

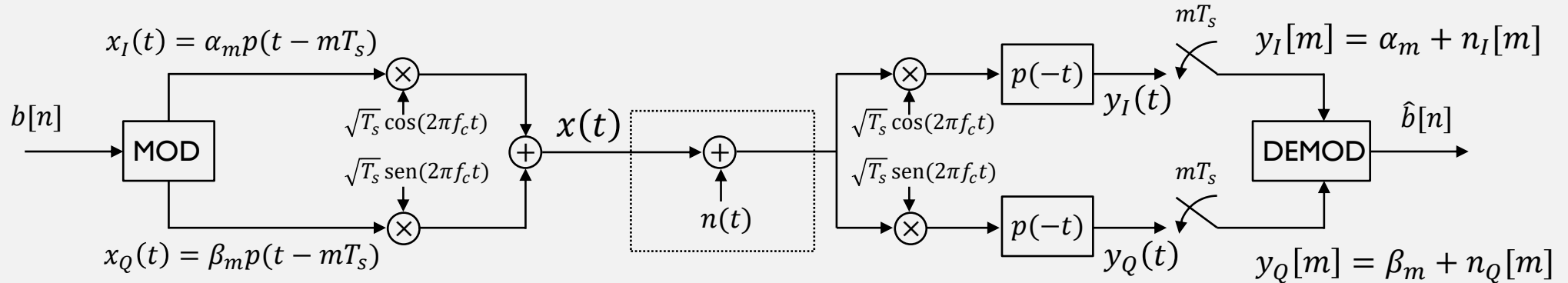
TEORÍA DE LA INFORMACIÓN Y CODIFICACIÓN

TRABAJO PRÁCTICO DE SIMULACIÓN

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital

Modelo de señal pasabanda

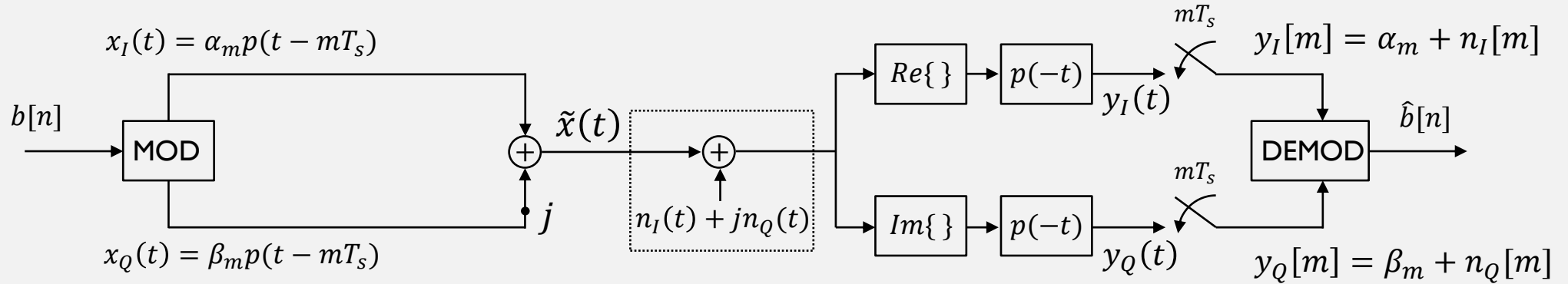


- $b[k]$: Secuencia de bits, 0 ó 1 de forma equiprobable e independiente.
- $x_I(t)$: Señal PAM banda base en fase de amplitudes α_m .
- $x_Q(t)$: Señal PAM banda base en cuadratura de amplitudes β_m .
- $p(t)$: Forma de pulso.
- $x(t)$: Señal pasabanda transmitida.
- $n(t)$: RABG con DEP plana $\frac{N_0}{2}$.
- $y_I(t)$: Salida del F.A., rama en fase.
- $y_Q(t)$: Salida del F.A., rama en cuadratura.
- $\hat{b}[k]$: Secuencia de bits estimada por el receptor.

