

DISEÑO DE UN DECODIFICADOR VITERBI PARA SER UTILIZADO EN ENLACES INFRARROJOS

Leonardo Arnone, Carlos Gayoso, Claudio González, Juan Carlos García, Jorge Castiñeira⁽¹⁾

Laboratorio de Componentes Electrónicos. Facultad de Ingeniería. Universidad Nacional de Mar del Plata. Argentina. ⁽¹⁾ Laboratorio RF UNMdP.

leoarn@fi.mdp.edu.ar

ABSTRACT

Signal-to-noise ratio for infrared communications is mainly determined by the distance between the transmitter and the receiver, the infrared radiated power, and the noise power produced by environmental lighting. A higher level of environmental noise power reduces the signal-to-noise ratio, thus generating an increase on the number of errors in digital transmissions using this kind of link. An error correcting coding technique becomes an alternative to reduce these errors. In this paper an implementation using ALTERA programmable devices of a (2,1,3) convolutional code for digital infrared communications is presented. This implementation shows an important improvement in performance for the infrared link.

RESUMEN

Entre los factores más importantes que afectan a las comunicaciones infrarrojas interiores, están el ruido producido por la iluminación ambiental, la potencia infrarroja irradiada y la distancia entre el transmisor y el receptor infrarrojo. Esto provoca que la relación señal a ruido (SNR) en el receptor pueda variar considerablemente produciendo un aumento en los errores durante la transmisión. En este trabajo se expone un sistema implementado con dispositivos programables de ALTERA, para ser utilizado en comunicaciones infrarrojas interiores. Utiliza un codificador convolucional (2,1,3) y un decodificador basado en el algoritmo de Viterbi. Este sistema presenta una notable mejora en el funcionamiento del enlace infrarrojo.

DISEÑO DE UN DECODIFICADOR VITERBI PARA SER UTILIZADO EN ENLACES INFRARROJOS

Leonardo Arnone, Carlos Gayoso, Claudio González, Juan Carlos García, Jorge Castiñeira⁽¹⁾

Laboratorio de Componentes Electrónicos. Facultad de Ingeniería. Universidad Nacional de Mar del Plata. Argentina. ⁽¹⁾ Laboratorio RF UNMdP.

RESUMEN

En este trabajo se expone un sistema implementado con dispositivos programables de ALTERA, para ser utilizado en comunicaciones infrarrojas interiores. Utiliza un codificador convolucional (2,1,3) y un decodificador basado en el algoritmo de Viterbi. Este sistema presenta una notable mejora en el funcionamiento del enlace infrarrojo.

1. INTRODUCCIÓN

Los sistemas de telemetría de corto alcance que emplean dispositivos infrarrojos presentan algunas ventajas con respecto a los sistemas que emplean radio frecuencia. El ancho de banda utilizado por los enlaces infrarrojos no está regulado permitiendo esto usar elevadas velocidades de transmisión. Dado que la radiación infrarroja es bloqueada por las paredes o barreras opacas, no se producen interferencias entre sistemas colocados en diferentes habitaciones o separados por paneles opacos[1].

Estos enlaces deben ser compactos, consumir poca potencia y ser robustos contra el ruido ambiental y la interferencia producida por otros usuarios.

Sin embargo, los enlaces infrarrojos deben operar en la presencia de una fuerte radiación ambiental producida por la luz solar, la iluminación incandescente y fluorescente. También el rango de operación del enlace [1] queda restringido debido a que la potencia de transmisión debe limitarse para evitar un consumo elevado y para prevenir daños en la vista. Por estos motivos la relación señal a ruido (SNR) en el receptor pueda variar considerablemente con el consiguiente aumento de los errores producidos durante la transmisión.

En este trabajo se expone el diseño de un codificador/decodificador Viterbi[2][6] que produce una notable mejora en el funcionamiento del enlace infrarrojo. Primero se describe el modelo del enlace infrarrojo, luego se exponen los fundamentos de un

codificador convolucional[2][6] y un decodificador que utiliza el algoritmo de Viterbi[2]. Posteriormente se trata el diseño de un codificador (2,1,3) y un decodificador Viterbi[2] implementado con dispositivos programables de ALTERA [3]. Por último se compara la mejora obtenida en el enlace con respecto a una modulación sin codificar.

