Enunciados

1) Explica cuál es el principal inconveniente del algoritmo LMS (en general), y propón algún método alternativo que lo solucione.

2) Explica cómo funciona el algoritmo sistema RLS. Detalla claramente la definición de la función error así como los parámetros que intervienen en ella, el procedimiento de resolución del sistema, la función del factor de memoria así como su impacto en las prestaciones del algoritmo. Compara este sistema con el método LMS indicando las ventajas e inconvenientes del algoritmo.

3) En un problema de cancelación de ruido tenemos el siguiente escenario (figura 1):

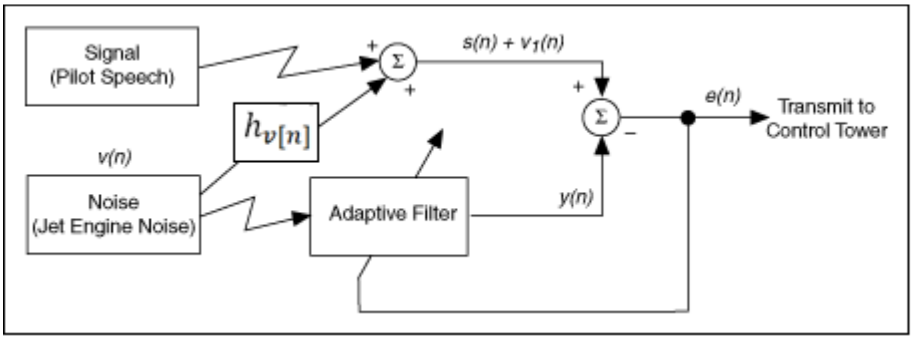


Figura 1: Cancelación de ruido mediante filtro adaptativo

Una fuente de señal $s[n]$ está mezclada con un ruido $v_1[n]$. Este ruido se ha podido medir en otro punto $v[n]$, de manera que$v_1[n] = v[n] * h_{v[n]}$. Se quiere aplicar un filtro adaptativo FIR con vector de coeficientes $\mathbf{h} = [h_0, h_1, h_2, \dots h_9]$ para estimar $v_1[n]$ a partir de $v[n]$. Llamaremos $y[n]$ a esta estimación: $y[n] \cong v_1[n]$

Se pide:

a) Suponiendo que $h_v = 0.9^n u[n]$, calcula la solución óptima del filtro $\mathbf{h}$ tal que la señal $e[n]$ sea lo más parecida posible a $s[n]$ (fuente de señal sin ruido). Para ello, suponemos que la señal $v[n]$ es de miedia cero, blanca y de potencia σ_v^2, y independient de la señal $s[n]$.

b) Explica de forma intuitiva cuál crees que sería la estructura más adiente del sistema h_v para conseguir eliminar totalmente el ruido en $e[n]$.

c) Suponiendo que $s[n] + v_1[n]$ y $v[n]$ son las señales periódicas de la figura 2, calcula las 4 primeras iteraciones de un filtro adaptativo, ahora de 2 coeficientes, que utilice el algoritmo LMS con un parámetro de adaptación $\mu = 0.5$. Suponemos que el registro del filtro se inicializa a 0 en $n = -1$, que empezamos a iterar el filtro a partir de $n \geq 0$ y que los coeficientes del filtro también se inicializan a 0.

Da como resultado tanto el valor de los coeficientes como el error para cada iteración. Razona claramente hacia qué solución crees que tenderán a estabilizarse en los coeficientes del filtro en régimen permanente.