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital

Modelo de envolvente compleja

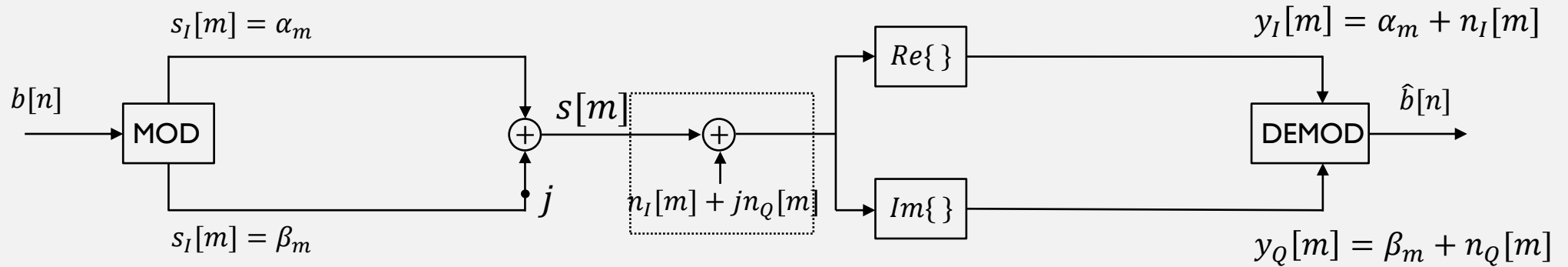


- $x(t) = Re\{\tilde{x}(t)e^{-j2\pi f_c t}\}$
- $\tilde{x}(t) = x_I(t) + jx_Q(t)$

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital

Modelo a nivel de símbolo

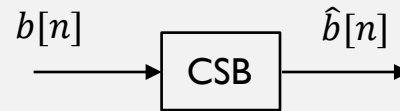


- $s[m] = s_I[m] + js_Q[m]$
- $n_I[m], n_Q[m] \sim N(0, \frac{N_0}{2})$ i.i.d. e indeptes entre sí

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital

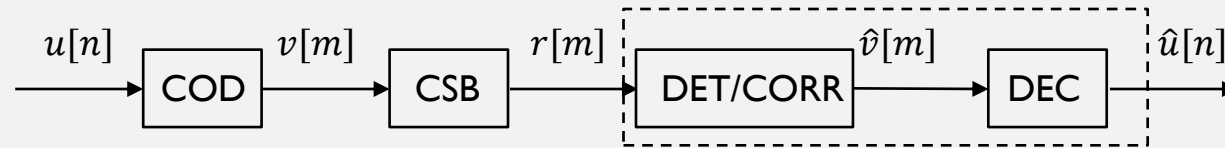
Modelo de canal simétrico binario



- CSB con probabilidad de transición p .

TRABAJO PRÁCTICO DE SIMULACIÓN

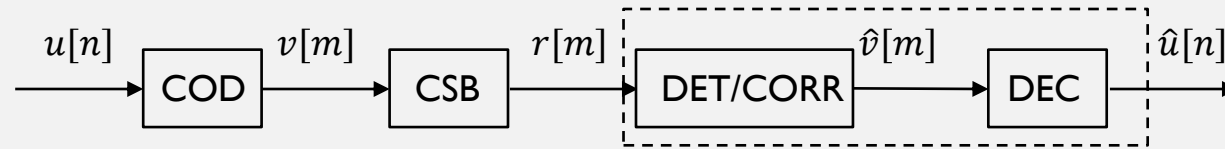
Simulación de sistema de comunicación digital con codificación de canal



- CSB con probabilidad de transición p .
- $u[n]$ secuencia de bits de fuente.
- $v[m]$ secuencia de bits de canal.
- $r[m]$ secuencia de bits recibidos.
- $\hat{v}[m]$ secuencia de bits corregidos.
- $\hat{u}[n]$ secuencia de bits de fuente estimada.

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal



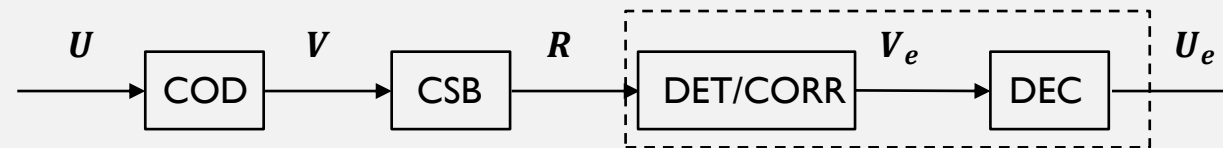
- Generar $u[n]$.
- Implementar la codificación de código de bloque indicada en el enunciado.
- Utilizar el código provisto por la cátedra para simular la transmisión por el canal AWGN con BPSK.
- Implementar la detección/corrección de errores en las palabras recibidas y decodificación.
- Estimar la probabilidad de error de palabra P_{ep} y la de bit P_{eb} .

