22

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

CAMPUS ARAGÓN

"CÓDIGOS CORRECTORES DE ERRORES"

TESIS
QUE PARA OBTENER EL TÍTULO DE
INGENIERO MECÁNICO ELECTRICISTA
AREA ELÉCTRICA ELECTRÓNICA

PRESENTA:

CORCHADO SALINAS SANDRA

ASESOR: ING. MARTIN HERNÁNDEZ HERNÁNDEZ

2001

A Dios y a mi madre,

A Dios que me ha dado todo lo que tengo y le debo todo lo que soy, y sobre todo porque me dio un ángel que ahora está a su lado, pero que me dejó la huella viva de su amor.

Al Dr. Robert Henry Morelos-Zaragoza Ascanio,
por su amabilidad, tiempo, paciencia, consejos y apoyo.

Y muy especialmente al Ingeniero Martín Hernández Hernández.

Indice

INTRODUCCIÓN

En las últimas décadas los códigos correctores de errores han tomado una importancia considerable, desde los trabajos de Shannon, Hamming y Golay, en los sistemas de comunicación. Su constante estudio ha dado origen a gran cantidad de resultados en la estructura matemática, así como en las técnicas empleadas. Esto, aunado al bajo costo de los componentes digitales, proporcionan los elementos necesarios para implementar sistemas de comunicación confiables y eficientes.

Dado el auge en el desarrollo de códigos correctores de errores, existen problemas para seleccionar alguno de ellos, aplicarlo a un canal de comunicaciones y adaptar sus parámetros. El diseñador por ende necesita conocer las técnicas de codificación, para seleccionar la más adecuada de acuerdo a los requerimientos de un sistema en particular.

Descritos en sus términos más simples, los códigos correctores de errores involucran la adición de redundancia[1] para datos transmitidos, tal que proporcionan los medios para detección y corrección de errores que ocurren inevitablemente en cualquier proceso de comunicación real. Estos errores se deben por ejemplo, a un pico eléctrico, un corto circuito momentáneo, o una interferencia electromagnética en el medio, fenómenos que afectan a más de un bit de información. Entonces puede decirse, que el objetivo de dichos códigos es proporcionar un nivel de seguridad a los datos digitales transmitidos.

Aunque existen, otras maneras de asegurar la transmisión de datos digitales, la codificación se prefiere debido a que resulta en la mayoría de los casos el proceso más económico para desarrollar un sistema confiable. Por ejemplo, en muchos sistemas de comunicación, una alternativa para el uso de codificación es simplemente proporcionar suficiente energía a la señal por unidad de información, para asegurar que la información

[1] En el Capítulo 3 se explica a detalle el concepto de redundancia.

CAPÍTULO 1

La naturaleza de la Información

La información que es procesada en los sistemas de comunicaciones, tiene su origen en el tipo de información[1] que se desea transmitir, de ahí la importancia de la naturaleza de la misma, ya que dependiendo de la presentación[2] que ésta tenga ante nuestros sentidos se verá forzada a cambiar a diferentes estados, sin modificar su mensaje, desde su fuente hasta su destino.

La información que nos interesa[3] transmitir tiene sus fundamentos básicos que se mencionaran en este primer capítulo.

I.1. ELEMENTOS DE COMUNICACIÓN

La información que viaja a través de todos los medios naturales adopta la forma de ondas, es decir, patrones repetitivos que oscilan. Por ejemplo, la luz es una serie de ondas, al igual que el sonido, la electricidad y las transmisiones de radio y televisión. Se dice que los medios propagan o reproducen esas ondas.

Un medio de transmisión se puede definir como algo a través de lo cual viaja información. El aire, el agua, el espacio e incluso los objetos sólidos son medios que llevan información en la naturaleza.

[1] Se refiere a la información que perciben nuestros sentidos, es decir, antes de que sea procesada por el sistema de comunicaciones.
[2] Para propósitos de este trabajo se tratará de información visual y sonora.
[3] Voz, datos y vídeo.

I.2.2.2. Frecuencia

La forma más común para describir una onda es especificando su frecuencia. La frecuencia es una medida del número de ciclos de onda que pasan por un punto fijo durante un periodo especificado. Generalmente la frecuencia se expresa en ciclos por segundo (cps). La nomenclatura para un ciclo por segundo es el hertz (abreviado Hz), en memoria del físico alemán Heinrich Hertz.

Mientras menor sea la longitud de onda, mayor será la frecuencia, y viceversa. Como las ondas viajan a través de un medio a velocidad fija, la frecuencia es inversamente proporcional a la longitud de onda:

velocidad = frecuencia x longitud de onda

La frecuencia determina los tonos que percibimos en el sonido, así como los colores que vemos.

I.2.2.3. Amplitud

La intensidad de onda o amplitud, es el máximo valor de la onda que determina la altura y profundidad de las cimas (crestas) y abismos(valles) respectivamente, a partir de su posición de equilibrio. Diferentes partes de una onda pueden tener amplitud positiva o negativa (Fig. 1.1). El punto donde la amplitud vale cero suele denominarse punto de cruce cero. La amplitud se mide en diferentes unidades[8].

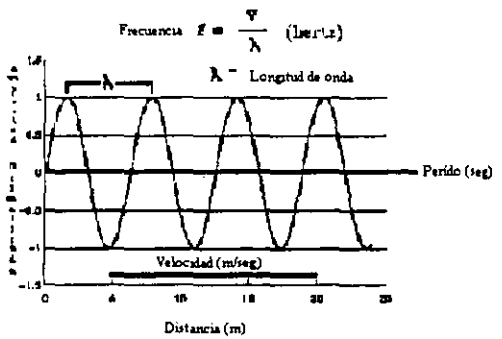


Figura 1.1. Características de una onda.

[8] En el campo de la electrónica por ejemplo, se mide el voltaje (unidad volt) y la corriente (unidad ampere).

Las ondas electromagnéticas incluyen las ondas de luz, radio, rayos X, ultravioleta, infrarrojo y rayos gamma. La única diferencia entre estas subclasificaciones de ondas es la longitud de onda. Las ondas electromagnéticas se comportan de manera similar a las de sonido, excepto que son oscilaciones de fuerzas eléctricas y magnéticas.

I.2.4.1. Electrostática y Magnetismo.

Los átomos tienen normalmente una carga neutral porque tienen igual cantidad de *protones* y *electrones*. Las cargas positivas y negativas se atraen entre sí para formar un enlace común; y otra fuerza cuántica mantiene una cierta distancia entre ellas. Los átomos buscan tener un número igual de protones y electrones dentro de su propia estructura atómica, manteniendo así una carga neutral.

Por ejemplo, dos átomos muy cercanos cada uno con un número internamente equilibrado de protones y electrones, si una fuerza externa hace que uno de esos átomos pierda un electrón y el otro lo capture, el átomo deficiente posee una carga positiva, y el que tiene un electrón adicional está cargado negativamente. Estas cargas opuestas se atraen entre sí, con una fuerza eléctrica que trata de corregir el desequilibrio de cargas positivas y negativas. Esta atracción se manifiesta como una serie de líneas radiantes de fuerza eléctrica: un *campo electrostático*.

Los campos electrostáticos se asemejan mucho a los campos magnéticos. La disipación de la energía hace que ambos tipos de campos, eléctrico y magnético cumplan la siguiente ley: "la fuerza de los campos varía inversamente con el cuadrado de la distancia a la fuente".

I.2.4.2. El Espectro Electromagnético

Las similitudes entre los campos electrostáticos y magnéticos no pasaron desapercibidas para los científicos del siglo pasado; sus experimentos demostraron que estos campos tienen una interrelación: un campo electrostático induce un campo magnético, y viceversa. Esta observación los llevó a la conclusión de que los dos tipos de campos son propiedades de un mismo tipo de onda, la onda electromagnética.

CAPÍTULO 2

Transmisión de Información

La humanidad al conocer mejor el funcionamiento básico de la naturaleza, ha sido capaz de aprovecharlo para lograr sus propios objetivos. El siglo pasado presenció más avances científicos que toda la historia precedente, tal crecimiento científico ha conducido a un avance en la forma en la cual nos comunicamos. En particular, el surgimiento de medios de almacenamiento electromagnéticos y ópticos, junto con la electrónica, han proporcionado las tecnologías fundamentales que hacen posibles los medios de almacenamiento actuales[10].

II.1. HISTORIA

La humanidad ha estado desarrollando métodos para comunicarse en una forma rápida y confiable. En un principio, utilizando señales (como el humo, gesticulaciones e incluso sonidos) o símbolos (pictogramas, letras, etc.). Sin embargo, su necesidad se fue acrecentando cuando comenzaron a relacionarse a mayores distancias, para las cuales hicieron uso de "mensajeros", que corrían a través de los caminos para llevar mensajes entre las poblaciones. También comenzaron a ayudarse de animales, como las palomas y los caballos para llevar los mensajes más rápido y con menor esfuerzo humano.

Dentro de los primeros intentos del hombre para comunicarse, en Nueva Guinea y las partes tropicales del continente americano, se ideó la telegrafía a base de tambores; en una forma similar, en China se empleaba el "Tam-tam", que era un plato de metal colgante al cual golpeaban para que produjera un tono audible. La antigua Grecia y el imperio Romano poseían sistemas telegráficos bien organizados, especialmente el telégrafo de antorchas. Los indios de Norteamérica utilizaron señales de humo para comunicarse.

[10] Como el disco compacto (CD, compact disk), y el DVD (digital versatil disc, o digital video disc).

operación de redes con satélites "inteligentes". Además, la "Internet" es una herramienta muy importante para las comunicaciones a largas distancias a través de casi todo el mundo, ofreciendo información y servicios a sus usuarios.

La transmisión de información a través de los años ha ido evolucionando, llevando a métodos más seguros, más rápidos y con mayor capacidad, pero aún sin ser perfectos, ya que el ruido y la atenuación siempre están presentes. De ahí el continuo desarrollo de técnicas de protección de información, como los códigos correctores de errores.

II.2. SEÑAL Y MENSAJE

El término "señal" se utiliza muchas veces para designar varios tipos de información emitida a través de un canal de comunicación. Por ejemplo:

- secuencias de letras que forman un telegrama,
- fluctuaciones de presión sonora que forman la entrada de un canal telefónico,
- imagen enviada a través de un sistema de televisión,
- magnitud transmitida a través de canales de telemetría o de control.

El término "mensaje" se utiliza preferentemente para la información antes de que haya sido modificada y pueda estar sujeta a errores y distorsiones del canal, y el término "señal" para el mensaje después de que haya sido modificado para la transmisión. Es también más útil considerar el término "señal" con respecto a la forma de onda eléctrica que lleva la información a través de las distintas etapas en el canal. "Señal de banda base" se utiliza para hacer referencia a la forma de onda eléctrica que aparece en la salida de un transductor[1] y en la entrada del sistema de comunicación. Se puede hacer una distinción general entre señales continuas y discretas. Por señal continua se entiende, una señal que puede ser representada por una función continua en el tiempo, $f(t)$. El término de "señal analógica" también se usa en las señales continuas, ya que la forma de onda eléctrica es análoga o similar a la forma de onda de entrada. Las señales discretas se producen en situaciones en las cuales el mensaje es una secuencia de letras o números, de forma que cada letra o número pueda ser definido por la representación de niveles de señal con un número limitado de posibilidades.

[1] Ver Glosario.

II.3.8. Distorsión

La distorsión es cualquier alteración de una señal con respecto a su forma original. Son muchos los tipos de distorsión que pueden afectar adversamente una señal electrónica, incluyendo la distorsión no lineal, la de frecuencia y la de fase.

La *distorsión no lineal* es el resultado de salidas que no suben ni bajan en proporción directa a las entradas, y hay varias subclases de esta distorsión. La *distorsión de amplitud* describe las diferencias en escala o razón entre las entradas y salidas al variar la amplitud. La *distorsión armónica* se debe a la adición de información con frecuencias armónicas por parte de un circuito o transductor, casi siempre en forma directamente proporcional a la amplitud de las entradas. La *distorsión de intermodulación* agrega frecuencias que no son por fuerza armónicos de las frecuencias componentes. La *distorsión de revoloteo* (*flutter*) se debe a desviaciones de la base de tiempo en mecanismos físicos, o en componentes electrónicos, como los osciladores.

La *distorsión de frecuencia* es un fenómeno en el que las salidas contienen frecuencias que no estaban presentes en las entrada. La *distorsión de fase* se refiere a relaciones de fase que difieren entre las entradas y las salidas.

II.3.9. Fidelidad

Fidelidad significa "fiel" en cualquier contexto. Aplicado a audio, vídeo e imágenes fotográficas, el término casi siempre indica qué tan fiel es la imagen producida en comparación con las entradas originales. La respuesta en frecuencia, la relación señal a ruido y la distorsión forman parte de la evaluación de la fidelidad.

II.3.10. Saturación

Cualquier medio o circuito llega a la *saturación* cuando entradas adicionales no pueden afectar ya las salidas. La *separación superior (headroom)* es la diferencia entre el nivel promedio de la señal y el nivel de saturación. Entre los elementos que están sujetos a saturación, se encuentran la cinta magnética, los circuitos electrónicos, y los colores y la luminancia en vídeo.

La *onda triangular* se asemeja a la sinusoidal en cuanto a que sube y baja suavemente de la amplitud máxima a la mínima, pero en este caso la acción es lineal.

Las *ondas diente de sierra* se llaman así por su forma. Cada ciclo sube gradualmente a su voltaje/amplitud máximo y después baja inmediatamente a su valor mínimo. Esta cualidad de rampa ha conferido a esta onda el sinónimo de *onda de rampa*; estas ondas se utilizan en la síntesis de sonido y en el control de cañones de electrones en pantallas de TRC, entre otras cosas.

La *onda rectangular* cambia bruscamente su amplitud mínima a la máxima, produciendo una serie de pulsos que pueden servir ya sea como señales cronométricas para sincronización o como representaciones de estados encendido/apagado (on/off). A este tipo de ondas se le denomina también *ondas de pulsos*, y la duración del estado encendido se conoce como *anchura de pulso* o *ciclo activo*. La *onda cuadrada* es un caso especial de onda rectangular que presenta un ciclo activo del 50%, siendo los tiempos de estado encendido y apagado iguales.

II.4.7. Modulación

Es posible crear formas de onda más complejas empleando una técnica denominada *modulación*. El concepto más simple es que una señal afecta o controla un cierto parámetro de otra[13], como su frecuencia, amplitud, fase o anchura de pulso. La modulación se utiliza en la transmisión, como las emisiones de radio AM y FM, televisión y fax. En la Figura 2.1 se muestra una clasificación de los tipos de modulación más importantes.

La modulación se requiere debido a que no es conveniente económica, física y socialmente transmitir las señales de información sin que antes se vean modificadas en una o varias de sus características fundamentales. Por ejemplo, sería costoso y difícil de construir antenas de varios kilómetros para transmitir y recibir señales de radio emitidas a su frecuencia de banda base y solamente poder escuchar una sola estación todos los días.

[13] Moduladora y portadora respectivamente.

Frecuentemente esta operación de muestreo es precedida por un filtro paso bajas, no mostrado en la Figura 2.1, que asegura que la señal está en una banda limitada. Cada muestra es entonces convertida a una palabra binaria de n-bits[14] por un convertidor analógico-digital (A/D). La salida es una ráfaga de bits con una velocidad de nf_s bits por segundo (frecuentemente escrito bps o b/s). La ráfaga de bits es entonces transmitida sobre un sistema de transmisión digital, es decir, la señal de voz o vídeo es transmitida digitalmente, y es reconstruida con un convertidor digital-analógico (D/A) y con un filtro paso bajas.

La implementación interna de cualquier sistema de transmisión digital puede ser compleja. Sin embargo, desde el punto de vista de transmisiones de voz y vídeo, el sistema de transmisión digital es simple, ya que este puede ser completamente caracterizado por cuatro parámetros:

* la velocidad de bit,
* la propagación y el proceso de retraso,
* la probabilidad de error, que indica como probablemente los bits llegan a un destino, y sus posibles diferencias con respecto a los bits transmitidos, y
* el defasamiento de la sincronización en la llegada de la ráfaga de bits.

Los errores en los bits y el defasamiento en la sincronización pueden causar degradación en la calidad de la señal de voz o vídeo recobrada, y el excesivo retraso puede deteriorar una conversación, así que el diseñador del sistema de comunicación digital debe controlar estos obstáculos. Estos, sin embargo, están lejos de ser menos complicados que los tipos de distorsiones comúnmente encontrados en transmisiones analógicas.