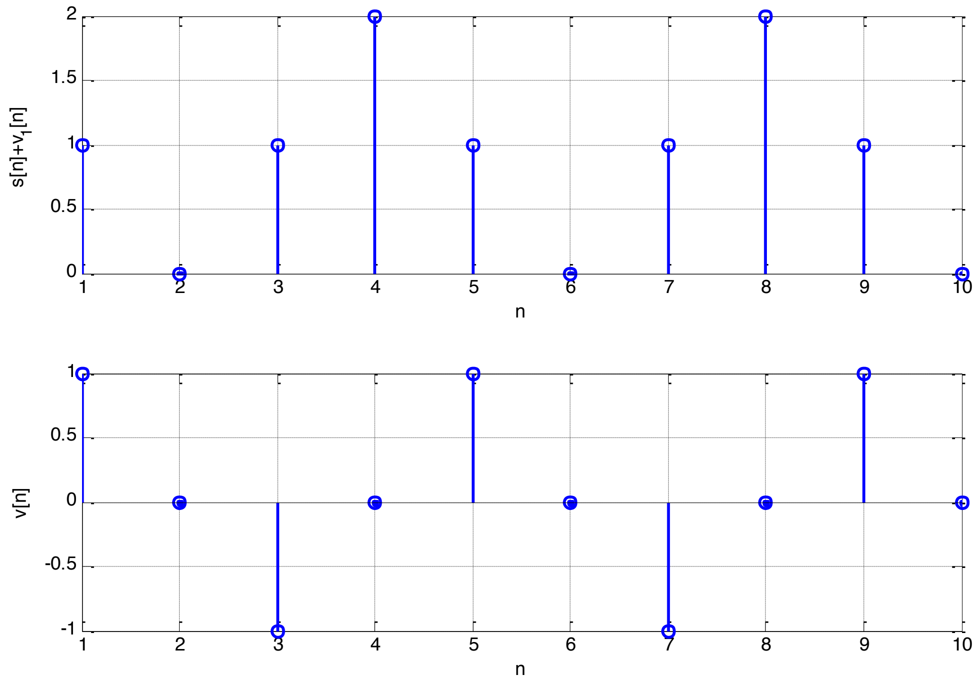
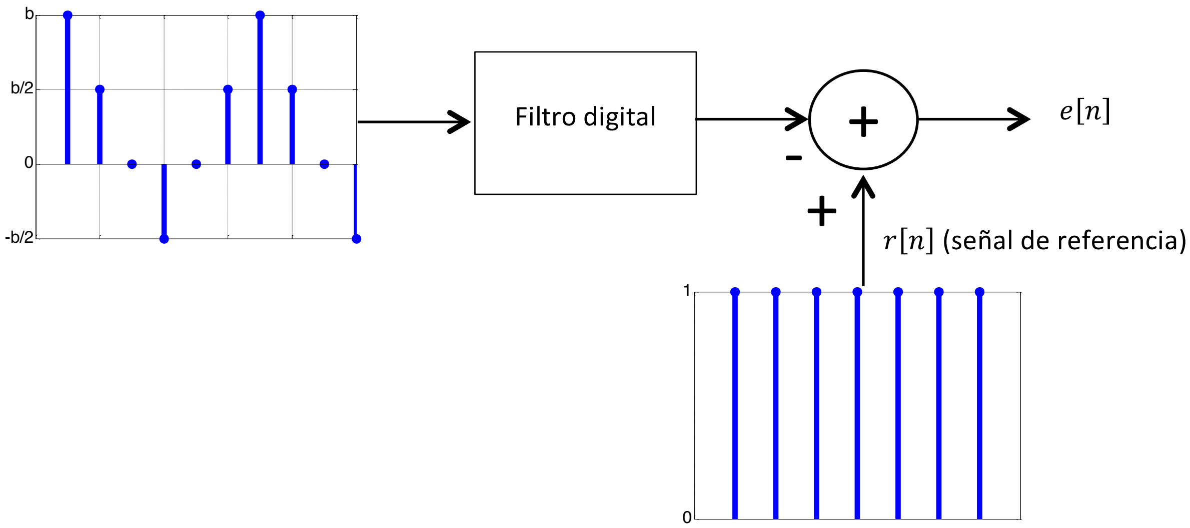


Figura 2: Señales disponibles en los sensores. Matemáticamente corresponden a
$\boldsymbol{s[n] + v_1[n] = 1 - \sin\left(\frac{\pi}{2}n\right)}$ y $\boldsymbol{v[n] = cos\left(\frac{\pi}{2}n\right)}$.

1) Supongamos que tenemos el siguiente esquema:



Razona de forma clara hacia cuál de las siguientes soluciones tenderá el filtro de 3 coeficientes (donde en ambos casos A es una constante):

$$\boldsymbol{h1 = A\,[1, 1, 1]}$$

$$\boldsymbol{h2 = A\,[1, -1.5, 0.5]}$$

4) Supongamos que tenemos un escenario en el cuál aplicamos filtraje de Wiener con las seguintes características:

- Matriz de correlación de la señal de entrada $\boldsymbol{x[n]}$:

$$\overline{\overline{\boldsymbol{R}}}_{xx} = \boldsymbol{E}[\vec{x}\vec{x}^T] = \begin{bmatrix} \boldsymbol{1} & \boldsymbol{0.5} \\ \boldsymbol{0.5} & \boldsymbol{1} \end{bmatrix}$$

- Vector de correlación cruzada entre la señal de entrada $\boldsymbol{x[n]}$ y la respuesta deseada $\boldsymbol{y[n]}$:

$$\vec{\boldsymbol{r}}_{yx} = \boldsymbol{E}[\boldsymbol{y}[\boldsymbol{n}]\,\vec{\boldsymbol{x}}] = \begin{bmatrix} \boldsymbol{0.5} \\ \boldsymbol{0.25} \end{bmatrix}$$

- Suponemos que la señal de referencia tiene potencia 1.

