



Casa abierta al tiempo

**UNIVERSIDAD AUTONOMA METROPOLITANA**

**ALGORITMOS LMS  
DE PASO VARIABLE**

**TESIS QUE PRESENTA EL**

**M. en C. FAUSTO CASCO SANCHEZ**

**PARA LA OBTENCION DEL GRADO DE  
DOCTOR EN CIENCIAS EN  
INGENIERIA ELECTRICA**

**SEPTIEMBRE 1997**

**UNIVERSIDAD AUTONOMA METROPOLITANA-  
IZTAPALAPA**

**DIVISION DE CIENCIAS BASICAS E INGENIERIA**

**UNIDAD IZTAPALAPA**

Av. Michoacán y La Purísima, Col. Vicentina, 09340 México, D.F. Tel.: 724-4600 TELEFAX: (5) 612 0885

## AGRADECIMIENTOS

Mi mas sincero agradecimiento :

Por su gran motivación, dirección y asesoría al  
Dr. Héctor Pérez Meana

Por su desinteresada participación en el diseño del programa al

M.en C. Ricardo Marcelín Jiménez

y

M.en C. Mariko Nakano Miyatake

Por su apoyo para ingresar al Doctorado al

Dr. Mario Vizcarra Mendoza

y

Dr. Juan José Ambriz García

Por su profesional participación en el jurado a los sinodales

Dra. Verónica Medina Bañuelos

Dr. José Luis Marroquín Zaleta

Dr. Guillermo Espinosa Flores Verdad

Dr. José Silva Martínez

Y a mis compañeros del Departamento de Ingeniería  
Eléctrica que contribuyeron en alguna forma por su  
valiosa cooperación

## INDICE

Lista de Símbolos	1
-------------------	---

### I.-INTRODUCCION

1.1.-Procesamiento Adaptable de Señales	3
1.2.-Desarrollo del Proyecto de Investigación	3
1.3.-Organización de la tesis	5
1.4.-Referencias	6

### II.-ECOS EN TELEFONIA

2.1.-Convertidor de 2 a 4 hilos	8
2.2.-Generación del eco	9
2.3.-Supresor de eco	10
2.4.-Canceladores de eco	11
2.5.-Términos "Single talk" y "Double talk"	12
2.6.-Configuración en un Cancelador de eco	13
2.7.-Referencias	14

### III.-ESTRUCTURAS PARA FILTRADO ADAPTABLE

3.1.-Concepto básico de un Filtro Adaptable	15
3.2.-Estructuras de Filtros Digitales	16
3.3.-El Filtro Transversal FIR	20
3.4.-El Filtro IIR	22
3.5.-Conclusiones	23
3.6.-Referencias	24

### IV.-ALGORITMOS PARA FILTRADO ADAPTABLE

4.1.-Algoritmo LMS	25
4.2.-Complejidad Computacional del Algoritmo LMS	30
4.3.-Algoritmo RLS	31
4.4.-Complejidad computacional del Algoritmo RLS	33
4.5.-El Algoritmo LMS Normalizado(NLMS)	34
4.6.-Efecto del factor de Convergencia $\alpha$ en el Nivel de Cancelación . del Algoritmo	38
4.7.-Desventajas del Algoritmo LMS	41
4.8.-Algoritmo de paso variable VSLMS	42
4.9.-Algoritmo VSS	44

4.10.-Algoritmo en Bloque(FLMS)	45
4.11.-Conclusiones	49
4.12.-Referencias	50

## **V.-ALGORITMOS NLMS DE PASO VARIABLE PROPUESTOS VSS.Doble paso y VSS-AC**

5.1.1.-Algoritmo VSS-Doble paso	53
5.1.2.-Resultados de la simulación del Algoritmo VSS-Doble paso	55
5.1.3.-Algoritmo VSS-AC	63
5.2.2.-Resultados de la simulación del Algoritmo VSS-AC	65
5.3.-Conclusiones	70
5.4.-Referencias	71

## **VI.-ALGORITMO NLMS DE PASO VARIABLE PROPUESTO VSS-CC**

6.1.-Algoritmo VSS-CC	72
6.2.-Comparación del Algoritmo VSS-CC con otros Algoritmos de paso variable	76
6.3.-Resultados de la simulación del Algoritmo VSS-CC	78
6.4.-Comparación del Algoritmo VSS-CC con los algoritmos VSS-Doble paso y VSS-AC	86
6.5.-Conclusiones	88
6.6.-Referencias	89

## **VII.-PERSPECTIVAS PARA CONTINUAR EN ESTA LINEA DE INVESTIGACION**

7.1.-Extensión del Algoritmo de paso variable VSS-CC al algoritmo en bloque FLMS	90
7.2.-Otras aplicaciones del algoritmo VSS-CC	91
7.2.1.-Identificación de Sistemas	91
7.2.2.-Cancelación de Interferencia Intersímbolo en Transmisión Digital	92
7.3.-Conclusiones	94
7.4.-Referencias	96

<b>APENDICE I.-Factor de olvido y memoria usados en este proyecto de investigación</b>	97
--	----

<b>BIBLIOGRAFIA</b>	99
---------------------	----

## Lista de Símbolos

$a_i$ :	coeficiente de un filtro digital no recursivo
$\alpha_1$ :	factor de convergencia grande en el algoritmo VSS-Doble paso
$\alpha_2$ :	factor de convergencia pequeño en el algoritmo VSS-Doble paso
$\alpha_{max}$ :	factor de convergencia máximo en el algoritmo VSS-CC
$\alpha_{min}$ :	factor de convergencia mínimo en el algoritmo VSS-CC
$b_i$ :	coeficiente de un filtro digital recursivo
$\beta$ :	factor de convergencia en el algoritmo LMS
$\mu$ :	factor de convergencia en el algoritmo LMS [ $\beta=2\mu$ ]
$d$ :	señal deseada a aproximar en un filtro adaptable
$E[x]$ :	valor esperado de $x$
ERLE:	figura de mérito
$e$ :	error residual en un filtro adaptable
$e'$ :	error residual en un filtro adaptable con estructura de polos y ceros
$e_1$ :	error de identificación [ $y(n)-\hat{y}(n)$ ]
$\varepsilon$ :	inverso de la memoria en el factor de olvido $\lambda_1$
$\xi$ :	vector que representa en cada iteración el valor esperado del vector de coeficientes $w$ del filtro adaptable
FFT:	algoritmo de la transformada rápida de fourier
$G_{dx}$ :	espectro de Potencia de la correlación entre las señales $d$ y $x$ .
$h(n)$ :	coeficientes de la trayectoria de eco en la iteración $n$
$H$ :	vector de coeficientes de la trayectoria de eco.
$K$ :	ganancia de Kalman
$K_L$ :	matriz Karhunen-Loeve
$L$ :	factor de convergencia en el algoritmo de Harris [29]
$\Phi$ :	matriz diagonal
$\lambda_{ii}$ :	elementos de la matriz diagonal $\Phi$
$\lambda$ :	factor de olvido en el algoritmo RLS
$\lambda_1$ :	factor de olvido en el algoritmo VSS-CC [101]
$\lambda_{max}$ :	máximo valor propio de la matriz de autocorrelación $R$
$M$ :	matriz de convergencia en el algoritmo de Harris [29]
MSE:	error cuadrático medio
$N$ :	orden del filtro transversal
$\nabla$ :	vector gradiente
$P$ :	vector del valor esperado de la correlación entre el vector de entrada $X$ y la señal deseada $d$
$Q$ :	matriz inversa de la matriz de autocorrelación $R$
$R$ :	matriz de autocorrelación
$r_{dx}$ :	correlación entre la señales $d$ y $x$
$r$ :	ruido aditivo

S/R: relación señal a ruido  
 TR(R): traza de la matriz de autocorrelación R  
 $V_0$ : componente ortogonal al vector de entrada X  
 $V_p$ : componente paralela al vector de entrada X  
 V: vector de coeficientes de un filtro adaptable rotado y trasladado  $K_L[W-W^*]$   
 $W_1$ : función de transferencia de un filtro digital  
 W(n): vector de coeficientes de un filtro adaptable en la iteración n  
 $W^*$ : vector de coeficientes óptimo  
 $w_k$ : k-ésimo coeficiente de un filtro adaptable FIR  
 $x_1$ : entrada a un filtro digital  
 x: entrada a un filtro adaptable  
 $\overline{x^2(n)}$ : valor cuadrático de x  
 X: vector de entrada a un filtro adaptable  
 $X_1$ : transformada z de la entrada  $x_1$   
 $y_1$ : salida de un filtro digital  
 y: salida de la trayectoria de eco  
 $\hat{y}$ : salida de un filtro adaptable  
 $Y_1$ : transformada z de la salida  $y_1$

## I.-INTRODUCCION

El presente proyecto de investigación está orientado hacia el Procesamiento Adaptable de Señales, en particular el trabajo tiene aplicación en cancelación de ecos en telefonía.

### 1.1.-Procesamiento Adaptable de Señales

En el Procesamiento Adaptable de Señales ha habido un progreso notable en los últimos años, esto es consecuencia del gran desarrollo tecnológico en la industria de los semiconductores, integración en gran y muy alta escala. Resultados teóricos del Procesamiento Adaptable de Señales, han sido llevados a la práctica en diversas aplicaciones específicas tal es el caso de los canceladores de eco, ecualizadores, codificadores en las comunicaciones, canceladores de ruido en señales biomédicas, identificación de sistemas en teoría del control, etc.

Una aplicación típica en el Procesamiento Adaptable de Señales, es la realización de filtros digitales, ya sea por medio de procesadores de propósito particular, ó bien a través de un programa que simula su comportamiento.

### 1.2.-Desarrollo del Proyecto de Investigación

El algoritmo LMS (Least Mean Squares) ha sido usado en diversos problemas, en particular éste algoritmo usa un paso constante  $\alpha$ . La selección del paso  $\alpha$  refleja un compromiso entre el error de salida en el algoritmo y la rapidez de adaptación; es decir si se selecciona una  $\alpha$  grande, se tiene una convergencia rápida del algoritmo, sin embargo el error de salida es grande; ahora bien si se usa una  $\alpha$  pequeña, la rapidez de convergencia disminuye, pero el error de salida es pequeño. Por lo anterior éste proyecto de investigación doctoral inició con la idea de trabajar con el algoritmo NLMS (LMS normalizado), pero usando dos pasos diferentes.