Otra aplicación de las técnicas de comunicación digital es el sistema de almacenamiento usando medios magnéticos u ópticos. Estos medios tienen degradaciones únicas, diferentes a aquellas de los medios de transmisión, pero muchas de las mismas técnicas básicas se aplican.

[14] Bit es la contracción de *binary digit*.

- Con la tecnología moderna de compresión y transmisión, la aplicación de PCM a señales analógicas puede ser desarrollada con menos ancho de banda que con transmisiones analógicas de la misma señal.
- La comunicación digital da lugar a métodos complicados de sincronización que son frecuentemente evitados en comunicaciones analógicas.

Las comunicaciones digitales son el resultado de una combinación de factores económicos, avances tecnológicos, y demandas de nuevos servicios.

II.6. ELEMENTOS DE UN SISTEMA DE COMUNICACIÓN DIGITAL

En esta sección, se describirán los elementos de un sistema de comunicación digital. donde se menciona implícitamente los principales resultados de la teoría de la información que se aplica en los códigos correctores de errores; debido a la importancia del teorema de codificación de canal, cuyo resultado explica el límite fundamental de la eficiencia de un sistema de comunicación digital.

El análisis estándar de los sistemas artificiales de comunicación esta basado en el concepto de canal de comunicación como un medio para transmitir mensajes de una fuente de información a un destino. El alcance de este concepto esta apoyado en el análisis de los parámetros del sistema de comunicación, tales como el ancho de banda, la relación señal a ruido, entropía, y la máxima velocidad de transmisión, debido a que el concepto discrimina los problemas no concernientes a un canal físico de comunicaciones.

Los elementos básicos de un sistema de comunicación en un sólo sentido (half duplex)[16] son ilustrados con el diagrama de bloques de la Fig. 2.2.

[16] Modo de transmisión en el cual un transmisor y un receptor ocupan el canal en un solo sentido a la vez.

II.6.4. Modulador Digital

La función del modulador es acoplar la salida del codificador al canal de transmisión. El modulador acepta símbolos binarios o *M*-arios codificados y produce formas de onda apropiadas para el medio físico de transmisión. En muchos sistemas donde se aplica codificación, las técnicas y equipo de modulación y demodulación son difíciles o imposibles de modificar o remplazar. En otros casos, la técnica de modulación es fija, pero los cambios en el método de demodulación son posibles. En algunas aplicaciones, es posible diseñar el sistema modulación-demodulación dada la técnica de codificación.

Para la *modulación binaria*, el modulador simplemente convierte un dígito binario, 0 ó 1, a una forma de onda, $s_0(t)$ o $s_1(t)$ respectivamente, de igual duración T_s. Para modulación *M*-aria, los *M* posibles símbolos codificados son convertidos a un conjunto correspondiente de *M* formas de onda $s_0(t)$, $s_1(t)$, . . . , $s_{M-1}(t)$. Para la señalización binaria, los tipos de modulación convencional incluyen variación por corrimiento de fase (PSK), PSK diferencialmente codificada (DPSK) y variación por corrimiento de frecuencia (FSK). Las formas no binarias de estos tipos de modulación básica son PSK *M*-aria (MPSK), DPSK *M*-aria (MDPSK) y FSK *M*-aria (MFSK). Con las formas convencionales de estos tipos de modulación, el ancho de banda nominal de cada forma de onda $s_i(t)$, donde i= 0,1,2, . . . , *M*-1, es aproximadamente $1/T_s$. Sin embargo, en señalización del espectro expandido, como su nombre lo implica, el ancho de banda de cada forma de onda puede ser mucho mayor que $1/T_s$. Por ejemplo, una versión de espectro expandido de PSK binaria utilizaría formas de onda $s_0(t)$ y $s_1(t)$, en las que $s_0(t)$ es una secuencia de pulsos binarios PSK mucho más cortos, normalmente llamados *chips*, y $s_1(t)$ es el complemento de la secuencia del chip en $s_0(t)$. La señalización del espectro expandido es usada como una técnica de acceso múltiple y también como un medio de protección de un sistema de comunicación contra las interferencias.

II.6.5. Canal de Transmisión

Se incluyen en el termino de canal de transmisión todas las operaciones necesarias para preparar las formas de onda en banda base moduladas para la transmisión en los canales físicos, los medios de transmisión y las operaciones de recepción requeridas para llevar las

II.6.6. Demodulador Digital

El demodulador proporciona la interface entre el canal de transmisión y las funciones que entregan los cálculos de los datos transmitidos al receptor. En esta definición de canal de transmisión se incluye el equipo de recepción de radiofrecuencia. El demodulador opera en la forma de onda recibida de cada símbolo de transmisión y produce un número o un conjunto de números que representa un cálculo de un símbolo binario M-ario transmitido.

En el más simple de los casos, el demodulador está diseñado para hacer una definición de cada símbolo recibido, por 0's ó 1's para transmisión binaria, o de 0, 1, ..., M-1 para transmisión M-aria; que puede ser demodulación estrictamente algebraica (*hard-decision demodulation*). Las formas de onda transmitidas que fueron corrompidas por el canal de transmisión, su decodificación esta expuesta a errores, y el índice promedio de ocurrencia de errores en los símbolos se toma como una fracción del número total de símbolos recibidos en un largo periodo de tiempo, y se le llama *índice de error por símbolo* (symbol-error rate) o *probabilidad de símbolo erróneo* (probability of symbol error). Para transmisión binaria, es el *índice de bit erróneo* (bit-error rate) o *probabilidad de bit erróneo* (probability of bit error). Convencionalmente se aplica la misma terminología de bit erróneo después de una decodificación de símbolos M-arios para sus representaciones binarias. En un sistema codificado el índice de error en este punto del sistema es frecuentemente llamado *índice de error de canal puro* (raw-channel error rate) o *índice de error no codificado* (uncoded error rate) para hacer distinción de los errores estadísticos medidos después de que la decodificación es desarrollada.

Todos los canales de transmisión reales son analógicos, los cuales manejan formas de onda que pueden variar continuamente sobre algún rango limitado por no idealidades del medio de transmisión y del equipo de recepción. Así que la salida del demodulador puede ser vista como una forma de onda filtrada seguida por cuantización, de Q niveles. En el caso de demodulación binaria estrictamente algebraica (*hard-decision binary demodulation*) se requieren Q=2 niveles de cuantización. Si la salida del demodulador binario es cuantizada en Q>2 niveles, se refiere a una demodulación probabilística (*soft-decision demodulation*). La cuantización incide en una pérdida de información, así que la demodulación probabilística tiene información que puede ser utilizada, como una técnica de decodificación para la adecuada corrección de errores.

CAPÍTULO 3

Fundamentos de los Códigos Correctores de Errores

En este capítulo se presenta la teoría elemental de los códigos correctores de errores, con el objetivo de cubrir el material esencial para el entendimiento de los conceptos del capítulo siguiente.

III.1. CLAUDE SHANNON Y LA TEORÍA DE LA INFORMACIÓN

Claude Elwood Shannon nació el 30 de abril de 1916 en Michigan, Estados Unidos. En 1932 comenzó a estudiar Matemáticas e Ingeniería Eléctrica en la Universidad de Michigan, donde se graduó. Ingreso al MIT (Massachusetts Institute of Technology) en 1936 como asistente de investigador. Un año más tarde presento su tesis de maestría "A Symbolic Analysis of Relay and Switching Circuits", sobre el uso del álgebra de Boole para el análisis y optimización de circuitos de conmutación por relevadores. En la primavera de 1940, Shannon termino sus estudios con títulos de M.S. en Ingeniería Eléctrica y de Ph. D. en Matemáticas.

Trabajó para los Bell Telephones Laboratories desde 1941 como investigador matemático, y en los cuales permaneció hasta 1972. Durante este período grandes matemáticos y científicos también pertenecían a los Bell Laboratories, incluyendo a los teóricos Harry Nyquist y Hendrik Bode, y a los inventores del transistor Bardeen, Brittain, y Shockley. Publicó "Una Teoría Matemática de la Comunicación" en el Bell System

codificar información. Esta teoría puede ser aplicada para medir por ejemplo el problema de borrar información o, equivalentemente, la cantidad de trabajo necesario para transmitir información confiablemente a través de canales con ruido.

La creación de esta teoría ha sido seguramente uno de los grandes avances científicos en este siglo, la cual establece firmemente el concepto de la información como un fundamento de los sistemas de comunicaciones.

III.2. TEORÍA DE LA COMUNICACIÓN

La palabra información se puede interpretar de varias maneras, como volumen de datos, datos disponibles o semántica. Shannon empleo la palabra información para indicar una medida del total de datos a ser comunicados. El trabajo de Shannon, debe su importancia a que proporciona un estándar de rendimiento que no puede ser excedido por ningún canal de comunicaciones, y también contempla los factores que limitan el rendimiento de un canal.

III.2.1. Información y Entropía

Antes de Shannon, Robert Wiener postulo que la información no es nada más que entropía, pero no desarrollo ningún concepto formal que hubiese apoyado su propuesta. Algunos consideran que el comienzo de la teoría de la información tuvo lugar en los documentos de H. Nyquist (1924) y R. V. L. Hartley (1928), quienes realizaron estudios matemáticos sobre la capacidad de comunicación desarrollada por circuitos telegráficos, y en general, acerca de la cantidad de datos que pueden ser transmitidos confiablemente sobre cualquier canal físico. Para la formulación de estos problemas emplearon una medida matemática de la información. Hartley considero una fuente de información generadora de mensajes, donde cada mensaje tiene la misma probabilidad de todo el conjunto discreto. Hartley sugirió que la medida más natural de la información era una función logarítmica, en donde el contenido de la información de un mensaje tomado de un conjunto de M mensajes igualmente probables es $\log M$, donde la base del logaritmo es arbitraria y depende de la unidad de información. Además todas las medidas logarítmicas pueden estar relacionadas directamente, por la expresión:

III.2.2. Teoría de Codificación

El estudio de los códigos correctores de errores y las matemáticas asociadas es conocido como Teoría de Codificación[21], y se ocupa del diseño de códigos correctores de errores para la transmisión confiable de información a través de canales ruidosos. Esto permite la utilización de técnicas algebraicas modernas y clásicas que involucran Campos Finitos, Teoría de Grupos y Álgebra de Polinomios. Además, se relaciona con otras áreas, como las que involucran Matemáticas Discretas, especialmente la Teoría Numérica y la Teoría de Diseños Experimentales.

III.2.3. El Concepto de Codificación Ideal

El receptor de la figura 3.2, es esencialmente un demodulador o decodificador, donde S_i y N_i son las potencias promedio de la señal y el ruido para la entrada, S_o y N_o son las potencias promedio de la señal y el ruido a la salida, B_T el ancho de banda de transmisión, y B_m el ancho de banda del mensaje, donde $B_T >> B_m$, para recibir una señal de alta calidad a pesar de una baja relación señal a ruido. Sin embargo un receptor práctico generalmente mejora la relación señal a ruido (SNR). Ejemplos de sistemas no codificados capaces de desarrollar esto son los que involucran FM y PPM.

Figura 3.2. Intercambio de potencia-ancho de banda en un receptor.

Pero la transmisión más eficiente puede solamente ser desarrollada usando un sistema codificado. De hecho, se busca un sistema codificado que proporcione una pequeña probabilidad de error a la salida del receptor, dado un valor infinito para S_i/N_i. Esto se puede ver como la transformación de una señal o símbolo dentro de otra señal o secuencia de símbolos para la transmisión. Usando un amplio concepto de codificación, es posible considerar un mensaje codificado como un vector en un espacio multidimensional de

[21] Algunas veces se le llama Codificación Algebráica.

Cada palabra de código está separada de las otras por alguna distancia. La mejor aproximación para todas las palabras de código es la igualmente espaciada (un código perfecto), por lo tanto un código con tales características es igualmente robusto contra la adición de ruido. Sin embargo, al incrementar las habilidades de corrección de errores de un código, implica la disminución en la velocidad de transmisión de los datos y la efectividad del ancho de banda.

Las señales a las que se aplican los códigos correctores de errores, tienen más bajas entropías que la señal original. El ancho de banda efectivo óptimo se obtiene cuando existe suficiente corrección de errores para cancelar el ruido en el canal. Conociendo el ruido del canal es posible calcular la entropía de una señal óptimamente codificada.

III.3.1. Breve Historia de la Corrección de Errores.

En 1948 con la publicación de la teoría de la información Claude Shannon, cuyo principal resultado es la "Teoría Matemática de Comunicación", demostró que la única forma de obtener la mayor capacidad en un dispositivo de almacenamiento o la más rápida transmisión en los canales de comunicación es a través del uso de sistemas correctores de errores muy poderosos. Shannon no ofreció algoritmos o procedimientos para alcanzar tales objetivos, sino que mediante sus estudios emergió una nueva área de investigación dentro de las comunicaciones digitales: la *Teoría de la Codificación*. Muchos investigadores se encargaron de fundamentar esta teoría.

1950: R.W. Hamming introdujo el primer código de bloques corrector de un solo error, aplicado todavía en los sistemas de comunicaciones.

1955: P. Elias publico el concepto de los códigos convolucionales.

1959: R. C. Bose y D. K. Chaudhuri propusieron una clase de códigos correcctores de errores múltiples que fue descubierta independientemente por A. Hocquenghem, tales códigos son conocidos como códigos BCH.

1960: I.S. Reed y G. Solomon desarrollaron un esquema de codificación que es usado por los códigos de bloques más poderosos, particularmente en aquellos que a causa de su capacidad de corrección se les relaciona con las ráfagas de errores.

La diferencia entre estos dos diagramas (Figuras 3.4 y 3.5) es que el primero tiene una distancia de dos entre cada par de palabras codificadas y el segundo diagrama tiene una distancia de tres entre cada par de palabras codificadas. En general, se pueden detectar t errores si la distancia mínima entre cualquiera de las dos palabras codificadas en un código es t+1. Y se pueden corregir t errores si la distancia mínima del código es $2t$+1.

Por ejemplo, si una palabra del código[23] o vector del código v_1 es recibida como un vector r con t errores (Figura 3.6), y si el vector de código más cercano v_2 esta por lo menos a $2t$+1 posiciones más allá de v_1, entonces r debe diferir de v_2 por menos t+1 posiciones.

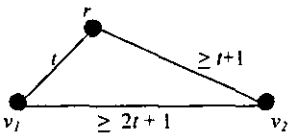


Figura 3.6. Distancia de código.

Asumiendo que un máximo de t errores han ocurrido, el receptor puede entonces asociar r con un vector legítimo, por ejemplo con v_1. Este criterio se aplica en general a los códigos convolucionales y a los de bloques que pueden corregir t errores, lo que se expresa como:

$$d \geq 2t + 1$$

donde d es la distancia mínima de Hamming entre todas las posibles parejas de secuencias de código.

Para la detección de errores se tiene que e errores pueden ser detectados de acuerdo a la desigualdad:

$$d \geq e + 1$$

y generalmente es:

$$d = e + t + 1$$

donde e denota el número de errores que pueden ser detectados y t denota el número de errores corregidos, pero se debe tener en cuenta que siempre $t \leq e$.

[23] Recomendación UIT-T V.42bis, Apéndice A.

Por ejemplo, un campo finito $GF(q)$, donde $q>1$ y q es un número primo, tendrá los elementos $0, 1, 2, \ldots, q-1$, con las operaciones de adición y multiplicación en módulo q. El más simple campo finito primo usa aritmética de módulo 2 y esta definido como:

$$GF(2)=\{0,1\}$$

donde 0 y 1 son los elementos identidad aditivo y multiplicativo respectivamente. Estos elementos deben satisfacer los postulados del campo, que conducen a las reglas de los códigos correctores de errores. Los elementos del campo finito $GF(2)$ por lo tanto satisfacen las tablas de adición y multiplicación en módulo 2.

