



UNIVERSIDADE DA CORUÑA

FACULTADE DE INFORMÁTICA

*Departamento de Electrónica e
Sistemas*

PROXECTO DE FIN DE CARREIRA DE ENXEÑERÍA
INFORMÁTICA

***Métodos de supresión de
interferencias basados en el
criterio de máxima verosimilitud
para sistemas de comunicaciones
multiusuario con modulación
multiportadora***

Autor: Manuel A. Vázquez López

Tutor: Joaquín Míguez Arenas

A Coruña, Xullo de 2003

A mis padres. Sobran los motivos.

Agradecimientos

A Joaquín Míguez por soportarme durante estos últimos meses y resolverme las dudas más variopintas que me surgieron durante la elaboración de este trabajo.

Resumen

En el presente proyecto se propondrá una nueva aproximación para la cancelación de interferencias en un sistema de comunicaciones de acceso múltiple por división de código con modulación multiportadora. Se basa en la aplicación del principio de máxima verosimilitud para seleccionar los coeficientes de un filtro lineal capaz de eliminar las interferencias de acceso múltiple, igualar el canal y mitigar el ruido térmico.

La implementación de la solución obtenida mediante el criterio de máxima verosimilitud se hará a través del algoritmo Space Alternating Generalized Expectation Maximization (SAGE), que es una metodología iterativa muy adecuada para resolver este tipo de problemas.

Una vez aplicado dicho algoritmo, resulta un receptor lineal que muestra un rendimiento prácticamente igual al del detector multiusuario de error cuadrático medio mínimo, y muy eficiente en términos de uso del canal, es decir, se consigue una superior tasa de transferencia de información por símbolo transmitido en comparación con los receptores multiusuario convencionales.

Palabras clave

CDMA, MC-CDMA, OFDM, máxima verosimilitud, algoritmo EM, cancelación de interferencias, detección multiusuario

Índice general

1. Introducción	1
1.1. Métodos de acceso múltiple	2
1.2. CDMA	5
1.2.1. Fundamentos de CDMA	7
1.2.2. Privacidad de la señal CDMA	9
1.2.3. Interferencias de banda estrecha	11
1.3. Modulaciones multiportadora	11
1.4. Sistemas CDMA multiportadora	16
2. Modelo de señal	19
2.1. El transmisor	19
2.2. El canal de comunicaciones	21
2.3. El receptor	22
2.4. Conclusiones	23
3. Técnicas clásicas	25
3.1. Receptores lineales	26
3.1.1. Detector decorrelador	27
3.1.2. Detector MMSE	28
3.1.3. Detector LS	31
3.1.4. Implementación adaptativa del criterio LS: algoritmo RLS	33

<i>ÍNDICE GENERAL</i>	v
3.2. Conclusiones	38
4. Supresión de interferencias basada en ML	41
4.1. Estimación de los parámetros del receptor	43
4.2. Supresión semiciega de interferencias	45
4.2.1. El algoritmo EM	46
4.2.2. Supresión semiciega de interferencias aplicando EM	47
4.3. Conclusiones	56
5. Simulaciones	61
5.1. Parámetros de la simulación	61
5.2. Breve descripción de los algoritmos simulados	65
5.3. Resultados numéricos	67
5.3.1. Probabilidad de error y MSE en función de la SNR	67
5.3.2. Probabilidad de error y MSE en función de M	69
5.3.3. Probabilidad de error y MSE en función del número de usuarios N	90
5.4. Conclusiones	94
6. Conclusiones y líneas futuras	96
6.1. Conclusiones	96
6.2. Líneas futuras de investigación	98
A. Abreviaturas	100
Bibliografía	102

Índice de figuras

1.1. Reparto del espectro entre los distintos usuarios de un sistema FD-MA dejando bandas de guarda entre usuarios consecutivos	3
1.2. Reparto del tiempo de transmisión en un sistema TDMA, dejando tiempos de guarda entre cada dos usos consecutivos del canal por parte de usuarios diferentes	4
1.3. Construcción de un código	7
1.4. Densidad espectral de potencia antes de la operación de ensanchado.	10
1.5. Densidad espectral de potencia después de la operación de ensanchado.	10
1.6. Densidad espectral de potencia de la señal recibida antes de la operación de desensanchado.	12
1.7. Densidad espectral de potencia de la señal recibida después de la operación de desensanchado.	13
1.8. Efecto dispersivo del canal en monoportadora	13
1.9. Ejemplo de respuesta en frecuencia de un canal dispersivo en un sistema con modulación monoportadora	14
1.10. Efecto dispersivo de un canal en un sistema con modulación multiportadora	15
1.11. Respuesta en frecuencia del canal en multiportadora	15
1.12. Primer tipo de CDMA Multiportadora para el caso de un usuario i transmitiendo un símbolo $b_i(n)$	16

1.13. Segundo tipo de CDMA Multiportadora	17
2.1. Diagrama de bloques de un sistema CDMA	20
3.1. Receptor lineal para el usuario i	27
5.1. Probabilidad de error y MSE promediados sobre 20 canales para $M = 1$	70
5.2. Probabilidad de error y MSE promediados sobre 20 canales para $M = 6$	71
5.3. Probabilidad de error y MSE promediados sobre 20 canales para $M = 11$	73
5.4. Probabilidad de error y MSE promediados sobre 20 canales para $M = 16$	74
5.5. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 0$	78
5.6. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 3$	80
5.7. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 6$	82
5.8. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 9$	84
5.9. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 12$	86
5.10. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 15$	88
5.11. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 18$	90
5.12. Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 12$	92

5.13. Probabilidad de error y MSE promediados sobre 20 canales para
 $SNR = 12$ y $M = 16$ 94

Índice de cuadros

3.1. Pseudocódigo del algoritmo RLS.	37
3.2. Pseudocódigo del algoritmo RLS DD.	39
4.1. Pseudocódigo del algoritmo EM.	54
4.2. Pseudocódigo del algoritmo SAGE.	57
5.1. Desviación típica de la probabilidad de error para $M = 1$	70
5.2. Desviación típica del MSE para $M = 1$	71
5.3. Desviación típica de la probabilidad de error para $M = 6$	72
5.4. Desviación típica del MSE para $M = 6$	72
5.5. Desviación típica de la probabilidad de error para $M = 11$	73
5.6. Desviación típica del MSE para $M = 11$	74
5.7. Desviación típica de la probabilidad de error para $M = 16$	75
5.8. Desviación típica del MSE para $M = 16$	75
5.9. Desviación típica de la probabilidad de error para $SNR = 0$	78
5.10. Desviación típica del MSE para $SNR = 0$	79
5.11. Desviación típica de la probabilidad de error para $SNR = 3$	79
5.12. Desviación típica del MSE para $SNR = 3$	81
5.13. Desviación típica de la probabilidad de error para $SNR = 6$	81
5.14. Desviación típica del MSE para $SNR = 6$	83
5.15. Desviación típica de la probabilidad de error para $SNR = 9$	83
5.16. Desviación típica del MSE para $SNR = 9$	85

5.17. Desviación típica de la probabilidad de error para $SNR = 12$	85
5.18. Desviación típica del MSE para $SNR = 12$	87
5.19. Desviación típica de la probabilidad de error para $SNR = 15$	87
5.20. Desviación típica del MSE para $SNR = 15$	89
5.21. Desviación típica de la probabilidad de error para $SNR = 18$	91
5.22. Desviación típica del MSE para $SNR = 18$	91
5.23. Desviación típica de la probabilidad de error para $SNR = 12$	91
5.24. Desviación típica del MSE para $SNR = 12$	93
5.25. Desviación típica de la probabilidad de error para $SNR = 12$ y $M = 16$	93
5.26. Desviación típica del MSE para $SNR = 12$ y $M = 16$	93

Capítulo 1

Introducción

En todo sistema de comunicaciones, la señal recibida en un extremo del canal difiere en cierta medida de la señal enviada desde el otro extremo. Ello es debido a que durante la transmisión se mezclan con la señal transmitida diversas señales no deseadas que distorsionan la señal original, y que se conocen como ruido. Además el canal de comunicaciones no tiene por qué tener una respuesta en frecuencia ideal, por lo que toda señal que lo atraviese sufrirá modificaciones, aún en ausencia de ruido aditivo. Para intentar recuperar en la medida de lo posible la señal original se emplean las técnicas de supresión de interferencias e igualación de canal.

En un medio de transmisión compartido por varios usuarios, además de las interferencias arriba comentadas, se producen Interferencias de Acceso Múltiple (MAI en inglés: *Multiple Access Interference*). Son debidas a que la señal de un determinado usuario de interés se recibe sumada con las señales de los restantes usuarios del sistema y para recuperarla es necesario llevar a cabo algún tipo de procesamiento.

En el presente proyecto se propondrá y estudiará un nuevo método de cancelación de interferencias en sistemas multiusuario capaz de mitigar el efecto del ruido y, sobre todo, reducir la MAI e igualar el canal.

1.1. Métodos de acceso múltiple

Existen dos formas básicas de compartir un medio de transmisión: en frecuencia y en tiempo.

En un sistema de acceso múltiple por división de frecuencia (FDMA o *Frequency Division Multiple Access*), se le asigna a cada usuario del sistema una banda de frecuencias sobre la que podrá transmitir sin que los espectros de las señales de los diferentes usuarios se solapen [14]. Para recuperar la señal deseada en el receptor bastará con utilizar un filtro paso banda adecuado. La figura (1.1) ilustra esto. En ella se puede ver como cada usuario dispone de una porción del ancho de banda del canal, que constituye un canal exclusivo para él. Además, entre cada dos usuarios será preciso dejar una estrecha banda de frecuencias desocupada, para asegurar la ortogonalidad entre los canales de los diferentes usuarios aún cuando se produzcan distorsiones de la señal del usuario en el dominio de la frecuencia debidas a un canal y un receptor no ideales.

En un sistema de acceso múltiple por división de tiempo (TDMA o *Time Division Multiple Access*) [14], el tiempo de transmisión se divide en ranuras que se asignarán circularmente a todos los usuarios del sistema, es decir, cuando a un usuario le llega su turno, dispone en exclusiva de todo el ancho de banda del canal durante el tiempo que dure la ranura y, una vez agotado este tiempo, no podrá volver a utilizar el canal hasta que los restantes usuarios hayan transmitido (cada uno dentro de su ranura correspondiente). Para recuperar la señal de un usuario concreto tan solo se precisa obtener la señal que se encuentra en el canal en unos momentos determinados: los correspondientes a las ranuras que le han sido asignadas a dicho usuario. En la figura (1.2) se puede observar como en un sistema TDMA el tiempo se divide en ranuras, cada una de las cuales está etiquetada con un número que se corresponde con el del usuario que tiene la posibilidad de transmitir durante ese intervalo temporal. Además, del mismo modo que en FDMA existían bandas de guarda, en TDMA se dejan tiempos de guarda entre cada dos

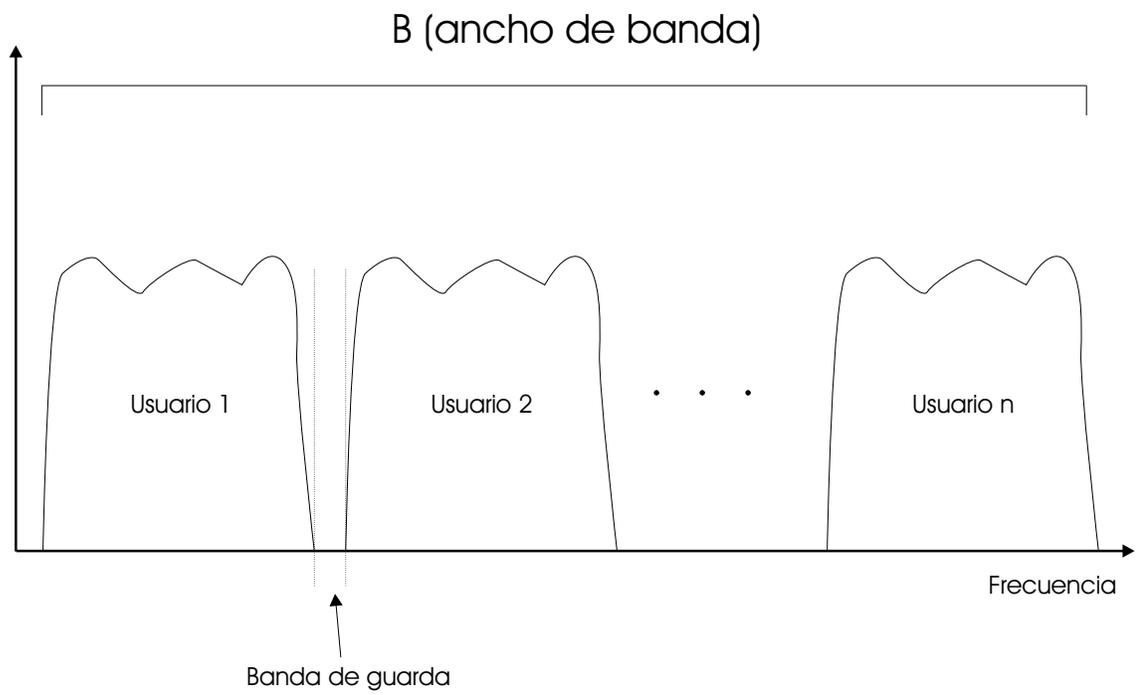


Figura 1.1: Reparto del espectro entre los distintos usuarios de un sistema FDMA dejando bandas de guarda entre usuarios consecutivos

transmisiones consecutivas por parte de usuarios diferentes.

Una limitación de TDMA es que requiere una estricta sincronización entre los diferentes usuarios del sistema. No ocurre lo mismo en FDMA, donde cada usuario puede transmitir en cualquier momento sin interferir con los demás. Además, en TDMA, cuando un usuario no tiene datos que transmitir el canal permanecerá desocupado durante todas las ranuras de tiempo que tiene asignadas dicho usuario, lo cual conlleva un considerable desaprovechamiento del medio de transmisión. Se puede paliar este inconveniente empleando protocolos estadísticos o dinámicos, que dividen el tiempo de transmisión en un número de ranuras inferior al de usuarios potenciales del sistema y luego las asignan dinámicamente. Sin embargo, estos protocolos suponen una sobrecarga del sistema y parte del tiempo de transmisión que se gana con su empleo se pierde gestionando el control y la coordinación que requiere dicha asignación dinámica [14].

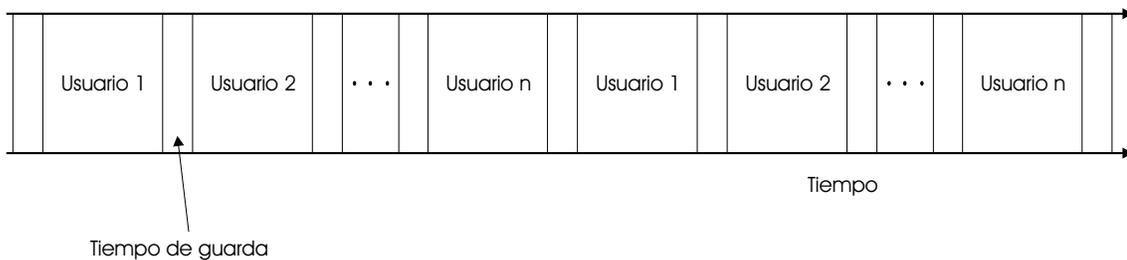


Figura 1.2: Reparto del tiempo de transmisión en un sistema TDMA, dejando tiempos de guarda entre cada dos usos consecutivos del canal por parte de usuarios diferentes

Una limitación práctica compartida por los esquemas FDMA y TDMA es que, debido a las características no ideales del canal y los filtros transmisor y receptor, se hace necesario dejar bandas de guarda en FDMA y tiempos de guarda en TDMA, de tal manera que podamos garantizar que no existe interferencia entre canales adyacentes. Esto supone una pérdida de eficiencia del sistema, pero es necesario para asegurar que las señales de los distintos usuarios sean ortogonales entre sí y, por tanto, los usuarios operen en canales no interferentes [16].

Otra característica común a estos dos métodos de acceso es que no se permite que las señales se solapen a la vez en tiempo y en frecuencia: o bien pueden solaparse en tiempo (FDMA) o bien en frecuencia (TDMA), pero no en ambos dominios a la vez. Se garantiza así la ortogonalidad de los canales de los distintos usuarios que se ha comentado antes. Nótese, sin embargo, que es posible separar las señales de dos o más usuarios que se solapan tanto en tiempo como en frecuencia. El acceso múltiple por división espacial (SDMA, en inglés: *Space Division Multiple Access*) es una posibilidad [17].

SDMA, efectivamente, permite transmitir a varios usuarios en la misma banda de frecuencias y al mismo tiempo. Utilizando un *array* de antenas y técnicas de procesado de señal es posible distinguir la señal de un usuario de la de otro (ambas propagándose a la vez y en la misma banda de frecuencias) en base a la dirección de llegada de las señales [17]. Sin embargo, en el campo de las comunicaciones móviles es difícil aplicar una técnica como SDMA, que se basa en que las señales de los diferentes emisores provienen de localizaciones específicas y, por tanto, han de incidir en cada una de las antenas receptoras con un ángulo conocido con cierta exactitud. En un sistema de comunicaciones móviles los terminales transmisores cambian con frecuencia de ubicación, pudiendo ocurrir que, ocasionalmente dos emisores se encuentren muy próximos y, por tanto, SDMA no permita separar las señales que transmiten. Una técnica que puede paliar las deficiencias de SDMA es la de acceso múltiple por división de código (CDMA o *Code Division Multiple Access*).

1.2. CDMA

CDMA debe su popularidad, en gran parte, a los sistemas de comunicaciones móviles de tercera generación, basados en el estándar europeo UMTS (*Universal Mobile Telecommunication System*), el cual a su vez, es una implementación de la recomendación técnica conocida como IMT-2000 (*International Mobile Telecommunication*

by the year 2000)¹ [2]. Son estos sistemas los que han impulsado la investigación y desarrollo de métodos de acceso que permitan un aprovechamiento más eficiente del canal de comunicaciones. La segunda generación de teléfonos celulares y dispositivos de comunicación móviles emplea mayoritariamente como método de acceso TDMA, lo cual limita mucho sus capacidades (el estándar dominante en Europa, GSM -*Global System for Mobile Communications*-, transmite a tan solo 9,6 kb/s [2], frente a los 2 Mb/s que prometen UMTS e IMT-2000). En un futuro a corto plazo, se prevé que la mayor parte del tráfico sea generado por aplicaciones multimedia, lo cual nos hace pensar en la necesidad de un sistema de comunicaciones de banda ancha. IMT-2000 y por tanto UMTS, están diseñados para soportar tráfico multimedia, y lo harán utilizando técnicas CDMA [6].

Al margen de este incremento en la velocidad de transmisión, CDMA tiene una serie de ventajas adicionales que lo convierten en una opción muy atractiva para cualquier medio de transmisión compartido. Como ya se comentó anteriormente, CDMA permite a los usuarios del sistema transmitir simultáneamente en el tiempo y empleando el mismo espectro radioeléctrico. Ello se consigue asignando a cada usuario un *código* o *firma*. Los códigos son formas de onda que cada usuario va a modular con los símbolos a transmitir. Un código de duración T_s (el período de símbolo) se compone, a su vez, de una secuencia de L *chips* binarios que modulan a un pulso $p(t)$ de duración T_c (ver figura 1.3). Se verifica pues la relación $T_s = LT_c$. El número de chips de que consta un código L , recibe el nombre de *longitud de código* o *ganancia de procesamiento*.

Por último, mencionar en este apartado que CDMA aporta una mayor flexibilidad en el diseño y la operación de la red de comunicaciones [6]. Cuando un usuario móvil cambia de celda mientras está transmitiendo (*handoff*) [12], es preciso llevar a cabo una serie de complejas operaciones para que no se interrumpa la comunicación. CDMA permite simplificar estas operaciones y realizar de ese modo un *soft*

¹En EE.UU. y Japón, la denominación IMT-2000 se emplea para referirse genéricamente a este tipo de sistemas.

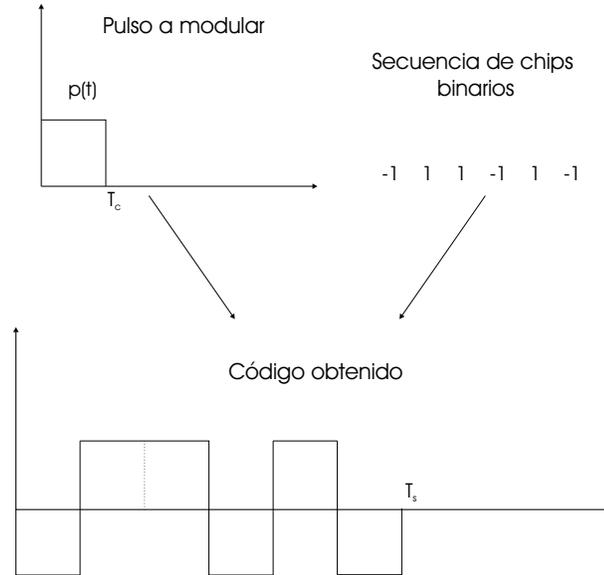


Figura 1.3: Construcción de un código

handoff. Todo ello da lugar a una mayor robustez del sistema.

1.2.1. Fundamentos de CDMA

La modulación de la señal del usuario mediante el código correspondiente provoca en dicha señal un ensanchado de espectro, pero a la vez consigue que cada usuario opere en su propio canal que, idealmente, es ortogonal con los de los demás usuarios. Esto permite que el receptor pueda extraer a un usuario dado sin interferencias. Si suponemos códigos ortogonales, podemos ilustrar fácilmente el funcionamiento de CDMA. Antes de llevar a cabo la transmisión, cada usuario ha de multiplicar el símbolo que va a transmitir por el código que le ha sido asignado. Utilizando una notación matricial, se puede obtener [11]:

$$\mathbf{x}(n) = \mathbf{C}\mathbf{b}(n) + \mathbf{g}(n) = \sum_{l=1}^N \mathbf{c}_l b_l(n) + \mathbf{g}(n) = \mathbf{c}_i b_i(n) + \overbrace{\sum_{l=1, l \neq i}^N \mathbf{c}_l b_l(n)}^{MAI} + \mathbf{g}(n) \quad (1.1)$$

donde $\mathbf{x}(n)$ es un vector de dimensiones $L \times 1$, que contiene los símbolos n -ésimos de todos los usuarios, \mathbf{C} es la matriz de códigos ($L \times N$), en la cual la columna \mathbf{c}_i

es el código del i -ésimo usuario, $\mathbf{b}(n)$ es un vector $N \times 1$ que contiene el n -ésimo símbolo que transmite cada usuario, y $\mathbf{g}(n)$ es un vector columna de dimensiones $L \times 1$, que representa el ruido que aparece durante la transmisión del n -ésimo vector de símbolos. Esta forma básica de CDMA recibe el nombre de CDMA por secuencia directa o DS-CDMA (*Direct Sequence-CDMA*).

Para extraer de $\mathbf{x}(n)$ el n -ésimo símbolo del i -ésimo usuario, solo es preciso multiplicar $\mathbf{x}(n)$ por la transposición hermítica del código del usuario correspondiente (en este caso el i). Ello es debido a que el vector \mathbf{c}_i , que representa el código del usuario i , es ortogonal a todos los vectores \mathbf{c}_j , $j \neq i$ y, por tanto, el segundo sumatorio que aparece en la ecuación 1.1 y que representa la MAI se anula. De esa manera:

$$\begin{aligned} \mathbf{y}(n) &= \mathbf{c}_i^H \mathbf{x}(n) = \mathbf{c}_i^H \mathbf{c}_i b_i(n) + \sum_{l=1, l \neq i}^N \mathbf{c}_i^H \mathbf{c}_l b_l(n) + \mathbf{c}_i^H \mathbf{g}(n) \\ &= \mathbf{c}_i^H \mathbf{c}_i b_i(n) + \mathbf{c}_i^H \mathbf{g}(n) = k b_i(n) + \mathbf{c}_i^H \mathbf{g}(n) \end{aligned} \quad (1.2)$$

Así pues, en el resultado, $\mathbf{y}(n)$, solo aparecen dos términos: un término de ruido $\mathbf{c}_i^H \mathbf{g}(n)$, y otro término constituido por el símbolo que queríamos extraer multiplicado por una constante, que en la ecuación 1.2 se denomina arbitrariamente k .

Sin embargo, en la práctica no es posible emplear códigos ortogonales porque se permite a los usuarios transmitir asincrónicamente y no es posible diseñar códigos que sean ortogonales para cualquier desplazamiento temporal arbitrario [13]. Además, los códigos se distorsionan al atravesar el canal y no es posible predecir con exactitud dicha distorsión. Pero la condición de ortogonalidad es innecesaria (la podemos pasar por alto siempre que la correlación cruzada entre los diferentes códigos no sea muy alta) e incluso indeseable, pues puede ocasionar problemas de sincronización en el receptor. Si los códigos son ortogonales, la función de autocorrelación de la señal que se obtiene en el receptor al multiplicar por el código de usuario (ecuación 1.2) tiene múltiples máximos locales que dificultan la tarea de

identificar el comienzo del símbolo. Sin el requisito de ortogonalidad de los códigos, el número de usuarios del sistema no está limitado por el número de códigos ortogonales que seamos capaces de diseñar [16].

Otro punto a favor de CDMA es que la calidad de servicio depende del número de usuarios que en un momento dado estén haciendo uso del sistema. Es decir, si hay pocos usuarios compartiendo el canal, la MAI será menor y la calidad de la señal demodulada mayor, y a medida que aumenta el número de usuarios, se va degradando la calidad, pero de una manera progresiva. Ello se debe a la imposibilidad de emplear códigos ortogonales, a consecuencia de lo cual, los usuarios se interfieren en cierta medida los unos con los otros.

1.2.2. Privacidad de la señal CDMA

Ya se ha comentado que el ensanchamiento de espectro provoca un incremento en el ancho de banda de la señal a transmitir, que permite el acceso múltiple simultáneamente y con solapamiento en frecuencia. Otro efecto beneficioso de ese proceso de ensanchamiento espectral es que la potencia de la señal se distribuye en un ancho de banda mucho mayor, con lo cual se reduce la densidad de potencia y la señal se ve protegida contra receptores no autorizados al quedar enmascarada por el ruido [11]. Antes de llevar a cabo el proceso de ensanchado de espectro, un receptor no autorizado podría inspeccionar una determinada franja de frecuencias del canal de comunicaciones y observar que en una zona concreta de esa franja la densidad de potencia es mucho mayor (ver figura 1.4). Ello le permitiría deducir que está teniendo lugar una transmisión, así como determinar qué banda de frecuencias está siendo usada para dicha transmisión.

Después del ensanchamiento de espectro, la densidad de potencia de la señal a transmitir presenta un aspecto similar al que se observa en la figura 1.5. Ahora la densidad espectral de potencia de la señal de interés se distribuye en un ancho de banda mucho mayor, de manera que no se puede distinguir fácilmente de la

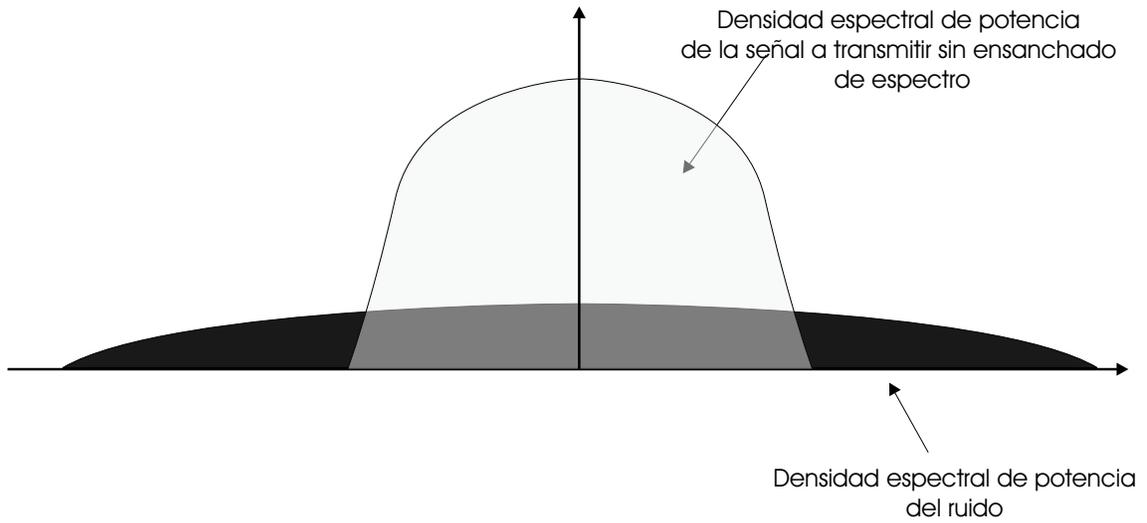


Figura 1.4: Densidad espectral de potencia **antes** de la operación de ensanchado.

del ruido. Además, para percibir la señal portadora de la información, el receptor tendría que rastrearla en un rango de frecuencias mucho mayor.