Los primeros resultados obtenidos dieron lugar al artículo en conferencia internacional "A two Step Size Algorithm" [1.6]. Este primer algoritmo usa dos pasos,  $\alpha_1$  (paso grande) y  $\alpha_2$  (paso pequeño); el ajuste del paso es controlado por el

cuadrado del error de salida  $e(n)$ . Al inicio se tiene un error de salida  $e(n)$  grande, lo cual hace que el paso a usar en el algoritmo sea grande ( $\alpha_1$ ), así se tiene una velocidad de convergencia rápida. Cuando el error de salida  $e(n)$  ha disminuido, y es igual a un umbral  $c_1$ , el algoritmo usa el paso pequeño ( $\alpha_2$ ), haciendo que el error de salida  $e(n)$  sea menor. El umbral de decisión  $c_1$  para seleccionar entre la  $\alpha_1$  y la  $\alpha_2$ , es igual al valor que toma  $\alpha_2$ . De esta forma se tiene una velocidad de convergencia rápida, con un error de salida  $e(n)$  pequeño. Este primer algoritmo propuesto es el algoritmo VSS-Doble paso descrito en el capítulo V.

Posteriormente se presentó un artículo en conferencia internacional "A Two Step Size NLMS Adaptive Filter Algorithm" [1.8]. En este algoritmo también se usan dos alfas diferentes (como en el algoritmo VSS-Doble paso); la diferencia consiste en que este algoritmo busca el umbral para cambiar entre  $\alpha_1$  y  $\alpha_2$ . Cuando hay un error de salida  $e(n)$  grande, el algoritmo usa el paso grande  $\alpha_1$ , para obtener un rastreo rápido; cuando el error de salida  $e(n)$  es igual al umbral  $c$ , el algoritmo usa el paso pequeño  $\alpha_2$ , dando lugar a una disminución en el error de salida  $e(n)$ . El umbral  $c$  es obtenido por la correlación entre la señal deseada  $d(n)$  y el error de salida  $e(n)$  del filtro adaptable, de esta manera cuando el umbral  $c$  es igual a la  $\alpha_2$ , el algoritmo comienza a trabajar con  $\alpha_2$ . De esta manera se disminuye el error de salida  $e(n)$  y se aumenta la velocidad de convergencia. Este segundo algoritmo propuesto es el algoritmo VSS-AC descrito también en el capítulo V.

Finalmente se publicó el artículo "A Variable Step Size (VSS-CC) NLMS Algorithm" [1.9]. Este algoritmo usa un paso variable, el cual se obtiene en función de la correlación entre el error de salida  $e(n)$  y la salida del filtro adaptable  $\hat{y}(n)$ . Al inicio  $e(n)$  y  $\hat{y}(n)$  están correlacionados, así el algoritmo usa una  $\alpha_{MAX}$  (paso grande) y se tiene una velocidad de convergencia rápida. Cuando el algoritmo converge, la correlación produce valores de  $\alpha$  pequeños hasta una  $\alpha_{MIN}$  (paso más pequeño), así se tiene una disminución en el error de salida  $e(n)$ . La particularidad de este algoritmo es precisamente que si la relación S/R presente en el filtro es pequeña, entonces el algoritmo continúa obteniendo un error de salida  $e(n)$  aceptable.

Cabe mencionar que el algoritmo VSS-CC no solo se comparó con el tradicional algoritmo normalizado NLMS, sino que también fue comparado con el algoritmo VSS [1.1], el cual usa un paso variable. El algoritmo VSS disminuye el error de salida  $e(n)$  del filtro adaptable con una velocidad de convergencia rápida, el



problema con el algoritmo VSS es que si la relación S/R es pequeña el algoritmo ya no obtiene un error de salida  $e(n)$  aceptable. El algoritmo propuesto VSS-CC se describe en el capítulo VI.

### **1.3.-Organización de la tesis**

En el capítulo I se describe al algoritmo LMS, el cual usa un solo paso, esto da lugar a un compromiso entre el error de salida y la rapidez de adaptación. Lo anterior dió lugar a la elaboración de algoritmos LMS que mejoraran dicho compromiso, que es el objetivo de este trabajo.

En el capítulo II se trata el problema de los ecos en telefonía a larga distancia, describiendo como se generan tales ecos y la forma primitiva en la que se suprimían los mismos. Cabe mencionar que este trabajo doctoral está enfocado a la cancelación de ecos en telefonía.

En el capítulo III se habla de las estructuras usadas en filtrado adaptable, esto es filtros no recursivos (FIR) y filtros recursivos (IIR).

En el capítulo IV se abordan los algoritmos de filtrado adaptable existentes al inicio de este trabajo doctoral (estado del arte), se describen los algoritmos RLS (Recursive Least Squares) [1.2], LMS (Least Mean Squares) [1.3], NLMS [1.4] (LMS normalizado), VSLMS (Variable Step Least Mean Squares) [1.5] y VSS (Variable Step Size) [1.1].

En el capítulo V se describen los algoritmos VSS-Doble paso y VSS-AC relacionados con las publicaciones "A two Step Size Algorithm"[1.6] y "A Two Step Size NLMS Adaptive Filter Algorithm" [1.8] respectivamente. Se comparan resultados contra el algoritmo NLMS, se obtiene un mejor resultado con los algoritmos propuestos.

En el capítulo VI se muestra al algoritmo VSS-CC (Variable Step Size with a Correlation Criterion) relacionado con la publicación "A Variable Step Size (VSS-CC) NLMS Algorithm", dicho algoritmo se compara contra los algoritmos NLMS [1.4] y VSS [1.1], obteniendo mejores resultados con el algoritmo propuesto. Las

señales usadas para la prueba de los algoritmos fueron señales de voz como señal de entrada y ruido blanco como ruido aditivo.

En el capítulo VII se habla de las perspectivas para continuar en esta línea de investigación, con posibles extensiones hacia filtros adaptables del tipo LMS en bloque (FLMS).

También se trata con la posible aplicación del algoritmo VSS-CC [1.9] en la identificación de sistemas e interferencia intersímbolo. Finalmente en el apéndice I se describe el concepto de factor de olvido y memoria usados en los algoritmos propuestos en este proyecto.

#### **1.4.-Referencias**

[1.1] Raymond ,H., Kwong,E., and Johnston,W.,"A Variable Step Size LMS Algorithm ",IEEE Transactions on Signal Processing,vol.40 No.7 pp. 1633-1642,July 1992.

[1.2] Carayannis,G.,Manolakis,D., and Kalopstidis, N.,"Fast Kalman Type Algorithms for Sequential Signal Processing,"Proc 1983 IEEE Internat.Conf Acoust.Speech, Signal Processing,pp 186-189.

[1.3] Solo,V., "The limiting behavior of LMS", IEEE Trans. Acoust. Speech Signal Process., vol. 37, pp. 1909-1922, 1989.

[1.4] Bershad, N.J,"Analysis of the normalized LMS algorithm with Gaussian inputs",IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-34, pp 793-806, 1986.

[1.5] Harris,R.W., Chabries,D.M.,and Bishop,F.A.,"A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316,Apr.1986.

#### **Publicaciones realizadas**

[1.6] Casco,F., Perez,H., "A Two Step Size LMS Algorithm", Proceedings of the International Conference on Signal Processing, vol 1, pp 543-546, October 1993, Beijing China.

[1.7] Casco,F.,Pérez,H.,López,M., "A three step size LMS algorithm", Proceedings of the 1993 IEICE FALL CONFERENCE, pp 1-84, 1-85 September 5 to 8 1993, Sapporo, Hokkaido Institute of Technology.

[1.8] Casco,F.,Pérez,H.,López,M.,and Marcelín,R., " A two Step Size NLMS Adaptive Filter Algorithm", IEEE Singapore International Conference on Communication Systems. ICCS'94, 14-18 November 1994, pp 814-819.

[1.9] Casco,F.,Pérez,H.,Nakano,M., and López,M., "A Variable Step Size (VSS-CC) NLMS Algorithm", IEICE Trans.Fundamentals, vol.E78-A, No 8 August, 1995, pp 1004-1009

## II.-ECOS EN TELEFONIA

En esta parte se describen los convertidores de 2 a 4 hilos, los cuales son usados en telefonía; cuando en éstos convertidores existe un desbalance de impedancias entre los circuitos de 2 y 4 hilos, entonces se genera el eco. Hay dos formas de resolver este problema, la primera de ellas es a través de supresores de eco, y la segunda con canceladores de eco, las cuales son descritas a continuación.

### 2.1.-Convertidor de 2 a 4 hilos

En la transmisión telefónica a larga distancia [2.1,2.6] es necesario usar convertidores de 2 a 4 hilos, uno en cada extremo, capaz de convertir el circuito de 2 hilos a uno de 4 hilos y viceversa (figura 2.1).

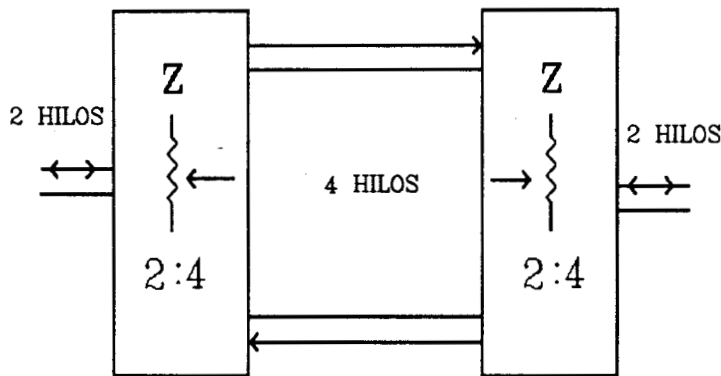


Figura 2.1.-Uso de un convertidor de 2 a 4 hilos.

Idealmente un convertidor de 2 a 4 hilos [2.11], debe hacer que toda la energía entrante del circuito a 4 hilos, pase al circuito de 2 hilos; y nada de esta energía entrante debe regresar por la otra rama del circuito a 4 hilos. Si la impedancia del circuito a 4 hilos se acopla al de 2 hilos, el convertidor de 2 a 4 hilos realiza su función correctamente.

