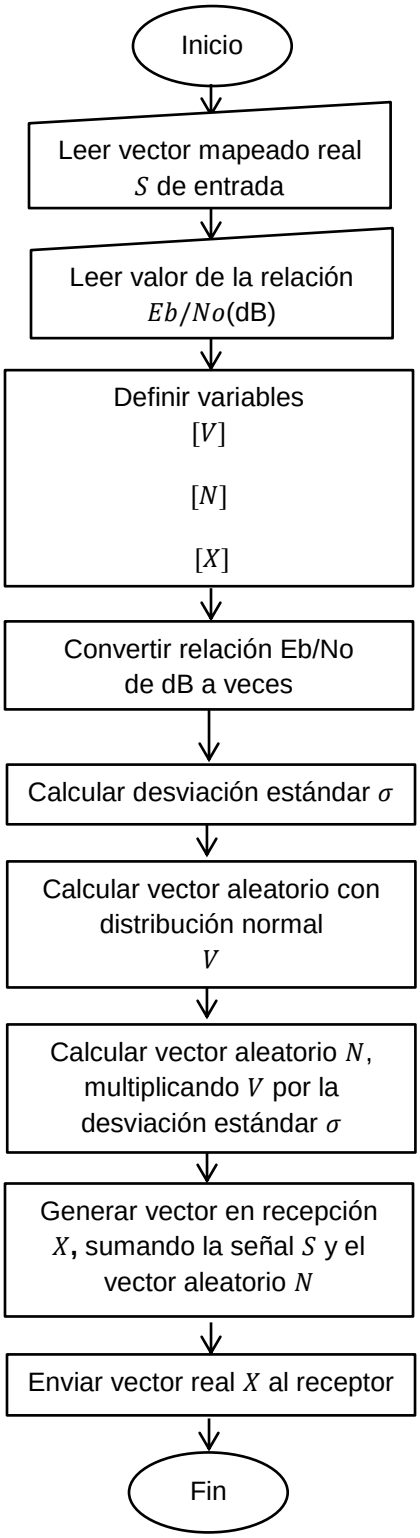


Figura 3.4 Diagrama de flujo del canal de comunicación.

3.2.3.4 Diagramas de flujo del receptor

El receptor está compuesto por 3 bloques funcionales: el demapeador de símbolos 2-PAM, el decodificador RM y por último el destino. El proceso de recepción se realiza separadamente de dos maneras utilizando dos tipos de decodificador, *hard* y *soft decision*, con el fin de evaluar el desempeño de ambos.

- **Diagrama de flujo del demapeador de símbolos 2- PAM.**

El demapeador de símbolos tiene como entrada el vector real X, que recibe del canal AWGN, y como salida un vector real R resultado de sumar la constante α a cada uno de los componentes del vector recibido X y dividir luego entre 2α para así obtener una versión ruidosa del vector codificado originalmente C. En la figura 3.5 se muestra el diagrama de flujo del demapeador de símbolos 2-PAM, donde:

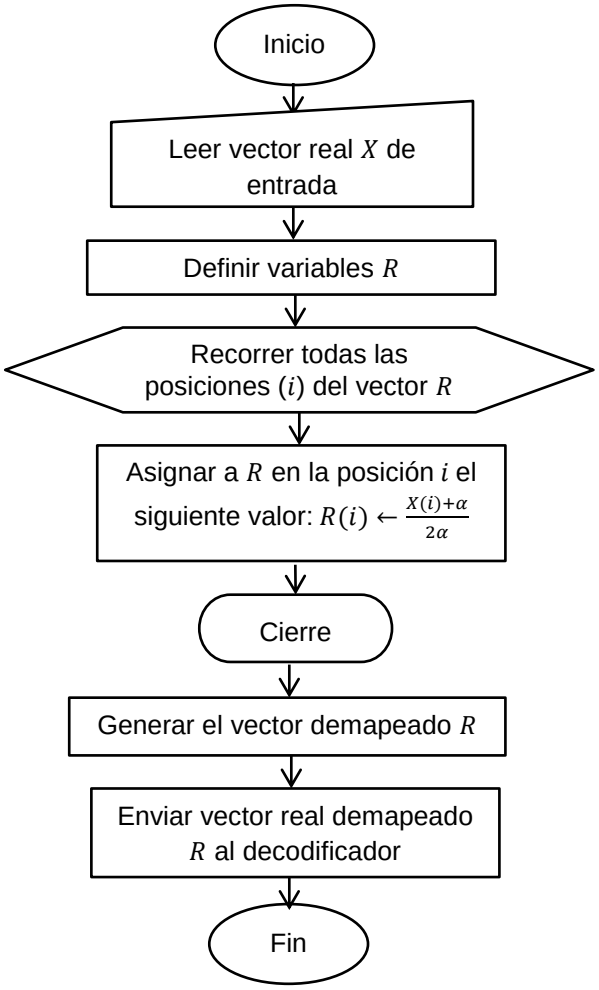


Figura 3.5 Diagrama de flujo del demapeador de símbolos 2-PAM.

X es el vector real a la entrada.

α es la constante definida en transmisión por el mapeador de símbolos 2-PAM.

R es el vector real a la salida del demapeador.

- **Diagrama de flujo del decodificador tipo *hard decision***

El decodificador *hard decision* tiene como entrada el vector de componentes reales R. Su función es encontrar la palabra codificada originalmente, mediante la corrección de errores propiciada por el método de decodificación por síndrome, para luego determinar la palabra mensaje enviada. En la figura 3.6 se muestra el diagrama de flujo del decodificador tipo *hard decision,* donde:

SL es la matriz Slepiana del código.

E es la matriz de patrones de error que surge de la matriz Slepiana SL.

$H_{RM(r,m)}$ es la matriz de verificación de paridad del código.

D es el vector binario resultado de discretizar el vector real recibido R.

$s\_recibido$ es el vector de síndrome del vector discretizado D.

s_error es el vector de síndrome del patrón de error E_i.

A es el alfabeto de palabras código conocido por la fuente y el destino.

C' es la palabra código corregida.

Me es la matriz de mensajes posibles conocida por la fuente y el destino.

M' es el mensaje decodificado.

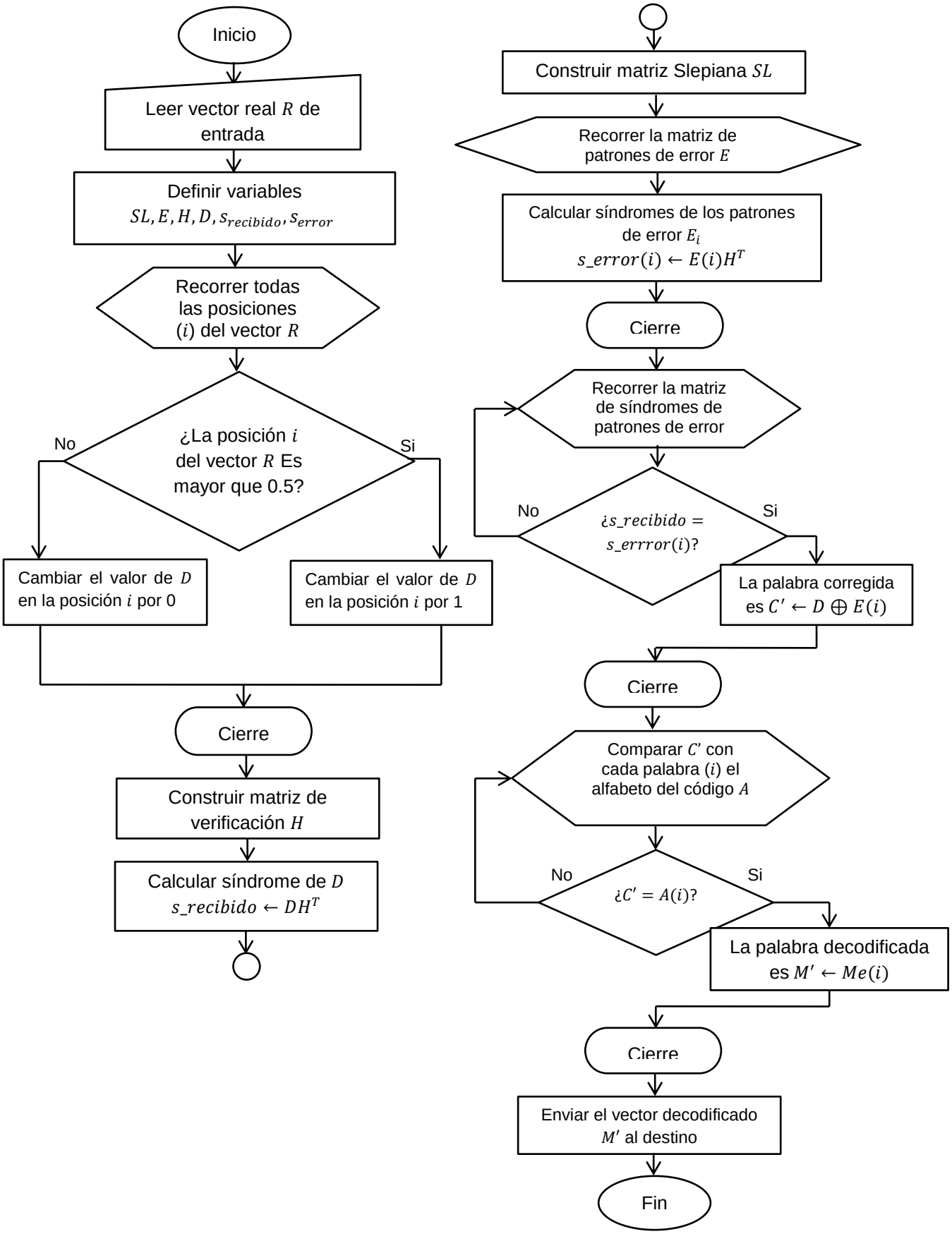


Figura 3.6 Diagrama de flujo del decodificador *hard decision*.

- **Diagrama de flujo del decodificador tipo *soft decision***

El decodificador tipo *soft decision* recibe el vector de números reales R. La decodificación se realiza mediante el cálculo de distancias Euclidianas entre el vector R y la imagen Euclidiana de cada palabra código del alfabeto. La palabra código con la que se obtiene la mínima distancia Euclidiana es la palabra corregida C', y con ésta es posible encontrar el mensaje enviado originalmente. En la figura 3.7 se muestra el diagrama de flujo del decodificador *soft decision,* donde:

d es el vector de distancias Euclidianas del vector R a la imagen Euclidiana de cada palabra código del alfabeto.

A es el alfabeto de palabras código conocido por la fuente y el destino.

A' es la imagen Euclidiana del alfabeto de palabras código.

C' es la palabra código corregida.

Me es la matriz de mensajes posibles conocida por la fuente y el destino.

M' es el mensaje decodificado.

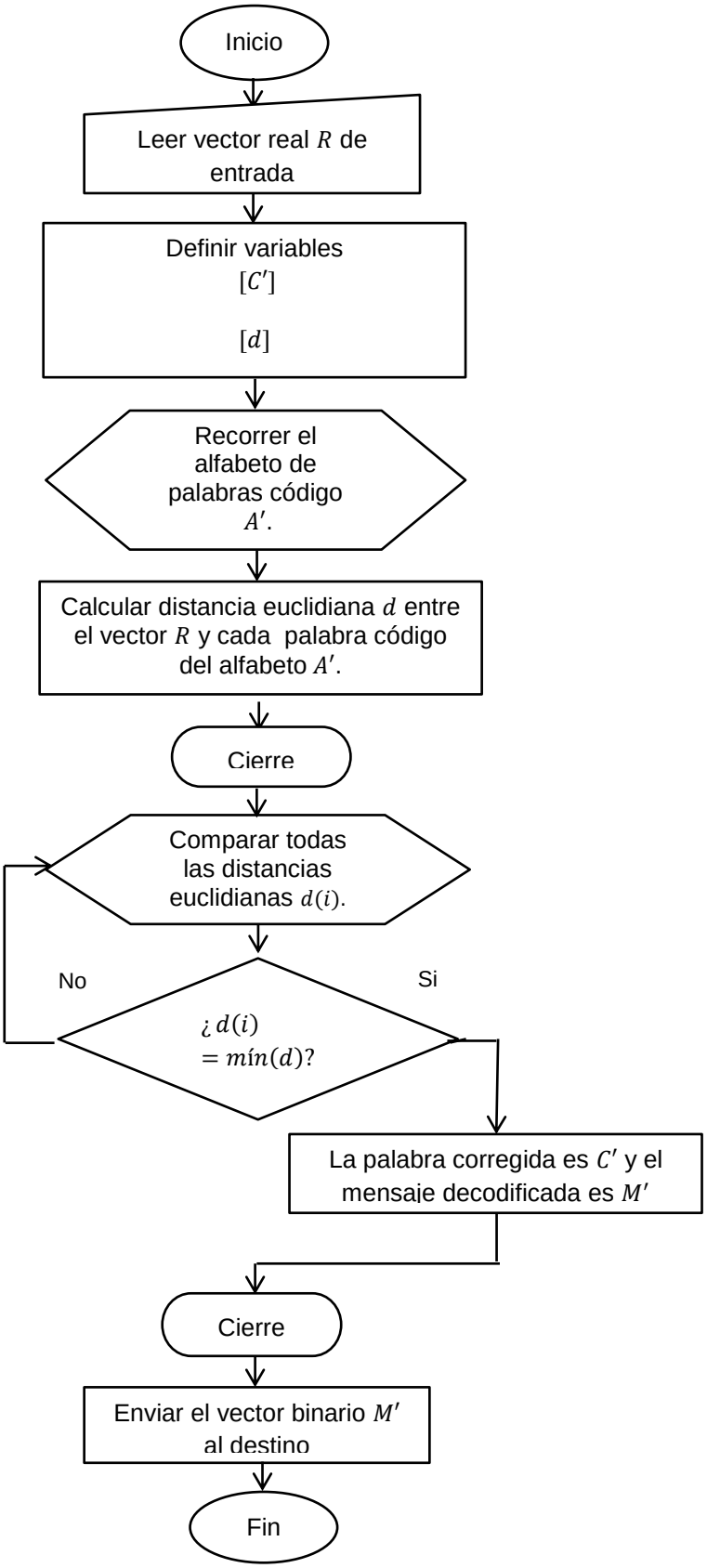


Figura 3.7 Diagrama de flujo del decodificador *soft decision*

3.2.4 Selección del Lenguaje

MATLAB R2010a ® es un lenguaje de alto nivel que integra distintas características que permiten explorar y visualizar ideas, así como colaborar en el procesamiento de señales, imágenes, comunicaciones, sistemas de control y finanzas computacionales. MATLAB cuenta con su propio lenguaje de programación (Lenguaje M) y está disponible para las plataformas Windows, UNIX y Apple Mac ® OS X. Algunas de las características más importantes, y que son de interés para el desarrollo del trabajo de grado, se presentan en la tabla 3.3.

Características	
Análisis Numérico	Sí
Cálculo de Matrices	Sí
Procesamiento de Señales	Sí
Procesamiento Gráfico 2D y 3D	Sí
Nivel de robustez	Alto
Programación modular	Sí
Soporte Canal AWGN	Sí
Soporte Aritmética de Galois	Sí
Soporte Procesamiento Estadístico	Sí
Soporte de Comunicaciones	Sí
Licencia	Propietaria
Aceptación Comunidad Académica	Alta
Soporte Comunidad Web	Alta

Tabla 3.3 Características de la herramienta de simulación MATLAB.