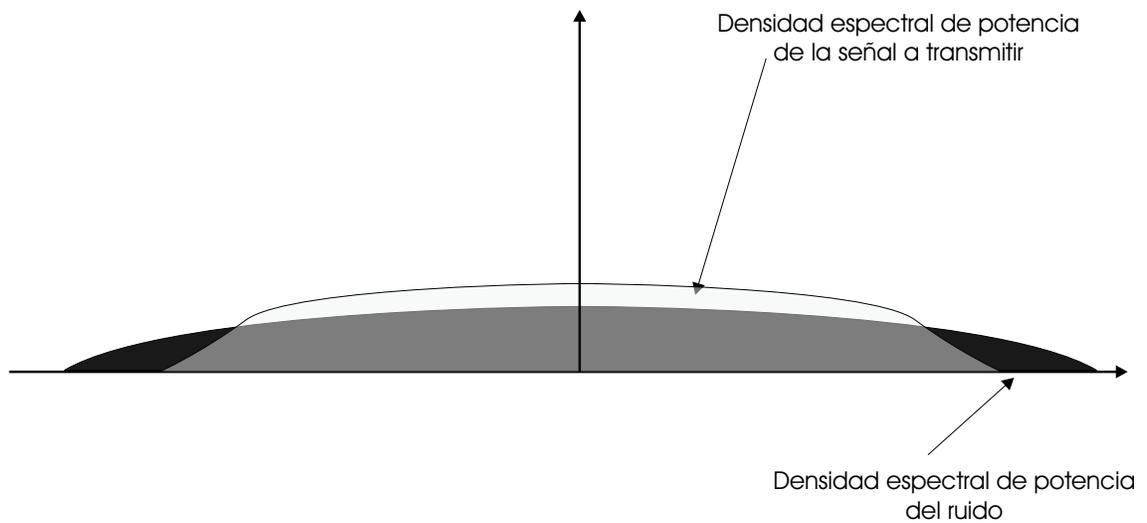


Figura 1.5: Densidad espectral de potencia **después** de la operación de ensanchado.

Así pues, tenemos un sistema de comunicación mucho más seguro y apto para ser utilizado en entornos donde la privacidad es un factor importante.

1.2.3. Interferencias de banda estrecha

En relación con el efecto producido por el ensanchado de espectro, CDMA tiene otra ventaja: favorece la robustez del sistema frente a las interferencias de banda estrecha (ruido que afecta a una estrecha banda de frecuencias) [11]. Ello es debido a que la operación de ensanchado (que tiene lugar en el emisor) y su operación inversa (que ocurre en el receptor) son en realidad la misma transformación. Ello se aprecia claramente en las ecuaciones (1.1) y (1.2). La primera describe la operación llevada a cabo en el emisor. Para transmitir un símbolo $b_i(n)$, un usuario i únicamente debe multiplicarlo por el código \mathbf{c}_i que le ha sido asignado. En el receptor, para obtener una estimación blanda del símbolo transmitido por el usuario i , hay que multiplicar de nuevo la señal recibida $\mathbf{x}(n)$ por el código \mathbf{c}_i , tal como se aprecia en la ecuación (1.2). Por ello, cuando en recepción se comprime de nuevo el espectro de la señal recibida (al aplicarle nuevamente la operación de multiplicación por el código), se produce al mismo tiempo el ensanchado del espectro de cualquier interferencia de banda estrecha. Es decir, si durante la transmisión de la señal aparece una interferencia de banda estrecha (ver figura 1.6), esa interferencia será ensanchada en el receptor, de tal manera que su densidad de potencia se reducirá considerablemente (ver figura 1.7). De esta manera, ahora es posible eliminar gran parte de esa interferencia no deseada aplicando un filtro paso bajo que capture solo las frecuencias que nos interesan y descarte aquellas que solo están ocupadas por la interferencia.

1.3. Modulaciones multiportadora

En un sistema con modulación monoportadora se procura aprovechar todo el ancho de banda del canal de comunicaciones haciendo que los símbolos que se transmiten tengan una duración tan pequeña como sea posible. Si tenemos un canal dispersivo, un símbolo de duración T , una vez haya atravesado el canal de

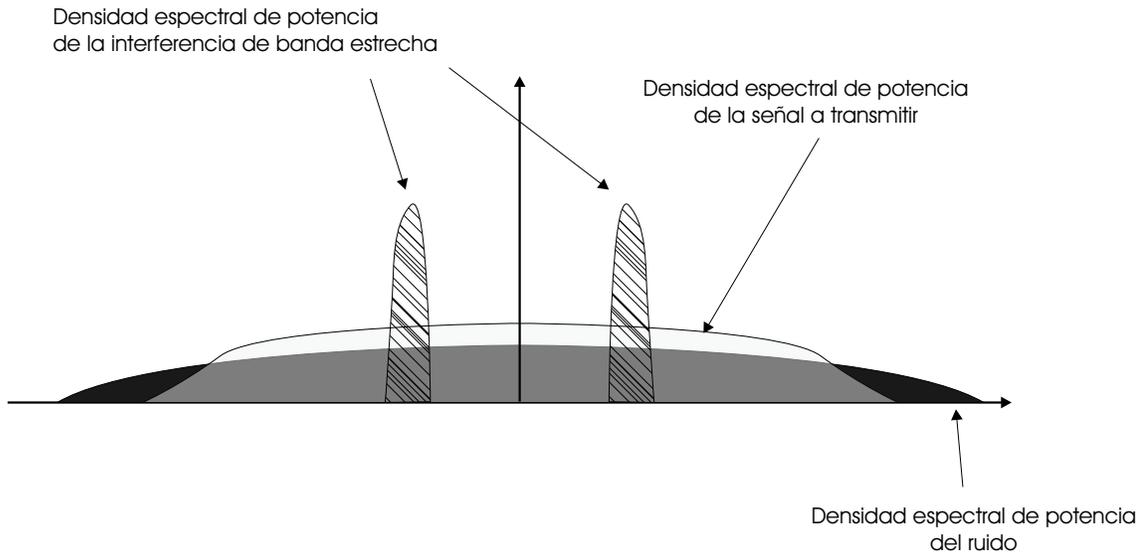


Figura 1.6: Densidad espectral de potencia de la señal recibida **antes** de la operación de desensanchado.

comunicaciones, tendrá una duración $T + \tau$, siendo τ la duración adicional del símbolo ocasionado por el efecto dispersivo del canal (ver figura 1.8). Este incremento en la duración del símbolo es ocasionado por la convolución con la respuesta al impulso del canal que, no siendo ideal, tiene una duración no despreciable, que se añade a la del símbolo.

Si queremos evitar que este incremento (τ) en la duración del símbolo afecte al símbolo siguiente será preciso dejar un tiempo de guarda entre símbolos. Si no lo hiciéramos se produciría un tipo de interferencia conocido como ISI (*Inter-Symbol Interference* o interferencia intersimbólica), que consiste en el solapamiento de símbolos que han sido transmitidos de manera consecutiva. Ahora bien, teniendo en cuenta que en monoportadora se trabaja con períodos de símbolo muy pequeños (máximo aprovechamiento del ancho de banda del canal), estos tiempos de guarda τ que hemos de dejar entre símbolo y símbolo suponen una notable pérdida de rendimiento ya que durante el tiempo que ocupan *todos* los intervalos de guarda no estamos transmitiendo nada, es decir, el cociente *duración del símbolo a transmitir-duración real* es muy bajo (en este caso $\frac{T}{T+\tau}$).

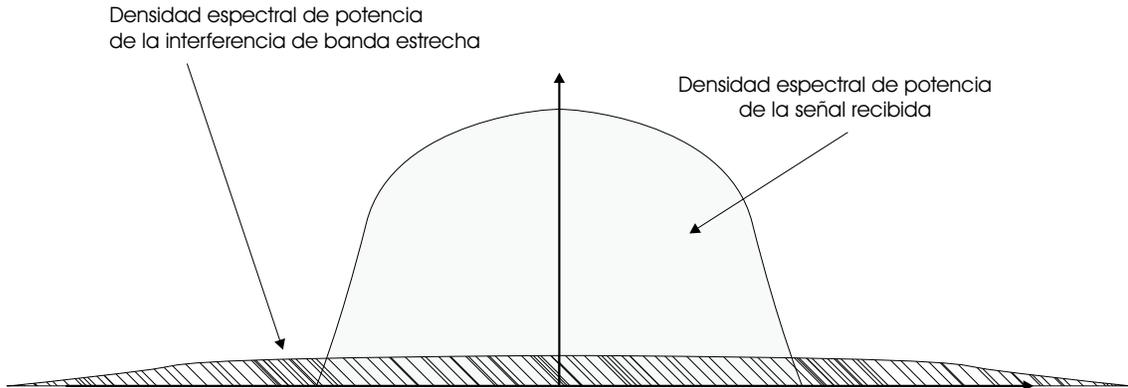


Figura 1.7: Densidad espectral de potencia de la señal recibida **después** de la operación de desensanchado.

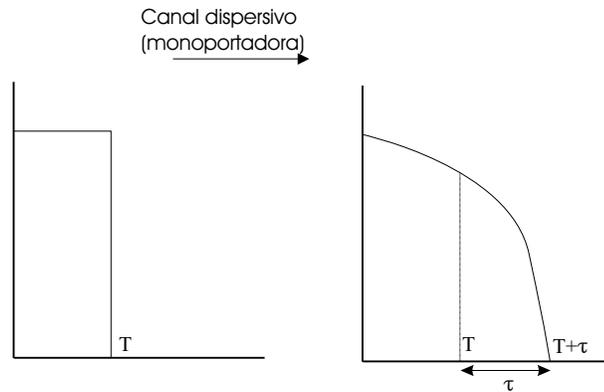


Figura 1.8: Efecto dispersivo del canal en monoportadora

El hecho de tener un canal dispersivo en el dominio del tiempo es algo que también se puede apreciar en el dominio de la frecuencia. Un canal dispersivo en tiempo equivale a un canal selectivo en frecuencia, es decir, un canal de comunicaciones que provocará una mayor o menor atenuación de la señal, dependiendo de las componentes en frecuencia de ésta. En la figura 1.9 se puede ver un ejemplo de la respuesta en frecuencia de un canal dispersivo.

En un sistema con modulación multiportadora (o MC: *MultiCarrier*), el ancho de banda total del canal se divide en L subcanales. Es decir, ya no tenemos un único canal de comunicaciones con un elevado ancho de banda B , sino que tenemos

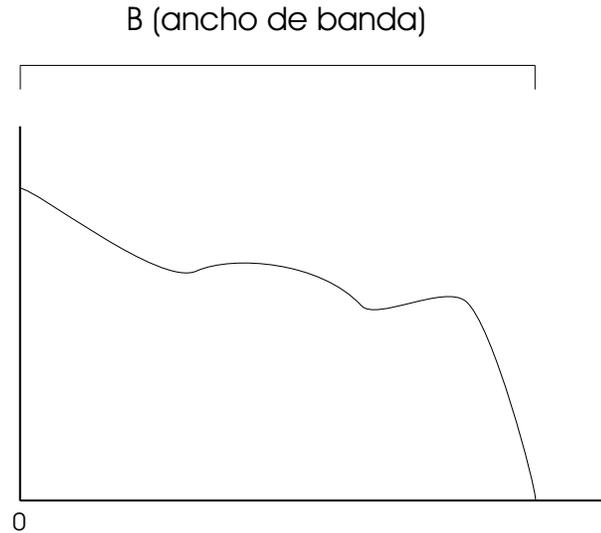


Figura 1.9: Ejemplo de respuesta en frecuencia de un canal dispersivo en un sistema con modulación monoportadora

L subcanales con ancho de banda $\frac{B}{L}$, por los que transmitiremos simultáneamente L señales portadoras de información. Que el ancho de banda de cada uno de los subcanales sea L veces menor que en monoportadora, conlleva que el período de símbolo sea L veces mayor. Debido a que el período de símbolo es mucho mayor, los efectos nocivos de un canal selectivo en frecuencia, es decir, una dispersión temporal de duración τ , son despreciables en comparación con el caso de monoportadora. El ratio *duración del símbolo a transmitir-duración real* es mucho más elevado. Ahora nuestro período de símbolo es LT , y τ (que viene dado por el canal de comunicaciones) se mantiene constante. Tenemos, pues, que dicho ratio es $\frac{LT}{LT+\tau}$, el cual tenderá a la unidad a medida que incrementamos L . Se puede observar en la figura 1.10.

Un punto de vista alternativo es que, en multiportadora, a medida que incrementamos L , obtenemos subcanales mucho más estrechos en el dominio de la frecuencia, que individualmente podemos considerar planos y que tienen, por tanto, un efecto dispersivo nulo o muy pequeño (ver figura 1.11).

Un tipo concreto de modulación multiportadora es la multiplexación por divi-

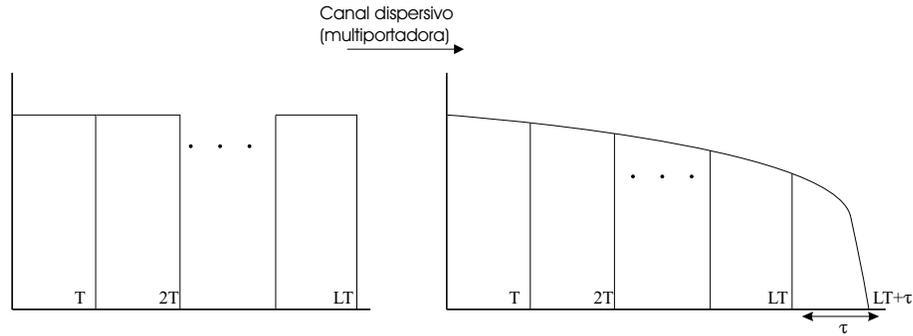


Figura 1.10: Efecto dispersivo de un canal en un sistema con modulación multipor-
tadora

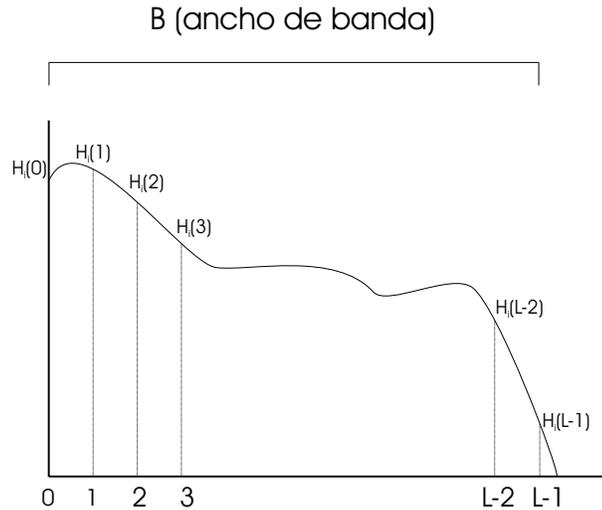


Figura 1.11: Respuesta en frecuencia del canal en multiportadora

sión ortogonal de frecuencia (OFDM: *Orthogonal Frequency Division Multiplexing*). OFDM permite que los diferentes subcanales se solapen, consiguiendo de este modo un mejor aprovechamiento del ancho de banda. Los subcanales todavía se pueden separar en el receptor porque se eligen las frecuencias de las subportadoras de tal manera que sean ortogonales [12].

1.4. Sistemas CDMA multiportadora

CDMA y MC se pueden usar conjuntamente de diversas formas. Se clasifican en dos grupos fundamentales [4].

En el primero de ellos, cada uno de los chips que componen un símbolo una vez ensanchado éste, será transmitido por un subcanal diferente. Así pues, realmente no existe un ensanchamiento del espectro, sino que los códigos únicamente expanden cada símbolo en L chips, cada uno de los cuales modulará una subportadora diferente. La figura (1.12) ilustra este primer tipo de CDMA multiportadora para un usuario i transmitiendo un símbolo $b_i(n)$.

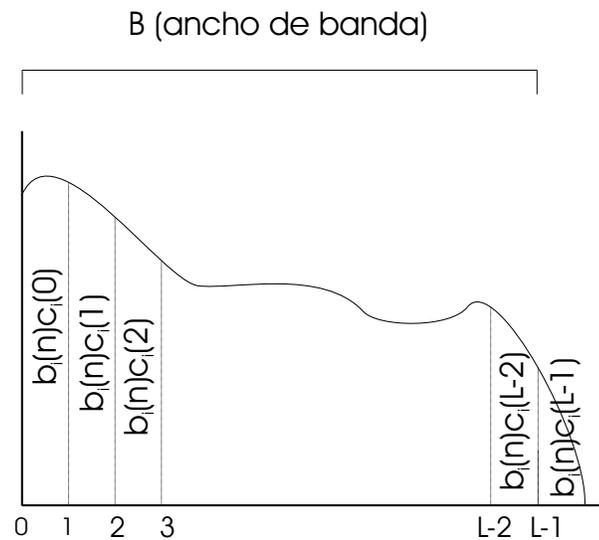


Figura 1.12: Primer tipo de CDMA Multiportadora para el caso de un usuario i transmitiendo un símbolo $b_i(n)$

El otro grupo se caracteriza primeramente por dividir la secuencia de símbolos que va a transmitir el usuario en varias secuencias mediante un convertidor de serie a paralelo. Después se lleva a cabo la operación de ensanchado sobre cada una de esas secuencias y se transmite cada una por un subcanal diferente. Básicamente consiste en aplicar DS-CDMA en cada subportadora. En este caso cada símbolo a transmitir se ve multiplicado por un código y el resultado de este producto (L

chips) se transmite íntegramente por un subcanal. Por tanto, en este caso sí tiene lugar un ensanchamiento de espectro dentro de cada subcanal. En la figura 1.13, en la que el ancho de banda disponible se ha dividido en M subcanales, el usuario i transmite a la vez M símbolos (desde $b_i(n)$ hasta $b_i(n + M - 1)$), uno por cada subcanal, multiplicando cada símbolo por el código que le ha sido asignado.

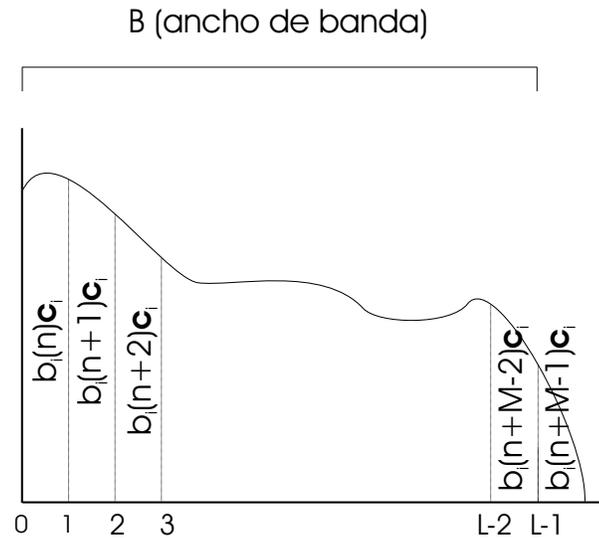


Figura 1.13: Segundo tipo de CDMA Multiportadora

Entre las técnicas que se engloban dentro del primer grupo está la combinación de OFDM y CDMA y que de aquí en adelante denotaremos como MC-CDMA. Su funcionamiento es exactamente el que está descrito arriba: en el transmisor se multiplica cada símbolo por el código y cada uno de los chips resultantes modula a una subportadora diferente. Esta es la técnica combinación de multiportadora y CDMA más utilizada y la que estudiaremos en detalle en el presente proyecto.

En el segundo grupo se clasifican las técnicas: Multicarrier DS-CDMA (*MultiCarrier Direct Sequence-CDMA*) y MT-CDMA (*MultiTone-CDMA*).

Multicarrier DS-CDMA consiste en dividir la secuencia de símbolos a transmitir en varias secuencias que se transmitirán en paralelo, cada una por un subcanal diferente. Se realizará el ensanchado de espectro sobre cada secuencia y dentro de su canal correspondiente, aplicando el código de usuario. Cuando solamente existe

un canal (sistema monoportadora: no hay división en subcanales), y por tanto la secuencia inicial de símbolos no se subdivide en varias secuencias, hablamos de DS-CDMA (*Direct Sequence-CDMA*).

MT-CDMA es similar a Multicarrier DS-CDMA, con la salvedad de que los espectros de cada una de las secuencias se solapan una vez llevada a cabo la operación de ensanchado. Tiene como ventaja que, en relación al número de subportadoras, permite utilizar códigos más largos que la técnica monoportadora DS-CDMA.

Capítulo 2

Modelo de señal

La técnica de acceso al medio que servirá de base para el presente trabajo es MC-CDMA (*MultiCarrier-Code Division Multiple Access*). En este capítulo se presenta el modelo equivalente discreto en banda base de un sistema de comunicaciones MC-CDMA con N usuarios. Para ello, nos serviremos de un diagrama de bloques, que nos permitirá comprender gráficamente las distintas fases que tienen lugar en el proceso de comunicación. Es el que se muestra en la figura 2.1.

2.1. El transmisor

El primer bloque que se aprecia claramente en el diagrama es el del transmisor. En este modelo, el i -ésimo usuario ($i = 1 \dots N$) transmite una secuencia de símbolos $b_i(n)$, $n = 1, \dots, K$, donde K es el tamaño de la trama. Cada usuario i dispone de un código $c_i(k)$, $k = 0, 1, \dots, L - 1$, donde L es la *longitud del código* o *ganancia de procesamiento* ($L = \frac{T_s}{T_c}$), que multiplica a cada uno de los símbolos que transmite y que permitirá en el receptor distinguir a ese usuario del resto (*Code Division*)¹. Así pues, para transmitir un símbolo $b_i(n)$, se construye una secuencia discreta

¹Puesto que debido a la distorsión del canal no es posible garantizar la ortogonalidad de los códigos o, incluso, que las correlaciones cruzadas sean bajas, la única hipótesis que se hace es que los códigos sean linealmente independientes tras pasar por el canal.

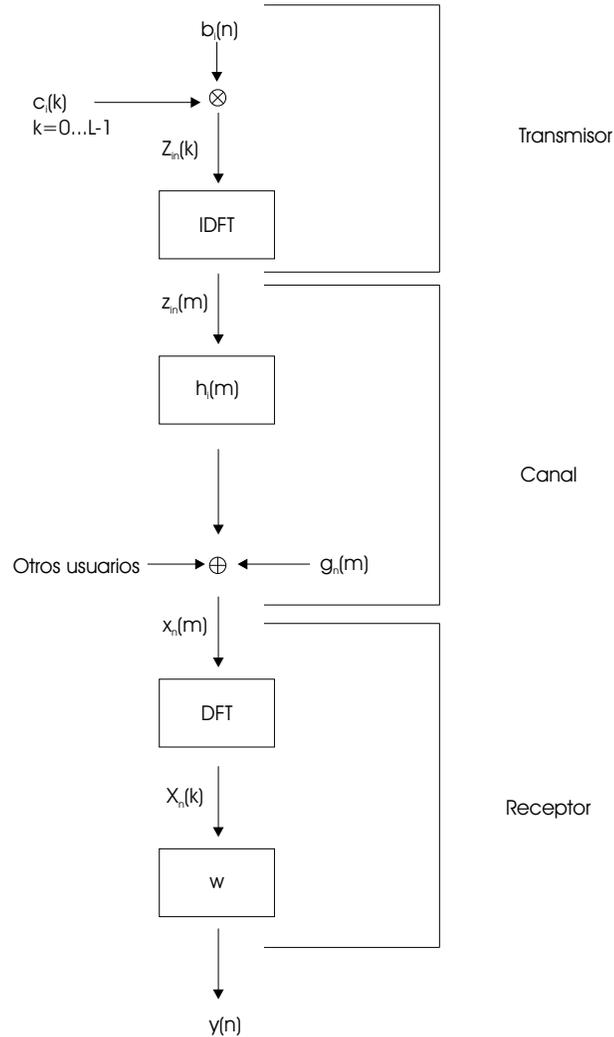


Figura 2.1: Diagrama de bloques de un sistema CDMA

$Z_{in}(k)$ multiplicando este símbolo por el código del usuario, es decir, $Z_{in}(k) = b_i(n)c_i(k)$, donde $c_i(k)$, $k = 0, \dots, L - 1$ es una secuencia binaria pseudoaleatoria, que constituye el código de usuario, y $b_i(n)$ es el el n -ésimo símbolo a transmitir por el i -ésimo usuario. Sobre esta señal discreta $Z_{in}(k)$, $k = 0, \dots, L - 1$, se calcula la transformada de Fourier discreta inversa (IDFT, del inglés *Inverse Discrete Fourier Transform*). El principio básico de MC-CDMA es que cada uno de los subcanales en que se divide el ancho de banda es ocupado por uno de los chips del símbolo a transmitir. En este caso $Z_{in}(k)$ es el k -ésimo chip correspondiente

al n -ésimo símbolo del i -ésimo usuario, es decir, es la componente de señal del usuario i que se debe transmitir a través del subcanal k . Por tanto $Z_{in}(k)$ se puede interpretar como una señal en el dominio de la frecuencia y calculando su IDFT obtenemos su transformada en el dominio del tiempo,

$$z_{in}(m) = IDFT\{Z_{in}(k)\} = \sum_{k=0}^{L-1} Z_{in}(k) e^{j\frac{2\pi k}{L}m}. \quad (2.1)$$

La señal a transmitir se construye finalmente añadiendo un prefijo cíclico a $z_{in}(m)$ para obtener $\tilde{z}_{in}(m)$ de tal manera que

$$\tilde{z}_{in}(-P-1)\dots\tilde{z}_{in}(L-1) = \overbrace{z_{in}(L-P-2)\dots z_{in}(L-2)}^{\text{prefijo}} z_{in}(L-1) \overbrace{z_{in}(0)\dots z_{in}(L-1)}^{z_{in}} \quad (2.2)$$

donde P es la longitud de la secuencia $h_i(m)$ (que, como se verá a continuación, representa el canal del usuario i) y L la de $z_{in}(m)$.

2.2. El canal de comunicaciones

La secuencia $z_{in}(m)$ es la señal portadora de información en tiempo discreto correspondiente al usuario i -ésimo. El canal del usuario i tiene una respuesta al impulso $h_i(m)$, de tal manera que el proceso de transmisión se reduce a una convolución entre el símbolo que se va a transmitir y $h_i(m)$. Sin embargo el resultado de esta convolución no es la señal que aparecerá a la entrada del receptor. A la entrada del receptor, la señal del usuario i se suma con las de los restantes usuarios del sistema y con el ruido térmico que se modela como gaussiano, blanco y aditivo (AWGN o **A**dditive **W**hite **G**aussian **N**oise). En la figura 2.1, $g_n(m)$ es el ruido que aparece en el canal durante la transmisión del símbolo n .

Aplicando la propiedad de convolución de la DFT [10], es sencillo demostrar que la secuencia recibida es de la forma:

$$x_n(m) = \sum_{i=1}^N \tilde{z}_{in}(m) * h_i(m) + g_n(m) = \sum_{i=1}^N IDFT\{Z_{in}(k)H_i(k)\} + g_n(m) \quad (2.3)$$

donde el operador $*$ indica convolución lineal discreta, $g_n(m)$ es el ruido para el n -ésimo símbolo y la secuencia $H_i(k)$ es la transformada de Fourier discreta (DFT) de $h_i(m)$.

El prefijo cíclico en $\tilde{z}_{in}(m)$ es necesario porque la propiedad de convolución de la DFT para el caso de secuencias de longitud finita, se refiere a la convolución circular. Añadiendo el prefijo de la ecuación (2.2) se consigue que

$$\tilde{z}_{in}(m) \stackrel{\text{conv.}}{\downarrow} * h_i(m) = z_{in}(m) \stackrel{\text{conv. circular}}{\downarrow} \otimes h_i(m) \quad (2.4)$$

y se cumple la igualdad de la ecuación (2.3).

2.3. El receptor

En el receptor se calcula la DFT de la secuencia recibida:

$$X_n(k) = DFT\{x_n(m)\} = \sum_{i=1}^N Z_{in}(k)H_i(k) + G_n(k) \quad (2.5)$$

donde $G_n(k)$ es la DFT del ruido blanco, gaussiano y aditivo, que todavía conserva estas propiedades una vez calculada su transformada de Fourier discreta [10].

Teniendo en cuenta que $Z_{in}(k) = b_i(n)c_i(k)$, podemos escribir el conjunto de ecuaciones que resultan de (2.5) para $k = 0, \dots, L - 1$ de forma matricial como:

$$\mathbf{x}(n) = \begin{bmatrix} X_n(0) \\ \vdots \\ X_n(L-1) \end{bmatrix} = \sum_{i=1}^N b_i(n)\mathbf{d}_i + \mathbf{g}(n) = \mathbf{D}\mathbf{b}(n) + \mathbf{g}(n) \quad (2.6)$$

donde $\mathbf{d}_i = [d_i(0), \dots, d_i(k), \dots, d_i(L-1)]^T$ es el vector con los códigos del usuario i , una vez han sido distorsionados por el canal y

$$d_i(k) = c_i(k)H_i(k). \quad (2.7)$$

La matriz \mathbf{D} , de dimensiones $L \times N$, es la matriz de códigos distorsionada, y cada una de sus columnas se refiere a un usuario diferente, $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_N]$. El vector de

símbolos $\mathbf{b}(n) = [b_1(n), \dots, b_N(n)]^T$ es un vector columna $N \times 1$ que contiene el n -ésimo símbolo transmitido por cada usuario. Por último $\mathbf{g}(n) = [G_n(0), \dots, G_n(L - 1)]^T$ es el vector de ruido, cuyas dimensiones son $L \times 1$.