2. ENLACE INFRARROJO

Para lograr una alta relación señal a ruido (SNR) los sistemas infrarrojos emplean modulación de la intensidad con detección directa (IM/DD)[1]. En modulación de intensidad, la señal transmitida $X(t)$ es la potencia óptica instantánea. La señal recibida $Y(t)$ es la corriente producida en el fotodetector del receptor, la cual es el producto de la respuesta R ($A \cdot cm^2/W$) del fotodetector y la potencia P_r recibida en su superficie. La potencia recibida está dada por [4]:

$$P_r = H_o(\rho) P_t \quad (1)$$

donde $H_o(\rho)$ (cm^{-2}) es la ganancia óptica del enlace entre el transmisor y el receptor separados por una distancia ρ y P_t es la potencia promedio transmitida, la cual está definida como:

$$P_t = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T X(t) dt \quad (2)$$

El ruido N_0 (A^2/Hz) presente en el enlace, producido por la luz ambiental, se puede considerar como ruido blanco gaussiano [1]. N_0 se supone constante e independiente de la posición del receptor. Si se define el parámetro γ_b [5] como la relación señal a ruido (SNR) dada por:

$$\gamma_b = \frac{R^2 P_r^2}{R_b N_0} = \frac{R^2 H_o^2(\rho) P_t^2}{R_b N} \quad (3)$$

entonces la probabilidad de error en términos de γ_b durante la transmisión estará dada por:

$$p_e = Q(\sqrt{\gamma_b}) \quad (4)$$

3. CÓDIGOS CONVOLUCIONALES

Existen dos tipos de codificación para el control de error en un canal, los códigos de bloques[2] y los códigos convolucionales[2]. En un código convolucional los n bits de la salida del codificador dependen no sólo de los k bits de entrada, sino también de K bloques de información de entrada previos. Un código convolucional (n,k,K) se implementa con k entradas que ingresan a un circuito secuencial de n salidas con un nivel de memoria K . Típicamente n y k son números enteros siendo normalmente $k < n$. El nivel de memoria K debe hacerse grande para aumentar la capacidad de detección de error del código.

La figura 1 representa un codificador para generar un código convolucional binario (2,1,3).

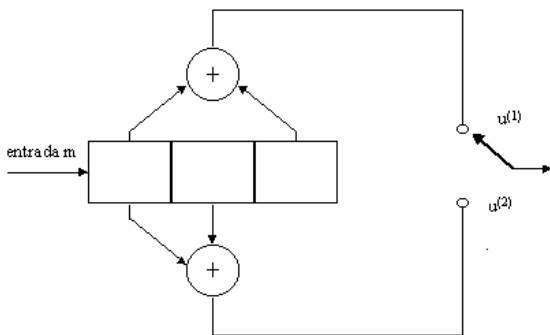


Figura 1. Codificador convolucional (2,1,3)

El vector de entrada es una secuencia que tiene la forma:

$$m = (m_0, m_1, m_2, \dots) \quad (5)$$

El vector de salida u se configura con las secuencias de salida $u^{(1)}$ y $u^{(2)}$:

$$\begin{aligned} u^{(1)} &= (u_0^{(1)}, u_1^{(1)}, u_2^{(1)}, \dots) \\ u^{(2)} &= (u_0^{(2)}, u_1^{(2)}, u_2^{(2)}, \dots) \end{aligned} \quad (6)$$

Un esquema general de un codificador convolucional emplea kK registros que operan en modulo-2. El parámetro K se denomina longitud de restricción. Consiste básicamente de una medida de la forma en que los bits previos afectan a los que se están generando en la salida. En la figura se ve que las secuencias de entrada de k bits son desplazadas hacia la derecha en las

primeras k etapas de los registros. La salida es muestreada para generar la secuencia de salida del código. Existen n bits de salida por cada k bits de entrada, por lo que se puede definir la velocidad del código como k/n siendo $k < n$. La mayoría de los códigos convolucionales emplean $k=1$ de forma que los bits de entrada son ingresados de forma serie. Para este tipo de codificador convolucional la velocidad de código es $1/n$. El registro del sistema que tiene kK etapas, tendrá en este caso entonces K etapas.

