

Sincronización

Francesc Rey Micolau
Javier Villares Piera

PID_00197108



Los textos e imágenes publicados en esta obra están sujetos –excepto que se indique lo contrario– a una licencia de Reconocimiento-NoComercial-SinObraDerivada (BY-NC-ND) v.3.0 España de Creative Commons. Podéis copiarlos, distribuirlos y transmitirlos públicamente siempre que citéis el autor y la fuente (FUOC. Fundació para la Universitat Oberta de Catalunya), no hagáis de ellos un uso comercial y ni obra derivada. La licencia completa se puede consultar en <http://creativecommons.org/licenses/by-nc-nd/3.0/es/legalcode.es>

Índice

Introducción	5
Objetivos	7
1. Motivación	9
2. Técnicas de sincronización	17
2.1. Estimación de parámetros de sincronismo	19
2.2. Corrección de parámetros de sincronismo	22
3. Sincronismo de frecuencia de portadora	26
4. Sincronismo de símbolo	32
5. Sincronismo de fase de portadora	37
6. Sincronismo de trama	41
7. Conclusiones	44
Actividades	45
Bibliografía	48

Introducción

En la asignatura *Sistemas de comunicaciones I* (SC1) se han presentado las principales modulaciones digitales y se han analizado cuáles son sus prestaciones en el caso de un canal ideal (AWGN) cuando el receptor está perfectamente sincronizado. Cuando decimos “perfectamente sincronizado”, queremos decir lo siguiente:

1) El receptor muestrea los pulsos recibidos en los instantes óptimos, que en general son los instantes en los que los pulsos recibidos tienen amplitud máxima. Se habla en este caso de *sincronismo de símbolo* o *timing*. Hay que tener en cuenta que cuando el receptor se enciende nunca conoce cuáles son esos instantes óptimos de muestreo porque el reloj del transmisor y el del receptor no están sincronizados y tanto el uno como el otro desconocen el retardo del canal.

2) En el caso de las comunicaciones paso banda con desmodulación coherente, el receptor utiliza un oscilador local que tiene la misma fase y frecuencia que la portadora sinusoidal recibida. Se habla en este caso de *sincronismo de portadora*. Esta sincronización no es necesaria en el caso de comunicaciones en banda base. Hay que tener en cuenta que al principio el receptor desconoce por completo la fase de la portadora recibida, por la misma razón que desconoce el *timing*: los osciladores del transmisor y el receptor no están sincronizados y el retardo del canal es desconocido. Por otro lado, la frecuencia de la portadora recibida no es perfectamente conocida en recepción porque, aunque los osciladores locales tengan la misma frecuencia nominal, las frecuencias reales son ligeramente diferentes debido a la tolerancia admitida en el proceso de fabricación de los osciladores. Además, la frecuencia de los osciladores presenta derivas térmicas y la frecuencia de la portadora recibida puede experimentar una desviación Doppler si el transmisor y los receptores no están en reposo el uno con respecto al otro.

En este módulo vamos a estudiar qué técnicas se tienen que implementar en un receptor digital para sincronizarse con la señal recibida. También vamos a analizar cómo y cuánto se degradan las prestaciones del sistema cuando, como sucede en realidad, la sincronización del receptor no es perfecta.

Para hacer la explicación más clara, hemos optado por considerar solo el caso de modulaciones lineales paso banda (QAM y PSK). En cualquier caso, los algoritmos de sincronización estudiados en este módulo se pueden adaptar para funcionar con otras modulaciones digitales utilizadas en la práctica. En

concreto, los algoritmos de sincronismo de símbolo que vamos a estudiar se pueden particularizar con facilidad para funcionar con las modulaciones en banda base estudiadas en SC1 (PAM y otros códigos de línea).

En cuanto al modelo de canal, en este módulo didáctico vamos a considerar que el canal es ideal (AWGN). Ello significa que vamos a suponer lo siguiente:

1) El pulso conformador utilizado garantiza que no hay interferencia intersimbólica (ISI) a la salida del filtro adaptado del receptor, siempre que se escojan correctamente los instantes de muestreo (sincronismo de símbolo perfecto). En concreto, vamos a considerar pulsos rectangulares sin retorno a cero (NRZ) y pulsos de Nyquist.

2) El canal no distorsiona los pulsos transmitidos, es decir, los pulsos recibidos son una versión atenuada y retardada de los pulsos transmitidos. Ello significa que la respuesta frecuencial del canal es plana dentro de la banda de la señal, o dicho de otro modo, la respuesta impulsional del canal se puede modelizar como una delta de Dirac.

3) Las etapas analógicas del receptor introducen ruido aditivo blanco gaussiano (AWGN) dentro de la banda de la señal. El nivel espectral del ruido viene dado por $N_0/2 = c \cdot F$, donde $c = 2 \cdot 10^{-21}$ W/Hz es una constante muy pequeña y F se denomina *cifra de ruido* del receptor.

La suposición de canal ideal no es restrictiva porque, cuando el canal no es ideal y aparece ISI en el receptor, el problema de sincronización forma parte de un problema más complejo que se denomina *ecualización de canal* y que abordaremos más adelante en este mismo curso.

Objetivos

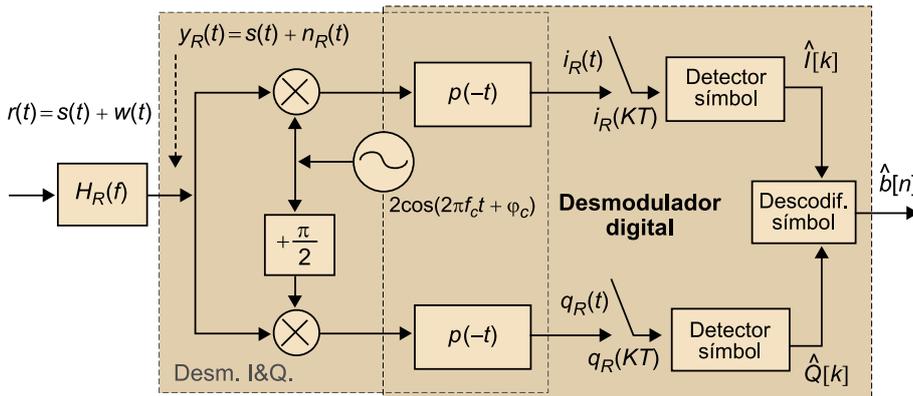
Los objetivos de este módulo didáctico son los siguientes:

- 1.** Conocer cuáles son los parámetros que tiene que sincronizar un receptor digital.
- 2.** Estudiar el impacto que tienen los diferentes errores de sincronismo sobre las prestaciones del sistema.
- 3.** Saber cuál es la teoría matemática con la que se diseñan los algoritmos de sincronización más utilizados.
- 4.** Entender cómo se pueden corregir digitalmente los diferentes errores de sincronismo.
- 5.** Distinguir las dos arquitecturas básicas de un sincronizador (lazo abierto y lazo cerrado).
- 6.** Saber cómo se clasifican los algoritmos de sincronización dependiendo del conocimiento que el receptor tiene sobre los símbolos de información transmitidos (DA, DD, CA y NDA).
- 7.** Conocer algunos ejemplos de algoritmos clásicos de sincronización para cada uno de los parámetros de sincronismo estudiados (fase, frecuencia, *timing* y trama).

1. Motivación

Si recuperamos de la documentación de la asignatura SC1 el esquema general de un receptor digital paso banda (figura 1), veremos que lo primero que se hace es filtrar la señal recibida con un filtro paso banda $H_R(f)$ para eliminar el ruido y las interferencias fuera de banda, y a continuación se baja la señal a banda base multiplicándola por un coseno y un seno de fase φ_c y frecuencia f_c .

Figura 1. Esquema desmodulador QAM



Como hemos dicho en la introducción, el sincronismo de portadora es el encargado de garantizar que estos seno y coseno tengan, dentro de lo posible, la misma fase y frecuencia que la portadora de la señal recibida $r(t)$. Así, pues, si el sincronismo de portadora fuera perfecto, a la salida de los dos filtros adaptados de la figura, con respuesta impulsional $p(-t)$, recuperaríamos por separado las componentes en fase y en cuadratura de la señal recibida:

$$\begin{aligned}
 i_R(t) &= A_c \sum_{n=-\infty}^{\infty} I[n]R_p(t - nT) + i_n(t) \\
 q_R(t) &= A_c \sum_{n=-\infty}^{\infty} Q[n]R_p(t - nT) + q_n(t)
 \end{aligned}
 \tag{1}$$

donde se ha mantenido la notación utilizada en SC1:

- A_c es la amplitud de la portadora recibida.
- $I[n]$ y $Q[n]$ son las componentes en fase (parte real) y en cuadratura (parte imaginaria) del símbolo complejo recibido en el instante nT .
- T es el periodo de símbolo, es decir, cada cuánto tiempo llega un nuevo símbolo (complejo) al receptor.

- $R_p(t) = p(t) * p(-t)$ es el pulso a la salida del filtro adaptado que garantiza que no habrá interferencia intersimbólica (ISI) siempre que se muestree $R_p(t)$ en su máximo ($t = 0$), es decir, $R_p(t)$ se ha diseñado de modo que cumpla que $R_p(0) = 1$ y $R_p(nT) = 0$ ($n \neq 0$). Como hemos explicado en la introducción, el sincronismo de símbolo es el encargado de garantizar, dentro de lo posible, que el muestreo se hace en el máximo de los pulsos recibidos.
- $i_\eta(t)$ y $q_\eta(t)$ son las componentes en fase y en cuadratura del ruido $n_R(t)$ (figura 1), que cumplen que son estadísticamente independientes el uno del otro, aunque no son espectralmente planos debido al efecto del filtro adaptado $p(-t)$.

Como va a quedar claro algo más adelante, antes de continuar la explicación es conveniente juntar las componentes en fase y en cuadratura en una sola señal compleja, que es el llamado *equivalente paso bajo* y que denotamos por $z(t)$:

$$z(t) = i_R(t) + j \cdot q_R(t) = A_c \sum_{n=-\infty}^{\infty} a[n] R_p(t - nT) + \eta(t) \quad (2)$$

donde $j = \sqrt{-1}$ es el número imaginario, y hemos introducido la notación compleja siguiente:

- Símbolos complejos: $a[n] = I[n] + j \cdot Q[n]$
- Ruido complejo: $\eta(t) = i_\eta(t) + j \cdot q_\eta(t)$

Una vez introducida toda la notación necesaria, vamos a considerar qué ocurre cuando el receptor todavía no ha sido sincronizado perfectamente y se presentan los errores de sincronismo siguientes:

- Error de fase (ϕ radianes): la fase del oscilador local del receptor en el instante $t = 0$ vale $\varphi_c = \varphi_o - \phi$, donde φ_o es la fase inicial de la portadora de la señal recibida. El objetivo del sincronismo de portadora es conseguir, dentro de lo posible, que $\varphi_c = \varphi_o$ (es decir, $\phi = 0$).
- Error de frecuencia (ν hercios): la frecuencia del oscilador local del receptor vale $f_c = f_o - \nu$, donde ν es la frecuencia de la portadora de la señal recibida. El objetivo del sincronismo de portadora es conseguir, dentro de lo posible, que $f_c = f_o$ (es decir, $\nu = 0$).
- Error de sincronismo de símbolo o *timing* (τ segundos): los instantes de muestreo son $t_k = kT$, mientras que $kT - \tau$ son los instantes óptimos de muestreo en los que los pulsos recibidos $R_p(t)$ toman el valor máximo. El objetivo del *timing* es conseguir, dentro de lo posible, retardar los instan-

tes de muestreo τ segundos, de modo que el muestreo se produzca en los instantes óptimos.

Considerando ahora los errores de sincronismo, tenemos que el equivalente paso bajo $z(t)$ tiene la expresión siguiente:

$$z(t) = A_c \sum_{n=-\infty}^{\infty} a[n] \exp\{j\phi\} \exp\{j2\pi\nu t\} R_p(t - nT + \tau) + \eta(t) \quad (3)$$

donde $\exp(jx)$ es la llamada exponencial compleja, que se tiene que interpretar como un giro de x radianes en el plano de los números complejos. Hay que recordar también que la igualdad de Euler nos dice que $\exp(jx)$ se puede escribir en coordenadas cartesianas como $\cos(x) + j\sin(x)$. Si ahora tomamos una muestra para cada pulso recibido en los instantes kT (donde k es un número entero), obtendremos la expresión de las muestras a la entrada del detector de símbolos, donde $\eta[k] = \eta(kT) = i_{np}(kT) + j \cdot q_{np}(kT)$ son las muestras del ruido complejo a la entrada del detector de símbolos:

$$\begin{aligned} z[k] = z(kT) &= A_c \sum_{n=-\infty}^{\infty} a[n] \exp\{j\phi\} \exp\{j2\pi\nu T k\} R_p(kT - nT + \tau) + \eta[k] = \\ &= \underbrace{A_c a[k] e^{j\phi} e^{j2\pi\nu k} R_p(\tau)}_{\text{Señal útil}} + \underbrace{A_c \sum_{n \neq k} a[n] \exp\{j\phi\} \exp\{j2\pi\nu T k\} R_p(kT - nT + \tau)}_{\text{Interferencia intersimbólica (ISI)}} + \underbrace{\eta[k]}_{\text{Ruido}} \end{aligned} \quad (4)$$

Se puede demostrar que la amplitud de la parte real e imaginaria del ruido $\eta[k]$ sigue una distribución gaussiana de media 0 y varianza N_0 . Tal como hemos dicho más arriba, se cumple que la parte real y la parte imaginaria de $\eta[k]$ están incorrelacionadas, y por lo tanto son estadísticamente independientes. También se puede demostrar que las muestras de $\eta[k]$ están temporalmente incorrelacionadas, y por lo tanto son estadísticamente independientes.

Examinando ahora con detalle la ecuación (4), podemos entender el impacto de los errores de sincronismo. Analicémoslos cada uno por separado.

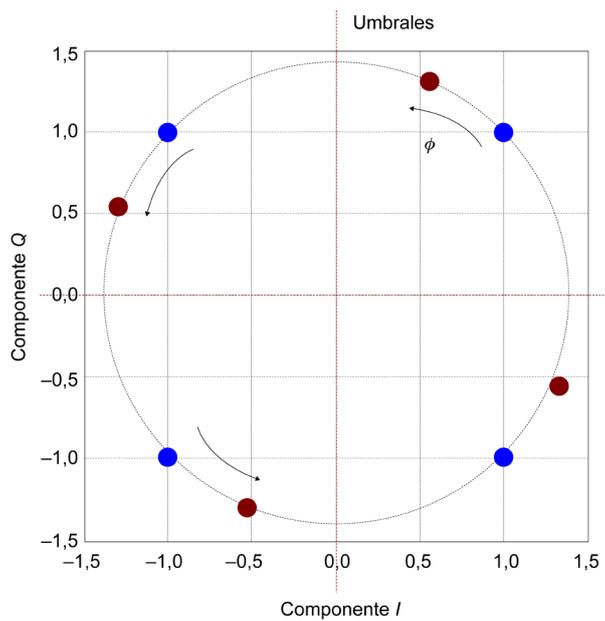
1) Impacto del error de fase (ϕ radianes)

Si dejamos a un lado el término de ISI (segundo término de la ecuación (4)), el error de fase hace que los símbolos recibidos aparezcan girados ϕ radianes. Observando la ecuación (4) se puede ver con facilidad que, debido al error de fase, la secuencia de símbolos recibidos se vuelve $a[k] \exp(j\phi)$ en lugar de $a[k]$, donde, como hemos dicho antes, $\exp(j\phi)$ provoca un giro fijo de ϕ radianes con respecto a la posición de los símbolos en la constelación original.