Se pide:

a) Evalua los pesos del filtro de Wiener.

b) Cuál es el mínimo error cuadrático producido por este filtro de Wiener?

5) Sea un procesador de Gram Schmidt como el de la figura 1, y sean las señales de entrada reales, con potencia 1 y correlaciones cruzadas que es detallan a continuación:

$$\boldsymbol{R}_{01} = \boldsymbol{E}\big[\boldsymbol{x}_0[\boldsymbol{n}]\boldsymbol{x}_1[\boldsymbol{n}]\big] = \frac{\boldsymbol{1}}{\boldsymbol{2}}$$

$$\boldsymbol{R}_{12} = \boldsymbol{E}\big[\boldsymbol{x}_1[\boldsymbol{n}]\boldsymbol{x}_2[\boldsymbol{n}]\big] = \frac{\boldsymbol{1}}{\boldsymbol{2}}$$

$$\boldsymbol{R}_{02} = \boldsymbol{E}[\boldsymbol{x}_0[\boldsymbol{n}]\boldsymbol{x}_2[\boldsymbol{n}]] = \boldsymbol{0}$$

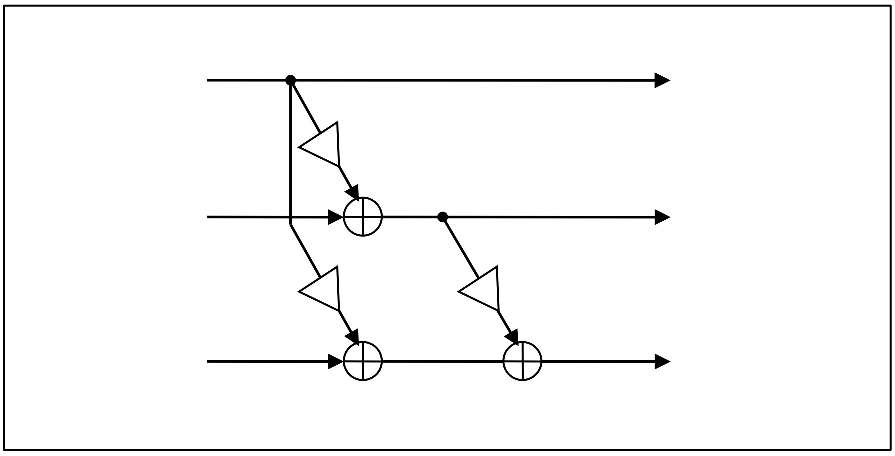


Figura 1: Procesador de Gram Schmidt

Se pide:

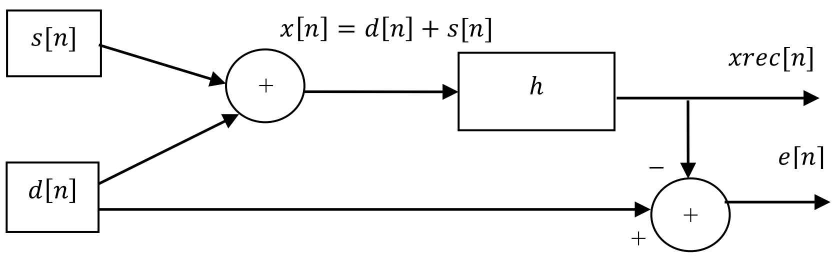
a) Explica cuál es la utilidad y para qué problemas es adecuada esta estructura.

b) Calcula el valor óptimo del parámetro a planteando la función de coste adecuada.

c) Calcula el valor óptimo de los parámetros b y c aplicando directamente el principio de ortogonalidad.

d) Propón un algoritmo para entrenar el coeficiente a de forma adaptativa y a partir de las señales implicadas.

6) Sea la estructura del ejercicio 2 de la PEC2, que por comodidad repetimos a continuación:



Se pide:

a) Rehacer el código que habías propuesto en tu PEC2 para que los valores del filtro $\boldsymbol{h}$ se calculen ahora de forma adaptativa.
b) Prueba diferentes valores para la constante de adaptación.
c) Dibuja (con Matlab) la evolución del error a medida que avanzan las iteraciones, para diferentes casos de constante de adaptación. Comenta el resultado.
d) Dibuja, para los casos anteriores, como queda la señal recuperada y compara con la señal original. Comenta el resultado.

