

Estudio y simulación de turbo códigos utilizando el algoritmo MAP y SOVA

Vallejo Huanga Diego Fernando, Álvarez Rueda Robin Gerardo

Ingeniería Eléctrica y Electrónica, Escuela Politécnica Nacional

Quito, Ecuador

diegovallejo1985@gmail.com

arobin7es@yahoo.es

Abstract- This paper presents at first instance, a general analysis of digital communication systems, channel communications, information coding theory and the main Shannon's theorems, in order to explain the operation of the schemes of detection and correction errors of which are part Turbo Codes. After examining the development of the computational tool for simulation of Turbo codes made in MATLAB, and run tests of the program. Attached conclusions and recommendations to improve and expand the project.

I. INTRODUCCIÓN

El presente trabajo presenta en primera instancia, un análisis general de los sistemas de comunicación digital, el canal de comunicaciones, la teoría de codificación e información y los principales teoremas de Shannon, para poder explicar el funcionamiento de los esquemas de detección y corrección de errores, de los que son parte los Turbo Códigos. Luego se analiza el desarrollo de la herramienta computacional para la simulación de los Turbo Códigos realizada en MATLAB, y se ejecutan pruebas del programa. Se adjuntan conclusiones y recomendaciones para mejorar y ampliar el proyecto.

II. SISTEMAS DE COMUNICACIÓN

A. Sistemas de Comunicación Digital

Las operaciones básicas de procesamiento de señales en un sistema de comunicación digital son: codificación de la fuente, codificación de canal y modulación digital en el lado transmisor, como se muestra en el diagrama de bloques de la Figura 1 [1], así como los procesos inversos en el lado receptor. Los códigos para detección y corrección de errores corresponden a la codificación de canal.

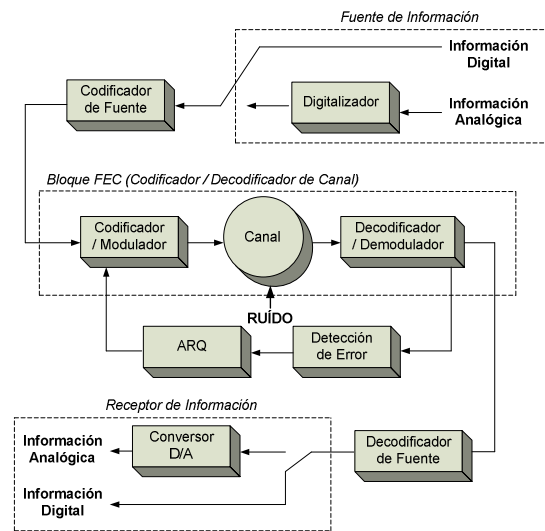


Fig. 1 Diagrama de bloques de un sistema de comunicación digital.

Uno de los parámetros que miden el desempeño de un sistema de comunicación digital es la tasa de bits errados (BER) para un cierto valor de la relación de energía de bit a densidad espectral de ruido (E_b/N_o). Generalmente, debido al ruido del canal de transmisión, la única manera de tener un valor de BER aceptable es agregando códigos para detección y corrección de los errores.

B. Canal de Comunicaciones AWGN (Additive White Gaussian Noise Channel)

En este modelo, la potencia del ruido posee una densidad espectral uniforme (ruido blanco), el cual se agrega a la señal original. La distribución del ruido resultante es un proceso gaussiano con media nula. Esto simplifica la matemática asociada con la estimación del rendimiento de un sistema de comunicación. La mayoría de las curvas de tasa de error (BER) se generan con la asunción de que el canal es con ruido gaussiano.

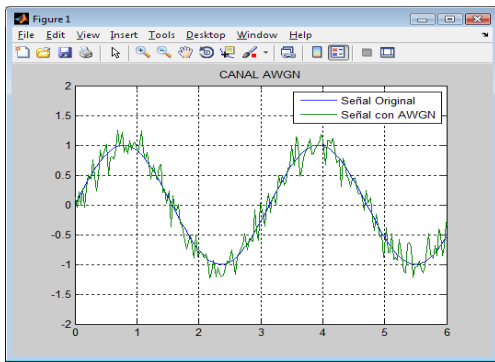


Fig. 2 Ruido AWGN agregado a una señal.

La Figura 2 muestra el uso de dicho canal con una señal sinusoidal, tomando en cuenta que el programa computacional Matlab posee una sub-función *awgn*, que simula este tipo de canal, tomando como entrada la S/N (o SNR en inglés). Mientras que la Figura 3, muestra el rendimiento de este mismo canal AWGN cuando no se ha usado una técnica de control de errores.

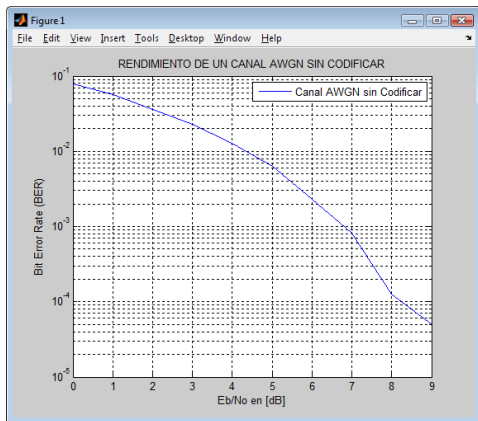


Fig. 3 Rendimiento de un canal AWGN sin codificar.

C. Teoremas de Shannon

1) *Teorema de Codificación de Fuente*: Estudia técnicas para convertir de manera eficiente fuentes arbitrarias de información en mensajes digitales, eliminando tanta redundancia como sea posible de la fuente de información [2].