3.1. Representación por diagrama de estados de un codificador convolucional

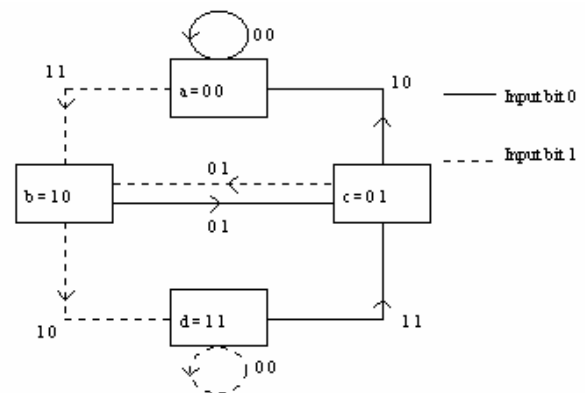


Figura 2. Diagrama de estados del codificador

El estado de un código convolucional $1/n$ se define como el contenido de los $K-1$ registros de estado que se encuentran a la derecha. De esta manera los $K-1$ registros de estado de la izquierda constituyen el próximo estado. El diagrama de estados[2][6] es una forma de representación de la evolución de las secuencias de estado para estos códigos. El codificador analizado en este ejemplo tiene un diagrama de estados como el que se ve en la figura 2.

El sistema se analiza considerando que existen cuatro estados representados por $a=00$, $b=10$, $c=01$ y $d=11$. Existen sólo dos transiciones que emergen de cada estado y tienen que ver con las dos posibilidades de entrada binaria existentes, '1' o '0'. La convención empleada es que un '1' lógico de entrada se representa con líneas de puntos, mientras que un '0' lógico de entrada se representa con líneas llenas.

Una característica importante de estos códigos se hace evidente de la observación del diagrama de estados. No es posible pasar de un estado a otro arbitrario. Esto resulta del mecanismo de memoria que el sistema tiene.

3.2. Trellis para un código convolucional

El diagrama de estados permite conocer la transición entre estados, pero no aporta información acerca de la historia del sistema, es decir, de los estados y transiciones iniciales.

El diagrama de Trellis[2] tiene en cuenta la dimensión del tiempo, poniendo en evidencia la forma en que la secuencia realmente sucede. El diagrama de Trellis para la secuencia de entrada $m=(100101)$ que entra al codificador convolucional se representa en la figura 3

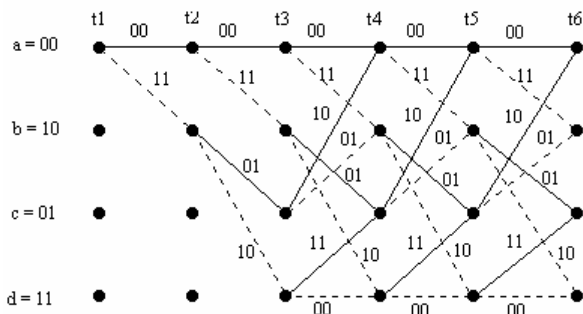


Figura 3. Diagrama de Trellis

Una propiedad importante de los códigos convolucionales es que, el sistema tiene, a partir de determinado momento, una repetición en su estructura. La estructura se repite después de K ramas, donde K es la constante de longitud asociada a estos códigos.

Las líneas sólidas representan un bit de entrada '0' mientras que líneas punteadas se usan para representar un bit de entrada '1'. Existen 2^{K-1} posibles estados para ser representados. También hay dos ramas que emergen de cada estado pertenecientes a los dos posibilidades de bits de entrada, y dos ramas que arriban a cada estado.

3.3. Decodificación por máxima similitud[2][6]

Para una cierta secuencia de salida de un código convolucional $U^{(m)}$ existe una secuencia recibida, denominada Z , que resulta de la aparición de errores por efecto del ruido sobre la secuencia original U . El decodificador óptimo es tal que compara las probabilidades condicionales $P(Z/U^{(m)})$ de que la secuencia recibida Z corresponda a una secuencia conocida $U^{(m)}$ y toma la decisión de elegir la secuencia que tiene la mayor probabilidad de coincidir con Z :

$$P(Z/U^{(m)}) = \max_{U^{(m)}} P(Z/U^{(m)}) \quad (7)$$

Este es el criterio de máxima similitud. Esta obviamente de acuerdo con el hecho intuitivo de asignar la secuencia

conocida en el decodificador que más se parece a la recibida.