Nota

La demostración de la ecuación (3) se pide en los ejercicios 1 y 2 de las actividades.

Figura 2. Constelaciones QPSK transmitida y recibida con error de fase



En azul se representa la constelación QPSK transmitida. En rojo, la constelación QPSK recibida afectada por un error de fase de Φ radianes. Las líneas rojas punteadas indican los umbrales de decisión.

Como consecuencia de este giro, algunos puntos de la constelación se acercan a los umbrales de decisión (figura 2), y fijada la E_b/N_0 , la probabilidad de error aumenta (figura 3). De manera equivalente, debido al error de fase, hay que aumentar la relación E_b/N_0 en recepción para mantener la misma probabilidad de error. Este incremento de E_b/N_0 debido a los errores de sincronismo son las pérdidas de sincronismo, que en un receptor bien diseñado no suelen superar 1 dB. Por ejemplo, mirando la figura 3, si la probabilidad de error esperada es 10^{-3} , un error de fase de 10° introduce unas pérdidas de sincronismo de aproximadamente 1,3 dB.

Para enfrentarse a este error de fase, hay varias estrategias:

a) Incluir un algoritmo que corrija automáticamente la rotación causada por el error de fase. Estudiaremos esta alternativa en el apartado 5.

b) Sustituir el desmodulador coherente por un desmodulador que pueda funcionar sin conocer la fase de la portadora recibida. Hay dos opciones:

- La primera es utilizar una modulación (de fase) diferencial, y en vez de codificar la información en la fase de los símbolos, hacerlo en la diferencia de fase entre un símbolo y el símbolo transmitido antes.
- La alternativa es codificar la información en la envolvente de la portadora (ASK) y utilizar en recepción un desmodulador no coherente que calcule la energía de la señal recibida, que es insensible a su fase, y compararla con diferentes umbrales de decisión.

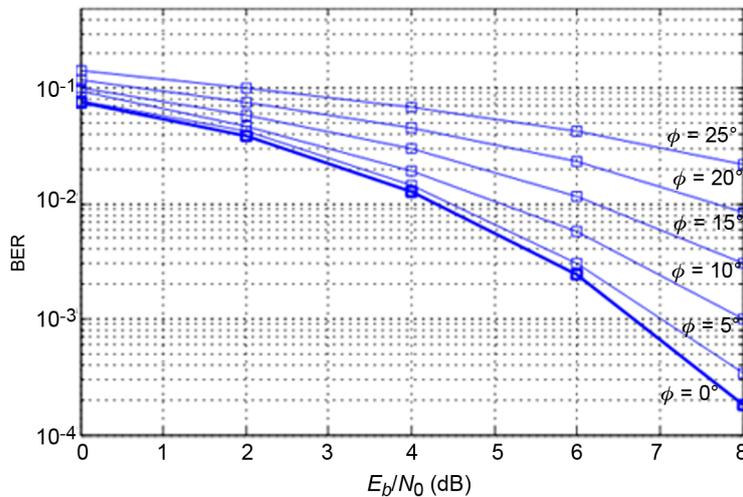
Nota

En el ejercicio 3 de las actividades se propone encontrar la expresión de la probabilidad de error de una modulación BPSK con error de fase ϕ .

Ved también

En el ejercicio 4 de las actividades se presenta como ejemplo la modulación QPSK diferencial (DQPSK) y en el ejercicio 5 se calcula la probabilidad de error de un receptor no coherente en el caso de la modulación ASK.

Figura 3. Probabilidad de error de bit (BER) en función de la E_b/N_0 para diferentes valores del error de fase cuando la modulación transmitida es la QPSK



La relación E_b/N_0 de la figura corresponde a la SNR por bit a la salida del filtro adaptado en condiciones de sincronismo perfecto.

La pregunta es por qué en general se prefiere utilizar esquemas coherentes y tener que sincronizar la portadora. La respuesta es porque los esquemas coherentes consiguen prestaciones mejores, es decir, fijada la SNR, dan una probabilidad de error menor.

Para terminar, cabe recordar que hay otras modulaciones que pueden ser desmoduladas de manera no coherente aunque espectralmente resulten ineficientes y solo se utilicen en casos en los que se tiene que trabajar a muy baja SNR. Este es el caso de la modulación FSK, en la que la información se codifica en la frecuencia de la portadora y la desmodulación se hace de manera no coherente calculando la energía recibida en cada una de las frecuencias y después decidiendo la frecuencia en la que se detecta más energía. Otro ejemplo de modulación que se puede desmodular de manera no coherente es la PPM (*pulse position modulation*), que analizamos en el ejercicio 6 de las actividades.

2) Impacto del error de frecuencia (ν hercios)

Si dejamos a un lado otra vez la ISI y nos fijamos en el primer término de la ecuación (4), el error de frecuencia hace que los símbolos recibidos giren a una velocidad angular de $2\pi\nu$ radianes cada segundo, o dicho de otro modo, cada símbolo recibido está girado $2\pi\nu T$ radianes con respecto al anterior. Observando la ecuación (4), se puede ver con facilidad que, debido al error de frecuencia, la secuencia de símbolos recibidos se vuelve $a[k]\exp(j2\pi\nu Tk)$ en lugar de $a[k]$.

Para enfrentarse a este error, hay varias opciones:

Impacto del error de frecuencia

Observemos que el impacto del error de frecuencia es mayor cuanto mayor es T , es decir, cuanto más baja es la velocidad de señalización $R = 1/T$.

- Utilizar **modulaciones coherentes** e incluir un algoritmo en el receptor que corrija automáticamente el error de frecuencia. Estudiaremos esta alternativa en el apartado 4. Hay que entender que, puesto que el error de frecuencia hace crecer el error de fase a un ritmo de $2\pi\nu$ radianes cada segundo, este error se tiene que corregir obligatoriamente, por pequeño que sea, en el caso de una transmisión continua que dura un tiempo indefinido (por ejemplo, una señal de televisión o radio). En tal caso, aunque el error de frecuencia sea muy pequeño, llegará un momento en el que el error de fase será demasiado grande y las prestaciones del receptor se colapsarán. Por el contrario, si consideramos un sistema TDMA¹, en el que múltiples usuarios comparten la misma portadora y transmiten en instantes diferentes, cada usuario tendrá que transmitir en una ráfaga de duración $T_{Ráfaga}$ todos los datos nuevos que tenga guardados desde la última transmisión. Por lo tanto, en este caso solo se tendrá que corregir el error de frecuencia cuando el incremento de error de fase al final de la ráfaga ($2\pi\nu T_{Ráfaga}$ radianes) sea excesivamente grande.
- Utilizar **modulaciones diferenciales**. En este caso, se puede demostrar que el error de frecuencia se convierte en un error de fase que es constante e igual a $2\pi\nu T$ radianes (ejercicio 7 de las actividades). En el caso de que este error de fase fuera demasiado grande, habría que utilizar un algoritmo de sincronización de fase como los que hemos explicado en el apartado 5. Por lo tanto, podemos concluir que el error de fase ϕ no afecta a las modulaciones diferenciales y que el error de frecuencia ν les afecta como un error de fase constante de valor $2\pi\nu T$.
- Utilizar **modulaciones de amplitud o frecuencia** que faciliten la desmodulación no coherente. El error de frecuencia provoca un error en la detección de la energía recibida que degradará las prestaciones del receptor no coherente, pero en cualquier caso se podrá hacer la desmodulación de la señal siempre que el error de frecuencia sea moderado. Hay que recordar que el desmodulador no coherente no es sensible a errores de fase.

⁽¹⁾ Los esquemas TDMA también se denominan *MCPC (multiple channel per carrier)*.

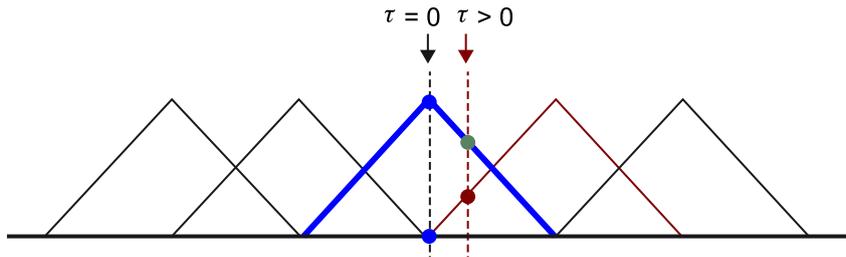
Transmisión continua

Por transmisión continua nos referimos a un sistema FDMA en el que cada transmisor tiene asignada una portadora diferente todo el rato y transmite de manera ininterrumpida por dicha portadora. Estos sistemas también se denominan *SCPC (single channel per carrier)*.

3) Impacto del error de *timing* (τ segundos)

El error de *timing* es el causante del término de ISI de la ecuación (4), ya que $R_p(iT + \tau) \neq 0$ cuando el error de *timing* τ es diferente a 0. Además, el error de *timing* provoca que la parte útil del símbolo detectado en la ecuación (4) pierda amplitud, ya que se cumple que $R_p(\tau) < R_p(0)$, y por lo tanto se reduce la SNR a la entrada del decisor de símbolos. Este doble efecto se muestra en las figuras 4 y 5. En la figura 4 se puede ver la señal temporal a la salida de uno de los filtros adaptados de la figura 1 cuando el pulso transmitido es rectangular.

Figura 4. Señal después del filtro adaptado de pulso rectangular



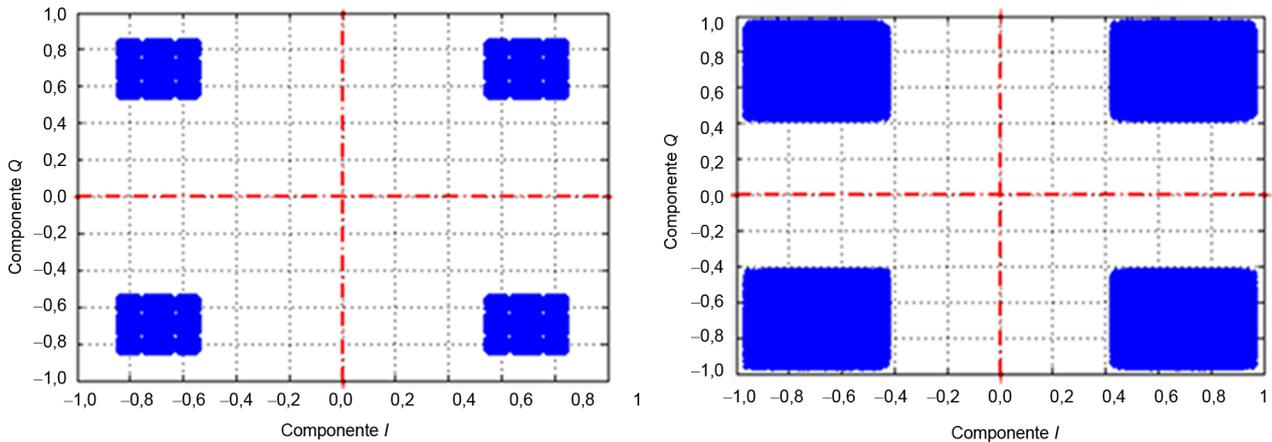
En la figura se representa la señal después del filtro adaptado de una de las dos componentes (I o Q) en el caso de utilizar un pulso rectangular. La línea vertical roja señala el instante de muestreo escogido correspondiente a un error de *timing* igual a τ segundos. El punto verde indica la muestra que se ha tomado del pulso deseado (pulso azul); el punto rojo corresponde a la interferencia intersimbólica del pulso interferente (pulso rojo). En el caso de pulsos de Nyquist, el número de símbolos interferentes (puntos rojos) sería mucho mayor.

En la figura 5 se representa la constelación sin ruido que se obtendría después de muestrear con un error de *timing* τ un pulso de Nyquist con factor de exceso de banda (*roll-off factor*) pequeño. Como se puede apreciar en la constelación de la figura 5, el efecto del error de sincronismo de *timing* es acercar los puntos de la constelación a los umbrales de decisión, y por lo tanto aumentar la probabilidad de error (figura 6).

Nota

En el ejercicio 8 de las actividades, se propone obtener la probabilidad de error con pulso rectangular y un error de *timing* dado.

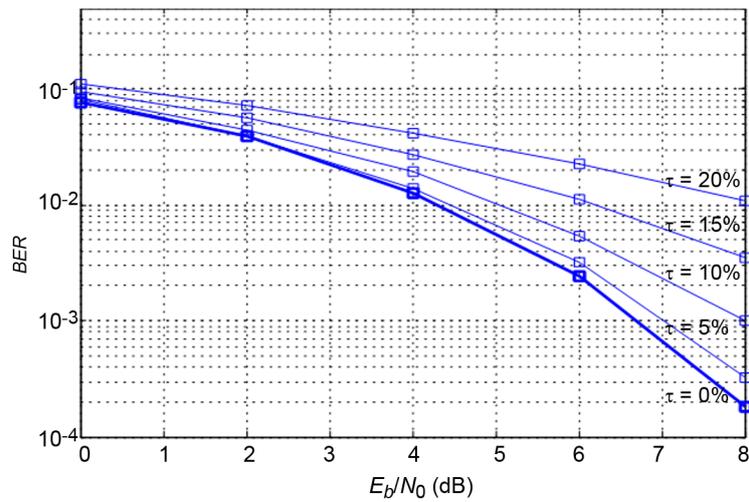
Figura 5. Constelación sin ruido con error de *timing*



En la figura se representa la constelación, **sin ruido**, cuando se muestrea con un error de *timing* de $0,1T$. La modulación es QPSK y el pulso a la salida del filtro adaptado es un pulso de Nyquist de factor de exceso de banda $0,5$ (izquierda) y $0,2$ (derecha). Para simular todas las combinaciones de la ISI se ha simulado un vector de símbolos muy largo. Las líneas rojas discontinuas indican los umbrales de decisión.

Así como hay sistemas de transmisión que no requieren sincronizar la portadora, todos los sistemas de comunicaciones digitales requieren sincronizar los instantes de muestreo con un error no muy grande.

Figura 6. Probabilidad de error de bit según la relación E_b/N_0 para diferentes valores del error de *timing*.



La gráfica muestra la probabilidad de error de bit en función de la relación E_b/N_0 (SNR por bit a la salida del filtro adaptado en condiciones de sincronismo perfecto) para diferentes valores del error de *timing* en el caso de un pulso de Nyquist de exceso de banda 0,2 y la modulación QPSK. En la figura, el error de *timing* se expresa como un porcentaje del periodo de símbolo.