Ejercicios resueltos

7) Sean los dos esquemas presentados en la figura 1 y en la figura 2, y que ya habíamos trabajado en la PEC2. Se pide:

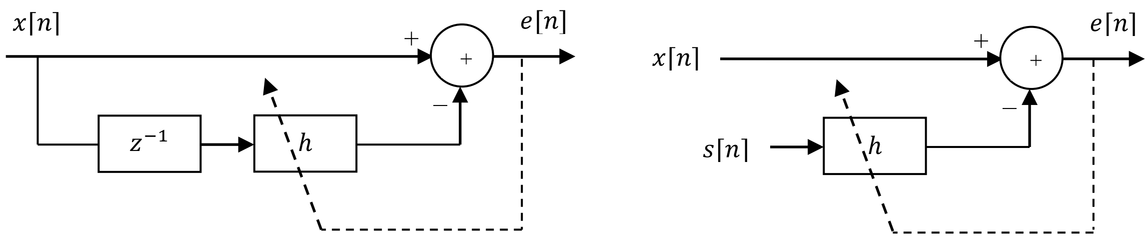


Figura 1 **Figura 2**

a) ¿Cuál es la ecuación de adaptación de los coeficientes del filtro en la versión del algoritmo LMS adaptativo para cada caso?

La ecuación de adaptación de los coeficientes para el algoritmo LMS es, en general (para señales reales):

$$\boldsymbol{h}(n+1) = \boldsymbol{h}(n) + \mu \boldsymbol{a}(n) e[n]$$

donde $\boldsymbol{a}(n)$ es el registro de los datos de entrada al filtro y $e[n]$ es la diferencia entre la señal de referencia (señal deseada) y la salida del filtro en cada instante. Por lo tanto, si particularizamos esta ecuación para cada caso, tendremos:

Caso I: $\boldsymbol{h}(n+1) = \boldsymbol{h}(n) + \mu \boldsymbol{x}(n-1) e[n]$ $\quad e[n] = x[n] - \boldsymbol{h}^{\mathrm{T}}(n)\boldsymbol{x}(n-1)$

Caso II: $\boldsymbol{h}(n+1) = \boldsymbol{h}(n) + \mu \boldsymbol{s}(n) e[n]$ $\quad e[n] = x[n] - \boldsymbol{h}^{\mathrm{T}}(n)\boldsymbol{s}(n)$

b) Comenta en cuál de los dos casos crees que la convergencia será más rápida, si se utiliza el mismo valor de la constante μ. Justifica tu respuesta.

La convergencia será, en principio, más rápida para el caso II. Esto es así porque en el segundo caso la señal de entrada está totalmente incorrelada, mientras que en el primero existe una cierta correlación. Por tanto, la función de coste presentará asimetrías del tipo elíptico que provocarán que la dirección inversa del máximo gradiente no enfoque hacia su mínimo absoluto.
Tal y como se comenta en los apuntes, "(...) cuando las señales están fuertemente correlacionadas, el algoritmo tenderá a presentar una velocidad de convergencia mucho más lenta. Esto adquiere sentido también si lo miramos en el dominio temporal, dado que cuando una señal está fuertemente correlacionada, cada muestra nueva aporta muy poca información que no esté contenida en las muestras anteriores, de modo que el algoritmo le costará más tiempo determinar en qué dirección debe modificar los coeficientes. En cambio, cuando la señal de entrada está incorrelada, cada muestra nueva aporta información que no estaba contenida en las muestras anteriores. Al haber esta aportación

de información nueva de manera mucho más clara, el algoritmo podrá evolucionar rápidamente hacia el mínimo de la función "

c) ¿A qué valor fijarías μ, para el caso de la figura 2, si se quiere reducir al máximo el tiempo de convergencia?

La condición de convergencia es que la constante μ cumpla la condición siguiente:

$$0 < \mu < \frac{2}{\lambda_{max}}$$

donde λ_{max} es el máximo autovalor de la matriz $\mathbf{R}$ (matriz de correlación).

Para el segundo caso, dado que $\mathbf{R} = \mathbf{I}$, esta matriz ya es diagonal y todos sus autovalores son iguales a 1. Por lo tanto, la condición de convergencia se traduce en:

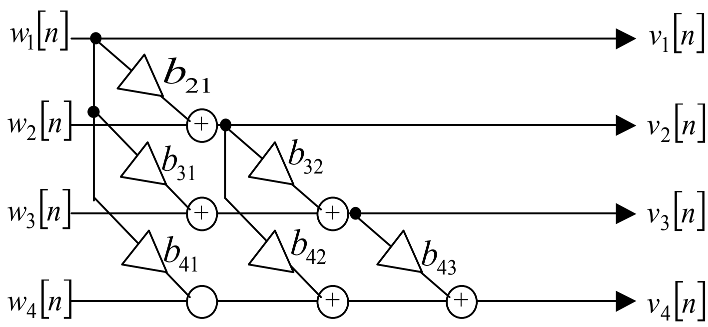
$$0 < \mu < 2$$

Por lo tanto, para acelerar al máximo la velocidad de convergencia escogemos un valor de μ próximo a 2, por ejemplo $\mu = 1.9$.

d) Explica qué utilidad tendría el hecho de utilizar un procesador de Gram-Schmidt antes de aplicar el filtro adaptativo para el caso I. Dibuja su estructura y filosofía de funcionamiento.