3.2.5 Codificación del Modelo

En esta fase se desarrollan dos códigos en el lenguaje M de MATLAB, integrando las funciones necesarias para simular los procesos de codificación y decodificación asociados a la codificación de canal RM, en un sistema de comunicación digital en banda base. Los códigos desarrollados se diferencian en el método de decodificación utilizado: 1) *hard decision* y 2) *soft decision*, esto con el fin de evaluar el desempeño de los mismos numéricamente y mediante gráficas de BER vs Eb/No.

3.2.6 Validación

En esta fase se comparan los resultados obtenidos en la simulación del sistema con las conjeturas teóricas soportadas en la literatura. La comparación se realiza mediante curvas de BER en función de la relación Eb/No. Dicha validación se muestra en el capítulo 4, sección 4.3.4.

3.2.7 Experimentación

La experimentación se realiza teniendo en cuenta los diferentes escenarios de simulación propuestos que permiten evaluar el desempeño de la codificación de canal RM y los algoritmos de decodificación *hard* y *soft decision,* ésta se presenta en el siguiente capítulo.

En este trabajo de grado se analiza el desempeño de la codificación RM frente al ruido AWGN en un ambiente de simulación, por lo tanto, según la metodología planteada en [21], solo se abarcan las primeras 7 fases. Las fases finales, correspondientes a la implementación en sistemas reales y monitoreo y control de los mismos, van más allá del alcance de los objetivos propuestos y es parte de los trabajos futuros.

4. Evaluación y Análisis de Resultados

4.1 Introducción

Este capítulo está enfocado en la evaluación y análisis de los resultados del desempeño, en cuanto a probabilidad de error, de los sistemas de comunicación en banda base con codificación RM y métodos de decodificación *hard* y *soft decisión*.

La simulación se desarrolló haciendo uso de la herramienta MATLAB, debido a que provee funciones nativas utilizadas en la aritmética de campos de Galois y a su gran capacidad de procesamiento de señales de comunicaciones.

Los escenarios de simulación propuestos son dos:

- Sistema con codificación RM y método de decodificación *hard decision*.
- Sistema con codificación RM y método de decodificación *soft decision*.

Para cada escenario propuesto, se variaron los parámetros de la codificación RM, r y m, de tal manera que se plantearon los siguientes casos de estudio:

Tabla 4.1 Casos de estudio.

Código	Algoritmo de Decodificación
RM(0,3)	*Hard decision*
	Soft decision
RM(1,3)	*Hard decision*
	Soft decision
RM(2,3)	*Hard decision*
	Soft decision
RM(2,4)	*Hard decision*
	Soft decision

La funcionalidad y características de cada uno de los códigos RM seleccionados para los diferentes casos de estudio, se describen con detalle en la sección 4.2. Teniendo en cuenta dichas características, se realizará el análisis de desempeño de los sistemas con codificación RM.

El plan de pruebas para la evaluación y análisis del desempeño de la codificación RM se realiza en cuatro etapas:

- Análisis del desempeño del sistema con codificación RM y método de decodificación *hard decision*.
- Análisis del desempeño del sistema con codificación RM y método de decodificación *soft decision*.
- Comparación entre los métodos de decodificación *hard decision* y *soft decision*.
- Validación de los sistemas con codificación RM y métodos de decodificación *hard* y *soft decision*.

Todas las pruebas se realizaron enviando 100000 secuencias binarias aleatorias de longitud k por ejecución. El objetivo de las pruebas es obtener varios valores de BER por cada valor de la relación Eb/No, para luego obtener una media muestral de los valores de BER obtenidos en cada iteración de la simulación. Es importante aclarar que dichos resultados constituyen una colección de variables aleatorias independientes e idénticamente distribuidas (iid), con distribución de probabilidad desconocida. Sin embargo, la media muestral de dichos resultados corresponde a una variable aleatoria con distribución de probabilidad aproximadamente Gaussiana[10]. Se varió la relación Eb/No desde 0 hasta 10 o 12 dB según fuera necesario, en pasos de 1dB, haciendo 10 ejecuciones por cada valor de Eb/No, con esto es posible obtener una curva aproximada para analizar el desempeño del sistema de comunicación.

En el anexo A se muestran las tablas de desempeño para cada uno de los esquemas de codificación evaluados. En ellas se muestran los valores de BER medidos y su correspondiente desviación estándar para cada valor de Eb/No, tanto para la técnica *hard decision* como *soft decision*.

[10] En razón al teorema del límite central, la distribución de probabilidad de la BER es aproximadamente Gaussiana.

4.2 Descripción de los Códigos Reed-Muller Seleccionados

Los códigos RM seleccionados, objeto de estudio de este trabajo de grado son 4, a saber, $RM(0,3)$, $RM(1,3)$, $RM(2,3)$ y $RM(2,4)$. Las características de estos códigos se describen a continuación:

- Código de Repetición: RM(0,3)

Este código es por definición un código bloque lineal $(8,1,8)$ correspondiente a un código de repetición de la forma $(n,1,n)$.

Los parámetros de $RM(0,3)$ son: $n = 8, k = 1, d = 8, h = 7, T = 0.125, t = 3$.

Este código consiste en repetir cada bit de información n veces, es decir, si se transmite un 1, el código de repetición $RM(0,3)$ envía "11111111". Puede recibirse el bit correcto siempre y cuando el número de errores en la palabra codificada no supere el valor de $t = 3$. Se puede referir a él como un código corrector de errores. Sin embargo, el código de repetición, en general, es un código ineficaz debido a que reduce la velocidad de transmisión en un factor de n (8 para el caso $RM(0,3)$);

La matriz generadora del código $RM(0,3)$ es:

$$G_{RM(0,3)} = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]_{1\times 8}$$

Los mensajes posibles a enviar son 0 ó 1, la tabla 6 muestra el alfabeto de palabras código de RM(0,3), el cual lo componen simplemente dos palabras código:

Tabla 4.2 Alfabeto de palabras código RM(0,3).

Mensajes posibles	Palabras Código
0	00000000
1	11111111

- Código Biortogonal: RM(1,3)

El código $RM(1,3)$ es un código bloque lineal $(8,4,4)$ que tiene las características de 3 tipos de códigos mencionados en la tabla 4 en el segundo capítulo: 1)

corresponde a un código biortogonal de la forma $RM(1,m)$, con una distancia mínima igual a la mitad de la longitud de palabra código, 2) es un código auto-dual, es decir su matriz generadora es igual a su matriz de verificación de paridad y 3) es un código de Hamming extendido, puesto que tiene un bit adicional de redundancia comparado con el código de Hamming[11] (7,4,3).

Los parámetros de $RM(1,3)$ son: $n = 8, k = 4, d = 4, h = 4, T = 0.5, t = 1$.

La matriz generadora de este código es:

$$G_{RM(1,3)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}_{4\times 8}$$

En la tabla 7 se muestra el alfabeto de palabras código del código RM(1,3). El número de mensajes posibles (tamaño del alfabeto de palabras código) es igual a $2^k = 16$.

Tabla 4.3 Alfabeto de palabras código RM(1,3).

Mensajes posibles	Palabras código
0000	00000000
0001	01010101
0010	00110011
0011	01100110
0100	00001111
0101	01011010
0110	00111100
0111	01101001
1000	11111111
1001	10101010
1010	11001100
1011	10011001
1100	11110000
1101	10100101
1110	11000011
1111	10010110

[11] En los datos codificados en Hamming se pueden detectar errores en un bit y corregirlos, sin embargo no se distingue entre errores de dos bits y de un bit (para lo que se usa Hamming extendido).

- Código de Verificación de Paridad: RM(2,3)

Este código corresponde a un código de verificación de paridad. Un código de verificación de paridad es un código bloque lineal, definido sobre $GF(2)$, cuyo funcionamiento consiste básicamente en agregar un bit adicional al mensaje para completar una palabra código, de tal manera que el número total de 1's en ella sea par. Así, en recepción, se puede detectar la presencia de errores simplemente calculando el peso de la palabra código. Según la definición de sus parámetros, este código no puede corregir errores.

Los parámetros de $RM(2,3)$ son: $n = 8, k = 7, d = 2, h = 1, T = 0.875, t = 0$.

La matriz generadora de este código es:

$$G_{RM(2,3)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{7\times 8}$$

Cada mensaje posible a enviar lo componen 7 bits. El número de mensajes posibles es 2^7. Por ende, el número de palabras código de 8 bits que componen el alfabeto es también 2^7.

- Código Hamming Extendido: RM(2,4)

Este código corresponde a otro código Hamming extendido con los siguientes parámetros: $n = 16, k = 11, d = 4, h = 5, T = 0.6875, t = 1$.

La tasa de codificación de $RM(2,4)$ es mayor la del código $RM(1,3)$, esto implica una mayor eficiencia espectral.

La matriz generadora de este código es:

$$G_{RM(2,4)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{11\times16}$$

Con este código se pueden enviar mensajes de 11 bits, es decir 2^{11} mensajes diferentes. Debido a que en el proceso de codificación se agregan 5 bits de redundancia, el alfabeto de $RM(2,4)$ está compuesto por 2^{11} palabras código de 16 bits cada una.

4.3 Análisis del Desempeño del Sistema de Comunicación con Codificación Reed-Muller

En las figuras 4.1 a 4.10 se presentan las curvas de desempeño de los sistemas con codificación RM y métodos de decodificación *hard decision* y *soft decision*, comparadas con la curva de desempeño del sistema sin codificación (2-PAM).

4.3.1 Análisis del Desempeño del Sistema de Comunicación con Método de Decodificación *Hard Decision*

- **RM(0,3)**

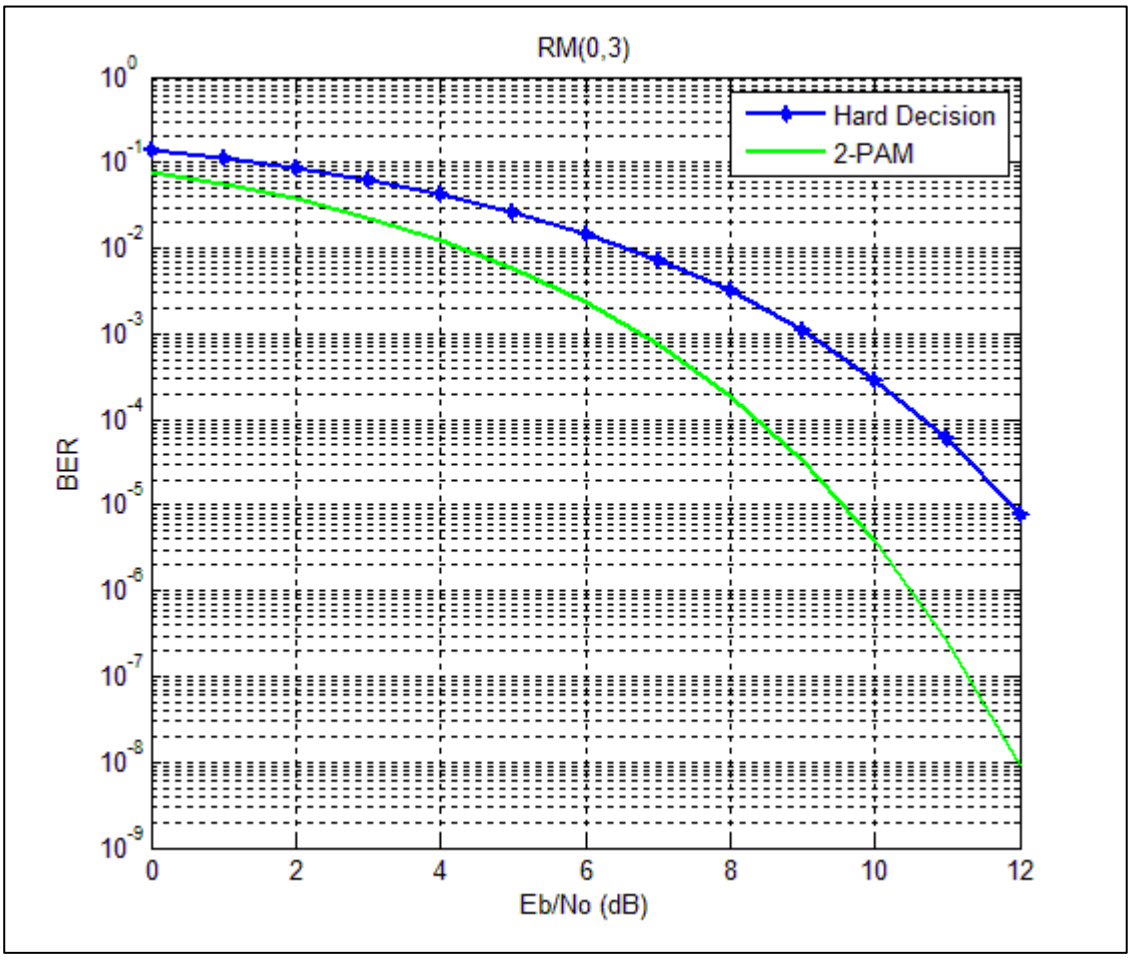


Figura 4.1 Curva de desempeño del código RM(0,3) - *Hard decision*.

La figura 4.1, muestra el desempeño para el sistema con codificación RM(0,3). Se observa que el sistema codificado demanda mayor cantidad de Eb/No para lograr la misma probabilidad de error de un sistema no codificado. Por ejemplo, para una

BER de $2x10^{-5}$, el esquema codificado requiere alrededor de 2.4 dB adicionales en Eb/No, es decir, un gasto de potencia de transmisión adicional del 73.79%.

- **RM(1,3)**

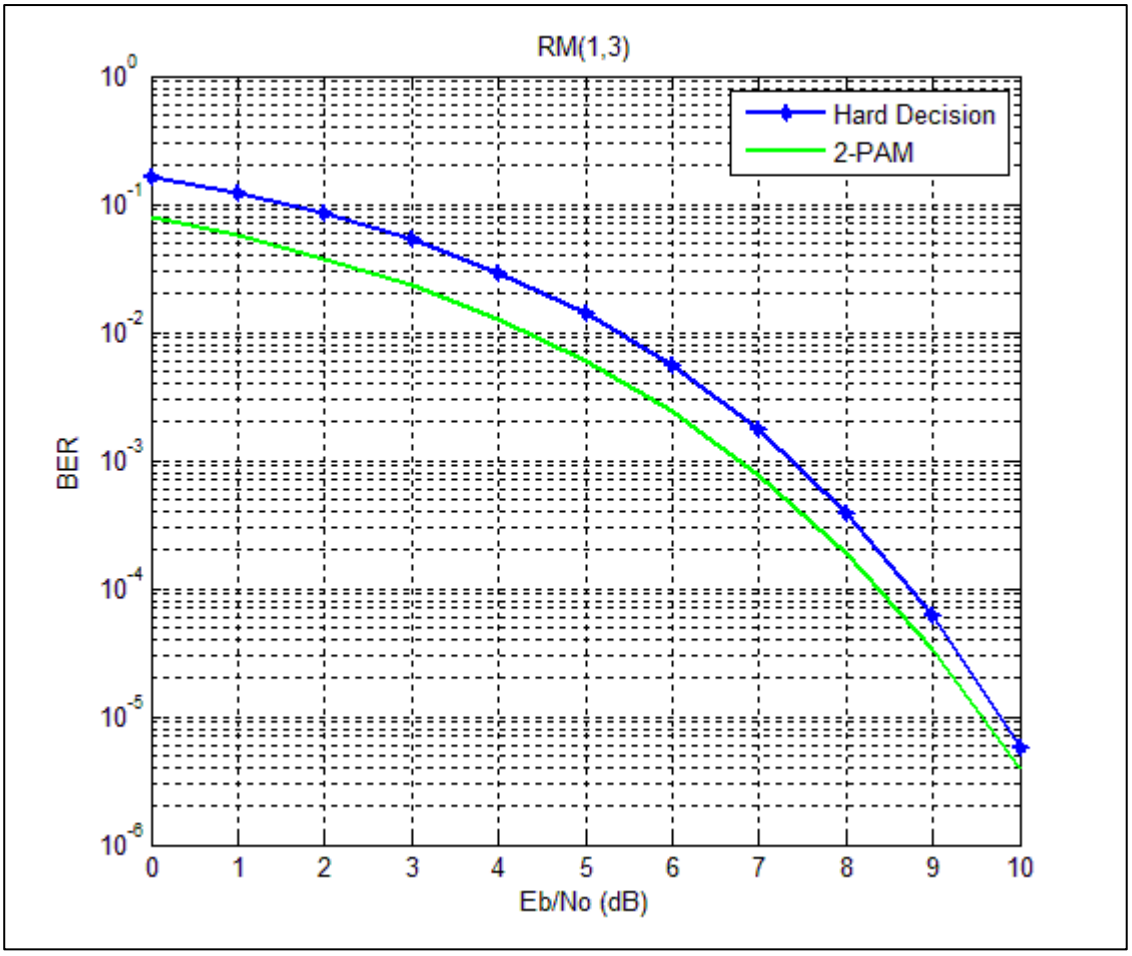


Figura 4.2 Curva de desempeño del código RM(1,3) - *Hard decision*.