## 2.2.-Generación del Eco

El eco ocurre en un convertidor de 2 a 4 hilos cuando parte de la señal transmitida regresa al transmisor, debido a un desbalance en las impedancias entre los circuitos de 2 y 4 hilos, esto se muestra en la figura 2.2.

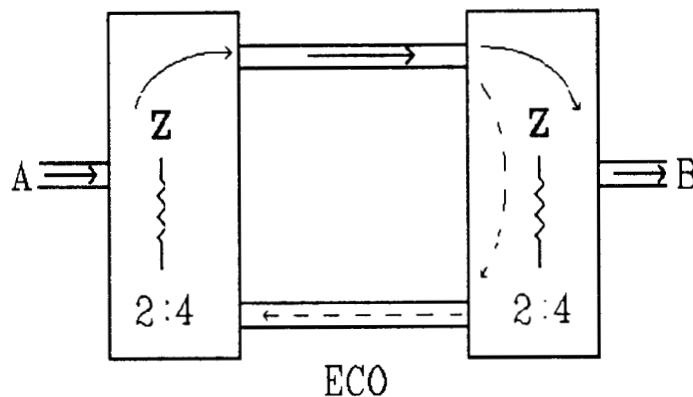


Figura 2.2.-Generación de eco en convertidores de 2 a 4 hilos.

El desbalance de impedancias en el convertidor de 2 a 4 hilos, hace que parte de la señal proveniente del extremo A, retorne a la fuente de origen. Es impracticable en un convertidor de 2 a 4 hilos, prever una impedancia que se adapte a todas las posibles conexiones en una red de conmutación, ya que esto involucra diferentes tipos de líneas.

En la figura 2.2, al primer reflejo que ocurre se le conoce como eco del que habla (A); si un segundo reflejo ocurre resulta un eco del que oye (B); cuando la señal que se refleja, lo hace repetidamente, esto da lugar a oscilaciones ("singing") [2.11]; esta oscilación ("singing") tiene lugar cuando la ganancia del lazo en alguna frecuencia es mayor que la unidad. Si la ganancia del lazo es menor que la unidad la oscilación ("singing") se amortigua. De los tres tipos de reflejo mencionados anteriormente el eco del que habla es el que más perturba la comunicación.

El grado del eco escuchado por el que habla depende de la magnitud de la señal reflejada y la cantidad de retardo involucrado. En comunicaciones cortas el retardo es pequeño, así que el eco no es percibido; sin embargo en larga distancia se requiere significativa atenuación para minimizar el eco, por ejemplo en enlaces con

mas de 45 milisegundos de retardo, es necesario atenuar el eco, esto se logra con 2 tipos de dispositivos, esto es con supresores de eco ó bien con canceladores de eco.

### 2.3.-Supresor de Eco

Cuando en los enlaces a 4 hilos, los retardos son grandes, se usa un supresor de eco [2.2] (un supresor de eco era la forma antigua de disminuir ecos). La forma adecuada de usar al supresor de eco, es localizarlo tan cerca como sea posible de la fuente de eco, y operarlo con las señales procedentes del extremo remoto (figura 2.3).

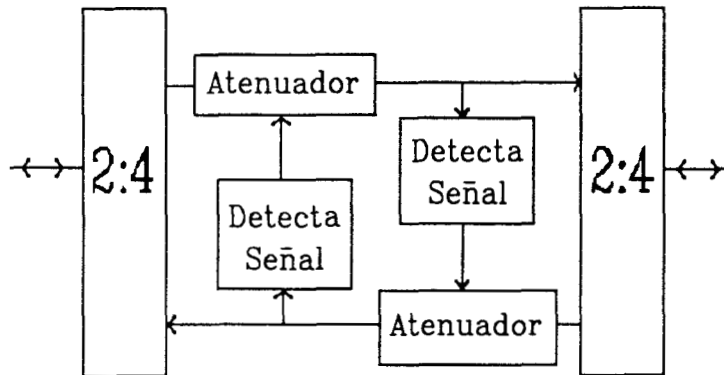


Figura 2.3.- Supresor de Eco.

En la figura 2.3 hay un supresor de eco en cada extremo; cuando éstos detectan señal de voz que llega del extremo distante, su diseño permite insertar atenuación en la trayectoria de retorno, disminuyendo así el eco que se produce.

Una desventaja del supresor de eco, es que introduce su propia forma de distorsión, es decir cuando ambas partes hablan al mismo tiempo, no solo se atenúa el eco, sino también la voz a ser escuchada.

## 2.4.-Canceladores de eco

Como se muestra en la figura 2.4 un cancelador de eco simula la trayectoria de eco [2.3-2.5,2.11], para substraer una copia propiamente atenuada y retardada de la señal transmitida, así se cancela el eco generado en el extremo receptor. De esta forma un cancelador de eco requiere de un tiempo, para determinar cuanto retardo y atenuación se requiere en la simulación de las características del eco. En la figura 2.4 los ecos son cancelados cerca de la fuente donde se generan, así los retardos son minimizados. Una característica importante de un cancelador de eco es que permite comunicación simultánea en ambos sentidos.

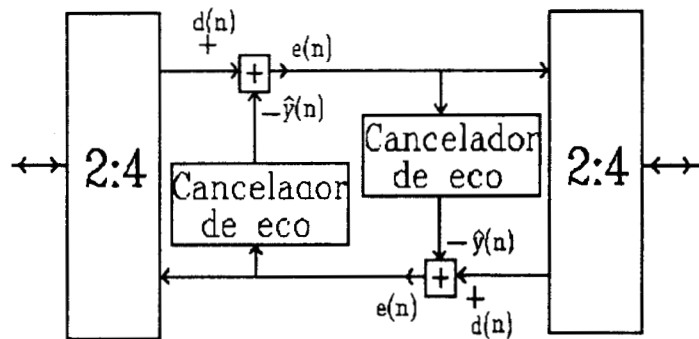


Figura 2.4.- Cancelador de eco.

En la figura 2.4 hay un cancelador de eco cercano a cada convertidor de 2 a 4 hilos. La señal proveniente de cada transmisor es la entrada de cada cancelador, y cada cancelador es ajustado de tal forma que su salida  $\hat{y}(n)$ , se resta a la porción de la señal  $d(n)$  que retorna a través del convertidor de 2 a 4 hilos; haciendo esto que el error  $e(n)$  sea pequeño.

En general el procedimiento usado para controlar ecos, también controla la oscilación ("singing"). Se invita al lector a acudir a la literatura referida por [2.6-2.10] para cancelación de eco en voz.

## 2.5.-Términos "Single talk" y "Double talk".

En telefonía hay dos formas de realizarse la comunicación, la primera de ellas es cuando la persona A habla y la persona B escucha y viceversa, esto equivale a que se efectúe la transmisión en un sentido solamente; a ésta forma de comunicación se le conoce como "single talk". La segunda ocurre cuando por alguna razón ambas personas A y B hablan al mismo tiempo, efectuando la comunicación de manera simultánea, a ésta forma de comunicación se le conoce como "double talk". En la figura 2.5 se ilustran las dos formas de comunicación.

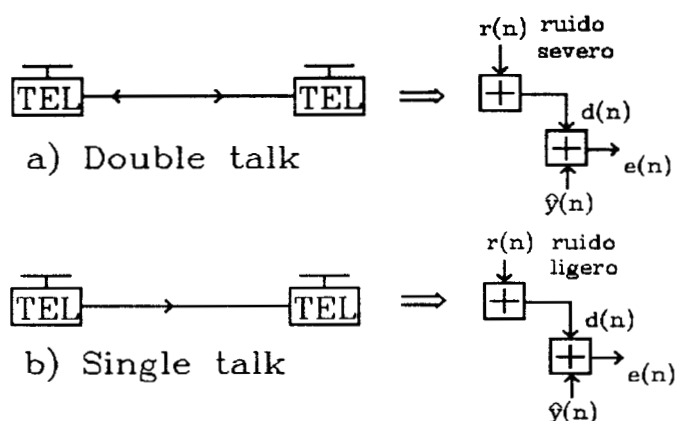


Figura 2.5.-Transmisión de voz :(a) en dos sentidos, (b) en un sentido.

En el caso de la comunicación en un sentido "single talk" se tiene presente en el canal de comunicación un ruido aditivo  $r(n)$  de nivel bajo, puede decirse que es un ruido de fondo; mientras que en el caso de la comunicación simultánea "double talk" el ruido aditivo  $r(n)$  es grande, provocando ello que la comunicación se vea afectada notablemente, siendo esto una molestia para los subscriptores que ocupan el servicio telefónico bajo ésta condición. En base en lo anterior un cancelador de eco debe eliminar el eco para ambas situaciones "single talk" y "double talk". Además debe ser capaz de discriminar entre la señal de eco y la voz del interlocutor cercano, de otra forma el algoritmo haría modificaciones infructuosas en un esfuerzo inútil por cancelar la voz del interlocutor cercano. Un cancelador desprotegido podría operar en forma deficiente en situaciones de "double talk".



## 2.6.-Configuración de un Cancelador de eco

En la figura 2.6  $x(n)$  es la entrada al cancelador de eco,  $h(n)$  es la respuesta a impulso del convertidor de 2 a 4 hilos, que permite que parte de  $x(n)$  se retroalimente (ver figura 2.5). La respuesta a impulso del cancelador de eco es  $w(n)$ ,  $y(n)$  es el eco generado y  $\hat{y}(n)$  es la salida del cancelador de eco. El ruido aditivo es  $r(n)$ ,  $d(n)$  es la señal que se desea cancelar (es la suma de  $y(n) + r(n)$ ). Finalmente  $e(n)$  es la señal de error, cuando es pequeña, el eco ha sido eliminado.

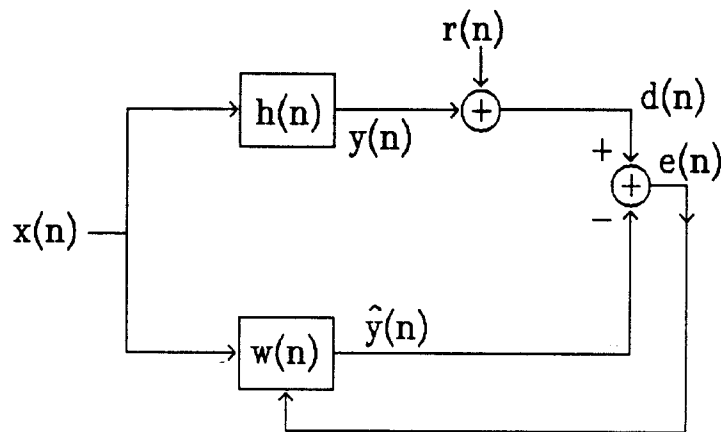


Figura 2.6.-Configuración para simular un cancelador de eco.