Un procesador de Gram Schmidt es un filtro adaptativo que decorrela variables de entrada, proporcionando a la salida señales que contienen la misma información que las primeras pero cumpliendo la propiedad de estar perfectamente incorreladas entre ellas. Si se aplicara el caso I, nos ayudaría a acelerar la convergencia del filtro adaptativo que pondríamos a continuación.

Su estructura es la siguiente:



La filosofía de funcionamiento es conseguir que las señales de salida estén incorreladas entre ellas: $E\left[v_i[n]v_j^*[n]\right] = 0, \forall i \neq j$. Los coeficientes se adaptan para conseguir minimizar la potencia de la señal de salida de cada rama, logrando de esta manera el objetivo anterior.

e) Comenta qué aportaciones hace el algoritmo RLS ante el LMS. ¿Cuál de las dos técnicas es computacionalmente más costosa? Porque? Explica el significado de la constante λ.

El algoritmo RLS se plantea a partir de una función de coste determinista, en vez de una función de coste estocástica que utiliza el planteamiento LMS.

La función de coste tiene en cuenta que el sistema a invertir puede ser variante en el tiempo, introduciendo el parámetro de memoria λ.

Este parámetro describe cuál puede ser la velocidad de variación del sistema a considerar, utilizando una memoria exponencialmente decreciente del error producido en instantes anteriores $k \leq n$ con la estimación del sistema inverso en el instante actual n y utilizando el factor de ponderación$\lambda^{(n-k)}$.

Esto hace que podamos ajustar la velocidad de adaptación del algoritmo según un criterio más adaptado al conocimiento del problema (por ejemplo, la máxima frecuencia de variación del entorno en donde hacemos la medida).

Ahora bien, el algoritmo RLS es más costoso computacionalmente ya que requiere, además de adaptar la respuesta impulsional del filtro lineal, adaptar también la estimación de la matriz inversa de correlaciones deterministas. En esta segunda ecuación intervienen matrices de orden N igual al orden del filtro lineal, por lo que la complejidad del algoritmo pasa de ser de orden N en orden N^2

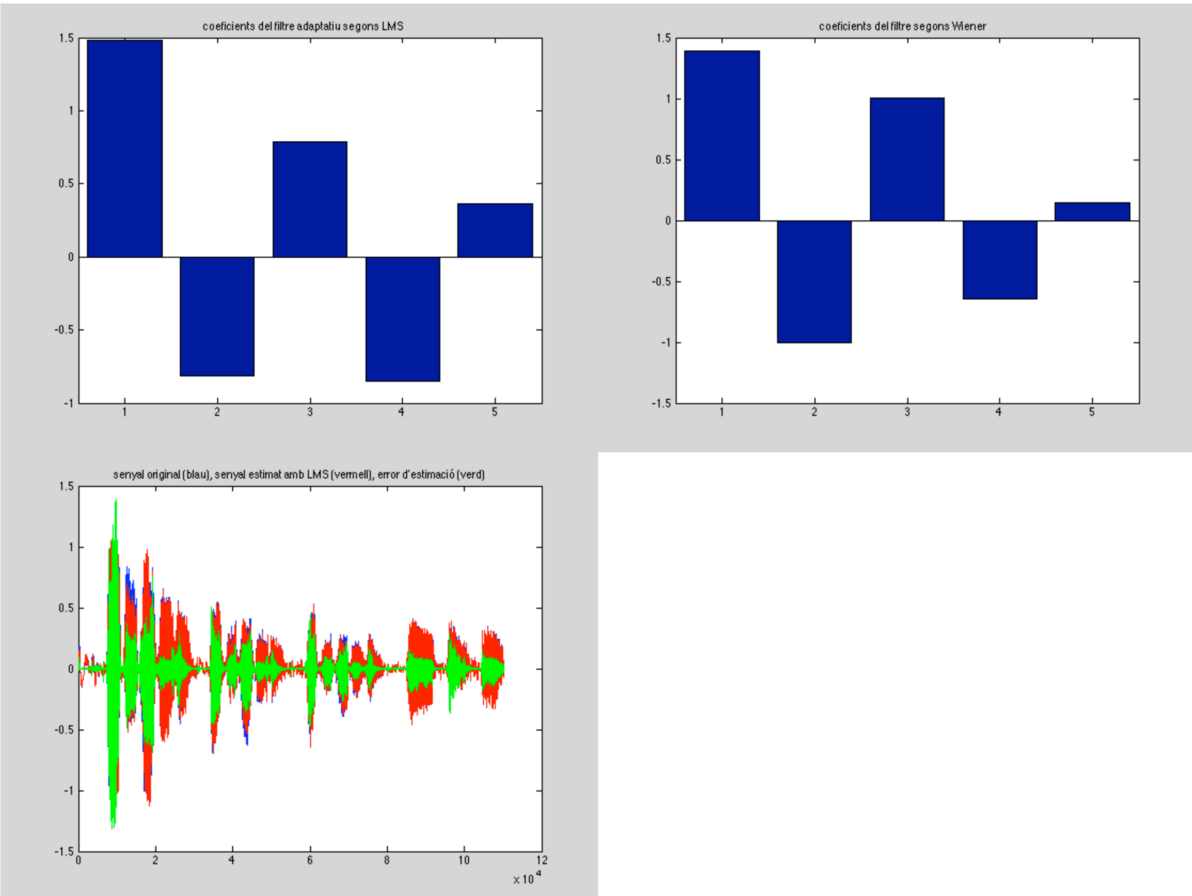
8) Sea el Código Matlab 'provaLPC.m' que implementa un filtro LMS para un caso de predicción lineal de la señal de voz.