III.3.5.3. Campos Extendidos

El campo $GF(2^m)$, donde $m>1$, es llamado *campo extendido* de $GF(2)$, son particularmente usados por los códigos cíclicos. Para generar un campo $GF(2^m)$ se extienden los elementos 0 y 1 usando un elemento primitivo α, donde:

$$\alpha \in GF(2^m)$$

entonces, el postulado de cerradura bajo la multiplicación implica que $\alpha \cdot \alpha = \alpha^2$, $\alpha \cdot \alpha^2 = \alpha^3$, etc. son elementos de $GF(2^m)$. Entonces:

$$GF(2^m)=\{0, 1, \alpha, \alpha^2, \alpha^3, \ldots\}$$

Por definición, $GF(2^m)$ debe ser finito y este puede desarrollarse asociándolo con un polinomio $p(x)$ de grado m que es primitivo sobre $GF(2)$. Esencialmente, $p(x)$ limita al campo a un número finito de elementos y define las reglas de la adición y multiplicación que deben ser satisfechas por estos elementos.

III.3.5.4. Multiplicaciones y Divisiones

Una propiedad de los campos extendidos es que existe por lo menos un elemento α, cuyas potencias generan todos los elementos diferentes de cero del campo. Como un ejemplo, un generador para $GF(5)$ es 2, cuyas potencias (comenzando de 2^0) son 1,2,4,3,1... . Potencias de α repetidas con un periodo de longitud $q-1$, por lo tanto $\alpha^{q-1}=\alpha^0=1$.

donde las s_i, i=1,2,3, cuentan los errores del canal que ocurren en un bloque de 6 bits y dan como resultado valores diferentes de cero. Cuando ningún error ocurre, las s_i=0, para i=1,2,3, porque los verificadores de paridad han sido seleccionados para satisfacer las ecuaciones de los bits de información (c_4, c_5, c_6). Existen 2^3=8 posibles valores para las s_i (en las ecuaciones s_1, s_2, s_3) ya que $r=n-k=3$.

Para m=(0,0,0), y c como un vector de 6 elementos, todos ellos siendo ceros. Se asume que un error ocurre en la posición 2, si el elemento c_2 es recibido como un 1 en vez de un 0. Entonces de las ecuaciones s_1, s_2, s_3, considerando la Tabla 3.4, se tiene que el vector s=(1, 1, 0) y ningún otro patrón de error contiene un solo error que proporciona este valor de s. Entonces, en el decodificador, si este valor de s ocurre, se complementa a c_2 para obtener el vector de código decodificado correctamente. En la Tabla 3.4 se listan los valores de s y sus patrones de error. El vector s es conocido como *vector síndrome*[31], y a la decodificación que utiliza los resultados de la Tabla 3.4 es conocida como *decodificación del síndrome*.

Error Patrón						Síndrome		
0	0	0	0	0	0	0	0	0
e	0	0	0	0	0	1	0	1
0	e	0	0	0	0	1	1	0
0	0	e	0	0	0	0	1	1
0	0	0	e	0	0	1	0	0
0	0	0	0	e	0	0	1	0
0	0	0	0	0	e	0	0	1
e	0	0	0	e	0	1	1	1

Tabla 3.4. Error patrón y Síndrome para el código (6, 3, 3).

Las ecuaciones de paridad como c_4, c_5, c_6 forman la base de los códigos correctores de errores.

III.3.7. Matriz de los Códigos de Bloques

Las ecuaciones verificadoras de paridad pueden ser escritas como vectores en un arreglo matricial de la forma:

$$\mathbf{c = mG}$$

donde:

[31] En terminología médica, un síndrome es un patrón o síntomas que son usados para diagnosticar una enfermedad. En este caso la "enfermedad" consiste de bits erróneos.

Mientras el número de las fallas de los componentes es menor o igual que la capacidad de corregir errores del código, las fallas no provocarán pérdidas de datos o de eficiencia.

III.4. DETECCIÓN DE ERRORES

Para que un código tenga la capacidad de detectar errores se requiere agregar bits dentro de las tramas o palabras para construir redundancia dentro de los mensajes. Las técnicas empleadas comúnmente son *verificación de paridad* y *verificación de redundancia.*

III.4.1. Paridad

En la verificación de paridad en el transmisor se cuenta el número de 1's en cada palabra a transmitir. Existen dos tipos de paridad: la impar y la par. Con paridad impar, si el número de 1's dentro de la palabra es par, entonces el bit de paridad es 1 para que el número total de 1's sea impar. Con paridad par, si el número de 1's dentro de la palabra es impar, entonces el bit de paridad es 1 para que el número total de 1's sea par. El dispositivo receptor cuenta los 1's de cada palabra, si el número de 1's de una palabra no es acorde con la paridad que utiliza el sistema, entonces se cuenta un error.

Este es un procedimiento poco exigente en cuanto al ancho de banda, puesto que la prueba de paridad requiere sólo de un dígito extra por palabra y el incremento necesario en el ancho de banda para introducirlo es relativamente pequeño. Si sólo ocurre un error en la palabra del mensaje, la prueba de paridad lo encontrara e indicará un error. La prueba de paridad fallará en detectar un error si dos dígitos de la palabra son incorrectos, siendo la probabilidad de esto de:

$$\textit{Probabilidad de decodificación errónea} = (m+1)^2 p^2/2$$

Para una palabra de longitud m=100 y probabilidad de que ocurra un error en un bit de $p=10^{-5}$, la probabilidad de que falle la verificación de paridad es aproximadamente de 0.5×10^{-6}. Reduciendo el tamaño de la palabra disminuye esta probabilidad; por ejemplo, la probabilidad de fallar en la detección de un error es de 10^{-7} cuando m=44 y $p=10^{-5}$.

Ejemplos de métodos de redundancia que son aplicados en los sistemas ARQ de comunicaciones son: LRC (longitudinal redundancy check), VRC (vertical redundancy check), y CRC (cyclical redundancy check). Todos estos definen un grupo de bits redundantes dentro del bloque del mensaje llamado BCC. En LRC los bits de datos son agrupados dentro de un bloque que contiene un número específico de caracteres y un contador de bits que se encarga de agregar la redundancia o paridad[35] BCC. Las terminales transmisora y receptora generan por separado el conteo de los bits. El caracter LRC generado en el transmisor es enviado al receptor para ser comparado con el caracter LRC generado en este último. Si ellos son iguales, un caracter ACK es enviado de nuevo al transmisor; en caso contrario un caracter NAK es enviado de regreso al transmisor (como en la figura 3.8). LRC es usado en protocolos punto a punto. En otras aplicaciones LRC y VRC se emplean juntamente.

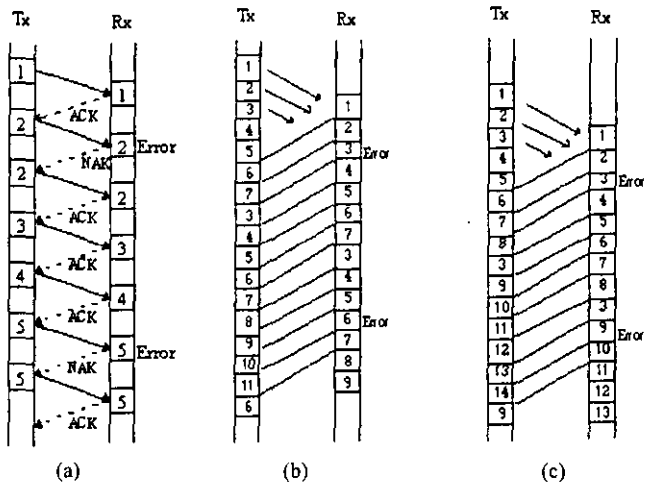


Figura 3.8. Tipos de operaciones ARQ. (a) ARQ Paro-y-espera (HDX), (b) ARQ continuo con retirada (FDX). (c) ARQ de selectiva repetición.

La técnica CRC es similar a la LRC, pero en esta se divide un bloque de bits entre un determinado número binario y el residuo de la división es el caracter verificador. Un caracter verificador es agregado y transmitido dentro del bloque del mensaje; este es comparado con el caracter verificador del receptor. Un CRC de 16 bits se usa principalmente en protocolos síncronos.

[35] A pesar de ser definido como un método de redundancia, comúnmente se le da el nombre de paridad a todos los caracteres binarios o no binarios, que son agregados a la información para detectar errores.

III.5.2.1. Errores Aleatorios y en Ráfagas

Los errores en la transmisión pueden ocurrir aleatoriamente o en ráfagas (cuando un sólo impulso de error afecta un número de símbolos contiguos transmitidos) y los códigos han sido diseñados específicamente para ambos casos. Por ejemplo los códigos RS son comúnmente usados para corregir las ráfagas de errores, mientras los códigos binarios BCH son principalmente usados para corregir múltiples errores aleatorios. En un código corrector de t errores aleatorios, cualquier patrón de t errores será corregido sobre una secuencia de longitud definida, y cada símbolo se considera que puede ser afectado *independientemente* por el ruido. Un código corrector de t errores puede también ser usado en un formato de codificación/decodificación *entrelazado*. El entrelazado (interleaving) es útil ya que permite la corrección de errores aleatorios y en ráfagas.

Otras técnicas para contrarrestar las mezclas de errores aleatorios y en ráfagas son los códigos *concatenados*, que utilizan en su desarrollo (Figura 3.10) un código *interno* para corregir la mayoría de los errores aleatorios y un código *externo* para corregir aquellas ráfagas de errores que saturan el código interno. El código interno puede ser un código binario o un RS, mientras que el código externo es generalmente un código RS. Sin embargo, su *índice de código* es el producto de los índices de los dos códigos (interno y externo) que puede ser demasiado bajo. Además, un código concatenado usualmente no es tan poderoso como el mejor de los códigos de un sólo estado con el mismo índice y longitud de bloque. Por otra parte, la complejidad del decodificador se reduce y los códigos concatenados han sido principalmente utilizados en grabaciones digitales, por ejemplo donde ocurren grandes pérdidas de datos.

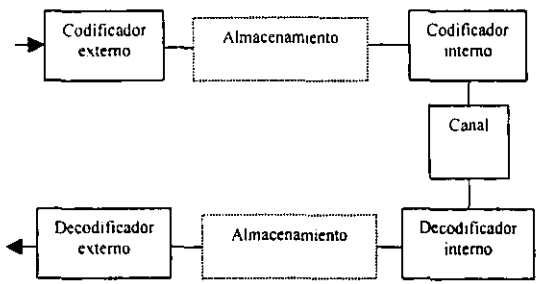


Figura 3.10. Código concatenado

CAPÍTULO 4

Tipos de Códigos Correctores de Errores

IV.1. RICHARD WESLEY HAMMING

Nació el 11 de febrero de 1915 en Chicago, Illinois, Estados Unidos. Estudió en la Universidad de Chicago donde recibió su B.S.[38] en 1937. Posteriormente ingresó a la Universidad de Nebraska donde realizó sus estudios de maestría y obtuvo su titulo en 1939. En la Universidad de Illinois obtuvo su título de Ph. D.[39] en matemáticas en 1942, con su tesis intitulada "Some Problems in the Boundary Value Theory of Linear Differential Equations".

En 1945 Hamming participó en el Proyecto Manhattan[40], un proyecto de investigación gubernamental de los Estados Unidos para producir una bomba atómica. Ingresó a los Bell Telephone Laboratories en 1946, donde trabajó hasta 1976 cuando acepto una cátedra de ciencias de la computación en la Naval Postgraduate School de Monterey, California.

Hamming es conocido principalmente por su trabajo sobre códigos detectores y correctores de errores. Su documento fundamental para la teoría de la codificación lo publicó en 1950.

[38] Se refiere al titulo profesional a nivel licenciatura.
[39] Se refiere al título profesional de doctorado.
[40] Este proyecto fue llamado Manhattan debido a que la primera investigación fue realizada en la Universidad de Columbia en Manhattan; sin embargo Hamming trabajó en el proyecto en Los Alamos.

Cada bit de paridad consiste de una suma en módulo-2 de los bits de datos, por ejemplo:

$$c_{k+1}=h_{11}\, d_1 \oplus h_{12}\, d_2 \oplus \ldots \oplus h_{1k}\, d_k$$

donde h_{11}, h_{12}, son los pesos binarios de los bits de datos correspondientes.

Por ejemplo, un código (15, 11)tiene las siguientes ecuaciones de paridad:

$$c_{12} = m_1 \oplus m_2 \oplus m_3 \oplus m_5 \oplus m_6 \oplus m_8 \oplus m_9.$$
$$c_{13} = m_2 \oplus m_3 \oplus m_4 \oplus m_6 \oplus m_7 \oplus m_9 \oplus m_{10}.$$
$$c_{14} = m_3 \oplus m_4 \oplus m_5 \oplus m_7 \oplus m_8 \oplus m_{10} \oplus m_{11}.$$
$$c_{15} = m_1 \oplus m_2 \oplus m_3 \oplus m_5 \oplus m_7 \oplus m_9 \oplus m_{11}.$$

Las palabras de código **c** se pueden representar como una operación matricial utilizando el vector de datos no codificados **m**:

$$\mathbf{c} = \mathbf{m}\, G$$

donde G es la matriz generadora de dimensión $k \times n$, y esta formada por la concatenación de la matriz identidad I y de la matriz de paridad P:

$$G = [I\ P]$$

Para el código (15, 11) de este ejemplo, la matriz P es:

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

Las columnas de la matriz P corresponden a las ecuaciones de los bits de paridad (c_{12}, c_{13}, c_{14}, c_{15}). Realizando la concatenación de la matriz identidad I y de la matriz de paridad P se obtiene:

$$M_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

donde las palabras de código son 00 y 01. La matriz M_4 proporciona cuatro palabras de código:

$$M_4 = \begin{bmatrix} M_2 & M_2 \\ M_2 & M_2^* \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

en donde M_2^* es la matriz complementaria de M_2 (donde cada elemento es reemplazado por su complemento). En general

$$M_{2n} = \begin{bmatrix} M_n & M_n \\ M_n & M_n^* \end{bmatrix}$$

La matriz Hadamard $n \times n$ proporciona n palabras de código cada una de n bits (con k bits de información por palabra, donde $n=2^k$ bits). Por ende, en los n bits de la palabra de código existen $r=n-k=2^k-k$ bits de paridad. Cabe denotar que incrementando k se incrementa también el número de bits de paridad en las palabras de código. Debido a esto, el índice del código llega a ser demasiado pequeño:

$$R_c = \frac{k}{n} = \frac{k}{2^k} = 2^{k-1}$$

Si para una k muy grande se pretendieran transmitir palabras codificadas al mismo índice que las palabras no codificadas, la transmisión codificada requerirá de un índice de bit más grande que el índice de bit no codificado por el factor $1/R_c$. Por ende, sería un incremento correspondiente en el ancho de banda requerido. A causa de lo mencionado, la codificación Hadamard demanda un gran ancho de banda, por lo que es usado solamente donde no hay restricciones de ancho de banda, como pruebas en el espacio profundo.

Considerando que la distancia de Hamming de un código ortogonal es:

$$d = \frac{n}{2} = 2^{k-1}$$

entonces el número de errores que pueden ser corregidos con un código Hadamard esta dado por:

Entonces el código Hamming extendido tiene una d_e=4. El incremento en este caso no mejora la capacidad correctora de errores del código, es decir con d_e=4 aún se tiene t=1. Sin embargo, permite que errores triples sean detectados, mientras que solamente dos errores pueden ser detectados con d_e=3.

IV.2.7. Códigos Borradura

En esta sección se describe un código basado en matrices Vandermonde calculadas sobre $GF(p^r)$[43]; que puede ser implementado eficientemente en microprocesadores comunes y estan disponibles para diferentes aplicaciones.

Los códigos borradura son códigos de bloques lineales. Los k bloques de la fuente de datos son codificados en el transmisor para producir n bloques codificados, de tal forma que cualquier subconjunto de k bloques codificados son suficientes para reconstruir los datos de la fuente. Tal código (n, k) permite al receptor recobrar n-k perdidas por lo menos en un grupo de n bloques codificados. La Figura 4.2 es una representación gráfica del proceso de codificación/decodificación.