2) *Teorema de la Capacidad del Canal*: A los principales parámetros de un sistema de comunicación, Shannon añadió el nivel de la potencia del ruido presente en un canal de comunicación denominado *Capacidad del Canal* "C" que define la máxima tasa de transmisión a la que aún es posible lograr que la información transmitida por un canal sea recibida por un usuario de manera confiable. Si la tasa de transmisión de un sistema permanece por debajo de C, entonces la probabilidad de error en la información transmitida podía hacerse muy baja empleando señales de transmisión codificadas suficientemente largas; por otro lado, si la tasa de transmisión sobrepasaba a la capacidad del canal entonces

no era posible lograr una transmisión confiable de información por ningún medio [2].

El modelo del canal continuo AWGN de banda limitada, es el tipo de canal que modela muchos canales de comunicación inalámbricos, incluyendo los enlaces de radio de línea de vista y de satélites. La capacidad del canal continuo AWGN de ancho de banda limitado está dada por:

$$C = W \log_2 \left(1 + \frac{S}{N} \right) \quad \left[\frac{\text{bits}}{\text{seg}} \right]$$

$$S = E_b R_s \quad N = N_o W$$

3) *Teorema de Codificación del Canal*: Para transmitir información con un nivel de confiabilidad tan alto como se desee, Shannon realizó uno de los enunciados más relevantes:

"Si se toma secuencias largas en aumento de dígitos de fuente, y se las proyecta en formas de onda de transmisión correspondientemente largas, entonces la tasa de error en la información transmitida puede hacerse arbitrariamente cercana a cero mientras no se pretenda transmitir información a una tasa superior a la capacidad del canal C. Por lo tanto, a cualquier nivel distinto de cero de la relación Señal a Ruido (S/N) del canal, existe alguna tasa de transferencia de información distinta de cero bajo la cual puede conseguirse en principio una comunicación arbitrariamente confiable". Este resultado es el Teorema de Codificación del Canal de Shannon [2].

4) *Límite de Shannon*: Se define un sistema ideal como aquel cuya tasa de transmisión R_s es igual a la capacidad del canal C. Para un sistema de este tipo, la ecuación puede escribirse como:

$$\frac{(E_b)_{\min}}{N_o} = \frac{2^{C/W} - 1}{C/W}$$

Donde $(E_b)_{\min}$ representa la mínima energía requerida por bit de información transmitido para una comunicación confiable. A la razón R_s/W se le conoce como Eficiencia del Ancho de Banda, y una gráfica de R_s/W en función de E_b/N_o se denomina Diagrama de Eficiencia del Ancho de Banda [2].

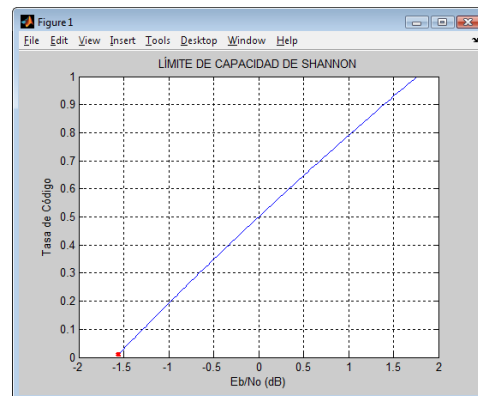


Fig. 4 Límite de capacidad de Shannon.

III. ESQUEMAS DE CODIFICACIÓN Y TURBO CÓDIGOS

A. Esquemas de Codificación BEC

El *Automatic Repeat Request (ARQ)* o también llamada *Backward Error Correction (BEC)* es una estrategia para control de errores en la que el nivel deseado de precisión en la información recibida se consigue mediante retransmisiones del mensaje enviado, dado que se emplean esquemas de codificación que son capaces de detectar errores, pero no de corregirlos. En un sistema ARQ, cuando se detectan errores en el receptor, se envía una solicitud de retransmisión del mensaje al transmisor, y esto se repite hasta que el mensaje es recibido correctamente por el receptor. Existen dos tipos de sistemas ARQ en general: ARQ Stop and Wait y ARQ Continuous.

Los esquemas de codificación más empleados por los sistemas ARQ son los Códigos de Paridad y los Códigos de Redundancia Cíclica (CRC – Cyclic Redundancy Check codes) [2].

B. Esquemas de Codificación FEC

FEC (Forward Error Correction) [3] es un tipo de mecanismo de corrección de errores que permite su corrección en el receptor sin retransmisión de la información original. Se utiliza en sistemas sin retorno o sistemas en tiempo real donde no se puede esperar a la retransmisión para mostrar los datos. [4]La posibilidad de corregir errores se consigue añadiendo al mensaje original unos bits de redundancia. La fuente digital envía la secuencia de datos al codificador, encargado de añadir dichos bits de redundancia. A la salida del codificador se obtiene la denominada palabra código. Esta palabra código es enviada al receptor y éste, mediante el decodificador adecuado y aplicando los algoritmos de corrección de errores, obtendrá la secuencia de datos original. Los dos principales tipos de codificación usados son:

1) *Códigos de Bloque*: La paridad en el codificador se introduce mediante un algoritmo algebraico aplicado a un bloque de bits. El decodificador aplica el algoritmo inverso para poder identificar y, posteriormente corregir los errores introducidos en la transmisión. Estos códigos se pueden clasificar como se muestra en la Figura 5:

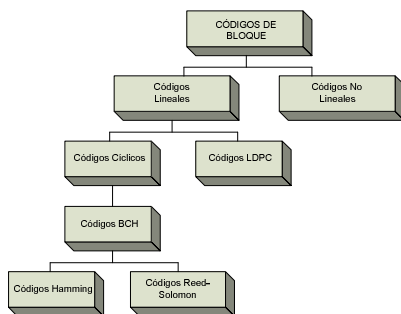


Fig. 5 Clasificación de los códigos de bloque [5].