La figura 4.2 muestra que la curva de desempeño del sistema con codificación RM(1,3) se aproxima a la curva correspondiente a 2-PAM a medida que la relación Eb/No aumenta. Por ejemplo, para una BER igual a $3*10^{-2}$ la diferencia en la relación Eb/No requerida es de aproximadamente 1.5 dB, y para una BER=10^{-4} la diferencia es aproximadamente 0.4 dB, lo que implica un aumento de potencia de transmisión aproximado de 41.6% y 9.7% en cada caso. Es notable, por la tendencia de la curva, que para valores mayores a 10 dB de la relación Eb/No, las curvas tendrán el mismo comportamiento asintótico.

- **RM(2,3)**

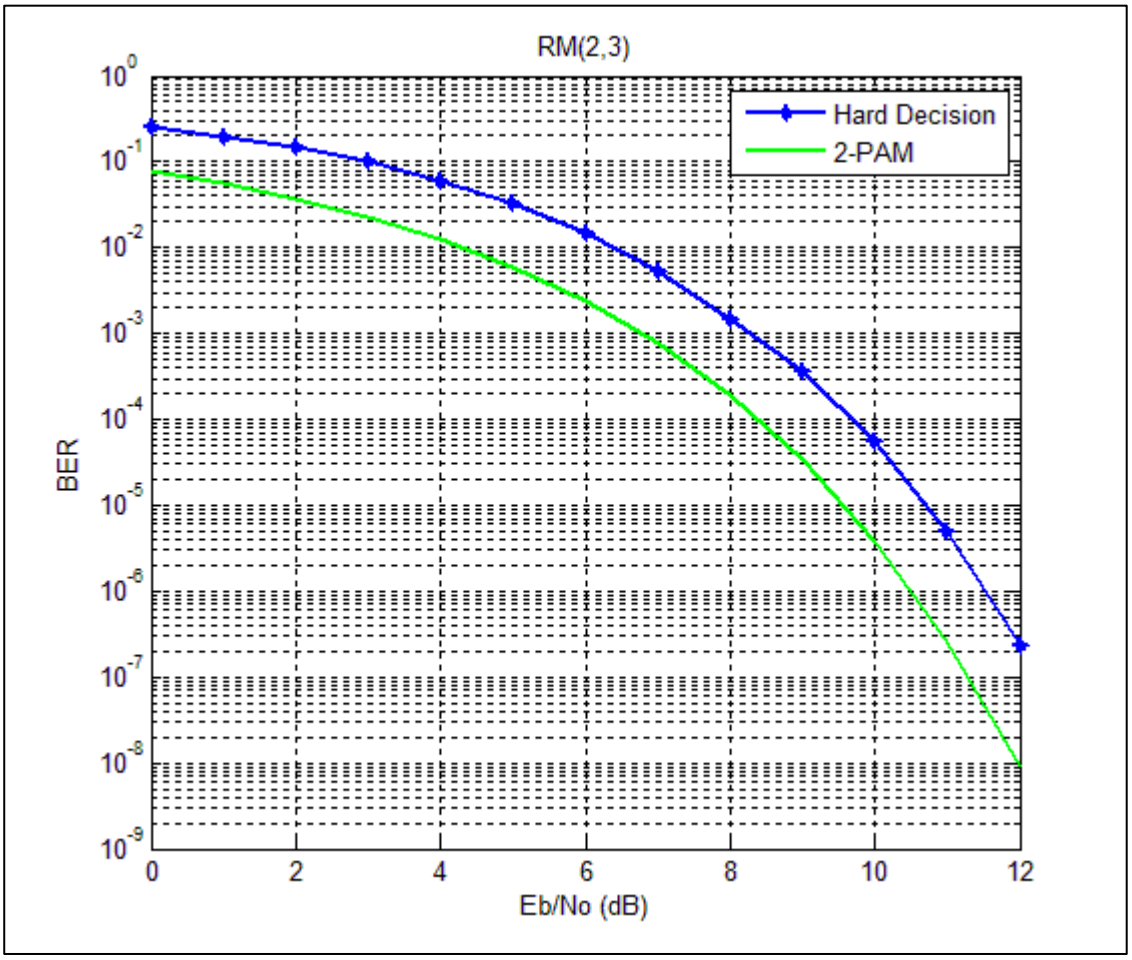


Figura 4.3 Curva de desempeño del código RM(2,3) - *Hard decision*.

La figura 4.3 corresponde a la curva de desempeño del sistema con codificación RM(2,3), la cual revela que su desempeño es inferior al de un sistema sin codificación. Esto se debe a que el código de verificación de paridad es un código detector mas no corrector de errores. Las palabras erróneas en el decodificador no se pueden recuperar, por lo tanto las Tasas de Error de Bit son más altas para una misma relación Eb/No.

Se observa que la diferencia entre las dos curvas para una BER= $2*10^{-3}$ en la relación Eb/No requerida es aproximadamente 1.8 dB y para una BER=$5*10^{-6}$ es 1.1 dB, esto implica un aumento en el gasto de potencia para el sistema con codificación del 51.3% y 28.7% respectivamente, comparado con el sistema sin codificación.

- **RM(2,4)**

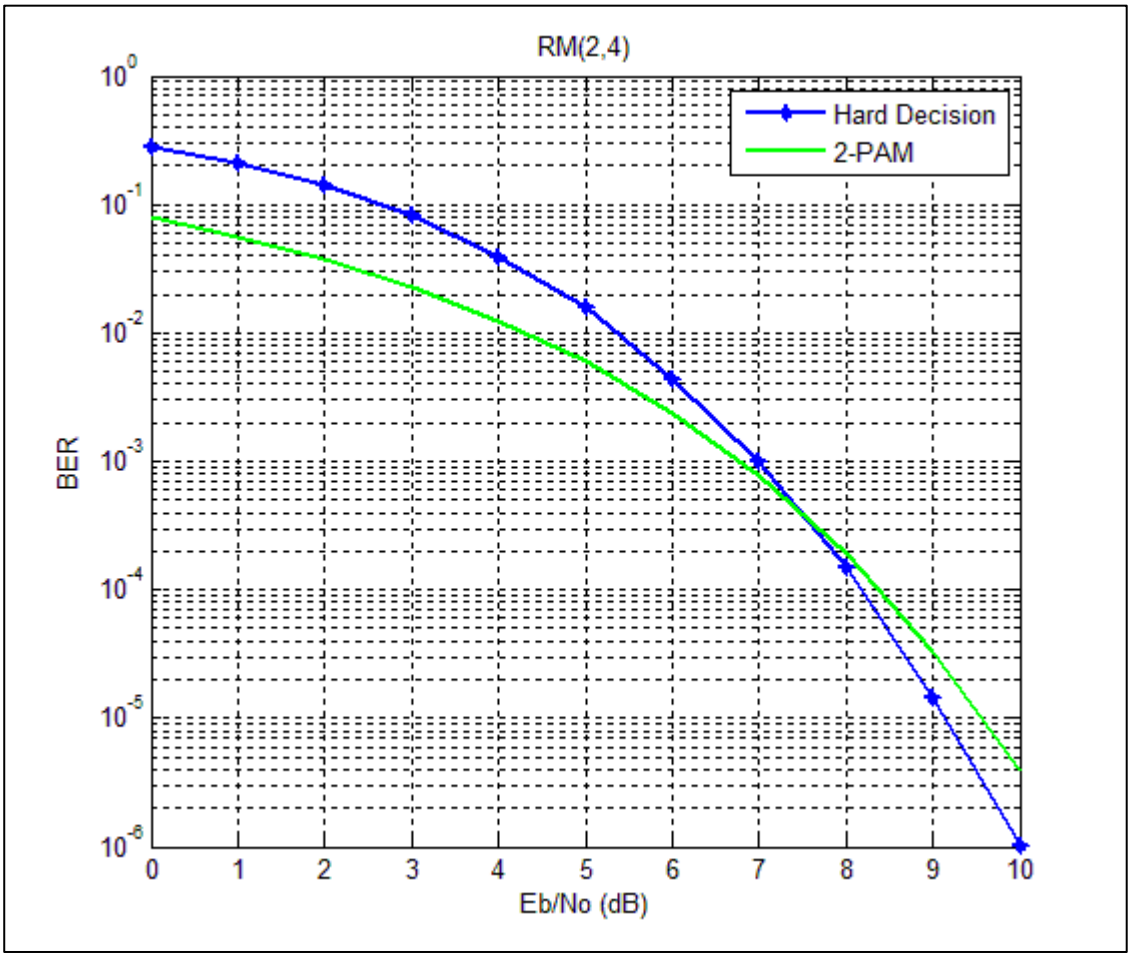


Figura 4.4 Curva de desempeño del código RM(2,4) - *Hard decision*.

Se puede apreciar en la figura 4.4, que para valores de relación Eb/No menores a 7.5 dB la curva del sistema con codificación RM(2,4) no presenta ninguna mejora en el desempeño comparada con la curva del sistema sin codificación. Sin embargo, para valores mayores a 7.5 dB de la relación Eb/No el código RM(2,4) presenta una ganancia de codificación, que se incrementa a medida que la relación Eb/No aumenta.

Por ejemplo, para alcanzar una Tasa de Error de Bit de $4*10^{-2}$ el sistema sin codificación necesita que la relación Eb/No sea cercana a 2 dB, mientras que el sistema codificado, requiere 4 dB. Es claro que en este punto la diferencia entre las curvas es aproximadamente 2 dB, lo que implica un aumento en el gasto de potencia para el sistema codificado de 58.8%. Esto sucede para Tasas de Error de Bit superiores a $4*10^{-4}$, donde la relación Eb/No equivale a 7.5 dB. En tanto que

para alcanzar una BER inferior a $4*10^{-4}$, el sistema con codificación RM presenta un mejor desempeño debido a su capacidad de corregir errores, por ejemplo, para una BER=$4*10^{-6}$ la ganancia de codificación es aproximadamente de 0.5 dB que significa un ahorro de potencia del 10.9%.

- **Comparación entre las Curvas de Desempeño de los Códigos Reed-Muller con Método de Decodificación *Hard Decision***

Para desarrollar un análisis detallado de los códigos bloque lineal se deben tener en cuenta tres parámetros fundamentales: 1) Eficiencia espectral o Tasa de codificación, 2) Distancia entre símbolos y 3) Número promedio de vecinos. Los valores particulares que tomen dichos parámetros hacen que un determinado código tenga un mejor comportamiento que otro.

En la figura 4.5 se observa que para valores pequeños de la relación Eb/No (entre 0 y 2 dB) donde la señal es más propensa a los efectos del ruido, el código RM(0,3) es el que tiene mejor comportamiento entre los códigos comparados, ya que posee la mayor distancia mínima Hamming, y la mayor capacidad de corrección entre los códigos evaluados, pues corrige hasta 3 bits, mientras que los códigos RM(1,3) y RM(2,4) únicamente corrigen un bit y el código RM(2,3) no es capaz de corregir. Para valores más grandes de la relación Eb/No, la capacidad de corrección de errores ya no es un parámetro relevante debido a que el efecto del ruido se reduce.

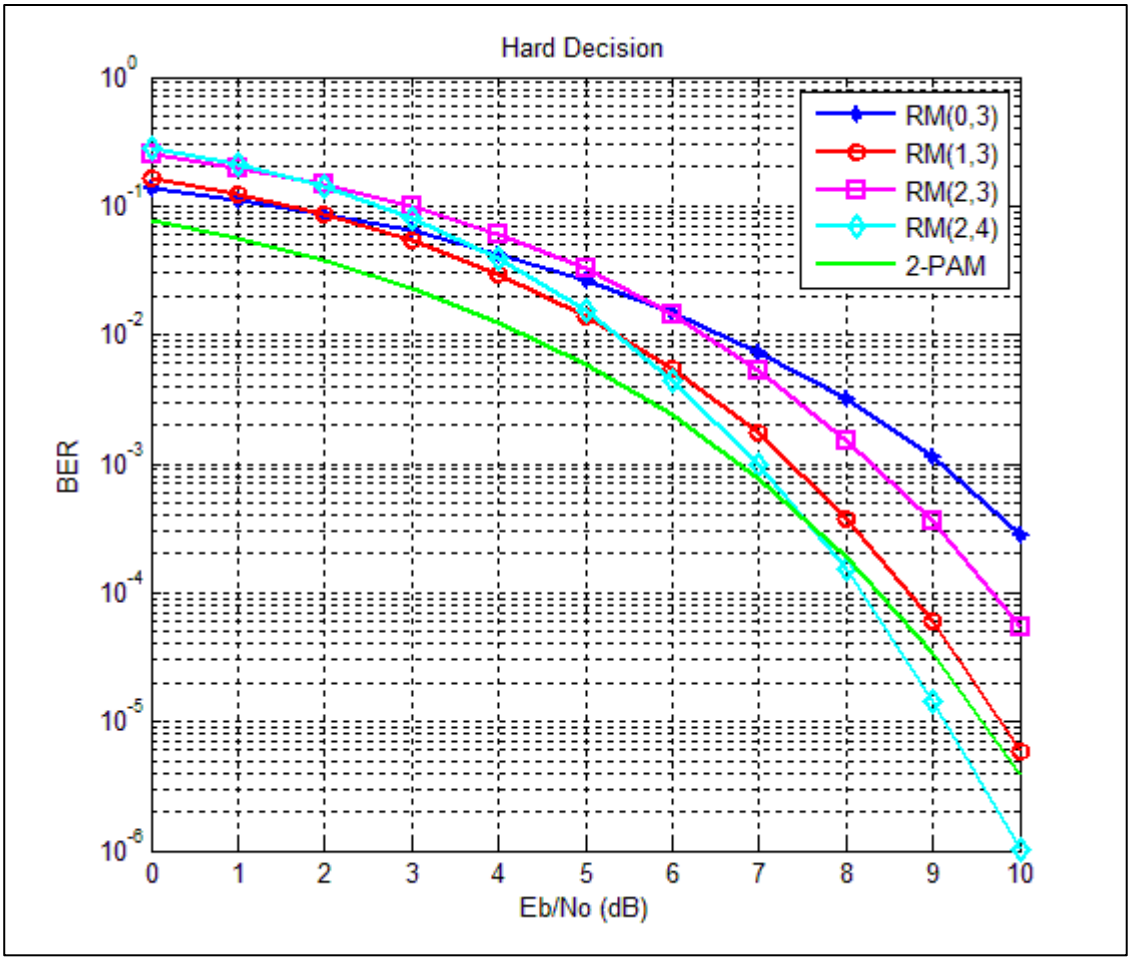


Figura 4.5 Comparación entre curvas de desempeño códigos RM - *Hard decision*.