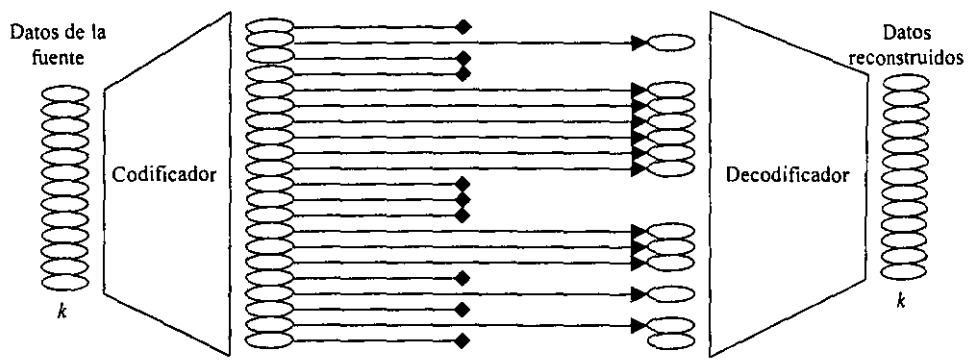


Figura 4.2. Representación gráfica del proceso de codificación-decodificación para un código borradura.

En telecomunicaciones, un bloque usualmente esta compuesto por un pequeño número de bits. En comunicaciones entre computadoras, la unidad de información es generalmente mucho más grande (un paquete de datos frecuentemente asciende a cientos o

[43] Campo Finito, ver Capitulo 3.

IV.2.7.4. Desempeño

Para un código sistemático, el codificador toma grupos de k bloques de datos de la fuente para producir n-k bloques redundantes. Esto significa que cada uno de los bloques de datos de la fuente es usado n-k veces y se puede esperar que el tiempo de codificación sea una función lineal de n-k. Esto es útil para medir el tiempo de generación de un bloque de datos que depende únicamente del parámetro k. Entonces, este tiempo es (para paquetes grandes) linealmente dependiente de k:

$$\text{Tiempo de codificación} = \frac{k}{c_e}$$

donde la constante c_e depende de la velocidad del sistema. Esta relación define la rapidez de construcción de los paquetes redundantes. Para un código sistemático los k bloques enviados desde la fuente de datos requieren del cálculo de n–k bloques redundantes. Así que la verdadera velocidad de codificación está dada por:

$$\text{velocidad de codificación} = \frac{c_e}{n-k}$$

El máximo índice de pérdidas que puede soportar el código es $(n\text{-}k)/n$, lo que significa que para un índice de máxima codificación dado, la velocidad de codificación puede decrementarse con n.

El tiempo que toma la decodificación depende del número de bloques de datos de la fuente. Para el proceso de reconstrucción es más práctico medir el tiempo por bloque reconstruido que es similar al tiempo de codificación. Entonces la velocidad de decodificación es escrita como:

$$\text{Velocidad de decodificación} = \frac{c_d}{l}$$

con el valor de la constante c_d ligeramente más pequeña que c_e.

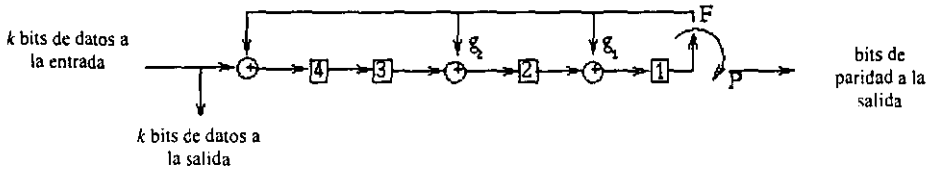
Los valores de c_e y c_d son lo suficientemente altos para permitir que estos códigos sean utilizados en un amplio rango de aplicaciones, dependiendo de los valores de k y l=n-k. Para una k dada, valores más grandes de l tienen ligeramente mejor desempeño en la

Para el polinomio generador $g(x) = x^4 + x^3 + x^2 + 1$, y n=7 por ejemplo, existen cuatro componentes diferentes de cero en g(x), esto es otro código cíclico (7, 3).

El polinomio generador G' es:

$$G' = \begin{bmatrix} x^6 & x^5 & x^4 & - & x^1 & - & - \\ - & x^5 & x^4 & x^3 & - & x^1 & - \\ - & - & x^4 & x^3 & x^2 & - & 1 \end{bmatrix} \leftarrow g(x)$$

Como se mencionó, los códigos cíclicos son muy sencillos de implementar, debido a que se basan directamente en el polinomio generador g(x). Usando el polinomio generador del ejemplo ($g(x) = x^4 + x^3 + x^2 + 1 = x^4 + g_1x^3 + g_2x^2 + 1$, $g_1=g_2=1$, $g_0=g_3=0$), se tiene su representación en la Figura 4.4.



Implementación del registro de corrimiento del ejemplo del código cíclico (7,3)

Figura 4.4. Ejemplo de un código cíclico con el polinomio generador $g(x) = x^4 + x^3 + x^2 + 1$.

Los cuadros numerados de la Figura 4.4 representan los elementos de un registro de corrimiento. Para esta implementación el número de bits del registro de corrimiento es igual al número de bits de paridad (n-k=4 con este código).

Como los bits son recorridos dentro del codificador, son leídos como los "k bits de datos a la salida", porque son los primeros k bits de las palabras de código sistemático de n bits. Después de que los últimos bits de datos han sido recorridos hacia afuera, el switch (interruptor) es movido desde F hasta P, y los (n-k) bits de paridad son leídos hacia afuera.

El ejemplo anterior puede ser adaptado por otros códigos, cambiando los pesos binarios (coeficientes de la matriz g(x)). La siguiente figura muestra este tipo de codificadores en términos más generales:

El síndrome requerido s(x) es el resultado de la división e(x)/g(x) y s(x) es cero si e(x) es cero.

El registro del síndrome (SR) es inicialmente puesto en ceros y r(x) es aplicado a la entrada (con el primer símbolo r_{n-1}). Posteriormente el polinomio r(x) es puesto dentro y el SR contiene el síndrome s(x) de r(x). La palabra de código recibida tendrá hasta *t* errores que pueden ser corregidos. La corrección de errores es desarrollada en un número de ciclos de reloj, usando un número de diferentes síndromes y está basado en el siguiente teorema de decodificación cíclica:

> Teorema. Si s(x) es el síndrome de r(x), entonces el residuo s'(x) resultante de la división xs(x) entre g(x) es el síndrome de r'(x), que es un corrimiento cíclico de r(x); la división xs(x) entre g(x) es realizada por un corrimiento en el SR una vez con s(x) como el contenido inicial.

Este teorema establece que dado el síndrome inicial s(x) de r(x), entonces con el reloj del SR y la entrada en circuito abierto se generarán los síndromes correspondientes a las versiones de corrimientos cíclicos de r(x). Entonces r(x) es almacenada en un buffer de *n* símbolos y se pueden decodificar los síndromes al final del buffer.

IV.2.8.4.2. Operación del Decodificador Meggitt

Paso 1. Con una compuerta abierta y otra compuerta cerrada el r(x) entero se pone dentro del buffer y simultáneamente dentro del SR. Cuando el símbolo r_{n-1} ocupa el último estado del buffer, el SR contiene a s(x) correspondiente al r(x).

Paso 2. La corrección de errores símbolo por símbolo es desarrollada con la primer compuerta cerrada y la segunda compuerta abierta. Si r_{n-1} no tiene errores, entonces el SR y el buffer son recorridos una vez simultáneamente resultando s'(x) correspondiente a r'(x).

Si el s(x) corresponde a un patrón de errores que pueden ser corregidos, por ejemplo *t* o menos errores en r(x), con un error en un símbolo r_{n-1}, entonces el patrón detector genera $\hat{e}_{n-1} = 1$ para completar o corregir r_{n-1}. El polinomio corregido es:

Esta ecuación se resuelve para los valores de X_k, pero involucra ecuaciones no lineales, por lo que se utiliza un polinomio localizador de errores, $\sigma(x)$. La importancia de $\sigma(x)$ es que este traduce el problema dentro de un conjunto de ecuaciones lineales y sus raíces son los localizadores de errores, X_k.

Existen algoritmos para encontrar $\sigma(x)$, por ejemplo el *método directo de Peterson*, que es una aproximación capaz de decodificar códigos binarios BCH, pero se vuelve ineficiente para grandes valores de t (se dice $t>5$). En tales casos se utiliza el *algoritmo de Euclides para polinomios o algoritmo iterativo de Berlekamp*. Por ejemplo, el método de Peterson, para $\sigma(x)$ es:

$$\sigma(x)=x^t+\sigma_1 x^{t-1}+\ldots+\sigma_t$$

donde las raíces son los localizadores de error $X_1, X_2, \ldots, X_t$, entonces:

$$X_k^t+\sigma_1 X_k^{t-1}+\ldots+\sigma_t=0, \qquad k=1,2,\ldots t$$

Multiplicando por X_k^j se expande:

$$\begin{aligned}
X_k^{t+j}+\sigma_1 X_k^{t+j-1}+\ldots+\sigma_t X_k^j &= 0 \qquad k=1,2,\ldots,t \\
\therefore\ X_1^{t+j}+\sigma_1 X_k^{t+j-1}+\ldots+\sigma_n X_k^j &= 0 \\
X_2^{t+j}+\sigma_1 X_2^{t+j-1}+\ldots+\sigma_t X_2^j &= 0 \\
&\vdots \\
X_t^{t+j}+\sigma_1 X_t^{t+j-1}+\ldots+\sigma_t X_t^j &= 0 \\
\therefore \sum_{k=1}^{t} X_k^{t+j}+\sigma_1\sum_{k=1}^{t} X_k^{t+j-1}+\ldots+\sigma_n\sum_{k=1}^{t} X_k^j &= 0
\end{aligned}$$

Finalmente, sustituyendo por los términos de sumatoria de $s_i=e(\alpha^i)=\sum_{k=1}^{t} X_k^i$, resulta:

$$s_{t+j}+\sigma_1 s_{t+j-1}+\ldots+\sigma_t s_j=0, \qquad j=1,2,\ldots,t$$

Esta ecuación representa un conjunto de ecuaciones lineales que pueden ser resueltas por los coeficientes σ_i ($i=1,2,\ldots,t$) por lo tanto resulta $\sigma(x)$. Además, para los códigos binarios

$$s_{2k}=s_k^3, \qquad \kappa=1,2,3,\ldots$$

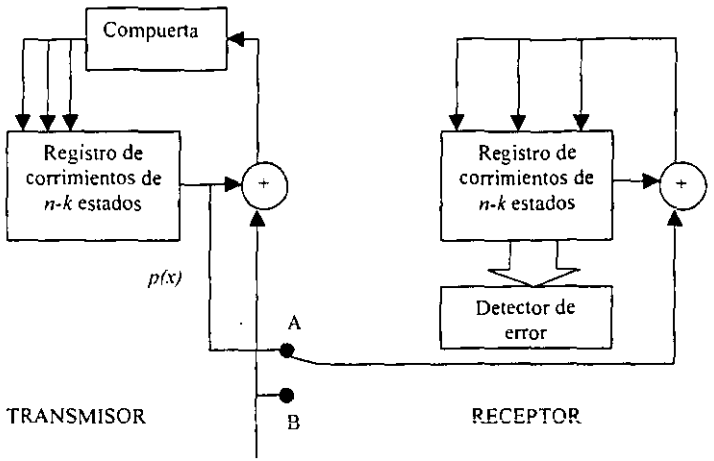


Figura 4.6. Detección de errores usando un CRC.

Estándar	g(x)
CRC-8	$1+x^8$
CRC-12	$1+x+x^2+x^3+x^{11}+x^{12}$
CRC-16	$1+x^2+x^{15}+x^{16}$
CRC-CCITT	$1+x^5+x^{12}+x^{16}$

Tabla 4.4. Polinomios CRC estándar.

IV.3.6. Códigos Reed-Solomon

Los códigos Reed-Solomon, son un subconjunto de los códigos cíclicos BCH diseñados para proporcionar corrección de múltiples errores. Se abrevian como códigos RS y son códigos no binarios, multisímbolo. Estos consideran símbolos de m bits y tienen $q=2^m$ posibles símbolos que son las palabras de código. Por ejemplo, para m=8 bits, existen 256 posibles símbolos. Un código Reed-Solomon (n, k) es también un código de bloques, en donde los k símbolos de información son introducidos al codificador y los n símbolos de las palabras de código se presentan a la salida. Los símbolos de los códigos Reed-Solomon están limitados en longitud por $n \leq q+1$, siendo generalmente de longitudes $n \leq q-1$ por símbolo las utilizadas como una restricción del diseño. Los códigos RS pueden ser extendidos en longitud de la palabra de código hasta $n = q$ y $n = q+1$.

$$m(x)= \alpha^3+\alpha x+\alpha^6 x^2$$

Aplicando las ecuaciones verificadoras de arriba resulta la palabra de código:

$$c=(\alpha,\alpha^2, \alpha^2, \alpha^6, \alpha^3, \alpha, \alpha^6)$$

IV.3.6.2. Decodificación de los Códigos RS

Para los códigos binarios BCH la tarea esencial es resolver la ecuación del síndrome para los localizadores de error X_k (donde k=1, 2, ... , t). Para los códigos RS es necesario resolverla para calcular los localizadores de errores (posiciones) y también los valores *erróneos*, Y_k, (donde k=1,2,...,t y cada símbolo es de m bits de profundidad). En este caso, para t errores en la palabra de código recibida, se puede escribir como

$$s_i = \sum_{k=1}^{t} Y_k X_k^i \ , i=1, 2, ..., 2t$$

El algoritmo de decodificación RS incluye todos los pasos de la decodificación de los códigos BCH binarios y un paso adicional para calcular la Y_k desde la ecuación anterior. Una vez que las X_k han sido determinadas, por ejemplo por medio del buscador Chien, entonces la ecuación se reduce a un conjunto de ecuaciones lineales simultaneas que son resueltas para las Y_k. Está ecuación resuelve errores simples y dobles en la palabra recibida:

$$t=1: \qquad Y_1 = \frac{s_1^2}{s_2}$$

$$t=2: \qquad Y_1 = \frac{s_1 X_2 + s_2}{X_1 X_2 + X_1^2}$$

$$Y_1 = \frac{s_1 X_1 + s_2}{X_1 X_2 + X_2^2}$$

La corrección de errores es desarrollada sumando Y_k al símbolo identificado como X_k, la adición es de acuerdo a las reglas de $GF(2^m)$.

Todos los símbolos del síndrome son requeridos y los coeficientes σ_i del localizador polinomial de error $\sigma(x)$ están (en general) dados por expresiones complejas que se encuentran en Tablas de valores calculados por los creadores de los códigos. Por ejemplo la Tabla 4.5 muestra los valores para un solo error y errores dobles.

IV.3.7.1. Detección de Ráfagas

Además de corregir de errores aleatorios, un código cíclico (n, k) puede también detectar ráfagas de errores. De $\frac{r(x)}{g(x)} = a(x) + \frac{e(x)}{g(x)}$, el síndrome s(x) es diferente de cero si el grado de e(x) es menor que el de g(x) (que tiene grado n-k). La división de r(x) entre g(x) tiene como resultado un síndrome diferente de cero. Una ráfaga de longitud n-k o menor es detectada por análisis del registro síndrome. Como ejemplo, el código BCH (15, 11) definido por

$$g(x)=1+x+x^4$$

detectaría la ráfaga

$$e(x)=1+x+x^2+x^3$$

$$\mathbf{e}=(1111000...)$$

La detección de fallas cuando el patrón de error es de la forma e(x)=a(x)g(x), indica que es verdaderamente una palabra de código. Sin embargo, si se considera por ejemplo:

$$e(x)=(x+x^3)(1+x+x^4)=x+x^2+x^3+x^4+x^5+x$$

$$\mathbf{e}=(01111101000...)$$

será indetectable para el código BCH (15, 11), entonces e(x) es divisible por g(x). Existen 2^k palabras de código, aproximadamente 2^n de las secuencias de n bits, la probabilidad de que un error no sea detectado es alrededor de $2^k/2^n=2^{-(n-k)}$. Para el código BCH(255,231) está probabilidad es aproximadamente de 2^{-24} ó $6\text{x}10^{-8}$. Una vez que una ráfaga ha sido detectada un requerimiento puede ser enviado para la retransmisión (ARQ).

IV.4. ENTRELAZADO

El entrelazado es una forma eficiente de usar un código corrector de t errores para evitar los errores aleatorios y en ráfagas. Puede ser implementado usando códigos de bloques y códigos convolucionales, la Figura 4.7(a) ilustra el principio para un código de bloques (n, k). Efectivamente, un bloque de i palabras de código está formado dentro de una matriz i x n y la matriz es transmitida *columna* por *columna* (como es indicado) para dar un *código entrelazado* (*in,ik*). En el receptor los datos son alimentados dentro de una columna matricial similar columna por columna y entonces de-entrelazada cuando se lee la matriz una fila a la vez.