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Generar $u[n]$.

Se sugiere representar a $u[k]$ de forma matricial con filas de largo k .
Generamos \mathbf{U} , matriz de dimensiones $\frac{M}{k} \times k$ con elementos 0 y 1 equiprobables e independientes.

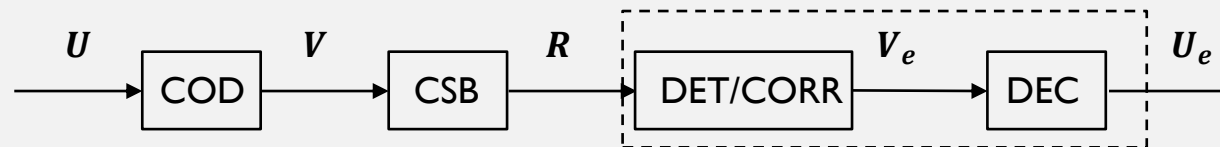


TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Implementar la codificación de código de bloque indicada en el enunciado.

Proponer una matriz G que genere un código de bloque sistemático (14,10).
Obtener $V = UG$, matriz de dimensiones $\frac{M}{k} \times n$ cuyas filas serán las palabras de código transmitidas. Tener en cuenta que ese producto matricial debe utilizar aritmética de módulo 2.



TRABAJO PRÁCTICO DE SIMULACIÓN

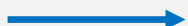

Simulación de sistema de comunicación digital con codificación de canal

- Utilizar el código provisto por la cátedra para simular la transmisión por el canal AWGN con BPSK.

Deben tenerse en cuenta las siguientes consideraciones:

1. Debe poder funcionar con entrada matricial.
2. La amplitud de los símbolos debe ser la apropiada para la E_b/N_0 deseada.

Canal BPSK:

- Energía de símbolo: $E_{sc} = A^2$  De canal
- Energía de bit de canal: $E_{bc} = E_s$  Por ser canal binario
- Energía de bit de fuente: $E_{bf} = \frac{n}{k} E_c$

$$A = \sqrt{\frac{k}{n} E_b}$$

La salida será la secuencia $r[m]$ en forma matricial, \mathbf{R} de dimensiones $\frac{M}{k} \times n$.

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Implementar la detección/corrección de errores en las palabras recibidas y decodificación.

Calcular los síndromes utilizando la matriz de control de paridad H^T .

Conviene seguir utilizando la forma matricial: $S = RH^T$.

1. Corrección de errores: Identificar los síndromes no nulos en las filas de S y compararlos con las filas de H^T . Cambiar el bit en la posición de la fila de H^T que coincida con el síndrome, obteniendo V_e .
2. Detección de errores: Identificar los síndromes no nulos en las filas de S y eliminar las palabras de R que correspondan a esos síndromes, obteniendo V_e . Llevar el registro de las filas eliminadas para poder hacer la estimación de P_{ep} y P_{eb} .

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Implementar la detección/corrección de errores en las palabras recibidas y decodificación.

Decodificar las palabras en V_e para obtener la secuencia $\hat{u}[k]$.

Esta secuencia la representamos en forma matricial como U_e de dimensiones $\frac{M}{k} \times k$ en el caso de código corrector y $\left(\frac{M}{k} - N_d\right) \times k$ en el de código detector, donde N_d es el número de palabras recibidas en las que se detectó error.

Como la codificación es sistemática, la decodificación es simplemente extraer la submatriz compuesta por las primeras k columnas de V_e .

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Estimar la probabilidad de error de palabra P_{ep} y la de bit P_{eb} .

Sugerencia:

Obtener una matriz de error E comparando la matriz de bits estimados U_e con los transmitidos U . En el caso de código detector hay que sacar de U las palabras en las que se detectó error antes de comparar con U_e .

La cantidad de filas no nulas de E será el número de palabras erradas. Se estima la P_{ep} dividiendo por la cantidad total de filas.