En las simulaciones realizadas se usa en la entrada  $x(n)$  a señal de voz , mientras que como ruido aditivo  $r(n)$  se usa ruido blanco.

## 2.7.-Referencias al capítulo

- [2.1] Herrera, E., Fundamentos de Ingeniería Telefónica, editorial Limusa 1989.
- [2.2] Oppenheim, A., Applications of Digital Signal Processing, editorial Prentice Hall 1978.
- [2.3] Sondhi M, and D.A. Berkley 1980 "Silencing echoes in the telephone network", proc. IEEE, vol 68, pp 948-963.
- [2.4] Sondhi M.M. and A.J.Presti 1966, "A self adaptive echo canceller", Bell Syst. Tech. J., vol 45 pp 1851-1854.
- [2.5] Sondhi, M.M. 1970 " Closed Loop Adaptive Echo Canceller using Generalized Filter Networks, U.S. Patent 3, 499,999.
- [2.6] Sondhi M.M. Mar. 1967 " An Adaptive Echo canceller", Bell System Tech, J, vol 46 No3 pp 497-511.
- [2.7] Lucky R.W. Feb 1966, "Techniques for Adaptive Equalization of Digital Communication Systems", Bell System Tech, J., vol 45 No 2 pp 255-286.
- [2.8] Demytko,N., and Mackechnie,L., "A High Speed Digital Adaptive Echo Canceller", Australian Telecommunication Review, vol. 7, No 1, pp 20-28, 1973.
- [2.9] Gritton,C.W.K.,D.W.Lin., "Echo Cancellation Algorithms".,Proc IEEE ASSP Magazine April 1984,pp 30-37.
- [2.10] Ochiai,K.,Araseki,T.,and Ogihara,T., "Echo Canceller with two Echo Path Models",IEEE Trans.COM-25,No 6, pp 589-595,June 1977.
- [2.11] Bellamy, J., Digital Telephony, editorial John Wiley 1991.

### III.-ESTRUCTURAS PARA FILTRADO ADAPTABLE

Son dos las partes básicas que conforman a los filtros adaptables, sus estructuras y el tipo de algoritmo a utilizar. En esta parte como primer punto se describe a los filtros adaptables desde el marco de operaciones que tienen que realizar. Como segundo punto desde la estructura a utilizar, esto es filtros no recursivos (FIR) y filtros recursivos (IIR).

#### 3.1.- Concepto básico de un Filtro Adaptable

Un filtro adaptable [3.1-3.5], está constituido por un filtro programable y un algoritmo de adaptación como se muestra en la figura 3.1. El filtro adaptable estima la estadística de la señal de entrada y ajusta su propia respuesta, en tal forma que minimiza una señal deseada  $d(n)$ ; esta señal deseada se obtiene a través del uso de una condición de entrada  $x(n)$ . La tarea del algoritmo adaptable es ajustar los coeficientes del filtro programable de manera que se minimice el error  $e(n)$  entre la salida del filtro  $\hat{y}(n)$  y la entrada  $d(n)$ .

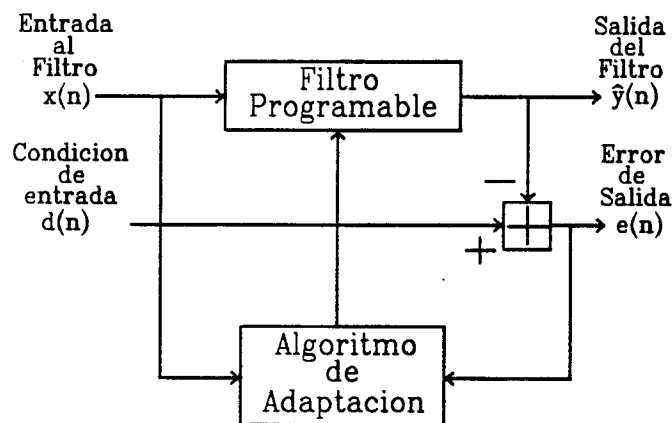


Figura 3.1.-El Filtro Adaptable.

Un filtro adaptable involucra dos procesos básicos:

- 1) El proceso de filtrado que produce una respuesta de salida a una entrada de datos.
- 2) El proceso de adaptación, cuyo propósito es proporcionar el mecanismo de control de un conjunto de parámetros usados en el proceso de filtrado.

Estos dos procesos trabajan de manera interactiva, desde luego la selección de una estructura tiene un gran efecto en la operación del algoritmo; básicamente son dos las estructuras, en las cuales se agrupan los filtros adaptables, ellas son :

1.-Filtros de Respuesta a impulso finita FIR

2.-Filtros de Respuesta a impulso infinita IIR

En la siguiente sección se presenta una revisión de las estructuras FIR e IIR.

### 3.2.-Estructuras de Filtros Digitales

Son dos las clases de estructuras de un filtro digital, ellas son recursivas (IIR) y no recursivas (FIR). La relación funcional entre la sucesión de entrada al filtro  $x(n)$  y la sucesión de salida resultante  $y(n)$  está descrita por :

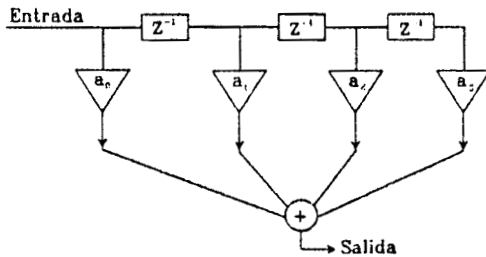
$$y_1(n) = F[y_1(n-1), y_1(n-2), \dots, x_1(n), x_1(n-1), \dots] \quad (3.1)$$

la muestra de salida  $y_1(n)$  es una función de las salidas pasadas también como de las muestras de entrada presente y pasadas.

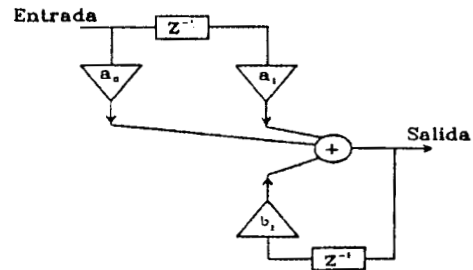
Para el caso de una realización no recursiva (FIR) la relación entre las sucesiones de entrada y salida es :

$$y_1(n) = F[x_1(n), x_1(n-1), x_1(n-2), \dots] \quad (3.2)$$

la muestra de salida  $y_1(n)$  es función solo de las entradas presente y pasadas. En la figura 3.2 se muestran las dos estructuras.



(a) Filtro No Recursivo ó Transversal



(b) Filtro Recursivo

Figura 3.2.-Estructuras de Filtros Digitales.

La Función de Transferencia  $W_1(z)$  de un filtro digital se puede expresar como :

$$W_1(z) = \frac{Y_1(z)}{X_1(z)} = \frac{\sum_{i=0}^N a_i z^{-i}}{\sum_{i=0}^N b_i z^{-i}} \quad (3.3)$$

donde  $Y_1(z)$  es la transformada  $z$  de la salida  $y_1$  del filtro digital,  $X_1(z)$  es la transformada  $z$  de la entrada  $x_1$  del filtro digital,  $z$  es una variable compleja y  $b_0 = 1$ .

A (3.3) se le puede escribir como :

$$Y_1(z) \sum_{i=0}^N b_i z^{-i} = X_1(z) \sum_{i=0}^N a_i z^{-i} \quad (3.4)$$

y de la propiedad de corrimiento de una sucesión :

$$y_1(n-k) \longleftrightarrow z^{-k} Y_1(z) \quad (3.5)$$

la ec. (3.4) queda

$$\sum_{i=0}^N b_i y_1(n-i) = \sum_{i=0}^N a_i x_1(n-i) \quad (3.6)$$

Como  $b_0 = 1$  la ec. (3.6) resulta

$$y_1(n) = \sum_{i=0}^N a_i x_1(n-i) - \sum_{i=1}^N b_i y_1(n-i) \quad (3.7)$$

La estructura que realiza a la ecuación (3.7) se muestra en la figura 3.3.

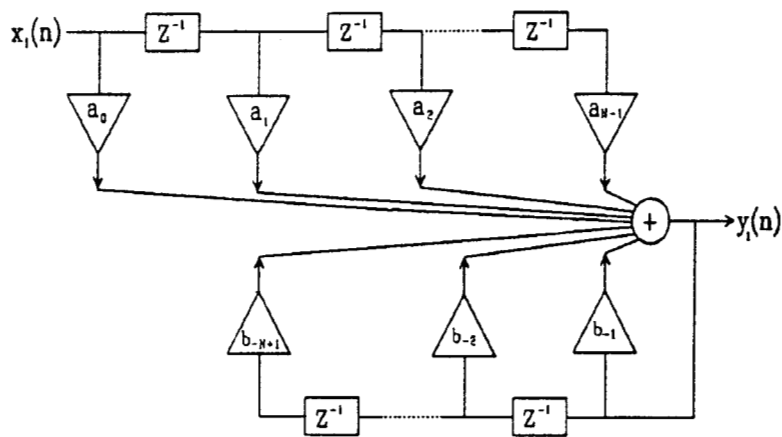


Figura 3.3.-Forma Directa 1.

La estructura de la figura 3.3 es conocida como forma directa 1, mantiene una relación directa a la transformada Z.

La expresión (3.3) es posible escribirla también como :

$$W_1(z) = \frac{Y_1(z)}{X_1(z)} = \left[ \frac{1}{\sum_{i=0}^N b_i z^{-i}} \right] \left[ \sum_{i=0}^N a_i z^{-i} \right] \quad (3.8)$$

Haciendo

$$W_2(z) = \frac{U(z)}{X_1(z)} = \frac{1}{\sum_{i=0}^N b_i z^{-i}} \quad (3.9)$$

donde  $U(z)$  es la transformada  $z$  de una sucesión  $u(n)$ , y

$$W_3(z) = \frac{Y_1(z)}{U(z)} = \sum_{i=0}^N a_i z^{-i} \quad (3.10)$$

Así de (3.9) y (3.10) se obtienen las ecuaciones de diferencia (suponiendo que  $b_0 = 1$ )

$$u(n) = x_1(n) - \sum_{i=1}^N b_i u(n-i) \quad (3.11)$$

$$y_1(n) = \sum_{i=0}^N a_i u(n-i) \quad (3.12)$$

Las expresiones (3.11) y (3.12) pueden ser realizadas como se muestra en la figura 3.4, dicha estructura contiene el número mínimo de multiplicadores, sumadores y elementos de retardo por lo cual se le conoce como canónica.