La aplicación de este criterio a los códigos convolucionales se enfrenta con el hecho de que un gran número de secuencias posibles debe ser considerada. Para una secuencia de L bits se tienen 2^L posibilidades. El decodificador de máxima similitud elige una secuencia $U^{(m)}$ del conjunto posible, que tenga la mayor probabilidad de parecerse al vector recibido.

Si el canal no tiene memoria, el ruido es aditivo blanco y gaussiano, por lo tanto, cada símbolo está afectado de una forma independiente de los otros. Para un código convolucional de velocidad $1/n$ la función probabilidad que mide la similitud respecto del vector recibido es:

$$\begin{aligned} P(Z/U^{(m)}) &= \prod_{i=1}^{\infty} P(Z_i/U_i^{(m)}) = \\ &= \prod_{i=1}^{\infty} \prod_{j=1}^n P(z_{ji}/u_{ji}^{(m)}) \end{aligned} \quad (8)$$

Donde Z_i es la i -ésima rama de la secuencia recibida Z , $U_i^{(m)}$ es la i -ésima rama de la secuencia de una palabra de código $U^{(m)}$, z_{ji} es el j -ésimo símbolo de código de Z_i , y $u_{ji}^{(m)}$ es el j -ésimo símbolo de código de $U_i^{(m)}$, donde cada rama está constituida por n símbolos de código. El proceso de decodificación consiste en la elección de una secuencia que maximice la función similitud. Este problema se analiza a continuación

3.4. Algoritmo de decodificación convolucional de Viterbi

El algoritmo de decodificación de Viterbi[2] realiza la detección por máxima similitud. Para eso utiliza las propiedades del Trellis de un código convolucional. El algoritmo intenta reducir la complejidad del cálculo evitando tener en cuenta la totalidad de las secuencias posibles. El procedimiento consiste en calcular la distancia entre la señal recibida en el instante t_i y los caminos o ramas entrantes del Trellis en ese instante en el estado analizado. En la medida que este criterio se va aplicando se evalúa la secuencia que tiene la menor distancia respecto de la recibida, de forma que la secuencia de máxima similitud finalmente aparece. El camino o secuencia de máxima similitud, que es al mismo tiempo aquel que presenta la menor distancia respecto del recibido, se denomina camino o secuencia sobreviviente, la cual es almacenada. De esta forma, el algoritmo de Viterbi trata de encontrar el camino con mínimo error. Decir entonces que la secuencia es la de máxima similitud, es también decir que se trata de

aquella secuencia que tiene la menor distancia respecto de la recibida.

4. IMPLEMENTACIÓN CON DISPOSITIVOS PROGRAMABLES DE ALTERA

El sistema implementado utiliza un codificador convolucional (2,1,3), con polinomios generadores[2] $g_1=101$ y $g_2=110$. La palabra sin codificar tiene una longitud de 5 bits y la palabra codificada de 10bits.

El transmisor utilizado se muestra en la figura 4, el primer bloque (*conversor paralelo/serie*) simplemente transforma los datos de entrada, de 5 bits, que ingresan en forma paralela a serie. Luego estos bits en formato serie ingresan al bloque que realiza la codificación (*codificador convolucional*). Este bloque se encarga de recorrer los diferentes estados del diagrama de estado que se muestra en la figura 2. Cada nueva palabra comienza siempre en el estado 00. A la salida se obtiene una palabra codificada de 10 bits. A continuación se ingresa a un *conversor paralelo/serie*, el cual ataca directamente al transmisor infrarrojo.

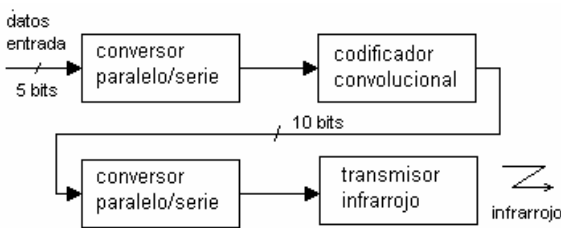


Figura 4. Sistema transmisor

En el sistema receptor se utiliza para decodificar la señal un decodificador Viterbi. Como se explicó anteriormente, el fundamento de este algoritmo está en que no almacena todas las secuencias a las que da lugar el decodificador, sino, sólo el mejor camino desde el principio del diagrama de Trellis hasta cada nodo. El bloque principal del decodificador Viterbi se muestra en la figura 5, cada celda representa un nodo del diagrama de Trellis. El bloque tiene las siguientes entradas:

- Prog_estado** = se utiliza para programar la ubicación de cada celda dentro del diagrama de Trellis.
- bA**= bit que provoca que del nodo o celda A se pase a la celda C.
- regA**= contiene la secuencia de bits que son necesarios para llegar desde el nodo de partida al nodo A.
- distA**= distancia de Hamming entre el vector recibido y la salida correspondiente del nodo A.
- acumA**= contiene la suma de las distancias mínimas Hamming, de los nodos, que se recorrieron previamente antes de llegar al nodo A.

Las entradas **bB**, **regB**, **distB** y **acumB** tienen las mismas funciones que las anteriores. Las salidas de cada celda son:

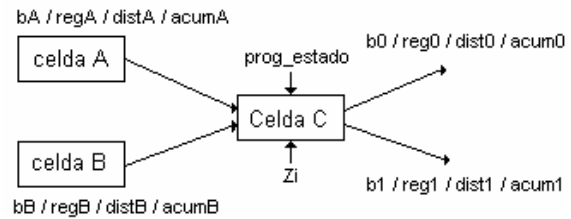


Figura 5. Celda básica del decodificador

b0= corresponde a la rama de salida que indica que el bit decodificado es un cero.

dist0= distancia de Hamming entre el vector recibido y la salida correspondiente de este nodo cuando el bit de entrada a esta celda es un cero.

acum0= contiene la suma de las distancias mínimas de Hamming, de los nodos, que se recorrieron previamente antes de llegar a este nodo.

reg0= contiene la secuencia de bits que son necesarios para llegar desde el nodo de partida a este nodo cuando el bit de entrada a esta celda es un cero.

Z_i= son los bits recibidos en ese nodo.

Las salidas **b1**, **dist1**, **acum1** y **reg1** tienen las mismas funciones que las anteriores, sólo que corresponden a cuando el bit decodificado es un uno.

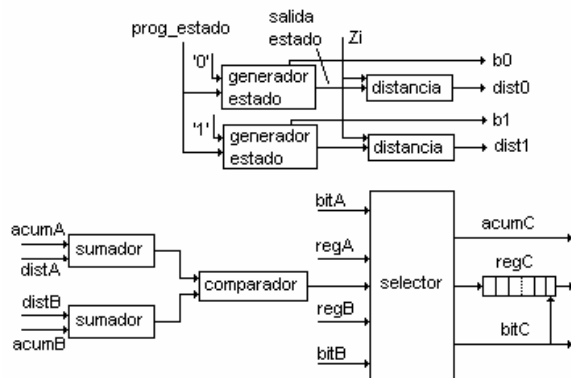


Figura 6. Bloques que componen cada nodo

En la figura 6, se muestra los diferentes bloques que componen cada nodo o celda del diagrama de Trellis. El circuito *Generador de Estado* produce la salida que corresponde al estado que se programó, dependiendo si el bit de entrada es un cero o un uno. El bloque *Distancia*, da la distancia de Hamming que hay entre la salida del *Generador de Estado* y los bits recibidos. Los

bloques *Sumadores* calculan la distancia total de cada camino que ingresa al nodo y mediante el *Comparador* y *Selector* se selecciona el camino que posee menor distancia de Hamming. Por último dependiendo de cual camino tiene menor peso se adiciona el bitA al regA o el bitB al regB, el cual pasará a ser el regC. La salida acumC es acumA+distA o acumB+distB, dependiendo cual tiene menor valor.

El transmisor[7] se implemento en un dispositivo EPM7128SLC84-7 de la familia MAX 7000S. El receptor[7], utiliza 19 celdas como la mostrada en la figura 6, además de un sistema multiplexor para elegir el camino final con menor distancia de Hamming. Para su implementación se utilizó un dispositivo ALTERA EPF10K20RC240-A de la familia FLEX 10K. Como transmisor y receptor infrarrojo se utilizó tranceptor HEWLETT PACKARD HSDL-1100[8]

5. RESULTADOS

En la figura 7 se puede observar la mejora obtenida cuando se utiliza un código convolucional (2,1,3) con respecto a una transmisión sin codificar. Los resultados se obtuvieron mediante el uso del simulador MAXPLUS II de ALTERA y el programa MATLAB. Se supone que el enlace es un canal simétrico binario con probabilidad p , es decir, es igualmente probable que cuando se transmita un 0 se reciba un 1 ó cuando se transmita un 1 se reciba un 0.