Una vez que se tienen las observaciones, el objetivo es estimar el símbolo transmitido por el i -ésimo usuario. Para ello emplearemos un filtro lineal de respuesta al impulso finita $\mathbf{w} = [w_0, \dots, w_{L-1}]^T$, específico para dicho usuario. La operación de filtrado tiene el objetivo de reducir la MAI, igualar el canal y mitigar los efectos del ruido. La estimación del n -ésimo símbolo obtenida aplicando el filtro es:

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \quad (2.8)$$

En la introducción se comentó que una de las grandes ventajas de utilizar un sistema multiportadora es que se corrige el efecto dispersivo del canal. En efecto, en la ecuación (2.5) podemos observar que, dado un símbolo n , la aportación del usuario i a la observación correspondiente al k -ésimo chip es $Z_{in}(k)H_i(k)$, es decir, el canal $H_i(k)$ tiene un efecto multiplicativo y no convolutivo sobre los símbolos del usuario.

Teniendo en cuenta lo anterior y que $Z_{in}(k) = b_i(n)c_i(k)$, entonces la aportación del usuario i a la observación correspondiente al k -ésimo chip del n -ésimo símbolo puede reescribirse como $b_i(n)c_i(k)H_i(k)$, lo cual quiere decir, que el k -ésimo subcanal del usuario i , $H_i(k)$, transportará la información correspondiente al k -ésimo chip del n -ésimo símbolo, $b_i(n)c_i(k)$. En resumidas cuentas, todos los usuarios transmiten a la vez un símbolo, introduciendo la información correspondiente a cada uno de los chips del símbolo en cada uno de los subcanales en los que se divide el ancho de banda total del canal.

2.4. Conclusiones

Se ha descrito un modelo de señal en el que se consideran las tres componentes fundamentales de todo sistema de comunicaciones: el transmisor, el canal y el

receptor.

Cada usuario del sistema tiene asignado un código que va a multiplicar a cada símbolo que transmita. Un código no es más que una secuencia binaria (1 ó -1) pseudoaleatoria de tamaño L . Por tanto, al multiplicar un usuario cada uno de sus símbolos por el mencionado código resulta una secuencia de L chips, que se transmiten en paralelo utilizando una modulación multiportadora de L subcanales de acuerdo al esquema MC-CDMA. La forma más eficiente de generar esta señal es mediante el cómputo de una IDFT.

Cada símbolo transmitido por un usuario se convoluciona con su canal correspondiente (hay tantos canales como usuarios). La señal resultante le llegará al receptor sumada con las señales de los restantes usuarios (obtenidas de forma análoga) y con el ruido térmico modelado como AWGN.

En el receptor se lleva a cabo una DFT de la señal recibida, para invertir la operación realizada en el transmisor, y se emplea un filtro lineal adecuado para extraer los símbolos del usuario deseado.

Capítulo 3

Técnicas clásicas de cancelación de interferencias en CDMA

Como ya se ha visto en el capítulo dedicado a la descripción del modelo de señal, en un momento dado N usuarios del sistema transmiten simultáneamente y contribuyen a un vector de observaciones $\mathbf{x}(n)$ de tamaño L (la longitud del código) en el receptor. Este vector de observaciones contiene, por tanto, información relativa a los símbolos n -ésimos de todos los usuarios, y no solo el símbolo transmitido por el usuario de interés. Para extraer este último será preciso emplear algún mecanismo que cancele las interferencias debidas a los símbolos transmitidos por los usuarios no deseados. También será necesario igualar el canal (atenuar la distorsión que introduce debido a que no es ideal) y mitigar en la medida de lo posible el ruido térmico que se añade a la señal a la entrada del receptor.

Para lograr este objetivo se pueden emplear diferentes arquitecturas receptoras. Por su simplicidad, son especialmente populares los receptores formados por un filtro transversal de respuesta al impulso finita. En el presente trabajo se describirán algunos de los más comunes.

Además de las aproximaciones que se van a exponer en este capítulo, existe un receptor óptimo que detecta los símbolos transmitidos por el usuario con probabili-

dad de error mínima [16]. Sin embargo, su prohibitiva complejidad computacional lo hace impracticable. Por ello se opta habitualmente por receptores lineales, que logran un equilibrio satisfactorio entre el rendimiento y la complejidad de su implementación [16, 11].

3.1. Receptores lineales

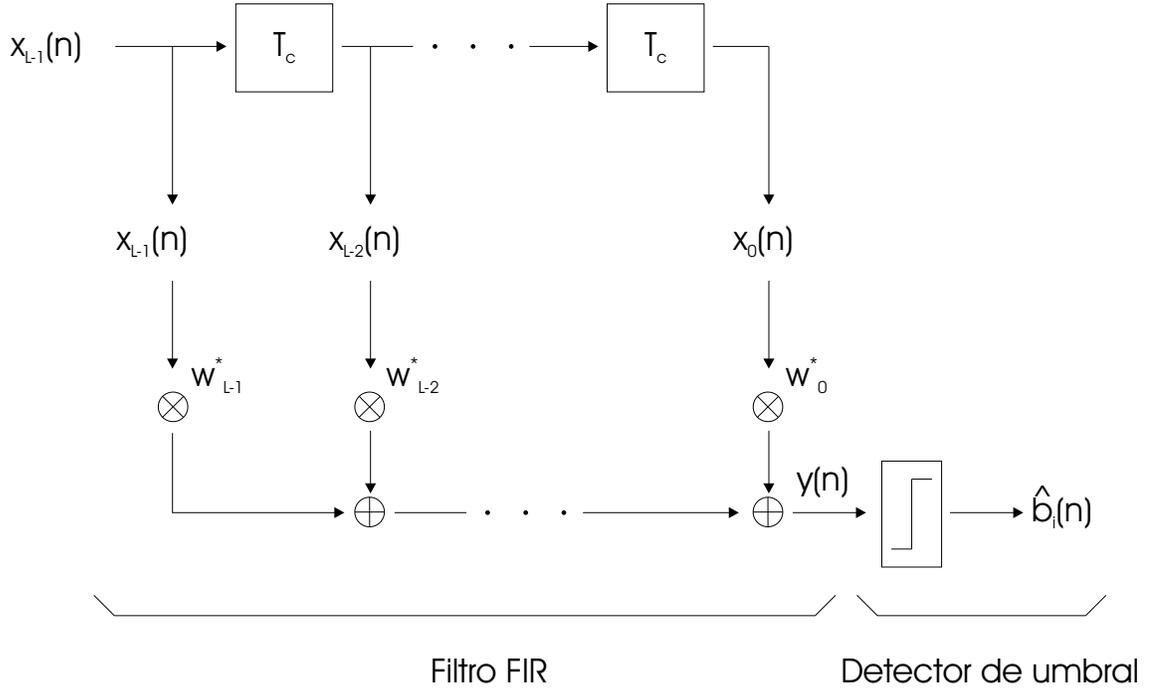
Un receptor lineal se puede implementar como un filtro transversal de respuesta al impulso finita (FIR o *Finite Impulse Response*) y un detector de umbral, tal como se muestra en la figura 3.1. El filtro está compuesto por una línea de retardos T_c , cuyas salidas son las observaciones que componen el vector $\mathbf{x}(n)$. Cada observación $X_n(k)$, $k = 0, \dots, L - 1$, se multiplica por el correspondiente coeficiente del filtro, w_k^* . La suma de todos esos productos proporciona una estimación *blanda*, $y(n)$, del símbolo transmitido por el usuario de interés. Esta estimación blanda es un número, posiblemente complejo, al que un detector de umbral asociará uno de los símbolos del alfabeto de la modulación, $\hat{b}_i(n)$, que constituye una estimación *dura*. El criterio para decidir que símbolo del alfabeto de la modulación ha de asignarse a una estimación blanda dada es el de la distancia mínima,

$$\hat{b}_i(n) = \arg \min_{b \in \mathcal{A}} |b - y(n)| \quad (3.1)$$

siendo

$$\mathcal{A} = \{b^{(1)}, \dots, b^{(B)}\} \quad (3.2)$$

el alfabeto de la modulación y B su longitud. Es decir, se elegirá aquel símbolo del alfabeto \mathcal{A} cuya distancia con la estimación blanda $y(n)$ sea mínima. Todos los receptores que se describen en esta sección tienen la estructura anteriormente reseñada. Únicamente se diferencian en la forma en que se seleccionan los coeficientes del filtro \mathbf{w} .


 Figura 3.1: Receptor lineal para el usuario i .

3.1.1. Detector decorrelador

La detección por decorrelación [16] consiste en obtener una estimación completamente libre de MAI e ISI del símbolo $b_i(n)$ transmitido por el usuario i . Es decir,

$$y(n) = \mathbf{w}^H(\mathbf{D}\mathbf{b}(n) + \mathbf{g}(n)) = b_i(n) + g_f(n) \quad (3.3)$$

donde \mathbf{D} es la matriz de códigos distorsionada, de dimensiones $L \times N$, $\mathbf{b}(n)$ es un vector $N \times 1$ con los símbolos n -ésimos de todos los usuarios del sistema, $\mathbf{g}(n)$ es el vector $L \times 1$, que constituye el ruido aditivo, y \mathbf{w} es el filtro decorrelador que se quiere calcular. Una vez aplicado el filtro, se obtiene el símbolo que transmitió el usuario $b_i(n)$ junto con un término $g_f(n) = \mathbf{w}^H \mathbf{g}(n)$, que es el ruido existente a la salida del filtro.

El filtro se elige como la i -ésima fila de la pseudoinversa por la izquierda de la matriz de códigos distorsionada, que denotaremos por \mathbf{D}^\dagger . Si la matriz \mathbf{D} es $L \times N$,

entonces la pseudoinversa \mathbf{D}^\dagger es $N \times L$ y la podemos escribir como

$$\mathbf{D}^\dagger = \begin{bmatrix} \mathbf{f}_1^H \\ \vdots \\ \mathbf{f}_N^H \end{bmatrix} \quad (3.4)$$

donde \mathbf{f}_k , $k = 1, \dots, N$ es un vector columna $L \times 1$. Con esta notación, el filtro de decorrelación para el i -ésimo usuario es $\mathbf{w} = \mathbf{f}_i$.

El resultado del producto $\mathbf{D}^\dagger \mathbf{D}$ es una matriz identidad de orden N . Por tanto, al multiplicar la i -ésima fila de \mathbf{D}^\dagger por la matriz \mathbf{D} resulta un vector fila de dimensiones $1 \times N$ en el que todos los elementos son nulos, excepto el i -ésimo que tiene valor unidad. De esa manera, el producto $\mathbf{w}^H \mathbf{D} \mathbf{b}(n)$ resulta en el i -ésimo elemento del vector columna $\mathbf{b}(n)$, que es el símbolo transmitido por el usuario deseado. Obsérvese que una matriz rectangular solo tiene inversa por la izquierda cuando el número de filas es mayor o igual que el de columnas, es decir, para emplear este método es necesario que la longitud del código L sea mayor o igual que el número de usuarios del sistema N .

El principal inconveniente del receptor decorrelador es la amplificación de la potencia del ruido a la salida del filtro. Se manifiesta en el término $g_f(n) = \mathbf{w}^H \mathbf{g}(n)$ que aparece a la salida del filtro. El grado de amplificación depende del condicionamiento de la matriz de códigos \mathbf{D} y, por tanto, del canal.

3.1.2. Detector MMSE

Un criterio intuitivamente atractivo para elegir adecuadamente los coeficientes del receptor es la minimización del error cuadrático medio esperado entre la estimación blanda que proporciona el filtro y el símbolo efectivamente transmitido. Esta es la idea fundamental del criterio MMSE o *Minimum Mean Square Error* [16, 7]. La MAI, la ISI y el ruido se eliminan en la medida en que se minimice el mencionado error cuadrático medio mediante una correcta selección de los coeficientes del

filtro. Es por tanto un problema de optimización, cuya formulación es

$$\mathbf{w}_{MMSE} = \arg \min_{\mathbf{w}} \{ \Gamma_{MMSE}(\mathbf{w}) = E_{\mathbf{x}(n)} [|y(n) - b_i(n)|^2] \} \quad (3.5)$$

donde $E_{\mathbf{x}(n)}[\cdot]$ es la esperanza estadística respecto del vector de observaciones $\mathbf{x}(n)$. Se trata de hallar el mínimo de $\Gamma_{MMSE}(\mathbf{w})$ respecto de \mathbf{w} . La función $\Gamma_{MMSE}(\mathbf{w})$ es una función cuadrática y, por tanto, el problema (3.5) tiene una solución única. Para hallar el mínimo de la función, el primer paso es desarrollar la esperanza en (3.5)

$$\begin{aligned} \Gamma_{MMSE}(\mathbf{w}) &= E_{\mathbf{x}(n)} [|y(n) - b_i(n)|^2] \\ &= E_{\mathbf{x}(n)} [|\mathbf{w}^H \mathbf{x}(n) - b_i(n)|^2] \\ &= E_{\mathbf{x}(n)} [(\mathbf{w}^H \mathbf{x}(n) - b_i(n))(\mathbf{w}^H \mathbf{x}(n) - b_i(n))^H] \\ &= E_{\mathbf{x}(n)} [\mathbf{w}^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{w}^H \mathbf{x}(n) b_i^H(n) - b_i(n) \mathbf{x}^H(n) \mathbf{w} \\ &\quad + b_i(n) b_i^H(n)] \end{aligned} \quad (3.6)$$

A continuación se procede a calcular el gradiente con respecto a \mathbf{w}^H de la función $\Gamma_{MMSE}(\mathbf{w})$ desarrollada en (3.6):

$$\begin{aligned} \nabla_{\mathbf{w}^H} [\Gamma_{MMSE}(\mathbf{w})] &= \nabla_{\mathbf{w}^H} [E_{\mathbf{x}(n)} [\mathbf{w}^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{w}^H \mathbf{x}(n) b_i^H(n) \\ &\quad - b_i(n) \mathbf{x}^H(n) \mathbf{w} + b_i(n) b_i^H(n)]] \end{aligned} \quad (3.7)$$

Puesto que tanto el gradiente como la esperanza son operadores lineales, se verifica que

$$\nabla_{\mathbf{w}^H} [E_{\mathbf{x}(n)}[\cdot]] = E_{\mathbf{x}(n)} [\nabla_{\mathbf{w}^H}[\cdot]] \quad (3.8)$$

Por tanto

$$\begin{aligned} \nabla_{\mathbf{w}^H} [\Gamma_{MMSE}(\mathbf{w})] &= E_{\mathbf{x}(n)} [\nabla_{\mathbf{w}^H} [\mathbf{w}^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{w}^H \mathbf{x}(n) b_i^H(n) \\ &\quad - b_i(n) \mathbf{x}^H(n) \mathbf{w}]] \end{aligned} \quad (3.9)$$

y utilizando las propiedades del gradiente y la esperanza [15]

$$\begin{aligned}\nabla_{\mathbf{w}^H} [\Gamma_{MMSE}(\mathbf{w})] &= E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w} - \mathbf{x}(n)b_i^H(n)] \\ &= E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)] \mathbf{w} - E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)]\end{aligned}\quad (3.10)$$

Por último, para obtener el vector \mathbf{w} que minimiza la función, hay que igualar el gradiente a cero y despejar \mathbf{w} de la ecuación resultante

$$\begin{aligned}0 &= E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)] \mathbf{w} - E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)] \\ &\quad \Downarrow \\ E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)] \mathbf{w} &= E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)] \\ &\quad \Downarrow \\ \mathbf{w} &= (E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)])^{-1} E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)]\end{aligned}\quad (3.11)$$

El filtro obtenido mediante el criterio MMSE puede expresarse como

$$\mathbf{w}_{MMSE} = \mathbf{R}_x^{-1} \mathbf{p} \quad (3.12)$$

donde $\mathbf{R}_x = E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)]$ es la matriz de autocorrelación de las observaciones y tiene dimensiones $L \times L$ y $\mathbf{p} = E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)]$ es un vector columna $L \times 1$ que representa la correlación entre las observaciones y el símbolo deseado.

La matriz \mathbf{R}_x puede expresarse de una manera más conveniente si se desarrolla la expresión que se encuentra en el interior de la esperanza

$$\begin{aligned}\mathbf{R}_x &= E_{\mathbf{X}(n)} [\mathbf{x}(n)\mathbf{x}^H(n)] = E [(\mathbf{D}\mathbf{b}(n) + \mathbf{g}(n))(\mathbf{D}\mathbf{b}(n) + \mathbf{g}(n))^H] \\ &= \sigma_b^2 \mathbf{D}\mathbf{D}^H + \sigma_g^2 \mathbf{I}_L\end{aligned}\quad (3.13)$$

donde se ha tenido en cuenta que el ruido y los símbolos son estadísticamente independientes (por lo que $E_{\mathbf{X}(n)} [\mathbf{b}(n)\mathbf{g}^H(n)] = E_{\mathbf{X}(n)} [\mathbf{b}(n)] E_{\mathbf{X}(n)} [\mathbf{g}^H(n)]$), y que el ruido tiene media cero ($E_{\mathbf{X}(n)} [\mathbf{g}(n)] = 0$). La potencia de símbolo se ha denotado por σ_b^2 , y se verifica que $\sigma_b^2 = E [|b_i(n)|^2] \forall i, n$

Por otro lado, el factor \mathbf{p} de la ecuación (3.12) también puede desarrollarse de la siguiente forma:

$$\begin{aligned}
\mathbf{p} &= E_{\mathbf{X}(n)} [\mathbf{x}(n)b_i^H(n)] = E_{\mathbf{X}(n)} [(\mathbf{D}\mathbf{b}(n) + \mathbf{g}(n))b_i^H(n)] \\
&= \mathbf{D}E_{\mathbf{X}(n)} [\mathbf{b}(n)b_i^H(n)] + E_{\mathbf{X}(n)} [\mathbf{g}(n)b_i^H(n)] = \mathbf{D}E_{\mathbf{X}(n)} [\mathbf{b}(n)b_i^H(n)] \\
&= \mathbf{D}E_{\mathbf{X}(n)} \begin{bmatrix} b_1(n)b_i^H(n) \\ \vdots \\ |b_i(n)|^2 \\ \vdots \\ b_N(n)b_i^H(n) \end{bmatrix} = \mathbf{D} \begin{bmatrix} 0 \\ \vdots \\ \sigma_b^2 \\ \vdots \\ 0 \end{bmatrix} = \sigma_b^2 \mathbf{d}_i \quad (3.14)
\end{aligned}$$

siendo \mathbf{d}_i la i -ésima columna de la matriz \mathbf{D} .

Finalmente, dadas las ecuaciones (3.13) y (3.14), el filtro MMSE para el usuario i puede escribirse de la siguiente manera:

$$\mathbf{w}_{MMSE} = (\sigma_b^2 \mathbf{D}\mathbf{D}^H + \sigma_g^2 \mathbf{I}_L)^{-1} \sigma_b^2 \mathbf{d}_i \quad (3.15)$$

3.1.3. Detector LS

El criterio de mínimos cuadrados (*Least Squares* o, de forma abreviada, LS) consiste en minimizar la suma de los errores cuadráticos entre las estimaciones blandas, $y(n)$, y la salida deseada, $b_i(n)$, siendo i el usuario deseado. Es un criterio similar al MMSE, pero mientras que MMSE minimiza el promedio estadístico del error cuadrático, LS minimiza la suma de los errores cuadráticos de un conjunto de observaciones, que es un promedio temporal. Así pues MMSE es un criterio estocástico, mientras que LS es un criterio determinista. Por la ergodicidad de los procesos considerados, no es difícil probar que la solución LS converge a la MMSE a medida que crece el número de observaciones [5].

El objetivo de LS es hallar \mathbf{w}_{LS} tal que

$$\mathbf{w}_{LS} = \arg \min_{\mathbf{w}} \left\{ \Gamma_{LS}(\mathbf{w}) = \frac{1}{K} \sum_{n=0}^{K-1} |y(n) - b_i(n)|^2 \right\} \quad (3.16)$$

donde K es el número de observaciones disponibles, que se corresponderá con la longitud de la trama.

Resulta un problema de optimización muy similar al de la sección 3.1.2, que se resuelve de manera análoga:

$$\begin{aligned}
\Gamma_{LS}(\mathbf{w}) &= \frac{1}{K} \sum_{n=0}^{K-1} |y(n) - b_i(n)|^2 = \frac{1}{K} \sum_{n=0}^{K-1} (y(n) - b_i(n))(y(n) - b_i(n))^* \\
&= \frac{1}{K} \sum_{n=0}^{K-1} (\mathbf{w}^H \mathbf{x}(n) - b_i(n))(\mathbf{w}^H \mathbf{x}(n) - b_i(n))^* \\
&= \frac{1}{K} \sum_{n=0}^{K-1} (\mathbf{w}^H \mathbf{x}(n) - b_i(n))(\mathbf{x}^H(n) \mathbf{w} - b_i^*(n)) \\
&= \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{w}^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{w}^H \mathbf{x}(n) b_i^*(n) - b_i(n) \mathbf{x}^H(n) \mathbf{w} + \\
&\quad + |b_i(n)|^2
\end{aligned} \tag{3.17}$$

Se calcula el gradiente de la función:

$$\begin{aligned}
\nabla_{\mathbf{w}^H}(\Gamma_{LS}(\mathbf{w})) &= \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{x}(n) b_i^*(n) \\
&= \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) b_i^*(n)
\end{aligned} \tag{3.18}$$

Igualando el resultado a cero,

$$\frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} = \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) b_i^*(n) \tag{3.19}$$

Así pues,

$$\begin{aligned}
\mathbf{w}_{LS} &= \underbrace{\left(\frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \right)^{-1}}_{\hat{\mathbf{R}}_x} \underbrace{\left(\frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n) b_i^*(n) \right)}_{\hat{\mathbf{p}}} \\
&= \left(\sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \right)^{-1} \left(\sum_{n=0}^{K-1} \mathbf{x}(n) b_i^*(n) \right)
\end{aligned} \tag{3.20}$$

donde $\hat{\mathbf{R}}_x = \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}^H(n)$ es una matriz $L \times L$ denominada matriz de autocorrelación empírica de las observaciones y $\hat{\mathbf{p}} = \frac{1}{K} \sum_{n=0}^{K-1} \mathbf{x}(n)b_i^*(n)$, de dimensiones $L \times 1$, es la correlación cruzada entre las observaciones y los símbolos transmitidos.

Para calcular $\hat{\mathbf{p}}$ es necesario conocer los símbolos transmitidos por el usuario i . En la práctica es habitual que solo se conozcan *a priori* M símbolos correspondientes a una secuencia de entrenamiento ($M \leq K$). Por lo tanto, y suponiendo que la secuencia de entrenamiento está constituida por los M primeros símbolos de la trama, $\hat{\mathbf{p}}$ se calculará de forma aproximada como $\hat{\mathbf{p}} = \frac{1}{M} \sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n)$. Además, se ha comprobado en las simulaciones que calculando $\hat{\mathbf{R}}_x$ utilizando tan sólo los símbolos de la secuencia de entrenamiento ($\hat{\mathbf{R}}_x = \frac{1}{M} \sum_{n=0}^{M-1} \mathbf{x}(n)\mathbf{x}^H(n)$), se obtienen unos resultados sensiblemente mejores que empleando todos los símbolos de la trama, por lo que será esta última expresión la que se empleará en las simulaciones para calcular la matriz de autocorrelación empírica. Como contrapartida a este mejora en el rendimiento, esta variante del algoritmo LS, conocida como LS no regularizado, tiene el inconveniente de que si M es muy pequeño, entonces la matriz $\hat{\mathbf{R}}_x$ está cerca de ser singular, lo cual conlleva problemas a la hora de invertirla.

3.1.4. Implementación adaptativa del criterio LS:

algoritmo RLS

Para calcular el filtro LS de acuerdo con la ecuación (3.20), es necesario llevar a cabo la inversión de la matriz $\tilde{\mathbf{R}}_x$. Ésta es una operación costosa que es deseable evitar. El algoritmo de mínimos cuadrados recursivo (RLS: *Recursive Least Squares*) [5] proporciona una forma de obtener los coeficientes del filtro que se utilizará para estimar el símbolo n -ésimo, en función de los coeficientes que se utilizaron para calcular el símbolo $n - 1$. Se trata de una implementación adaptativa del algoritmo LS de inferior complejidad computacional, ya que evita tener que invertir la matriz de autocorrelación $\tilde{\mathbf{R}}_x$.

Considerando la sucesión de estimadores LS:

$$\begin{aligned}\hat{\mathbf{w}}_{LS}(j) &= \left(\sum_{n=0}^j \mathbf{x}(n)\mathbf{x}^H(n) \right)^{-1} \sum_{n=0}^j \mathbf{x}(n)b_i^*(n) \\ &= \tilde{\mathbf{R}}_x^{-1}(j)\tilde{\mathbf{p}}(j)\end{aligned}\quad (3.21)$$

donde $\tilde{\mathbf{R}}_x(j) = \sum_{n=0}^j \mathbf{x}(n)\mathbf{x}^H(n)$ es una matriz $L \times L$, $\tilde{\mathbf{p}}(j) = \sum_{n=0}^j \mathbf{x}(n)b_i^*(n)$ es un vector columna $L \times 1$ y i es el usuario de interés¹. Cuando se cumple $j = K - 1$, el estimador resultante es el que se mostraba en la ecuación (3.20). Si se aísla el último término (cuando $n = j$) del sumatorio que define $\tilde{\mathbf{R}}_x(j)$, se obtiene:

$$\tilde{\mathbf{R}}_x(j) = \sum_{n=0}^{j-1} \mathbf{x}(n)\mathbf{x}^H(n) + \mathbf{x}(j)\mathbf{x}^H(j) \quad (3.22)$$

Pero tal como se ha definido $\tilde{\mathbf{R}}_x(j)$ (ecuación (3.21)), el primer sumando de la ecuación (3.22) es la matriz de autocorrelación $\tilde{\mathbf{R}}_x(j - 1)$. Así pues:

$$\tilde{\mathbf{R}}_x(j) = \tilde{\mathbf{R}}_x(j - 1) + \mathbf{x}(j)\mathbf{x}^H(j) \quad (3.23)$$

Del mismo modo, se puede obtener una expresión recursiva para la correlación cruzada $\tilde{\mathbf{p}}(j)$:

$$\tilde{\mathbf{p}}(j) = \tilde{\mathbf{p}}(j - 1) + \mathbf{x}(j)b_i^*(j) \quad (3.24)$$

El siguiente paso es invertir la matriz de autocorrelación $\tilde{\mathbf{R}}_x(j)$. Para ello se puede hacer uso del *lema de inversión* [5], que permite obtener la inversa de una matriz \mathbf{A} , definida positiva y de tamaño $R \times R$, que se pueda descomponer como

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^H \quad (3.25)$$

donde \mathbf{B} y \mathbf{D} son matrices definidas positivas de tamaño $R \times R$ y $S \times S$ respectivamente, y \mathbf{C} es una matriz de tamaño $R \times S$. Entonces, la inversa de \mathbf{A} se puede obtener como:

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^H\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^H\mathbf{B} \quad (3.26)$$

¹Nótese que $\hat{\mathbf{R}}_x = \frac{1}{K}\tilde{\mathbf{R}}_x(K - 1)$ y $\hat{\mathbf{p}} = \frac{1}{K}\tilde{\mathbf{p}}(K - 1)$

Si la matriz $\tilde{\mathbf{R}}_x(j)$ es definida positiva (y, por tanto, no singular), se puede aplicar el lema de inversion para obtener $\tilde{\mathbf{R}}_x^{-1}(j)$ si, estableciendo un paralelismo entre las ecuaciones (3.23) y (3.25), se consideran las siguientes equivalencias:

$$\begin{aligned}\mathbf{A} &= \tilde{\mathbf{R}}_x(j) \\ \mathbf{B}^{-1} &= \tilde{\mathbf{R}}_x(j-1) \\ \mathbf{C} &= \mathbf{x}(j) \\ \mathbf{D} &= 1\end{aligned}$$

Sustituyendo en el resultado que proporciona el lema de inversión (ecuación (3.26)), se obtiene:

$$\tilde{\mathbf{R}}_x^{-1}(j) = \tilde{\mathbf{R}}_x^{-1}(j-1) - \frac{\tilde{\mathbf{R}}_x^{-1}(j-1)\mathbf{x}(j)\mathbf{x}^H(j)\tilde{\mathbf{R}}_x^{-1}(j-1)}{1 + \mathbf{x}^H(j)\tilde{\mathbf{R}}_x^{-1}(j-1)\mathbf{x}(j)} \quad (3.27)$$

Para simplificar los cálculos se pueden definir:

$$\mathbf{P}(j) = \tilde{\mathbf{R}}_x^{-1}(j) \quad (3.28)$$

$$\mathbf{k}(j) = \frac{\mathbf{P}(j-1)\mathbf{x}(j)}{1 + \mathbf{x}^H(j)\mathbf{P}(j-1)\mathbf{x}(j)} \quad (3.29)$$

que permiten reescribir la ecuación (3.27) como:

$$\mathbf{P}(j) = \mathbf{P}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1) \quad (3.30)$$

A la vista de las ecuaciones (3.24) y (3.28), y sustituyendo en la ecuación (3.21) resulta:

$$\begin{aligned}\hat{\mathbf{w}}_{RLS}(j) &= \tilde{\mathbf{R}}_x^{-1}(j)\tilde{\mathbf{p}}(j) \\ &= \mathbf{P}(j)\tilde{\mathbf{p}}(j) \\ &= \mathbf{P}(j)\tilde{\mathbf{p}}(j-1) + \mathbf{P}(j)\mathbf{x}(j)b_i^*(j)\end{aligned} \quad (3.31)$$

Si ahora se reescribe el primer $\mathbf{P}(j)$ que aparece en la ecuación (3.31) utilizando la ecuación (3.30)

$$\begin{aligned}\hat{\mathbf{w}}_{RLS}(j) &= \mathbf{P}(j-1)\tilde{\mathbf{p}}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1)\tilde{\mathbf{p}}(j-1) + \mathbf{P}(j)\mathbf{x}(j)b_i^*(j) \\ &= \tilde{\mathbf{R}}_x^{-1}(j-1)\tilde{\mathbf{p}}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\tilde{\mathbf{R}}_x^{-1}(j-1)\tilde{\mathbf{p}}(j-1) + \mathbf{P}(j)\mathbf{x}(j)b_i^*(j) \\ &= \hat{\mathbf{w}}_{RLS}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1) + \mathbf{P}(j)\mathbf{x}(j)b_i^*(j).\end{aligned} \quad (3.32)$$

Por último, y teniendo en cuenta que la ecuación (3.29) puede expresarse como

$$\begin{aligned}
 \mathbf{k}(j) &= \mathbf{P}(j-1)\mathbf{x}(j) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1)\mathbf{x}(j) \\
 &= [\mathbf{P}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1)]\mathbf{x}(j) \\
 &= \mathbf{P}(j)\mathbf{x}(j)
 \end{aligned} \tag{3.33}$$

la ecuación final para actualizar los coeficientes del filtro RLS viene dada por

$$\begin{aligned}
 \hat{\mathbf{w}}_{RLS}(j) &= \hat{\mathbf{w}}_{RLS}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1) + \mathbf{k}(j)b_i^*(j) \\
 &= \hat{\mathbf{w}}_{RLS}(j-1) + \mathbf{k}(j) [b_i^*(j) - \mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1)]
 \end{aligned} \tag{3.34}$$

La expresión $\mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1)$ es el conjugado de una estimación blanda calculada usando el filtro de la iteración anterior. Por tanto, la expresión entre corchetes de la ecuación (3.34) es el conjugado del error que se obtendría empleando la última actualización del filtro disponible.