La Figura 6 muestra el rendimiento de un código de Hamming (tipo de código de bloque) en un canal AWGN.

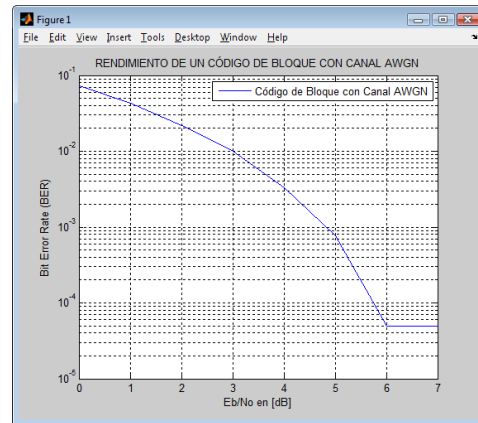


Fig. 6 Rendimiento de un código de Hamming sobre canal AWGN.

2) *Códigos Convolucionales*: Los bits se van codificando en forma serial, además poseen una cantidad de memoria para estimar la secuencia de datos más probable para los bits recibidos; por lo que son más eficientes que los códigos de bloque. La decodificación para este tipo de código es compleja ya que en principio, es necesaria una gran cantidad de memoria para estimar la secuencia de datos más probable para los bits recibidos. En la actualidad se utiliza para decodificar este tipo de códigos algoritmo de Viterbi, por su gran eficiencia en el consumo de recursos.

Un *Código Convolutivo Sistemático* es aquel en que la secuencia de información es contenida sin ser modificada, en la secuencia codificada (palabra código) producida por el codificador. Mientras que un codificador convolutivo es *Recursivo* si existe retroalimentación de los registros del codificador hacia la entrada del mismo. Los Turbo Códigos emplean códigos convolucionales sistemáticos recursivos en su estructura [2].



Fig. 7 Comparación de rendimiento de un canal AWGN sin codificar, un código de bloque y un código convolutivo sobre canal AWGN.

C. Los Turbo Códigos

Los Turbo Códigos son un método de corrección de errores basado en los códigos convolucionales más intercalación y realimentación. Consiste en una estructura

de codificación concatenada más un algoritmo iterativo; estos fueron introducidos en 1993 por Berrou y Glavieux en la conferencia internacional de la IEEE en Ginebra, Suiza. El esquema propuesto en dicho trabajo alcanzaba un BER de 10^{-5} usando una tasa de codificación de 1/2 sobre un canal AWGN, modulación BPSK con una relación E_b/N_o de 0.7 dB, lo cual está cercano al límite de Shannon que es 0.1dB [6]. Las tres áreas clave en que los Turbo Códigos proveen un mejoramiento en el desempeño son [7]:

- **Capacidad:** los Turbo Códigos logran un desempeño cercano a los límites teóricos de la capacidad);
- **Eficiencia en el Costo del Sistema:** un usuario es capaz de enviar la misma cantidad de información empleando únicamente la mitad del ancho de banda y;
- **Número de Usuarios:** un proveedor de servicio satelital es capaz de duplicar el número de usuarios sin incrementar la capacidad del satélite.

1) **Arquitecturas del Codificador:** [2]La combinación de codificadores RSC se conoce como *concatenación* y para codificadores convolucionales puede realizarse de las siguientes maneras:

- **Concatenación Paralela de Códigos Convolucionales (PCCC):** es la más usada por los Turbo Códigos y recibe ese nombre debido a que la misma secuencia de información es codificada dos (o más) veces, en paralelo, utilizando las secuencias normal y permutada(s) de los bits de información.

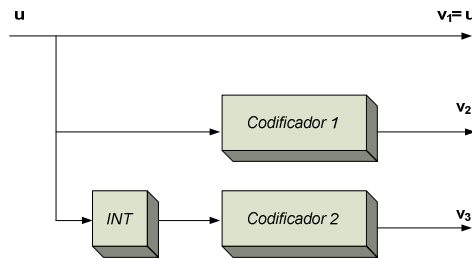


Fig.8 Arquitectura PCCC Básica.

- **Concatenación Serial de Códigos Convolucionales (SCCC):** en la Figura 9 el codificador 1 se conoce como *Codificador Externo* y el codificador 2 como *Codificador Interno*. La presencia del *interleaver* entre ambos codificadores convolucionales, provoca que este dispositivo produzca efectos distintos en el desempeño de las arquitecturas PCCC y SCCC. Una característica de los esquemas PCCC es que presentan un mejor desempeño que las arquitecturas SCCC a bajas SNR's; sin embargo, a SNR's elevadas los esquemas SCCC superan en desempeño a las arquitecturas PCCC. El tamaño y diseño del *interleaver* definen el punto de la SNR en que el desempeño de una

estructura PCCC deja de ser mejor que el de una estructura SCCC.

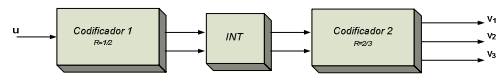


Fig. 9 Arquitectura SCCC, Tasa $r = 1/3$.

- **Concatenación Híbrida de Códigos Convolucionales (HCCC):** es una combinación de las concatenaciones paralela y serial, como se muestra:

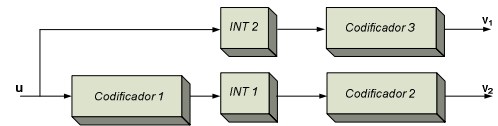


Fig. 10 Arquitectura HCCC de tasa $r = 1/2$.

Esta arquitectura es un mejoramiento definitivo de la estructura SCCC.

2) **Algoritmos de Decodificación SISO:** [2] Existen dos categorías importantes de algoritmos de decodificación SISO:

- Los algoritmos que minimizan la probabilidad de error de símbolo y;
- Los algoritmos de decodificación de máxima probabilidad, que minimizan la probabilidad de error de palabra o secuencia.