Cuando la codificación es realizada, todo el contenido de los $k \times l$ registros de información así como los $r \times l$ bits de paridad son transmitidos sobre el canal. Generalmente la transmisión es (serial) bit por bit, fila por fila, esto es en el orden:

$$c_{rl} \cdots c_{r1} \cdots c_{1l} \cdots c_{11}\, a_{kl} \cdots a_{k1} \cdots a_{2l} \cdots a_{21} a_{1l} \cdots a_{12} a_{11}$$

Cada dato es transmitido exactamente en el mismo orden que es ingresado al registro; sin embargo, los bits de paridad son también transmitidos. El dato recibido es de nuevo almacenado en el mismo orden como en el transmisor y la decodificación correctora de errores es desarrollada. Los bits de paridad son entonces descartados y los bits de datos son desplazados fuera del registro.

Por ejemplo, un entrelazado afecta a las ráfagas de errores, por medio de un código incorporado dentro de cada palabra de código (una columna de la figura anterior 4.8 (b)) para corregir un sólo error. Además, si se supone que dentro de la ráfaga de datos transmitidos ocurre una ráfaga de ruido que afecta a l bits consecutivos codificados. Entonces, debido a la organización mostrada en la figura 4.8 (b) se observa que solamente un error aparecerá en cada una de las columnas y este único error será corregido. Si ocurren l+1 errores consecutivos, entonces una columna tendrá dos errores y la corrección no será confiable. En general, si el código es capaz de corregir t errores entonces, el proceso de entrelazado permitirá la corrección de una ráfaga de B bits con:

$$B \leq tl$$

IV.4.2. Entrelazado Convolucional

Un esquema alternativo de entrelazado es el convolucional (Figura (4.9)). Los cuatro switches (interruptores) operan a paso y se mueven línea a línea de acuerdo al índice de bit de la ráfaga de bits de entrada *d(k)*. Cada switch (interruptor) hace contacto con la línea 1 al mismo tiempo, después se mueven a la línea 2, y así sucesivamente hasta regresar a la línea 1 después de la línea l. La cascada de elementos de almacenamiento en las líneas son registros de corrimiento. Comenzando con la línea l en el lado transmisor que no tiene elementos de almacenaje, el número de los elementos se incrementa por s progresando así de línea en línea. La última línea l tiene $(l\text{-}1)s$ elementos de almacenaje. El número total de elementos de almacenaje en cada línea (transmisor más el lado receptor) es el mismo. Por lo tanto, en cada línea hay un total de $(l\text{-}1)s$ elementos de almacenaje.

$$c_1 = d_0 \oplus d_1 \oplus d_2$$
$$c_1 = d_0 \oplus d_2$$

La Tabla 4.6 es un ejemplo de la corrida de codificación usando el codificador convolucional de la Figura 4.11, donde el registro de corrimiento tiene todos sus elementos igual cero en un principio.

Intervalo de Tiempo	1	-	2	-	3	-	4	-	5	-	6	-	7	-	8	-
Bits de Entrada	0	-	1	-	1	-	0	-	1	-	0	-	0	-	1	-
Bits de Salida	0	0	1	1	0	1	0	1	0	0	1	0	1	1	1	1
Variable de Salida	c_1	c_2	c_1	c_2	c_1	c_2	c_1	c_2	c_1	c_2	c_1	c_2	c_1	c_2	c_1	c_2

Tabla. 4.6. Muestra de la corrida para el Codificador Convolucional de la Figura 4.11.

Por ejemplo, el intervalo de tiempo tres muestra los estados (K-1)(K=3) más hacia la izquierda que contienen "0 1". Un 1 que llega en el intervalo 3 produce "0 1" (c_1 c_2) que son los dos bits de salida. El codificador convolucional es como una máquina de estados finitos, moviendo desde un estado K-1 el contenido del registro de corrimiento (los últimos bits de entrada), hacia otro (Figura 4.12).

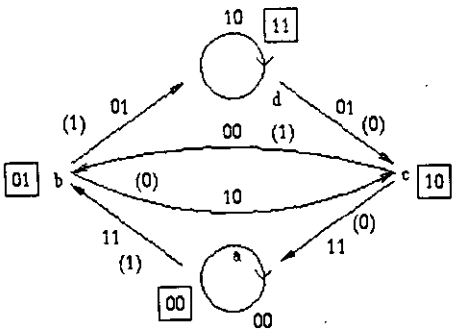


Figura 4.12. Diagrama de estados de un codificador convolucional de Rc=1/2 y K=3.

Las transiciones entre los estados son gobernadas por los bits de datos (0 ó 1) a la entrada (no codificados). La máquina de estados tiene 2^{K-1} posibles estados y en este ejemplo donde K=3, se tienen cuatro posibles arreglos de los bits en los 2 estados más hacia la izquierda del estado K en el registro de corrimiento correspondiente a los cuatro estados de la máquina de estados finitos (llamados a,b,c,d).

Conexiones de Generación de Código (M_1, M_2, . . . M_K)		
K (Número de Estados en el Registro)	C_1	c_2
3	1,1,1	1,0,1
4	1,1,1,1	1,1,0,1
5	1,1,1,0,1	1,0,0,1,1
6	1,1,10,1,1	1,1,0,0,0,1
7	1,1,1,1,0,0,1	1,0,1,1,0,1,1
8	1,1,1,1,1,0,0,1	1,0,1,0,0,1,1,1

Tabla 4.7. Velocidades de Codificadores Convolucionales de ½.

Para los códigos convolucionales las palabras de códigos son frecuentemente generadas a través de operaciones lineales. Sin embargo, generalmente los sistemas son lineales sobre $GF(2)$ y tienen memoria. Un ejemplo con dos fuentes de entradas binarias y tres salidas de bits esta dada en la Figura 4.15. Este código tiene un índice R_c=2/3. El número de elementos almacenados es dos, y como las entradas son binarias, es conocido como un *código de cuatro estados*. Los códigos convolucionales son comúnmente usados para generar códigos de enrejado.

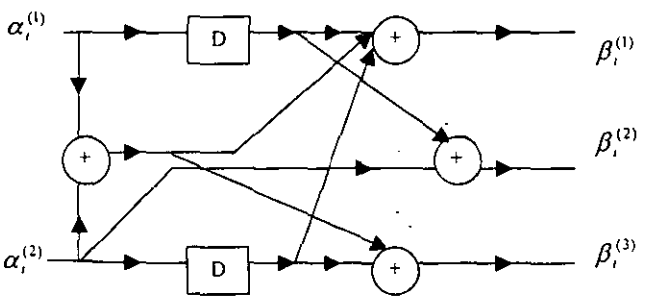


Figura 4.15. Código de 2/3.

IV.5.2. FSM y Representación de Enrejado

La máquina de estados finitos (finite-state machine FSM) es una representación de los códigos de enrejado. En la figura 4.16, las variables de estado son a=00, b=10, c=01, y d=11; $\alpha_{i-1}^{(1)}$ y $\alpha_{i-1}^{(2)}$ son los estados con las entradas $(\alpha_i^{(1)}, \alpha_i^{(2)})$. La máquina de estados

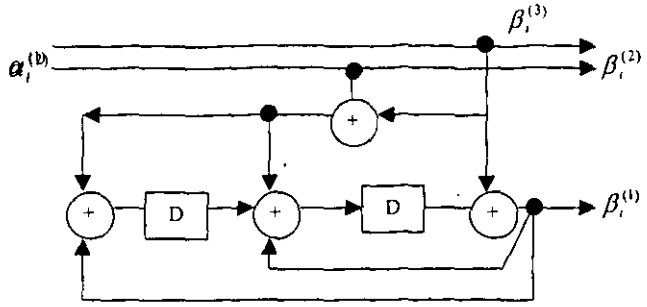


Figura 4.18. Realimentación para un código convolucional.

IV.5.4. Representación Polinomial

Un código de bloques puede ser definido por un generador polinomial *g(x)*, mientras que para un código convolucional puede ser definidó por los polinomios k_0 y n_0. Frecuentemente k_0 =1. En este caso, un código convolucional puede ser definido por la matriz generadora:

$$G(x)=[g_1(x), g_2(x), \ldots, g_{n_0}(x)]$$

donde

$$g_i(x)= g_{j0} + g_{j1}x + g_{j2}x^2 + \cdots + g_{jK-1}x^{K-1}$$

Al igual que los códigos de bloques, el problema es encontrar polinomios generadores que tengan buenas propiedades para el control de errores.

IV.5.5. Decodificación de un Código Convolucional

IV.5.5.1. El Código de Árbol

Una forma de decodificar un código convolucional es a través de un código de árbol como el de la Figura 4.19. Este código de árbol se aplica a un codificador convolucional con $K=4$ y $c=3$, para el caso $L=5$ correspondiente a una secuencia de mensajes de 5 bits.

Entonces en el diagrama de árbol, los nodos proliferan sin límite, como el árbol se expande el mismo conjunto de nodos reaparecen en el tiempo una y otra vez.

Si se comienza en un estado arbitrario, por ejemplo un estado b durante el intervalo k en la Figura 4.22. La siguiente entrada llevará al codificador sobre una de las dos ramas hacia un estado a o a un estado c, donde este resultará durante el intervalo k+1. El segundo bit llevará al codificador desde k+1 a k+2 sobre uno de las cuatro posibles ramas, dos desde el estado a y dos desde el estado c. Durante el intervalo k+2 cualquiera de los cuatro estados son posibles. Entonces el número de ramas que dejará cada estado es dos, el número de ramas desde k+2 a k+3 es 4x2=8. Entonces, el número de ramas disponibles es siempre ocho. Además, el número total de trayectorias a través del enrejado es 2x4x8x8···. A través del enrejado desde k hasta k+1 el número total de posibles trayectorias es $8^{l-1}=2^{3(l-1)}$ (excepto para l=1). El rápido incremento en las trayectorias (exponencial) es con respecto a l.

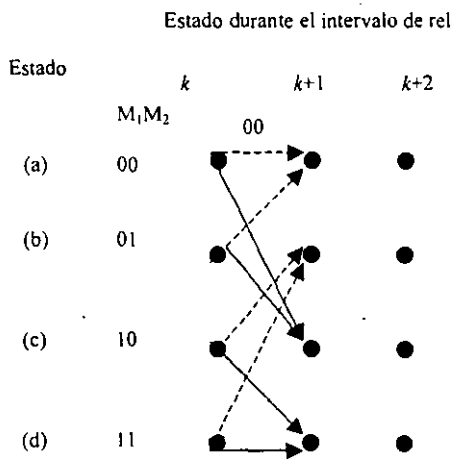


Figura 4.22. Un diagrama de enrejado para el codificador de la figura 4.20.

IV.5.5.3. Medidas de Distancias

La distancia mínima de un código convolucional es un parámetro fundamental que determina su capacidad de control de errores. La medida de distancia para los códigos convolucionales es algo más compleja que los códigos de bloques, ya que depende del número de *m* tramas usadas por el decodificador, ver Figura 4.23.

donde $\hat{e}'(x)$ es un estimado de e'(x). Si la decisión es correcta, el error es borrado. En la Figura 4.24 la decisión trata de corregir el símbolo de información recibida i_0' por sustracción el estimado $\hat{e}_0'$ del símbolo erróneo e_0'. Si el estimado es correcto, el error dentro de i_0' es eliminado y el símbolo i está presente a la salida del decodificador. Los errores dentro de la secuencia verificadora deben también ser tomados en cuenta pero estos no necesitan ser corregidos.

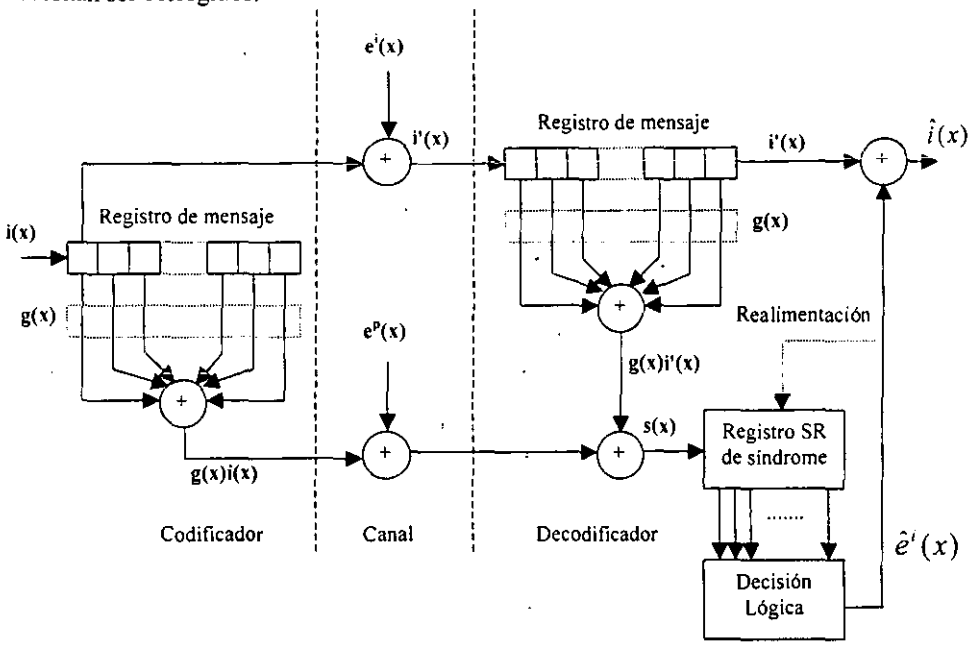


Figura 4.24. Codificador paralelo general y decodificador síndrome para un código sistemático (R=1/2, k_0=1).

IV.5.6. Algoritmo Viterbi

Es un método popular para decodificar códigos convolucionales: El decodificador Viterbi, es usado frecuentemente en TCM.

El problema de la decodificación es determinar la trayectoria transmitida a través del enrejado, por ejemplo si la trayectoria de todos los ceros a-a-a-... es enviada, se esperaría que está sea la trayectoria decodificada. Si a la salida del canal el resultado de enviar la trayectoria transmitida a través del BSC, presenta una perturbación de la trayectoria transmitida, entonces existirá un error.

embargo, el número de trayectorias se incrementa exponencialmente con la longitud de la secuencia. Por ende, el procedimiento sugerido es factible para muy cortas secuencias. Sin embargo, a través de la aplicación del algoritmo Viterbi, muchas trayectorias pueden ser descartadas resultando un procedimiento más útil.

La Figura 4.28 muestra un codificador con M_1M_2=00. Si una secuencia de cinco bits de información es ingresada al codificador, y los correspondientes bits c_{1R},c_{2R} recibidos (no los bits de salida transmitidos al codificador) son:

$$c_{1R},\ c_{2R}=10\ 00\ 10\ 00\ 00$$

Esto es para introducir por lo menos un error (y posiblemente más errores se introducen durante la transmisión). Para la Figura 4.28, en el estado *a*, si el primer bit de información fuera un 0 los primeros dos bits recibidos habrían sido c_{1R},c_{2R}=00, mientras si el primer bit de información es 1 el primero de dos bits recibidos sería c_{1R},c_{2R}=11. En otro caso c_{1R},$c_{2R} \neq 10$.

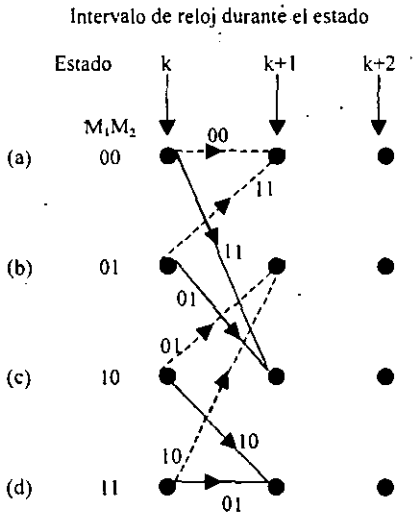


Figura 4.28. Un diagrama de enrejado para el codificador de la Figura 4.20.

Primero, se trazan las posibles rutas a través de los estados del codificador como se muestra en el enrejado de la Figura 4.29. Comenzando en el estado a, en el intervalo de reloj k=1, un 0 (línea punteada) provocará una salida 00 y llevará al codificador al estado c.