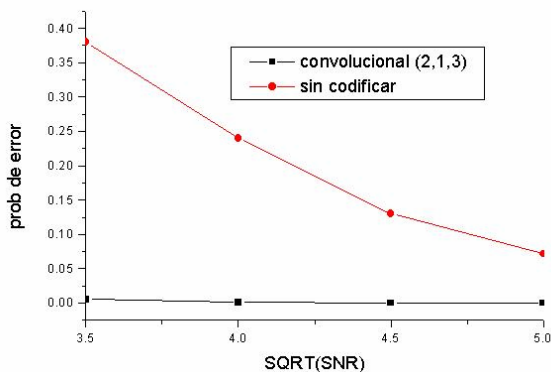


Figura 7. P_{err} para un código convolucional (2,1,3)

Para medir la tasa de error[9] en forma experimental, se construyó un prototipo utilizando dos plaquetas UP1 Educational Board, de ALTERA, y se situaron ambos dispositivos a una distancia variable entre 50cm y 130cm. Se hizo que el transmisor transmitiera secuencias de 600.000 palabras al receptor, en un ambiente iluminado con tubos fluorescentes e incidiendo alternadamente luz solar directa en el circuito transmisor y receptor infrarrojo. En la figura 8 se

muestra como varía el porcentaje de palabras recibidas con respecto al total de palabras transmitidas, para diferentes distancias, cuando se transmite una palabra sin codificar y una codificada. Se hace evidente la mejora que se obtiene con el uso de un codificador convolucional.

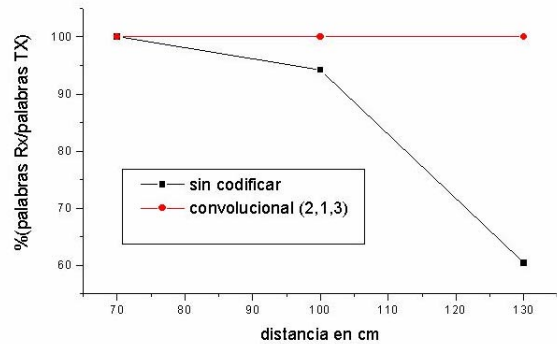


Figura 8. porcentaje de palabras recibidas en función de la distancia del enlace.

6. CONCLUSIONES

En este trabajo se expuso el diseño de un codificador convolucional, y un decodificador que utiliza el algoritmo de Viterbi, implementado con dispositivos programables de ALTERA, para ser utilizado en comunicaciones infrarrojas interiores. Debido a que este sistema produce una notable mejora en el funcionamiento con respecto a un sistema sin codificar, el uso del algoritmo de Viterbi es un método promisorio para lograr un enlace infrarrojo con bajo error de transmisión.

7. REFERENCIAS

- [1] J. M. Kahn, J. R. Barry. Wireless Infrared Communications. Proceedings of the IEEE. Vol 85. No 2. February 1997.
- [2] B. Sklar. Digital Communications, Fundamentals and Applications, Prentice Hall, 1993.
- [3] Altera. Data book 1998. Altera Corporation.
- [4] J. R. Barry, J. M. Kahn, W. J. Krause, E. A. Lee, D. G. Messerschmitt. Simulation of multipath impulse response for wireless optical channels. IEEE J. Select Areas in Commun. Vol 11 No 3. April 1993.
- [5] G. W. Marsh, J. M. Kahn. Channel Reuse Strategies for Indoor Infrared Wireless Communications
- [6] J. G. Proakis, M. Salehi. Communication Systems Engineering. Ed Prentice Hall
- [7] Altera. MAX + PLUS II VHDL. Altera Corporation
- [8] HSDL-1100, Infrared Transceiver, Technical Data. HEWLETT PACKARD
- [9] M. Blaum. A Course on Error Correcting Codes. IBM Research Division.