A la primera categoría pertenece el algoritmo *MAP* (*Maximum A Posteriori*) y a la segunda el algoritmo *SOVA* (*Soft Output Viterbi Algorithm*), ambos algoritmos muy utilizados en Turbo Decodificación.

La complejidad del algoritmo MAP se debe a que debe efectuar una enorme cantidad de multiplicaciones y adiciones por cada estimado que produce la probabilidad a-posteriori de cada bit de información decodificado. Este problema se aminora si el algoritmo entero se lleva a cabo en el dominio logarítmico, en vez de esperar hasta el último paso para tomar el logaritmo de la LLR. El principal beneficio es que, en el dominio logarítmico la multiplicación se convierte en suma. Esta es la idea principal de los algoritmos Max-Log-MAP y Log-MAP para reducir la complejidad.

La diferencia entre los algoritmos Max-Log-MAP y Log-MAP radica en la forma en que efectúan la operación de suma en el dominio logarítmico. En el dominio logarítmico, el algoritmo Max-Log-MAP efectúa la operación de suma como $\ln(e^x + e^y) \approx \max(x, y)$, mientras que el algoritmo Log-MAP efectúa dicha operación como $\ln(e^x + e^y) = \max(x, y) + f_c(|y - x|)$. Por esto, el desempeño del algoritmo Max-Log-MAP en comparación con el MAP es sub-óptimo debido a que utiliza una

aproximación para efectuar las operaciones de suma en el dominio logarítmico. El algoritmo Log-MAP fue propuesto para corregir dicha aproximación y su desempeño es similar al del algoritmo MAP.

En 1989 Hagenauer y Hoeher propusieron una modificación al algoritmo de Viterbi que produce las probabilidades a-posteriori de las transiciones de estado, o equivalentemente la confiabilidad de los estimados de bit. Este algoritmo se conoce como SOVA (*Soft Output Viterbi Algorithm*) y tiene dos modificaciones con respecto al algoritmo de Viterbi clásico que lo hacen adecuado para ser utilizado en Turbo Decodificación: la primera consiste en que las métricas de trayectoria usadas son modificadas para tomar en cuenta información a-priori cuando se selecciona la trayectoria de máxima probabilidad a través del trellis, y la segunda radica en que el algoritmo provee una salida suave en forma del LLR a-posteriori $L(uk|y)$ para cada bit decodificado.

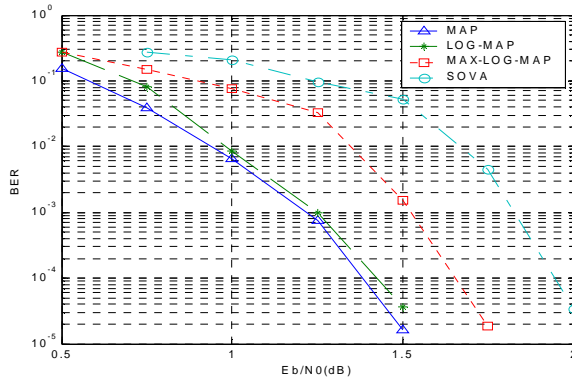


Fig. 11 Rendimiento de los algoritmos MAP, Log-MAP, Max-Log-MAP y SOVA.

D. Aplicaciones de los Turbo Códigos

[8] Ambos estándares de 3G, UMTS y cdma2000, satisfacen la mayoría de los requerimientos definidos por la ITU-R para los sistemas de radio móviles 3G. Debido al amplio desarrollo y gran popularidad que han adquirido los servicios inalámbricos, el número de usuarios móviles se ha incrementado en los últimos años, lo que conlleva a la necesidad de hacer un uso lo más eficientemente posible del espectro disponible. El tráfico de información multimedia requiere de tasas de transmisión y ancho de banda mucho mayores, que los requeridos para transmitir señales de voz, además de que la información multimedia es mucho más sensible a errores producidos durante su transmisión por el canal, por lo que, para servicios de transmisión multimedia en tiempo real, una transmisión rápida y libre de errores se convierte en una prioridad.

En ambos estándares, las especificaciones técnicas para el control de errores en la Capa Física incluyen tanto a los Códigos Convolucionales como a los Turbo Códigos: los primeros utilizados en canales de voz y control, y los segundos utilizados, mayormente, en canales de transferencia de datos (como información

multimedia), aunque también se considera su uso en algunos canales de control.

El hecho de que los Turbo Códigos no hayan sido considerados para ser utilizados en los canales de voz, es debido al ligero retardo que presenta el Turbo Decodificador al emplear un proceso de decodificación iterativa; resulta obvio que la transmisión de voz en un sistema de comunicaciones móviles debe realizarse en tiempo real. Sin embargo, este retardo es tan pequeño que no representa problema alguno para el caso de transmisiones de datos.

IV. DISEÑO DEL TURBO CODEC IMPLEMENTADO

Para el diseño del modelo de simulación del esquema de Turbo Codificación se hizo el análisis de investigaciones y estudios previos, buscando implementar bloques básicos y óptimos, que al final obtuvieran resultados con datos y gráficas consistentes. Los bloques que conforman el modelo realizado se observan en la Figura 12. Estos elementos fueron desarrollados utilizando el programa computacional MATLAB versión R2008a, aprovechando las características de eficiencia y rapidez de su compilador, ya que los cálculos que implican estas simulaciones así lo requieren.

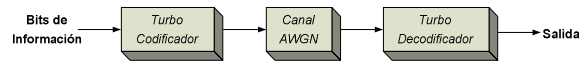


Fig. 12 Diagrama de bloques del modelo.

D. Sub-Funciones del Turbo Codec

Las sub-funciones que permiten el trabajo del Turbo Codec, están organizadas y articuladas tal como se muestra en la Figura 13.