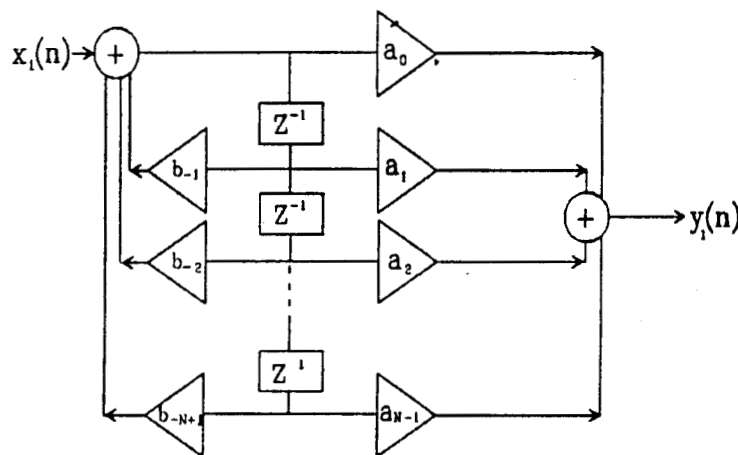


Figura 3.4.-Forma Directa 2.

Existen mas estructuras que realizan la función de transferencia de un filtro digital [3.3-3.4].

### 3.3.-El Filtro Transversal (FIR)

Cuando el denominador de la expresión (3.3) es una constante, la ecuación de diferencias que describe al sistema es del tipo no recursivo (ó transversal), la salida  $y_1(n)$  depende únicamente del número finito de valores presente y pasados de la entrada.

De esta forma escribimos a la función de transferencia de (3.3) como :

$$W_1(z) = \frac{Y_1(z)}{X_1(z)} = \sum_{i=0}^{N-1} w_i z^{-i} \quad (3.13)$$

nótese que el limite superior de la sumatoria se cambió a N-1 con el fin de describir a un filtro realizable con una duración de respuesta a impulso de N muestras.

La ecuación de diferencias correspondiente a (3.13) es

$$y_1(n) = w_0 x_1(n) + w_1 x_1(n-1) + \dots + w_{N-1} x_1(n-N+1) \quad (3.14)$$

La estructura para realizar filtros con respuesta a impulso finita se muestra en la figura 3.5.

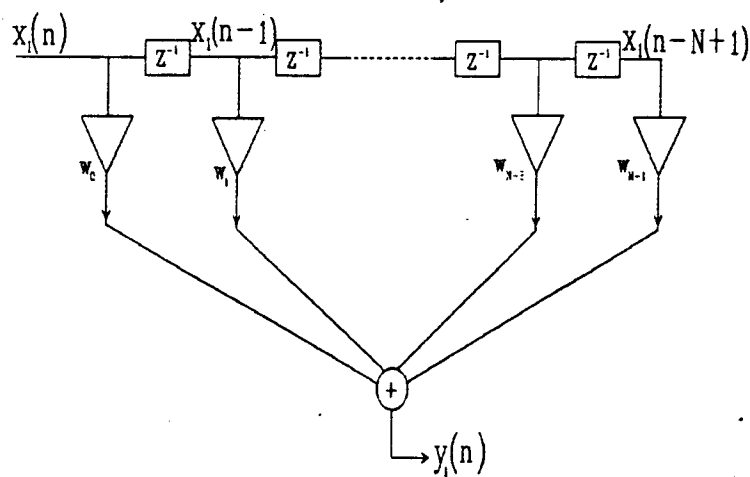


Figura 3.5.-Forma Directa para Filtros de Respuesta a Impulso Finita (FIR)



En la figura 3.5 los elementos de retardo se identifican por el operador  $z^{-1}$ , en particular cuando  $z^{-1}$  opera sobre la entrada  $x_1(n)$ , la salida resultante es  $x_1(n-1)$ . En cada derivación del filtro se multiplica la entrada a la derivación por el coeficiente del filtro, así la  $k$ -ésima entrada  $x_1(n-k)$  genera el producto  $w_k x_1(n-k)$ , donde  $w_k$  es el respectivo coeficiente del filtro. Finalmente todos los productos son sumados dando lugar a una salida del filtro  $y_1(n)$ , dicha salida está dada por:

$$y_1(n) = \sum_{k=0}^{M-1} w_k x_1(n-k) \quad (3.15)$$

La ec.(3.15) es la convolución de la respuesta a impulso del filtro  $w_n$  con la entrada al filtro  $x_1(n)$ , produciendo la salida  $y_1(n)$ .

### 3.4.-El Filtro IIR

La ecuación de diferencias de un filtro IIR esta dada por la ec. (3.16), esto es :

$$y_1(n) = \sum_{i=0}^N a_i x_1(n-i) - \sum_{i=1}^N b_i y_1(n-i) \quad (3.16)$$

la muestra de salida  $y_1(n)$  es una función de las salidas pasadas, como de las muestras de entrada presente y pasadas. En la figura 3.6 se muestra la estructura de un filtro IIR.

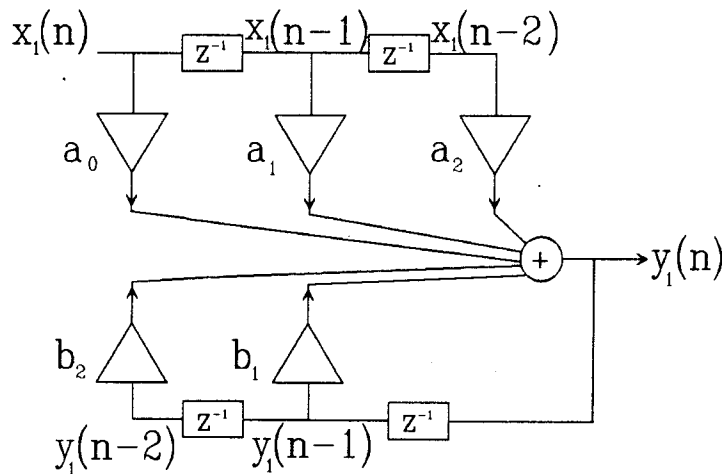


Figura 3.6.-Filtro IIR.

Una diferencia entre los filtros FIR e IIR, es que en los IIR se usan menos coeficientes para obtener una respuesta similar a la del filtro FIR [3.1,3.5] , esto es son filtros de menor orden. Otra diferencia entre un filtro FIR y un filtro IIR es que en el filtro IIR se incluyen trayectorias de retroalimentación, lo cual hace que la duración de la respuesta a impulso sea infinita, sin embargo la inclusión de estas trayectorias de retroalimentación introducen el problema de inestabilidad, debido a que los polos del filtro pueden estar fuera del círculo unitario; en particular para evitar que un filtro IIR sea inestable, debe tenerse mucho cuidado con los coeficientes de retroalimentación [3.6-3.9]. En contraste a lo anterior los filtros FIR siempre son estables.

### **3.5.-Conclusiones**

Dado que la estructura de los filtros FIR es incondicionalmente estable, son los mas usados en el filtrado adaptable. Por otra parte, la estructura FIR mas sencilla es el filtro transversal, la cual forma la base de estructuras mas complejas como filtrado en el dominio de frecuencia [3.1], etc. Así por lo anteriormente mencionado el filtro transversal fue el seleccionado para usar en este trabajo.

### 3.6.-Referencias

[3.1] Cowan,C.F.N.,Grant,P.M., Adaptive Filters,Prentice Hall,Englewood Cliffs,N.J.,1985.

[3.2] Haykin,S., Adaptive Filter Theory, Editorial Prentice Hall 1991.

[3.3] Widrow,B.,and Stearns,S.D.,Adaptive Signal Processing, Prentice-Hall,Englewood Cliffs,NJ, 1984.

[3.4] Haykin,S.,"Adaptive Filters:past,present and future",Proc IMA Conf. math.Signal Process,Warwick,England,1989.

[3.5] Treichler,J.R.,Johnson, C.R.,and Larimore,M.G.,"Theory and Design of Adaptive Filters",Wiley-Interscience, New York 1987.

[3.6] Honig, M.L., and Messersmitt, D.G., "Adaptive Filters: Structures, Algorithms and Applications", Boston : Kluwer 1984.

[3.7] Claasen, T.A.C.M., Melenbrauker.W.F.G.,"Adaptive Techniques for Signal Processing in Communications"., IEEE Communications Magazine., November 1985- Vol 23, No 11., pp 8-19.

[3.8] Feintuch, P.L.,"An Adaptive Recursive LMS Filter", Proceedings IEEE,vol. 64, No 11, pp 1622-1624, November 1976.

[3.9] Treichler, J.R.,Johnson, C.R.,and Larimore, M.G., Theory and Design of Adaptive Filters,Wiley-Interscience, New York 1987.

## IV.- ALGORITMOS PARA FILTRADO ADAPTABLE

En este capítulo se presentan los algoritmos : LMS, RLS, NLMS (LMS normalizado). Se aborda el efecto del factor de convergencia  $\alpha$  en el nivel de cancelación de los algoritmos presentados. Finalmente se presentan dos algoritmos de paso variable existentes en la literatura VSLMS [4.1] y VSS [4.2], los cuales superan los resultados del algoritmo NLMS; los resultados de los algoritmos NLMS y VSS son usados para compararlos con los algoritmos propuestos en los capítulos V y VI.

### 4.1.-Algoritmo LMS (Least Mean Squares)

Una estructura para filtrado adaptable [4.8] se muestra en la figura 4.3.