2. Técnicas de sincronización

Como hemos visto en el apartado anterior, antes de hacer la desmodulación de la señal, el receptor tiene que reducir tanto como pueda los errores de fase, frecuencia y *timing* para acercarse así a las prestaciones que la modulación y el código escogidos en teoría permiten. La manera de hacerlo es la siguiente:

- A partir de la señal recibida, el receptor tiene que ser capaz de **estimar** con la máxima exactitud posible el valor de los errores de fase, frecuencia y *timing*. Como la señal recibida es ruidosa, los parámetros estimados tendrán necesariamente un pequeño error de estimación que será aleatorio y causará una leve pérdida de prestaciones del receptor. Esta pérdida se tiene que evaluar y tener presente en el balance de potencias del enlace.
- A partir de estas estimaciones, el receptor tiene que **corregir** los errores estimados de fase, frecuencia y *timing*. Esta corrección se puede hacer modificando la señal recibida, o de modo equivalente, modificando la configuración del mismo receptor. Por ejemplo, si se muestrea con un retardo de $0,1 \mu s$, este error se puede corregir avanzando los instantes de muestreo $0,1 \mu s$, o alternativamente, retardando $0,1 \mu s$ la señal recibida. Cuando se hace la corrección del error de sincronismo también se suele cometer un pequeño error que, como el error de estimación, degradará levemente las prestaciones del receptor y afectará al balance de potencias del enlace.

Ved también

En el subapartado 2.1 se explica la teoría matemática que se usa para construir buenos estimadores, y en el subapartado 2.2 se explican diferentes maneras de corregir digitalmente los errores de fase, frecuencia y *timing*.

Las técnicas de sincronización que vamos a explicar se clasifican según los dos aspectos siguientes:

1) El conocimiento que tenemos sobre los símbolos transmitidos dentro del intervalo de señal observada.

Si el receptor conoce los símbolos transmitidos, se habla de **técnicas asistidas por datos** (*data-aided, DA*). Así pues, estas técnicas se pueden aplicar cuando el estándar de comunicaciones prevé la transmisión de secuencias de símbolos conocidos para asistir al receptor en las tareas de sincronización y equalización, siempre que el receptor ya haya adquirido el sincronismo de trama (ved el apartado 6).

Si los símbolos no son conocidos pero el decisor de símbolos es capaz de tomar decisiones *hard* suficientemente fiables, se habla de técnicas **dirigidas por decisiones** (*decision directed, DD*).

Decisión *soft* y *hard*

Se habla de una decisión *soft* cuando, en vez de dar el valor más probable del símbolo, se da la probabilidad de que este símbolo tome cada uno de sus posibles valores, o equivalentemente, el valor medio del símbolo. Por ejemplo, si sabemos que el símbolo trans-

mitido ha sido +1 o -1 (BPSK) con probabilidades $p(+1) = 0,3$ y $p(-1) = 0,7$, la decisión *soft* sobre este símbolo será el número real $0,3 \cdot (+1) + 0,7 \cdot (-1) = -0,4$ mientras que la decisión *hard* será simplemente -1, ya que este es el más probable de los dos posibles valores.

Para ganar fiabilidad, se pueden utilizar las decisiones *soft* que entrega el decodificador. En este caso, se habla de **esquemas de sincronización asistidos por el código** (*code-aided*, *CA*), también conocidos como *esquemas de turbo-sincronización*.

Por último, si el único conocimiento que tenemos sobre los símbolos transmitidos es que pertenecen a un alfabeto finito (por ejemplo, QPSK), podemos utilizar esquemas de sincronización llamados **esquemas no dirigidos por datos** (*non-data-aided*, *NDA*), que implícitamente utilizan decisiones *soft* sobre los símbolos a la salida del desmodulador.

Aunque los esquemas DA son los que funcionan mejor y es muy habitual que la señal recibida incluya datos conocidos, los esquemas DA no se suelen utilizar porque requieren haber sincronizado la trama y ello implica que el resto de parámetros de sincronismo (fundamentalmente el *timing*) se hayan sincronizado de manera previa.

Ved también

En los apartados 3, 4 y 5 vamos a ver algunos de los algoritmos de sincronización NDA y DD más utilizados.

2) En qué orden se hace la estimación y la corrección de los errores de sincronismo.

Cuando la estimación de un parámetro de sincronismo se hace a partir de la observación de la señal recibida durante un intervalo y a continuación la estimación obtenida se usa para sincronizar la señal, se habla de esquemas en **lazo abierto** (*open loop* o *feedforward*). En esta arquitectura, la estimación siempre se hace utilizando las muestras de la señal recibida original, es decir, antes de aplicar ninguna corrección a la señal.

En cambio, si las muestras de la señal recibida durante el intervalo observado se procesan secuencialmente, alternando estimaciones y correcciones, se habla de esquemas en **lazo cerrado** (*closed loop* o *feedback*). En ese caso, la estimación de los parámetros de sincronismo se hace a partir de muestras de la señal recibida que se han corregido previamente utilizando estimaciones hechas antes. En la figura 7 se representan gráficamente ambas arquitecturas.

Corrección de intervalo

Si hay que aplicar la corrección al mismo intervalo de señal que se ha utilizado para hacer la estimación, este intervalo se ha de guardar en memoria y procesarlo después en diferido.

Muchas veces, el mismo algoritmo se puede implementar en lazo abierto o en lazo cerrado. A la hora de decidir cuál es la opción más adecuada, el diseñador debe tener en cuenta una serie de aspectos:

Ved también

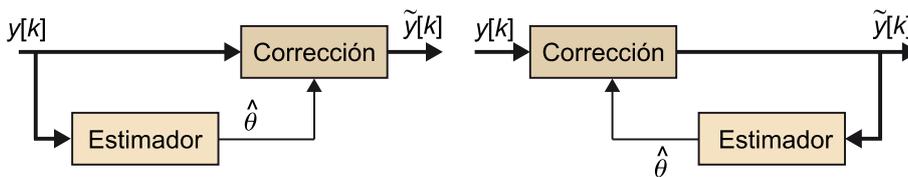
En los apartados 3, 4 y 5 vamos a ver ejemplos de algoritmos de sincronización en lazo abierto y lazo cerrado.

1) los algoritmos dirigidos por datos (DD) requieren una implementación en lazo cerrado;

2) los esquemas en lazo cerrado suelen tener un coste computacional más bajo;

- 3) los esquemas en lazo cerrado son más adecuados cuando el parámetro que queremos sincronizar no es constante sino que cambia con el tiempo;
- 4) los esquemas en lazo cerrado son más adecuados cuando se trata de una transmisión continua;
- 5) los esquemas en lazo abierto son más adecuados para transmisión discontinua (paquetes) porque no tienen los problemas de convergencia de los esquemas en lazo cerrado (transitorios).

Figura 7. Esquema general de un sincronizador en lazo abierto (izquierda) y en lazo cerrado (derecha)



En los dibujos, las muestras de la señal recibida se denotan por $y[k]$ y las muestras de la señal recibida después de corregir el error de sincronismo, por $\tilde{y}[k]$.

2.1. Estimación de parámetros de sincronismo

En este apartado vamos a dar una explicación muy breve de la teoría matemática en la que se basan la mayoría de los algoritmos de sincronización utilizados hoy en día, incluyendo los que explicamos en los apartados 3, 4, 5 y 6 de este módulo. Antes de todo, debemos tener presente que la señal recibida es ruidosa, y por lo tanto es una señal aleatoria que tendremos que tratar estadísticamente.

En concreto, para estimar los errores de sincronismo tenemos que observar la señal recibida durante cierto intervalo y guardar las muestras de la señal en ese intervalo en el vector \mathbf{y} . Este vector \mathbf{y} es una de las posibles realizaciones de un proceso aleatorio del que solo conocemos parcialmente la función densidad de probabilidad porque no sabemos cuál es el valor de los parámetros de sincronismo ni tampoco conocemos el contenido del vector de símbolos transmitidos durante el intervalo observado. Si ahora evaluamos esta función densidad de probabilidad para el caso particular del vector \mathbf{y} observado, obtendremos la llamada función de verosimilitud (*likelihood function*), que es función de los parámetros de sincronismo (ϕ , ν y τ) y del vector \mathbf{a} de símbolos transmitidos, ya que el vector \mathbf{y} que aparece en la ecuación es constante y conocido:

$$p(\mathbf{y}|\mathbf{a}, \phi, \nu, \tau) \quad (5)$$

Si estamos diseñando un algoritmo asistido por datos (DA), el vector de símbolos transmitidos es también conocido, y por lo tanto la función de verosimilitud solo depende de los parámetros de sincronismo. En caso contrario, hay que eliminar la dependencia con el vector \mathbf{a} de símbolos transmitidos. Como los símbolos transmitidos son aleatorios, la manera óptima de eliminar

esta dependencia es calculando el valor medio de la función de verosimilitud con respecto al vector de símbolos \mathbf{a} , como se indica a continuación, donde la media incluye todos los posibles valores del vector \mathbf{a} ponderados por su probabilidad $p(\mathbf{a})$:

$$p(\mathbf{y}|\phi, \nu, \tau) = E_{\mathbf{a}}\{p(\mathbf{y}|\mathbf{a}, \phi, \nu, \tau)\} = \sum_{\mathbf{a}} p(\mathbf{y}|\mathbf{a}, \phi, \nu, \tau) p(\mathbf{a}) \quad (6)$$

En el caso de esquemas no asistidos por datos (NDA) se considera que el vector \mathbf{a} está formado por N símbolos independientes, extraídos aleatoriamente de un alfabeto discreto de dimensión M , por ejemplo, $M = 4$ en el caso de QPSK. Por lo tanto, se considera que hay M^N vectores de símbolos posibles y que la probabilidad de cada uno vale $p(\mathbf{a}) = M^{-N}$. Usando esta distribución en la ecuación (6), se puede demostrar que los esquemas NDA óptimos implícitamente toman decisiones *soft* a la salida del desmodulador y usan estas decisiones $\hat{\mathbf{a}}_{nda}$ para calcular la función de verosimilitud $p(\mathbf{y}|\phi, \nu, \tau) = p(\mathbf{y}|\hat{\mathbf{a}}_{nda}, \phi, \nu, \tau)$.

En el caso de los esquemas DD, se considera que las decisiones a la salida del desmodulador son suficientemente fiables y se pueden sustituir las decisiones *soft* de los esquemas NDA por decisiones *hard*. De este modo, la función de verosimilitud que se ha de considerar en el caso de algoritmos de sincronización DD es $p(\mathbf{y}|\phi, \nu, \tau) = p(\mathbf{y}|\hat{\mathbf{a}}_{dd}, \phi, \nu, \tau)$, donde $\hat{\mathbf{a}}_{dd}$ son las decisiones *hard* a la salida del desmodulador.

Para que las decisiones *hard* sean lo suficientemente fiables, es necesario que el nivel de ruido sea bajo (alta SNR) y que el resto de parámetros de sincronismo se hayan sincronizado previamente, o cuando menos, que se sincronicen todos a la vez. En este caso, cuando las decisiones son lo bastante fiables, los esquemas DD funcionan igual de bien que los esquemas NDA. Por el contrario, cuando la SNR es baja o persisten errores de sincronismo demasiado grandes, los esquemas DD no se utilizan porque sus prestaciones son malas.

Para terminar, los esquemas asistidos por el código (CA) se obtienen a partir de la ecuación (6) teniendo en cuenta que el vector de símbolos transmitidos, \mathbf{a} , ha sido codificado en el transmisor utilizando un código corrector/detector de errores (codificación de canal). Debido a esta codificación, no son posibles todas las secuencias de símbolos a la salida del codificador, y además las secuencias de símbolos que son posibles están formadas por símbolos que no son estadísticamente independientes. Esta dependencia estadística es consecuencia de la redundancia que el codificador genera a su salida.

Los esquemas CA óptimos incorporan todo este conocimiento estadístico sobre el vector de símbolos codificado, \mathbf{a} , por medio de la distribución $p(\mathbf{a})$ que aparece en la ecuación (6). Usando esta distribución, se demuestra que los esquemas CA óptimos toman decisiones *soft* a la salida del descodificador y usan estas decisiones $\hat{\mathbf{a}}_{ca}$ para calcular la función de verosimilitud:

$$p(\mathbf{y}|\phi, \nu, \tau) = p(\mathbf{y}|\hat{\mathbf{a}}_{ca}, \phi, \nu, \tau) \quad (7)$$

Los esquemas CA son los que funcionan mejor cuando no se conoce el vector de símbolos transmitidos y sus prestaciones se acercan mucho a las que se conseguirían con esquemas DA. Aunque tradicionalmente no se han utilizado porque son más complejos, en los últimos tiempos están sustituyendo los esquemas clásicos (NDA y DD) cuando hay que trabajar a muy baja SNR. Hay que tener en cuenta que la SNR de trabajo se ha reducido en los últimos años de manera considerable por la incorporación en todos los estándares de comunicación de códigos de canal mucho más potentes (LDPC y turbocódigos), con los que se puede bajar la SNR de trabajo hasta valores cercanos al valor mínimo establecido por C. E. Shannon.

Si juntamos por comodidad los tres parámetros de sincronismo en el vector $\theta = [\phi, \nu, \tau]$, se puede demostrar que el estimador óptimo de θ es el que maximiza la función de verosimilitud de la ecuación (6). Por esta razón, este estimador se conoce como *estimador de máxima verosimilitud (maximum likelihood, ML)*, y formalmente se calcula de la manera siguiente, donde $\arg \max_x f(x)$ es el operador que calcula el valor de x que maximiza la función $f(x)$:

$$\hat{\theta}_{ml} = \arg \max_{\theta} p(\mathbf{y}|\theta) \quad (8)$$

Se puede demostrar que, siempre que el tiempo de observación sea lo bastante grande, el estimador ML no tendrá sesgo y no habrá ningún otro estimador que tenga una varianza menor que él, es decir:

$$\begin{aligned} \text{BIAS}(\hat{\theta}_{ml}) &= E\{\hat{\theta}_{ml} - \theta_0\} = 0 \\ \text{VAR}(\hat{\theta}_{ml}) &= E\{(\hat{\theta}_{ml} - \theta_0)^2\} \leq E\{(\hat{\theta} - \theta_0)^2\} \end{aligned} \quad (9)$$

En estas expresiones, θ_0 es el valor desconocido que en realidad tienen los parámetros de sincronismo y la última expresión es válida para cualquier estimador $\hat{\theta}$ que podamos considerar.

Hay que destacar otras características del estimador ML cuando el tiempo de observación es grande:

- El error de estimación $(\hat{\theta}_{ml} - \theta_0)$ tiene una distribución gaussiana de media cero (sesgo cero) y varianza mínima, como se indica en la ecuación (9).

Nota

La segunda línea de la ecuación (9) es formalmente incorrecta porque θ es un vector y habría que trabajar con la matriz de covarianza en lugar de con la varianza. Sin embargo, para no complicar la explicación, se ha preferido asumir en este punto que θ es un escalar.