Se pide:

a) Ejecuta el código tal y como está y comenta el resultado obtenido.

El código determina, mediante el algoritmo LMS, los coeficientes del filtro óptimo para implementar el algoritmo LPC sobre la señal de voz 'vocals.wav'. Se trata de hacer una predicción lineal de orden p (inicialmente p = 5) de la muestra n de la señal de voz a partir de las muestras anteriores.

Observamos que el sistema converge rápidamente. La figura 1 muestra los coeficientes del filtro obtenidos, la figura 2 la señal original y la señal obtenida con el predictor y el error, y la figura 3 los coeficientes del filtro óptimo obtenido a partir de las ecuaciones de Wiener (implementado con la función LPC de Matlab).



Nótese que los coeficientes obtenidos con el sistema adaptativo se aproximan mucho a los óptimos obtenidos por Wiener:

Filtro adaptativo	Filtro óptimo
1.4799	1.3862
-0.8165	-1.0006
0.7906	1.0012
-0.8516	-0.6483
0.3657	0.1425

b) Modifica el parámetro μ y explica qué sucede en cada caso (prueba valores superiores e inferiores a lo que hay ahora).

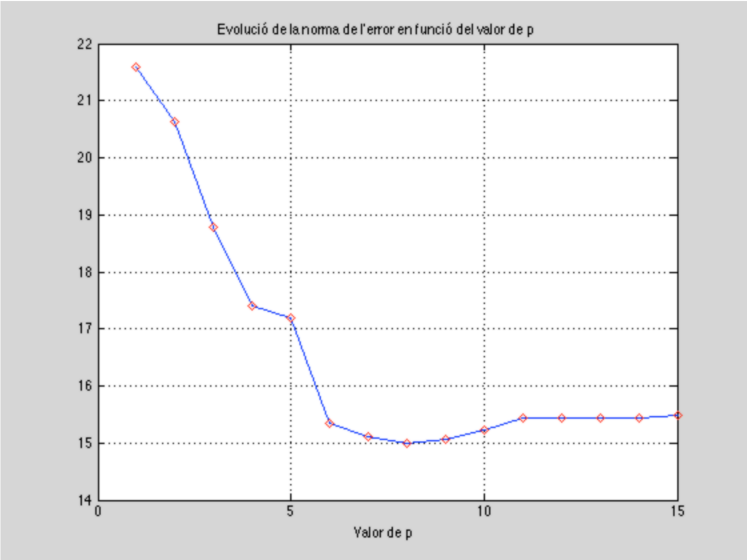
Resumimos los resultados en la tabla siguiente:

$\mu = 0.01$	$\mu = 0.05$	$\mu = 0.1$	$\mu = 0.5$	$\mu = 1$
1.0314	1.3641	1.4799	1.6123	-0.3536×10^{87}
-0.2261	-0.6671	-0.8165	-0.9606	0.8859×10^{87}
0.2655	0.6731	0.7906	0.8511	-1.0702×10^{87}
-0.1259	-0.6796	-0.8516	-0.9953	0.8828×10^{87}
-0.1303	0.2416	0.3657	0.4798	-0.3484×10^{87}
NE = 20.5162	NE= 17.6089	NE = 17.1883	NE = 20.8110	NE = 4.2102×10^{87}

Observamos que para valores pequeños de μ el sistema no llega a converger y el error (NE es la norma del error) es mayor que para μ = 0.1. Para valores grandes del error también crece (el sistema tiende a ser inestable) y para valores demasiado grandes el sistema diverge y el error se va a infinito.

c) Para un valor de $\mu = 0.1$, varía el parámetro p de 1 a 15 y explica qué sucede en cada caso. Representa gráficamente cómo evoluciona la norma del error de predicción. ¿Cuál es el valor óptimo de p?