Simulaciones computacionales muestran que a una profundidad de decodificación W de 4 ó 5 veces la longitud restringida, usualmente provoca una degradación insignificante comparada con un decodificador de memoria "infinita".

IV.6. SOC's

Estos códigos generan un conjunto de ecuaciones verificadoras ortogonales. Los códigos generadores para los SOCs son fácilmente construidos desde diferencia triángulos[51]. Si la primera fila de este triángulo es de la forma d_{11}, d_{12}, d_{13},...., entonces el generador polinomial es:

$$g(x)=1+x^{d_{11}}+x^{d_{11}+d_{12}}+x^{d_{11}+d_{12}+d_{13}}+\cdots$$

Los SOCs más simples están dados dentro de la Tabla 4.8.

J	K	Primera fila del triángulo	G(x)
2	2	1	$1+x$
4	7	2,3,1	$1+x^2+x^5+x^6$
6	18	2,5,6,3,1	$1+x^2+x^7+x^{13}+x^{16}+x^{17}$
8	36	7,3,6,2,12,1,4	$1+x^7+x^{10}+x^{16}+x^{18}+x^{30}+x^{31}+x^{35}$

Tabla 4.8. SOCs para R=1/2.

IV.7. CÓDIGOS DE PRUEBA-Y-ERROR

Estos códigos son más eficientes que los SOC's, en el sentido de que alcanzan un valor dado de J y con una capacidad de corrección de errores con una longitud restringida más corta (Tablas 4.8 y4.9). Su construcción no es sencilla y son derivados por un buscador de prueba y error. La decodificación con realimentación es usada. Además, algunas veces son llamados códigos *ortogonalizables* y pueden proporcionar un conjunto ortogonal de ecuaciones de síndrome tomando combinaciones lineales de símbolos del síndrome:

[51] La diferencia de triángulos puede ser encontrada en Robinson y Bernstein, 1967; Klieber, 1970 (Signal and Coding Processing, Graham Wade, Prentice Hall).

IV.10. DECODIFICACIÓN SOFT

Una alternativa a la decodificación hard decision se utiliza para diseñar el sistema de salida multinivel del demodulador para que sea una señal cuantizada, lo cual resulta no solamente la decodificación hard decision, sino también un nivel de "confianza" dentro de la decodificación. La idea detrás de la decodificación *soft decision* es proporcionar al decodificador más información y así mejorar la recuperación de la secuencia de datos.

En la práctica, la entrada al decodificador es frecuentemente una señal de 8 niveles (3 bits) y esto puede ser representado por enteros o métricas dentro del rango 0-7, Figura 4.31(b). Una entrada de (111) al decodificador entonces denota un 1 con alta certeza, mientras una entrada de (000) denota un 0 con alta certeza. De otro modo, una entrada de (100) denota un 1 con baja confianza. Existe una distancia de unidad entre niveles adyacentes, así que un carácter (011) tiene una distancia *d*, de 3 desde (000) y 4 desde (111), como se muestra en la Figura 4.31(b). Las distancias apropiadas son sumadas para cada dígito dentro de la palabra de código recibida, para proporcionar una *distancia soft decision* total entre la palabra de código recibida y cada una de las posibles palabras de código transmitidas.

El decodificador entonces toma la mínima distancia soft decision y la salida es la palabra de código recibida (11) (a pesar del error en el último dígito). Entonces, dada una entrada (1, 0), un decodificador hard decision puede solamente *detectar* un error y no tiene forma de corregirlo.

La técnica descrita es fácilmente extendida a la decodificación Viterbi. Pero se requiere de un cambio de métrica de distancia de Hamming (para hard decision) hacia una métrica soft decision. La ventaja del desempeño de la decodificación soft decision depende por ejemplo, de las características del canal de transmisión, el número y espaciamiento de los niveles de cuantización y del algoritmo de decodificación. Para un canal Gaussiano y BER constante, 3 bits de cuantización permiten a la soft decision decoding que la E_b/N_0 (como en la Figura 4.33) sea reducida aproximadamente a 2 dB comparada con la decodificación hard decision.

energía por bit *codificado* será de $E_c=RE_b$. Este promedio para una densidad de ruido fija N_0, donde más bits serán recibidos en error a la salida del detector comparado con un sistema no codificado. Consecuentemente, el código debe tener suficiente capacidad para el control de errores para que además de compensar los errores de canal también proporcione una ganancia. Dada $E_b=E_c/R$, se tiene

$$\frac{E_b}{N_0}=\frac{E_c}{N_0}+10\log\frac{1}{R}\,dB$$

En otras palabras, al comparar los sistemas codificados y los no codificados, el índice E_c/N_0 medido sobre un canal codificado debe ser corregido por un factor $10\log(1/R)$dB para considerar los bits verificadores extra.

El uso de FEC ilustra un principio fundamental en la teoría de comunicación (el intercambio de ancho de banda por SNR). De acuerdo a la Figura 4.32, para un índice de información fijo, un índice de error particular a la salida p_d puede ser alcanzado también:

(a) incrementando la potencia de la portadora modulada, por ejemplo E_b/N_0 o,

(a) aplicando FEC, que requiere ancho de banda incrementado. Generalmente, FEC es la solución más económica.

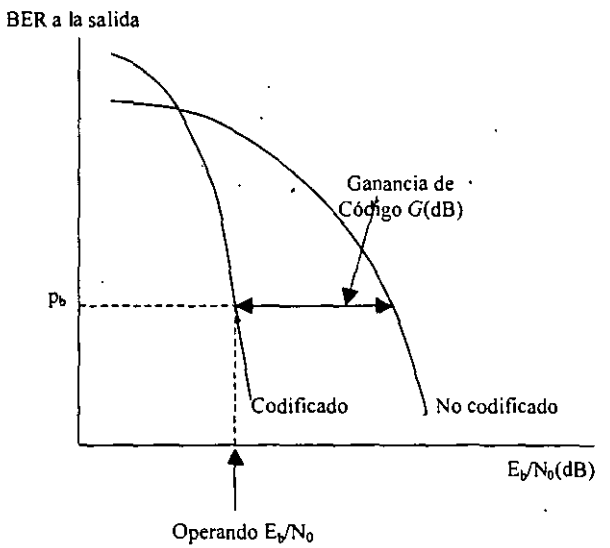


Figura 4.33. Definición de la ganancia de codificación.

CAPÍTULO 5

Turbo códigos

V.1. CÓDIGOS CONCATENADOS

Los esquemas de codificación concatenada fueron propuestos por G. D. Forney,[52] para alcanzar grandes ganancias de codificación combinando dos o más componentes relativamente sencillos o construyendo códigos de bloques. Los códigos resultantes tenían la capacidad de corregir errores más grandes que los tradicionales y su estructura permite disminuir la complejidad de la decodificación. Una concatenación serial de códigos es usada frecuentemente para sistemas de potencia limitada, como las pruebas en espacio profundo. El más popular de estos esquemas consiste de un código Reed-Solomon externo (aplicado primero, removido a final) seguido por un código convolucional interno (aplicado al final, removido primero). Un turbo código puede ser concebido como un refinamiento de la estructura de codificación y un algoritmo iterativo para decodificar la secuencia de código asociada.

El uso de códigos no binarios (códigos basados en símbolos en vez de bits individuales) es una forma efectiva de distribuir las ráfagas de errores. El código RS es eficiente y útil cuando m (la longitud del símbolo) es mucho mayor que la unidad. El código RS es importante porque dispone de un algoritmo de decodificación hard-decision muy eficiente, que hace posible emplear códigos largos. Una segunda forma eficiente para tratar las ráfagas de errores es el entrelazado. Sin embargo, los códigos no binarios y el entrelazado no son lo suficientemente eficientes en cuanto a evitar errores aleatorios que afectan a un solo bit o a un pequeño número de bits consecutivos. Cuando se presentan

[52] Investigador de los códigos correctores de errores.

Los turbo códigos están definidos en términos de sus técnicas de codificación y decodificación:

- El codificador del turbo código está basado en "concatenación paralela de dos códigos recursivos sistemáticos convolucionales".
- El codificador del turbo código está basado en "decodificación realimentada".

V.2.1. Introducción

Los turbo códigos representan el trabajo de investigación más importante en la Teoría de la Información desde que Underboeck introdujo los códigos de enrejado (trellis codes) en 1982. Así como el trabajo de Underboeck permitió a los esquemas de modulación codificada tener la capacidad de operar cerca de la capacidad de canal, el desempeño ofrecido por los turbo códigos es cercano a la máxima capacidad de canal en aplicaciones de espacio profundo y canales satélitales. La introducción de los turbo códigos involucro el restablecimiento de conceptos, algoritmos y nuevas combinaciones ingeniosas. Los principios que rodean a los turbo códigos son poco comunes y nuevos, por lo que ha resultado difícil iniciar el estudio de estos códigos.

La funcionalidad de los turbo códigos puede ser descrita como:

- Codificación de la fuente (de ráfagas de bits) con dos códigos de canal;
- Decodificación iterativa de los dos códigos;
- Durante cada iteración los dos codificadores toman ventaja de las probabilidades *a posteriori* obtenidas del paso previo de decodificación en una manera de lazo de realimentación;
- El turbo decodificador tiene las salidas de una estimación MAP (Maximun A Posterior) de las palabras de código recibidas;

Para mejorar el desempeño de los turbo códigos, existen dos métodos:

- El uso de códigos sistemáticos recursivos convolucionales;
- Entrelazado de datos entre los dos pasos de codificación.

V.2.3.2. Los Codificadores Recursivos Sistemáticos

La matriz generadora para un código recursivo convolucional con un índice de ½ tiene la forma $G_{NR}(D)=[g_1(D),\ g_2(D)]$, el codificador recursivo sistemático tiene la matriz generadora:

$$G_R(D)=\left[1\ \frac{g_2(D)}{g_1(D)}\right]$$

La secuencia de código correspondiente a la entrada del codificador $u(D)$ es $u(D)G_{NR}(D)=[u(D)g_1(D),\ u(D)g_2(D)]$, donde la secuencia de código idéntica es producida dentro del código recursivo por la secuencia $u'=u(D)g_1(D)$, entonces en este caso la secuencia de código es $u(D)g_1(D)G_R(D)= u(D)G_{NR}(D)$. El par de polinomios $u(D)G_{NR}(D)$ son una secuencia de código; además, la secuencia de código verdadera es derivada de este par polinomial.

Para el codificador recursivo, la secuencia de código será de peso finito si y solo si la secuencia de entrada es divisible entre $g_1(D)$.

Corolario 1. Una entrada de peso uno producirá una salida de peso finito (por tal, una entrada nunca es divisible por un polinomio $g_1(D)$).

Corolario 2. Para cualquier $g_1(D)$ no trivial, existe una familia de entradas de dos pesos de la forma $D^j(1+D^{q-1})$, $j\geq 0$, que producirá salidas de peso finito, por ejemplo que son divisibles entre $g_1(D)$. Cuando $g_1(D)$ es un polinomio primitivo de grado m, entonces $q=2^m$; más generalmente, $q-1$ es la longitud de la secuencia pseudoaleatoria generada por $g_1(D)$.

En el contexto del código de enrejados, el corolario 1 dice que una entrada de peso uno creara una trayectoria que diverge de la trayectoria de todos ceros, pero que nunca resurge. El Corolario 2 dice que siempre existirá una trayectoria de enrejado que diverge y resurge después que corresponde a una secuencia de datos de peso dos.

$$d_{2,min}^{TC} \geq d_{2,min}^{CC} - 2,$$

en la desigualdad anterior cuando ambos codificadores constituyentes producen palabras de código de peso $d_{2,min}^{CC}$ (menos 2, por el codificador inferior), el valor exacto de $d_{2,min}^{TC}$ depende del permutador. El número de las entradas de peso 2 que producen las turbo palabras de código de por n_2 para w=2, la suma interna en P_b es aproximadamente:

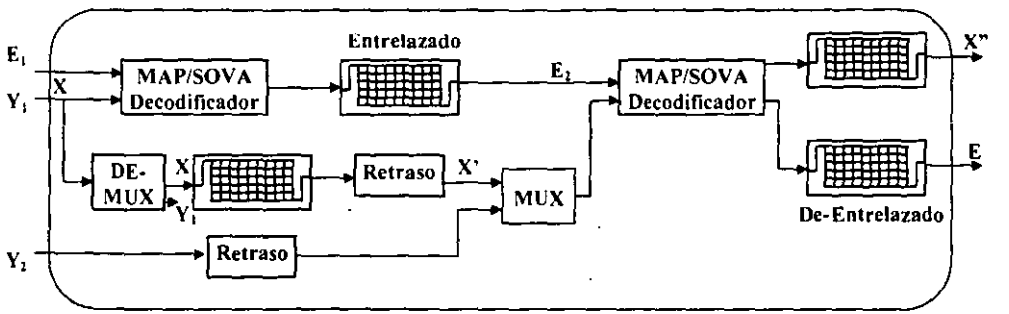
$$\sum_{v=1}^{\binom{N}{2}} \frac{2}{N} Q\left(\sqrt{\frac{2rd_{2,v}E_b}{N_0}}\right) \approx \frac{2n_2}{N_0} Q\left(\sqrt{\frac{2rd_{2,min}^{TC}}{N_0}}\right),$$

<u>w=3:</u> Siguiendo un argumento similar al del caso w=2, se puede aproximar la sumatoria interna en P_b por w=3 como:

$$\sum_{v=1}^{\binom{N}{3}} \frac{3}{N} Q\left(\sqrt{\frac{2rd_{3,v}E_b}{N_0}}\right) \approx \frac{3n_3}{N_0} Q\left(\sqrt{\frac{2rd_{3,min}^{TC}}{N_0}}\right),$$

donde n_3 y $d_{3,min}^{TC}$ están definidos. Mientras n_3 es dependiente del entrelazador, se pueden hacer algunos comentarios sobre su tamaño relativo para n_2 para un entrealzador "generado aleatoriamente". Además, existen (N-2)/3 veces tantos términos con w=3 en la sumatoria interna de P_b, como w=2 términos. Entonces, la mayoría de los $\binom{N}{3}$ términos en P_b pueden ser eliminados de la consideración de este razonamiento. Además, dada una entrada $u(D)$ en el codificador de peso tres divisible por $g_1(D)$, se vuelve muy poco probable que la entrada permutada $u'(D)$ vista por el segundo codificador, también sea divisible por $g_1(D)$. Suponiendo $u(D)=g_1(D)=1+D+D^4$, entonces la salida del permutador será un múltiplo de $g_1(D)$ si la entrada de tres 1's llega a ser $j^{ésimo}$, $(j+1)^{ésimo}$, y $(j+4)^{ésimo}$ bits fuera del permutador actúan en una moda puramente aleatoria, así que la probabilidad de que uno de los 1's consiga una posición dada es $1/N$, la salida del permutador es $D^j g_1(D)=D^j(1+D+D^4)$ con la probabilidad $3!/N^3$. Para comparar entradas w=2, una salida patrón del permutador dada ocurre con una probabilidad $2!/N^2$. Se puede esperar que el número de entradas de peso tres, n_3, resulten trayectorias remergentes en ambos codificadores por ser mucho menores que n_2:

de información hacia un segundo decodificador Viterbi, para utilizar a este último como una etapa de decodificación subsecuente. El algoritmo de Bahl, máximo a posteriori (MAP) símbolo a símbolo, publicado hace 20 años, también fue considerado como un algoritmo importante para la decodificación de los turbo códigos. Este último algoritmo, es frecuentemente llamado "algoritmo BCJR", utilizado por Berrou, en la decodificación iterativa de los turbo códigos.



X: Información sistemática X": Información decodificada
Y_1: Redundancia-Información E_i: Información -Extrinseca

Figura 5.6. Decodificación Turbo códigos (una Iteración)

V.2.11. Algoritmo BCJR Modificado

Existe una versión modificada del algortimo BCJR para mejorar el desempeño de la decodificación MAP símbolo por símbolo. Este algoritmo es incorporado dentro de un decodificador iterativo empleando dos decodificadores BCJR-MAP. Para la explicación de su desarrollo, se tiene la siguiente notación:

- E_1 es la notación para el codificador 1
- E_2 es la notación para el codificador 2
- D_1 es la notación para el decodificador 1
- D_2 es una notación para el decodificador 2
- m es la memoria del codificador constituyente
- S es el conjunto de todos los 2^m estados del codificador constituyente.