- La varianza del estimador ML coincide con un límite teórico importante que se denomina cota de Cramer-Rao, que es la mínima varianza que puede tener cualquier estimador no sesgado de los parámetros.
- Aunque el estimador ML solo es óptimo cuando el tiempo de observación es muy grande, en general sus prestaciones son también buenas aunque el tiempo de observación sea relativamente corto. Se demuestra que, cuanto más ruidosa es la señal recibida, más largo tiene que ser el tiempo de observación para que el estimador ML sea óptimo.

Como hemos visto, el sincronizador óptimo requiere que se haga la estimación conjunta de los tres parámetros $\theta = [\phi, \nu, \tau]$. En general, esto es demasiado complicado y el receptor suele sincronizar los parámetros uno detrás de otro, empezando habitualmente por la frecuencia de la portadora, siguiendo por el *timing* y terminando por la fase de la portadora. En qué orden se estiman y se corrigen los tres parámetros es un tema crítico que condiciona qué algoritmos de sincronización se usan para cada parámetro. Aunque la solución no es única, en los apartados 3, 4 y 5 siguientes presentamos la que consideramos que es la secuenciación más habitual y explicamos algunos algoritmos de sincronización que se podrían usar en cada caso.

2.2. Corrección de parámetros de sincronismo

En este apartado vamos a explicar cómo se puede corregir digitalmente la fase y el *timing* de la señal recibida a partir de los errores de sincronismo estimados mediante los algoritmos que se explican en los apartados 3, 4 y 5. Explicaremos primero la corrección del *timing* y después la corrección de la fase.

1) Corrección del *timing*

Supongamos que hemos hecho la estimación del error del *timing* y tenemos que avanzar los instantes de muestreo $\hat{\tau}$ segundos para obtener las muestras de la señal recibida en los instantes $t_k = kT_s - \hat{\tau}$. Esta corrección se puede hacer de dos maneras. La primera es actuando directamente sobre el muestreador, obligándolo a retardar los instantes de muestreo $\hat{\tau}$ segundos. La idea es utilizar el valor de $\hat{\tau}$ para modificar los cruces por cero de la onda sinusoidal que marca los instantes de muestreo t_k , tal como se ilustra en la figura 8.

Lectura recomendada

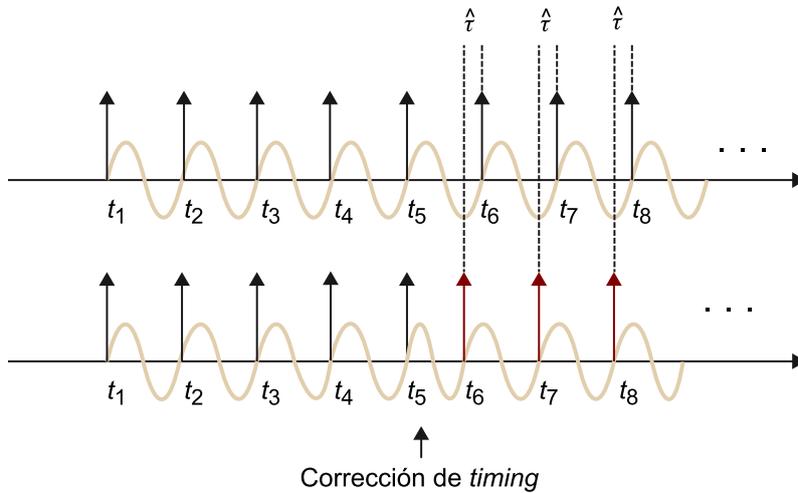
Sobre la cota de Cramer-Rao, podéis consultar la obra siguiente:

Mengali, U.; d'Andrea, A. N. (1997). *Synchronization techniques for digital receivers* (sección 2.4). Nueva York: Plenum Press.

Actuación sobre el muestreador

Esta opción está ligada normalmente a sincronizadores en lazo cerrado como los que presentamos más adelante en la figura 16 del apartado 4 de este módulo.

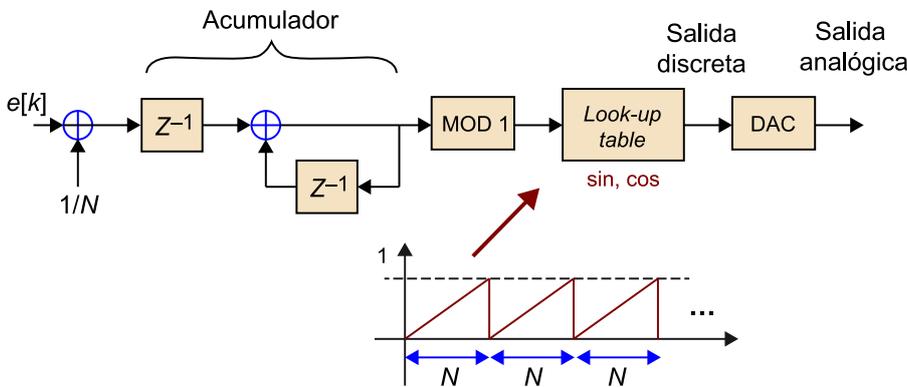
Figura 8. Impulsos enviados al muestreador que marcan los instantes de muestreo t_k



Sobre el eje superior se han representado los impulsos originales y, en el eje inferior, los impulsos avanzados \hat{t} segundos. Para hacerlo, se modifica en el instante t_5 la frecuencia de la onda sinusoidal, que se muestra en color marrón, que marca los instantes en los que se envían los impulsos al muestreador.

El circuito que hace esta operación se denomina *NCO (numerically controlled oscillator)*, que es la implementación digital del circuito electrónico denominado *VCO (voltage controlled oscillator)*. Con el NCO podemos generar cualquier onda periódica (por ejemplo, una sinusoidal) controlando la frecuencia con la señal discreta de entrada (figura 9).

Figura 9. Esquema de un NCO con salida discreta y analógica



La entrada $e[n]+1/N$ se acumula hasta que se llega a 1. Entonces, el acumulador vuelve a cero (suma módulo 1). Cuando la señal de control es $e[n] = 0$, la salida del acumulador es una señal diente de sierra de periodo N como la que muestra la figura. La señal $e[n]$ permite cambiar instantáneamente la fase y la frecuencia de la salida del acumulador, alargando o acortando los periodos de la señal diente de sierra. La señal periódica que sale al final de la NCO se obtiene leyendo una tabla (*look-up table*) en la que se guarda un periodo de la señal periódica esperada, como por ejemplo una señal sinusoidal.

La alternativa para corregir el *timing* es completamente digital y consiste en remuestrear la señal mediante un filtro interpolador. Esto es posible porque la señal útil se ha muestreado cumpliendo el teorema del muestreo, y por lo tanto, es posible reconstruir el valor de la señal en cualquier instante a partir de sus muestras (10):

$$z(t) = \sum_{n=-\infty}^{\infty} z(nT_s)h(t - nT_s) \quad (10)$$

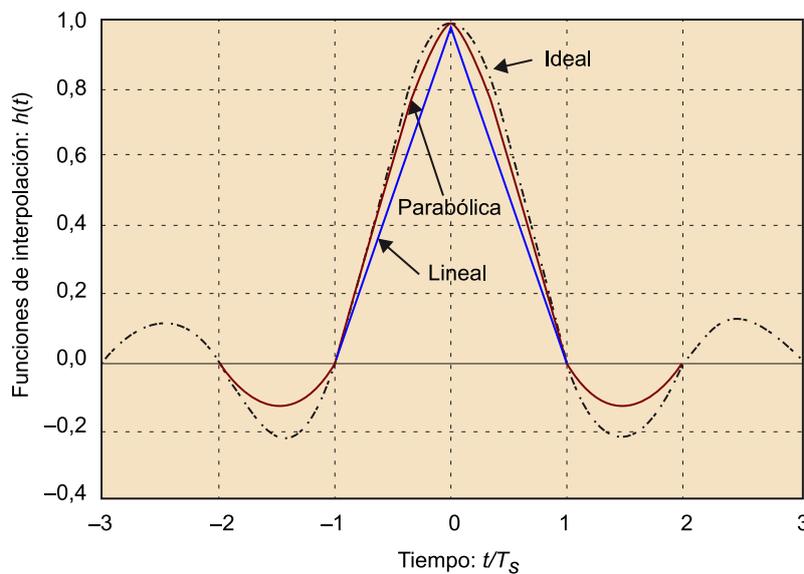
La función interpoladora utilizada en la ecuación (10) idealmente tendría que ser una función sinc:

$$h(t) = \frac{\text{sen}(\pi t / T_s)}{\pi t / T_s} \quad (11)$$

Aunque la duración de la función sinc es en teoría infinita, $h(t)$ se puede aproximar tomando solo los $2K + 1$ lóbulos centrales, con K en torno a 4 o 5.

En el caso de la corrección del *timing*, en general no hace falta utilizar el interpolador ideal y se pueden usar en su lugar interpoladores polinómicos de primer orden (lineal), de segundo orden (cuadrático) o, en algunos casos, de tercer orden (cúbico). El orden del interpolador depende de la distancia entre muestras (T_s) y del error de interpolación que toleremos. En la figura 10 se muestra la respuesta de algunos de los interpoladores más habituales.

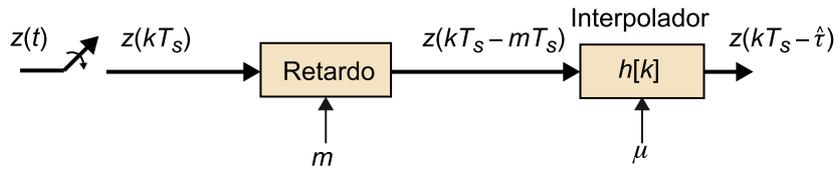
Figura 10. Funciones de interpolación $h(t)$ consideradas: lineal, parabólica e ideal (5 lóbulos)



Utilizando ahora la ecuación (10), podemos remuestrear la señal $z(t)$ en los instantes $t_k = kT_s - \hat{\tau}$. La expresión es la siguiente, donde hemos dividido la corrección de *timing* $\hat{\tau}$ en dos partes: $\hat{\tau} = mT_s + \mu$:

$$z(kT_s - \hat{\tau}) = \sum_{n=-\infty}^{\infty} z(nT_s) h(kT_s - \hat{\tau} - nT_s) = \sum_{n=-\infty}^{\infty} z(nT_s - mT_s) h(kT_s - \mu - nT_s) \quad (12)$$

Mientras que podemos corregir con facilidad la componente entera de $\hat{\tau}$ (mT_s) atrasando m muestras la secuencia de entrada $z(nT_s)$, la componente fraccional de $\hat{\tau}$ ($0 \leq \mu < T_s$) se tiene que corregir mediante un filtro interpolador con respuesta impulsional $h[k] = h(kT_s - \mu)$. El esquema completo se muestra en la figura 11.

Figura 11. Corrección digital del *timing* mediante un filtro interpolador

La señal se retarda $mT_s + \mu$ utilizando el filtro interpolador $h[k]$, que corrige el error fraccional μ . El periodo de muestreo, T_s , tiene que cumplir el teorema de muestreo, es decir, $T_s < 1/2B$, donde B es el ancho de banda de $z(t)$.

2) Corrección de la fase y la frecuencia

Supongamos que disponemos de una estimación del error de fase $\hat{\phi}$ y del error de frecuencia $\hat{\nu}$. El objetivo es corregir la señal multiplicándola muestra a muestra por la exponencial compleja $\exp(-j2\pi\hat{\nu}kT_s - j\hat{\phi}) = \cos(\hat{\phi} + 2\pi\hat{\nu}kT_s) - j \sin(\hat{\phi} + 2\pi\hat{\nu}kT_s)$ que hace girar la fase de la muestra k de la señal en una cantidad $-(\hat{\phi} + 2\pi\hat{\nu}kT_s)$ radianes. Para generar esta señal sinusoidal discreta se puede utilizar una señal NCO como la de la figura 9, ajustando la señal de control $e(n)$ y la constante $1/N$ para que generen un seno y un coseno de la fase y la frecuencia esperadas. Además, si se trabaja en formato de coma fija y se quiere evitar el coste computacional de las multiplicaciones, se puede recurrir al algoritmo llamado *CORDIC* (*coordinate rotation digital computer*), que permite descomponer cualquier rotación de x radianes en múltiples microrrotaciones que se pueden hacer desplazando el valor binario de la muestra de la señal.

3. Sincronismo de frecuencia de portadora

Cuando la señal se recibe con un error de frecuencia grande, lo primero que tiene que hacer el receptor es sincronizar la frecuencia de la portadora para reducir este error por debajo de aproximadamente el 10% de la velocidad de símbolo, es decir, $\nu T < 0,1$. Este valor es orientativo, pero garantiza que las pérdidas debidas a la desadaptación del filtro adaptado no sean excesivas. Como suponemos que en un inicio el error de frecuencia puede ser mayor que $0,1/T$, habrá que estimar este error y corregirlo procesando la señal a la entrada del filtro adaptado. Es decir, no se podrá trabajar a la salida del filtro adaptado hasta que no reduzcamos el error de frecuencia lo suficiente ($\nu T < 0,1$) (ved el ejercicio 2 de las actividades).

Si denominamos $\gamma(t)$ a la señal compleja a la entrada del filtro adaptado, tenemos la expresión siguiente, donde $n(t) = i_n(t) + jq_n(t)$ es el ruido complejo presente a la entrada del filtro adaptado:

$$\gamma(t) = A_c \sum_{n=-\infty}^{\infty} a[n] \exp\{j\phi\} \exp\{j2\pi\phi t\} p(t - nT + \tau) + n(t) \quad (13)$$

Como se puede ver, la expresión de $\gamma(t)$ es la misma que la expresión de $z(t)$ a la salida del filtro adaptado (ecuación (3)) si se sustituye el pulso a la salida del filtro adaptado $R_p(t) = p(t) * p(-t)$ por el pulso a su entrada $p(t)$ y se considera el ruido $n(t)$ a la salida del filtro adaptado en lugar del ruido $\eta(t)$ a la entrada. Se puede demostrar que las componentes en fase y en cuadratura del ruido $i_n(t)$ y $q_n(t)$ son independientes la una respecto de la otra y tienen amplitud gaussiana de media cero y varianza N_0B , donde $N_0/2$ es la densidad espectral de potencia del ruido del receptor $w(t)$ y B es el ancho de banda del filtro paso banda $H_R(f)$ (figura 1).

El ancho de banda de este filtro tiene que ser suficientemente grande para que la señal útil pase sin ninguna distorsión. Eso significa que B debe ser mayor o igual que $B_s + 2\nu_{max}$, donde B_s es el ancho de banda de la señal transmitida y ν_{max} es el máximo error de frecuencia que sabemos que puede tener la señal $\gamma(t)$. Para entender la razón del término $2\nu_{max}$, hay que tener presente que la exponencial compleja $\exp(j2\pi\nu t)$ que aparece en la ecuación (13) hace que el espectro de la señal $\gamma(t)$ no esté centrado en torno a 0 Hz sino que esté desplazado ν hercios, donde este desplazamiento sabemos que está delimitado dentro del intervalo $-\nu_{max} < \nu < \nu_{max}$ (ved la figura 12).