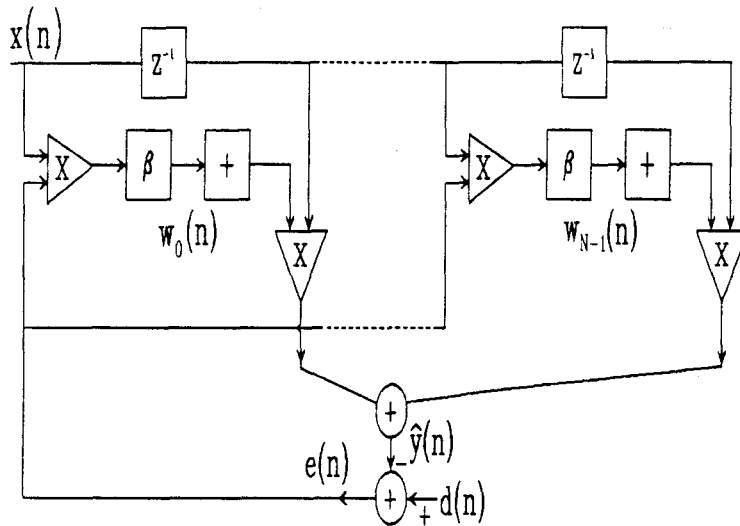


Figura 4.3.- Estructura del filtro adaptable LMS.

El algoritmo se basa en el método del paso descendente [4.19], su ecuación es:

$$e(n) = d(n) - X^T(n)W(n) \quad (4.1)$$

donde  $d(n)$  es la respuesta a seguir por el filtro adaptable

$$W^T(n) = [w_0(n), w_1(n-1), \dots, w_{N-1}(n-N+1)] \quad (4.2)$$

es el vector de coeficientes del filtro adaptable,

$$X^T(n) = [x(n), x(n-1), \dots, x(n-N+1)] \quad (4.3)$$

es el vector de entrada al filtro adaptable.

La expresión donde se modifican los coeficientes del filtro adaptable es :

$$W(n+1) = W(n) - \mu \nabla(n) \quad (4.4)$$

y

$$\nabla(n) = \frac{\partial}{\partial W} E[(d(n) - \hat{y}(n))^2] = \frac{\partial}{\partial W} E[e^2] \quad (4.5)$$

es el gradiente del error cuadrático medio del error,  $x(n)$ ,  $y(n)$  y  $e(n)$  representan a las muestras de las formas de onda  $x(t)$ ,  $y(t)$  y  $e(t)$  en el instante de muestreo  $nT$ ,  $N$  es el número de coeficientes en la estructura del filtro,  $\mu$  es un factor de control de la estabilidad, la convergencia y el error residual. En la práctica se usa el gradiente instantáneo del error cuadrático medio, es decir:

$$\nabla(n) = \frac{\partial}{\partial W} [e^2(n)] \quad (4.6)$$

Esto da origen al llamado algoritmo LMS, el cual se describe a continuación. Substituyendo la ec.(4.6) en (4.4) resulta :

$$W(n+1) = W(n) - \mu \frac{\partial e^2(n)}{\partial W} \quad (4.7)$$

$$W(n+1) = W(n) + 2 \mu e(n) X(n) \quad (4.8)$$

En la ec.(4.8), para asegurar la convergencia del algoritmo deben encontrarse los límites de  $\mu$ , para ello se minimiza el error cuadrático medio :

$$E[e^2(n)] = E[(d(n) - \hat{y}(n))^2] = E[(d(n) - X^T(n) W(n))^2] \quad (4.9)$$

usando el principio de ortogonalidad en mínimos cuadrados :

$$E[e(n) X(n)] = 0$$

$$E[X(n)(d(n) - X^T(n)W(n))] = 0$$

$$E[d(n) X(n)] = E[X(n) X^T(n) W(n)]$$

$$E[d(n) X(n)] = E[X(n) X^T(n)] W(n) \quad (4.10)$$

haciendo

$$P = E[d(n)X(n)] \quad (4.11)$$

es el vector de correlación entre la señal de referencia y el vector de entrada.

y

$$R = \begin{bmatrix} r_{xx}(0) & r_{xx}(1) & \dots & r_{xx}(N-1) \\ r_{xx}(1) & r_{xx}(0) & \dots & \dots \\ \dots & \dots & \dots & \dots \\ r_{xx}(N-1) & \dots & \dots & r_{xx}(0) \end{bmatrix} = E[X(n)X^T(n)] \quad (4.12)$$

es la matriz de autocorrelación de la señal de entrada.

Así de las ecuaciones (4.11) y (4.12), la ecuación (4.10) resulta :

$$P = RW \quad (4.13)$$

$$W_{\text{óptima}} = W^* = R^{-1}P \quad (4.14)$$

Para asegurar la convergencia del algoritmo LMS sustituímos la señal de error ec.(4.1) en la ec.(4.8) :

$$W(n+1) = W(n) + 2\mu X(n)(d(n) - X^T(n)W(n))$$

$$W(n+1) = W(n) + 2\mu d(n)X(n) - 2\mu X(n)X^T(n)W(n)$$

$$W(n+1) = W(n) + 2\mu d(n)X(n) - 2\mu X(n)X^T(n)W(n)$$

De la ec.(4.13), tomando el valor esperado de  $W(n+1)$  en la expresión anterior, restando  $R^{-1}P$  en ambos miembros de la ecuación y suponiendo que los coeficientes  $W(n)$  no están correlacionados con el vector de entrada, se obtiene :

$$\begin{aligned} E[W(n+1)] - R^{-1}P &= E[W(n)] - R^{-1}P + 2\mu[P - RE[W(n)]] \\ &= E[W(n)] - R^{-1}P + 2\mu[RR^{-1}P - RE[W(n)]] \\ &= E[W(n)] - R^{-1}P + 2\mu R[R^{-1}P - E[W(n)]] \end{aligned}$$

donde  $RR^{-1}$  es la matriz identidad, así :

$$E[W(n+1)] - R^{-1}P = E[W(n)] - R^{-1}P - 2\mu R[E[W(n)] - R^{-1}P] \quad (4.15)$$

definiendo :

$$\xi(n+1) = E[W(n+1)] - R^{-1}P \quad (4.16)$$

$$\xi(n) = E[W(n)] - R^{-1}P \quad (4.17)$$

substituyendo (4.16) y (4.17) en (4.15) se tiene :

$$\xi(n+1) = (I - 2\mu R) \xi(n) \quad (4.18)$$

Usando la transformada KL (Karhunen Loeve):

$$R = K_L^T \Phi K_L \quad (4.19)$$

donde

$$\Phi = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_N] \quad (4.20)$$

$$K_L K_L^T = I \quad (4.21)$$

multiplicando la ec.(4.19) por  $K_L$ , se tiene :



$$K_L R = K_L K_L^T \Phi K_L$$

$$K_L R = \Phi K_L$$

$$K_L R K_L^T = \Phi \quad (4.22)$$

Así la ec.(4.18) resulta :

$$\xi(n+1) = (I - 2\mu K_L^T \Phi K_L) \xi(n)$$

$$\xi(n+1) = (K_L^T K_L - 2\mu K^T \Phi K_L) \xi(n) \quad (4.23)$$

Definiendo

$$V(n+1) = K_L \xi(n+1) \quad (4.24)$$

entonces

$$V(n+1) = K_L K_L^T (I - 2\mu \Phi) K_L \xi(n)$$

$$V(n+1) = (I - 2\mu \Phi) V(n) \quad (4.25)$$

$$\begin{bmatrix} V_0(n+1) \\ V_1(n+1) \\ \vdots \\ V_{(N+1)}(n+1) \end{bmatrix} = \begin{bmatrix} 1-2\mu\lambda_1 & \cdot & \cdot & \cdot \\ \cdot & 1-2\mu\lambda_2 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & 1-2\mu\lambda_N \end{bmatrix} \begin{bmatrix} V_0(n) \\ V_1(n) \\ \vdots \\ V_{N-1}(n) \end{bmatrix}$$

$$V_i(n+1) = (1 - 2\mu\lambda_i) V_i(n) \quad (4.26)$$

Iterando la ec.(4.26), resulta :

$$V_i(n+1) = (1 - 2\mu\lambda_i)^{n+1} V_i(0) \quad (4.27)$$

De esta forma la convergencia del algoritmo LMS [4.9,4.22], se garantiza bajo la condición:

$$|1 - 2\mu\lambda_i| < 1 \quad (4.28)$$

ó bien

$$0 < \mu < \frac{1}{\lambda_{max}} \quad (4.29)$$

La razón de convergencia del algoritmo LMS depende de las características de la señal de entrada, donde  $\lambda_{max}$  es el máximo valor propio de la matriz de autocorrelación de la señal de entrada [4.13]. Debido a que el valor de  $\lambda_{max}$  depende de las características de la señal de entrada, no puede determinarse en situaciones prácticas,  $\mu$  debe tener un valor pequeño para asegurar estabilidad [4.8-4.11].

#### **4.2.-Complejidad Computacional del Algoritmo LMS**

Cuando el Algoritmo LMS es usado para adaptar los coeficientes de un filtro transversal, requiere  $N$  multiplicaciones para calcular la salida del filtro,  $N+1$  multiplicaciones para calcular  $2\mu e(n)X(n)$ , ya que el factor  $2\mu$  permanece constante durante toda la operación del sistema y  $2\mu e(n)$  es común para todos los coeficientes del filtro. Así la complejidad del filtro medida en términos del número de multiplicaciones es :

$$2N+1 \text{ multiplicaciones/período de muestreo} \quad (4.30)$$

### 4.3.-Algoritmo RLS (Recursive Least Squares)

De la expresión (4.13)

$$P = RW \quad (4.31)$$

P el vector de correlación entre la señal de referencia  $d(n)$  y el vector de entrada  $X(n)$ ;  $P^1$  se calcula como:

$$P(n) = \sum_{k=0}^n \lambda^{n-k} d(k) X(k) \quad (4.32)$$

donde  $\lambda^2$  es un factor de olvido, para asegurar que datos del pasado sean olvidados, con el propósito de seguir las variaciones estadísticas del vector de entrada  $X(n)$ .