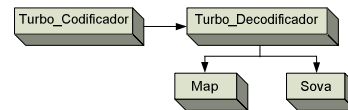


Fig. 13 Esquema de Funcionamiento de las sub-funciones del Turbo Codec.

La labor en conjunto de estas sub-funciones es la base de la herramienta de simulación de Turbo Códigos.

1) *Diseño del Turbo Codificador*: Un turbo codificador convolucional se basa en la concatenación en paralelo de dos codificadores RSC separados por un interleaver. Los bits de información son codificados por el primer componente codificador directamente, mientras que el segundo componente codificador opera con una versión 'desordenada' (por el interleaver) de ellos. La Figura 14 muestra un diagrama de bloques que ejemplifica el proceso:

V. INTERFAZ GRÁFICA DE LA HERRAMIENTA DE SIMULACIÓN

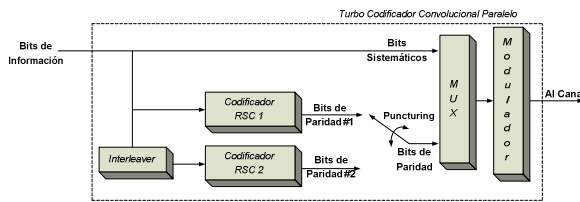


Fig. 14 Diagrama de Bloques del Turbo Codificador.

3) *Diseño del Turbo Decodificador:* La Turbo Decodificación se realiza de forma iterativa, es decir, la salida suave del proceso es realimentada y utilizada como entrada en la siguiente iteración de decodificación. Las iteraciones, aunque aumentan la latencia del sistema, reducen considerablemente la potencia requerida para alcanzar un mejor desempeño y aumentar la eficiencia de los receptores.

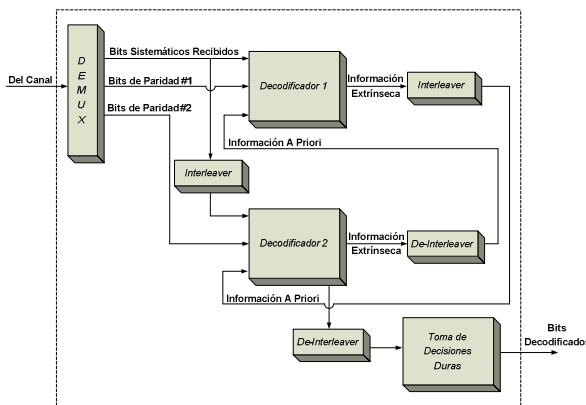


Fig. 15 Diagrama de Bloques del Turbo Decodificador.

Con la finalidad de observar la mejora en el rendimiento de los diversos códigos se ha construido la gráfica BER vs E_b/N_0 de la Figura 16 que sintetiza la información obtenida como resultado del proceso de simulación de los diferentes esquemas FEC.

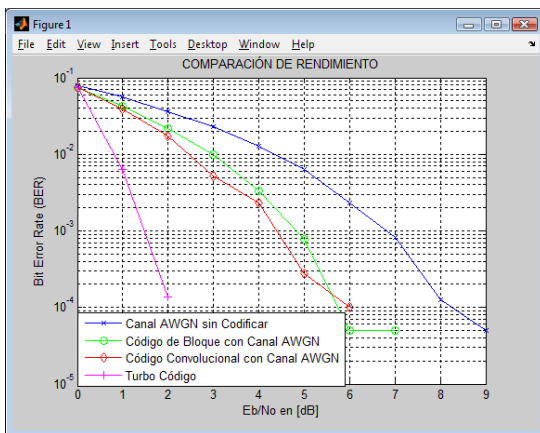


Fig. 16 Comparación de Rendimiento de un Canal AWGN sin Codificar, un Código de Bloque, un Código Convencional y un Turbo Código sobre un canal AWGN.

La interfaz gráfica se desarrolló en MATLAB R2008a, dada la familiaridad con las herramientas encontradas en el directorio `matlab\uitools` que facilitan la creación de interfaces con el usuario GUIDE, que permite crear interfaces de usuario de manera ágil y sencilla. GUIDE permite realizar conjuntos de pantallas, con botones, menús, ventanas, etc; generando dos archivos con el mismo nombre para cada pantalla creada: un archivo con extensión `.m` y un archivo con extensión `.fig`.

Estos archivos contienen toda la información básica de la interfaz, el archivo `.fig` contiene las características visuales de los elementos de cada pantalla y el archivo `.m` contiene la parte estructural de la misma, es decir, las funciones que ejecutan cada uno de los elementos que la conforman.

En la interfaz se tienen ocho diferentes pantallas en las que el usuario puede trabajar con las diversas opciones planteadas para las simulaciones.

A. Pantalla de Presentación

Brinda ligeros datos del prototipo y de sus autores, permite al usuario elegir si se requiere continuar con la ejecución del mismo o salir del programa.



Fig. 17 Pantalla de Presentación.

B. Pantalla de Simulación

Entre los parámetros de codificación que el usuario debe ingresar están:

- *Tamaño de la Trama:* puede ser ingresado como un número entero positivo o a su vez ingresando la trama bit a bit en formato binario.
- *Perforación del Código:* que determina si el código es punctured o unpunctured.
- *Código Generator:* presenta dos opciones, el escoger entre polinomios generadores preasignados para tecnologías como: ADSL, GSM, D-AMPS, UMTS, CDMA2000, WIMAX y uno por DEFAULT; o ingresar el código generador en formato octal.

Mientras que los parámetros de decodificación son:

- **Algoritmo de Decodificación:** el que puede ser Log-MAP o SOVA.
- **Número de Iteraciones por Trama:** que debe ser un número entero positivo.
- **E_b/N_0 (en dB):** siendo la única opción válida un entero positivo.