Si queremos implementar digitalmente el sincronizador de portadora será necesario muestrear la señal $\gamma(t)$ a una frecuencia F_s que tiene que ser mayor o igual que el ancho de banda del receptor (B hercios). Por sencillez, escogemos

Nota

Hay que recordar que estamos trabajando con notación compleja en la que la parte real corresponde a la componente en fase, y la parte imaginaria a la componente en cuadratura. Por otro lado, como el filtro adaptado de las componentes en fase y en cuadratura es idéntico, consideraremos que tenemos un único filtro adaptado que procesa la señal compleja a su entrada.

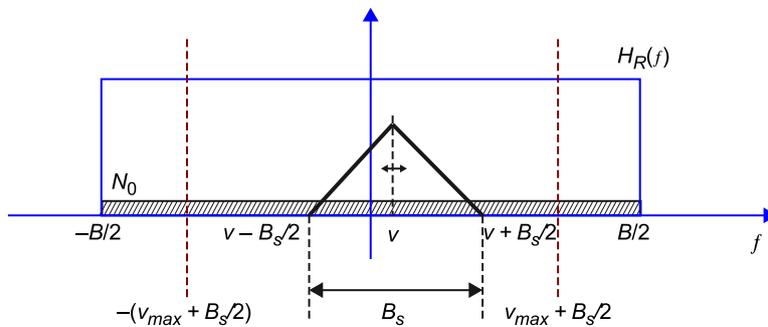
una frecuencia de muestreo múltiple del periodo de símbolo, es decir, $F_s = N_{ss}/T$, donde N_{ss} es el llamado *factor de sobremuestreo*, que nos indica cuántas muestras por símbolo estamos adquiriendo (figura 12).

Una vez fijada la frecuencia de muestreo, podemos muestrear la señal $y(t)$ de la ecuación (13) y obtenemos la señal discreta siguiente, donde $T_s = 1/F_s$ es el periodo de muestreo y $n[k] = n(kT_s)$ son las muestras del ruido a la entrada del filtro adaptado:

$$y[k] = y(kT_s) = A_c \sum_{n=-\infty}^{\infty} a[n] \exp\{j\phi\} \exp\{j2\pi\nu kT_s\} p(kT_s - nT + \tau) + n[k] \quad (14)$$

Para simplificar el diseño del sincronizador, aunque no es imprescindible, consideraremos que las muestras de $n[k]$ son estadísticamente independientes (ruido blanco), es decir que tienen densidad espectral de potencia constante en el intervalo $\pm F_s/2$. Para que esto sea cierto es necesario que la frecuencia de muestreo F_s coincida exactamente con el ancho de banda del receptor B , es decir, $F_s = B$ (figura 12). Teniendo en cuenta todo esto, de ahora en adelante consideraremos que $n[k]$ son muestras independientes de ruido gaussiano (complejo) de media cero y varianza $\sigma_n^2 = 2N_0B = 2N_0F_s = 2N_0N_{ss}/T$, cantidad que corresponde a la suma de la varianza de las componentes en fase y en cuadratura del ruido.

Figura 12. Espectro de la señal a la entrada del filtro adaptado



En la figura se representa el espectro de la señal $y(t)$ a la entrada del filtro adaptado mediante un triángulo, desplazado ν Hz debido al error de frecuencia. La señal $y(t)$ puede tener contenido espectral en el intervalo marcado en rojo, que comprende las frecuencias entre $-(\nu_{max} + B_s/2)$ y $\nu_{max} + B_s/2$, donde ν_{max} es el máximo error de frecuencia que puede tener la señal. En azul, se representa el equivalente paso bajo del filtro receptor $H_R(f)$ que limita el ruido entre $\pm B/2$ y deja pasar la señal útil sin distorsión sea cuál sea el valor de ν , es decir, $B \geq 2\nu_{max} + B_s$. Para no muestrear con *aliasing* el ruido y que las muestras de ruido estén incorrelacionadas, se tiene que cumplir que la frecuencia de muestreo F_s sea exactamente igual a B .

Considerando que el receptor todavía no ha sincronizado la trama y que tampoco se pueden tomar decisiones fiables con respecto a los símbolos transmitidos debido a los errores de sincronismo, para hacer una estimación del error inicial de frecuencia se utiliza un esquema no asistido por datos. En ese caso, lo único que sabemos con respecto a los símbolos transmitidos $a[n]$ de la ecuación (14) es que son independientes y que pertenecen a un alfabeto discreto (constelación) de M valores equiprobables, es decir, $a[n] \in \mathcal{A} = \{a_1, \dots, a_M\}$ con $p(a_i) = 1/M$. A partir de la teoría explicada en el subapartado 2.1, el estimador NDA óptimo se puede formular como aquel que maximiza la función de ve-

rosimilitud de la ecuación (6). En concreto, si tomamos K muestras de la señal recibida $y[k]$, se puede demostrar que la función de verosimilitud que hay que maximizar es la siguiente:

$$p(\mathbf{y}|\phi, \nu, \tau) = C \sum_{\{a[n]\} \in \mathcal{A}} \exp\left(\frac{2}{\sigma_n^2} \sum_{k=1}^K \Re\{x^*[k]y[k]\}\right) \quad (15)$$

Nota

La demostración se pide en el ejercicio 9.

donde C es una constante irrelevante, σ_n^2 es la varianza de las muestras del ruido blanco $n[k]$, $y[k]$ son las muestras de la señal recibida, $x[k]$ son las muestras de la señal reconstruida asumiendo que el valor de los parámetros de sincronismo es $\theta = [\phi, \nu, \tau]$, y $a[n]$ es la secuencia de símbolos transmitidos. Hay que tener presente que el valor real de los parámetros de sincronismo θ_0 de la señal observada y es desconocido y que θ es solo el vector de parámetros que estamos considerando tentativamente. Para encontrar las muestras de la señal reconstruida, $x[k]$, solo necesitamos eliminar de la ecuación (14) el ruido, y obtenemos la expresión siguiente, donde no hemos hecho explícita la dependencia de $x[k]$ con los parámetros de sincronismo y $a[n]$, para no complicar en exceso la notación:

$$x[k] = A_c \sum_{n=-\infty}^{\infty} a[n] \exp\{j\phi\} \exp\{j2\pi\nu kT_s\} p(kT_s - nT + \tau) \quad (16)$$

Mirando la ecuación (15) se nos plantean dos problemas:

1) El primero es que no es posible calcular analíticamente la media con respecto a los símbolos transmitidos $a[n]$. Este problema se soluciona diseñando el sincronizador para que funcione a muy baja SNR , es decir, $\sigma_n^2 \rightarrow \infty$. Puesto que la varianza del ruido es grande, el argumento del exponencial en la ecuación (15) es pequeño y podemos aproximar la exponencial por una serie de Taylor de orden 2, es decir, $\exp(x) \approx 1 + x + x^2/2$. De este modo, se puede hacer fácilmente la media con respecto a los símbolos transmitidos (sumatorio de la ecuación (15)), y cuando se hace se concluye que a baja SNR el resultado es independiente de la modulación digital utilizada. Hay que destacar que la aproximación de baja SNR es suficientemente correcta en la mayoría de los sistemas de comunicaciones digitales actuales.

2) El segundo problema es que la ecuación (15) depende de los errores de sincronismo de fase (ϕ) y *timing* (τ), que todavía no se han sincronizado. Este problema se resuelve modelizando estos parámetros de sincronismo como variables aleatorias uniformes en el intervalo $\pm\pi$ (error de fase) y $\pm T/2$ (error de *timing*). De este modo, el sincronizador de portadora obtenido funciona correctamente con independencia del valor concreto que tomen los errores de *timing* y fase durante el tiempo de observación. Esta estrategia no es óptima,

ya que idealmente habría que hacer una estimación conjunta de todos los parámetros, pero permite secuenciar la sincronización de los diferentes parámetros de sincronismo.

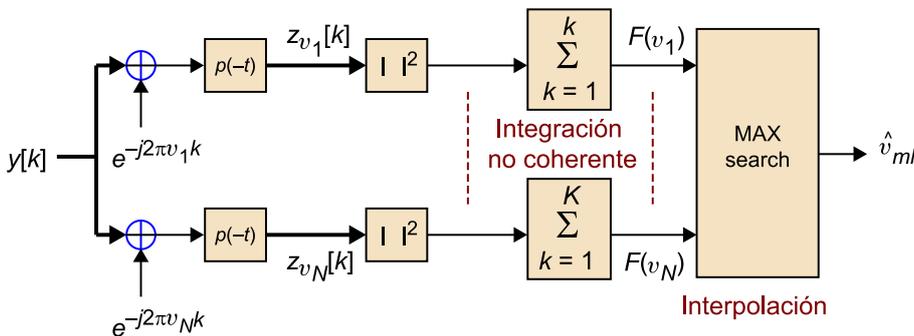
Aplicando ahora las recomendaciones de los dos párrafos anteriores a la función de verosimilitud de la ecuación (15), obtenemos el siguiente estimador del error de frecuencia ν :

$$\hat{\nu}_{ml} = \arg \max_{\nu} E_{\phi, \tau} \{ p(y|\phi, \nu, \tau) \} = \arg \max_{\nu} \sum_{k=1}^K |z_{\nu}[k]|^2 \quad (17)$$

donde $z_{\nu}[k]$ es la salida del filtro adaptado muestreada cada $T_s = T/N_{ss}$ segundos cuando entramos en ella $y[k] \exp\{-j2\pi\nu kT_s\}$, que son las muestras de la señal recibida $y[k]$ corregidas en frecuencia $-\nu$ Hz.

Por lo tanto, el estimador NDA del error de frecuencia ha de encontrar cuál es la corrección de frecuencia que se tiene que aplicar a la señal para maximizar la energía medida a la salida del filtro adaptado durante el intervalo de observación escogido de K muestras.

Figura 13. Esquema del sincronizador NDA del error de frecuencia



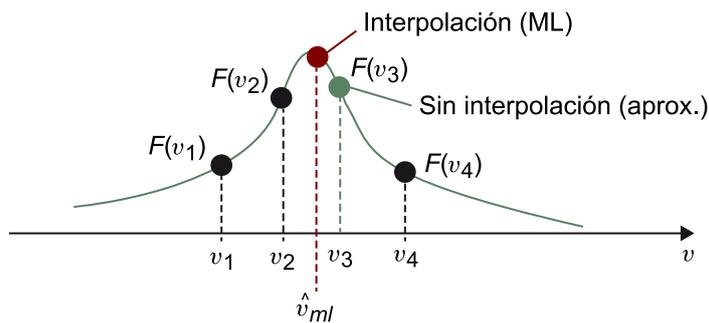
El filtro adaptado $p(-t)$ de la figura se implementa digitalmente y la respuesta impulsional es $h[k] = p(-kT_s)$. En la figura, se habla de integración no coherente porque se integra el módulo al cuadrado de la señal (energía).

Como ν es una variable continua que toma valores en el intervalo $-\nu_{max} < \nu < \nu_{max}$, hay que evaluar la función $F(\nu)$ siguiente en N valores ν_1, \dots, ν_N dentro de este intervalo y después interpolar la función $F(\nu)$ para encontrar el máximo:

$$F(\nu) = \sum_{k=1}^K |z_{\nu}[k]|^2 \quad (18)$$

Dependiendo del valor de N elegido, la interpolación de la función $F(\nu)$ no será necesaria; utilizaremos un interpolador lineal, o tendremos que utilizar un interpolador cuadrático o cúbico (ejercicio 10 de las actividades). En la figura 13 se muestra la implementación del estimador de frecuencia propuesto y en la figura 14, un ejemplo del proceso de interpolación.

Figura 14. Ejemplo de la búsqueda del máximo de $F(\nu)$ para el caso $N = 4$, con y sin interpolación



Para terminar, es interesante destacar los siguientes aspectos del algoritmo de sincronización de portadora presentado (ecuación (17)):

1) El estimador es una función no lineal (cuadrática) de los datos. En concreto, se calcula la energía de la señal sobremuestreada a la salida del filtro adaptado. En cambio, el estimador DA de la frecuencia es lineal con los datos.

2) Como la energía no depende de la fase de la señal, este esquema permite sincronizar la frecuencia de la portadora sin conocer su fase.

3) Como el estimador propuesto trabaja con sobremuestreo ($N_{ss} > 1$), la energía medida a la salida del filtro adaptado tampoco depende del error de *timing*, que únicamente introduce una fase lineal en el dominio de la frecuencia. De este modo, el esquema propuesto puede hacer una estimación de la frecuencia de la portadora sin conocer este error de *timing*.

4) En el caso de transmisión de paquetes, el esquema de la figura 5 permite hacer una estimación del error de frecuencia y a la vez detectar aproximadamente el instante de llegada del paquete. Para hacerlo, hay que estar comparando todo el rato el máximo de $F(\nu)$ con un umbral λ . Cuando $\max_{\nu} F(\nu) > \lambda$, se considera que las primeras K muestras del paquete ya han entrado en el integrador (sumador) de la figura 13 y que el error de frecuencia del paquete vale $\hat{\nu}_{ml} = \arg \max_{\nu} F(\nu)$. Sin embargo, debido al ruido, no es posible con este método establecer con precisión cuál es el principio del paquete, y por lo tanto hay que sincronizar con posterioridad la trama (apartado 6).

5) Hay otros algoritmos NDA para sincronizar la frecuencia de la portadora. Uno de los más destacados es el conocido como *delay and multiply*, que permite hacer la estimación de ν utilizando una fórmula cerrada y evitar de este modo la búsqueda del máximo que requiere la ecuación (17). Este algoritmo se describe en el ejercicio 11 de las actividades.

6) En el caso de la estimación de la frecuencia, el precio que se paga por utilizar un algoritmo NDA en lugar de un algoritmo asistido por datos es muy grande: ¡la varianza del error de estimación de un algoritmo DA es mucho menor! Así

como en el caso DA la varianza decae de manera inversamente proporcional al cubo de K , en el caso NDA decae solo de manera inversamente proporcional a K . Sin embargo, los algoritmos DA requieren que se haya sincronizado la trama (apartado 6) y tienen un margen de funcionamiento mucho más pequeño que los NDA, y por lo tanto no se utilizan para compensar el error de frecuencia inicial.

4. Sincronismo de símbolo

En el apartado anterior hemos sincronizado la frecuencia de la portadora, y el error de frecuencia residual ν se ha hecho suficientemente pequeño. Hay que recordar que consideramos que el error de frecuencia es pequeño cuando no supera el 10% de la velocidad de símbolo ($\nu T < 0,1$) y se puede trabajar a la salida del filtro adaptado sin pérdidas significativas de desadaptación. Una vez estimada la frecuencia, el paso siguiente es estimar y corregir el error de *timing* (τ) para encontrar los instantes de muestreo óptimos a la salida del filtro adaptado.