La cantidad de elementos no nulos de E será el número de bits errados. Se estima la P_{eb} dividiendo por la cantidad total de elementos.

TRABAJO PRÁCTICO DE SIMULACIÓN

Simulación de sistema de comunicación digital con codificación de canal

- Estimar la probabilidad de error de palabra P_{ep} y la de bit P_{eb} .

Comparar las curvas estimadas con las cotas teóricas en todos los casos (ver [clase teórica de desempeño de códigos lineales](#)).

TRABAJO PRÁCTICO DE SIMULACIÓN

Tips:

- Dada la baja probabilidad de error que se obtendrá, se requieren secuencias más largas para la estimación. En caso de tener problemas de falta de memoria, puede utilizar un bucle general que repita varias veces las estimaciones por valor de E_b/N_0 .
- Utilizar `mod(x, 2)` para las operaciones módulo 2.
- Para comparar bloques de bits, conviene pasarlos a representación decimal usando `bi2de`. Se puede usar directamente sobre una matriz, dando como resultado un vector columna cuyos elementos son las representaciones decimales de los números binarios formados en cada fila de la matriz original.
- Utilizar indexación para seleccionar submatrices dentro de otras. Ej:
`A=B(:, 1:4);`

TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente



- Estimar las probabilidades a priori de los mensajes para fuente extendida de orden 2.
- Utilizar algoritmo de Huffman para hallar la regla de codificación.
- Codificar la imagen.
- Calcular el largo promedio de la secuencia codificada y la tasa de compresión.
- Repetir para fuente extendida de orden 3.

TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente

- Estimar las probabilidades a priori de los mensajes para fuente extendida de orden 2 y 3.



Leer el archivo y estimar en base a la imagen.
Por ej., en Matlab: `imread`

Fuente binaria: $m_0 = 0, \quad m_1 = 1.$

Fuente binaria de orden 2: $m_0 = 00, \quad m_1 = 01, \quad m_2 = 10, \quad m_3 = 11.$

Fuente binaria de orden 3: $m_0 = 000, m_1 = 001, m_2 = 010, m_3 = 011,$
 $m_4 = 100, m_5 = 101, m_6 = 110, m_7 = 111,$



TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente

- Estimar las probabilidades a priori de los mensajes para fuente extendida de orden 2 y 3.

Mensaje más probables

Fuente binaria: $m_0 = 0$, $m_1 = 1$.

Fuente binaria de orden 2: $m_0 = 00$, $m_1 = 01$, $m_2 = 10$, $m_3 = 11$.

Fuente binaria de orden 3: $m_0 = 000$, $m_1 = 001$, $m_2 = 010$, $m_3 = 011$,
 $m_4 = 100$, $m_5 = 101$, $m_6 = 110$, $m_7 = 111$,



TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente

- Estimar las probabilidades a priori de los mensajes para fuente extendida de orden 2 y 3.

Conviene representación decimal

Fuente binaria: $m_0 = 0, \quad m_1 = 1.$

Fuente binaria de orden 2: $m_0 = 0, \quad m_1 = 1, \quad m_2 = 2, \quad m_3 = 3.$

Fuente binaria de orden 3: $m_0 = 0, \quad m_1 = 1, \quad m_2 = 2, \quad m_3 = 3,$
 $m_4 = 4, \quad m_5 = 5, \quad m_6 = 6, \quad m_7 = 7,$



TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente

- Utilizar algoritmo de Huffman para hallar la regla de codificación.
- Codificar la imagen.

Usando los conceptos adquiridos en los ejercicios de la práctica, hallar la regla de codificación con fuente de orden 2 y 3 utilizando los valores de probabilidades a priori estimados.

Emplearla para codificar la imagen en una nueva secuencia binaria.



TRABAJO PRÁCTICO DE SIMULACIÓN

Codificación de fuente

- Calcular el largo promedio de la secuencia codificada y la tasa de compresión.

Se estima el nuevo largo promedio del mensaje comprimido, como el largo total de la secuencia codificada dividido por la cantidad de bits de fuente que representan al mensaje.

La tasa de compresión la calculamos como el largo de la secuencia sin codificar dividido el largo de la secuencia codificada. En este caso, como la fuente es binaria, será la inversa del largo promedio.