La pantalla además muestra espacios en los que se visualizarán los *Detalles del Proceso* y la *Ventana de Resultados*. La pantalla de Simulación está acompañada en la parte inferior de botones que permiten ejecutar diferentes procesos y órdenes.

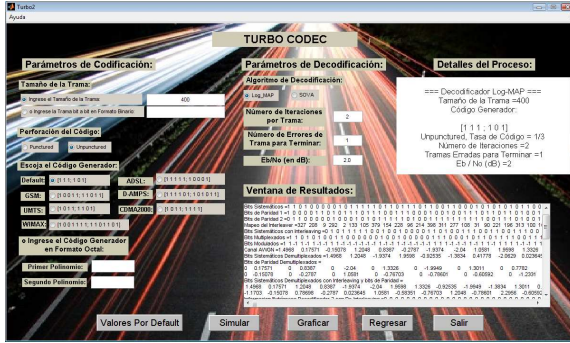


Fig. 18 Pantalla de Simulación del Turbo Codec

C. Pantalla de Graficación

Se presenta de manera gráfica los resultados del proceso del Turbo Codec, mediante curvas de rendimiento BER vs. E_b/N_0 . Esta pantalla se ejecuta al presionar el botón graficar en el Turbo Codec.

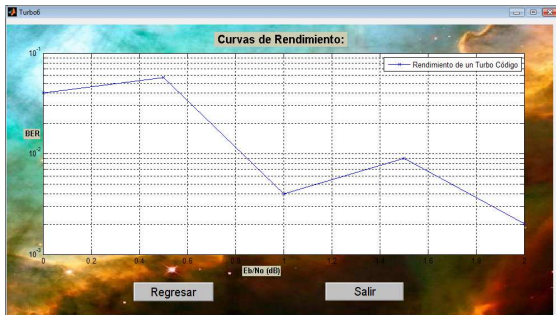


Fig. 19 Pantalla de Gráficas: BER vs. E_b/N_0 .

D. Pantallas de Ayuda

Mediante el uso del *Menu Editor* del MATLAB, se han creado tres sub-menús en la pantalla de Simulación del Turbo Codec, que permiten acceder a *Ayudas*.

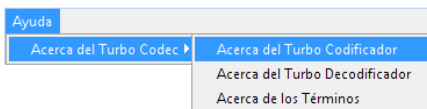


Fig. 20 Menú de Ayudas del Programa.

1) **Pantalla de Ayuda Acerca del Turbo Codificador:** Muestra el diagrama de bloques implementado para la Codificación Turbo.

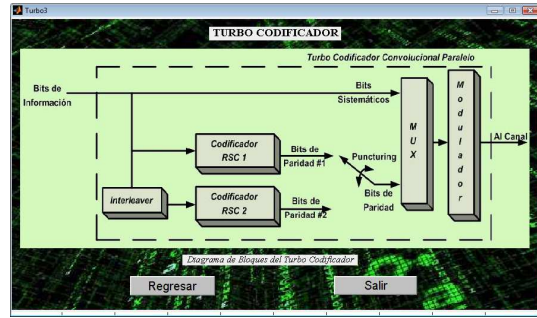


Fig. 21 Pantalla de Ayuda del Turbo Codificador.

2) **Pantalla de Ayuda Acerca del Turbo Decodificador:** Muestra el diagrama de bloques implementado para la Decodificación Turbo.

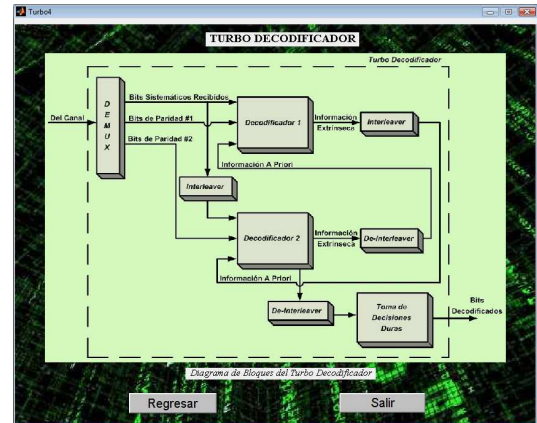


Fig. 22 Pantalla de Ayuda del Turbo Decodificador.

3) **Pantalla de Ayuda Acerca de los Términos Turbo Codificador:** Muestra el significado de una serie de palabras y términos que se han empleado en el proceso.