El cuadro 3.1 muestra el pseudocódigo del algoritmo arriba descrito. Obsérvese que la inicialización del filtro se realiza igualando el filtro a un vector de ceros ($\hat{\mathbf{w}}_{RLS}(-1) = \mathbf{0}$) y la matriz \mathbf{P} a una matriz identidad de orden L ($\mathbf{P}(-1) = \mathbf{I}_L$).

RLS dirigido por decisión

Como se puede observar en el cuadro 3.1, el algoritmo RLS necesita el j -ésimo símbolo transmitido por el usuario deseado, $b_i(j)$, para obtener la j -ésima actualización de los coeficientes del filtro. Por tanto sólo es posible actualizar los coeficientes del filtro M veces, una vez por cada símbolo de la secuencia de entrenamiento. Sin embargo, para $j = M, \dots, K-1$ podría emplearse una estimación dura del símbolo transmitido por el usuario obtenida empleando la última actualización del filtro disponible. Esta variación del algoritmo RLS recibe el nombre de RLS dirigido por decisión, o de forma abreviada RLS DD.

Si el filtro calculado por medio del algoritmo RLS proporciona buenas estimaciones de los símbolos transmitidos por el usuario deseado, entonces el empleo del

Inicialización del algoritmo

$$\hat{\mathbf{w}}_{RLS}(-1) = \mathbf{0}_{L \times 1}$$

$$\mathbf{P}(-1) = \mathbf{I}_L$$

Para $j = 0, \dots, M - 1$ (secuencia de entrenamiento)

$$\mathbf{k}(j) = \frac{\mathbf{P}(j-1)\mathbf{x}(j)}{1 + \mathbf{x}^H(j)\mathbf{P}(j-1)\mathbf{x}(j)}$$

$$\hat{\mathbf{w}}_{RLS}(j) = \hat{\mathbf{w}}_{RLS}(j-1) + \mathbf{k}(j) [b_i^*(j) - \mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1)]$$

$$\mathbf{P}(j) = \mathbf{P}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1)$$

Cuadro 3.1: Pseudocódigo del algoritmo RLS.

algoritmo RLS DD será beneficioso pues la solución obtenida mediante éste último se acercará más a la óptima. Es decir, cada estimación dura que se emplee para actualizar el filtro y coincida con el símbolo transmitido por el usuario, tiene el efecto que tendría el disponer de una secuencia de entrenamiento una unidad mayor.

Un inconveniente de este método es que está sujeto a un fenómeno de propagación de error, ya que si la estimación dura empleada para actualizar el filtro es incorrecta, entonces los coeficientes del filtro se alejarán de la solución óptima y las futuras estimaciones que se obtengan con el filtro recién actualizado serán peores que las obtenidas con el filtro de la iteración anterior. Ello ocasionará muy posiblemente más errores en las estimaciones duras que nuevamente se emplearán para actualizar incorrectamente los coeficientes del filtro. De esa manera el error cometido en una estimación dura se va propagando a medida que se actualiza el filtro.

El pseudocódigo de este algoritmo se muestra en el cuadro 3.2.

3.2. Conclusiones

En el receptor se dispone de un vector de observaciones que contiene los símbolos transmitidos en un momento dado por los N usuarios del sistema. De ese vector de observaciones se pretende extraer el símbolo transmitido por un usuario deseado. Para ello se empleará un filtro transversal de respuesta al impulso finita (FIR), que será el encargado de eliminar la MAI, igualar el canal y mitigar los efectos del ruido térmico.

Básicamente un receptor lineal consta del mencionado filtro FIR y de un detector de umbral que decidirá que símbolo del alfabeto de la modulación ha de asignarse a cada estimación blanda (calculada como el producto escalar entre el vector de coeficientes del filtro y el vector de observaciones). El símbolo del alfabeto de la modulación que hay a la salida del detector de umbral constituye una estimación dura y es el símbolo que se considera que transmitió el emisor.

Inicialización del algoritmo

$$\hat{\mathbf{w}}_{RLS}(-1) = \mathbf{0}_{L \times 1}$$

$$\mathbf{P}(-1) = \mathbf{I}_L$$

Para $j = 0, \dots, M - 1$ (secuencia de entrenamiento)

$$\mathbf{k}(j) = \frac{\mathbf{P}(j-1)\mathbf{x}(j)}{1 + \mathbf{x}^H(j)\mathbf{P}(j-1)\mathbf{x}(j)}$$

si $j < M$

$$\hat{b}_i(j) = b_i(j)$$

en caso contrario

$$\hat{b}_i(j) = \arg \min_{a \in \mathcal{A}} \{ |\mathbf{w}_{RLS}^H(j-1)\mathbf{x}(j) - a| \}$$

fin si

$$\hat{\mathbf{w}}_{RLS}(j) = \hat{\mathbf{w}}_{RLS}(j-1) + \mathbf{k}(j) \left[\hat{b}_i^*(j) - \mathbf{x}^H(j)\hat{\mathbf{w}}_{RLS}(j-1) \right]$$

$$\mathbf{P}(j) = \mathbf{P}(j-1) - \mathbf{k}(j)\mathbf{x}^H(j)\mathbf{P}(j-1)$$

Cuadro 3.2: Pseudocódigo del algoritmo RLS DD.

En este capítulo se han estudiado las aproximaciones más clásicas para el diseño de receptores lineales. Los receptores MMSE y decorrelador requieren conocer perfectamente el canal \mathbf{H} y son por ello difíciles de llevar a la práctica (requieren una etapa previa de estimación del canal). En este trabajo se utilizarán básicamente como referencia para evaluar el rendimiento de los restantes métodos. Los algoritmos LS, RLS y RLS DD, al contrario, sí son aproximaciones prácticas, aunque, como se verá, tienen importantes limitaciones derivadas de que se requiere conocer un amplio subconjunto de los símbolos transmitidos para que los coeficientes del receptor calculados proporcionen unos resultados satisfactorios.

Capítulo 4

Supresión de interferencias basada en el principio ML

En este capítulo se presenta una nueva aproximación al problema de la supresión de interferencias en un sistema MC-CDMA. Al igual que las técnicas clásicas vistas en el capítulo 2, el método propuesto consiste en construir un filtro lineal transversal de respuesta al impulso finita que cancela la MAI, iguala el canal y mitiga en la medida de lo posible los efectos del ruido. La novedad de esta técnica estriba en el criterio de selección de los coeficientes del filtro, que en este caso se calcularán de acuerdo con el principio de máxima verosimilitud (ML, del inglés *Maximum Likelihood*).

La idea básica del método propuesto es la siguiente: suponiendo que existe un filtro *óptimo* capaz de suprimir por completo la MAI e igualar el canal al mismo tiempo, una vez aplicado éste sobre el vector de observaciones se obtendría una estimación blanda que sería una suma de dos variables aleatorias. Una de ellas sería el símbolo transmitido por el usuario y la otra se correspondería con un término de ruido. Es sencillo calcular la f.d.p. (*función de densidad de probabilidad*) de esta estimación blanda que, claramente, depende de los coeficientes del filtro pues resulta del producto escalar entre el vector formado por éstos y el vector de observaciones.

Entonces se plantea el estimador ML de los coeficientes del filtro, que es la solución a un problema de optimización consistente en maximizar la f.d.p. de las estimaciones blandas que se obtendrían con el filtro óptimo (y que por tanto son suma de las dos variables aleatorias arriba mencionadas), con respecto a los coeficientes del filtro. De esta manera, se asegura que los coeficientes del filtro así seleccionados dan lugar a estimaciones blandas con una distribución de probabilidad cercana a la deseada.

Esta técnica se aplicó por primera vez en un sistema CDMA por secuencia directa (por tanto, con modulación monoportadora) en [1], donde también puede encontrarse una justificación teórica rigurosa del método.

Este método aprovecha el conocimiento estadístico *a priori* de la señal deseada (se sabe que es suma de dos variables aleatorias: una correspondiente al símbolo transmitido por el usuario y otra de ruido). Pero esta caracterización estadística es común a todos los usuarios del sistema. Por lo tanto, se necesita algún tipo de información adicional para asegurar que el receptor obtiene efectivamente los símbolos transmitidos por el usuario deseado y no los transmitidos por alguno de los usuarios interferentes. Hay dos posibilidades:

- Aproximación semiciega: Se conocen en el receptor un subconjunto de los símbolos transmitidos por el usuario. Es decir, se hace uso de una secuencia de entrenamiento. El conocimiento *a priori* de éstos símbolos impone una restricción en el cálculo de los coeficientes del filtro, obligando a éste a capturar la señal del usuario deseado y no las de los usuarios interferentes. Los sistemas de comunicaciones móviles de tercera generación emplean protocolos cuyas tramas incluyen una secuencia de entrenamiento, por lo que es realista suponer que algunos símbolos de la trama son conocidos a priori. Obviamente se debe asignar una secuencia de entrenamiento diferente a cada usuario, o de lo contrario el filtro podría capturar la señal de un usuario que no es el deseado.
- Aproximación ciega: Se impone una restricción lineal adecuada sobre los co-

eficientes del filtro que garantice que el filtro extrae los símbolos del usuario deseado.

No es posible hallar una solución analítica en ninguno de los dos casos, por lo que será preciso emplear algún método numérico para estimar los parámetros del receptor. En el presente trabajo se estudiará la aproximación semiciega y su implementación por medio del algoritmo EM (*Expectation Maximization*), que transforma el problema de maximizar la verosimilitud en una serie de problemas más sencillos que sí tienen solución analítica.

4.1. Estimación de los parámetros del receptor

Cuando el filtro del receptor elimina la MAI y compensa la distorsión producida por el canal, la estimación blanda $y(n)$ es la suma de dos variables aleatorias: el símbolo transmitido por el usuario deseado, $b_i(n)$, y un término de ruido aditivo gaussiano $g_f(n)$. Si denotamos como \mathbf{w}_* el filtro óptimo capaz de igualar el canal y eliminar por completo la MAI, resulta:

$$y(n) = \mathbf{w}_*^H \mathbf{x}(n) = A_i b_i(n) + g_f(n) \quad (4.1)$$

donde A_i es un factor complejo, que depende tanto del canal como del filtro. El término $g_f(n)$ es una variable aleatoria compleja gaussiana con media cero y varianza

$$\begin{aligned} \sigma_f^2 &= E[|g_f(n)|^2] = E[|\mathbf{w}_*^H \mathbf{g}(n)|^2] = E[\mathbf{w}_*^H \mathbf{g}(n) \mathbf{g}^H(n) \mathbf{w}_*] \\ &= \mathbf{w}_*^H E[\mathbf{g}(n) \mathbf{g}^H(n)] \mathbf{w}_* = \sigma_g^2 \mathbf{w}_*^H \mathbf{w}_* \end{aligned} \quad (4.2)$$

siendo $E[\mathbf{g}(n) \mathbf{g}^H(n)] = \sigma_g^2 \mathbf{I}_L$ la matriz de covarianza del ruido a la entrada del filtro.

En [1] se demuestra que la f.d.p. de una estimación blanda $y(n)$ es

$$f_{y;\Theta}(y(n)) = f_{A_i b_i(n) + g_f(n)} = \frac{1}{\pi \sigma_f^2} E_{b_i(n)} \left[e^{-\frac{|y(n) - A_i b_i(n)|^2}{\sigma_f^2}} \right] \quad (4.3)$$

donde $\Theta = [\mathbf{w}_*, A_i]$ es un conjunto de parámetros a estimar y $E_{b_i}[\cdot]$ es la esperanza respecto al símbolo transmitido por el usuario deseado. Los símbolos transmitidos por los distintos usuarios $(b_1(n), b_2(n), \dots, b_N(n))$ se modelan como variables aleatorias discretas i.i.d. (*independientes e idénticamente distribuidas*) con f.m.p. (*función de masa de probabilidad*) conocida y alfabeto finito.

Como se vio en la sección 1.3, en los sistemas multiportadora el efecto dispersivo del canal es despreciable en comparación con el que se produce en los sistemas monoportadora por lo que se puede considerar que no hay ISI, es decir, los símbolos transmitidos no interfieren entre sí. Si los símbolos transmitidos no interfieren los unos con los otros, tampoco lo harán las observaciones ni las estimaciones blandas que de ellas se deriven. Por lo tanto, podemos considerar que las estimaciones blandas correspondientes a una trama de K símbolos son independientes entre sí. Entonces la f.d.p. conjunta es equivalente al producto de las f.d.p. marginales de cada una de las estimaciones

$$f_{\mathbf{y};\Theta}(\mathbf{y}) = \prod_{n=0}^{K-1} f_{y;\Theta}(y(n)) = \left(\frac{1}{\pi\sigma_f^2} \right)^K \prod_{n=0}^{K-1} E_{b_i(n)} \left[e^{-\frac{|y(n)-A_i b_i(n)|^2}{\sigma_f^2}} \right] \quad (4.4)$$

donde $\mathbf{y} = [y(0), y(1), \dots, y(K-1)]^T$.

Una vez calculada la f.d.p. conjunta, se puede aplicar fácilmente el principio de estimación de máxima verosimilitud, que resulta en el problema de optimización

$$\hat{\Theta} = \arg \max_{\Theta} \{ f_{\mathbf{y};\Theta}(\mathbf{y}) \} \quad (4.5)$$

donde $\hat{\Theta} = [\hat{\mathbf{w}}, \hat{A}_i]$ son estimaciones de los coeficientes del receptor lineal, \mathbf{w} , y de la amplitud compleja, A_i . En lugar de maximizar la función $f_{\mathbf{y};\Theta}$ directamente, y con el objetivo de simplificar la resolución del problema, se puede maximizar el logaritmo natural de esa función,

$$\hat{\Theta} = \arg \max_{\Theta} \{ \mathcal{L}(\Theta) \} \quad (4.6)$$

donde

$$\mathcal{L}(\Theta) = \log \left(\prod_{n=0}^{K-1} E_{b_i(n)} \left[e^{-\frac{|y(n) - A_j b_i(n)|^2}{\sigma_j^2}} \right] \right) = \sum_{n=0}^{K-1} \log \left(E_{b_i(n)} \left[e^{-\frac{|y(n) - A_j b_i(n)|^2}{\sigma_j^2}} \right] \right) \quad (4.7)$$

es la verosimilitud logarítmica de Θ respecto al bloque de estimaciones $\mathbf{y} = [y(0), \dots, y(K-1)]^T$ y ha sido obtenida hallando el logaritmo de la función de densidad que aparece en la ecuación (4.4), e ignorando las constantes que no aportan nada al problema de optimización. La función $\mathcal{L}(\Theta)$ es una función no cuadrática que tiene varios máximos locales. La solución al problema (4.6) garantiza que las estimaciones blandas son de la forma dada por la ecuación (4.1), es decir, que tienen una f.d.p. próxima a $f_{A_i b_i(n) + g_f(n)}$. Sin embargo, en los sistemas de comunicaciones CDMA todos los usuarios utilizan la misma modulación, por lo que las variables aleatorias $b_1(n), b_2(n), \dots, b_N(n)$ están idénticamente distribuidas. Por lo tanto, las f.d.p. de las estimaciones blandas de los diferentes usuarios solo se diferencian en la amplitud compleja (A_j , $j = 1, \dots, N$), que es desconocida *a priori*, y al resolver el problema se puede obtener una solución que *capture* la señal de uno de los usuarios interferentes. Esta ambigüedad surge también en otros métodos aplicados en la cancelación de MAI y se suele denominar *problema de captura* [9].

4.2. Supresión semiciega de interferencias

Una posibilidad para resolver el problema de captura descrito en la sección 4.1 es utilizar una secuencia de entrenamiento. Se obtiene así una aproximación semiciega que emplea tanto el conocimiento *a priori* de un conjunto de símbolos incluidos en la trama como la información estadística de la señal recibida.

Como ya se comentó al inicio del capítulo, el problema de optimización derivado del criterio ML no tiene solución analítica. Por consiguiente, para maximizar la función de la ecuación (4.7) será preciso emplear un método numérico. En la sección 4.2.1 se describirá el uso del algoritmo EM con este propósito.

Sin pérdida de generalidad, se puede suponer que los símbolos de la secuencia de entrenamiento, \mathbf{b}_t , son los M primeros de la trama, es decir, $\mathbf{b}_t = [b_i(0), \dots, b_i(M-1)]^T$. Condicionando la esperanza que aparece en la ecuación (4.7) respecto de los símbolos de la secuencia de entrenamiento, que ya son conocidos *a priori*, se obtiene un filtro cuyos coeficientes se calculan como la solución a

$$\begin{aligned} \hat{\mathbf{w}} &= \arg \max_{\mathbf{w}_*} \left\{ \mathcal{L}(\mathbf{w}_*) | \mathbf{b}_t = \sum_{n=0}^{K-1} \log \left(E_{b_i(n)} | \mathbf{b}_t \left[e^{-\frac{|y(n)-A_i b_i(n)|^2}{\sigma_f^2}} \right] \right) \right\} \\ &= \arg \min_{\mathbf{w}_*} \left\{ \sum_{n=0}^{M-1} |(y(n) - A_i b_i(n))|^2 - \sum_{n=M}^{K-1} \log \left(E_{b_i(n)} \left[e^{-\frac{|y(n)-A_i b_i(n)|^2}{\sigma_f^2}} \right] \right) \right\}. \end{aligned} \quad (4.8)$$

Puesto que el primer término de la ecuación (4.8) es cuadrático y, por tanto, tiene un único mínimo, el hecho de conocer *a priori* un conjunto de símbolos suficientemente grande de la trama restringe el problema planteado en las ecuaciones (4.6) y (4.7), y lo convierte en un problema con una solución única. Otra ventaja de conocer una secuencia de símbolos es que es posible corregir la amplitud y la fase de $y(n)$ de manera que coincida con la de $b_i(0), \dots, b_i(M-1)$. Esto se traduce en la posibilidad de fijar $A_i = 1$ *a priori* y calcular el filtro receptor como

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}_*} \left\{ \sum_{n=0}^{M-1} |(y(n) - b_i(n))|^2 - \sum_{n=M}^{K-1} \log \left(E_{b_i(n)} \left[e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right] \right) \right\} \quad (4.9)$$

Así pues, la resolución del problema planteado por la ecuación (4.9) permite obtener un filtro capaz de extraer los símbolos del usuario deseado, evitando la captura de la señal de alguno de los usuarios interferentes. Sin embargo, el problema no tiene una solución cerrada, por lo que para resolverlo será necesario emplear algún método numérico. Un método muy adecuado a este tipo de problemas es el algoritmo EM.

4.2.1. El algoritmo EM

El algoritmo EM [8] se emplea fundamentalmente en problemas de estimación paramétrica. Parte de la hipótesis de que existen un conjunto de *datos no observa-*

dos, cuyo conocimiento simplificaría notablemente el problema de estimación. Es un algoritmo iterativo que consta de dos etapas:

1. **Etapa E:** Los *datos incompletos* (observados) y la estimación actual de los parámetros se utilizan para calcular un conjunto de estadísticos suficientes para los *datos completos*. Los datos completos resultan de la unión de los datos incompletos y los datos no observados.
2. **Etapa M:** A partir de los estadísticos calculados en la etapa E, se actualiza la estimación de los parámetros.

Se puede demostrar que la secuencia de estimaciones así construida es monótona no decreciente en verosimilitud [8, 15].

4.2.2. Supresión semiciega de interferencias aplicando EM

En el problema planteado por la ecuación (4.9), los datos incompletos son las estimaciones blandas $y(n)$, $n = 0, \dots, K-1$, los datos no observados son los símbolos de la trama que no forman parte de la secuencia de entrenamiento $b_i(n)$, $n = M, \dots, K-1$, y los datos completos son vectores extendidos formados a partir de una estimación blanda añadiendo su correspondiente símbolo transmitido $\mathbf{y}_e(n) = [y(n), b_i(n)]^T$, $n = 0, \dots, K-1$. Se pueden agrupar en una matriz de datos completos $\mathbf{Y}_e = [\mathbf{y}_e(0), \dots, \mathbf{y}_e(K-1)]$. Aplicando la definición de probabilidad condicionada, la f.d.p. de \mathbf{Y}_e , $f_{\mathbf{Y}_e; \mathbf{w}_*}(\cdot)$ puede descomponerse como

$$f_{\mathbf{Y}_e; \mathbf{w}_*}(\mathbf{Y}_e) = f_{\mathbf{Y}_e | \mathbf{y}; \mathbf{w}_*}(\mathbf{Y}_e) f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y}) \quad (4.10)$$

donde \mathbf{w}_* se refiere al filtro óptimo e $\mathbf{y} = [y(0), \dots, y(K-1)]$.

Calculando el logaritmo de las expresiones a ambos lados de la igualdad (4.10), despejando a continuación la verosimilitud $f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y})$, y calculando luego la esperanza respecto de la variable $\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T$, que representa el condicionamiento de los datos completos respecto de los datos observados y la secuencia de entrenamiento,

resulta

$$\begin{aligned} \log(f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y})) &= E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} [\log(f_{\mathbf{Y}_e; \mathbf{w}_*}(\mathbf{Y}_e)) \\ &\quad - \log(f_{\mathbf{Y}_e | \mathbf{y}; \mathbf{w}_*}(\mathbf{Y}_e))] \end{aligned} \quad (4.11)$$

puesto que trivialmente se verifica que

$$E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} [\log(f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y}))] = \log(f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y})) \quad (4.12)$$

Definiendo

$$\mathcal{U}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) = E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} \left[\log \left(f_{\mathbf{Y}_e; \mathbf{w}_*}(\mathbf{Y}_e) \right) \right] \quad (4.13)$$

$$\mathcal{V}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) = E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} \left[\log \left(f_{\mathbf{Y}_e | \mathbf{y}; \mathbf{w}_*}(\mathbf{Y}_e) \right) \right] \quad (4.14)$$

y sustituyendo las ecuaciones (4.13) y (4.14) en la ecuación (4.11) se obtiene

$$\log(f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y})) = \mathcal{U}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) - \mathcal{V}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) \quad (4.15)$$

que es la expresión de la verosimilitud logarítmica $\log(f_{\mathbf{y}; \mathbf{w}_*}(\mathbf{y}))$ en base a las funciones \mathcal{U} y \mathcal{V} .