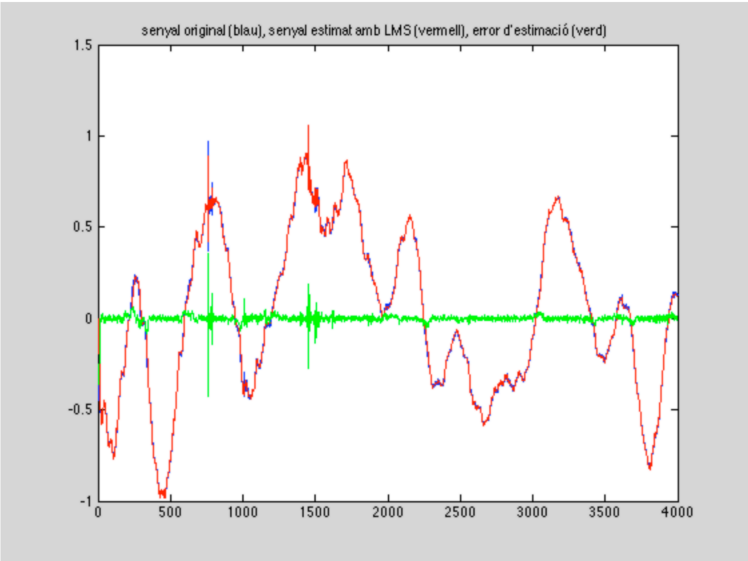
Cambiando el valor de p estamos cambiando el número de coeficientes del filtro. Haciendo el barrido de 1 a 15 obtenemos la siguiente gráfica de la evolución del error:



El valor mínimo se obtiene para $p = 8$ y da un error de 14.9882. Para valores pequeños de p el error es claramente mayor, va disminuyendo a medida que p se acerca al valor óptimo y finalmente termina estabilizando por encima del valor mínimo pero sin crecer demasiado.
Esto nos dice que pocos coeficientes no consiguen hacer una buena estimación, pero más coeficientes del filtro de los necesarios tampoco aportan nada al sistema y de hecho, como los coeficientes sobrantes que en teoría deberían ser cero y no lo son, terminan haciendo que el error aumente un poquito (aunque de forma muy suave).

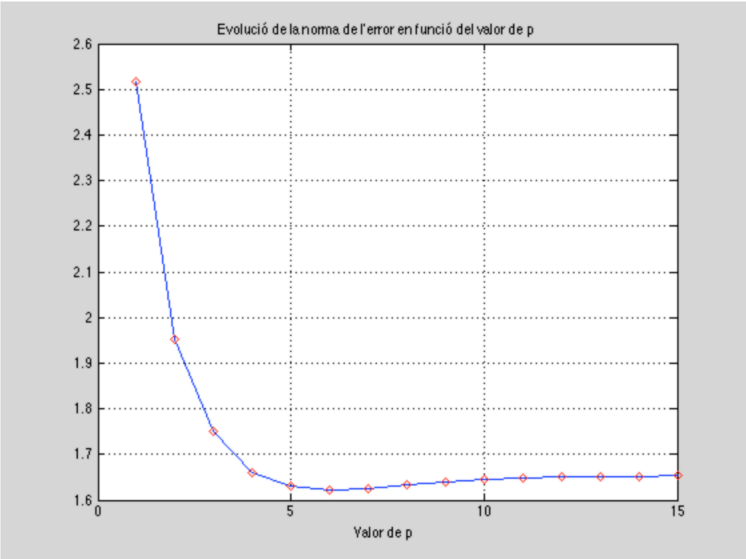
d) Define un valor de $inicio = 30000$ (línea 7) y $longitud = 4000$ (línea 8). Fija $\mu = 0.1$. Comenta el resultado obtenido cuando p varía entre 1 y 15..

La forma de onda de la señal es la que se muestra en la figura siguiente:



Observamos que la señal se predice muy bien y el error (curva verde) es pequeño. La señal es bastante periódica y por lo tanto tiene una correlación más o menos elevada, y evoluciona muy lentamente (el espectro tendrá básicamente frecuencias bajas), lo que permite que el predictor funcione muy bien.