Aunque RM(2,3) no corrige errores, característica que se ve reflejada en su bajo desempeño en los valores bajos de la relación Eb/No, su tasa de codificación es mayor que la tasa de codificación de los demás códigos, pero para valores superiores a 6 dB solamente supera en desempeño al código RM(0,3).

Es evidente que los códigos RM(0,3) y RM(2,3) no presentan ninguna mejora en cuanto a ganancia de codificación en comparación con el sistema sin codificación.

En general, los códigos evaluados de igual distancia mínima Hamming, RM(1,3) y RM(2,4), son los que mejor desempeño presentan entre los códigos RM evaluados y tienden a superar el sistema sin codificación, aunque RM(1,3) requiere una relación Eb/No aún mayor que RM(2,4) para lograrlo. Son capaces de corregir un error de bit por palabra código, y aunque RM(0,3) es capaz de corregir un mayor número de bits por palabra código, la tasa de codificación de los códigos RM(1,3)

y RM(2,4) es más alta. Lo anterior se ve reflejado a partir de los 4 dB de la relación Eb/No aproximadamente.

Para valores de 0 dB a 5 dB en la relación Eb/No, RM(1,3) presenta menores Tasas de Error de Bit que RM(2,4) debido a que el número promedio de vecinos es menor y por lo tanto existe una menor probabilidad de que los bits decodificados sean erróneos en valores pequeños de la relación Eb/No. El número promedio de vecinos es un parámetro crítico para RM(2,4) ya que por ser muchos, en el rango de relación Eb/No indicado, la decodificación tiende a presentar mayores Tasas de Error de Bit. Para valores de la relación Eb/No mayores a 5 dB, el código RM(2,4) evidencia un mejor comportamiento al transmitir mayor cantidad de bits de información por unidad de ancho de banda ya que su tasa de codificación es más alta.

Es importante destacar que para valores prácticos de la relación Eb/No, el único sistema que presenta mejor desempeño que el sistema sin codificación es el que utiliza el código RM(2,4). El valor de 7.5 dB constituye un umbral en el valor de relación Eb/No para el desempeño satisfactorio del código. A valores de relación Eb/No mayores a 7.5 dB, con la técnica de decodificación *hard decision*, es evidente que resulta favorable la codificación RM(2,4), ya que logra disminuir la Tasa de Error de Bit.

En la siguiente tabla están consignados los valores de la relación Eb/No requerida por los códigos RM evaluados, para alcanzar una Tasa de Error de Bit de 10^{-5}, y la ganancia de codificación que cada uno logra.

Tabla 4.4 Datos de desempeño de los códigos RM con algoritmo de decodificación *hard decision* para una BER de 10^{-5}.

Código RM	Eb/No para alcanzar 10^{-5} en BER (dB)	Ganancia de Codificación (dB)
(0,3)	11.5	-1.5
(1,3)	9.8	-0.3
(2,3)	10.5	-1
(2,4)	9.1	0.4

Los valores negativos registrados en la tabla anterior no representan en sí una “ganancia” sino una “pérdida” por codificación, lo que implica un gasto adicional de energía en relación al sistema sin codificación. Cabe anotar que la relación Eb/No

requerida por el sistema sin codificación para alcanzar una BER de 10^{-5} es de 9.5 dB.

4.3.2 Análisis del Desempeño del Sistema de Comunicación con Método de Decodificación *Soft Decision*

En la presente subsección se presentan las curvas de desempeño de los sistemas con codificación RM y método de decodificación *soft decision* y su análisis respectivo.

- **RM(0,3)**

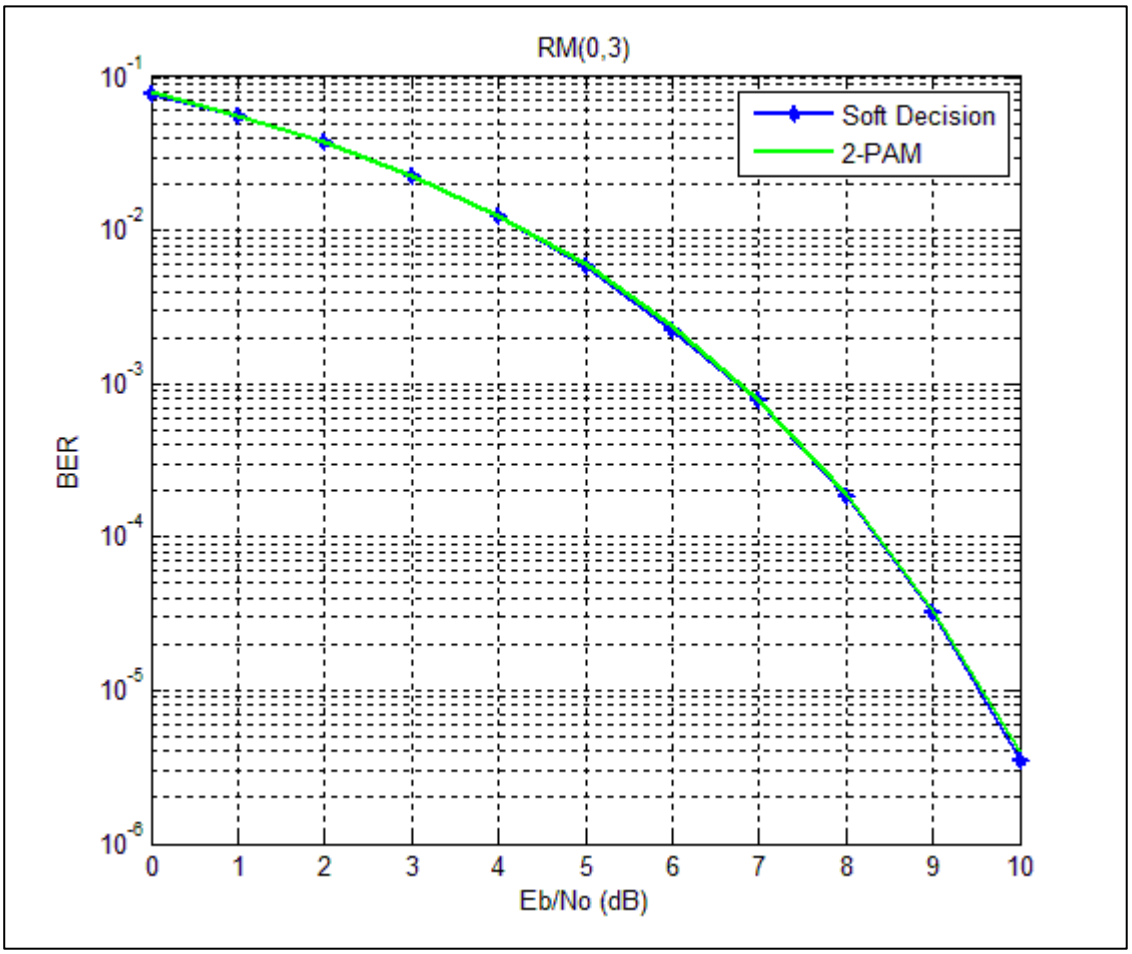


Figura 4.6 Curva de desempeño del código RM(0,3) - *Soft decision*.

En la figura 4.6 se aprecia que para todos los valores de la relación Eb/No, las curvas presentan el mismo comportamiento, por lo tanto la ganancia de codificación es 0. Aunque la distancia entre símbolos es mayor en el sistema con codificación, su eficiencia espectral es inferior por lo tanto estos parámetros se

compensan resultando un sistema con un comportamiento idéntico al sistema sin codificación (2-PAM).

- **RM(1,3)**

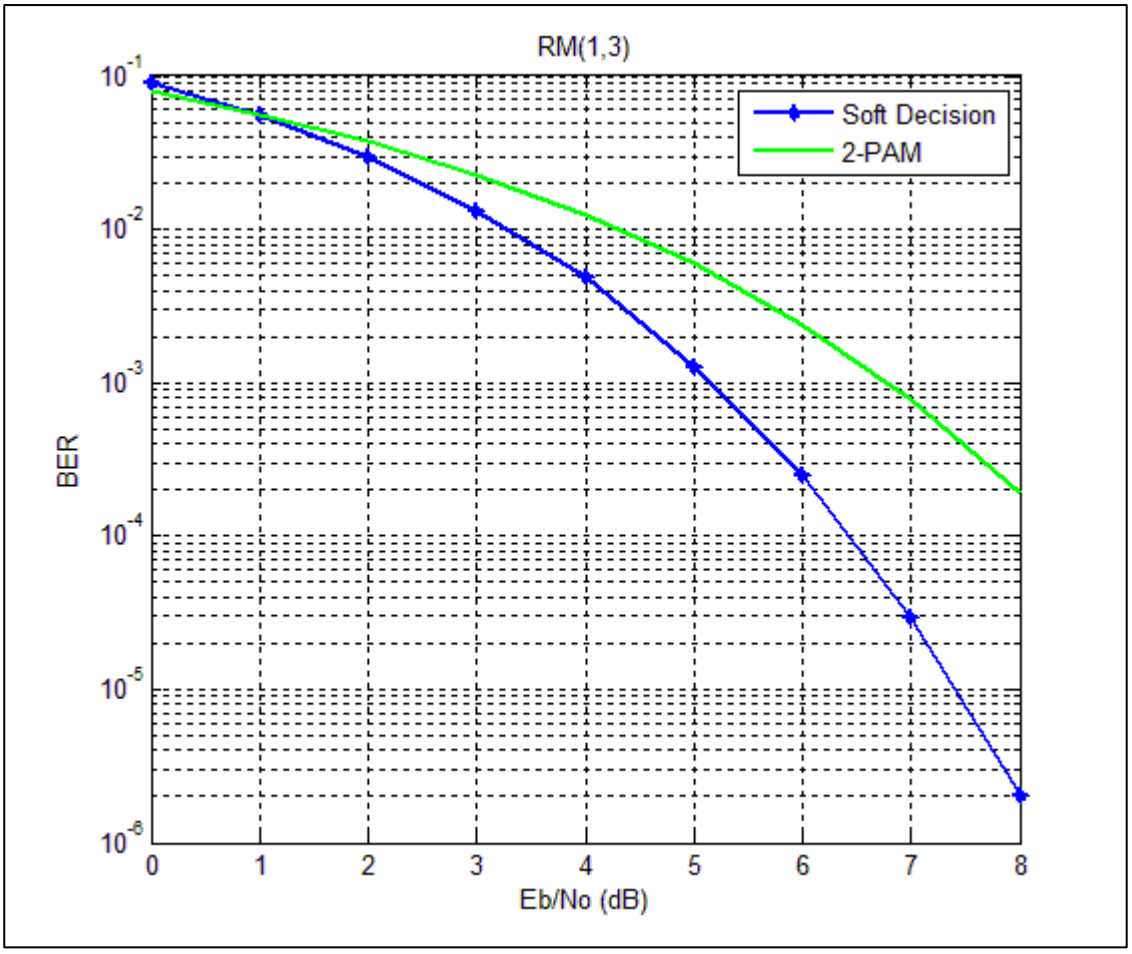


Figura 4.7 Curva de desempeño del código RM(1,3) - *Soft decision*.

La figura 4.7 muestra el desempeño del sistema de comunicación que utiliza codificación RM(1,3). Los resultados muestran claramente que, con el método de decodificación *soft decision*, este código presenta una ganancia de codificación considerable para cualquier valor de la relación Eb/No. Por ejemplo, para una BER de $1.5*10^{-2}$, la diferencia en la relación Eb/No requerida es de aproximadamente 1 dB, lo que implica un ahorro de potencia aproximado del 20.71%. La ganancia es aún más prominente para BER más bajas, por ejemplo para una BER= $2*10^{-4}$ la ganancia de codificación es de casi 2 dB, generando un ahorro de potencia aproximadamente de 36.9%.

- **RM(2,3)**

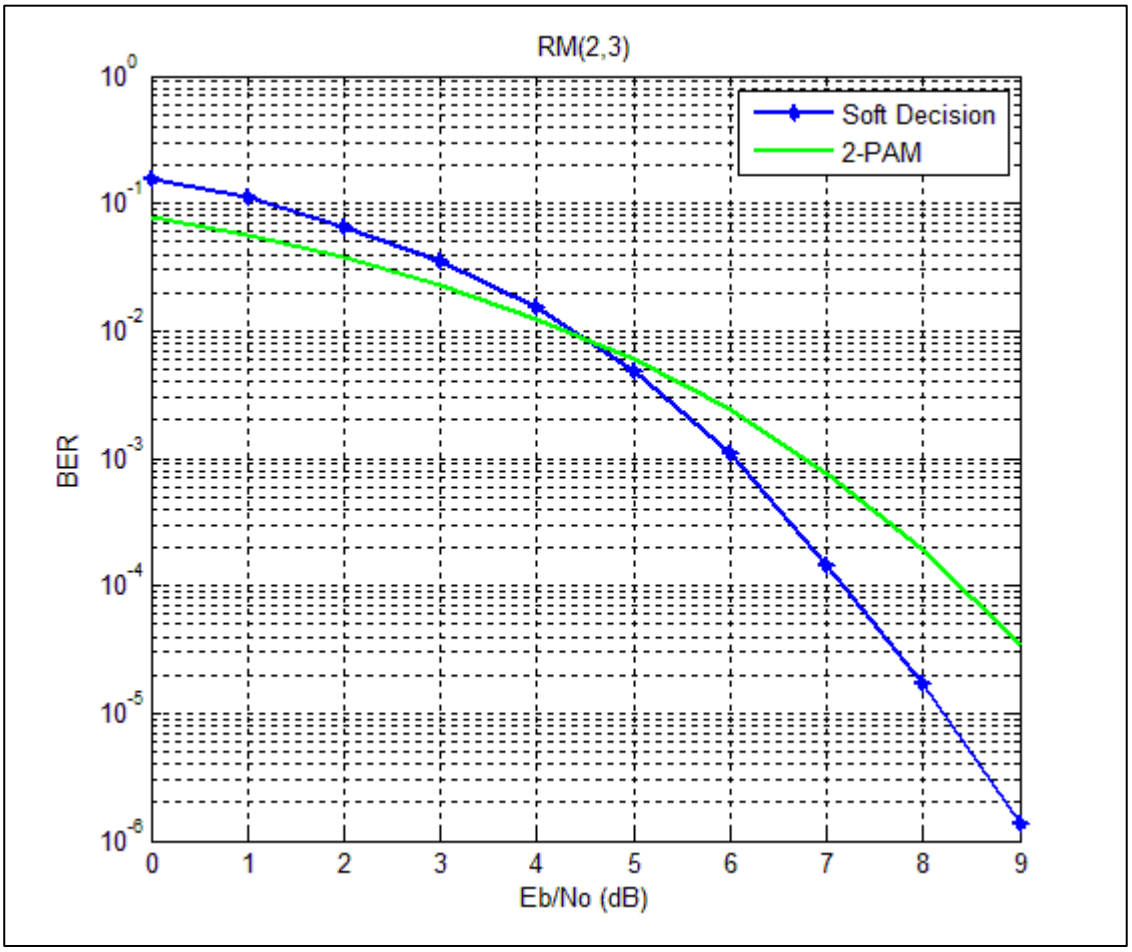


Figura 4.8 Curva de desempeño del código RM(2,3) - *Soft decision*.