No es difícil usar la desigualdad de Jensen para demostrar que [15]

$$\mathcal{V}(\mathbf{w}, \hat{\mathbf{w}}(i)) \leq \mathcal{V}(\hat{\mathbf{w}}(i), \hat{\mathbf{w}}(i)) \quad \forall \mathbf{w} \quad (4.16)$$

Teniendo en cuenta ésto y eligiendo $\hat{\mathbf{w}}(i+1)$ tal que

$$\mathcal{U}(\hat{\mathbf{w}}(i+1), \hat{\mathbf{w}}(i)) \geq \mathcal{U}(\hat{\mathbf{w}}(i), \hat{\mathbf{w}}(i)) \quad (4.17)$$

la verosimilitud logarítmica dada por la ecuación (4.15) es monótona no decreciente con respecto a cada nueva estimación del filtro pues

$$\begin{aligned} \log \left(f_{\mathbf{y}; \hat{\mathbf{w}}(i+1)}(\mathbf{y}) \right) &= \mathcal{U}(\hat{\mathbf{w}}(i+1), \hat{\mathbf{w}}(i)) - \mathcal{V}(\hat{\mathbf{w}}(i+1), \hat{\mathbf{w}}(i)) \\ &\geq \mathcal{U}(\hat{\mathbf{w}}(i), \hat{\mathbf{w}}(i)) - \mathcal{V}(\hat{\mathbf{w}}(i), \hat{\mathbf{w}}(i)) = \log(f_{\mathbf{y}; \hat{\mathbf{w}}(i)}(\mathbf{y})) \end{aligned} \quad (4.18)$$

Por lo tanto, cada nueva estimación del filtro se va a obtener como

$$\hat{\mathbf{w}}(i+1) = \arg \max_{\mathbf{w}} \{\mathcal{U}(\mathbf{w}, \hat{\mathbf{w}}(i))\}. \quad (4.19)$$

En la sección 4.1 se explicó que, puesto que en el sistema de comunicaciones considerado no hay ISI, tanto los símbolos transmitidos como sus estimaciones blandas correspondientes son independientes. Por lo tanto, la f.d.p. de la matriz de datos completos, $f_{\mathbf{Y}_e; \mathbf{w}_*}$, se puede calcular como el producto de las f.d.p. marginales, $f_{\mathbf{y}_e; \mathbf{w}_*}$, de cada uno de los vectores extendidos, es decir,

$$f_{\mathbf{Y}_e; \mathbf{w}_*}(\mathbf{Y}_e) = \prod_{n=0}^{K-1} f_{\mathbf{y}_e; \mathbf{w}_*}(\mathbf{y}_e) \quad (4.20)$$

donde, a su vez, la f.d.p. $f_{\mathbf{y}_e; \mathbf{w}_*}$ puede calcularse como

$$\begin{aligned} f_{\mathbf{y}_e; \mathbf{w}_*}(\mathbf{y}_e) &= f_{\mathbf{y}_e; \mathbf{w}_*} \left(\begin{bmatrix} y(n) \\ b_i(n) \end{bmatrix} \right) \\ &= f_{\mathbf{y}_e; \mathbf{w}_*}(y(n), b_i(n)) \\ &= f_{y(n)|b_i(n); \mathbf{w}_*}(y(n)|b_i(n)) f_{b_i(n)}(b_i(n)) \end{aligned} \quad (4.21)$$

Dado que, según la ecuación (4.1) y teniendo en cuenta que $A_i = 1$, $f_{y(n)|b_i(n); \mathbf{w}_*}$ es una f.d.p. normal de media el símbolo transmitido y varianza la varianza del ruido a la salida del filtro ($f_{y(n)|b_i(n); \mathbf{w}_*} = N(b_i(n), \sigma_f^2)$), se verifica que

$$f_{\mathbf{y}_e; \mathbf{w}_*}(\mathbf{y}_e) = \frac{1}{\pi \sigma_f^2} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} f_b(b_i(n)) \quad (4.22)$$

Por lo tanto

$$f_{\mathbf{Y}_e; \mathbf{w}_*}(\mathbf{Y}_e) = \prod_{n=0}^{K-1} \frac{1}{\pi \sigma_f^2} f_b(b_i(n)) e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \quad (4.23)$$

Sustituyendo (4.23) en (4.13) resulta

$$\begin{aligned} \mathcal{U}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) &= E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} \left[\log \left(\prod_{n=0}^{K-1} \frac{1}{\pi \sigma_f^2} f_b(b_i(n)) e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \right] \\ &= E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i)} \left[\sum_{n=0}^{K-1} \log \left(\frac{1}{\pi \sigma_f^2} f_b(b_i(n)) e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \right] \end{aligned}$$

$$\begin{aligned}
 &= \sum_{n=0}^{M-1} \log \left(\frac{1}{\pi \sigma_f^2} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \\
 &+ E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T, \hat{\mathbf{w}}(i)} \left[\sum_{n=M}^{K-1} \log \left(\frac{1}{\pi \sigma_f^2} f_b(b_i(n)) e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \right]
 \end{aligned} \tag{4.24}$$

que es la expresión de $\mathcal{U}(\mathbf{w}_*, \hat{\mathbf{w}}(i))$ que se va a maximizar de acuerdo con la ecuación (4.19). Así pues, teniendo en cuenta que los símbolos no conocidos se modelan como variables aleatorias uniformes y, por tanto, $f_b(b_i(n))$ es una constante, se obtiene que

$$\begin{aligned}
 \hat{\mathbf{w}}(i+1) &= \arg \max_{\mathbf{w}_*} \{ \mathcal{U}(\mathbf{w}_*, \hat{\mathbf{w}}(i)) \} \\
 &= \arg \max_{\mathbf{w}_*} \left\{ \sum_{n=0}^{M-1} \log \left(\frac{1}{\pi \sigma_f^2} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \right. \\
 &+ \left. E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T, \hat{\mathbf{w}}(i)} \left[\sum_{n=M}^{K-1} \log \left(\frac{1}{\pi \sigma_f^2} f_b(b_i(n)) e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \right) \right] \right\} \\
 &= \arg \min_{\mathbf{w}_*} \left\{ \sum_{n=0}^{M-1} |y(n) - b_i(n)|^2 \right. \\
 &+ \left. E_{\mathbf{Y}_e | \mathbf{y}; \hat{\mathbf{w}}(i)} \left[\sum_{n=M}^{K-1} |y(n) - b_i(n)|^2 \right] \right\} = \arg \min_{\mathbf{w}_*} \{ \mathcal{M}(\mathbf{w}) \}
 \end{aligned} \tag{4.25}$$

Para minimizar la expresión entre corchetes de la ecuación (4.25) (denominada de aquí en adelante $\mathcal{M}(\mathbf{w})$) será preciso calcular su gradiente con respecto a \mathbf{w}^H que, teniendo en cuenta las propiedades del gradiente, resulta ser

$$\begin{aligned}
 \nabla_{\mathbf{w}^H} [\mathcal{M}(\mathbf{w})] &= \sum_{n=0}^{M-1} \nabla_{\mathbf{w}^H} |y(n) - b_i(n)|^2 \\
 &+ E_{\mathbf{Y}_e | \mathbf{y}; \hat{\mathbf{w}}(i)} \left[\sum_{n=M}^{K-1} \nabla_{\mathbf{w}^H} |y(n) - b_i(n)|^2 \right]
 \end{aligned} \tag{4.26}$$

Calculando ahora el gradiente del término $|y(n) - b_i(n)|^2$

$$\nabla_{\mathbf{w}^H} [|y(n) - b_i(n)|^2] = \nabla_{\mathbf{w}^H} \left[(\mathbf{w}^H \mathbf{x}(n) - b_i(n)) (\mathbf{w}^H \mathbf{x}(n) - b_i(n))^H \right]$$

$$\begin{aligned}
 &= \nabla_{\mathbf{w}^H} [\mathbf{w}^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{w}^H \mathbf{x}(n) b_i^*(n) \\
 &\quad - b_i(n) \mathbf{x}^H(n) \mathbf{w} + |b_i(n)|^2] \\
 &= \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{x}(n) b_i^*(n)
 \end{aligned} \tag{4.27}$$

y sustituyendo en la ecuación (4.26)

$$\begin{aligned}
 \nabla_{\mathbf{w}^H} [\mathcal{M}(\mathbf{w})] &= \sum_{n=0}^{M-1} (\mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{x}(n) b_i^*(n)) \\
 &\quad + E_{\mathbf{Y}_e | \mathbf{y}; \hat{\mathbf{w}}^{(i)}} \left[\sum_{n=M}^{K-1} (\mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{x}(n) b_i^*(n)) \right] \\
 &= \sum_{n=0}^{M-1} (\mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \mathbf{x}(n) b_i^*(n)) \\
 &\quad + \sum_{n=M}^{K-1} (\mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - E_{b_i(n) | y(n); \hat{\mathbf{w}}^{(i)}} [\mathbf{x}(n) b_i^*(n)]) \\
 &= \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \sum_{n=0}^{M-1} \mathbf{x}(n) b_i^*(n) \\
 &\quad - \sum_{n=M}^{K-1} E_{b_i(n) | y(n); \hat{\mathbf{w}}^{(i)}} [b_i^*(n)] \mathbf{x}(n)
 \end{aligned} \tag{4.28}$$

donde se ha tenido en cuenta que la expresión $\mathbf{x}(n) b_i^*(n)$ solo depende de la n -ésima columna de la matriz de datos completos, es decir, el vector $\mathbf{y}_e(n)$, por lo que

$$E_{\mathbf{Y}_e | \mathbf{y}; \hat{\mathbf{w}}^{(i)}} [\mathbf{x}(n) b_i^*(n)] = E_{\mathbf{y}_e(n) | y(n); \hat{\mathbf{w}}^{(i)}} [\mathbf{x}(n) b_i^*(n)]. \tag{4.29}$$

Además, puesto que la única parte aleatoria de $\mathbf{y}_e(n) | y(n)$ es $b_i(n)$, se verifica que

$$E_{\mathbf{y}_e(n) | y(n); \hat{\mathbf{w}}^{(i)}} [\cdot] = E_{b_i(n) | y(n); \hat{\mathbf{w}}^{(i)}} [\cdot].$$

Una vez calculado el gradiente se iguala éste a cero

$$\begin{aligned}
 0 &= \sum_{n=0}^{K-1} \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w} - \sum_{n=0}^{M-1} \mathbf{x}(n) b_i^*(n) \\
 &\quad - \sum_{n=M}^{K-1} E_{b_i(n) | y(n); \hat{\mathbf{w}}^{(i)}} [b_i^*(n)] \mathbf{x}(n)
 \end{aligned} \tag{4.30}$$

y se despeja el filtro \mathbf{w}

$$\begin{aligned}
 \sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w} &= \sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n) + \sum_{n=M}^{K-1} E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i)}} [b_i^*(n)] \mathbf{x}(n) \\
 &\Downarrow \\
 \mathbf{w} &= \left(\sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}^H(n) \right)^{-1} \left(\sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n) \right. \\
 &\quad \left. + \sum_{n=M}^{K-1} E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i)}} [b_i^*(n)] \mathbf{x}(n) \right) \quad (4.31)
 \end{aligned}$$

Por la definición de la esperanza estadística se verifica que

$$E_{b|y; \hat{\Theta}_{i,j}} [\phi(b)] = \sum_b f_{b|y; \hat{\Theta}_{i,j}}(b) \phi(b) \quad (4.32)$$

Si se desarrolla la f.d.p. que aparece en la ecuación (4.32) aplicando el teorema de Bayes resulta

$$f_{b|y; \hat{\Theta}_{i,j}}(b) = \frac{f_{y|b; \hat{\Theta}_{i,j}}(y) f_b(b)}{f_{y; \hat{\Theta}_{i,j}}(y)} = \frac{f_b(b) e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}}}{E_b \left[e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}} \right]} = \frac{e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}}}{\sum_b e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}}} \quad (4.33)$$

donde $\sigma_{y|b}^2$ es la varianza de la variable aleatoria $y|b$. Sustituyendo (4.33) en (4.32) se obtiene

$$E_{b|y; \hat{\Theta}_{i,j}} [\phi(b)] = \frac{\sum_b e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}} \phi(b)}{\sum_b e^{-\frac{|y-b|^2}{\sigma_{y|b}^2}}}. \quad (4.34)$$

Siendo $b = b_i(n)$, $y = y(n)$, $\phi(b) = b_i^*(n)$ y $\sigma_{y|b}^2 = \sigma_f^2$ (pues la varianza de una estimación blanda conocido su símbolo transmitido correspondiente es la varianza del ruido filtrado) la esperanza condicionada que aparece en la ecuación (4.31) se puede escribir como

$$E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i)}} [b_i^*(n)] = \frac{\sum_{b_i(n)} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} b_i^*(n)}{\sum_{b_i(n)} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}}} \quad (4.35)$$

Finalmente, sustituyendo (4.35) en (4.31) y teniendo en cuenta que el filtro \mathbf{w} así calculado se corresponde con una nueva actualización de los coeficientes del filtro receptor, $\hat{\mathbf{w}}(i+1)$, resulta

$$\hat{\mathbf{w}}(i+1) = \left(\sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}^H(n) \right)^{-1} \left(\sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n) + \sum_{n=M}^{K-1} \frac{\sum_{b_i(n)} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}} \mathbf{x}(n)b_i^*(n)}{\sum_{b_i(n)} e^{-\frac{|y(n)-b_i(n)|^2}{\sigma_f^2}}} \right). \quad (4.36)$$

En la ecuación (4.36) se observa que para actualizar los coeficientes del filtro en cada iteración del algoritmo EM es preciso conocer la varianza del ruido a la salida del filtro σ_f^2 . Teniendo en cuenta que dicha varianza depende a su vez del filtro \mathbf{w} (ecuación (4.2)), es posible actualizar σ_f^2 de forma muy sencilla cada vez que se actualicen los coeficientes del filtro \mathbf{w} de la siguiente forma

$$\sigma_f^2(i+1) = \hat{\mathbf{w}}^H(i)\hat{\mathbf{w}}(i)\sigma_g^2 \quad (4.37)$$

El valor de la varianza $\sigma_f^2(i+1)$ se utilizará en el cálculo de $\hat{\mathbf{w}}(i+1)$. El algoritmo resultante se resume en el cuadro 4.1.

SAGE (*Space Alternating Generalized EM*)

El SAGE es una generalización del algoritmo EM que se utiliza cuando no es sencillo (o conveniente) aplicar EM sobre todos los parámetros a estimar simultáneamente. En esos casos, la idea subyacente en esta nueva aproximación es aplicar de forma alternada el algoritmo EM sobre conjuntos disjuntos de parámetros. Esto supone un ligero incremento en la complejidad computacional con respecto al algoritmo EM, pero también da lugar a una mayor velocidad de convergencia [3].

En el problema de optimización planteado por la ecuación (4.9) hay dos conjuntos de parámetros que es necesario estimar: por un lado el de los coeficientes del filtro y, por otro, un conjunto cuyo único elemento es la varianza del ruido a la

Inicialización del algoritmo:

$$\tilde{\mathbf{R}}_x(K) = \sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}(n)^H$$

$$\tilde{\mathbf{p}}(M) = \sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n)$$

$$\hat{\mathbf{w}}(-1) = \mathbf{w}_{LS} = \left(\frac{1}{K} \tilde{\mathbf{R}}_x(K) \right)^{-1} \left(\frac{1}{M} \tilde{\mathbf{p}}(M) \right)$$

$$\hat{\sigma}_f^2(0) = \sigma_g^2$$

$$i = 0$$

Paso iterativo (hasta la convergencia)

$$\hat{\mathbf{w}}(i) = \tilde{\mathbf{R}}_x(K) \left(\tilde{\mathbf{p}}(M) + \sum_{n=M}^{K-1} \frac{\sum_{b_i(n)} e^{-\frac{|\hat{\mathbf{w}}^H(i-1)\mathbf{X}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}} \mathbf{x}(n)b_i^*(n)}{\sum_{b_i(n)} e^{-\frac{|\hat{\mathbf{w}}^H(i-1)\mathbf{X}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}}} \right)$$

$$\hat{\sigma}_f^2(i+1) = \hat{\mathbf{w}}^H(i)\hat{\mathbf{w}}(i)\sigma_g^2$$

$$i = i + 1$$

Cuadro 4.1: Pseudocódigo del algoritmo EM.

salida del filtro. La estimación de los coeficientes del filtro por medio del algoritmo EM ha sido descrita en detalle a lo largo de la sección 4.2.2. Por lo tanto, solo falta estimar σ_f^2 utilizando el algoritmo EM de forma análoga a como se hizo para estimar el filtro \mathbf{w} . Se trata de maximizar la verosimilitud de una trama de estimaciones eligiendo un valor apropiado de σ_f^2 . Para ello se procederá siguiendo los pasos dados para obtener \mathbf{w} con la salvedad de que ahora σ_f^2 no es una constante sino el parámetro a estimar. Así pues, fijando \mathbf{w} al valor de su estimación i -ésima, $\mathbf{w} = \hat{\mathbf{w}}(i)$, se tiene

$$\mathcal{U}(\sigma_{f_*}^2, \hat{\sigma}_f^2(i)) = E_{\mathbf{Y}_e | \mathbf{y}, \mathbf{b}_T; \hat{\mathbf{w}}(i), \hat{\sigma}_f^2(i)} \left[\log \left(f_{\mathbf{Y}_e; \hat{\mathbf{w}}(i), \sigma_{f_*}^2}(\mathbf{Y}_e) \right) \right] \quad (4.38)$$

A partir del resultado obtenido en (4.24) y teniendo en cuenta la nueva definición de la función \mathcal{U} dada por la ecuación 4.38, en la que depende de la varianza que se desea estimar y de la utilizada en la iteración anterior, se puede continuar el desarrollo de $\mathcal{U}(\sigma_{f_*}^2, \hat{\sigma}_f^2(i))$ del siguiente modo

$$\begin{aligned} \mathcal{U}(\sigma_{f_*}^2, \hat{\sigma}_f^2(i)) &= \sum_{n=0}^{M-1} \left(\log(f_b(b_i(n))) - \log(\pi \sigma_{f_*}^2) - \frac{|y(n) - b_i(n)|^2}{\sigma_{f_*}^2} \right) \\ &+ E_{b_i(n) | y(n); \hat{\mathbf{w}}(i), \hat{\sigma}_f^2(i)} \left[\sum_{n=M}^{K-1} \left(\log(f_b(b_i(n))) - \log(\pi \sigma_{f_*}^2) - \frac{|y(n) - b_i(n)|^2}{\sigma_{f_*}^2} \right) \right] \\ &= K \log(f_b(b_i(n))) - K \log(\pi \sigma_{f_*}^2) - \sum_{n=0}^{M-1} \left(\frac{|y(n) - b_i(n)|^2}{\sigma_{f_*}^2} \right) \\ &- \sum_{n=M}^{K-1} E_{b_i(n) | y(n); \hat{\mathbf{w}}(i), \hat{\sigma}_f^2(i)} \left[\frac{|y(n) - b_i(n)|^2}{\sigma_{f_*}^2} \right] \end{aligned} \quad (4.39)$$

donde $y(n) = \hat{\mathbf{w}}^H(i) \mathbf{x}(n)$.

Se maximiza a continuación $\mathcal{U}(\sigma_f^2, \hat{\sigma}_f^2(i))$ respecto a σ_f^2 calculando para ello su derivada parcial

$$\begin{aligned} \frac{\partial}{\partial \sigma_f^2} \mathcal{U}(\sigma_f^2, \hat{\sigma}_f^2(i)) &= -K \frac{1}{\sigma_f^2} + \sum_{n=0}^{M-1} \left(\frac{|y(n) - b_i(n)|^2}{(\sigma_f^2)^2} \right) \\ &+ \sum_{n=M}^{K-1} E_{b_i(n) | y(n); \hat{\mathbf{w}}(i), \hat{\sigma}_f^2(i)} \left[\frac{|y(n) - b_i(n)|^2}{(\sigma_f^2)^2} \right] \end{aligned}$$

$$\begin{aligned}
 &= \frac{1}{\sigma_f^2} \left[-K + \sum_{n=0}^{M-1} \left(\frac{|y(n) - b_i(n)|^2}{\sigma_f^2} \right) \right. \\
 &+ \left. \sum_{n=M}^{K-1} E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i), \hat{\sigma}_f^2(i)} \left[\frac{|y(n) - b_i(n)|^2}{\sigma_f^2} \right]} \right] \quad (4.40)
 \end{aligned}$$

Igualando el resultado final de la ecuación (4.40) a cero resulta

$$\begin{aligned}
 K &= \frac{1}{\sigma_f^2} \left[\sum_{n=0}^{M-1} |y(n) - b_i(n)|^2 + \sum_{n=M}^{K-1} E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i), \hat{\sigma}_f^2(i)} [|y(n) - b_i(n)|^2] \right] \\
 \Downarrow \\
 \sigma_f^2 &= \frac{1}{K} \left[\sum_{n=0}^{M-1} |y(n) - b_i(n)|^2 + \sum_{n=M}^{K-1} E_{b_i(n)|y(n); \hat{\mathbf{w}}^{(i), \hat{\sigma}_f^2(i)} [|y(n) - b_i(n)|^2] \right] \quad (4.41)
 \end{aligned}$$

Y teniendo en cuenta la ecuación (4.34)

$$\hat{\sigma}_f^2(i+1) = \frac{1}{K} \left[\sum_{n=0}^{M-1} |y(n) - b_i(n)|^2 + \sum_{n=M}^{K-1} \frac{\sum_{b_i(n)} e^{-\frac{|y(n) - b_i(n)|^2}{\hat{\sigma}_f^2(i)}} |y(n) - b_i(n)|^2}{\sum_{b_i(n)} e^{-\frac{|y(n) - b_i(n)|^2}{\hat{\sigma}_f^2(i)}}} \right] \quad (4.42)$$

La ecuación (4.42) proporciona una estimación de la varianza σ_f^2 mucho más ajustada que la dada por la ecuación (4.37) y será, por lo tanto, la que se utilizará en la práctica.

El algoritmo SAGE consiste esencialmente en estimar alternativamente los coeficientes del filtro y de la varianza utilizando el algoritmo EM. Su pseudocódigo será, por lo tanto, idéntico al del algoritmo EM (cuadro 4.1) salvo en la estimación de la varianza, donde se empleará la expresión dada por la ecuación (4.42). Se muestra en el cuadro 4.2.

4.3. Conclusiones

Cada una de las aproximaciones clásicas descritas en el capítulo anterior presenta algún inconveniente a la hora de llevarlas a la práctica. Por un lado, los

Inicialización del algoritmo:

$$\tilde{\mathbf{R}}_x(K) = \sum_{n=0}^{K-1} \mathbf{x}(n)\mathbf{x}(n)^H$$

$$\tilde{\mathbf{p}}(M) = \sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n)$$

$$\hat{\mathbf{w}}(-1) = \mathbf{w}_{LS} = \left(\frac{1}{K} \tilde{\mathbf{R}}_x(K) \right)^{-1} \left(\frac{1}{M} \tilde{\mathbf{p}}(M) \right)$$

$$\hat{\sigma}_f^2(0) = \sigma_g^2$$

$$i = 0$$

Paso iterativo (hasta la convergencia)

$$\hat{\mathbf{w}}(i) = \tilde{\mathbf{R}}_x(K) \left(\tilde{\mathbf{p}}(M) + \sum_{n=M}^{K-1} \frac{e^{-\frac{|\hat{\mathbf{w}}^H(i-1)\mathbf{x}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}}}{\sum_{b_i(n)} e^{-\frac{|\hat{\mathbf{w}}^H(i-1)\mathbf{x}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}}} \mathbf{x}(n)b_i^*(n) \right)$$

$$\hat{\sigma}_f^2(i+1) = \frac{1}{K} \left[\sum_{n=0}^{M-1} |\hat{\mathbf{w}}^H(i)\mathbf{x}(n) - b_i(n)|^2 + \right.$$

$$\left. + \sum_{n=M}^{K-1} \frac{e^{-\frac{|\hat{\mathbf{w}}^H(i)\mathbf{x}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}}}{\sum_{b_i(n)} e^{-\frac{|\hat{\mathbf{w}}^H(i)\mathbf{x}(n)-b_i(n)|^2}{\hat{\sigma}_f^2(i)}}} |\hat{\mathbf{w}}^H(i)\mathbf{x}(n)-b_i(n)|^2 \right]$$

$$i = i + 1$$

Cuadro 4.2: Pseudocódigo del algoritmo SAGE.

detectores MMSE y decorrelador parten de un conocimiento completo de la matriz de códigos distorsionada, y en problemas prácticos de cancelación de MAI, esta información no está disponible para el receptor. Por otra parte, los algoritmos basados en mínimos cuadrados (LS, RLS y RLS DD) precisan de una secuencia de entrenamiento de gran longitud para obtener unos resultados satisfactorios (como se ilustrará en el próximo capítulo). Además, el algoritmo RLS DD tiene un problema adicional, que es el fenómeno conocido como propagación de errores y que ya ha sido explicado en detalle en la sección 3.1.4.

Para paliar las mencionadas deficiencias que afectan a métodos de supresión de interferencias clásicos como LS o RLS, en el presente capítulo se ha expuesto una aproximación basada en el principio de máxima verosimilitud (ML) que mejora en gran medida las técnicas que se han venido empleando hasta la fecha. La idea básica que da origen a esta nueva aproximación es la siguiente: partiendo de la hipótesis de que existe un filtro óptimo capaz de suprimir por completo la MAI e igualar el canal, al aplicar dicho filtro sobre el vector de observaciones se obtiene una estimación blanda que está formada por el símbolo transmitido (posiblemente con un factor de escala complejo) y un término de ruido, es decir

$$y(n) = A_i b_i(n) + g_f(n) \quad (4.43)$$

para el caso del usuario i . Es posible calcular la f.d.p de las estimaciones blandas así obtenidas, que claramente depende de los coeficientes del filtro. Se puede considerar esta f.d.p. como una densidad objetivo y aplicar sobre ella el criterio ML para *estimar* el filtro \mathbf{w} . La resolución del problema así planteado garantiza que las estimaciones que se obtienen con el filtro tienen una distribución de probabilidad cercana a la de $A_i b_i(n) + g_f(n)$ [1].

De acuerdo al modelo MC-CDMA propuesto, todos los usuarios del sistema utilizan la misma modulación (y por tanto $b_i(n)$ son variables aleatorias i.i.d. $\forall i, n$), por lo que la f.d.p. de sus estimaciones blandas solo se va a diferenciar en un factor complejo desconocido *a priori*. Se hace necesario algún mecanismo que permita

extraer correctamente la señal del usuario deseado, estadísticamente indistinguible de las señales de los usuarios interferentes. En el presente trabajo se ha utilizado una aproximación semiciega para resolver el *problema de captura*, que consiste en el uso de un número reducido de símbolos piloto. De esta forma se puede asegurar la unicidad de la solución del problema de estimación planteado evitando la captura de interferencias. La aproximación basada en estimación ML propuesta no da lugar a soluciones analíticas. Un algoritmo numérico especialmente adecuado para resolver este tipo de problemas es el método de Esperanza-Maximizar (EM), que permite obtener una expresión para actualizar iterativamente los coeficientes del receptor. En las ecuaciones recursivas del algoritmo EM interviene la varianza del ruido a la salida del filtro \mathbf{w} . Ésta puede ser actualizada iterativamente mediante un procedimiento *ad hoc* muy simple. Sin embargo, también es posible emplear el principio ML y la metodología SAGE [3] para llevar a cabo la actualización de este parámetro. SAGE es una generalización del procedimiento EM con mejores propiedades de convergencia y un amplio rango de aplicaciones.

La técnica de supresión de interferencias basada en el principio ML propuesta en el presente trabajo tiene unos requisitos computacionales considerablemente superiores a los de las técnicas clásicas vistas en el capítulo 3. Esto es debido a que no es posible obtener una solución analítica al problema de optimización que plantea, sino que es preciso realizar sucesivas actualizaciones de los coeficientes del filtro hasta que estos converjan a una solución definitiva. Por tanto, no es posible acotar el número de operaciones que son necesarias para calcular el filtro en base a los parámetros del sistema de comunicaciones, sino que existe una cierta incertidumbre asociada al tiempo de cálculo que se va a necesitar para alcanzar una solución definitiva.

A cambio de esta relativamente alta complejidad computacional, el algoritmo EM solventa la mayor parte de los problemas que afectan a las aproximaciones clásicas vistas en el capítulo 3. Primeramente, EM no necesita conocer el canal de

comunicaciones, lo cual es una ventaja importante, pues normalmente es desconocido o difícil de estimar. Por otro lado, a diferencia de los algoritmos LS, RLS y RLS DD, EM utiliza la información estadística disponible *a priori* de los símbolos transmitidos y del ruido aditivo. Por ello puede alcanzar un mejor rendimiento que el algoritmo LS y sus variantes utilizando las mismas observaciones y los mismos símbolos piloto. Por último, el algoritmo EM, al no realimentar decisiones, no sufre de propagación de errores como RLS dirigido por decisión.

Por lo tanto, esta nueva aproximación a la supresión de interferencias basada en el principio ML ofrece *a priori* varias ventajas frente a la técnicas clásicas estudiadas en el capítulo 3. En el próximo capítulo se presentan resultados de simulaciones por ordenador que ilustran la validez del método propuesto.

Capítulo 5

Simulaciones

Para comparar el rendimiento de los diferentes algoritmos se han realizado diversas simulaciones por ordenador. Tienen por objeto determinar la probabilidad de error y el error cuadrático medio (MSE o *Mean Square Error* en inglés) de las diferentes técnicas de supresión de interferencias descritas en el presente trabajo y estudiar como varían dichas magnitudes al modificar algunos de los parámetros típicos que afectan al rendimiento de las mencionadas técnicas. Concretamente se va a mostrar como afecta a la probabilidad de error de símbolo (P_e) y al MSE la variación de la relación señal a ruido (SNR: *Signal to Noise Ratio*), de la longitud de la secuencia de entrenamiento, M , y del número de usuarios N .

5.1. Parámetros de la simulación

En todas las simulaciones se han asumido diversas consideraciones generales acerca del sistema de comunicaciones. Todos los usuarios del sistema emplean una modulación QPSK (*Quadrature Phase Shift Keying*). Por tanto, el alfabeto de los símbolos es $\mathcal{A} = \{-1, 1, -j, j\}$, siendo j la unidad imaginaria. La longitud de código es ocho ($L = 8$) y se ha considerado una trama de tamaño $K = 400$ para ejecutar los diferentes algoritmos que darán lugar a los distintos filtros (uno por cada técnica de

supresión de interferencias). Una vez calculados todos los filtros, se utilizarán éstos sobre una secuencia de observaciones de tamaño 1.000.000 para estimar el valor de la probabilidad de error alcanzado por cada técnica. Se ha procedido de esta manera porque, por un lado, es preciso emplear una longitud de trama muy grande para calcular probabilidades de error puesto que para SNR's grandes la tasa de fallos puede llegar a ser muy reducida (del orden de 10^{-5} o incluso más baja) y, por otro, es inviable simular la detección de 1.000.000 de símbolos repartidos en 2500 tramas de 400 símbolos, pues el tiempo necesario para completar la simulación sería extraordinariamente elevado. Así pues, se han estimado los coeficientes de cada uno de los filtros sobre una trama del tamaño usual, y se han empleado dichos filtros sobre tramas de una longitud muy superior para obtener unas estimaciones de las probabilidades de error más precisas.

Para que los resultados plasmados en las gráficas tengan la mayor fiabilidad posible, se han obtenido promediando los resultados obtenidos para un conjunto de 20 canales generados aleatoria e independientemente. Cada canal queda completamente determinado por su matriz \mathbf{H} . Las columnas de la matriz \mathbf{H} se corresponden con los canales de los diferentes usuarios y se generan utilizando un proceso autorregresivo de orden 2 con la finalidad de que exista una cierta correlación entre muestras contiguas de la transformada de Fourier de un determinado canal. Así pues

$$H_i(k) = a_1 H_i(k-1) + a_2 H_i(k-2) + u(k) \quad (5.1)$$

donde $u(k) = N(0, \sigma_u^2)$. A la desviación típica σ_u^2 se le suelen asignar valores pequeños (en las simulaciones presentadas $\sigma_u^2 = 0,01$). Permite introducir aleatoriedad en la construcción de los diferentes canales, sin que por ello deje de existir cierta correlación entre muestras contiguas de la DFT de cada uno de ellos. Los coeficientes a_1 y a_2 se pueden elegir arbitrariamente según la correlación que se desee entre subportadoras (para los resultados mostrados en las gráficas $a_1 = a_2 = \frac{1-\sigma_u^2}{2} = 0,4950$). Para que el proceso autorregresivo considerado sea estacionario los valores elegidos para representar las muestras del i -ésimo canal,

$H_i(0), \dots, H_i(L-1)$, son el resultado de iterar sobre la ecuación (5.1) 200 veces antes de fijar el canal, es decir, se inicializa el proceso AR con $H_i(-200+L-2)$ y $H_i(-200+L-1)$, que siguen una distribución normal de media 0 y desviación típica $\sqrt{3}$. Este último valor se ha elegido al azar y no tiene ninguna repercusión en los resultados que se van a mostrar pues, como ya se ha comentado con anterioridad, entre los objetivos de las distintas técnicas de supresión de interferencias descritas se encuentra el de llevar a cabo la igualación del canal. Únicamente sirve para garantizar la suficiente diversidad en las condiciones iniciales del proceso autorregresivo.