Como desconocemos el vector de símbolos transmitidos \mathbf{a} y aún nos falta sincronizar la fase de la portadora (ϕ), empezaremos planteando un algoritmo de sincronización del *timing* no asistido por datos. El estimador NDA óptimo del error de *timing* se formula siguiendo los mismos pasos que hemos seguido en el apartado anterior. Primero, tenemos que tomar un vector de K muestras de la señal recibida (ecuación (14)), y después plantear la función de verosimilitud de la ecuación (15) considerando ahora que el error de frecuencia es prácticamente nulo, es decir, $\nu \approx 0$. Entonces, se hace la aproximación de baja *SNR* y se calcula la media con respecto al vector de símbolos transmitidos \mathbf{a} y con respecto al error de fase ϕ , que se modeliza como una variable aleatoria uniforme entre $\pm\pi$ radianes. Como resultado de ello, obtenemos la siguiente aproximación a baja *SNR* del estimador ML del error de *timing*, donde $z_\tau[k] = z(kT + \tau)$ es la salida del filtro adaptado (ecuación (3)) muestreada cada T segundos aplicando un retardo de $-\tau$ segundos (τ tomará valores negativos):

$$\hat{\tau}_{ml} = \arg \max_{\tau} E_{\phi} \{ p(\mathbf{y} | \phi, \nu = 0, \tau) \} = \arg \max_{\tau} \sum_{k=1}^K |z_\tau[k]|^2 \quad (19)$$

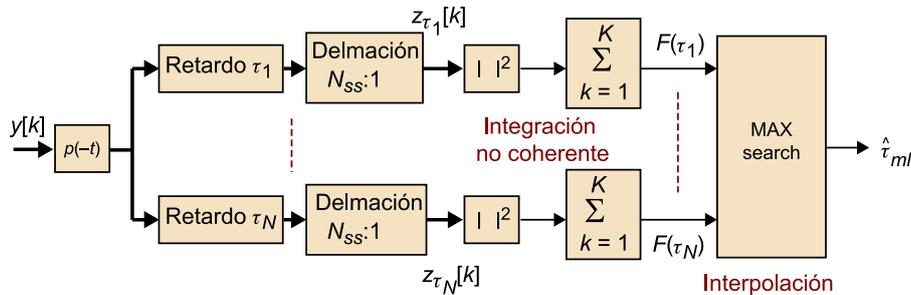
Por lo tanto, el estimador NDA del error de *timing* busca cuál es la corrección de *timing* que hay que aplicar a la señal recibida para maximizar la energía de los K símbolos detectados a la salida del filtro adaptado, que reciben el nombre coloquial de *strokes*.

Como τ es una variable continua que toma valores en el intervalo $-T/2 < \tau \leq T/2$, hay que evaluar la función $F(\tau)$ en N valores τ_1, \dots, τ_N dentro de este intervalo y después interpolar la función $F(\tau)$ para encontrar el máximo:

$$F(\tau) = \sum_{k=1}^K |z_\tau[k]|^2 \quad (20)$$

Si estos N retardos son múltiplos del periodo de muestreo $T_s = T/N_{ss}$, solo hay que retardar o avanzar la señal a la salida del filtro adaptado un número entero de muestras y obtener así hasta $N = 2 N_{ss} - 1$ retardos diferentes. Si los retardos no son múltiplos de T_s , se tiene que remuestrear la señal tal como explicamos en el subapartado 2.2. En la figura 15 se muestra la implementación del estimador de *timing* propuesto.

Figura 15. Esquema del sincronizador NDA del error de *timing* propuesto



El filtro adaptado $p(-t)$ de la figura se implementa digitalmente y su respuesta impulsional es $h[k] = p(-kT_s)$. La operación de diezmado conserva solo una de cada N_{ss} muestras, de modo que a su salida se trabaja a velocidad de símbolo $(1/T_s)$.

Es interesante destacar los siguientes aspectos del algoritmo de sincronización de *timing* presentado (ecuación (19)):

- 1) El estimador es una función no lineal (cuadrática) de los datos. En concreto, se calcula la energía de los símbolos detectados (*strokes*) a la salida del filtro adaptado. Por el contrario, el estimador DA del *timing* es lineal con los datos.
- 2) Como la energía no depende de la fase de la señal, este esquema permite sincronizar el *timing* sin conocer la fase de la portadora. Sin embargo, se asume que el error de frecuencia es suficientemente pequeño ($\nu T < 0,1$).
- 3) Se puede demostrar que el estimador de *timing* propuesto funciona si se muestrea la señal recibida a 2 muestras por símbolo ($N_{ss} = 2$). Sin embargo, puede interesar aumentar el valor de N_{ss} por encima de 2 para trabajar con más de tres retardos que sean múltiplos del periodo de muestreo T_s , y simplificar de este modo el interpolador.

Antes de concluir este apartado, vamos a presentar una posible implementación en lazo cerrado del sincronizador de *timing* obtenido en la ecuación (19). Tal como hemos comentado al principio del apartado 2, los esquemas en lazo cerrado tienen menos complejidad y son más adecuados para seguir las variaciones del parámetro en el caso de paquetes largos o transmisiones continuas.

En un esquema en lazo cerrado, el máximo de la función de verosimilitud se busca iterativamente. Esto significa que cada vez que nos llega una nueva muestra de $z[k]$, la estimación del parámetro se corrige en función del valor de la derivada (gradiente) de la función de verosimilitud, que en el caso del *timing* es la función siguiente:

$$G(\tau) = \frac{\partial F(\tau)}{\partial \tau} = 2 \sum_{k=1}^K \Re \left\{ z_{\hat{\tau}}^*[k] \frac{\partial z_{\hat{\tau}}[k]}{\partial \tau} \right\} \quad (21)$$

Esta función vale cero cuando $\tau = \hat{\tau}_{ml}$ (máxima verosimilitud), toma un valor positivo cuando $\tau < \hat{\tau}_{ml}$, y un valor negativo cuando $\tau > \hat{\tau}_{ml}$. De este modo, $G(\tau)$ nos indica en cada instante en qué dirección nos tenemos que mover para llegar a la solución ML. Antes de continuar, hay que señalar que la derivada de $z_{\hat{\tau}}[k]$ con respecto a τ que aparece en la ecuación (21) se puede aproximar numéricamente de la manera siguiente:

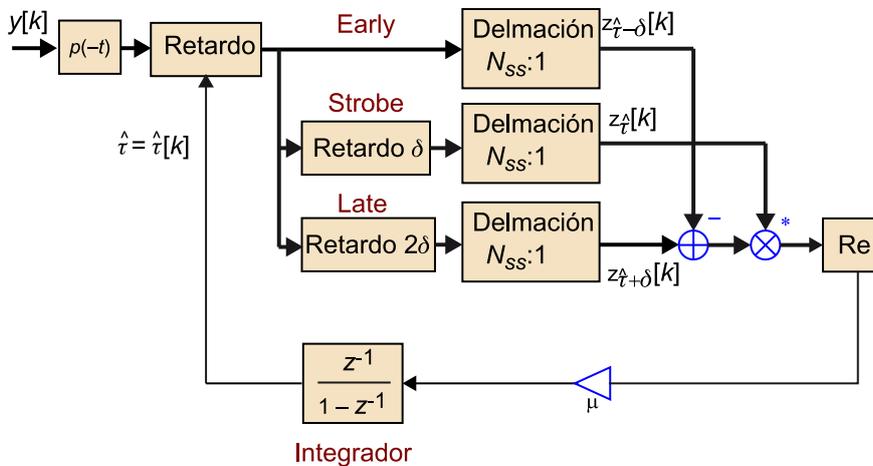
$$\frac{\partial z_{\hat{\tau}}[k]}{\partial \tau} \approx \frac{z_{\tau+\delta}[k] - z_{\tau-\delta}[k]}{2\delta} \quad (22)$$

Tomando como base las ecuaciones anteriores, se puede formular el siguiente sincronizador NDA en lazo cerrado del error de *timing*, que se denomina *early and late* porque calcula la diferencia entre la salida del filtro muestreada δ segundos antes (*early*) y la salida del filtro adaptado muestreada δ segundos más tarde (*late*):

$$\hat{\tau}[k+1] = \hat{\tau}[k] + \mu \Re \left\{ z_{\hat{\tau}}^*[k] \cdot (z_{\hat{\tau}[k]+\delta}[k] - z_{\hat{\tau}[k]-\delta}[k]) \right\} \quad (23)$$

En la figura 16 se muestra la implementación.

Figura 16. Esquema del sincronizador en lazo cerrado propuesto por el *timing* (*early and late*)

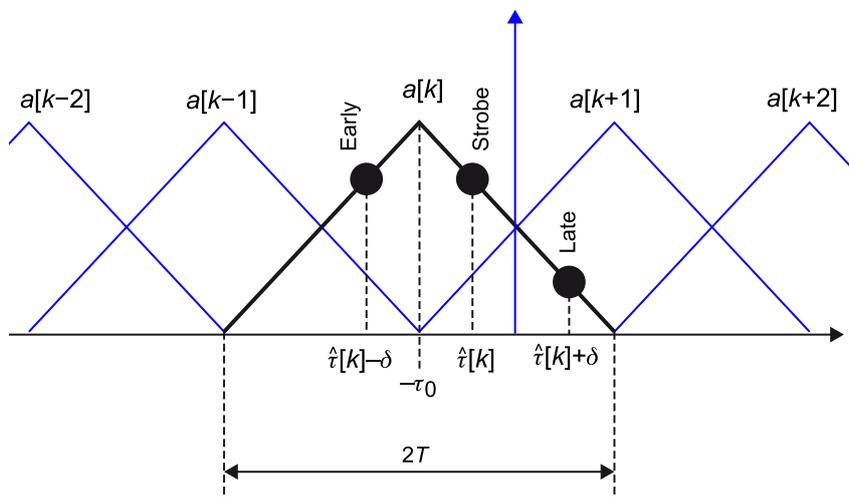


El primer retardador tiene que remuestrear la señal en cada iteración del lazo cerrado (cada T segundos) porque la corrección de *timing* $\hat{\tau}$ no es casi nunca múltiple del periodo de muestreo T_s .

Para entender la ecuación (23), hay que tener presente lo siguiente:

- Se ha eliminado el sumatorio de la ecuación (21) porque esta integración se hace en la ecuación (23) de manera recursiva. En concreto, la ecuación (23) corresponde a la ecuación de un filtro recursivo de orden 1 con respuesta impulsional $h[k] = \exp(-k/\mu)$, donde μ es la constante que determina la memoria del filtro. Si se quiere que el lazo cerrado tenga la misma capacidad de filtrado que el estimador ML de la ecuación (19), que integraba K símbolos, el valor de μ se tiene que fijar aproximadamente a $2/K$, para valores de K suficientemente grandes (por ejemplo, $K = 100$). La demostración se pide en el ejercicio 12 de las actividades.
- Se han eliminado todos los factores constantes de la ecuación (21) porque quedan absorbidos dentro de la constante del lazo μ .
- La ecuación (23) se tiene que entender como una balanza. Cuando no hay error de *timing*, ambos lados de la balanza, $z_{\hat{\tau}[k]}^*[k]z_{\hat{\tau}[k]}[k]$ y $z_{\hat{\tau}[k]}^*[k]z_{\hat{\tau}[k]}[k]$, tienen el mismo valor medio y la balanza se mantiene equilibrada. Si en un momento dado hay un error de *timing* positivo ($\tau_0 + \hat{\tau}[k] > 0$), $z_{\hat{\tau}[k]}^*[k]z_{\hat{\tau}[k]}[k]$ será más pequeño (en valor medio) que $z_{\hat{\tau}[k]}^*[k]z_{\hat{\tau}[k]}[k]$ y el término corrector de la ecuación (23) será negativo para reducir el valor de $\hat{\tau}[k]$ y compensar así el error de *timing*. En la figura 17 se muestra gráficamente este comportamiento.

Figura 17. Señal a la salida del filtro adaptado con un pulso conformador rectangular



En la figura se muestra la señal $z(t)$ a la salida del filtro adaptado en caso de utilizar un pulso conformador rectangular. El triángulo central (más grueso) es el pulso recibido correspondiente al símbolo $a[k]$ y los triángulos azules más finos son los pulsos anteriores y posteriores que generan interferencia (ISI) en el cálculo de las muestras *early*, *strobe* y *late*. Esta interferencia se denomina *ruido propio* (*self-noise*) en los libros de sincronización y limita las prestaciones del sincronizador cuando trabaja a media y alta SNR, mientras que a baja SNR predomina el ruido del receptor.

Figura 17
Como se ve en la figura, si se muestrea a la derecha del instante óptimo, $\hat{\tau}[k] > -\tau_0$, las muestras *late* y *early* están desbalanceadas y el lazo cerrado de la ecuación (23) reduce el valor de $\hat{\tau}[k]$ para balancearlas y muestrear en el pulso siguiente, correspondiente al símbolo $a[k+1]$, más cerca del máximo del pulso.

Hay otros algoritmos NDA para sincronizar el *timing*. Cabe destacar el algoritmo conocido como *square timing*, que permite hacer una estimación del *timing* en lazo abierto con una fórmula cerrada y evitar así la búsqueda del máximo

de la función de verosimilitud que se hace en la ecuación (19). En el ejercicio 13 de las actividades se analiza el funcionamiento de este sincronizador NDA del *timing*.

5. Sincronismo de fase de portadora

Una vez sincronizada la frecuencia de la portadora ($\nu T < 0,1$) y el error de *timing*, el paso siguiente es sincronizar el error de fase de la portadora (ϕ). Como desconocemos el vector de símbolos transmitidos \mathbf{a} , vamos a utilizar otra vez un algoritmo no asistido por datos. El estimador NDA óptimo del error de fase se formula siguiendo los mismos pasos que hemos seguido en los dos apartados anteriores:

- Primero tenemos que tomar un vector de K muestras de la señal recibida (ecuación (14)) y después plantear la función de verosimilitud de la ecuación (16) considerando ahora que el error de frecuencia y el error de *timing* son prácticamente nulos, es decir: $\nu \approx 0$ y $\tau \approx 0$.
- Entonces, se aproxima la función de verosimilitud a baja *SNR* mediante una serie de Taylor y se calcula la media con respecto al vector de símbolos transmitidos \mathbf{a} .

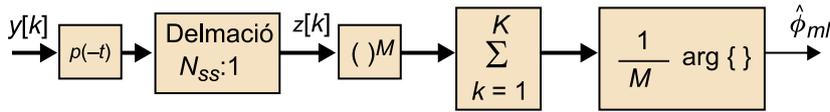
Ahora bien, en el caso de la fase, hay que extender la serie de Taylor hasta el término de orden M porque los términos anteriores son de media cero. El valor de M que escogemos depende de la constelación utilizada: M es el número de niveles en el caso de modulaciones de fase (PSK), mientras que M tiene que ser 4 para modulaciones rectangulares (QAM). Como resultado de esto, obtenemos la siguiente aproximación a baja *SNR* del estimador ML del error de fase (ecuación (24)), que en el caso de la fase se puede resolver analíticamente y da la expresión (25), donde la función $\arg\{x\}^2$ calcula el ángulo de x y $z^M[k]$ se calcula elevando a M el valor de los símbolos detectados a la salida del filtro adaptado.