En la figura 4.8 se observa el desempeño del sistema con codificación RM(2,3). Los resultados indican que para valores de BER de alrededor de $8*10^{-3}$ y mayores, el sistema con codificación presenta peor desempeño, lo que implica un mayor gasto de potencia para alcanzar una misma Tasa de Error de Bit que el sistema sin codificación, sin embargo es en ese punto donde el desempeño del sistema con codificación mejora con respecto a 2-PAM, el ahorro de potencia crece a medida que aumenta la relación Eb/No. Para lograr una BER equivalente a $6*10^{-2}$, el sistema sin codificación necesita una relación de 1 dB y el sistema con codificación requiere de una relación de 2.3 dB y para una BER de $2*10^{-4}$ el sistema sin codificación requiere de una relación de 8 dB y el sistema con codificación únicamente 6.8 dB. En el primer caso, se necesita 1.3 dB de más para alcanzar la misma BER que el sistema sin codificación, por lo tanto un gasto

adicional de potencia del 34.89% y en el segundo caso, se presenta un ahorro que equivale al 24.14%. El código con método de decodificación *soft decision* presenta un mejor comportamiento para valores de Eb/No comunes en sistemas de comunicaciones reales.

- **RM(2,4)**

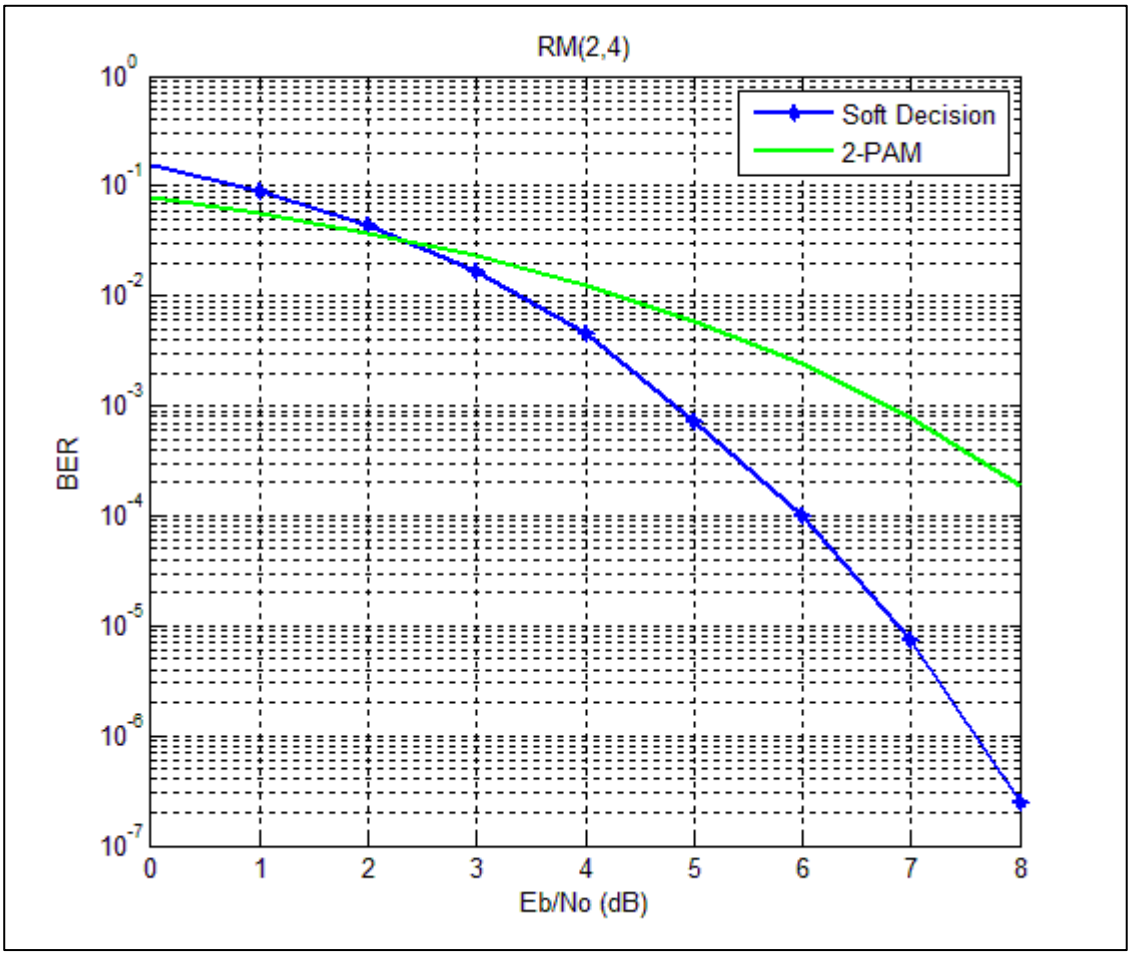


Figura 4.9 Curva de desempeño del código RM(2,4) - *Soft decision.*

La figura 4.9 muestra las curvas de desempeño del sistema con codificación RM(2,4). Los resultados muestran que la diferencia entre los dos sistemas para valores prácticos de la relación Eb/No es notable y cada vez se hace más grande a favor del código RM(2,4). Por ejemplo para una BER de $5*10^{-3}$ la ganancia de codificación es aproximadamente de 1.1 dB, lo que genera un ahorro de potencia del 22.3% y para una BER=$7*10^{-4}$ la ganancia es aproximadamente de 2 dB, lo que implica un ahorro de potencia de 36.9%. La BER más baja alcanzada por la codificación RM(2,4) es de $2.5*10^{-7}$ para lo cual requiere de 8 dB en la relación

Eb/No; para lograr esta BER, el sistema sin codificación requiere aproximadamente de 11 dB en la relación Eb/No (ver anexo A), lo que significa una diferencia de 3 dB y un ahorro de potencia del 50.15%.

- **Comparación entre las Curvas de Desempeño de los Códigos Reed-Muller con Método de Decodificación *Soft Decision***

La figura 4.10 presenta las curvas de desempeño de los sistemas con método de decodificación *soft decision*. Al igual que en el análisis realizado para el método de decodificación *hard decision*, se deben tener en cuenta los tres parámetros fundamentales mencionados: 1) Eficiencia espectral o Tasa de codificación, 2) Distancia entre símbolos y 3) Número promedio de vecinos.

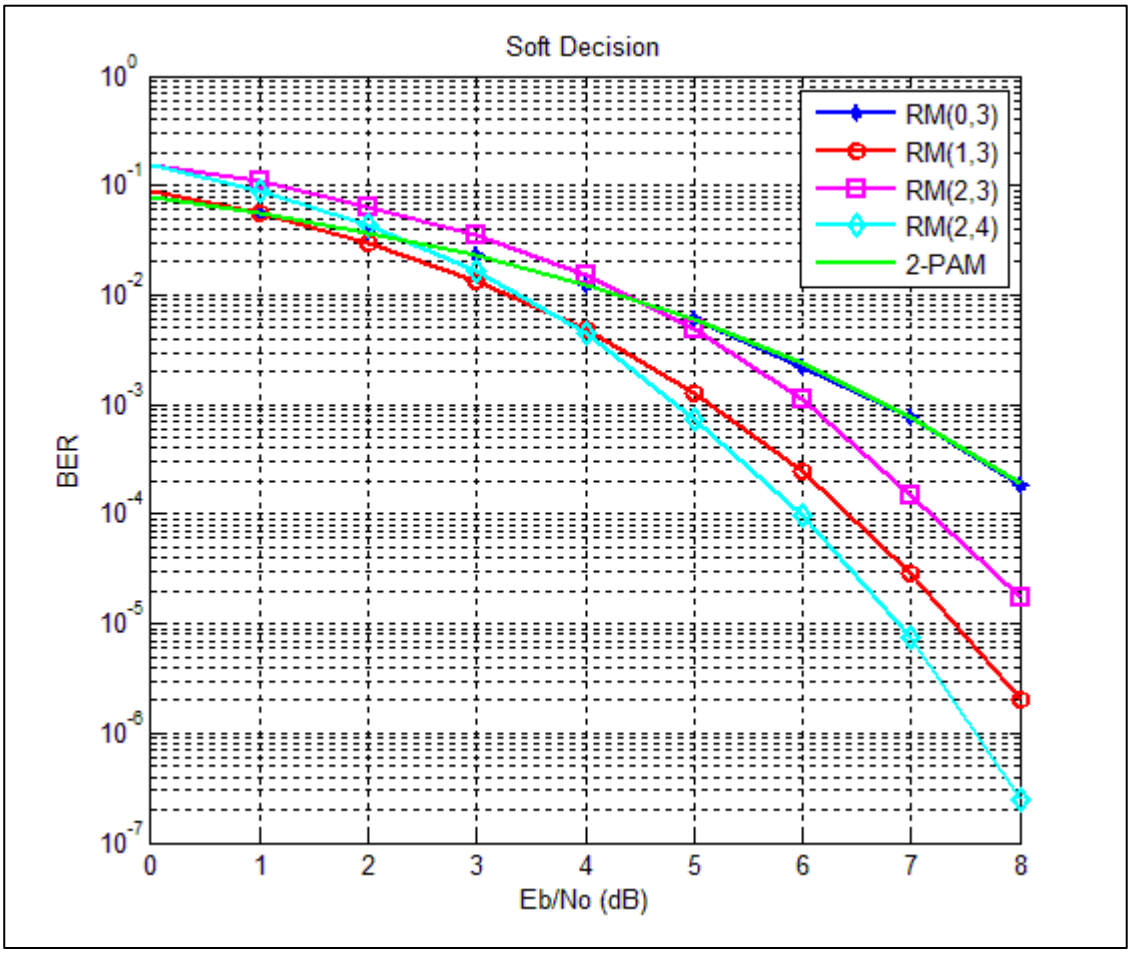


Figura 4.10 Comparación entre curvas de desempeño códigos RM - *Soft decision*.

Es evidente que para valores prácticos (mayores a 4.5 dB) de la relación Eb/No, todos los sistemas con codificación RM que utilizan el método de decodificación

soft decision mejoran el desempeño del sistema sin codificación, exceptuando el código RM(0,3). La tabla 4.5 muestra las ganancias de codificación de cada uno de los códigos y la relación Eb/No requerida para alcanzar una Tasa de Error de Bit de 10^{-5}.

Tabla 4.5 Datos de desempeño de los códigos RM con algoritmo de decodificación *soft decision* para una BER de 10^{-5}.

Código RM	Eb/No para alcanzar 10^{-5} en BER (dB)	Ganancia de Codificación (dB)
(0,3)	9.5	0
(1,3)	7.3	2.2
(2,3)	8.2	1.3
(2,4)	6.9	2.6

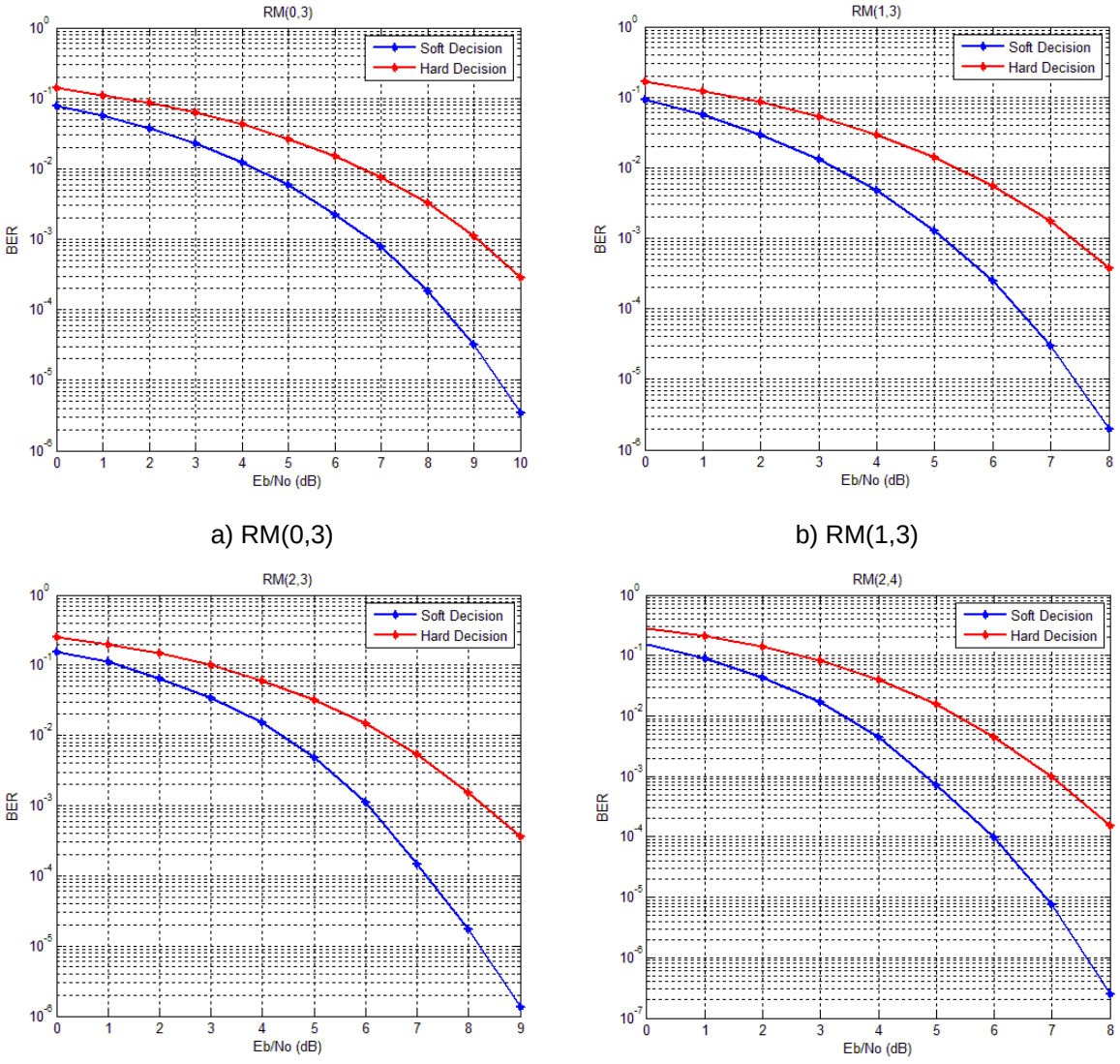
Se puede notar claramente que el comportamiento de los códigos RM mejora ampliamente al cambiar el método de decodificación, debido a que su desempeño no depende estrictamente de los valores de los parámetros (r, m), sino también de la técnica de decodificación utilizada. Como se observa en las gráficas, el desempeño del sistema con decodificación *soft decision* es superior a su contraparte *hard decision* debido, justamente, a que el proceso de decodificación *soft decision* se realiza teniendo en cuenta la mínima distancia Euclidiana como factor de decisión, tomando palabras código completas sin omitir información; mientras que el método *hard decision* realiza el proceso de decodificación bit a bit obligando a los mismos a relegarse a una región de decisión donde se ven truncados y/o redondeados[12] omitiendo así información relevante. Por lo tanto la BER para el sistema con decodificación *hard decision* es considerablemente mayor.

Para ratificar lo anteriormente expuesto, a continuación se muestra una comparación entre los métodos de decodificación *hard decision* y *soft decisión*.