Como se puede ver en la ecuación (2.6), para calcular los vectores de observaciones $\mathbf{x}(n)$ se necesita la matriz de códigos distorsionada, \mathbf{D} . Esta matriz se calcula multiplicando elemento a elemento la matriz \mathbf{H} por una matriz de códigos \mathbf{C} . Los elementos de la matriz \mathbf{C} se han generado como muestras de una variable aleatoria uniforme binaria ($C_{ij} \sim U(\pm 1)$). Es conveniente que la correlación cruzada entre las columnas de la matriz (los códigos de los diferentes usuarios) sea lo más baja posible.

Se ha generado una matriz de códigos de dimensiones 8×5

$$\mathbf{C}_{SNR,M} = \begin{bmatrix} 1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 & -1 \end{bmatrix}$$

y otra de tamaño 8×11

$$\mathbf{C}_N = \begin{bmatrix} 1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \\ 1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 \\ -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \end{bmatrix}.$$

Ambas matrices poseen una baja correlación entre sus columnas según las comprobaciones hechas y, por tanto, son aptas para ser utilizadas como matrices de códigos. La matriz denotada por $\mathbf{C}_{SNR,M}$ se empleará para medir la probabilidad de error y el MSE en función de la SNR y de la secuencia de entrenamiento M (en ambos casos se supone que el sistema está siendo utilizado por cinco usuarios), mientras que \mathbf{C}_N se utilizará para estudiar como varían esas mismas magnitudes en función del número de usuarios. La segunda dimensión de esta última matriz (once) es el número máximo de usuarios para el que se va a estudiar el rendimiento del sistema, de tal manera que para probar el sistema con un número inferior de usuarios solo es necesario eliminar por la derecha algunas de las columnas de \mathbf{C}_N .

Así pues, en las simulaciones no se han generado matrices de códigos al azar, sino que se ha empleado o bien $\mathbf{C}_{SNR,M}$, o bien \mathbf{C}_N .

Para una velocidad de transmisión fija, la tasa de errores en un sistema de comunicaciones digitales depende de la SNR, que definimos como

$$SNR = 10 \log_{10} \frac{P_s}{P_g} \text{ (dB)} \quad (5.2)$$

siendo P_s la potencia de señal y $P_g = \sigma_g^2$ la potencia del ruido.

Teniendo en cuenta que cada usuario del sistema tiene una SNR distinta puesto que cada uno puede transmitir con una potencia diferente, la SNR para el i -ésimo usuario es:

$$SNR_i = 10 \log_{10} \frac{P_i}{P_g} \text{ (dB)}. \quad (5.3)$$

Es sencillo calcular la varianza del AWGN a partir de la SNR de operación del sistema. Atendiendo a la ecuación (2.6) de la página 22, la potencia de la señal se calcula como

$$\begin{aligned} P_s &= E [\mathbf{b}^H(n) \mathbf{D}^H \mathbf{D} \mathbf{b}(n)] = E \left[\left(\sum_{k=1}^N \mathbf{d}_k^H b_k^*(n) \right) \left(\sum_{j=1}^N \mathbf{d}_j b_j(n) \right) \right] \\ &= E \left[\sum_{k=1}^N \sum_{j=1}^N \mathbf{d}_k^H \mathbf{d}_j b_k^*(n) b_j(n) \right] = \sum_{k=1}^N \sum_{j=1}^N \mathbf{d}_k^H \mathbf{d}_j E [b_k^*(n) b_j(n)] \end{aligned}$$

$$= \sum_{k=1, k \neq j}^N \sum_{j=1}^N \mathbf{d}_k^H \mathbf{d}_j E [b_k^*(n)] E [b_j(n)] + \sum_{j=1}^N \mathbf{d}_j^H \mathbf{d}_j E [|b_j(n)|^2] \quad (5.4)$$

Suponiendo que la esperanza de los símbolos que transmiten los usuarios es nula ($E [b_i(n)] = 0 \forall i$), circunstancia que se da en la mayor parte de los sistemas de comunicaciones,

$$P_s = \sum_{j=1}^N \mathbf{d}_j^H \mathbf{d}_j E [|b_j(n)|^2] = \sigma_b^2 \sum_{j=1}^N \mathbf{d}_j^H \mathbf{d}_j \quad (5.5)$$

donde $\sigma_b^2 = E [|b_j(n)|^2]$ es la potencia de los símbolos.

La potencia P_s es la potencia de la señal que le llega al receptor y es la suma de las potencias individuales recibidas para cada uno de los usuarios del sistema, siendo

$$P_i = \sigma_b^2 \mathbf{d}_i^H \mathbf{d}_i \quad (5.6)$$

la potencia de señal recibida para el i -ésimo usuario del sistema.

Sustituyendo (5.6) en (5.3) y utilizando la definición de P_g

$$\begin{aligned} SNR_i &= 10 \log_{10} \frac{\sigma_b^2 \mathbf{d}_i^H \mathbf{d}_i}{\sigma_g^2} \\ &\Downarrow \\ \sigma_g^2 &= 10^{-\frac{SNR_i}{10}} \sigma_b^2 \mathbf{d}_i^H \mathbf{d}_i \end{aligned} \quad (5.7)$$

Por lo tanto, la ecuación (5.7) permite obtener la varianza del ruido σ_g^2 asociada a una determinada SNR para un determinado canal de comunicaciones.

5.2. Breve descripción de los algoritmos simulados

En esta sección se introducirán brevemente cada uno de los algoritmos utilizados en las simulaciones de la sección 5.3. En todos ellos el rendimiento está limitado claramente por el valor de la SNR, por lo que las descripciones se centrarán en otras particularidades de cada algoritmo individual.

- **SAGE**: Por su mayor velocidad de convergencia y por su mejor rendimiento se ha simulado el algoritmo EM generalizado (SAGE), cuyo pseudocódigo puede consultarse en el cuadro 4.2. Se considera que el algoritmo converge cuando $|\hat{\mathbf{w}}(i) - \hat{\mathbf{w}}(i-1)|^2 < 10^{-8}$, permitiéndose un máximo de 100 iteraciones para alcanzar la convergencia.
- **LS(M)**: Es el algoritmo de mínimos cuadrados (*Least Squares*) utilizando una secuencia de entrenamiento de M símbolos ($M < K$). Cuanto mayor sea el valor de M más precisas serán las estimaciones que se obtengan con el filtro construido utilizando LS. La expresión empleada para calcular el filtro LS es la que se muestra en la ecuación (3.20), teniendo en cuenta que, puesto que solo son conocidos M símbolos, $\hat{\mathbf{p}}$ debe calcularse de forma aproximada como $\hat{\mathbf{p}} = \frac{1}{M} \sum_{n=0}^{M-1} \mathbf{x}(n)b_i^*(n)$, y que $\hat{\mathbf{R}}_x$ se calculará como $\hat{\mathbf{R}}_x = \frac{1}{M} \sum_{n=0}^{M-1} \mathbf{x}(n)\mathbf{x}^H(n)$ para obtener unos mejores resultados.
- **DEC**: El detector decorrelador parte de un conocimiento completo de la matriz de códigos distorsionada, por lo que su rendimiento no depende en modo alguno de M . El filtro decorrelador se obtiene como la i -ésima fila de la pseudoinversa por la izquierda de la matriz \mathbf{D} (página 28).
- **MMSE**: Obtiene los coeficientes del filtro minimizando el error cuadrático medio esperado (*Minimum Mean Square Error*) y tampoco hace uso de la secuencia de entrenamiento. La expresión obtenida para el filtro utilizando el criterio MMSE es la dada por la ecuación (3.15).¹
- **RLS**: Es la implementación adaptativa del algoritmo LS (*Recursive Least Squares*) y proporciona unos resultados muy similares a los de éste, siendo la longitud de la secuencia de entrenamiento, M , un factor crucial. Su pseudocódigo se muestra en el cuadro 3.1.

¹Nótese que tanto DEC como MMSE proporcionan una referencia teórica de rendimiento, pero no son receptores prácticos.

- **RLS DD:** El algoritmo RLS dirigido por decisión es una variante del RLS, en la que se lleva a cabo una realimentación de de decisiones (estimaciones duras). En algunos situaciones puede proporcionar mejores resultados que el RLS, mientras que en otras puede ocurrir lo contrario al tratarse de un algoritmo sujeto al fenómeno de propagación de errores. El pseudocódigo correspondiente es el del cuadro 3.2.
- **LS(K):** Consiste en utilizar el algoritmo LS para calcular los coeficientes del filtro suponiendo que son conocidos todos los símbolos de la trama. Puesto que en la práctica es inviable, los resultados que obtenga servirán únicamente de referencia para comparar con otros algoritmos. La expresión que se utiliza para obtener el filtro \mathbf{w} es la dada por la ecuación (3.20).

5.3. Resultados numéricos

5.3.1. Probabilidad de error y MSE en función de la SNR

La relación señal a ruido o SNR es el cociente entre la potencia de la señal transmitida y la potencia del ruido AWGN. En esta sección se va a estudiar de que manera la variación de esta magnitud repercute en la probabilidad de error de símbolo y en el error cuadrático medio para diversas valores fijos de la longitud de la secuencia de entrenamiento, M .

Una SNR alta quiere decir que la potencia del ruido es pequeña en comparación con la potencia de la señal deseada. Por tanto, cuanto mayor sea la SNR, menores serán los efectos nocivos de la distorsión que el ruido provoca en la señal transmitida y, en consecuencia, menor será la probabilidad de error de símbolo. Esto se aprecia claramente en todas las gráficas, pues en todas ellas es posible observar que un incremento en la SNR siempre lleva asociado un decremento en la probabilidad de error de símbolo.

En cada figura se muestra una gráfica con la evolución de la probabilidad de

error y otra que refleja el MSE asociado a las estimaciones blandas, es decir

$$MSE = \frac{1}{K} \sum_{n=0}^{K-1} |y(n) - b_i(n)|^2 \quad (5.8)$$

siendo $y(n)$ las estimaciones blandas obtenidas para los símbolos del usuario i .

Las figuras 5.1, 5.2, 5.3 y 5.4 ilustran el comportamiento de los distintos algoritmos cuando la longitud de la secuencia de entrenamiento es $M = 1$, $M = 6$, $M = 11$ y $M = 16$ respectivamente. Para los algoritmos de mínimos cuadrados una secuencia de entrenamiento de longitud $M = 1$ o $M = 6$ es insuficiente para ajustar correctamente los coeficientes del filtro. Así lo demuestran las curvas de los algoritmos LS(M), RLS y RLS dirigido por decisión (RLS DD) en las figuras 5.1 y 5.2. En las figuras 5.3 y 5.4 ($M = 11$ y $M = 16$ respectivamente) se aprecia una clara mejora en el MSE y la probabilidad de error del algoritmo LS. Así, para $M = 11$ y una SNR aproximada de 17, alcanza probabilidades de error del orden de 10^{-5} . Una probabilidad de error de la misma magnitud la alcanza el algoritmo SAGE para $SNR = 12$. Por su parte, los algoritmos RLS y RLS DD obtienen un rendimiento muy inferior al de LS. Es debido a que el algoritmo RLS es adaptativo y no siempre es capaz de converger a una solución similar a la obtenida por LS con tan solo 11 iteraciones para el caso $M = 11$, ó 16 iteraciones para el caso $M = 16$. El algoritmo RLS DD utiliza en un primer momento el filtro que se obtendría con el algoritmo RLS para hacer estimaciones duras de los símbolos desconocidos, por lo que su rendimiento va a estar fuertemente condicionado por el funcionamiento de éste último. Esta incertidumbre asociada a la convergencia del algoritmo RLS (y, por tanto, del RLS DD) se puede observar en los cuadros 5.5 y 5.7, que muestran que la desviación típica de la probabilidad de error para estos dos algoritmos es anormalmente elevada en comparación con los restantes. El comportamiento del algoritmo SAGE en relación con la longitud de la secuencia de entrenamiento es sustancialmente distinto al mostrado por el algoritmo LS. Como se explicó en la sección 4.2, sólo se necesita la secuencia de entrenamiento para que los coeficientes del filtro converjan a aquella solución del problema que corresponde a la extracción de

los símbolos del usuario deseado, evitando extraer los de los usuarios interferentes. Así pues, el algoritmo SAGE comenzará a proporcionar resultados satisfactorios tan pronto como sea capaz de distinguir al usuario deseado de los restantes en virtud de la secuencia de entrenamiento. Una longitud $M = 1$ es insuficiente pues si el número de usuarios del sistema es $N = 5$, entonces se puede asegurar que al menos dos de ellos han empleado las misma secuencia de entrenamiento y, por tanto, no es posible diferenciarlos. Por consiguiente, en la figura 5.1, en la que $M = 1$, el algoritmo SAGE no funciona y así aparece reflejado en la curva que representa su probabilidad de error. Para $M \geq 6$ el algoritmo SAGE ya es capaz de extraer los símbolos del usuario deseado y su curva de probabilidad de error converge a la de los algoritmos teóricos. Así, en la figura 5.2 ($M = 6$) su curva de probabilidad de error se ajusta a la de MMSE y LS(K) en todo el rango de SNR's de interés y de forma prácticamente exacta para $SNR > 9 \text{ dB}$. Lo mismo ocurre para el caso de $M = 11$ y $M = 16$ en las figuras 5.3 y 5.4 respectivamente.

Indudablemente los resultados que se obtienen con cada algoritmo dependen del canal y por ello se han incluido los cuadros 5.1, 5.3, 5.5 y 5.7 para dar una idea de cuanto pueden llegar a variar las probabilidades de error en función del canal de comunicaciones empleado. Los cuadros 5.2, 5.4, 5.6 y 5.8 cumplen una función análoga al indicar la desviación típica del error cuadrático medio.

5.3.2. Probabilidad de error y MSE en función de M

La longitud de la secuencia de entrenamiento, M , es uno de los factores que más directamente afectan al rendimiento de los filtros que se obtienen con los algoritmos examinados. Sin embargo, la importancia de este parámetro depende del algoritmo, como se vio en la sección anterior. En esta nueva sección se mostrará como varía la probabilidad de error y el error cuadrático medio en función del parámetro M , para distintos valores de la SNR.

Cuando la SNR es muy baja, ninguno de los algoritmos proporciona unos re-

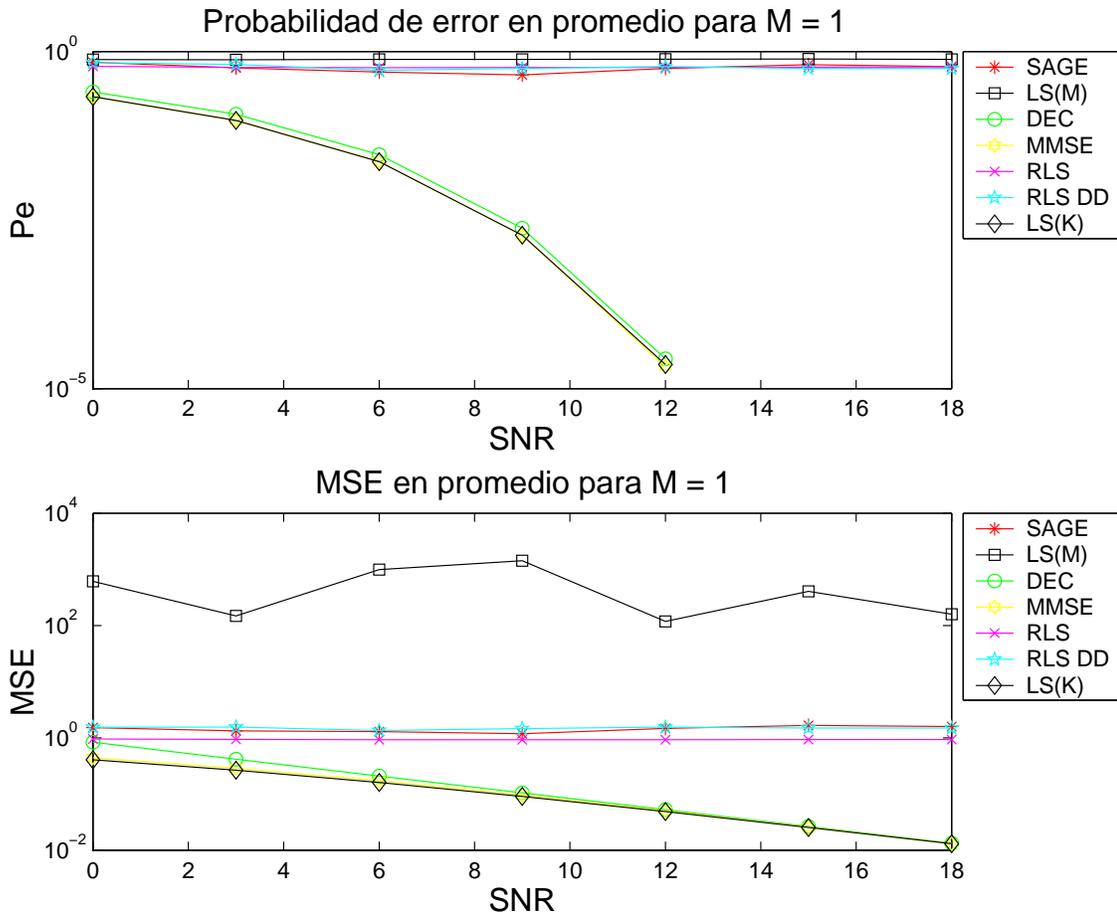


Figura 5.1: Probabilidad de error y MSE promediados sobre 20 canales para $M = 1$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.1266	0.2863	0.3564	0.3762	0.3331	0.2748	0.3079
LS(M)	0.1120	0.0910	0.1506	0.1415	0.1313	0.0870	0.1135
DEC	0.0324	0.0259	0.0110	0.0015	0.0000	0.0000	0.0000
MMSE	0.0356	0.0275	0.0107	0.0013	0.0000	0.0000	0.0000
RLS	0.1446	0.1733	0.1924	0.1993	0.1998	0.1991	0.1980
RLS DD	0.0970	0.2002	0.3278	0.3308	0.3066	0.3330	0.3333
LS(K)	0.0338	0.0263	0.0103	0.0012	0.0000	0.0000	0.0000

Cuadro 5.1: Desviación típica de la probabilidad de error para $M = 1$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.4157	0.6760	0.8557	0.9242	0.8484	0.7119	0.8080
LS(M)	1044.4158	171.3498	3544.4406	5255.6156	240.0060	941.1783	179.3924
DEC	0.1232	0.0617	0.0309	0.0155	0.0078	0.0039	0.0020
MMSE	0.0419	0.0371	0.0261	0.0156	0.0084	0.0042	0.0021
RLS	0.2356	0.2698	0.2902	0.3028	0.3109	0.3160	0.3190
RLS DD	0.4079	0.5935	0.8259	0.8246	0.7900	0.8626	0.8737
LS(K)	0.0396	0.0330	0.0229	0.0139	0.0077	0.0040	0.0020

Cuadro 5.2: Desviación típica del MSE para $M = 1$

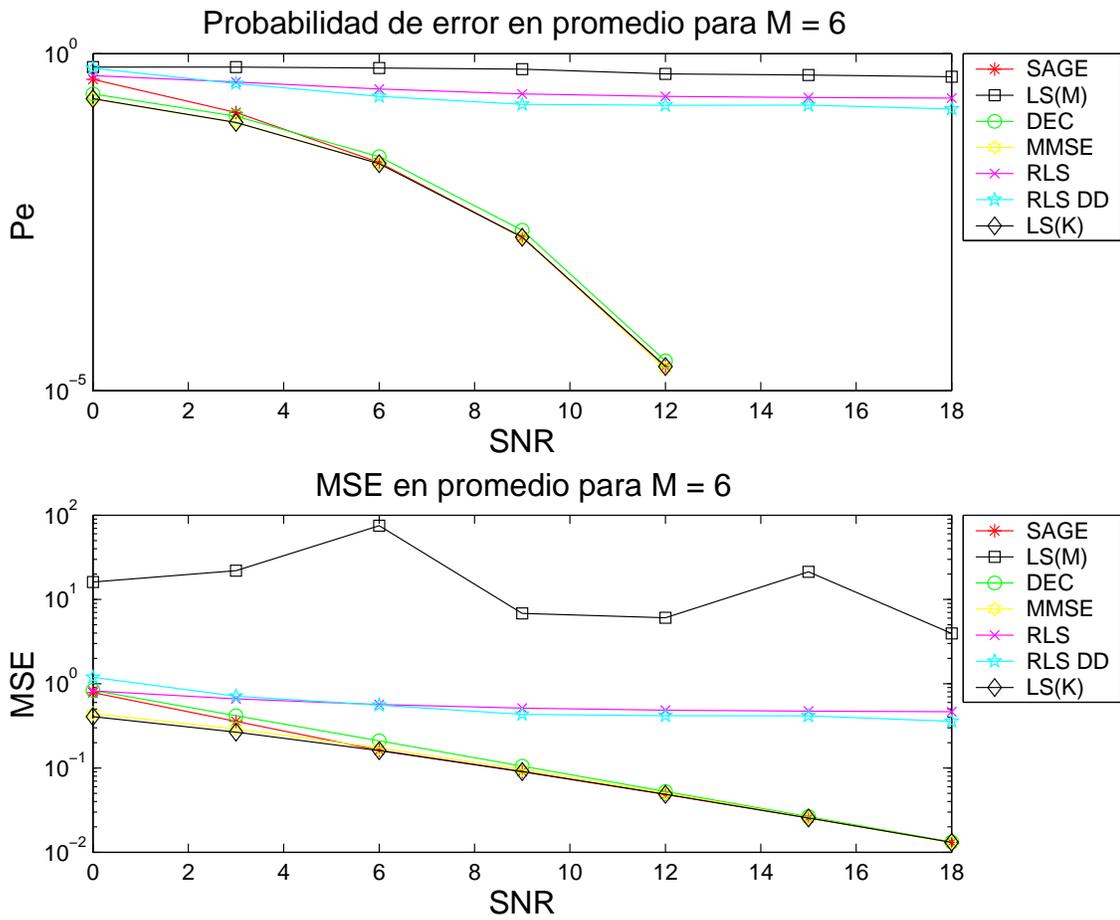


Figura 5.2: Probabilidad de error y MSE promediados sobre 20 canales para $M = 6$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.1859	0.1481	0.0108	0.0012	0.0000	0.0000	0.0000
LS(M)	0.1213	0.1468	0.1449	0.1443	0.1933	0.2456	0.2266
DEC	0.0324	0.0259	0.0110	0.0015	0.0000	0.0000	0.0000
MMSE	0.0356	0.0275	0.0107	0.0013	0.0000	0.0000	0.0000
RLS	0.1278	0.1735	0.2183	0.2452	0.2572	0.2622	0.2644
RLS DD	0.1122	0.2426	0.3041	0.2965	0.3096	0.3119	0.3076
LS(K)	0.0338	0.0263	0.0103	0.0012	0.0000	0.0000	0.0000

Cuadro 5.3: Desviación típica de la probabilidad de error para $M = 6$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.4559	0.3752	0.0235	0.0139	0.0077	0.0040	0.0020
LS(M)	31.7567	36.9655	226.2925	8.5325	6.6518	74.6646	6.2839
DEC	0.1232	0.0617	0.0309	0.0155	0.0078	0.0039	0.0020
MMSE	0.0419	0.0371	0.0261	0.0156	0.0084	0.0042	0.0021
RLS	0.2770	0.3158	0.3676	0.4042	0.4271	0.4409	0.4493
RLS DD	0.3341	0.4855	0.6291	0.6152	0.6853	0.7220	0.6983
LS(K)	0.0396	0.0330	0.0229	0.0139	0.0077	0.0040	0.0020

Cuadro 5.4: Desviación típica del MSE para $M = 6$

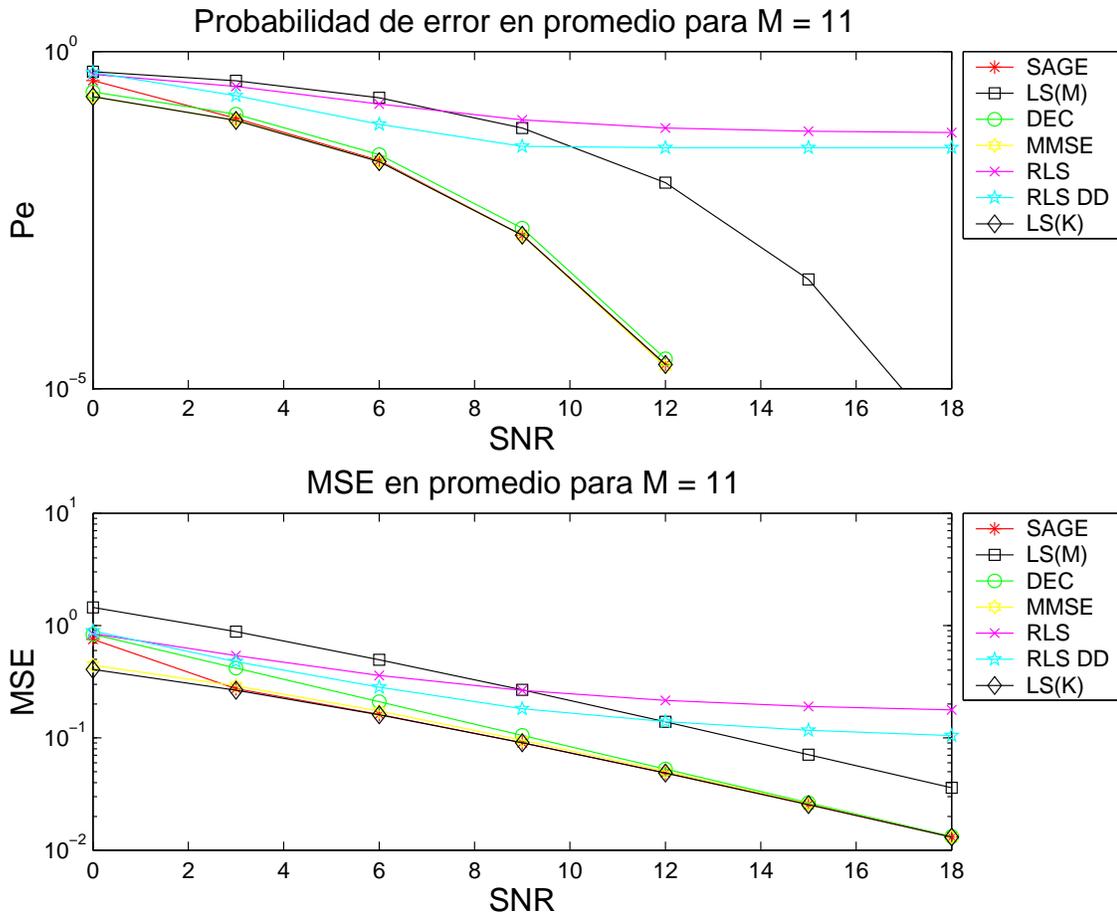


Figura 5.3: Probabilidad de error y MSE promediados sobre 20 canales para $M = 11$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.2062	0.0301	0.0108	0.0012	0.0000	0.0000	0.0000
LS(M)	0.0667	0.0637	0.0603	0.0380	0.0110	0.0008	0.0000
DEC	0.0324	0.0259	0.0110	0.0015	0.0000	0.0000	0.0000
MMSE	0.0356	0.0275	0.0107	0.0013	0.0000	0.0000	0.0000
RLS	0.0799	0.1164	0.1483	0.1559	0.1533	0.1508	0.1496
RLS DD	0.1130	0.1876	0.1819	0.1674	0.1678	0.1678	0.1678
LS(K)	0.0338	0.0263	0.0103	0.0012	0.0000	0.0000	0.0000

Cuadro 5.5: Desviación típica de la probabilidad de error para $M = 11$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.5518	0.0371	0.0235	0.0139	0.0077	0.0040	0.0020
LS(M)	0.4832	0.2567	0.1236	0.0624	0.0326	0.0163	0.0081
DEC	0.1232	0.0617	0.0309	0.0155	0.0078	0.0039	0.0020
MMSE	0.0419	0.0371	0.0261	0.0156	0.0084	0.0042	0.0021
RLS	0.1794	0.1666	0.1976	0.2162	0.2247	0.2283	0.2298
RLS DD	0.3241	0.4052	0.4123	0.4126	0.4076	0.4093	0.4110
LS(K)	0.0396	0.0330	0.0229	0.0139	0.0077	0.0040	0.0020