⁽²⁾Recordemos que $\arg\{x\}$ se calcula como el arcotangente de $\text{Im}\{x\}/\text{Re}\{x\}$.

$$\hat{\phi}_{ml} = \arg \max_{\phi} p(\mathbf{y}|\phi, \nu = 0, \tau = 0) = \arg \max_{\phi} \Re \left\{ e^{-jM\phi} \sum_{k=1}^K z^M[k] \right\} \quad (24)$$

$$\hat{\phi}_{ml} = \frac{1}{M} \arg \left\{ \sum_{k=1}^K z^M[k] \right\} \quad (25)$$

En la figura 18 se muestra la implementación del estimador de fase de la ecuación (25).

Figura 18. Esquema del estimador NDA óptimo de la fase a baja SNR 

Es interesante destacar los siguientes aspectos del algoritmo de sincronización de fase que hemos presentado (ecuación (25)):

- La estimación NDA de la fase requiere en general utilizar no linealidades de orden superior a 2, y por lo tanto la estimación de fase es mucho más sensible al ruido que las estimaciones de frecuencia y *timing* cuando se trabaja a baja SNR . Hay que señalar que el estimador DA de la fase es lineal en los datos.
- Los estimadores NDA de la fase no pueden detectar errores de fase que sean múltiplos de $2\pi/M$ radianes. Esto quiere decir que la salida del estimador de la ecuación (25) será la misma si el error de fase de la portadora vale $\phi + i2\pi/M$ para cualquier valor del número entero i (ved el ejercicio 14 de las actividades). Esta ambigüedad de fase solo se puede resolver mediante estimadores asistidos por datos y se solucionará cuando se sincronice la trama en el apartado 6.

Tal como hemos hecho en el apartado anterior, el estimador ML que hemos obtenido también se puede implementar utilizando un esquema en lazo cerrado. Para hacerlo, tenemos que encontrar la derivada (gradiente) de la función de verosimilitud de la fase, que obtenemos derivando la ecuación (24) con respecto a ϕ , y obtenemos la expresión siguiente, donde $\mathcal{I}\{x\}$ es la parte imaginaria de x , y hemos definido $z_\phi[k]$ como la salida del filtro adaptado conjugada $-\phi$ radianes, es decir, $z_\phi[k] = z[k]e^{-j\phi}$:

$$G(\phi) = \frac{\partial F(\phi)}{\partial \phi} = M \mathcal{I} \left\{ e^{-jM\phi} \sum_{k=1}^K z^M[k] \right\} = M \sum_{k=1}^K \mathcal{I} \left\{ z_\phi^M[k] \right\} \quad (26)$$

A partir del gradiente de la ecuación (26), encontramos el siguiente sincronizador NDA en lazo cerrado de la fase:

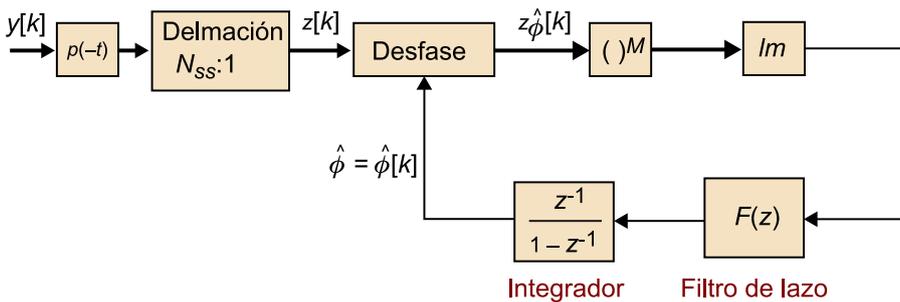
$$\hat{\phi}[k+1] = \hat{\phi}[k] + \mu \mathcal{I} \left\{ z_{\hat{\phi}[k]}^M[k] \right\} \quad (27)$$

Este sincronizador se suele denominar *lazo de Costas* y se representa en la figura 15, considerando un filtro de lazo genérico $F(z)$ en lugar del filtro de lazo $F(z) = \mu$ considerado en la ecuación (27). Este filtro es muy importante cuando hay un error de frecuencia residual. Hay que recordar que, aunque hemos asumido que $\nu = 0$ en el diseño de los sincronizadores del *timing* y la fase, el error residual de frecuencia después de sincronizar el error inicial de frecuencia (apartado 3) puede llegar a ser hasta un 10% de la velocidad de símbolo ($\nu T < 0,1$).

Una manera de compensar este error de frecuencia es sustituyendo el filtro que hemos considerado en la ecuación (27) por el filtro siguiente, donde las constantes μ y γ se tienen que ajustar para garantizar al mismo tiempo que el lazo tenga una memoria efectiva de K símbolos (ecuación (25)) y que el tiempo de convergencia del lazo sea tan corto como sea posible. Para $\gamma = 0$ se habla de un lazo de fase de primer orden (ecuación (27)), mientras que para $\gamma > 0$ obtenemos un lazo de fase de segundo orden:

$$F(z) = \mu + \frac{\mu\gamma}{1-z^{-1}} \quad (28)$$

Figura 19. Implementación en lazo cerrado del sincronizador NDA óptimo de la fase a baja SNR (lazo de Costas)



En la figura se ha considerado un filtro de lazo genérico $F(z)$. En el caso del lazo de orden 1 de la ecuación (27), se considera que $F(z) = \mu$.

Hasta ahora, todos los esquemas de sincronismo que hemos considerado han sido no asistidos por datos. Las razones fundamentales son dos. La primera es que en muchos casos estamos trabajando a baja SNR y las decisiones a la salida del desmodulador son demasiado ruidosas para utilizar esquemas de sincronismo dirigidos por decisiones. La segunda razón es que estas decisiones no pueden ser fiables en ningún caso cuando hay parámetros sin sincronizar. Este era el caso de los sincronizadores de frecuencia y *timing* utilizados en los apartados 3 y 4. Si consideramos ahora que los errores de *timing* y frecuencia son suficientemente pequeños y que la SNR a la que estamos trabajando es razonablemente buena, podremos sustituir el lazo cerrado de fase NDA de la ecuación (27) por el siguiente lazo cerrado DD, donde $\hat{a}[k]$ son las decisiones *hard* a la salida del desmodulador y $z_{\phi}[k] = z[k]e^{-j\phi}$:

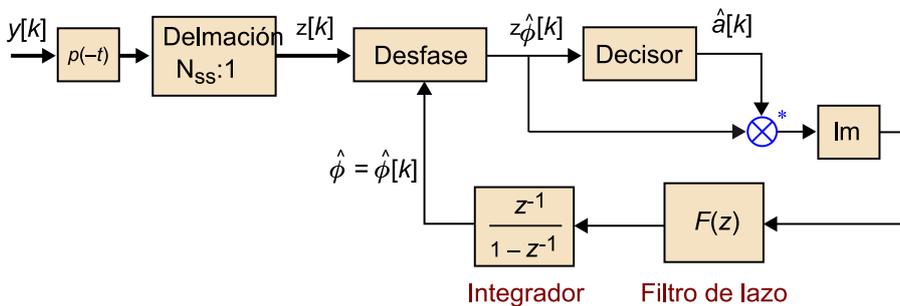
$$\hat{\phi}[k+1] = \hat{\phi}[k] + \mu\mathcal{I} \{ \hat{a}^*[k] z_{\phi}[k] \} \quad (29)$$

Nota

Si utilizamos el sincronizador de fase de la ecuación (27) y hay un error de frecuencia residual ν , la fase estimada tiene un error sistemático igual a νTK radianes. Esto significa que, si necesitamos aumentar la memoria del lazo (K) para filtrar mejor el ruido, el error del sincronizador de fase crecerá de manera proporcional con K .

El sincronizador DD de la fase propuesto se muestra en la figura 19. Para llegar a la ecuación (29), hay que construir la función de verosimilitud de la fase $p(\hat{y}_{dd}, \phi, \nu=0, \tau=0)$ a partir de la observación de la señal recibida durante KT segundos (ecuación (14)) y de las decisiones *hard* \hat{a}_{dd} calculadas a la salida del desmodulador durante el intervalo observado. Hay que señalar que los esquemas DD se tienen que implementar obligatoriamente en lazo cerrado, puesto que las decisiones no son fiables hasta que el error de fase inicial no se ha reducido suficientemente. Igual que en el caso NDA, los esquemas DD no son capaces de resolver la ambigüedad de fase, es decir, convergen a la misma solución para todos los errores de fase que cumplen $\phi + i2\pi/M$ (i entero) (ved el ejercicio 15).

Figura 20. Implementación en lazo cerrado del sincronizador DD propuesto



En la figura se ha considerado un filtro de lazo genérico $F(z)$. En el caso del lazo de orden 1 de la ecuación (29) se considera que $F(z) = \mu$.

6. Sincronismo de trama

Una vez sincronizados todos los parámetros de sincronismo (fase, frecuencia y *timing*), el receptor todavía tiene que hacer dos tareas más antes de poder recuperar correctamente los datos de información. La primera tarea es resolver la ambigüedad de fase que no se ha podido resolver con el sincronizador de fase utilizado en el apartado 5. La segunda es determinar cuál es el índice temporal de los símbolos de información que tenemos que entregar al decodificador. Esta segunda operación se denomina *sincronización de trama* y requiere la transmisión de una secuencia de datos conocidos, llamada *palabra inequívoca* (*unique word*). El receptor tiene que detectar la llegada de esta palabra inequívoca y utilizarla como referencia temporal para localizar dónde están los datos de información, y finalmente procesar la palabra inequívoca para resolver la ambigüedad de fase. Como el receptor conoce el contenido y la localización de la palabra inequívoca, la resolución de la ambigüedad de fase se hace utilizando un esquema asistido por datos.

El problema del sincronismo de trama se puede formular de la manera siguiente. Supongamos que observamos la salida del filtro adaptado, muestreada cada T segundos, durante K instantes consecutivos ($k = 1, \dots, K$). Las muestras obtenidas tienen la expresión siguiente, que son las muestras de la ecuación (4) considerando que los errores de frecuencia y símbolo son prácticamente nulos, pero manteniendo el error de fase por culpa de la ambigüedad de fase:

$$z[k] = A_c e^{j\psi} a[k] + \eta[k] \quad (30)$$

El problema que se plantea es encontrar el índice κ en el que empieza la palabra inequívoca $b[1], \dots, b[N]$ sabiendo que este comienzo tiene que estar comprendido entre $\kappa = 1$ y $\kappa = K - N$ (figura 21). Por lo tanto, el sincronismo de trama consiste en decidir cuál de los $K - N$ posibles valores de κ es el más probable teniendo en cuenta que se desconoce la fase de la portadora (ambigüedad de fase), y en muchos casos, el valor de los $K - N$ símbolos $a[k]$ antes y después de la palabra inequívoca.

Figura 21. Procedimiento que se sigue para sincronizar la trama



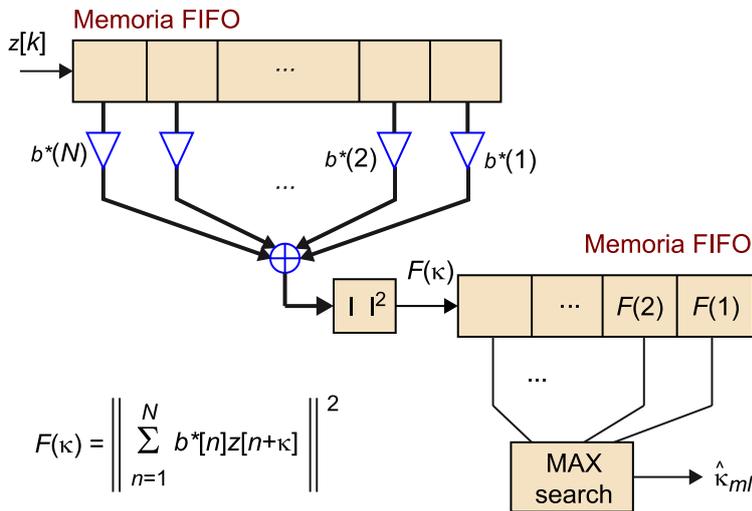
En la figura se muestra el procedimiento que se sigue para sincronizar la trama: hay que buscar el inicio (índice κ) de la palabra inequívoca $b[1], \dots, b[N]$ dentro de un vector de K símbolos $a[1], \dots, a[K]$. Estos símbolos se observan a la salida del filtro adaptado con un error de fase todavía no resuelto y en presencia de ruido.

Utilizando otra vez la teoría del estimador/detector ML y haciendo la aproximación a baja SNR , se puede encontrar el detector óptimo de κ a baja SNR , que tiene la expresión siguiente, donde la constante α depende de la modulación que hayamos considerado:

$$\hat{\kappa}_{ml} = \arg \max_{\kappa \in \{1, K-N\}} \left(\left\| \sum_{n=1}^N b^*[n]z[n+\kappa] \right\|^2 - \alpha \sum_{n=1}^N \|z[n+\kappa]\|^2 \right) \quad (31)$$

Por ejemplo, en el caso de PSK, esta constante vale 1. Aunque la solución óptima requiere considerar los dos términos de la ecuación (31), en la práctica el segundo término no se suele considerar, cosa que da lugar al esquema mostrado en la figura 22. Fijándonos en el primer término, el sincronismo de trama se establece cuando la correlación entre la palabra inequívoca (conocida) y la salida del filtro adaptado es máxima.

Figura 22. Sincronizador ML de la trama a baja SNR



Las muestras a la salida del filtro adaptado $z[k]$ entran en la primera memoria FIFO (registro), empezando por la muestra $z[1]$ y acabando por la muestra $z[K]$. Cuando el registro está lleno con las primeras N muestras de $z[k]$, se calcula $F(1)$, y en los próximos $K - N$ instantes se calculan $F(2)$, ..., $F(K-N)$. Estos valores se guardan en la segunda memoria FIFO, y cuando esta memoria está llena, se busca el valor de k que maximiza $F(\kappa)$.

Una vez sincronizada la trama hay que resolver la ambigüedad de fase. Para hacerlo, hay que maximizar la función de verosimilitud de la ecuación (5) teniendo en cuenta que los errores de símbolo y frecuencia son nulos y ahora conocemos el contenido y posición de la palabra inequívoca $b[1], \dots, b[N]$. En concreto, la función de verosimilitud que hay que maximizar es $p(\mathbf{y}|\mathbf{b}, \phi, \nu = 0, \tau = 0)$, donde el vector \mathbf{b} contiene los N símbolos que forman la palabra inequívoca. Por lo tanto, se encuentra de manera sencilla que el máximo de la función de verosimilitud se tiene que calcular de la manera siguiente, donde $\hat{\kappa}_{ml}$ es la estimación ML de la posición de la palabra inequívoca (ecuación (31)) y hemos llamado $\phi_i = \hat{\phi}_{ml} + i2\pi/M$ a los M posibles valores de la fase con $\hat{\phi}_{ml}$ la estimación (ambigua) de fase calculada en el apartado 5:

$$\hat{\phi}_{ml} = \arg \max_{\phi \in \{\phi_1, \dots, \phi_M\}} \Re \left\{ \sum_{n=1}^N b^*[n] z_{\phi}[n - \hat{k}_{ml}] \right\} \quad (32)$$

Llegados a este punto, el desmodulador ya se ha sincronizado con la señal recibida y está en condiciones de tomar decisiones sobre los símbolos de información transmitidos (si la SNR es suficientemente buena) o, en la mayoría de los casos, entregar al decodificador decisiones *soft* sobre los símbolos (o bits) transmitidos para que sea este quien descodifique el mensaje enviado por el transmisor.