V.2.12. Decodificación MAP Iterativa

El procedimiento óptimo (con respecto a la probabilidad de error en la palabra de código) para decodificar la secuencia recibida es el desarrollo de máxima probabilidad. Esto es solamente para calcular la función de probabilidad para todas las posibles secuencias de código y elegir uno que maximice esta función. Existe por lo menos un problema con este desarrollo, la complejidad. Los códigos están propuestos para el caso de SNR bajas y en el caso general de considerable longitud, especialmente después de los procesos de concatenación y de entrelazado. El proceso de entrelazado a través de una matriz de entrelazamiento (por ejemplo, de 256 veces 256) complica la estrategia óptima de decodificación. El problema es encontrar un esquema de decodificación de baja complejidad, posiblemente iterativo que tenga un buen desempeño para diferentes valores de SNR's.

MAP, cumple con las necesidades mencionadas. El desarrollo de MAP, en contraste al desarrollo ML, minimiza la probabilidad de símbolos erróneos (o bits). Pero el procedimiento MAP es en general igual o más complejo que ML. La ventaja de la decodificación MAP es que este caso concatenado; además, de que se decrementa la probabilidad de símbolo erróneo, también que es posible explotar la estructura concatenada para obtener un estimador iterativo MAP.

MAP tiene una variante que básicamente consiste de un procedimiento iterativo de dos pasos, donde se realiza el cálculo de las máximas probabilidades a posteriori (MAP) de los símbolos. Estos dos pasos son:

- Decodificar los bits recibidos correspondientes al código exterior y al mismo tiempo calcular las probabilidades a posteriori, de los bits de información, dando la secuencia recibida.
- Se usan las probabilidades a posteriori del primer paso y la secuencia recibida correspondiente al código interno como una entrada al segundo paso y decodificar.

Estos dos pasos no son una forma óptima de decodificar el código concatenado. En el primer paso la información del código interno no es considerada, lo que provoca un procedimiento subóptimo. Sin embargo, se procede a iterar el procedimiento a dos pasos:

$$\gamma_k(s,s') \sim \exp\left[\frac{1}{2}u_k(L^e(u_k)+L_c y_k^s)+\frac{1}{2}L_c y_k^p x_k^p\right]$$
$$= \exp\left[\frac{1}{2}u_k(L^e(u_k)+L_c y_k^s)\right]\cdot\gamma_k^e(s',s)$$

donde $L_c \triangleq \frac{4E_c}{No}$ y donde:

$$\gamma_k^e(s',s) \triangleq \exp\left[\frac{1}{2}L_c y_k^p x_k^p\right].$$

Combinando $\gamma_k(s',s)$ con $L(u_k)$ se obtiene:

$$L(u_k) = \log\frac{\sum_{s^+}\tilde{\alpha}_{k-1}(s')\cdot\gamma_k^e(s',s)\cdot\tilde{\beta}_k(s)\cdot C_k}{\sum_{s^-}\tilde{\alpha}_{k-1}(s')\cdot\gamma_k^e(s',s)\cdot\tilde{\beta}_k(s)\cdot C_k}$$

$$= L_c y_k^s + L^e(u_k) + \log\frac{\sum_{s^+}\tilde{\alpha}_{k-1}(s')\cdot\gamma_k^e(s',s)\cdot\tilde{\beta}_k(s)}{\sum_{s^-}\tilde{\alpha}_{k-1}(s')\cdot\gamma_k^e(s',s)\cdot\tilde{\beta}_k(s)}$$

donde $C_k \triangleq \exp\left[\frac{1}{2}u_k(L^e(u_k)+L_c y_k^s)\right]$. En la segunda igualdad $C_k(u_k=+1)$ y $C_k(u_k=-1)$ que pueden ser factorizados fuera de las sumatorias en el numerador y denominador respectivamente. El primer término en $L(u_k)$ es algunas veces llamado el *valor del canal*, el segundo término representa cualquier información *a priori* acerca de la u_k proporcionada por un decodificador previo, y el tercer término representa la *información extrínseca* que puede ser pasada hacia un decodificador subsecuente. Así por ejemplo, en cualquier iteración dada, D_1 calcula

$$L_1(u_k) = L_c y_k^s + L_{21}^e(u_k) + L_{12}^e(u_k)$$

donde $L_{21}^e(u_k)$ es la información extrínseca pasada desde D_2 hasta D_1, y $L_{12}^e(u_k)$ es el tercer término en $L(u_k)$ que se usa como información extrínseca desde D_1 hasta D_2:

1. Colocando los bits de información dentro de un arreglo de k_1 filas y k_2 columnas,
2. Codificando las k_1 filas del código C_2,
3. Codificando las n_2 columnas usando el código C_1.

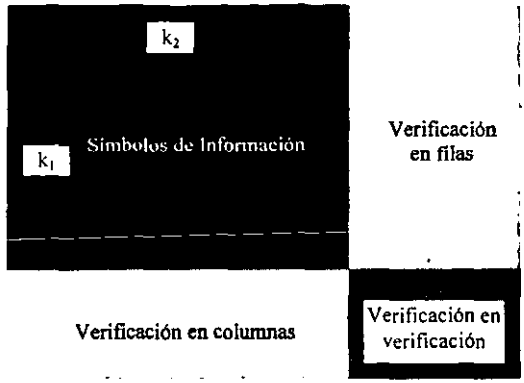


Figura 5.8. Ejemplo de un producto de código $P=C_1*C_2$

Los parámetros del producto de código P son: $n=n_1*n_2$, $k=k_1*k_2$, $d=d_1*d_2$ y el indice de código R esta dado por R_1*R_2 donde R_i es el índice de código de C_i. Así que se pueden construir códigos de bloques muy grandes con una distancia de Hamming muy pequeña por combinación de pequeños códigos con una pequeña distancia de Hamming. Dado el procedimiento usado para construir el producto de código, las últimas columnas (n_2-k_2) de la matriz son palabras de código de C_1, y las últimas filas de la matriz P son palabras de código C_2. Por ende, todas las filas de la matriz P son palabras de código de C_1 y todas las columnas de la matriz P son palabras de código de C_2.

V.4.2. Decodificación Iterativa

Considerando la decodificación de las filas y columnas de un producto de código P transmitido sobre un canal Gaussiano usando señalización QPSK. En la matriz recibida [R], correspondiente a la palabra de código transmitida [E], el primer decodificador desarrolla la decodificación soft (probabilística) de las filas (o columnas) de P usando como matriz de entrada a [R]. La decodificación Soft Input/Soft Output es desarrollada. Por substracción de

retransmisión es solamente requerida para los bloques recibidos de tipo B. Para los bloques recibidos de tipo C, el paso de decodificación con el número mínimo de bits erróneos es estimado y la decisión de decodificación es tomada en este paso de decodificación.

V.5.3. Simulación de Resultados

No existe un método analítico confiable para estimar el desempeño de los turbo códigos y la frecuencia de la ocurrencia de los bloques inestables. La única forma posible para establecer el desempeño del sistema turbo HARQ es desarrollar simulaciones por computadora.

Por ejemplo, un sistema turbo HARQ con un índice de código R_c=1/2 y transmisión sobre un canal AWGN usando BPSK, el turbo código consiste de dos códigos idénticos convolucionales recursivos, sistemáticos, de 16 estados. Con un índice de 1/2 y un entrelazador pseudo aleatorio de longitud 1024. La matriz generadora de cada código constituyente es:

$$(1,(1+D^2+D^3+D^4)/(1+D+D^4)).$$

El máximo número de pasos de decodificación está restringido hasta 40 (por ejemplo) y el número máximo de iteraciones es 20. El ARQ utilizado, cuenta con un número ilimitado de retransmisiones. La capacidad resultante y el BER alcanzado son trazados en la Figura 5.10. El eje de las abcisas es (E_b/N_0) depende de la capacidad del esquema, por ejemplo de un valor de E_s/N_0, donde E_s es la energía promedio por símbolo transmitido y N_0 la densidad espectral de ruido. Con el propósito de tener un esquema confiable de transmisión HARQ con BER<10^{-5} es posible alcanzar para E_s/N_0<-3 dB. Sin embargo, para E_s/N_0<-2 dB el número de bloques retransmitidos se incrementa rápidamente y la capacidad resultante se vuelve extremadamente baja. Dentro del rango -2 dB<E_s/N_0<-1 dB el sistema turbo HARQ muestra mejor desempeño que el clásico esquema de turbo codificación sin pérdidas sustanciales de eficiencia.

- Tanner-Wiberg-Loeliger graphs (gráficas Tanner-Wiberg-Loeliger). Estos gráficos son lineales, dependientes del esquema de codificación, diagramas de estado cíclicos; que muestran símbolos, verificadores de paridad y estados "invisibles".

Recientes investigaciones han mostrado, que simplificaciones en las estrategias y algoritmos para decodificadores de soft decision pueden reducir la complejidad de las implementaciones significantemente, sin reducir el desempeño del decodificador, como es caso del decodificador SOVA (Soft Output Viterbi Algorithm).

CAPÍTULO 6

Aplicaciones

VI.1 INTRODUCCIÓN

Las aplicaciones de los códigos correctores de errores considerados comprenden los sistemas de comunicaciones digitales y los sistemas de almacenamiento. Nuevas vertientes de éstos están siendo probadas en ambos sistemas, códigos y decodificadores en espacio Euclidiano basados en la teoría de enrejados y paquetes de esferas[55]. Los códigos y algoritmos basados en geometría algebraica y álgebra conmutativa están disponibles en los circuitos actuales, mientras que los códigos de bloques y los códigos de árbol se están mejorando para que tomen decisiones algebraicas en los canales y de este modo puedan aplicarse con mayor frecuencia.

Respecto a los sistemas de almacenamiento, la corrección de errores es usada para desarrollar altas capacidades (o densidades) de almacenamiento en dispositivos como discos magnéticos y ópticos, y cintas confiables. Los drives de discos magnéticos baratos y confiables manejan por ejemplo capacidades de 2 a 20 GigaBytes la cual no sería posible sin la corrección de errores.

La corrección de errores puede ser usada para reducir el número de componentes DRAM[56], lo cual implica la disminución del costo de los sistemas de almacenamiento y el incremento en la confiabilidad de los módulos de memoria DRAM. Además la corrección de errores puede prolongar la vida de la batería de computadoras portátiles y disminuir la potencia requerida por la batería de otros dispositivos electrónicos.

[55] Ambos métodos, enrejados y paquetes de esferas son matemáticamente complejos que no son abordados en este trabajo de tesis debido a que las bases de estas teóricas no son propias de un plan de estudios a nivel licenciatura.

[56] Memorias de Acceso Aleatorio Dinámicas (Dynamic Random Aleatory Memory).

Los módems telefónicos más utilizados actualmente transmiten a 14.4 Kbps, 28.8 Kbps, y recientemente a 56 Kbps. Como la velocidad del módem se incrementa, la detección y corrección de errores se vuelve crítica. Con la adición de poderosos códigos correctores de errores, se pueden alcanzar índices de datos de más de 100 Kbps.

En los Estados Unidos por ejemplo, se emplea el código BCH binario(48, 36, 5) que es utilizado en los canales de control para TDMA celular. Este código tiene solamente la capacidad de corregir dos errores, pero sin embargo es de rápida decodificación la cual se realiza mediante la solución de un sistema de dos ecuaciones (el síndrome).

Compañías dedicadas al desarrollo de códigos correctores de errores[59] proporcionan grupos de desarrollo de productos en está categoría con software para códigos correctores de errores, generalmente programas escritos en lenguaje "C" pueden ser diseñados de acuerdo a las necesidades del cliente para un canal particular, considerando la longitud del bloque, el tamaño del símbolo y el número de errores capaces de corregir. En tanto que los decodificadores pueden ser configurados para corregir solamente errores, borraduras o ambos, donde las borraduras son conocidas como errantes.

Las empresas dedicadas al diseño e implementación de los códigos correctores de errores, pueden optimizar el software CCE, así como minimizar el tiempo de ejecución, el número de instrucciones, o la cantidad de memoria usada. Sin embargo, de acuerdo al sistema de comunicaciones se debe considerar que la disminución del tiempo de ejecución generalmente significa aumentar la memoria y viceversa. El software puede ser sustituido por un lenguaje DSP (Digital Signal Processor, Procesador Digital de Señales) o puede ser compilado directamente para correr sobre un microprocesador de propósito general, como un Pentium.

Pero el software para los códigos correctores de errores no es lo suficientemente rápido para las necesidades de algunos canales, y de este modo las mismas compañías se ven obligadas a proporcionar diseños en hardware, realizados de acuerdo a las necesidades del cliente, escritos generalmente en Verilog[60]. Rápidos prototipos de los códigos correctores de errores en hardware pueden ser desarrollados usando PLDs (programmable

[59] Tal como LSI Techonologies y ECC Technologies.
[60] Lenguaje de programación que pertenece a la familia HDL (High Level Desing Languages).

mientras que OOB-FEC fue introducido en sistemas DWDM para permitir un mejor desempeño de las redes con múltiples canales. La misma tecnología OOB-FEC que fue desarrollada para redes submarinas es usada en DWDM de los sistemas terrestres.

Al agregar códigos FEC a los canales ópticos las portadoras pueden requerir menor número de amplificadores intermedios en las líneas de transmisión. Esta aplicación está actualmente siendo diseñada para nuevas redes DWDM, de tal modo que es posible disminuir costos en el equipo y mantenimiento del mismo.

En IB-FEC, los bits no utilizados del encabezado de la trama Sonet[63] son utilizados para agregar el código para la corrección de errores en los bytes. Esto permite a IB-FEC alcanzar una tolerancia de la relación señal a ruido sobre otras señales que no utilizan cualquier tipo de FEC. IB-FEC es vulnerable a los cambios en los estándares Sonet, debido a que utiliza los bytes de reserva de estos estándares. Si los estándares Sonet cambian, y requieren alguno de los bytes útiles para la corrección de errores (no utilizados por las presentes versiones de Sonet) para nueva información, entonces la capacidad de corregir errores de IB-FEC sería menor porque ya no tendría tantos bytes disponibles para realizar sus operaciones.

OOB-FEC tiene una capacidad más poderosa para corregir errores que IB-FEC. Mientras IB-FEC solamente permite corregir ocho errores por trama, OOB-FEC permite hasta 1024 correcciones de errores por trama. OOB-FEC no usa cualquiera de los bytes del encabezado de Sonet como IB-FEC, sino que utiliza bits adicionales que son agregados a la señal transmitida del Sonet.

OOB-FEC incrementa el índice de bit debido a que agrega bits a la señal transmitida. Además, no es susceptible a los cambios en los estándares Sonet.

OOB-FEC puede proporcionar administración del desempeño en tiempo real para portadoras, lo cual permite observar como esta trabajando el código FEC. Por ejemplo, si hay un incremento en el índice de corrección de errores el operador podrá detectar si existe algún problema en la red y puede tomar la acción correctiva antes de que los clientes vean tal problema.

[63] Estándar para la sincronización de redes ópticas (Synchronous Optical Network).

2. Cuando los datos se necesitan leer, el código almacenado se calcula de nuevo usando el algoritmo original. El código resultante es comparado con el código generado cuando la palabra fue almacenada.

3. Si los códigos corresponden, el dato esta libre de errores y es enviado.

4. Si los códigos no corresponden, la pérdida de bits o los bits erróneos se determinan a través de un código de comparación y el bit o los bits son proporcionados, o bien, corregidos, según el caso.

5. Si la prueba no corrige los datos almacenados, se aplicará la prueba sobre nuevos datos y si los errores fueron transitorios, los bits incorrectos se eliminan.

6. Cualquier error que reincide en el mismo lugar de almacenamiento después de que el sistema ha sido reinicializado, indica un error permanente en el hardware y un mensaje es enviado para un truncamiento o para un sistema administrador, indicando la localización con los errores recurrentes.

Los códigos Reed-Solomon son comúnmente implementados; ya que permiten detectar y restaurar bits "borrados" así como bits incorrectos.

Los errores son corregidos "durante el vuelo" y los datos corregidos son colocados de nuevo en memoria. Si el mismo dato se lee nuevamente como corrompido, el proceso de corrección se repite; y si los errores ocurrieron debido a eventos aleatorios y no es un defecto de la memoria, la dirección de la memoria será limpiada del error cuando los datos se sobre escriban con otros.