Cuadro 5.6: Desviación típica del MSE para $M = 11$

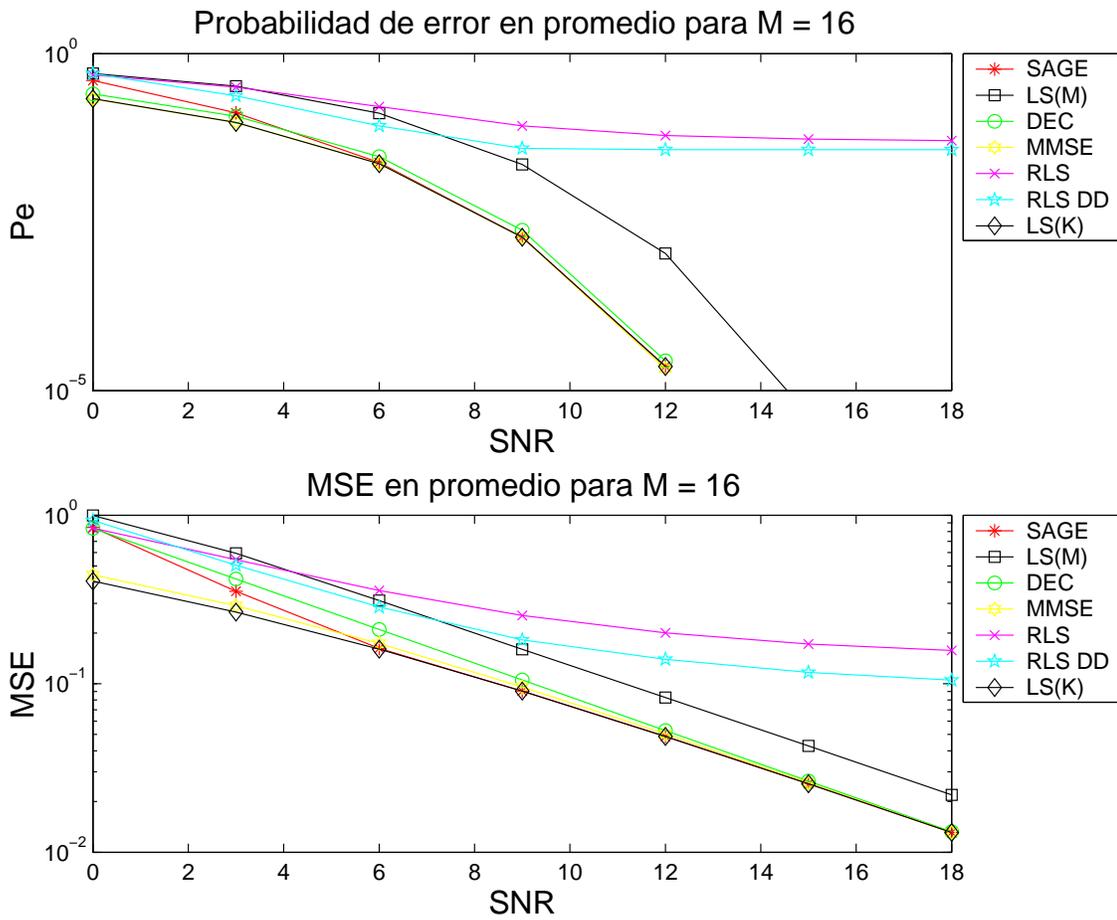


Figura 5.4: Probabilidad de error y MSE promediados sobre 20 canales para $M = 16$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.2364	0.1481	0.0108	0.0012	0.0000	0.0000	0.0000
LS(M)	0.0917	0.1012	0.0652	0.0170	0.0012	0.0000	0.0000
DEC	0.0324	0.0259	0.0110	0.0015	0.0000	0.0000	0.0000
MMSE	0.0356	0.0275	0.0107	0.0013	0.0000	0.0000	0.0000
RLS	0.0986	0.1231	0.1390	0.1464	0.1455	0.1435	0.1421
RLS DD	0.1166	0.2020	0.1816	0.1674	0.1678	0.1678	0.1678
LS(K)	0.0338	0.0263	0.0103	0.0012	0.0000	0.0000	0.0000

Cuadro 5.7: Desviación típica de la probabilidad de error para $M = 16$

	SNR						
	0	3	6	9	12	15	18
SAGE	0.6299	0.3672	0.0235	0.0139	0.0077	0.0040	0.0020
LS(M)	0.4440	0.2404	0.0835	0.0319	0.0147	0.0074	0.0039
DEC	0.1232	0.0617	0.0309	0.0155	0.0078	0.0039	0.0020
MMSE	0.0419	0.0371	0.0261	0.0156	0.0084	0.0042	0.0021
RLS	0.2580	0.1740	0.1823	0.1999	0.2099	0.2151	0.2177
RLS DD	0.3479	0.4221	0.4118	0.4126	0.4076	0.4093	0.4110
LS(K)	0.0396	0.0330	0.0229	0.0139	0.0077	0.0040	0.0020

Cuadro 5.8: Desviación típica del MSE para $M = 16$

sultados aceptables. Es debido a que una SNR baja quiere decir que la potencia del ruido es casi tan grande como la potencia de la señal deseada (igual para el caso de $SNR = 0$) y, por tanto, el ruido puede distorsionar la señal portadora de información de tal manera que sea imposible extraer los símbolos transmitidos con una calidad suficiente. Esto es lo que se observa en las figuras 5.5 y 5.6 ($SNR = 0$ y $SNR = 3$ respectivamente). Todos los algoritmos obtienen una probabilidad de error por encima de 10^{-1} , o en el mejor de los casos cercana a ese valor, para todos los valores de M probados. Además, se puede comprobar en los cuadros 5.9 y 5.11 en términos de probabilidad de error, y en los cuadros 5.10 y 5.12 en términos de error cuadrático medio, que el rendimiento de los diferentes algoritmos para estos dos valores de la SNR es fuertemente dependiente del canal utilizado. Sin embargo, es un hecho a destacar que cuando la $SNR = 3$ y $M \geq 6$ (figura 5.6) SAGE obtiene una tasa de fallos muy parecida a la del detector decorrelador, y algo inferior a la de LS, RLS y RLS DD.

A partir de $SNR = 6$, el algoritmo SAGE tan sólo precisa 6 símbolos para converger, y que el receptor demodule los símbolos del usuario deseado. En estas condiciones SAGE produce los mismos resultados que los algoritmos MMSE y decorrelador, e incluso que el algoritmo LS(K), que traza la que puede ser considerada la curva de referencia para sistemas prácticos. Así lo demuestran las figuras 5.7, 5.8 y 5.9, en las que se puede comprobar como SAGE alcanza un rendimiento similar al del algoritmo de mínimos cuadrados que supone conocida toda la trama de símbolos, LS(K). Además, cuando la SNR es 9 (figura 5.8) SAGE tiene una probabilidad de error de símbolo cercana a 10^{-3} , mientras que las conseguidas por los algoritmos LS o RLS, están por encima de 10^{-2} . Los cuadros 5.13, 5.15 y 5.17 contienen las desviaciones típicas asociadas a los datos revelados en las gráficas de probabilidad de error, mientras que los cuadros 5.14, 5.16 y 5.18 tienen esa misma información pero en relación con las gráficas de MSE.

En las simulaciones realizadas para $SNR \geq 15$ y $M \geq 6$, el algoritmo SAGE

detectó correctamente todos los símbolos transmitidos (1.000.000) por lo que su probabilidad de error exacta no aparece reflejada en las gráficas 5.10 y 5.11. Se puede comprobar su rendimiento en términos de error cuadrático medio, siendo éste del orden de 10^{-2} . Los algoritmos LS, RLS y RLS DD para estas SNR's tienen un rendimiento bastante inferior al de SAGE, aunque mejora a medida que se aumenta la longitud de la secuencia de entrenamiento. Por tanto, estos algoritmos son claramente dependientes del parámetro M . Las desviaciones típicas asociadas a las mencionadas figuras se pueden ver en los cuadros 5.19 y 5.21 (probabilidad de error), y 5.20 y 5.22 (error cuadrático medio).

Los resultados obtenidos en las simulaciones corroboran las afirmaciones hechas en la sección anterior. Aquí se puede apreciar con mayor claridad que la probabilidad de error y el MSE de los algoritmos MMSE y decorrelador dependen exclusivamente de la SNR. Se manifiesta en las diferentes figuras en que su traza de resultados es una línea recta, es decir, tanto la probabilidad de error como el MSE se mantienen constantes al aumentar M . Ocurre lo mismo para el caso del algoritmo LS(K), pues dicho algoritmo consiste en emplear una secuencia de entrenamiento de tamaño fijo tal que $M = K$.

En las simulaciones también se ha verificado que, dada una SNR fija, para que el algoritmo LS(M) alcance un rendimiento similar al del algoritmo SAGE es necesaria una secuencia de entrenamiento mucho mayor. Se pone de manifiesto en la figura 5.12, en la que se observa que para $SNR = 12$, el algoritmo LS(M) precisa de una secuencia de entrenamiento de longitud 100 ($M = 100$) para que su probabilidad de error pueda equiparse a la obtenida por el SAGE con una secuencia de entrenamiento de tan sólo 6 símbolos. También se puede comprobar en la misma gráfica que, incluso cuando el LS(M) utiliza 200 símbolos piloto, su rendimiento es ligeramente inferior al del SAGE para $M = 6$. La desviación típica de los datos de probabilidad de error mostrados en la figura 5.12 puede verse en el cuadro 5.23. La de los datos de MSE se muestra en el cuadro 5.24.

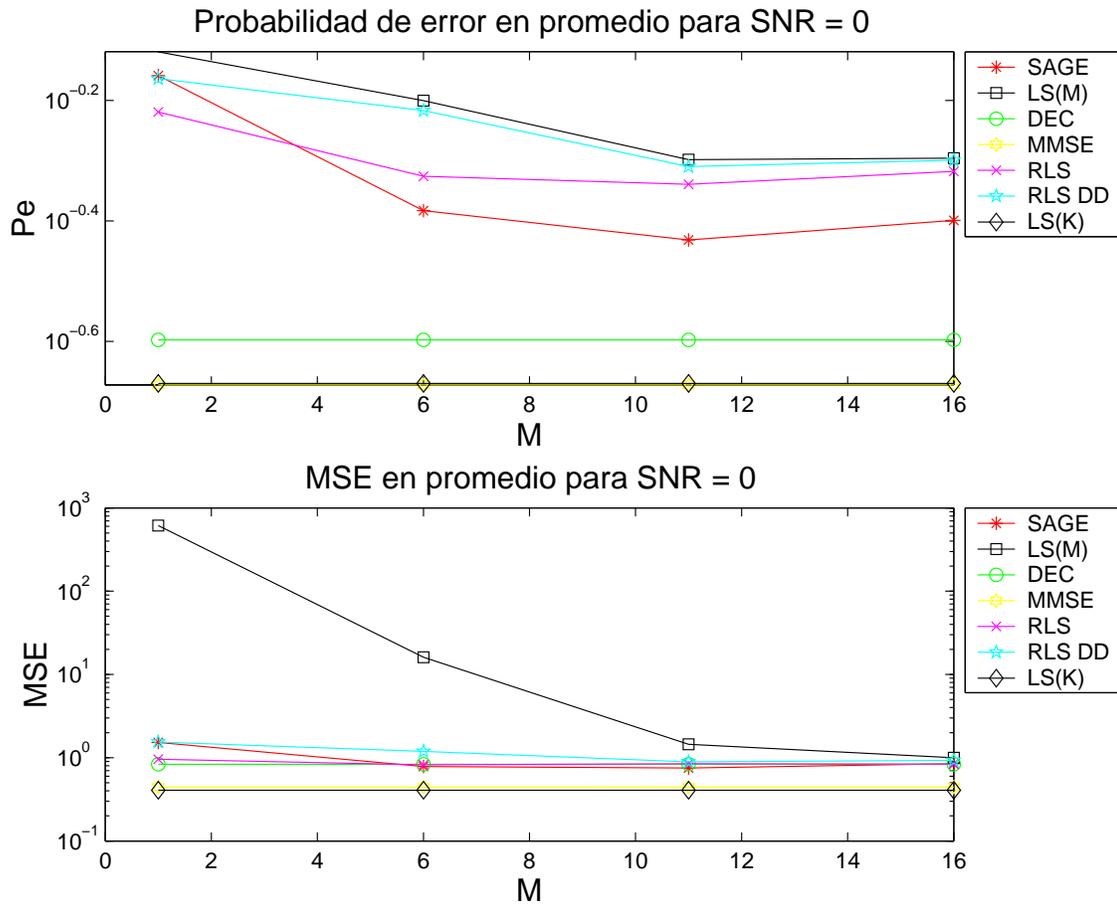


Figura 5.5: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 0$

	M			
	1	6	11	16
SAGE	0.1266	0.1859	0.2062	0.2364
LS(M)	0.1120	0.1213	0.0667	0.0917
DEC	0.0324			
MMSE	0.0356			
RLS	0.1446	0.1278	0.0799	0.0986
RLS DD	0.0970	0.1122	0.1130	0.1166
LS(K)	0.0338			

Cuadro 5.9: Desviación típica de la probabilidad de error para $SNR = 0$

	M			
	1	6	11	16
SAGE	0.4157	0.4559	0.5518	0.6299
LS(M)	1044.4158	31.7567	0.4832	0.4440
DEC	0.1232			
MMSE	0.0419			
RLS	0.2356	0.2770	0.1794	0.2580
RLS DD	0.4079	0.3341	0.3241	0.3479
LS(K)	0.0396			

Cuadro 5.10: Desviación típica del MSE para $SNR = 0$

	M			
	1	6	11	16
SAGE	0.2863	0.1481	0.0301	0.1481
LS(M)	0.0910	0.1468	0.0637	0.1012
DEC	0.0259			
MMSE	0.0275			
RLS	0.1733	0.1735	0.1164	0.1231
RLS DD	0.2002	0.2426	0.1876	0.2020
LS(K)	0.0263			

Cuadro 5.11: Desviación típica de la probabilidad de error para $SNR = 3$

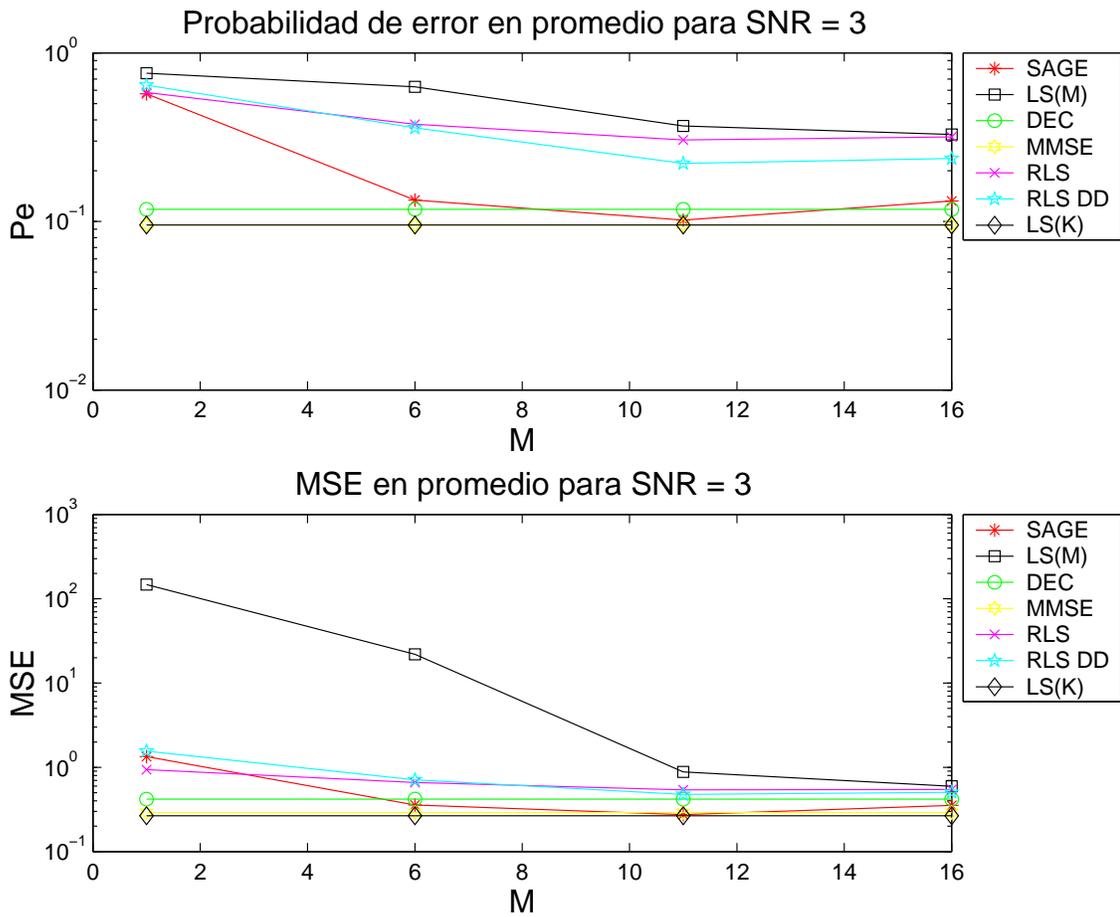


Figura 5.6: Probabilidad de error y MSE promediados sobre 20 canales para $SNR =$

	M			
	1	6	11	16
SAGE	0.6760	0.3752	0.0371	0.3672
LS(M)	171.3498	36.9655	0.2567	0.2404
DEC	0.0617			
MMSE	0.0371			
RLS	0.2698	0.3158	0.1666	0.1740
RLS DD	0.5935	0.4855	0.4052	0.4221
LS(K)	0.0330			

Cuadro 5.12: Desviación típica del MSE para $SNR = 3$

	M			
	1	6	11	16
SAGE	0.3564	0.0108	0.0108	0.0108
LS(M)	0.1506	0.1449	0.0603	0.0652
DEC	0.0110			
MMSE	0.0107			
RLS	0.1924	0.2183	0.1483	0.1390
RLS DD	0.3278	0.3041	0.1819	0.1816
LS(K)	0.0103			

Cuadro 5.13: Desviación típica de la probabilidad de error para $SNR = 6$

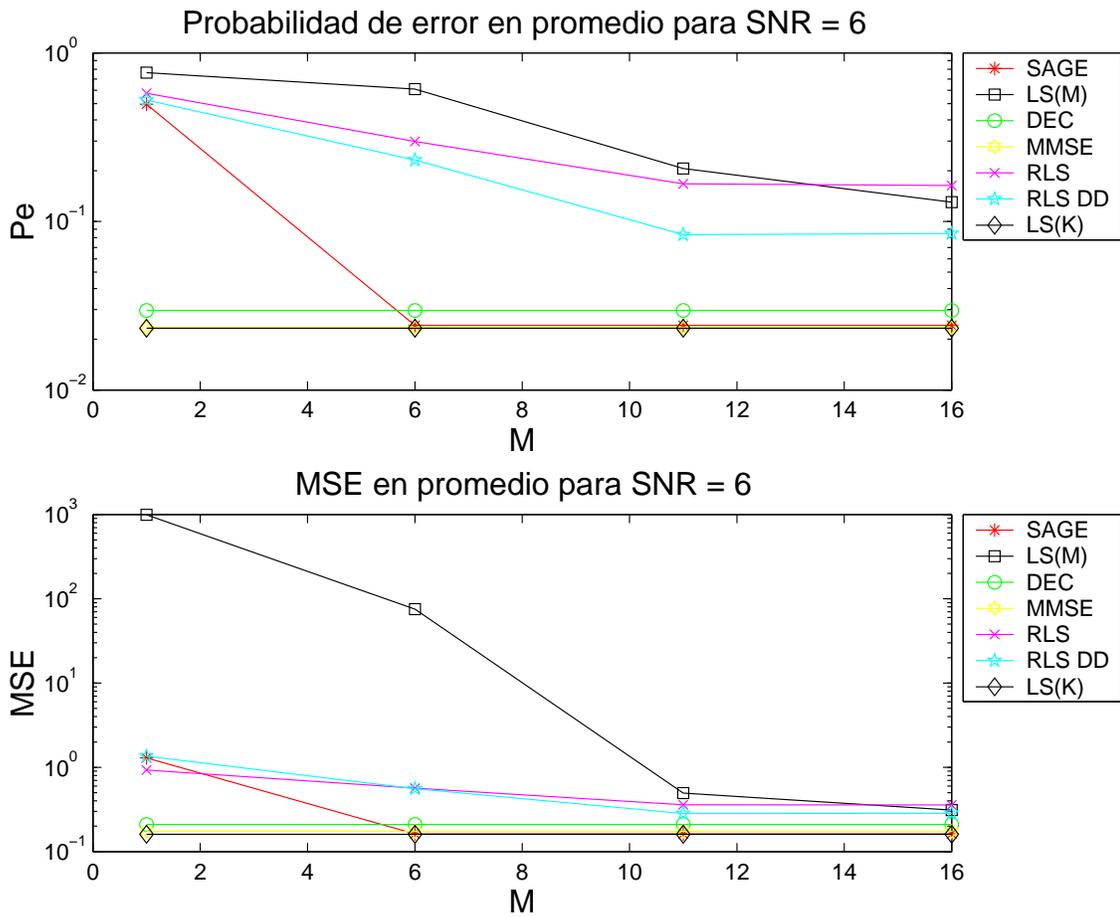


Figura 5.7: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 6$

	M			
	1	6	11	16
SAGE	0.8557	0.0235	0.0235	0.0235
LS(M)	3544.4406	226.2925	0.1236	0.0835
DEC	0.0309			
MMSE	0.0261			
RLS	0.2902	0.3676	0.1976	0.1823
RLS DD	0.8259	0.6291	0.4123	0.4118
LS(K)	0.0229			

Cuadro 5.14: Desviación típica del MSE para $SNR = 6$

	M			
	1	6	11	16
SAGE	0.3762	0.0012	0.0012	0.0012
LS(M)	0.1415	0.1443	0.0380	0.0170
DEC	0.0015			
MMSE	0.0013			
RLS	0.1993	0.2452	0.1559	0.1464
RLS DD	0.3308	0.2965	0.1674	0.1674
LS(K)	0.0012			

Cuadro 5.15: Desviación típica de la probabilidad de error para $SNR = 9$

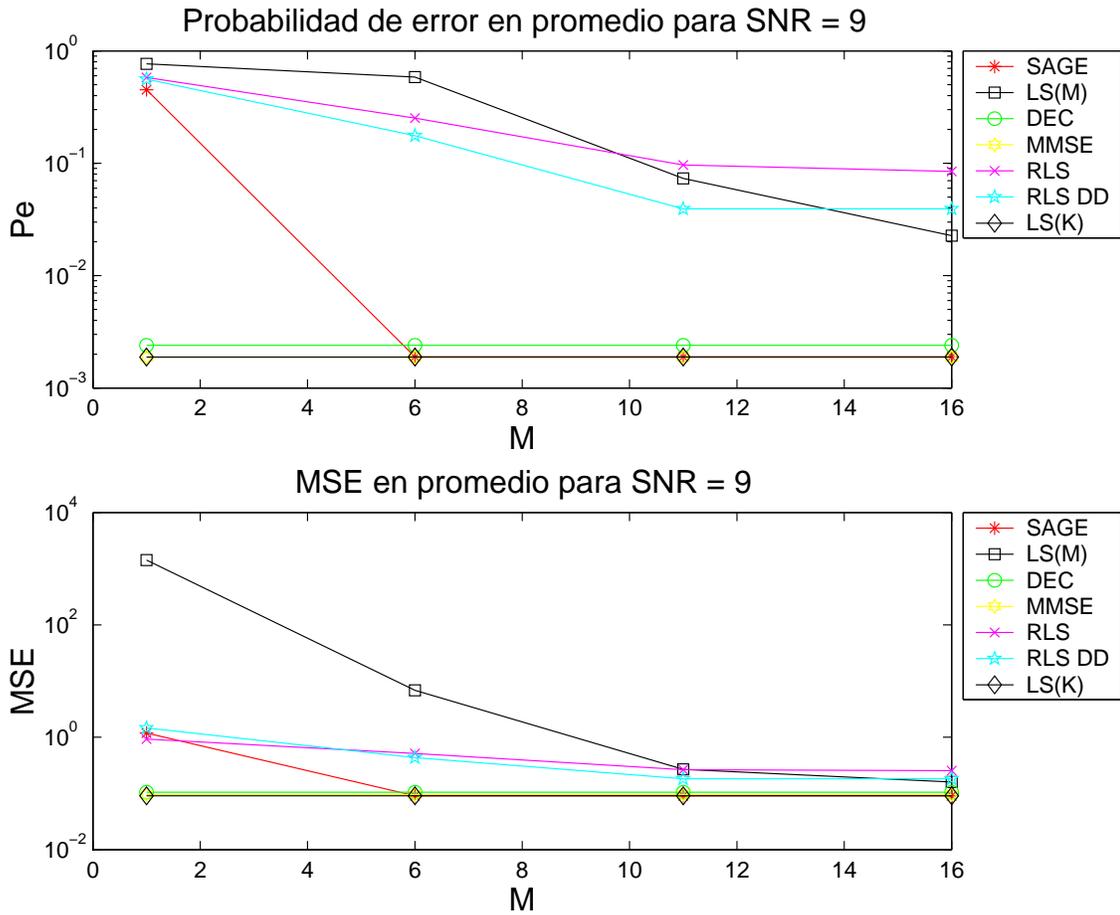


Figura 5.8: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 9$

	M			
	1	6	11	16
SAGE	0.9242	0.0139	0.0139	0.0139
LS(M)	5255.6156	8.5325	0.0624	0.0319
DEC	0.0155			
MMSE	0.0156			
RLS	0.3028	0.4042	0.2162	0.1999
RLS DD	0.8246	0.6152	0.4126	0.4126
LS(K)	0.0139			

Cuadro 5.16: Desviación típica del MSE para $SNR = 9$

	M			
	1	6	11	16
SAGE	0.3331	0.0000	0.0000	0.0000
LS(M)	0.1313	0.1933	0.0110	0.0012
DEC	0.0000			
MMSE	0.0000			
RLS	0.1998	0.2572	0.1533	0.1455
RLS DD	0.3066	0.3096	0.1678	0.1678
LS(K)	0.0000			

Cuadro 5.17: Desviación típica de la probabilidad de error para $SNR = 12$

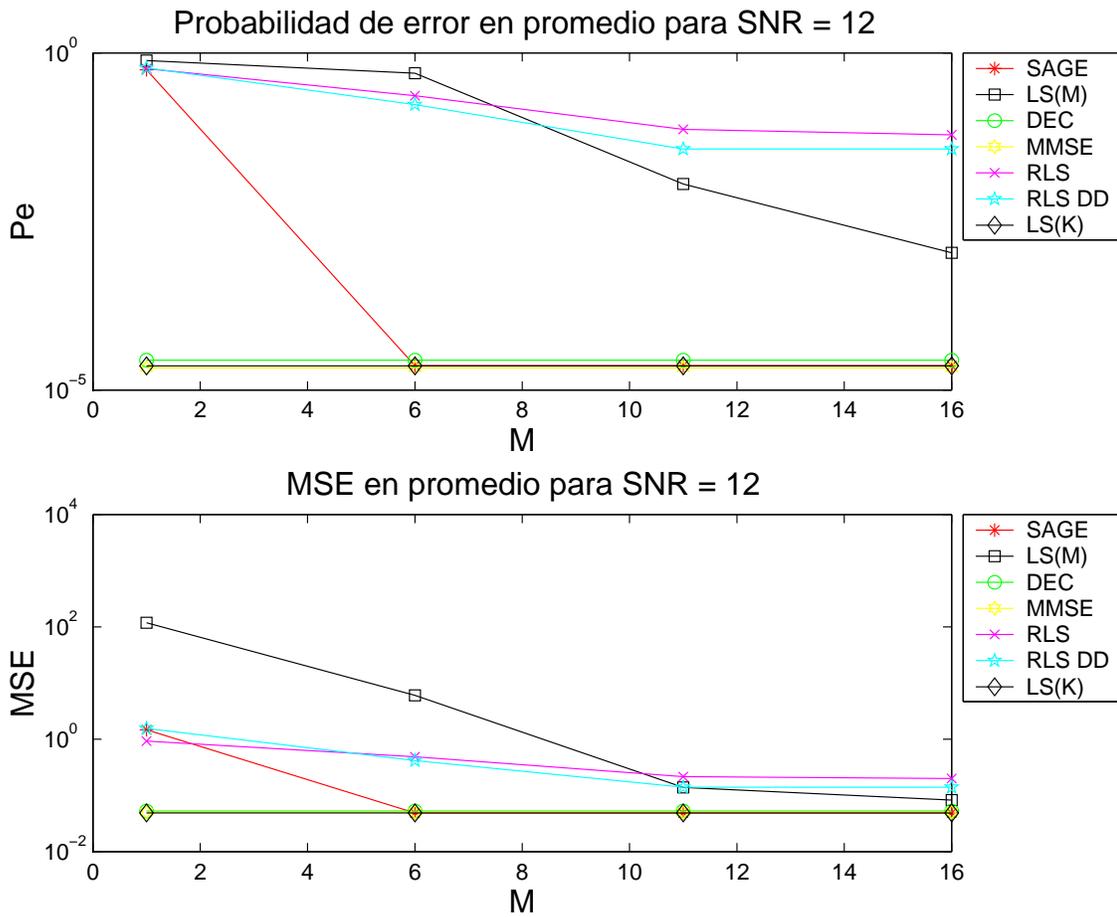


Figura 5.9: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 12$

	M			
	1	6	11	16
SAGE	0.8484	0.0077	0.0077	0.0077
LS(M)	240.0060	6.6518	0.0326	0.0147
DEC	0.0078			
MMSE	0.0084			
RLS	0.3109	0.4271	0.2247	0.2099
RLS DD	0.7900	0.6853	0.4076	0.4076
LS(K)	0.0077			