[12] Un bit es truncado si su energía es inferior al umbral de decisión y es redondeado si por el contrario su energía es mayor.

4.3.3 Comparación entre los Métodos de Decodificación *Hard Decision* y *Soft Decision*

En la figura 4.11 se presentan la comparación entre las curvas de desempeño de los sistemas con decodificación *hard decision* y los sistemas con decodificación *soft decision*. Esto con el fin de observar la diferencia entre los dos métodos. En ella se observa que el método de decodificación *soft decision* presenta un mejor desempeño que su contraparte *hard decision.*



a) RM(0,3)

b) RM(1,3)

c) RM(2,3)

d) RM(2,4)

Figura 4.11 Comparación de curvas de desempeño entre *hard* y *soft decision.*

Se aprecia que para valores prácticos de la relación Eb/No, la diferencia entre los dos métodos de decodificación está alrededor de 2 dB; mostrando que el método con criterio de distancia mínima Euclidiana proporciona una ganancia considerable en el desempeño del sistema.

En tabla 4.6 se consignan los valores de la relación Eb/No requerida por el código RM(2,4) para diferentes valores de BER y la ganancia en dB que presenta el método de decodificación *soft decision* respecto a *hard decision*.

Tabla 4.6 Ganancia de codificación del método de decodificación *soft decision* vs *hard decision* para diferentes valores de BER.

Codificación	BER	Eb/No requerida *Hard decision* (dB)	Eb/No requerida *Soft decision* (dB)	Ganancia de Codificación (dB)
RM(0,3)	10^{-3}	9	6.9	2.1
	10^{-4}	10.7	8.4	2.3
	10^{-5}	11.9	9.5	2.4
RM(1,3)	10^{-3}	7.4	5.1	2.3
	10^{-4}	8.8	6.4	2.4
	10^{-5}	9.7	7.4	2.3
RM(2,3)	10^{-3}	8.3	6	1.7
	10^{-4}	9.7	7.1	2.6
	10^{-5}	10.7	8.2	2.5
RM(2,4)	10^{-3}	7	4.9	2.1
	10^{-4}	8.1	6	2.1
	10^{-5}	9.1	6.9	2.2

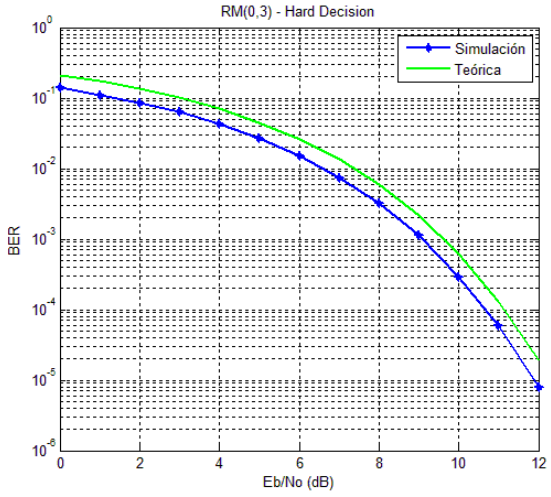
4.3.4 Validación de los Sistemas con Codificación RM y Métodos de Decodificación *Hard* y *Soft Decision*

Para validar el correcto funcionamiento de los códigos RM simulados en MATLAB, se tienen como referencia las curvas teóricas de desempeño BER vs Eb/No dadas por las ecuaciones indicadas en el capítulo 2 para los tipos de decodificación *hard decision* y *soft decision*, (2.31) y (2.33) respectivamente. En los apéndices A y B se muestra el desarrollo para la obtención de tales ecuaciones.

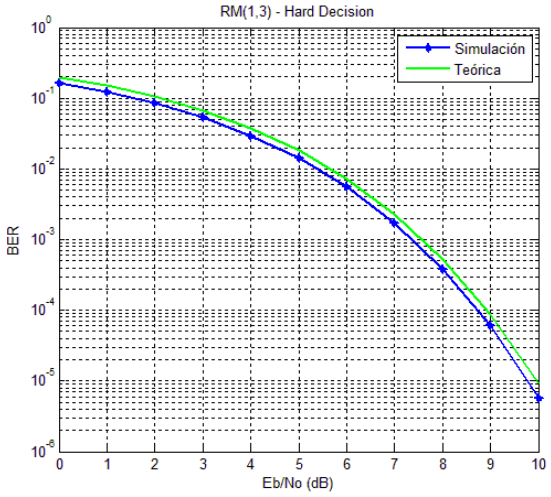
Una vez obtenidas las curvas teóricas, se trasponen con las curvas simuladas para realizar la comparación y validación de los resultados de cada uno de los códigos RM simulados. En las figuras 4.12 y 4.13 se muestra el desempeño

teórico vs simulación de los métodos de decodificación *hard* y *soft decision* respectivamente.

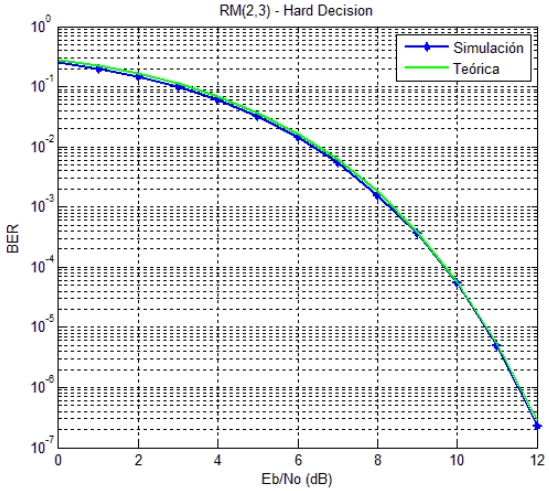
- **Sistemas de comunicación con decodificación *hard decision***



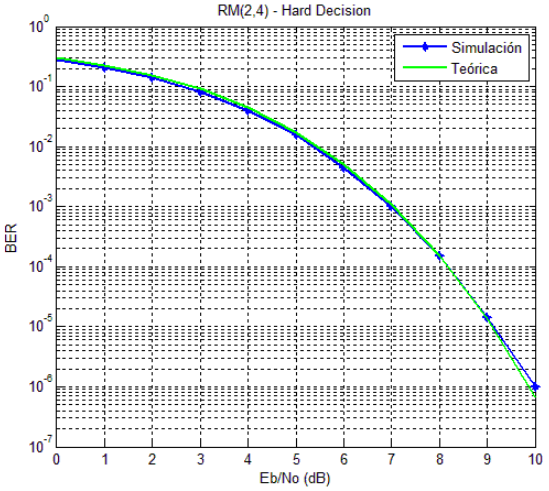
a) RM(0,3)



b) RM(1,3)



c) RM(2,3)

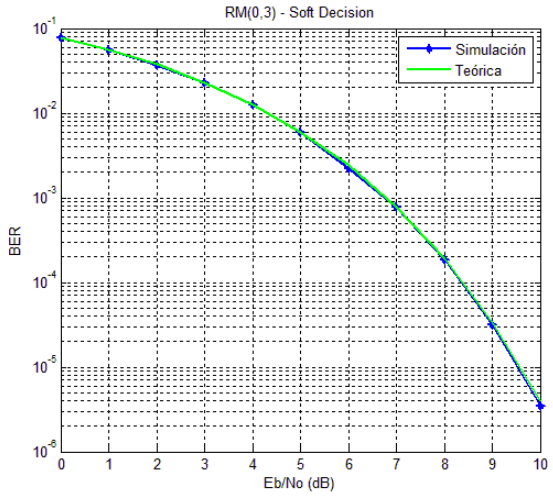


d) RM(2,4)

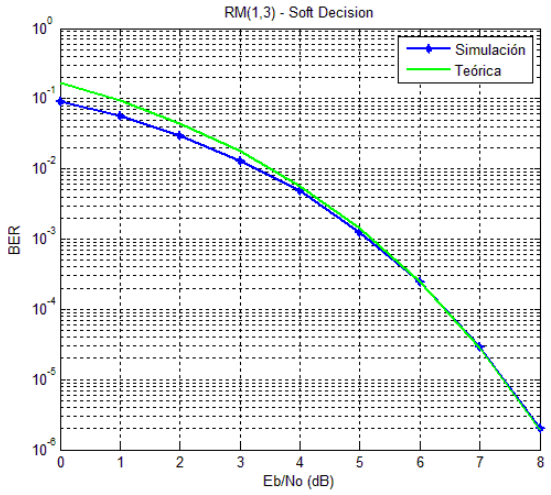
Figura 4.12 Desempeño teórico vs simulación - *Hard decision.*

En las gráficas a, b, c y d de la figura 4.12, se puede observar que los valores obtenidos en las simulaciones realizadas en MATLAB y los valores proporcionados por las ecuaciones de probabilidad de error de bit son muy aproximados, tanto así que las curvas tienden a traslaparse.

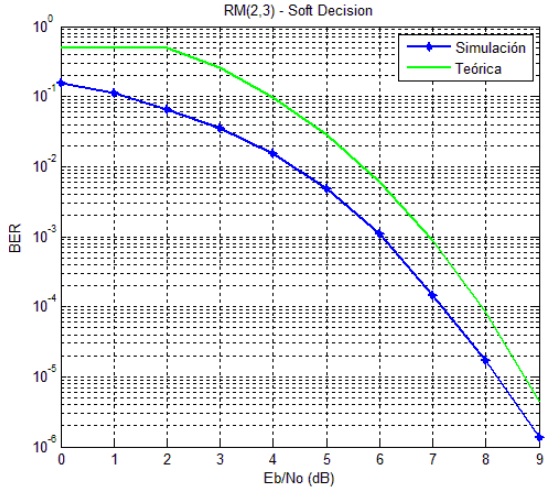
- **Sistemas de comunicación con decodificación *soft decision***



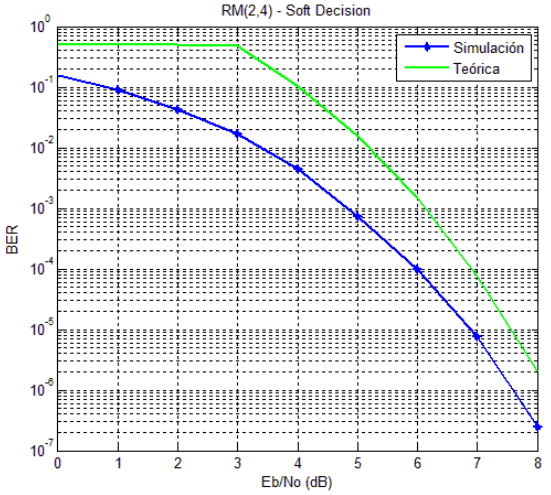
a) RM(0,3)



b) RM(1,3)



c) RM(2,3)



d) RM(2,4)

Figura 4.13 Desempeño teórico vs simulación - *Soft decision*.

En las gráficas a y b de la figura 4.13, se observa que las curvas de desempeño obtenidas en las simulaciones, también son muy aproximados a las curvas teóricas.

En las gráficas c y d de la figura 4.13, se observa que las curvas de desempeño de la simulación tienen un mejor comportamiento que las curvas teóricas, esto se explica con detalles a continuación:

Se observa que la curva teórica muestra una Tasa de Error de Bit igual a 0,5 para los valores 0, 1, y 2 dB de la relación Eb/No, ya que para estos valores, la

ecuación de referencia da como resultado probabilidades de error mayores a 1. Estos resultados son imposibles de alcanzar en simulación porque la medición de Tasa de Error de Bit se realiza dividiendo la cantidad de bits erróneos recibidos entre la cantidad total de bits enviados por tanto es un número entre 0 y 1. Cabe notar que una probabilidad de error de 0,5 indica que se está presentando el peor caso esperado.

Con lo anterior se justifica y se corrobora entonces el correcto funcionamiento de los sistemas de comunicación con codificación RM.

Los valores de BER simulados y teóricos a partir de los cuales se obtienen las curvas de desempeño de las Figuras 4.12 y 4.13, están consignados en el Anexo A.

4.3.5 Complejidad Computacional de los Sistemas de Comunicación Simulados

Esta sección presenta la complejidad computacional[13] de cada uno de los códigos RM evaluados, al utilizar los algoritmos *hard* y *soft decision* como método de decodificación.

La complejidad computacional de los sistemas de comunicación simulados depende del tiempo de simulación de los mismos. En la tabla 4.7 se consignan los tiempos registrados por la herramienta MATLAB en los casos de estudio de cada escenario. Los resultados de esta comprobación se obtuvieron realizando la transmisión y posterior recepción de 100000 mensajes de longitud k en una sola ejecución.

Tabla 4.7 Tiempos de simulación por iteración en código .m.

Código RM	Tiempo de Simulación *Hard Decision* (s)	Tiempo de Simulación *Soft Decision* (s)
(0,3)	116,26	42,81
(1,3)	90,96	133,08
(2,3)	88,53	843,45
(2,4)	425,67	27010,509

[13] La complejidad computacional estudia la eficiencia de los algoritmos estableciendo su efectividad de acuerdo al tiempo de corrida y al espacio requerido en la computadora o almacenamiento de datos, ayudando a evaluar la viabilidad de la implementación práctica en tiempo y costo.

Se observa en la tabla 4.7 que, para los códigos RM(1,3), RM(2,3) y RM(2,4), los tiempos de simulación son mayores con el método de decodificación *soft decision*; por lo tanto, éste método tiene una mayor complejidad computacional en comparación con *hard decision*. Con el código RM(0,3) ocurre lo contrario, el tiempo de simulación con el método *soft decision* es menor que con el método *hard decision*, esto se debe a que el decodificador de tipo *soft decision* únicamente necesita realizar el cálculo de 2 distancias Euclidianas ya que las palabras código del alfabeto del código de repetición son solamente 2; mientras que el decodificador *hard*, por utilizar la técnica de síndrome, necesita realizar una mayor cantidad de operaciones.

5. Conclusiones y Trabajos Futuros

En este capítulo se presentan las conclusiones y algunas alternativas para trabajos futuros resultado de la realización del trabajo de grado: Análisis del Desempeño de la Codificación Reed-Muller en un Canal con Ruido Blanco Aditivo Gaussiano.

5.1 Conclusiones

1. El comportamiento de los sistemas de comunicación con codificación RM simulados en la herramienta MATLAB, es válido debido a que los resultados obtenidos son muy similares a las conjeturas teóricas.

2. A nivel de simulación los sistemas de comunicación con codificación RM(0,3), RM(1,3), RM(2,3) y RM(2,4) presentan diferencias en complejidad computacional evidenciadas en los tiempos de simulación, ya que cada uno posee unas características específicas en términos de la longitud del mensaje y de palabra código, las cuales determinan si los procesos de codificación y decodificación tendrán mayor o menor complejidad.

3. Los algoritmos *hard decision* y *soft decision*, como paso de decodificación, para los códigos RM en la detección y corrección de errores, son sencillos de implementar, pero resultan poco eficientes cuando los códigos RM son demasiado grandes.

4. El decodificador *soft decision* presenta ganancia de codificación de aproximadamente 2 dB respecto al decodificador *hard decision*, en función del parámetro BER, tal ganancia se logra a costa de mayor complejidad computacional.

5. Según los resultados obtenidos en el trabajo, fue posible comprobar teóricamente y en simulación que la distancia minima Hamming y la distancia mínima Euclidiana están estrechamente relacionadas entre sí por un factor constante. Esto se puede corroborar en el apéndice B.

6. La distancia mínima entre palabras código cumple una función importante en la implementación de sistemas con codificación, ya que influye directamente en el desempeño dependiendo del algoritmo de decodificación utilizado. Esta es una característica favorable de los códigos RM.