Nota

Dependiendo de la SNR y de la longitud de la palabra inequívoca (N), puede convenir esperar a tener sincronizada la trama y hacer entonces la estimación de la fase utilizando un esquema DA, en lugar de los esquemas NDA/DD del apartado 5. En el ejercicio 16 de las actividades se considera el estimador DA de la fase.

7. Conclusiones

En los apartados de este módulo hemos estudiado por qué siempre tenemos que sincronizar el receptor con la señal recibida antes de descodificar la información enviada por el transmisor. Hemos diseñado los algoritmos de sincronización considerados a partir de la teoría del estimador de máxima verosimilitud (ML), que permite obtener estimadores óptimos de los parámetros de sincronismo cuando el número de datos procesados es grande y la señal recibida es bastante ruidosa (baja SNR). Hemos implementado estos estimadores utilizando esquemas en lazo abierto y lazo cerrado, y hemos analizado las ventajas e inconvenientes de ambas alternativas.

Por otro lado, hemos considerado diferentes algoritmos de sincronización según el conocimiento que el receptor tiene sobre la secuencia de símbolos transmitidos (algoritmos DA, DD, CA y NDA). Los algoritmos no asistidos por datos (NDA) han recibido una atención especial porque permiten hacer la estimación de todos los parámetros de sincronismo cuando no se conoce la posición de los símbolos dentro de la trama (sincronismo de trama), y por lo tanto no se pueden utilizar esquemas asistidos por datos (DA). En el caso de la fase, hemos estudiado también un sincronizador dirigido por decisiones (DD), que se puede aplicar siempre que las decisiones del desmodulador sean suficientemente fiables (SNR alta). Para terminar, hemos presentado el algoritmo DA utilizado habitualmente para sincronizar la trama y resolver entonces la llamada *ambigüedad de fase*.

El objetivo de este módulo didáctico no ha sido en ningún caso conocer todos los algoritmos de sincronización que podemos encontrar en la bibliografía, que son muchos y están ampliamente documentados, sino conocer cómo se diseñan y se implementan en la actualidad esos algoritmos en la práctica. Con este propósito, hemos seleccionado algunos de los algoritmos de sincronización más conocidos y los hemos utilizado para sincronizar el receptor desde el momento en el que lo encendemos. Por ello, hemos sincronizado uno por uno todos los parámetros de sincronismo y hemos dejado finalmente el receptor en condiciones de descodificar el mensaje de información.

Actividades

1. Considerad que el receptor no está sincronizado con la señal recibida y los errores de sincronismo son ϕ (fase), ν (frecuencia) y τ (*timing*). Demostrad que la ecuación (3) nos da el equivalente paso bajo a la salida de los filtros adaptados de la figura 1. Considerad que el error de frecuencia ν es mucho menor que la velocidad de símbolo $1/T$, y por lo tanto podemos considerar que no distorsiona la forma de los pulsos recibidos. Si lo necesitáis, utilizad las identidades trigonométricas siguientes:

$$2\cos(A)\cos(B) = \cos(A-B) + \cos(A+B)$$

$$2\sin(A)\cos(B) = \sin(A-B) + \sin(A+B)$$

$$2\sin(A)\sin(B) = \cos(A-B) - \cos(A+B)$$

2. Encontrad cuál es la pérdida de SNR a la salida del filtro adaptado (muestreando cada T segundos) cuando hay un error de frecuencia ν . Considerad que el pulso conformador es rectangular sin retorno a cero y que el error del sincronismo de símbolo es nulo. Tomando como base el resultado, calculad la degradación si $\nu T = 0,1$.

3. Encontrad la probabilidad de error de la modulación BPSK cuando se desmodula con un error de fase constante ϕ . Asumid que no hay ningún otro error de sincronismo. Expresad el resultado en función de ϕ y la relación E_b/N_0 recibida.

4. La modulación DQPSK es una modulación diferencial de 4 niveles donde los símbolos transmitidos son $b[n] = b[n-1]d[n]\exp(j\pi/4)$ con $d[n] \in \mathcal{A} = \{1, \exp(j\pi/2), \exp(j\pi), \exp(j3\pi/2)\}$ los símbolos de información. Proponed la manera de recuperar en recepción los símbolos de información $a[n]$ a partir de los símbolos $b[n]$ recibidos. Demostrad que el esquema propuesto recupera los símbolos de información correctamente, aunque los símbolos recibidos $b[n]$ estén girados un ángulo ϕ . Asumid que no hay ningún otro error de sincronismo ni ruido.

5. La modulación ASK (*amplitude shift keying*) es una modulación de amplitud digital. En el caso de la modulación ASK de dos niveles, los símbolos transmitidos $a[n]$ pertenecen al alfabeto $\mathcal{A} = \{0, 1\}$. Considerad que los símbolos recibidos han experimentado una rotación de ϕ radianes y encontrad un esquema de desmodulación no coherente que sea insensible al error de fase ϕ y que recupere los símbolos transmitidos $a[n]$ a partir de las muestras $z[k]$ a la salida del filtro adaptado. Asumid que no hay ningún otro error de sincronismo. Calculad entonces la probabilidad de error de ASK y comparadla con la que se obtendría en BPSK utilizando un desmodulador coherente con un error de fase ϕ (ejercicio 3). Si lo necesitáis, tened en cuenta que, si las muestras del ruido n son variables aleatorias gaussianas complejas de media cero y varianza σ^2 , su módulo al cuadrado, $u = |n|^2$, sigue una distribución exponencial de media σ^2 , es decir, $f_u(u) = \sigma^{-2}\exp(-\sigma^2 u)$. Por otro lado, si lo necesitáis, considerad que $|n| \ll 1$ (SNR alta) y haced la aproximación $|n|^2 \ll |n|$.

6. La modulación PPM (*pulse position modulation*) es una modulación en la que el periodo de símbolo se divide en M intervalos de duración T/M y la información se codifica en la posición del pulso transmitido, es decir, el pulso transmitido (de duración T/M segundos) se retarda $a[k]$ segundos, donde $d[k] \in \mathcal{A} = \{0, 1, \dots, M-1\}$ es la secuencia de símbolos de información. Si nos centramos en el caso de 2 niveles ($M = 2$), encontrad un esquema de desmodulación no coherente que sea insensible a los errores de fase y permita recuperar los símbolos $a[k]$ a partir de la señal recibida $y_R(t)$ (figura 1). Considerad que la señal recibida tiene un error de fase de ϕ radianes, pero que el resto de errores de sincronismo vale cero.

7. Volviendo a la modulación diferencial DQPSK, demostrad que los símbolos de información recuperados por el desmodulador del ejercicio 4 son $a[n]\exp(j2\pi\nu T)$, en lugar de $a[n]$, cuando los símbolos recibidos son $b[n]\exp(j(2\pi\nu T + \phi))$, en lugar de $b[n]$. Considerad que el sincronismo de *timing* es perfecto.

8. Calculad la probabilidad de error de la modulación BPSK en el caso de pulso conformador rectangular cuando el desmodulador muestrea la señal recibida con un error constante de τ segundos. Considerad que no hay ningún otro error de sincronismo. Dejad el resultado expresado en términos de τ , el periodo de símbolo T y la E_b/N_0 de la señal recibida.

9. Demostrad que la función de verosimilitud en el caso NDA es la indicada en la ecuación (15), donde el contenido de $x[n]$ se da en la ecuación (16). Para hacerlo, empezad buscando la función de verosimilitud $p(y|a, \phi, \nu, \tau)$ que aparece en la ecuación (6) a partir de las muestras complejas $y[k] = x[k] + n[k]$ a la entrada del filtro adaptado (ecuación (14)). Recordad que las muestras del ruido complejo $n[k]$ son gaussianas de media cero y varianza σ_n^2 .

10. Se quiere calcular el valor de x que hace máxima la función $F(x) = 6 - (x - 3,3)^2$ a partir de los valores $F(1) = 0,71$, $F(2) = 4,31$, $F(3) = 5,91$, $F(4) = 5,51$ y $F(5) = 3,11$ utilizando un

interpolador cuadrático (parabólico). Calculad el error de interpolación y comparadlo con el error de un interpolador lineal. Repetid el ejercicio considerando que ahora se dispone de 9 valores de $F(x)$ correspondientes a los puntos $x = 1; 1,5; 2; \dots; 4,5$ y 5.

Nota: dados 3 puntos en torno al máximo de $F(x)$, el interpolador cuadrático busca la parábola que pasa por los tres puntos y calcula el máximo de $F(x)$ como el valor de x en el que la parábola es máxima.

11. Demostrad que el estimador que se presenta a continuación hace una estimación correcta de ν a partir de las muestras $y[k]$ de la señal a la entrada del filtro adaptado, muestreando cada $T_s = T/N_{ss}$ segundos (ecuación (14)). Este estimador se conoce como *delay and multiply* porque multiplica $y[k]$ por una versión retardada de $y[k]$:

$$\hat{\nu}_{d\&m} = \frac{1}{2\pi T_s} \arg \left(\sum_{k=1}^K y[k] y^*[k-1] \right) \quad (33)$$

Considerad que no hay ruido, y para simplificar, asumid que el error de *timing* es nulo. Haced el estudio considerando que todavía no se ha sincronizado la fase de la portadora y hay un error de fase ϕ .

Ayuda: calculad el valor medio con respecto a $a[k]$ del producto $y[k] \cdot y[k-1]$ teniendo en cuenta que $a[n]$ es una secuencia de símbolos incorrelacionados. Las variaciones de $y[k] \cdot y[k-1]$ con respecto al valor medio calculado se consideran ruido generado por el mismo estimador (ruido propio o *self-noise*) y no las vamos a considerar en el ejercicio.

12. Considerad el filtro recursivo de la ecuación (23), que podemos escribir de la manera siguiente: $\hat{\tau}[k+1] = \hat{\tau}[k] + \mu \epsilon[k]$, donde $\epsilon[k]$ es la corrección de símbolo que calcula el lazo en la iteración k .

Como hemos explicado, $\epsilon[k]$ es una medida del error del sincronismo de símbolo que se puede escribir como $\epsilon[k] = (\tau - \hat{\tau}[k]) + e_{in}[k]$, donde $e_{in}[k]$ es el error en la iteración k debido al ruido y el término de ruido propio (ved el pie de la figura 17). El objetivo de este ejercicio es estudiar cómo el lazo cerrado de la ecuación (23) filtra el término de ruido $e_{in}[k]$.

a) Para hacerlo, os pedimos que busquéis la función de transferencia (transformada Z) del lazo cuando la entrada es $e_{in}[k]$ y la salida es el error de sincronización del lazo en la iteración k , que llamaremos $e_{out}[k] = \tau - \hat{\tau}[k]$.

A partir de la función de transferencia calculada, encontrad la respuesta impulsional del lazo y calculad la varianza de los errores de sincronización $e_{out}[k]$ en función de μ y de la varianza del ruido de entrada $e_{in}[k]$. Considerad que $e_{in}[k]$ tiene de media cero y está incorrelacionado temporalmente.

b) Si en vez del filtro recursivo de la ecuación (23) se adopta un esquema en lazo abierto y los errores $e_{in}[k]$ se filtran mediante un integrador de K muestras, encontrad cuál es la varianza del error de sincronización $e_{out}[k]$ en función de K y la varianza del ruido de entrada $e_{in}[k]$. Para terminar, comparad el resultado obtenido en ambos casos (lazo cerrado y lazo abierto) y encontrad la relación entre μ y K que garantiza la misma varianza a la salida. Si es necesario, considerad que $K \gg 1$, o equivalentemente, $\mu \ll 1$.

13. Demostrad que el estimador que se presenta a continuación hace una estimación correcta del símbolo a partir de las muestras $z[k]$ de la señal a la salida del filtro adaptado (ecuación (3)), muestreando cada $T_s = T/N_{ss}$ segundos. Este estimador se conoce como *square timing* y funciona de la manera siguiente:

$$\hat{\tau}_{st} = -\frac{T}{2\pi} \arg \left(\sum_{k=1}^K \|z[k]\|^2 \exp(-j2\pi k / N_{ss}) \right) \quad (34)$$

Considerad que no hay ruido y el error de frecuencia es nulo.

Ayuda: calculad el valor medio respecto a $a[k]$ del término $|z[k]|^2$ teniendo en cuenta que $a[n]$ es una secuencia de símbolos incorrelacionados. Las variaciones de $|z[k]|^2$ con respecto al valor medio calculado se consideran ruido generado por el mismo estimador (ruido propio) y no las tendremos en cuenta en el ejercicio. Para entender qué hace el estimador hay que

darse cuenta de que el valor medio de $|z[k]|^2$ es una secuencia periódica de periodo $N_{ss} = T/T_s$, y que por lo tanto tiene armónicos a frecuencias múltiples de $1/N_{ss}$.

14. Demostrad que el estimador NDA de la ecuación (25) da la misma salida para cualquier error de fase $\phi + i\pi/M$ (i entero).

15. Demostrad que el estimador DD de la ecuación (29) da la misma salida para cualquier error de fase $\phi + i\pi/M$ (i entero).

16. Una vez conozcamos la posición de la palabra inequívoca podemos hacer la estimación de la fase mediante el estimador DA siguiente, donde $\arg\{x\}$ es el ángulo del número complejo x y $z[n]$ son las muestras a la salida del filtro adaptado (muestreado cada T segundos):

$$\hat{\phi}_{ml} = \arg \left\{ \sum_{n=1}^N b^*[n] z[n - \hat{\kappa}_{ml}] \right\} \quad (35)$$

En este ejercicio, os pedimos que demostréis que, asumiendo que la trama se ha sincronizado correctamente ($\hat{\kappa}_{ml}$ es correcto), este es el estimador ML de la fase cuando se conocen los N símbolos de la palabra inequívoca $b[1], \dots, b[N]$.

Ayuda: calculad la función densidad de probabilidad de la observación z que contiene las N muestras $z[n - \hat{\kappa}_{ml}] = A_c \exp(j\phi) b[n] + \eta[n]$ ($n = 1, \dots, N$). Tened en cuenta que las muestras del ruido $\eta[n]$ son gaussianas de media cero y varianza N_0 (ecuación (4)) y están temporalmente incorrelacionadas.

Bibliografía

Bibliografía básica

Mengali, U.; d'Andrea, A. N. (1997). *Synchronization techniques for digital receivers*. Nueva York: Plenum Press.

Bibliografía complementaria

Meyr, H.; Moeneclaey, M.; Fichtel, S. A. (1998). *Digital communications receivers. Synchronization, channel estimation and signal processing*. Nueva York: Wiley.

Artículos de interés

Noels, N.; Lottici, V.; Dejonghe, A.; Steendam, H.; Moeneclaey, M.; Luise, M.; Vanderdorpe, L. (2005, febrero). "A theoretical framework for soft-information-based synchronization in iterative (turbo) receivers". *Eurasip Journal on Wireless Communications and Networking*.