R es la matriz de autocorrelación de la señal de entrada, la cual esta dada por :

$$R(n) = \sum_{k=0}^n \lambda^{n-k} X(k) X^T(k) \quad (4.33)$$

Desarrollando la expresión (4.32), P resulta :

$$P(n) = \sum_{k=0}^{n-1} \lambda^{n-k} d(k) X(k) + d(n) X(n) \quad (4.34)$$

$$P(n) = \lambda \sum_{k=0}^{n-1} \lambda^{n-k-1} d(k) X(k) + d(n) X(n) \quad (4.35)$$

$$P(n) = \lambda P(n-1) + d(n) X(n) \quad (4.36)$$

Desarrollando R en forma similar se obtiene

$$R(n) = \lambda R(n-1) + X(n) X^T(n) \quad (4.37)$$

Ahora multiplicando con  $R^{-1}$  por la izquierda a la expresión (4.31) resulta

---

<sup>1</sup>Referencia[4.25]

<sup>2</sup>Ver Apéndice I

$$W = R^{-1}P \quad (4.38)$$

Substituyendo (4.36) y (4.37) en (4.38)

$$W(n) = [\lambda R(n-1) + X(n)X^T(n)]^{-1} [\lambda P(n-1) + d(n)X(n)] \quad (4.39)$$

Finalmente desarrollando la expresión (4.39) (referencia [4.25]), se llega a:

$$W(n) = W(n-1) + K(n) e(n) \quad (4.40)$$

donde

$$K(n) = \frac{R^{-1}(n-1)X(n)}{\lambda + X^T(n)R^{-1}(n-1)X(n)} \quad (4.41)$$

La expresión (4.40) muestra el tiempo de cómputo necesario por el algoritmo RLS, para encontrar un nuevo valor de  $w(n)$ , ya que requiere de invertir la matriz de autocorrelación  $R^{-1}$  en cada iteración; en la figura 4.2 se muestra la estructura del algoritmo RLS [4.3-4.5,4.23].

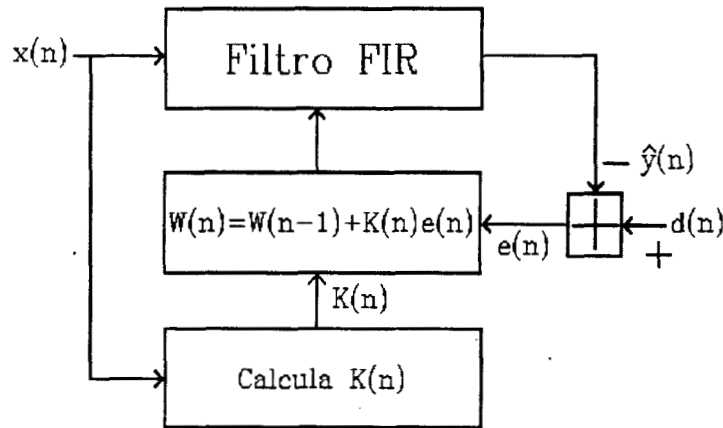


Figura 4.2.-Algoritmo RLS.

#### 4.4.-Complejidad computacional del Algoritmo RLS

Las expresiones (4.40) y (4.41) ilustran la complejidad computacional del algoritmo RLS [4.6], ya que es necesario calcular la matriz inversa  $R^{-1}(n)$  en cada iteración. Para un filtro de orden  $N$ , se requerirían del orden de  $N^2$  multiplicaciones, y dado que en canceladores de eco deben usarse filtros adaptables con  $N=100$ , se requerirían 10,000 multiplicaciones por iteración, esto hace que la aplicación del algoritmo RLS esté limitada en nuestro caso. Por lo anterior el algoritmo mas usado en esta aplicación es el LMS descrito en la sección 4.5, el cual tiene una complejidad del orden de  $N$  multiplicaciones [4.7,4.14].

#### 4.5.-El Algoritmo LMS Normalizado (NLMS)

De los resultados obtenidos en la sección (4.5), la convergencia del algoritmo LMS depende de la potencia de la señal de entrada, buscando eliminar esta dependencia dió como resultado la aparición del algoritmo NLMS, el cual se describe en esta sección.

De la ec.(4.8), el algoritmo LMS [4.16-4.17] es :

$$W(n+1) = W(n) + 2\mu e(n)X(n) \quad (4.42)$$

donde  $\mu$  es un factor que controla la estabilidad y velocidad de convergencia.

Considerando la diferencia entre el vector óptimo  $W^*$  y el vector de coeficientes  $W(n)$ , dada por :

$$V(n) = W^* - W \quad (4.43)$$

Suponiendo que la señal deseada está dada por :

$$d(n) = W^{*T}X(n) \quad (4.44)$$

y que la señal de error es :

$$e(n) = d(n) - W^T(n)X(n) \quad (4.45)$$

substituyendo (4.44) en (4.45) a partir de (4.43), se obtiene :

$$e(n) = W^{*T}X(n) - W^T(n)X(n) \quad (4.46)$$

$$e(n) = [W^{*T} - W^T(n)]X(n) \quad (4.47)$$

$$e(n) = V^T(n) X(n) \quad (4.48)$$

Ahora haciendo una proyección del vector de diferencia  $V(n)$  en el vector  $X(n)$  (figura 4.4).

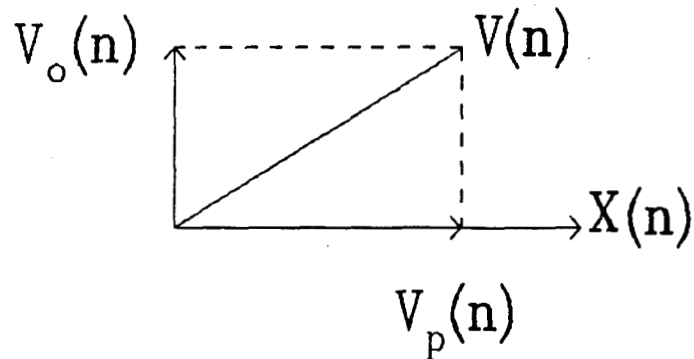


Figura 4.4.- Descomposición del vector  $V(n)$ .

Haciendo esto se descompone  $V(n)$  como :

$$V(n) = V_o(n) + V_p(n) \quad (4.49)$$

donde  $V_o(n)$  es la componente de  $V(n)$  ortogonal al vector de entrada  $X(n)$  y  $V_p(n)$  es la componente de  $V(n)$  paralela a  $X(n)$ , esto implica :

$$V_p(n) = CX(n) \quad (4.50)$$

substituyendo (4.49) y (4.50) en (4.48) se obtiene :

$$e(n) = [V_o(n) + V_p(n)]^T X(n) \quad (4.51)$$

$$e(n) = [V_o(n) + CX(n)]^T X(n) \quad (4.52)$$

Debido a que  $V_o(n)$  es ortogonal a  $X(n)$ , entonces se tiene :

$$V_o(n) X(n) = 0 \quad (4.53)$$

y entonces :

$$e(n) = C X^T(n) X(n) \quad (4.54)$$

despejando C se obtiene :

$$C = \frac{e(n)}{X^T(n) X(n)} \quad (4.55)$$

así substituyendo (4.55) en (4.50) se obtiene :

$$V_p(n) = \frac{e(n) X(n)}{X^T(n) X(n)} \quad (4.56)$$

El objetivo es hacer al vector  $V(n)$  mas ortogonal a  $X(n)$  en cada iteración (ver la ecuación (4.48)), en forma tal que  $V(n)$  sea mas ortogonal a  $X(n)$ , que como lo fue el vector  $V(n-1)$  (figura 4.5).

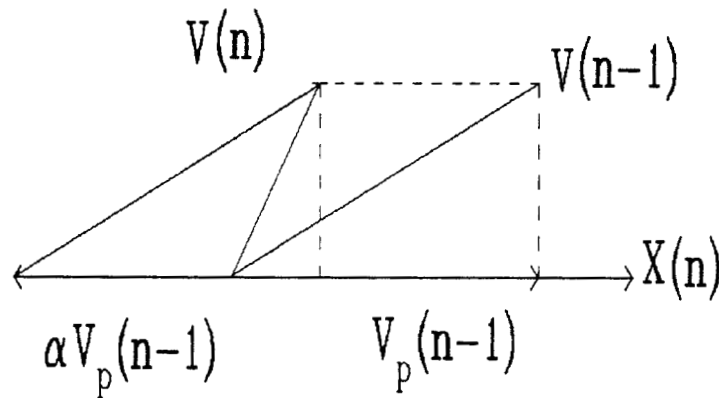


Figura 4.5.-Descomposición de los vectores  $V(n)$  y  $V(n-1)$ .

Lo anterior se logra substrayendo el vector  $\alpha V_p(n-1)$  del vector  $V(n-1)$ , esto es :

$$V(n) = V(n-1) - \alpha V_p(n-1) \quad (4.57)$$



Finalmente substituyendo (4.43) en (4.57) :

$$W^* - W(n) = W^* - W(n-1) - \alpha \frac{e(n-1) X(n-1)}{X^T(n-1) X(n-1)} \quad (4.58)$$

$$W(n) = W(n-1) + \alpha \frac{e(n-1) X(n-1)}{X^T(n-1) X(n-1)} \quad (4.59)$$

Con  $\beta = 2\mu$  la ecuación (4.42) se transforma en :

$$W(n+1) = W(n) + \beta e(n) X(n) \quad (4.61)$$

donde

$$\beta = \frac{\alpha}{X^T(n) X(n)} \quad (4.62)$$

En este caso  $\alpha$  es una constante, así que puede seleccionarse de manera independiente de las características de la señal de entrada [4.10].

#### 4.6.-Efecto del Factor de Convergencia $\alpha$ en el Nivel de Cancelación del Algoritmo.

La figura 4.6 es útil para analizar el efecto del factor de convergencia  $\alpha$  en las características del algoritmo LMS [4.12].

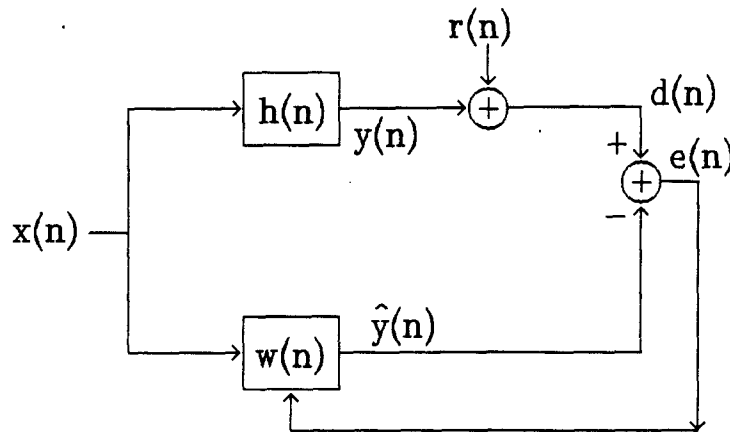


Figura 4.6.-Filtrado Adaptable en configuración de identificador de sistemas .