7. De acuerdo a los parámetros que caracterizan la codificación RM, especialmente la tasa de codificación, se logró establecer que para alcanzar Tasas de Error de Bit (BER) menores a $4*10^{-3}$ utilizando el método de decodificación *soft decision*, el código con mejor desempeño es el código RM(2,4).

8. Utilizando el método de decodificación *hard decision*, el código RM(2,4) es el que presenta mejor desempeño entre los códigos evaluados, para Tasas de Error de Bit (BER) menores a $7*10^{-3}$, es capaz de superar al sistema sin codificación.

5.2 Trabajos Futuros

1. Analizar el desempeño de la codificación Reed-Muller en un canal con ruido blanco aditivo gaussiano utilizando algún tipo de algoritmo de decodificación híbrido, es decir que posea características de *hard decision* y *soft decision* simultáneamente, y comparar los resultados con los obtenidos en el presente trabajo de grado.

2. Diseñar un decodificador para códigos Reed-Muller utilizando el concepto de memorias asociativas como el planteado por Ionescu, Anton, Tutanescu, Mazare y Serban en [20], con el fin de disminuir la exigencia computacional y reducir los tiempos de respuesta prolongados que presenta la decodificación con métodos clásicos.

3. Considerar el esquema de codificación Reed-Muller presentado aquí para la implementación de un sistema de comunicación digital en tiempo real con los algoritmos de decodificación *hard decision* y *soft decision*.

4. Evaluar el desempeño de un sistema de comunicación que utilice el algoritmo de decodificación *hard decision* e implemente un protocolo de retransmisión

automática con la codificación Reed-Muller, basado en hardware, y comparar los resultados con los obtenidos en el presente trabajo de grado.

Referencias Bibliográficas

[1] J. G. Proakis and M. Salehi, *Communication Systems Engineering*, 2nd ed. New Jersey: Tom Robbins - Prentice Hall International Editions, 2002.

[2] S. Lin and D. J. Costello, *Error Control Coding*, 2nd ed. Prentice Hall, 2004.

[3] R. S. Pressman, *Ingeniería del Software, un enfoque práctico*, 7a ed. México DF, México: McGraw Hill, 2010, pp. 33–34.

[4] S. Haykin, *Communication Systems*. New York: Wiley, 2002.

[5] B. Carlson, P. Crillly, and J. Rutledge, *Communication Systems, An Introduction To Signals And Noise In Electrical Communication*, 4a ed. New York: Editorial Mc Graw Hill, 2002.

[6] T. K. Moon, *Error Correction Coding, Mathematical Methods and Algorithms*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.

[7] A. Kacewicz and S. Wicker, “Application of Reed-Muller Codes for Localization of Malicious Nodes,” 2010.

[8] C. T. Gueye, E. L. Hadji, M. Mboup, D. Mathématiques, U. Cheikh, and A. Diop, “Secure Cryptographic Scheme based on Modified Reed Muller Codes,” vol. 7, no. 3, pp. 55–64, 2013.

[9] J. H. van Lint, *Introduction to Coding Theory*, 2nd ed. Springer-Verlag, 1982, p. 47.

[10] C. Argyrides, P. Reviriego, C. Kokkinos, and J. A. Maestro, “Enhanced Decoding of Triple Error Correction Reed-Muller Codes to Reduce Silent Data Corruption in Memories,” no. Mld.

[11] I. S. Reed, “A class of multiple-error-correcting codes and the decoding scheme,” *IEEE Trans. Inf. Theory*, vol. 4, pp. 38–49, 1954.

[12] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Prentice Hall, 1995.

[13] H. C. Dzul and D. Javier, “La forma normal algebraica de una función booleana,” vol. 48, pp. 47–57, 2009.

[14] S. Raaphorst, “Reed-Muller Codes,” *Carlet. Univ.*, 2003.

[15] O. O. Khalifa, A.-H. Abdullah, N. Suriyana, S. Zawanah, and S. A. Hameed, “Reed-Muller Codec Simulation Performance,” *Comput. Sci.*, vol. 4, no. 10, pp. 792–798, 2008.

[16] R. Gallager, “Course materials for 6.450 Principles of Digital Communications I,” *MIT OpenCourseWare*, pp. 278–279, 2006.

[17] I. Dumer and R. Krichevskiy, “Soft-Decision Majority Decoding of Reed–Muller Codes,” vol. 46, no. 1, pp. 258–264, 2000.

[18] J. Bracho, “Capítulo I Plano Euclidiano,” in *Introducción Análitica a la Geometría*, 2003, p. 70.

[19] L. M. Tomás, “Conceptos Básicos sobre Modulación y Codificación de Canal en Comunicaciones Móviles,” pp. 1–9, 2008.

[20] J. M. Martínez, “Transmisión de Datos Codificación de Canal,” no. 2, pp. 2011–2012, 2012.

[21] E. Astaiza, H. Bermudez, and P. Muñoz, *Simulación de Sistemas de Telecomunicaciones*. Popayán: Padilla Bejarano, 2007.

[22] T. Noguchi, Y. Daido, and J. Nossek, “Modulation Techniques For Microwave Digital Radio,” *IEEE Commun. Mag.*, vol. 24, no.10, pp. 21–30, 1986.

[23] L. Ionescu, C. Anton, I. Tutanescu, A. Mazare, and G. Serban, “Reed-Muller Decoder with Associative Memories,” pp. 11–14, 2011.

[24] D. Forney, *Performance of Small Signal Sets*. Massachusetts: M.I.T. Open Courseware, pp. 43–57.

[25] D. Lapedes, *McGraw-Hill Dictionary of Scientific and Technical Terms*, 2da ed. New York: McGraw-Hill Book Co, 1978.

Apéndices

Apéndice A. Probabilidad de Error con Decodificación *Hard Decision*

La probabilidad de error en la detección de una palabra código se lleva a cabo asumiendo que los n bits que conforman dicha palabra corresponden a n ensayos Bernoulli[14] idénticos e independientes, donde se considera como éxito el hecho de que un bit llega errado al destino. En ese sentido, el número de bits errados en una palabra código corresponden a una variable aleatoria con distribución binomial, de parametros n y p, donde p es la probabilidad de que un dígito binario llegue errado. Si t representa la capacidad de corrección (en bits) del código bloque lineal, la probabilidad de error en la decodificación de una palabra código se muestra en la ecuación A.1:

$$\begin{aligned} \Pr\{\mathcal{E}c\} &= \Pr\{\mathrm{X} > \mathrm{t}\} \\ &= \Pr\{\mathrm{X} \geq \mathrm{t}+1\} \\ &= \sum_{\mathrm{i}=\mathrm{t}+1}^{\mathrm{n}} \binom{\mathrm{n}}{\mathrm{i}} \mathrm{p}^{\mathrm{i}}(1-\mathrm{p})^{\mathrm{n}-\mathrm{i}}. \end{aligned} \tag{A.1}$$

Donde X es la variable aleatoria “número de bits errados en una palabra código de n bits”.

El valor de p depende del esquema de modulación usado, de la tasa de codificación y de la relación Eb/No. Si el esquema de modulación es 2-PAM[15], el cual presenta el mejor desempeño sobre un canal tipo AWGN, entonces la probabilidad de que un bit llegue errado se indica en la ecuación A.2.

$$p = Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}}\right). \tag{A.2}$$

[14] Experimento aleatorio con dos posibles resultados, etiquetados como éxito y fracaso.
[15] De los esquemas de modulación conocidos, 2-PAM es el que requiere menor valor de E_b/N_0 para lograr una determinada probabilidad de error.

Por otra parte, la probabilidad de error en la detección de un bit de información $\mathrm{P_r}\{\mathcal{E}b\}$, se puede estimar en forma aproximada escalando la cantidad $\Pr\{\mathcal{E}c\}$, como se expone en la ecuación A.3:

$$\Pr\{\mathcal{E}c\} = \mathrm{a}\,\Pr\{\mathcal{E}b\}, \qquad (A.3)$$

donde "a" corresponde al número promedio de bits de información que se detectan erradamente al cometer un error en la decodificación de un palabra código. Según [1], "a" se aproxima al valor mostrado en la ecuación A.4:

$$\mathrm{a} = \frac{2^k - 1}{2^{k-1}}. \qquad (A.4)$$

Dicho factor se calcula teniendo en cuenta el número promedio de vecinos, es decir, el número promedio de elementos del alfabeto de palabras código que se encuentran a una distancia d_H de la palabra en cuestión. De esta forma, la expresión aproximada para la probabilidad de error de bit se muestra en la ecuación A.5:

$$\Pr\{\mathcal{E}b\} = \frac{2^k/2}{2^k - 1} \sum_{i=t+1}^{n} \binom{n}{i} \left[Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}} \right) \right]^i \left[1 - Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}} \right) \right]^{n-i}. \qquad (A.5)$$

La anterior ecuación da una estimación sobre la probabilidad de error de bit con el método de decodificación *hard decision*.

Apéndice B. Probabilidad de Error con Decodificación *Soft Decision*

Es necesario aclarar ciertos aspectos teóricos antes de iniciar el desarrollo de la expresión para la probabilidad de error en *soft decision*.

Los puntos de la señal son equiprobables sobre un canal de ruido gaussiano distribuido idéntica e independientemente, en cuyo caso la regla de decisión óptima es una regla de distancia mínima basada en regiones de decisión óptima o regiones de decisión mínima [21]. En contextos diferentes a la codificación RM se asumen todas las palabras código equiprobables porque este es un supuesto peor caso.

Un alfabeto de palabras código se denota de la siguiente manera:

$$A = \{a_j, 0 \leq j \leq 2^k - 1\}. \qquad (B.1)$$

Los parámetros básicos de la constelación o alfabeto son: la dimensión n; el tamaño 2^k, la energía promedio $E(A)$:

$$E(A) = \frac{1}{2^k}\sum_j \|a_j\|^2. \qquad (B.2)$$

El cuadrado de la distancia mínima $d^2_{min}(A)$ y el número promedio de vecinos cercanos denotado por $k_{min}(A)$.

Para decodificar bloque por bloque, el modelo de canal es $Y = S + N$, donde todas las secuencias son n-tuplas. La secuencia S transmitida es seleccionada de forma equiprobable entre las 2^k n-tuplas a_j del alfabeto A.

Bajo la regla de decisión de distancia mínima, un espacio real $\mathbb{R}$ (Espacio denotado por A, el alfabeto de palabras código) es particionado en 2^k regiones de decisión, donde R_j denota los vectores recibidos $y \in \mathbb{R}^N$ que son al menos tan cercanos a a_j como a alguna otra palabra código en el alfabeto A.

$$R_j = \left\{y \in \mathbb{R}^N : \|y - a_j\|^2 \leq \|y - a_{j'}\|^2 \; para\; todo\; j \neq j'\right\}. \qquad (B.3)$$

Bajo la regla de distancia mínima, la decisión es a_j si y solo si la secuencia recibida $y \in R_j$. Las regiones de decisión R_j cubren todo el N-espacio, y son disjuntos excepto en sus límites.

Puesto que el vector de ruido N es un vector aleatorio continuo, la probabilidad que y esté precisamente en el límite de la región R_j es cero.

La región de decisión R_j es la intersección de las $2^k - 1$ regiones de decisión par $R_{jj'}$, definidas por:

$$R_{jj'} = \left\{y \in \mathbb{R}^N : \|y - a_j\|^2 \leq \|y - a_{j'}\|^2\right\}$$

$$R_j = \bigcap R_{jj'} \qquad (B.4)$$

$R_{jj'}$ Contiene la mitad del espacio de a_j que está limitado por un híper-plano bisector perpendicular entre a_j y $a_{j'}$, denotado por $H_{jj'}$ el cual es un conjunto de puntos en $\mathbb{R}^N$ equidistantes de a_j y $a_{j'}$.

La probabilidad de un error en decisión al transmitir una palabra a_j es la probabilidad que $y = a_j + N$ esté ubicado fuera de la región de decisión R_j, cuyo centro es a_j. De igual manera, es la probabilidad que la variable de ruido N se ubique fuera de la región trasladada $R_j - a_j$, cuyo centro es cero.

$$\Pr(E|a_j) = \int_{R_j} Py(y|a_j)dy = 1 - \int_{R_j} P_N(y - a_j)dy = 1 - \int_{R_j - a_j} P_N(n)dn. \quad (B.5)$$

Aunque no existen expresiones cercanas en forma a la integral gaussiana $\Pr(E|a_j)$, es posible obtener un límite superior en términos de probabilidades de error par, llamado límite de unión. El término superior del límite de unión se encuentra por estimación de frontera de unión, que es por lo general una buena aproximación, y el límite inferior (con el mismo comportamiento exponencial) puede ser obtenido al considerar el peor caso de la probabilidad de error par.

Cada probabilidad par posee una expresión que depende solo de la distancia al cuadrado $d^2(a_j, a_{j'}) = \|a_j - a_{j'}\|^2$ y la varianza de ruido $\sigma^2 = \frac{N_0}{2}$.

Un error puede ocurrir si y solo si la magnitud de la componente de ruido unidimensional $u_1 = u_{|a_{j'} - a_j}$, la cual es la proyección de u en el vector diferencia $a_{j'} - a_j$, sobrepasa la mitad de la distancia $d(a_{j'}, a_j) = \|a_{j'} - a_j\|$.

La distribución $f_N(u)$ del vector de ruido gaussiano distribuido independiente e idénticamente N es esféricamente simétrico, por lo tanto la fdp de alguna proyección unidimensional tal como u_1 es:

$$f_N(u_1) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(\frac{{u_1}^2}{2\sigma^2}\right)}. \quad (B.6)$$

Entonces, la probabilidad de error par $\Pr(a_j \to a_{j'})$ si a_j es transmitida, es en otras palabras, la probabilidad que el vector recibido $y = a_j + n$ se encuentre ubicado al menos tan cerca a a_j como a $a_{j'}$, su expresión es la siguiente:

$$\Pr(a_j \to a_{j'}) = \frac{1}{(2\pi\sigma^2)^{1/2}} \int_{d(a_{j'},a_j)/2}^{\infty} e^{-x^2/2\sigma^2}\, dx = Q\left(\frac{d(a_{j'}, a_j)}{2\sigma}\right). \qquad (B.7)$$

Donde $Q(\cdot)$ es la probabilidad gaussiana de una función de error.

El límite de unión en la probabilidad de error está basado en el límite de unión elemental de la teoría de probabilidad: Si A y B son dos eventos cualquiera, entonces $\Pr(A \cup B) \leq \Pr(A) + \Pr(B)$. En consecuencia, la probabilidad de que y sea tan próximo a $a_{j'} \in A$ como a a_j está limitado superiormente por la suma de probabilidades de errores par de todas las otras palabras código $a_{j'} \neq a_j \in A$.

$$\Pr(E|a_j) \leq \sum_{a_j \neq a_{j'} \in A} \Pr(a_j \to a_{j'}) = \sum_{a_j \neq a_{j'} \in A} \mathrm{Q}\left(\frac{d(a_j, a_{j'})}{2\sigma}\right). \qquad (B.8)$$

D denota el conjunto de distancias entre las palabras código pertenecientes a A; entonces se escribe el límite de unión como:

$$\Pr(E|a_j) \leq \sum_{d \in D} K_d(a_j)\mathrm{Q}\left(\frac{d}{2\sigma}\right). \qquad (B.9)$$

$K_d(a_j)$ es el número de palabras código $a_{j'} \neq a_j \in A$ a distancia d de a_j. Ya que $Q(x)$ decrece exponencialmente como $e^{-x^2/2}$, el factor $Q(d/2\sigma)$ será mayor para la distancia euclidiana:

$$d_{\min}(A) = \min_{a_{j'} \neq a_j \in A} \|a_{j'} - a_j\|. \qquad (B.10)$$

Y decrece rápidamente para distancias más grandes.