Cuadro 5.18: Desviación típica del MSE para $SNR = 12$

	M			
	1	6	11	16
SAGE	0.2748	0.0000	0.0000	0.0000
LS(M)	0.0870	0.2456	0.0008	0.0000
DEC	0.0000			
MMSE	0.0000			
RLS	0.1991	0.2622	0.1508	0.1435
RLS DD	0.3330	0.3119	0.1678	0.1678
LS(K)	0.0000			

Cuadro 5.19: Desviación típica de la probabilidad de error para $SNR = 15$

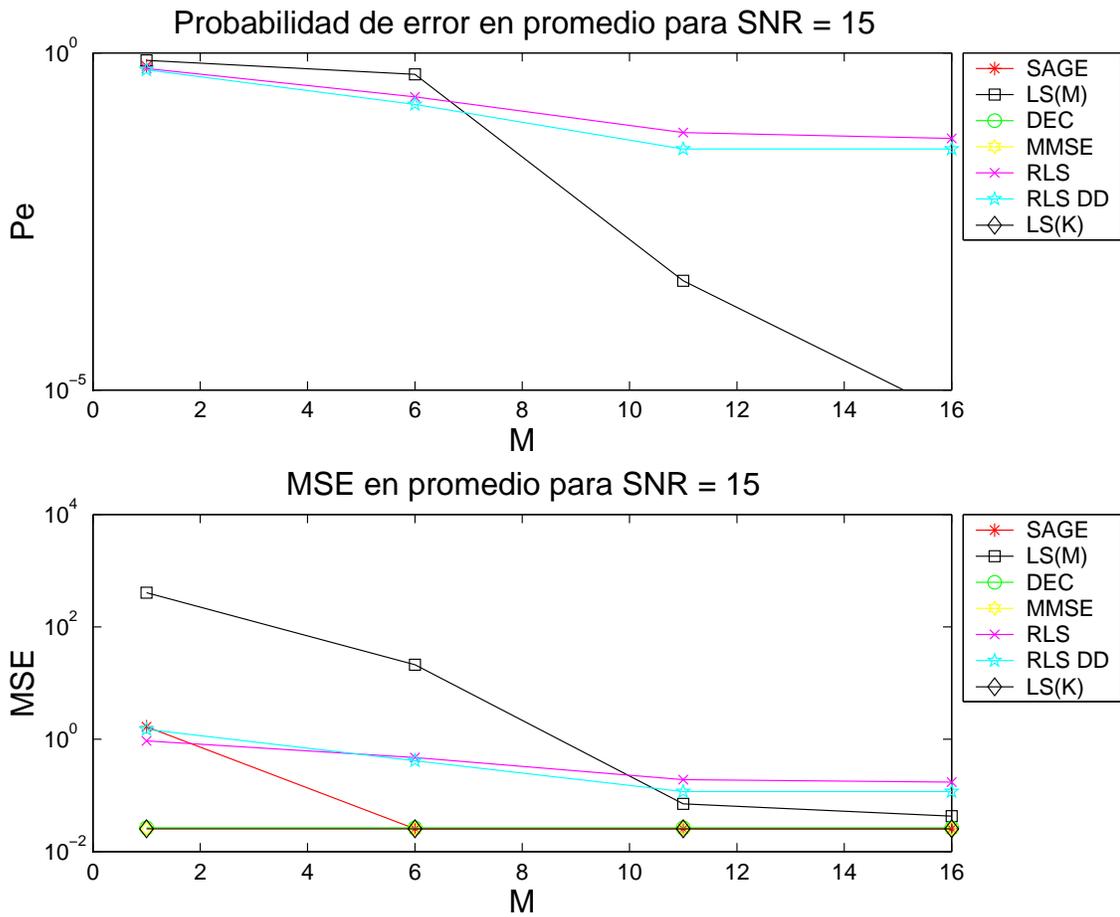


Figura 5.10: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 15$

	M			
	1	6	11	16
SAGE	0.7119	0.0040	0.0040	0.0040
LS(M)	941.1783	74.6646	0.0163	0.0074
DEC	0.0039			
MMSE	0.0042			
RLS	0.3160	0.4409	0.2283	0.2151
RLS DD	0.8626	0.7220	0.4093	0.4093
LS(K)	0.0040			

Cuadro 5.20: Desviación típica del MSE para $SNR = 15$

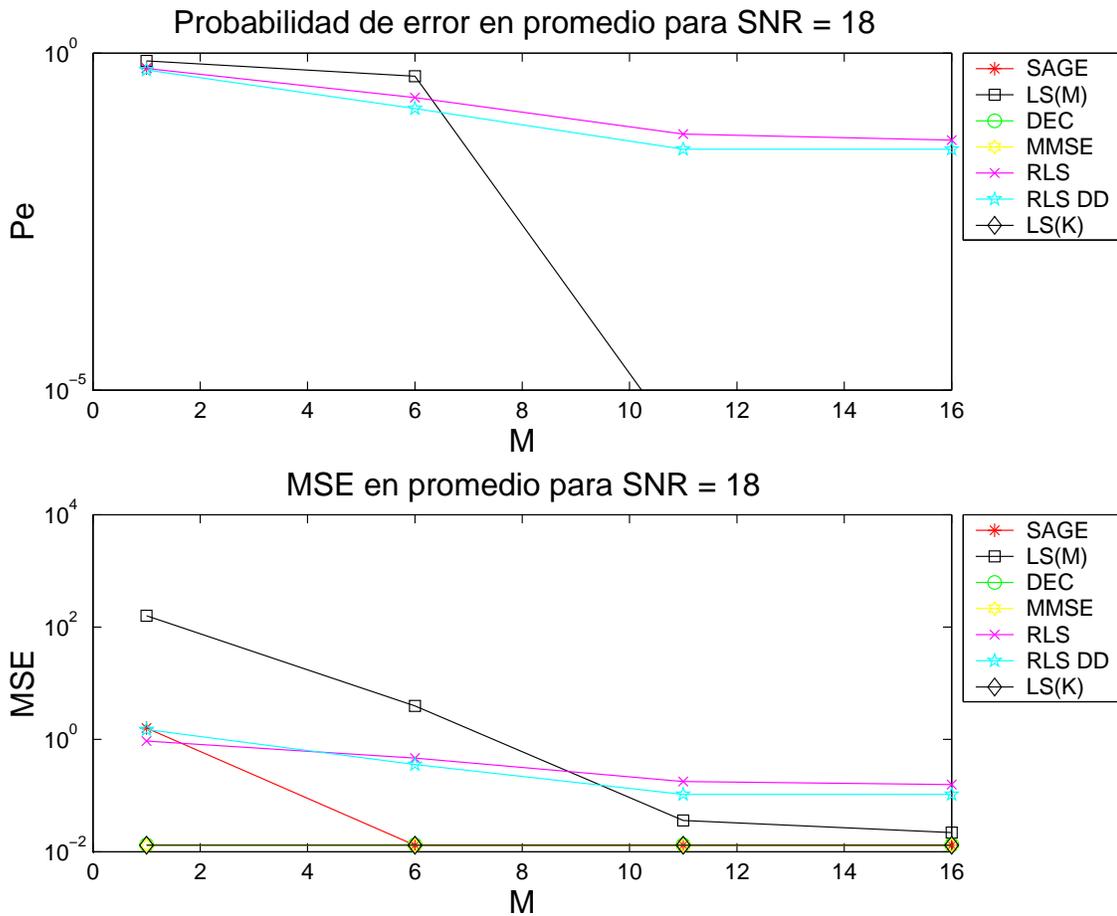


Figura 5.11: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 18$

5.3.3. Probabilidad de error y MSE en función del número de usuarios N

La probabilidad de error de un sistema CDMA depende también del número de usuarios, N , que lo utilicen. En la presente sección se analiza como varía la probabilidad de error y el MSE en función del número de usuarios que en un determinado momento estén haciendo uso del sistema.

En la figura 5.13 se puede comprobar como el rendimiento de los diferentes

	M			
	1	6	11	16
SAGE	0.3079	0.0000	0.0000	0.0000
LS(M)	0.1135	0.2266	0.0000	0.0000
DEC	0.0000			
MMSE	0.0000			
RLS	0.1980	0.2644	0.1496	0.1421
RLS DD	0.3333	0.3076	0.1678	0.1678
LS(K)	0.0000			

Cuadro 5.21: Desviación típica de la probabilidad de error para $SNR = 18$

	M			
	1	6	11	16
SAGE	0.8080	0.0020	0.0020	0.0020
LS(M)	179.3924	6.2839	0.0081	0.0039
DEC	0.0020			
MMSE	0.0021			
RLS	0.3190	0.4493	0.2298	0.2177
RLS DD	0.8737	0.6983	0.4110	0.4110
LS(K)	0.0020			

Cuadro 5.22: Desviación típica del MSE para $SNR = 18$

	M									
	1	6	11	16	21	26	31	50	100	200
SAGE	0.3078	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
LS(M)	0.1900	0.2245	0.0183	0.0021	0.0008	0.0002	0.0002	0.0001	0.0000	0.0000
MMSE	0.0000									
LS(K)	0.0000									

Cuadro 5.23: Desviación típica de la probabilidad de error para $SNR = 12$

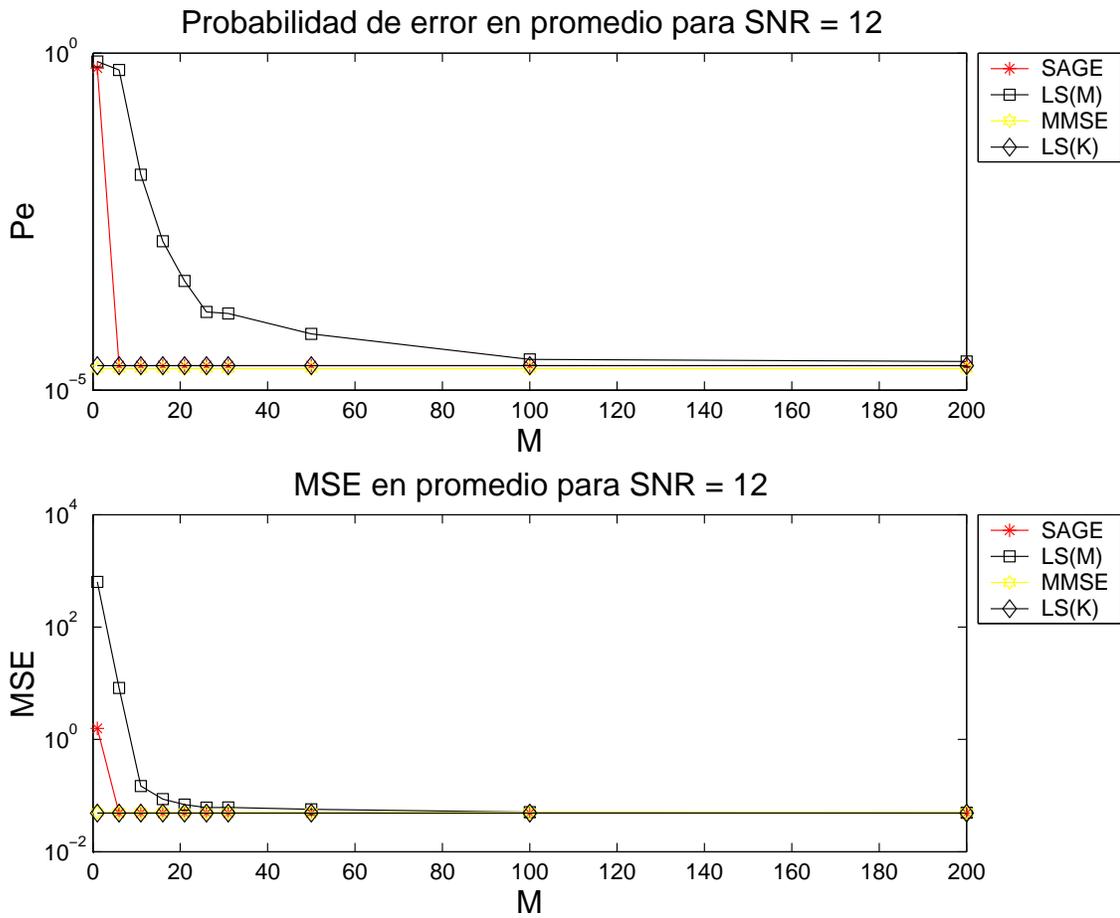


Figura 5.12: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 12$

algoritmos se va degradando a medida que aumenta N . Sin embargo, el rendimiento va empeorando de forma progresiva, y en ningún caso se puede hablar de un número de usuarios umbral, sobrepasado el cual el sistema no funciona. Esto es aplicable a todos los algoritmos estudiados.

En los cuadros 5.25 y 5.26 aparece reflejada la desviación típica de los datos plasmados en las gráficas.

	M									
	1	6	11	16	21	26	31	50	100	200
SAGE	0.7854	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077	0.0077
LS(M)	1317.2645	11.2481	0.0453	0.0194	0.0139	0.0111	0.0106	0.0093	0.0077	0.0079
MMSE	0.0083									
LS(K)	0.0077									

Cuadro 5.24: Desviación típica del MSE para $SNR = 12$

	N					
	1	3	5	7	9	11
SAGE	0.0000	0.0000	0.0000	0.0034	0.0350	0.2899
LS(M)	0.0000	0.0017	0.0011	0.0083	0.0576	0.1935
DEC	0.0000	0.0000	0.0000	0.0061	0.0337	0.1623
MMSE	0.0000	0.0000	0.0000	0.0032	0.0335	0.1626
RLS	0.0000	0.0009	0.0815	0.1443	0.1715	0.1998
RLS DD	0.0000	0.0000	0.1489	0.1511	0.2022	0.2518
LS(K)	0.0000	0.0000	0.0000	0.0031	0.0329	0.1654

Cuadro 5.25: Desviación típica de la probabilidad de error para $SNR = 12$ y $M = 16$

	N					
	1	3	5	7	9	11
SAGE	0.0001	0.0044	0.0063	0.0189	0.0600	0.5487
LS(M)	0.0028	0.0137	0.0139	0.0235	0.0771	0.3509
DEC	0.0000	0.0052	0.0069	0.0304	0.0744	0.1996
MMSE	0.0000	0.0048	0.0067	0.0205	0.0624	0.2013
RLS	0.0287	0.0580	0.1217	0.1893	0.2379	0.3105
RLS DD	0.0002	0.0045	0.2424	0.2816	0.3662	0.4244
LS(K)	0.0001	0.0044	0.0063	0.0185	0.0579	0.2057

Cuadro 5.26: Desviación típica del MSE para $SNR = 12$ y $M = 16$

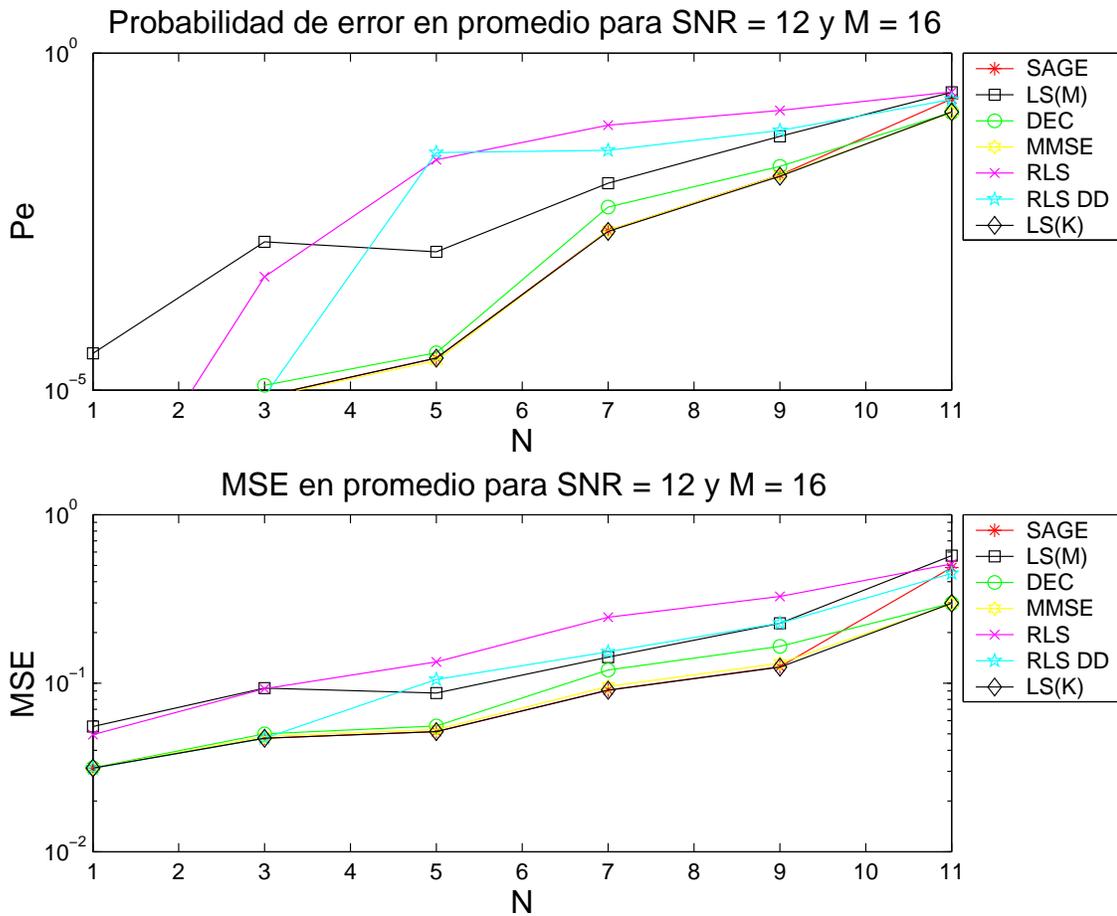


Figura 5.13: Probabilidad de error y MSE promediados sobre 20 canales para $SNR = 12$ y $M = 16$

5.4. Conclusiones

Los resultados de las simulaciones dejan patente la influencia que tienen los parámetros estudiados tanto en la probabilidad de error como en el error cuadrático medio de cada uno de los algoritmos.

Los algoritmos clásicos (LS, RLS y RLS DD) son muy sensibles a la longitud de la secuencia de entrenamiento, pues los símbolos que componen esta última constituyen la única información que estos algoritmos emplean para calcular los coeficientes del filtro. No hacen uso pues de ningún tipo de información estadística,

y son por ello poco eficientes. El algoritmo EM por su parte, utiliza toda la información estadística de que dispone y la secuencia de entrenamiento únicamente para evitar la captura de la señal de uno de los usuarios interferentes, por lo que su rendimiento (medido en términos de probabilidad de error de símbolo y error cuadrático medio) no depende de M una vez que se garantiza la convergencia de los coeficientes del filtro a una solución que extraiga los símbolos del usuario deseado. EM necesitará, por lo tanto, una secuencia de entrenamiento mucho más corta que la requerida por los algoritmos basados en mínimos cuadrados. Ello se traduce en un mayor aprovechamiento de la trama, y en una mayor eficiencia que se puede medir como:

$$\mu = \frac{K - M}{K} \quad (5.9)$$

Desde este punto de vista, la mayor eficiencia se consigue cuando $M = 0$, es decir, cuando no es necesario utilizar secuencia de entrenamiento, en cuyo caso $\mu = 1$.

Capítulo 6

Conclusiones y líneas futuras de investigación

6.1. Conclusiones

A lo largo del presente proyecto se ha estudiado una nueva aproximación para la cancelación de interferencias en un sistema de comunicaciones multiusuario con modulación multiportadora.

Para ello se ha empleado un receptor lineal, que consta de un filtro FIR (que cumple la función de eliminar la MAI, igualar el canal y mitigar los efectos del ruido térmico) y de un detector de umbral.

Lo que diferencia a un receptor lineal de otro es básicamente la manera de seleccionar los coeficientes del filtro. En el capítulo 3 se han estudiado las aproximaciones más clásicas para llevar a cabo esta tarea: MMSE, detector decorrelador, LS, RLS y RLS dirigido por decisión. Los receptores MMSE y decorrelador requieren conocer perfectamente el canal \mathbf{H} y son por ello difíciles de llevar a la práctica (requieren una etapa previa de estimación del canal). Además, desde el punto de vista de la seguridad no es conveniente que un usuario del sistema conozca los códigos de los demás usuarios. En este trabajo se han utilizado como referencia

para evaluar el rendimiento de los restantes métodos. Los algoritmos LS, RLS y RLS DD, al contrario, sí son aproximaciones prácticas, aunque tienen importantes limitaciones derivadas de que se requiere una secuencia de entrenamiento de un tamaño considerable para que los resultados obtenidos sean satisfactorios.

La aproximación propuesta en el presente proyecto para resolver el problema de cancelación de interferencias está basada en el principio de máxima verosimilitud (ML). Parte de la hipótesis de que en caso de existir un filtro óptimo capaz de eliminar por completo la MAI e igualar el canal, las estimaciones que se obtendrían utilizando el mismo serían la suma de dos variables aleatorias: el símbolo transmitido por el usuario (pudiendo estar en principio escalado por un factor complejo) y una variable aleatoria representando el ruido que hay a la salida del filtro. La f.d.p. asociada a las estimaciones blandas que proporcionaría ese filtro óptimo depende de los coeficientes del mismo. Entonces, se puede interpretar el problema de seleccionar los coeficientes del filtro como un problema de estimación ML: se trata de elegir aquellos coeficientes del receptor que maximicen la probabilidad de obtener estimaciones blandas de la forma deseada (el símbolo transmitido por el usuario más un término de ruido). El problema de optimización resultante no tiene solución analítica, pero se puede convertir en una secuencia de problemas cuadráticos (con solución cerrada, por tanto) mediante la metodología SAGE.

La técnica resultante utiliza toda la información estadística disponible en el problema, pero sufre un problema de captura de interferencias por parte del receptor debido a que todos los usuarios del sistema MC-CDMA tienen las mismas características estadísticas a priori. Podría ocurrir por lo tanto, que el filtro así obtenido extrajera los símbolos de un usuario que no es el de interés.

Para evitar este *problema de captura* se ha optado por utilizar un reducido número de símbolos piloto dentro del algoritmo SAGE, con lo que la aproximación propuesta se considera *semiciega*. El uso de esta secuencia de entrenamiento sirve únicamente para garantizar la extracción del usuario deseado, por lo que su longitud

no condiciona el rendimiento del receptor, una vez que los coeficientes de éste convergen a la solución del usuario de interés.

Se ha hecho un estudio detallado del rendimiento del receptor propuesto mediante simulaciones por ordenador. Los resultados obtenidos ponen de manifiesto que el receptor SAGE alcanza prácticamente el mismo rendimiento que los detectores teóricos LS(K) y MMSE, y supera al detector DEC. Además, es mucho más eficiente que los métodos clásicos porque utiliza secuencias de entrenamiento mucho más cortas (hasta un factor cercano a 40 comparado con el receptor LS). Ello da lugar a una mayor eficiencia en el aprovechamiento de la trama, que puede medirse como:

$$\mu = \frac{K - M}{K} \quad (6.1)$$

6.2. Líneas futuras de investigación

La investigación en torno a la aplicación del principio de máxima verosimilitud en la supresión de interferencias en sistemas MC-CDMA debe orientar sus esfuerzos a eliminar por completo la secuencia de entrenamiento, de tal manera que $\mu \simeq 1$ y la eficiencia de la trama de comunicaciones sea máxima.

En [1] se demuestra que cuando el algoritmo EM converge, las estimaciones blandas obtenidas con los coeficientes del filtro calculado nunca son suma de los símbolos transmitidos por varios usuarios, es decir, cada estimación blanda se compone del símbolo transmitido por *uno* de los usuarios (escalado posiblemente por un factor complejo) y un término de ruido. Así pues, para evitar la necesidad de utilizar una secuencia de entrenamiento podría pensarse en imponer algún tipo de restricción a los coeficientes del filtro \mathbf{w} , de tal manera que el filtro construido para extraer los símbolos del usuario i verificase

$$\mathbf{w}^H \mathbf{d}_i = k \quad (6.2)$$

siendo k una constante y \mathbf{w} el filtro para extraer los símbolos del i -ésimo usuario.

De esta forma, se garantiza que la estimación blanda obtenida por el filtro contiene siempre al símbolo transmitido por el usuario de interés, y no se está capturando ninguno de los símbolos transmitidos por los usuarios interferentes:

$$y(n) = \mathbf{w}^H \mathbf{x}(n) = \underbrace{\sum_{j \neq i} \mathbf{w}^H \mathbf{d}_j b_j(n)}_{MAI} + kb_i(n) + g_f(n) = kb_i(n) + g_f(n). \quad (6.3)$$

En [1] se proponen restricciones lineales para lograr este objetivo en un sistema DS-CDMA que, sin embargo, no pueden ser aplicadas al modelo estudiado debido a la naturaleza multiplicativa del canal en MC-CDMA.

Teniendo en cuenta que la restricción impuesta por la ecuación (6.2) implica conocer el canal del usuario deseado, una posible solución para evitar el uso de secuencia de entrenamiento sería utilizar algún modelo estadístico o estructural del canal que obligue a que se verifique dicha ecuación.

En el futuro, también sería interesante estudiar el rendimiento del receptor propuesto sobre canales reales, o sobre un demostrador de MC-CDMA.

Apéndice A

Abreviaturas

La lista de abreviaturas utilizadas a lo largo del trabajo es la siguiente:

- AWGN: Additive White Gaussian Noise.
- CDMA: Code Division Multiple Access.
- DEC: Decorrelador (detector).
- DFT: Discreet Fourier Transform
- DS-CDMA: Direct-Sequence CDMA
- EM: Expectation Maximization.
- FDMA: Frequency Division Multiple Access.
- FIR: Finite Impulse Response.
- f.d.p.: función de densidad de probabilidad.
- f.m.p.: función de masa de probabilidad.
- IDFT: Inverse Discreet Fourier Transform
- i.i.d.: idénticas e independientemente distribuidas.

- IMT-2000: International Mobile Telecommunication by the year 2000
- ISI: Inter-Symbol Interference
- LS: Least Squares.
- MAI: Multiple Access Interference.
- MC: MultiCarrier
- MC-CDMA: MultiCarrier-CDMA
- ML: Maximum Likelihood.
- MMSE: Minimum Mean Square Error.
- MSE: Mean Square Error.
- MT-CDMA: MultiTone-CDMA.
- OFDM: Orthogonal Frequency Division Multiplexing
- QPSK: Quadrature Phase Shift Keying.
- RLS: Recursive Least Squares.
- RLS DD: RLS Dirigido por Decisión.
- SAGE: Space Alternating Generalized EM
- SDMA: Space Division Multiple access.
- SNR: Signal to Noise Ratio
- TDMA: Time Division Multiple Access.
- UMTS: Universal Mobile Telecommunication system.

Bibliografía

- [1] M.F. Bugallo, J. Míguez, y L. Castedo. A maximum likelihood approach to blind multiuser interference cancellation. *IEEE transactions on signal processing*, 49(6):1228–1239, Junio 2001.
- [2] Prodip Chaudhury, Werner Mohr, y Seizo Onoe. The 3GPP proposal for IMT-2000. *IEEE Communications magazine*, páginas 72–81, Diciembre 1999.
- [3] J.A. Fessler y A.O. Hero. Space-alternating generalized expectation-maximization algorithm. *IEEE Trans. Signal Processing*, 42(10):2664–2677, Octubre 1994.
- [4] Shinsuke Hara y Ramjee Prasad. Overview of Multicarrier CDMA. *IEEE Communications magazine*, páginas 126–133, Diciembre 1997.
- [5] S. Haykin. *Adaptive Filter Theory, Third Edition*. Prentice Hall, 1996.
- [6] Harri Holma y T. Antti Toskala. *WCDMA for UMTS*. Wiley, 2000.
- [7] U. Madhow y M.L. Honig. Mmse interference suppression for direct-sequence spread-spectrum cdma. *IEEE Trans. Commun.*, 42:3178–3188, Diciembre 1994.
- [8] Geoffrey J. McLachlan y T. Hriyambakam Krishna. *The EM algorithm and extensions*. John Wiley and sons, inc., 1997.

- [9] Joaquín Míguez y Luis Castedo. A constant modulus blind adaptive receiver for multiuser interference suppression. *Signal processing*, (71):15–27, 1998.
- [10] Alan V. Oppenheim y Ronald W. Schaffer. *Discrete-time signal processing*. Prentice Hall, 1989.
- [11] Ramjee Prasad. *CDMA for wireless personal communication*. Artech house, 1996.
- [12] Ramjee Prasad. *Universal wireless personal communications*. Artech House Publishers, 1998.
- [13] J. G. Proakis. *Digital Communications*. McGraw-Hill, 1995.
- [14] William Stallings. *Comunicaciones y redes de computadores*. Prentice Hall, 2000.
- [15] C. W. Therrien. *Discrete Random signals and statistical signal processing*. Prentice Hall, 1992.
- [16] Sergio Verdú. *Multiuser Detection*. Cambridge University Press, 1998.
- [17] Ulrich Vornefeld, Christoph Walke, y Bernhard Walke. SDMA Techniques for Wireless ATM. *IEEE Communications magazine*, páginas 52–57, Noviembre 1999.