Término	Definición	Definición
ENCODIFICACIÓN	El emisor envía un mensaje que permite la transmisión de los datos de información en forma de bits, a través de un canal de transmisión, pero en un código de control, de forma tal que los bits que están sujetos a errores en la recepción original pueden ser recuperados por el receptor. Para evitar el deterioramiento por ruido, se transmite por el canal.	La E_b/N_0 (relación energía por bit / potencia de ruido) es una medida de la SNR normalizada. También se conoce como "SNR por bit".
DECODIFICACIÓN	Este elemento generalmente consiste en un par de decodificadores que recibe la señal de bits y son procesados para extraer los datos de la bits de paridad producido por los codificadores, para luego pasar a codificación más alta.	El algoritmo MAP (Maximum A Posteriori) maximiza la probabilidad de error de palabra y minimiza el error de bits. Este algoritmo las matrices de transición usando un modelo de canal para tener en cuenta el ruido y para cuando se selecciona la secuencia de bits con la máxima probabilidad a través del ruido, y además este algoritmo permite una tabla rápida en forma de LUTs (Look Up Tables) para cada bit decodificado.
DECODIFICACIÓN	El algoritmo MAP (Maximum A Posteriori) maximiza la probabilidad de error de palabra y minimiza el error de bits. Este algoritmo las matrices de transición usando un modelo de canal para tener en cuenta el ruido y para cuando se selecciona la secuencia de bits con la máxima probabilidad a través del ruido, y además este algoritmo permite una tabla rápida en forma de LUTs (Look Up Tables) para cada bit decodificado.	Los estados y salidas de los decodificadores son representados por los valores Log Likelihood (LLR). Entre los usuarios y se requiere las informaciones tales el tipo de cada bit y la probabilidad de un bit de error.
DECODIFICACIÓN	La modulación BPSK (Binary PSK) es un tipo de modulación, se le hace pasar los valores de 1 y 0. Entendido por modulación BPSK, la modulación de los bits en la que el parámetro de la portadora que pasa en función de la información binaria (0) de control, el modula en la fase.	La decodificación es el proceso de recuperar los datos de un mensaje.
DECODIFICACIÓN	En una trama que permite un flujo de datos. Cuando se transmite un mensaje, se transmite el código de control en la trama de la portadora que pasa en función de la información binaria (0) de control, el modula en la fase.	La decodificación es el proceso de recuperar los datos de un mensaje. Este proceso es el proceso de recuperar los datos de un mensaje, que no siempre se realiza.
DECODIFICACIÓN	El ruido es un fenómeno que ocurre en un canal de transmisión, que causa que los bits de información que están sujetos a errores en la recepción original pueden ser recuperados por el receptor. Para evitar el deterioramiento por ruido, se transmite por el canal.	Una tabla rápida es una tabla que permite acceder rápidamente a los datos de un mensaje. Este tipo de tabla es una tabla que permite acceder rápidamente a los datos de un mensaje, que no siempre se realiza.
DECODIFICACIÓN	El ruido es un fenómeno que ocurre en un canal de transmisión, que causa que los bits de información que están sujetos a errores en la recepción original pueden ser recuperados por el receptor. Para evitar el deterioramiento por ruido, se transmite por el canal.	Una tabla rápida es una tabla que permite acceder rápidamente a los datos de un mensaje. Este tipo de tabla es una tabla que permite acceder rápidamente a los datos de un mensaje, que no siempre se realiza.
DECODIFICACIÓN	El ruido es un fenómeno que ocurre en un canal de transmisión, que causa que los bits de información que están sujetos a errores en la recepción original pueden ser recuperados por el receptor. Para evitar el deterioramiento por ruido, se transmite por el canal.	Una tabla rápida es una tabla que permite acceder rápidamente a los datos de un mensaje. Este tipo de tabla es una tabla que permite acceder rápidamente a los datos de un mensaje, que no siempre se realiza.

Fig. 23 Pantalla de Ayuda Acerca de los Términos Usados.

E. Pantalla de Comparación de Rendimiento

Permite comparar de rendimiento de los diferentes

esquemas de codificación para detección y corrección de errores mediante el cálculo del BER y su graficación. Se puede acceder a esta pantalla a través de la pantalla de presentación del interfaz gráfico. Adicionalmente permite comparar la información obtenida si se varían los polinomios generadores del Turbo Codec. Al iniciar esta pantalla presenta las dos opciones propuestas (Comparar Turbo Código con Otros Esquemas FEC y Comparar Rendimiento de los Códigos Generadores del Turbo Codec).



Fig. 24 Pantalla de Comparación de Rendimiento.

La Figura 25 muestra la pantalla de gráficas, cuando ha sido ejecutada una comparación, entre diferentes esquemas de codificación FEC.

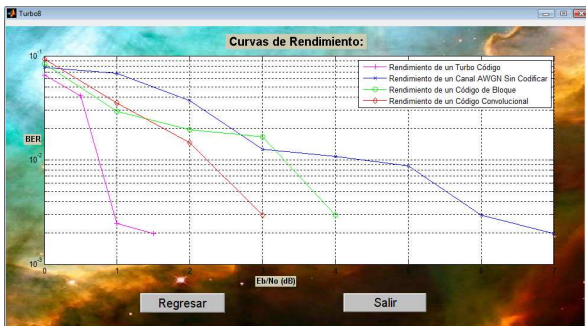


Fig. 25 Pantalla de Comparación de Gráficas: BER vs. E_b/N_0 con Códigos FEC.

La Figura 26, a continuación mostrada, presenta la pantalla de gráficas al compararse diferentes polinomios generadores con una misma secuencia de bits.

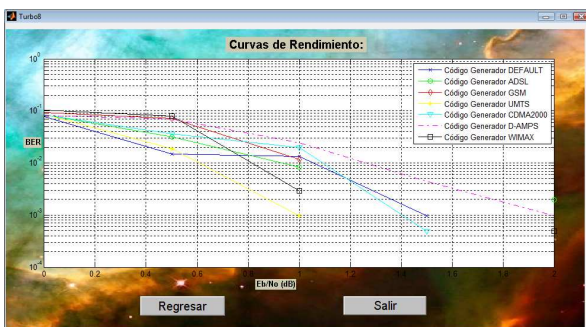


Fig. 26 Pantalla de Comparación Gráficas: BER vs. E_b/N_0 con Varios Códigos Generadores.

F. Mensajes de Error, Aviso y Espera

La Figura 27, presenta un mensaje de error, cuando no se han ingresado números o los campos están vacíos en la pantalla de simulación del Turbo Codec.

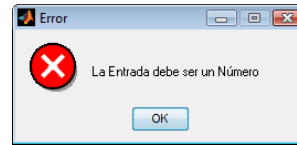


Fig. 27 Mensaje de Error en el Ingreso de Datos.

La Figura 28, muestra un mensaje de espera, mediante una barra durante el procesamiento de datos; que se presenta cuando se está ejecutando el Turbo Codec.

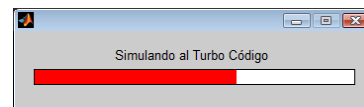


Fig. 28 Barra de Espera para Procesamiento de Datos.