La estimación de límite de unión (Union Bound Estimate, UBE) de $\Pr(E|a_j)$ está fundamentada en la idea que los vecinos más cercanos a a_j a distancia $d_{min}(A)$ (si hay alguno) dominará la sumatoria. Si hay $K_{\min}(A)$ vecinos a distancia $d_{min}(A)$ de a_j, entonces:

$$\Pr(E|a_j) \approx K_{min}(a_j)\mathrm{Q}\left(\frac{d_{min}(A)}{2\sigma}\right), \qquad (B.11)$$

Este estimado solo es válido si los vecinos próximos más cercanos están a una distancia lo suficientemente grande y el número de ellos es pequeño; si estos

supuestos son violados, entonces los términos adicionales deben ser tenidos en cuenta en el estimado. Finalmente si hay al menos un vecino $a_{j'}$ a distancia $d_{min}(A)$, entonces se obtiene el límite inferior par:

$$\Pr(E|a_j) \geq \Pr(a_j \to a_{j'}) = \mathrm{Q}\left(\frac{d_{min}(A)}{2\sigma}\right). \qquad (B.12)$$

Puesto que debe haber un error en detección si y está más próximo a $a_{j'}$ que a a_j. Por lo tanto es el momento de obtener los limites superior e inferior de $\Pr(E|a_j)$ que tienen el mismo exponente o argumento de la función $Q(\cdot)$ y que difieren solo de un pequeño factor del orden de $K_{min}(A)$.

Es posible obtener límites superior e inferior similares y estimar la probabilidad total de error:

$$\Pr(E) = \overline{\Pr(E|a_j)}, \qquad (B.13)$$

La barra superior en la anterior expresión denota la expectativa sobre el conjunto de palabras código equiprobables en A. Por ejemplo, si $K_{min}(A) = \overline{K_{min}(a_j)}$ es el promedio de vecinos más cercanos a distancia $d_{min}(A)$, entonces la estimación del límite de unión de $\Pr(E)$ es:

$$\Pr(E) \approx K_{min}(A)\mathrm{Q}\left(\frac{d_{min}(A)}{2\sigma}\right), \qquad (B.14)$$

El desarrollo de una expresión para el cálculo de la probabilidad de error para un código de bloque lineal depende de los parámetros básicos tales como (n, k, d). Estos parámetros permiten obtener la probabilidad de ocurrencia, la distancia entre símbolos, la distancia de cada símbolo al origen y la varianza.

Para el caso particular del código lineal $RM(r, m)$, el desarrollo de la expresión para la probabilidad de error se describe con detalle en los pasos a continuación:

a. Cálculo de la distancia entre las posibles palabras código, denotada por d como la distancia mínima entre estas.

$$d^2 = \|a_j - a_i\|^2; i \neq j; a_j, a_i \in A. \qquad (B.15)$$

Las palabras código se mapean de tal manera que si el valor del bit es 1 se reemplaza por α y 0 por $-\alpha$, entonces, se realiza la resta entre 2 palabras código diferentes.

b. Cálculo de la distancia de las palabras código al origen. Es la distancia de cada bit de la palabra código al origen:

$$d_0^2 = \sqrt{\alpha_0^2 + \cdots + \alpha_{n-1}^2}$$

$$d_0 = \sqrt{n}\alpha. \qquad (B.16)$$

c. Cálculo de energía de símbolo y energía de bit. Como se muestra en las siguientes ecuaciones (B.17) y (B.18):

$$Es = 2^k \frac{1}{2^k} n\alpha^2,$$

$$= n\alpha^2. \qquad (B.17)$$

$$Eb = \frac{Es}{k},$$

$$= \frac{n}{k}\alpha^2, \qquad (B.18)$$

d. Finalmente, se reemplazan en el argumento de la función $Q(\cdot)$ los valores de la distancia mínima entre palabras código, la cual depende directamente de la distancia mínima Hamming y de la desviación estándar:

$$d = \sqrt{4 d_H \alpha^2}. \qquad (B.19)$$

$$\sigma = \sqrt{\frac{\frac{n}{k}\alpha^2}{2Eb/No}}. \qquad (B.20)$$

Ahora se sustituyen los valores calculados anteriormente como se indica a continuación:

$$\Pr(E) = K_{min}(A)\, Q\left(\frac{d}{2\sigma}\right),$$

$$= K_{min}(A)\ Q\left(\frac{2\sqrt{d_H\alpha^2}}{2\sqrt{\frac{\frac{n}{k}\alpha^2}{2Eb/No}}}\right),$$

$$\Pr(E) = K_{min}(A)\ Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}d_H}\right). \qquad (B.21)$$

$K_{min}(A)$ es la cantidad promedio de vecinos que tiene una palabra código, y es aproximadamente:

$$K_{min}(A) = \frac{2^k - 1}{2}. \qquad (B.22)$$

Entonces la probabilidad de error de bit con método de decodificación *soft decision* está dada por:

$$\Pr(E) = \frac{2^k - 1}{2}\ Q\left(\sqrt{2\frac{k}{n}\frac{Eb}{No}d}\right). \qquad (B.23)$$

Anexos

Anexo A. Tablas de desempeño para los sistemas con Codificación Reed-Muller y el sistema sin Codificación

Tabla A.1 Desempeño del sistema con codificación RM(0,3) – *Hard decision*.

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,13877	0,00354026	0,20988
1	0,11104	0,00271915	0,17185
2	0,08493	0,00213023	0,13488
3	0,06373	0,0031145	0,10044
4	0,04224	0,00172833	0,07009
5	0,02648	0,00103254	0,04509
6	0,01497	0,0010122	0,02623
7	0,00742	0,00088794	0,01346
8	0,00323	0,00077896	0,00591
9	0,00113	0,00011986	0,00214
10	2,84E-4	4,0332E-05	6,10E-4
11	6E-5	2,4944E-05	1,29E-4
12	8E-6	4,2164E-06	1,90E-5

Tabla A.2 Desempeño del sistema con codificación RM(0,3) – *Soft decision.*

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,07811	0,00375483	0,0786
1	0,05568	0,00267158	0,0563
2	0,03714	0,00228045	0,0375
3	0,02272	0,00143046	0,0229
4	0,01244	0,00140728	0,0125
5	0,00592	0,00097045	0,0060
6	0,00222	0,00043153	0,0024
7	7,74E-4	0,00011946	7,72E-4
8	1,82E-4	3,9384E-5	1,91E-4
9	3,20E-5	2,201E-5	3,36E-5
10	3,50E-6	1,5E-6	3,87E-6

Tabla A.3 Desempeño del sistema con codificación RM(1,3) – *Hard decision*.

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,16435	0,00290407	0,19743
1	0,12282	0,00248302	0,15060
2	0,08515	0,00278053	0,10612
3	0,05373	0,00113339	0,06761
4	0,02938	0,00115864	0,03797
5	0,01400	0,00015537	0,01822
6	0,00550	0,00013126	0,00721
7	0,00172	7,2434E-05	0,00225
8	4E-4	6,0077E-05	5,25E-4
9	6,08E-5	2,061E-05	8,61E-5
10	4,80E-6	5,3036E-06	9,12E-6

Tabla A.4 Desempeño del sistema con codificación RM(1,3) – *Soft decision.*

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,0906125	0,00225414	0,17062
1	0,0564725	0,00194052	0,09311
2	0,0296500	0,00106504	0,04427
3	0,0131825	0,00060174	0,01772
4	0,0048325	0,00067824	0,00572
5	0,0012550	0,00020062	0,00140
6	2,4925E-4	3,7732E-5	2,47E-4
7	2,9250E-5	1,7914E-5	2,83E-5
8	2,06E-06	3,49E-6	1,90E-6

Tabla A.5 Desempeño del sistema con codificación RM(2,3) – *Hard decision*.

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,24952	0,00281256	0,27301
1	0,19871	0,00317764	0,21918
2	0,14731	0,00232206	0,16370
3	0,1	0,00250003	0,11170
4	0,06006	0,00167043	0,06820
5	0,03232	0,00121485	0,03639
6	0,01455	0,0009099	0,01650
7	0,00535	0,00062195	0,00613
8	0,00151	0,00025745	0,00179
9	3,63E-4	0,00011092	3,88E-4
10	5,44E-5	1,2749E-5	5,79E-5
11	4,87E-6	2,68E-6	5,41E-6
12	2,3E-7	1,89E-7	2,80E-7

Tabla A.6 Desempeño del sistema con codificación RM(2,3) - *Soft decision.*

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,15639	0,01124174	0,5
1	0,11180	0,00506923	0,5
2	0,06451	0,00685357	0,5
3	0,03474	0,00320062	0,2612
4	0,01511	0,00255089	0,0960
5	0,00480	0,00104372	0,0278
6	0,00110	0,00030556	0,0060
7	1,47E-4	5,4315E-5	8,92E-4
8	1,74E-5	1,3945E-5	8,28E-5
9	1,36E-6	3,49E-7	4,27E-6

Tabla A.7 Desempeño del sistema con codificación RM(2,4) - *Hard decision*.

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,27816	0,00335659	0,29552
1	0,20855	0,00257538	0,22640
2	0,14189	0,00326633	0,15478
3	0,08156	0,00146077	0,09123
4	0,03952	0,00075736	0,04461
5	0,01560	0,00097138	0,01734
6	0,00439	0,00031905	0,00511
7	0,00092	0,00016708	0,00108
8	1E-4	7,0745E-5	1,5E-4
9	1,45E-5	9,8508E-6	1,35E-5
10	1E-6	2,1092E-6	6,54E-7

Tabla A.8 Desempeño del sistema con codificación RM(2,4) - *Soft decision.*

Eb/No	BER Promedio	Desviación Estándar	BER Teórico
0	0,15338182	0,0102949	0,5
1	0,09012727	0,00534575	0,5
2	0,04324545	0,00223136	0,5
3	0,01686364	0,00364198	0,47287
4	0,00450909	0,00201519	0,10321
5	0,00072636	9,7201E-5	0,01555
6	9,8182E-5	7,1428E-5	0,00147
7	7,51E-6	3,207E-6	7,77E-5
8	2,52E-7	1,63E-7	1,97E-6

Tabla A.9 Desempeño del sistema sin codificación – 2-PAM.

Eb/No	BER Teórico
0	0,07864
1	0,05628
2	0,03751
3	0,02287
4	0,01250
5	0,00595
6	0,00238
7	7,73E-4
8	1,91E-4
9	3,36E-5
10	3,87E-6
11	2,61E-7
12	9E-9

Anexo B. Comprobación de Equiprobabilidad de símbolos

Es importante destacar que se pudo comprobar la equiprobabilidad de los símbolos por medio de simulación. En la siguiente tabla están consignadas las probabilidades de cada uno de los posibles mensajes de un código, con longitud de palabra mensaje igual a 4, al enviar 100000 mensajes aleatoriamente.

Tabla B.10 Probabilidad de mensajes enviados de longitud 4.

Mensaje	Probabilidad
0000	0,06147
0001	0,06347
0010	0,06154
0011	0,06295
0100	0,0619
0101	0,06231
0110	0,0619
0111	0,06255
1000	0,06245
1001	0,06378
1010	0,0642

1011	0,06204
1100	0,06283
1101	0,06163
1110	0,06237
1111	0,06261
Total	1

Anexo C. Proceso de Decodificación *Hard Decision*

Para ilustrar los procesos de decodificación en forma clara, se supone el siguiente mensaje codificado del código RM(1,3), que se envía por un canal que introduce ruido y atenuación.

1 1 0 0 0 0 1 1

El receptor en este ejemplo en particular recibe la siguiente palabra código afectada por ruido:

1 1 0 0 0 1 1 1

Ahora se realiza el proceso detallado de la decodificación con el método *hard decision* descrito en la sección 2.9.2:

Inicialmente se calcula la matriz de paridad, Un código $RM(r,m)$ posee como matriz de verificación la transpuesta de la matriz generadora del código $RM(m-r-1,m)$. Sea H la matriz de verificación de paridad del código $RM(1,3)$ equivalente a la matriz generadora del código binario $RM(3-1-1,3) = RM(1,3)$ es:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Entonces,

$$H^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Nótese que para este caso en particular, la matriz de verificación H es también la matriz generadora G.

Antes de iniciar con el proceso, se construye la matriz Slepiana, que relaciona los patrones de error con sus respectivos síndromes. Los resultados se resumen en la siguiente tabla:

Tabla B.11 Patrones de error hasta de peso Hamming 2 del código RM(1,3) y sus síndromes.

Síndrome	Patrón de Error
1000	10000000
1001	01000000
1010	00100000
1011	00010000
1100	00001000
1101	00000100
1110	00000010
1111	00000001
0001	11000000
0010	10100000
0011	10010000
0100	10001000
0101	10000100
0110	10000010
0111	10000001

La región sombreada de la tabla corresponde a valores de síndrome relacionados con patrones de error de peso Hamming 2, dado que la capacidad de corrección del código RM(1,3) es 1, los valores restantes se consideran irrelevantes.

Paso 1: Cálculo del síndrome de la palabra recibida afectada por el ruido:

$$s(R) = RH^T$$

$$s(R) = [1\ 1\ 0\ 0\ 0\ 1\ 1\ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$s(R) = 1\ 1\ 0\ 1$$

Paso 2: Asociar el valor del síndrome de la palabra recibida con los síndromes de la tabla anterior para encontrar el patrón de error correspondiente. Para el caso, el patrón de error propio del síndrome calculado en el paso 1 es el siguiente:

$$0\ 0\ 0\ 0\ 0\ 1\ 0\ 0$$

Paso 3: Se evalúa la siguiente ecuación para finalizar el proceso de corrección:

$$C' = R \oplus E$$

$$C' = 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1 \oplus 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0$$

La palabra código corregida es:

$$C' = 1\ 1\ 0\ 0\ 0\ 0\ 1\ 1$$

Anexo D. Proceso de Decodificación *Soft Decision*

La decodificación *soft decision* realiza operaciones reales para la toma de decisiones, por lo tanto es necesario aclarar que la decisión depende de los valores de los niveles de energía de cada bit de la palabra código transmitida. A continuación se indica la palabra código a la salida del transmisor (la misma

palabra código del ejemplo) anterior y posteriormente los valores de energía de bit de la palabra en el receptor:

1 1 0 0 0 0 1 1

Suponiendo los valores de energía de cada bit en recepción afectados por el ruido:

0,82 0,64 0,33 0,26 0,4 0,51 0,62 0,89

A continuación se desarrollan los pasos expuestos en la sección 2.9.3:

Paso 1: Cálculo de la distancia euclidiana entre la palabra recibida y todas las posibles palabras código, para el caso, la mínima distancia euclidiana encontrada es la siguiente:

$$d_e(C,R) = \sqrt{\begin{matrix}(1-0{,}82)^2+(1-0{,}64)^2+(0-0{,}33)^2+(0-0{,}26)^2+\\(0-0{,}4)^2+(0-0{,}51)^2+(1-0{,}62)^2+(1-0{,}89)^2\end{matrix}} = 0{,}956$$

Paso 2. Finalmente se selecciona la palabra código como la palabra válida para la cual se obtuvo la mínima distancia euclidiana en el paso anterior. La palabra relacionada con la mínima distancia euclidiana es:

1 1 0 0 0 0 1 1