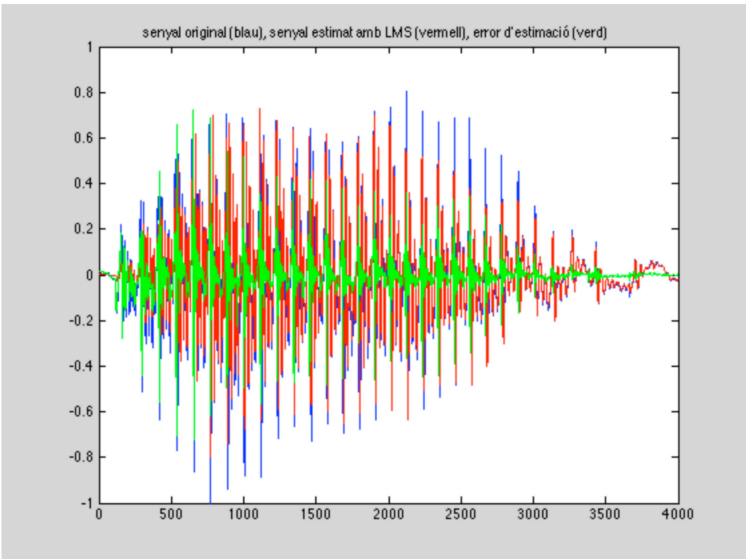
La curva de la evolución del error en función de p se muestra a continuación:



El valor óptimo está ahora en $p = 6$ y la norma mínima es 1.6232.

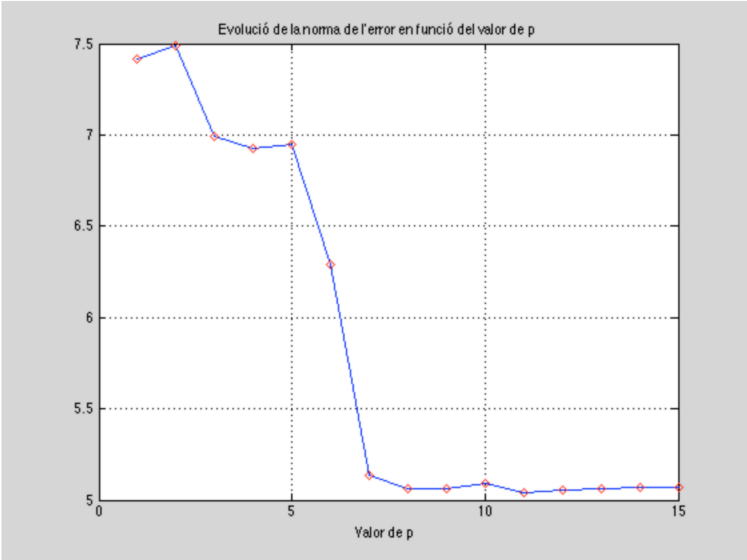
e) Define un valor de $inicio = 34000$ (línea 7) y $longitud = 4000$ (línea 8). Fija $\mu = 0.1$. Comenta el resultado obtenido cuando p varía entre 1 y 15 .

La forma de onda de la señal es la que se muestra en la figura siguiente:



La señal, aunque es de la misma longitud que el caso anterior, se mueve ahora mucho más y tiene muchos más cambios bruscos. Observamos que la señal se predice peor que antes y el error (curva verde) es mayor. Intuimos, pues, que ahora las cosas no serán tan fáciles.

La curva de la evolución del error en función de p se muestra a continuación:



Podemos ver ahora que el error de partida es mayor que antes, y se mantiene elevado hasta llegar a $p = 7$. A partir de ahí, disminuye un poco alcanzando el valor mínimo en $p = 11$, pero ya casi no cambia. Este valor mínimo del error, que ahora es de 5.0374, es bastante más grande que antes.

f) ¿Cómo explicarías las diferencias que observas entre los dos últimos apartados anteriores?

En los dos casos anteriores hemos analizado un segmento de señal de voz de la misma longitud, pero de características diferentes. En el primer caso se trataba de una parte de transición entre dos vocales, señal que evoluciona lentamente en el tiempo. En el segundo caso, se trataba de una parte inicial de una vocal (la vocal /a/ locutada en inglés), forma de onda más compleja y por lo tanto más difícil de predecir.

g) Imagínate ahora que de toda la señal, usas sólo una parte para calcular los coeficientes del filtro predictor mediante el algoritmo LMS, y luego utilizas estos coeficientes para predecir el resto de la señal. ¿Qué crees que pasará?

Como ahora el filtro lo dejaremos fijado una vez estimado con la primera parte de la señal, la estimación que haremos del resto de la señal con este filtro no podrá ser óptima porqué las características de la señal de voz varían a lo largo del tiempo (no es una señal estacionaria). Ya hemos visto en los apartados anteriores que las cosas iban diferente en función de las características de la señal. Por tanto, aunque la estimación puede ser más o menos buena (dependerá de la señal utilizada en la parte de obtención del filtro), nunca será tan buena como podría ser.