Generalmente, cuando los datos requieren corrección, el sistema operativo trunca el error y lo reporta al administrador del sistema. Los errores múltiples pueden ser reportados por la misma localidad de memoria, si el dato es leído más de una vez sin ser reemplazado por datos diferentes. Si la misma localidad de memoria es corregida después de que un sistema se apaga, es sumamente probable que un defecto se presente en la memoria y deba ser reemplazada.

inventó un eficiente algoritmo para decodificar el código Reed-Solomon. El algoritmo de Berlekamp fue usado por el Voyager II y es la base de decodificación de los reproductores de CD. Algunos discos compactos, por ejemplo, usaron una versión llamada código CIRC (cross-interleaved Reed-Solomon code).

Reed, es actualmente profesor de ingeniería eléctrica de la Universidad del Sur de California, y continua trabajando en problemas de la teoría de codificación. Solomon, recientemente retirado de la compañía Hughes Aircraft, realiza consultas para el Jet Propulsion Laboratory.

Estos códigos en particular son un sistema de codificación basado en grupos de bits, tales como los bytes. Tal característica hace que los códigos Reed-Solomon sean particularmente buenos tratando con ráfagas de errores. Seis errores consecutivos por ejemplo, pueden afectar a más de dos bytes, así que una versión de corrección de errores dobles de un código Reed-Solomon puede proporcionar un adecuado factor de seguridad. Actuales implementaciones de los códigos Reed-Solomon en tecnologías de CD están disponibles para hacer frente a las ráfagas de errores tan frecuentes como 4000 bits consecutivos.

Algunos investigadores opinan que se ha resuelto la pregunta de la puesta en práctica e importancia de los códigos Reed-Solomon: "Es claro que ellos son prácticos, porque todo el mundo los usa ahora"[68]. Billones de dolares en moderna tecnología dependen de ideas que contienen el trabajo original de Reed y Solomon. Pero todos concuerdan en que el documento de Reed y Solomon, "ha sido un documento extraordinariamente influyente."[69]

[68] Berlekamp.
[69] McEliece.

VI.6.2. Aplicaciones Multicast

El principal campo de aplicación de la codificación redundante es probablemente en las aplicaciones multicast. Los receptores pueden perder paquetes y no asegurar una vía fiable para repararlos individualmente, pues llegaría a ser extremadamente costoso. Sin embargo, reducen la necesidad de sostener un canal de realimentación desde los receptores. El reducir la cantidad de realimentación les permite a los protocolos escalar a muchos de receptores.

Las aplicaciones que no dependen de una confiable liberación, se pueden aún beneficiar con una codificación redundante, porque una fiabilidad mejorada en la transmisión permite a las técnicas de codificación ser más dinámicas, que pueden proporcionar un uso más efectivo del ancho de banda disponible.

Una lista de aplicaciones multicast que se beneficiaría con el uso de codificación redundante es la siguiente:

- **Herramientas de videoconferencia**. Una codificación redundante con pequeños valores de k y de n-k puede proporcionar una efectiva protección contra las pérdidas en aplicaciones de videoconferencia. Para reducir el índice de perdidas efectivas se puede también utilizar una técnica de codificación más eficiente que proporcione una basta reducción en el ancho de banda. El grupo PET de Berkeley[75] ha hecho algo similar para el vídeo MPEG.

- **Fiable multicast para groupware**. Una codificación redundante puede ser usada para reducir las retransmisiones en aplicaciones que necesitan fiables multicast. Un ejemplo está dado por el tipo de aplicaciones "red whiteboard", donde la transferencia fiable se requiere para los objetos, tales como archivos Postscript o dibujos.

- **Transferencia de archivos uno-a-muchos en LANs (FTP multicast)**. Los salones de clases con estaciones de trabajo, frecuentemente utilizan este patrón de acceso a archivos, además en el proceso de booting (todos los nodos descargan del kernel[76] o ponen en marcha archivos desde un servidor) o durante las clases (donde los estudiantes descargan casi simultáneamente los mismos documentos o aplicaciones desde un servidor centralizado).

[75] http://www.icsi.berkeley.edu/PET/icsi-pet.html

[76] Parte principal del código del sistema operativo, el cual se encarga de controlar y administrar los servicios y peticiones de recursos y de hardware con respecto a uno o varios procesos.

pueden agregar verificaciones en otras direcciones ortogonales (diagonales) para incrementar la capacidad de corrección de errores del código, sin embargo, se reduciría el índice de código.

En la Figura 6.1, se muestra un código RAC 3x3 como un ejemplo de los arreglos de código. Este código corrige un solo error y detecta dos errores. Los bits de paridad de las filas y columnas en la Figura 6.1(b) son generados desde los bits de información de la Figura 6.1(a). En general, la verificación de paridad está basada en el conjunto de los bits de paridad de la fila y la columna. Los bits de paridad y los bits de información combinados en la Figura 6.1(b) forman el código de bloque almacenado dentro de la memoria óptica. Cada fila y cada columna de este bloque codificado tiene un número par de bits "encendidos" que indican la paridad par. La Figura 6.1(c) muestra la ocurrencia de un error en el bloque de datos recuperado. Para decodificar el código de bloques de la Figura 6.1(c), se calculan las paridades de las filas y columnas como se muestra en la Figura 6.1(d). El resultado de las posiciones del síndrome para la localización exacta del bit erróneo, puede proporciona solamente la detección de un error. Una página sin errores tendrá todos los bits del síndrome apagados. Cuando un código SPA detecta un error puede entonces desechar el bit erróneo y pasar los bits de información corregidos al anfitrión electrónico. Si dos errores ocurren, todo el síndrome de paridad esta apagado mientras uno o más de los bits del síndrome de la fila y columna están encendidos, indicando que la capacidad de corrección del código ha sido excedida. En este caso, la página de datos debe ser leída desde la memoria.

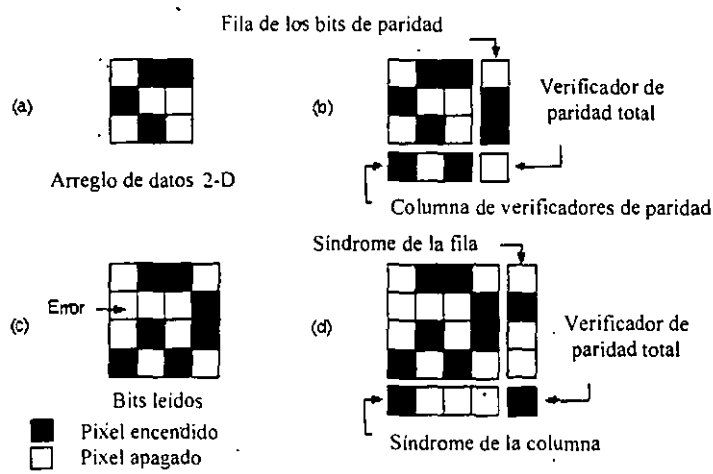


Figura 6.1. Ejemplo del código de paridad para un arreglo de bits de datos 3x3.

VI.7.5. Código Multibloque Fila y Columna

El esquema de codificación MRAC ha sido propuesto para el almacenamiento de hologramas en una base de datos relacional. El ejemplo general de MRAC se observa en la Figura 6.3, donde se muestra una página $p_1\ p_2$ dividida en bloques de código m_1 y m_2. Cada bloque puede corregir un error y detectar dos. Como el tamaño del bloque de código disminuye, el número de errores aleatorios distribuidos que pueden ser corregidos se incrementa. Cada bloque de código está codificado de acuerdo al método representado en la Figura 6.1.

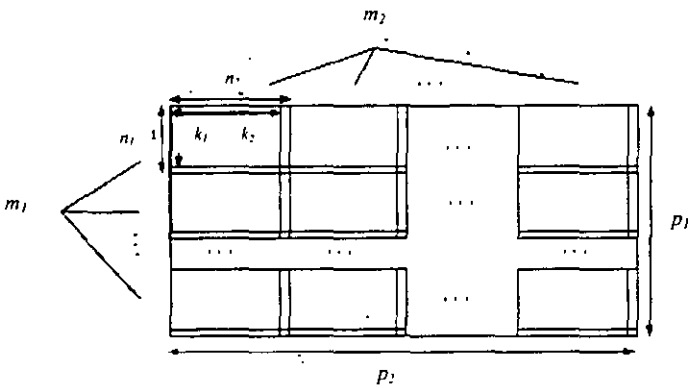


Figura 6.3. Código multibloque fila y columna.

El SPA decodifica los códigos de bloques en paralelo, razón por la cual desarrolla un alto nivel verificador de los síndromes de paridad y determina si la capacidad de los códigos ha sido excedida. El número de retrasos de las compuertas, D, para esta arquitectura esta dada por la ecuación:

$$D = n_1 + n_2 + \frac{1}{2}(m_1 + m_2 - 2)$$

donde n_1 es el número de pixeles en una columna de un código de bloque, n_2 es el número de pixeles en una fila de un código de bloque, m_1 es el número de códigos de bloques en una columna de la página de datos, y m_2 es el número de códigos bloques en una fila de la página de datos. Los términos tercero y cuarto son multiplicados por ½ porque una compuerta de 4 entradas puede ser usada para el nivel más alto del árbol de paridad.

CONCLUSIONES

Los códigos correctores de errores se aplican en los sistemas de comunicación y de almacenamiento de datos. Si se considera el problema del diseño de los sistemas de comunicación, se puede describir brevemente como sigue. Primero se desea eliminar toda la redundancia desde la fuente de información (la cantidad de datos a ser transmitidos es minimizada) y también se requiere una comunicación con seguridad que tenga el menor gasto de energía de la señal. Dos parámetros importantes son la velocidad de la información R_s de la fuente (el número mínimo de bits por segundo necesarios para representar la salida de la fuente) y la capacidad de canal C (la máxima velocidad a la que la información puede ser transmitida a través del canal y recibida confiablemente). El teorema de codificación de canal proporciona el importante resultado donde menciona que si la velocidad de la fuente R_s no excede la capacidad de canal C, es posible enviar la información de la fuente con una baja probabilidad de error. Esto es de hecho lo que la teoría de la información propone que se realice, si se desea lograr una transmisión de información de forma segura, con el menor gasto posible de la energía en la señal.

Estos códigos pueden auxiliar en las tareas de diseño de sistemas de comunicaciones, cuyo objetivo es el de proporcionar un sistema costo-eficiencia para realizar transmisiones de información de acuerdo a las necesidades del usuario. Los otros parámetros clave de diseño de un sistema de comunicaciones, son el ancho de banda de transmisión, la potencia de la señal y la complejidad de la implementación elegida. El índice de transmisión de información y la seguridad de la información enviada están típicamente determinados por los requerimientos del usuario.

digitales e industrias de procesamiento de señales. Su objetivo esta en el desarrollo, modelado, e implementación de inovados algoritmos digitales y técnicas para conocer las demandas de crecimiento de la industria de las comunicaciones digitales. Además, cuenta con expertos especializados y tecnología de punta dentro del área de los códigos correctores de errores.

Y aunque cuando se inicio el estudio de los códigos correctores de errores, algunos investigadores creyeron que habían encontrado aplicaciones en otros campos muy distintos al campo de las telecomunicaciones, incluso algunos han llegado a afirmar que estos códigos se encuentran dentro del DNA y que por lo tanto pueden corregir algunas distorsiones formadas en las proteínas y aminoácidos.

Esto último da pauta a que estos códigos pueden ser desarrollados para formar sistemas de comunicación y almacenamiento "casi perfectos".

GLOSARIO

Ancho de banda: Su unidad es el hertz (Hz). Puede ser definido como el rango entre las frecuencias más alta y más baja, usadas para una aplicación particular. Una de las definiciones más comunes se debe a la potencia de la señal. Para filtros, atenuadores, amplificadores, líneas de transmisión y otros equipos electromagnéticos pasivos y activos, estos límites generalmente los toman de una señal por debajo de los 3dB, abajo del nivel de potencia promedio en un filtro pasabanda, o bajo el nivel de una frecuencia de referencia. El ancho de banda de un medio de transmisión está determinado por el medio en sí mismo.

Baudio: Es una medida de la capacidad de transmisión de información. Su nombre se debe a Emile Baudot, quien es considerado el padre de la telegrafía automática. Los baudios y bits por segundo son sinónimos solamente en el dominio binario. En el dominio *M*-ario no son sinónimos. Por lo tanto, el baudio es la medida de la velocidad de señalización. En otras palabras, esto es una medida de las transiciones por segundo. El número de transiciones por segundo determina el ancho de banda.

BER: (Bit error rate). Es el índice del número de bits recibidos incorrectamente con respecto del total de número de bits enviados.

Borradura: Es un bit que de algún modo está perdido en el canal a lo largo del camino. Una borradura es un bit recibido que no puede ser interpretado como un 1 o un 0.

Browser: Herramienta de acceso universal. Buscador único y de bajo coste para los datos de la red. Es una parte del software que actúa como una interface entre el usuario y el encargado interno de Internet, específicamente la World Wide Web. También se les conoce como clientes web, o clientes Universales, porque en el modelo cliente/servidor, el navegador funciona como el programa cliente. El navegador actúa como intermediario del usuario. El navegador contacta a un servidor web y envía un pedido de información, recibe la información y entonces la expone en el ordenador del usuario. Se pretende que en el futuro, el browser —con interfaz de usuario y capacidades generales extendidas por Java, ActiveX y otras tecnologías— se convierta en un compañero estratégico en el proceso de distribución de información sobre la administración de la red.

TCM: Es la combinación de QAM y codificación convolucional.

TDMA: Técnica de acceso múltiple, donde múltiples usuarios de comparten uncanal en el dominio del tiempo. Los mensajes son siempre digitales y transmitidos en tramas. Esta técnica es utilizada ampliamente en sistemas celulares y satelitales.

Transductor: es un dispositivo para convertir una señal no eléctrica, tal como el sonido, luz, calor, etc., a una señal eléctrica, o viceversa. Por ejemplo, los micrófonos y las bocinas son transductores electroacústicos. Un transductor activo es aquel que puede por sí mismo introducir una ganancia en potencia y tiene su propia fuente de energía (potencia). Un transductor pasivo no tiene fuente de potencia y no puede introducir ganancia a la señal actuante.

Vibración simpática: vibración de la materia debida a la aplicación de la frecuencia de resonancia de la misma

Whiteboard: El whiteboard es una herramienta compartida, interactiva y multiusuario. Este incluye un drawing space y una caja de "chat". Se utiliza cuando dos o más personas desean discutir interactivamente; por ejemplo, matemática, tutoriales, o enseñar un concepto de matemáticas a través de la Internet. Los whiteboards permiten a un documento o a una imagen ser vistos simultáneamente por dos o más participantes en una conferencia. Todos los participantes en la conferencia pueden agregar a la imagen o documento asíncronamente y compartir ideas interactivamente. Esta imagen es el equivalente electrónico del pizarrón blanco físico. Los whiteboards están caracterizados por:

- Ser una sencilla herramienta de edición para gráficos 2D,
- Puede ser compartido por todos los participantes en la conferencia,
- Es de acceso asíncrono,
- Permiten a los participantes compartir ideas interactivamente.

BIBLIOGRAFÍA

Algebraic Coding Theory.
Autores varios.
Ed. McGraw-Hill.
Estados Unidos de América, 1984.
474 pp.

Digital Transmission Theory.
BENEDETTO, Sergio.
Prentice-Hall, Inc.
Estados Unidos de América, 1987.
639 pp.

Introduction to Trellis-Coded Modulation with Applications.
BIGLIERI, Ezio.
McMillan Publishing Company.
Estados Unidos de América, 1991.
548 pp.

La Biblia del Multimedia.
BURGER, Jeff.
Ed. Addison-Wesley Iberoamérica.
615 pp.

Sistemas de Comunicación.
LATHI, B.P.
McGraw-Hill.
México, 1986.
703 pp.

Digital Communication.
LEE, Edward A.
Kluwer Academic Publishers.
Estados Unidos de América, 1996.
893 pp.

APÉNDICES

APÉNDICE B

Los códigos correctores de errores presentan la ventaja de reducir el costo en la potencia consumida por el sistema de comunicación. La cantidad en la que reducen la potencia del sistema depende del ruido del canal, la longitud del bloque de código y el índice de código que se utilice. Además del tipo de código que es usado y de la confiabilidad que requiera el sistema.

En ésta sección se muestra una gráfica con diferentes ahorros de potencia en decibeles para algunos códigos, que se considera que tienen un buen desempeño