La Figura 29, muestra un mensaje de aviso, que se presenta cuando se ejecuta el botón *Salir*, este mensaje permite confirmar o no el salir de la aplicación.



Fig. 29 Mensaje de Salida.

VI. CONCLUSIONES

- El análisis de las curvas de desempeño de los Turbo Códigos BER vs. E_b/N_0 , permiten plasmar el efecto que produce el variar algunos parámetros importantes como el algoritmo de decodificación SISO, el número de iteraciones utilizadas en el proceso de decodificado, la tasa de código y el tamaño y diseño del interleaver. De esta manera se concluye, que a mayor tamaño del interleaver y mayor número de iteraciones utilizadas, se incrementa el desempeño ofrecido por un Turbo Código pero aumenta también la memoria y el tiempo requeridos por el turbo decodificador para producir su secuencia de salida. Es por ello que el tamaño del interleaver y el número de iteraciones deben elegirse de forma tal que se satisfagan los requerimientos del sistema de BER, tardanza de decodificación y memoria utilizada por el decodificador.

- El sistema ideal de Shannon tiene un límite teórico que se encuentra alrededor de los 1.2 dB para un BER de 10^{-8} . Teniendo en cuenta que para las aplicaciones que usan al Turbo Código como esquema FEC se considera tener una tasa de bits errados en el rango **medio** cuando el BER fluctúa entre 10^{-2} y 10^{-6} , una tasa **baja** cuando el BER fluctúa entre 10^{-6} y 10^{-11} y una tasa **muy baja** cuando el BER es mayor a 10^{-11} . Se puede concluir que el Turbo Código configurado con los parámetros que se muestran en la Tabla 4.12, a pesar de ser un esquema para fines didácticos y encontrarse únicamente examinado a nivel de simulación tiene un rendimiento aceptable que se enmarca dentro del nivel **medio**.
- Los resultados de las simulaciones nos han mostrado la gran ventaja que supone utilizar turbo codificación frente a un canal sin codificación para control de errores, un código de bloque y la codificación convolucional convencional para longitudes de paquetes de entrada medianos, pero también que no siempre es preferible utilizar la turbo codificación frente a la codificación convolucional. Para longitudes de entrada menores a 160 bits, por ejemplo, se debe utilizar esta última en lugar de la turbo codificación.
- En la práctica no se utiliza un número de iteraciones muy alto para el proceso de turbo decodificación, ya que esto implica mayor tiempo de procesamiento por parte de la computadora en la parte del receptor, haciendo más lento todo el sistema.
- Se da a conocer el uso de una herramienta computacional creada para analizar códigos que detectan y corrigen errores, misma que podría ser usada para desarrollar nuevas técnicas de codificación de canal que mejorarían la confiabilidad de los sistemas de comunicación digitales existentes o futuros. Con el desarrollo de la interfaz gráfica de esta herramienta de simulación, se presentan de forma clara y explícita cada una de las variables que determinan el rendimiento de los Turbo Códigos, facilitando la labor de investigación, y a la vez, permitiendo su manejo a usuarios con ideas básicas en comunicaciones que deseen conocer la estructura, el funcionamiento y las ventajas de este importante esquema de codificación.

- Wireless Channels. Electronic Book, Department of Electronics and Computer Science, Ltd. United Kingdom. 2002.
- [4] ALVARADO, Raúl; Códigos para Detección y Corrección de Errores en Comunicaciones Digitales. Universidad Autónoma de Nuevo León, Facultad de Ingeniería Mecánica y Eléctrica. Vol. 7, N°25. Octubre-Diciembre 2004.
- [5] THE MATHWORKS; Communications Toolbox™ 4: User's Guide. The MathWorks, Inc; 3 Apple Hill Drive; Natick, MA. Marzo, 2009.
- [6] SACANAMBOY, Maribell; Tesis de Maestría: Diseño e Implementación de los Turbo Codificadores definidos en los estándares de Telecomunicaciones cdma2000 (TIA/EIA 2002.2D) y WCDMA (3GPP TS 25.212 v7.2.0) usando Hardware Reconfigurable. Universidad del Valle, Facultad de Ingeniería; Santiago de Cali. Octubre 20 del 2006.
- [7] BARBULESCU, Sorin A.; What a Wonderful Turbo World. Electronic Book, Version 1.2.
- [8] KEATTISAK, Sripimanwat; Turbo Code Applications: A Journey From a Paper to Realization. National Electronics and Computer Technology Center (NECTEC). Pathumthani, Thailand. Octubre, 2005.

Autores:



Diego Vallejo nacido en Quito - Ecuador, en 1985. Ing. en Electrónica y Telecomunicaciones, graduado de la Escuela Politécnica Nacional.



Robín Álvarez Rueda nacido en Cayambe - Ecuador, en 1969. Ingeniero en Telecomunicaciones, graduado de la Escuela Politécnica Nacional, Quito - Ecuador, 1995; MSc. en Telecomunicaciones por la Universidad de Cantabria, Santander - España, 2001; PhD en Telecomunicaciones por la Universidad Politécnica de Madrid - España, Enero-2006.

REFERENCIAS

- [1] SCHLEGEL, Christian B.; PÉREZ Lance C.; Trellis and Turbo Coding. Sexta Edición. IEEE Press Series on Digital & Mobile Communication. United States of America. 2004.
- [2] FRANCOS, Alfonso; Estudio Teórico de la Arquitectura de Turbo-Códigos para Aplicaciones de Telefonía Celular de 3G. Universidad de las Américas; Puebla - México. Septiembre 17, 2007.
- [3] HANZO, L.; LIEW, T.; YEAP, B.; Turbo Coding, Turbo Equalization and Space-Time Coding for Transmission over