En la fig. 4.6 el error de salida está dado por :

$$e(n) = X^T(n) [H - W(n)] + r(n) \quad (4.63)$$

donde H es el vector de coeficientes del sistema desconocido y r(n) es una señal no correlacionada con x(n), la cual dependiendo de la aplicación, podría ser la señal útil (cancelación de ruido) ó una distorsión (identificación de sistemas). De (4.63) en (4.61) y substrayendo H en ambos lados de la ecuación resultante, se obtiene :

$$W(n+1) - H = W(n) - H + \beta X(n) (X^T(n) [H - W(n)] + r(n)) \quad (4.64)$$

$$W(n+1) - H = [I - \beta X(n) X^T(n)] [W(n) - H] + \beta r(n) X(n) \quad (4.65)$$

donde  $I$  denota la matriz identidad. Después de que la convergencia ha sido obtenida,  $W(n+1)$  se puede suponer muy próxima a  $W(n)$ . Así se tiene que el error de identificación  $e_1$  definido como :

$$e_1 = y(n) - \hat{y}(n) = X^T [H - W(n)] \quad (4.66)$$

a partir de (4.65)  $e_1(n)$  se puede escribir aproximadamente como :

$$e_1(n) = X^T [I - \beta X(n) X^T(n)] [H - W(n)] + \beta r(n) X^T(n) X(n) \quad (4.67)$$

$$e_1(n) = [X^T(n) - \beta X^T(n) X(n) X^T(n)] [H - W(n)] + \beta r(n) X^T(n) X(n) \quad (4.68)$$

Substituyendo (4.62) en (4.68) se tiene :

$$e_1(n) = \left[ X^T(n) - \alpha \frac{X^T(n) X(n)}{X^T(n) X(n)} X^T(n) \right] [H - W(n)] + \alpha \frac{X^T(n) X(n)}{X^T(n) X(n)} r(n) \quad (4.69)$$

$$e_1(n) = [X^T(n) - \alpha X^T(n)] [H - W(n)] + \alpha r(n) \quad (4.70)$$

Factorizando  $X^T(n)$  se obtiene :

$$e_1(n) = [1 - \alpha] e_1(n) + \alpha r(n) \quad (4.71)$$

elevando al cuadrado (4.71) se tiene :

$$e_1^2(n) = [1 - \alpha]^2 e_1^2(n) + 2[1 - \alpha] \alpha e_1(n) r(n) + \alpha^2 r^2(n) \quad (4.72)$$

Tomando el valor esperado y suponiendo que  $X(n)$  y  $r(n)$  no están correlacionados y por lo tanto  $e_1(n)$  y  $r(n)$  tampoco lo están, se obtiene :

$$E[e_1^2(n)] = [1 - \alpha]^2 E[e_1^2(n)] + \alpha^2 E[r^2(n)] \quad (4.73)$$

$$E[e_1^2(n)] = [1 - 2\alpha + \alpha^2] E[e_1^2(n)] + \alpha^2 E[r^2(n)] \quad (4.74)$$

$$E[e_1^2(n)](1 - 2\alpha + \alpha^2) = \alpha^2 E[r^2(n)] \quad (4.75)$$

$$\alpha(2 - \alpha)E[e_1^2(n)] = \alpha^2 E[r^2(n)] \quad (4.76)$$

despejando a  $E[e_1^2(n)]$  se obtiene :

$$E[e_1^2(n)] = \frac{\alpha}{2 - \alpha} E[r^2(n)] \quad (4.77)$$

De aquí se observa que cuando  $\alpha = 1$  el algoritmo converge al nivel de ruido y que el nivel de cancelación obtenido mejora inversamente proporcional al valor de  $\alpha$ . Una aplicación de esto es la cancelación de eco.

Por otra parte una Figura de Mérito para un filtro adaptable operando en configuración de identificación de sistemas [4.15] es el ERLE (echo return loss enhancement), el cual representa una mejora en la cancelación del eco. El ERLE se define como :

$$ERLE = 10 \log_{10} \frac{E[y^2(n)]}{E[e_1^2(n)]} \quad (4.78)$$

esto indica el nivel de cancelación de la señal  $y(n)$  obtenida. Substituyendo (4.77) en (4.78) se obtiene :

$$ERLE = 10 \log_{10} \frac{E[y^2(n)]}{\frac{\alpha}{2 - \alpha} E[r^2(n)]} \quad (4.80)$$

$$ERLE = 10 \log_{10} \frac{E[y^2(n)]}{E[r^2(n)]} + 10 \log_{10} \left[ \frac{2}{\alpha} - 1 \right] \quad (4.81)$$

Este factor indica el grado de eliminación ó de atenuación de la interferencia obtenida. De aquí se puede ver que con  $\alpha=1$  el nivel de cancelación alcanzado es igual a la relación :

$$ERLE = 10 \log_{10} \frac{E[y^2(n)]}{E[r^2(n)]} \quad (4.82)$$

Conforme  $\alpha$  decrece el grado de atenuación de la señal de interferencia aumenta inversamente proporcional a los decrementos de  $\alpha$ .

#### 4.7.-Desventajas del Algoritmo NLMS.

La desventaja del algoritmo NLMS es el hecho de seleccionar una  $\alpha$  en particular; por ejemplo si se selecciona una  $\alpha$  grande, se tiene una rápida convergencia del algoritmo, sin embargo el error de salida es grande y el algoritmo puede quedarse oscilando alrededor del mínimo (ver figura 4.6'); ahora bien si se usa una  $\alpha$  pequeña, el error de salida es pequeño, pero la rapidez de convergencia es lenta. Por lo anterior la selección del paso  $\alpha$  refleja un compromiso entre el error de salida en el algoritmo y la rapidez de adaptación [4.13].

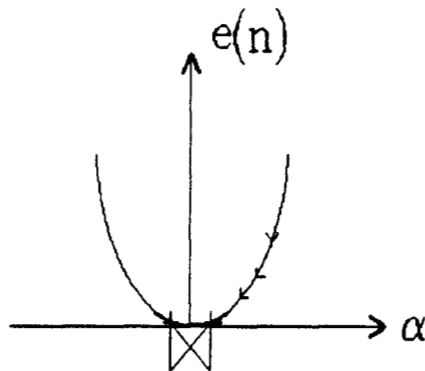


Figura 4.6'.-Búsqueda del mínimo error en el algoritmo NLMS.

Dado el compromiso entre el tamaño del paso y la velocidad de convergencia del algoritmo NLMS, este ha motivado al desarrollo de algoritmos con factores de convergencia variables, los cuales proporcionan velocidades de convergencia relativamente altas al inicio y altos niveles de cancelación de eco al alcanzar la convergencia. En las siguientes dos secciones se presenta una revisión de los algoritmos de paso variable VSLMS [4.1] y VSS [4.2], con ellos se obtiene un mejor nivel de cancelación de eco respecto al algoritmo NLMS.

#### 4.8.-Algoritmo VSLMS (Variable Step Least Mean Squares)

El algoritmo VSLMS [4.1] usa una  $\alpha$  de paso variable, se basa en el principio mostrado en la figura 4.7.

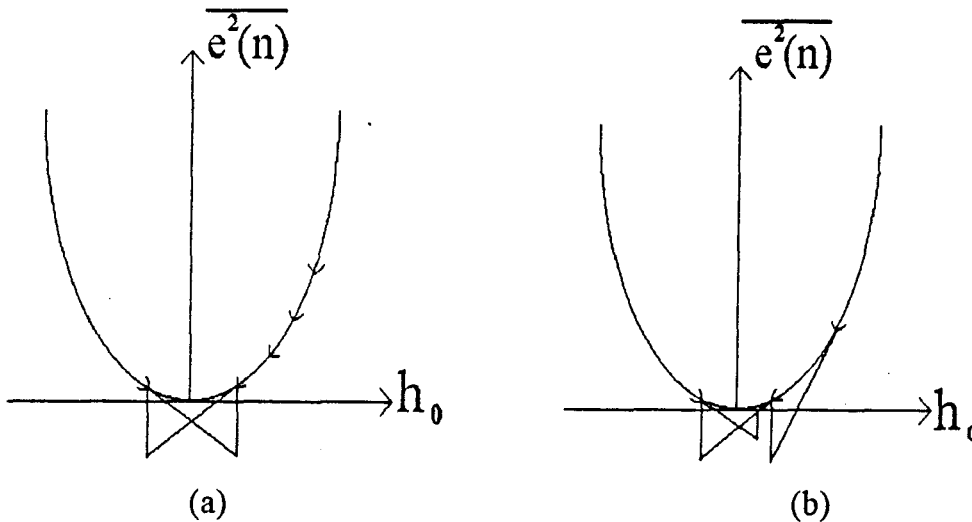


Figura 4.7.-Efecto del factor  $\alpha$  en la convergencia del Algoritmo VSLMS.

En la figura 2.7 se observa que cuando el factor de convergencia es mas pequeño que el requerido, el gradiente no cambia de signo durante varias iteraciones. Por otra parte cuando el factor de convergencia es mayor que el máximo requerido, el gradiente cambia de signo en cada iteración debido a que la adaptación sobrepasa el mínimo de la superficie MSE. Así atendiendo a esto, en [4.1] propusieron un algoritmo LMS con coeficiente de convergencia variable el cual está dado por :

$$H(n+1) = H(n) + \mu e(n)X(n) \quad (4.83)$$

donde

$$M = \begin{bmatrix} 2\mu_0 & & & \\ & 2\mu_1 & & \\ & & 2\mu_2 & \\ & & & \ddots \\ & & & & 2\mu_{N-1} \end{bmatrix} \quad (4.84)$$

cada  $\mu_i$  es calculada en base al número de cambios de signo ( $N_{SD}$ ) ó de signos iguales ( $N_{SI}$ ) en el gradiente del MSE. Si el número de signos iguales consecutivos del gradiente :

$$e(n) \times (n - k) \quad (4.85)$$

es igual a  $N_{SI}$  entonces :

$$\mu_k = L\mu_k \quad (4.86)$$

y si por el contrario el número de cambios de signo consecutivos del gradiente es igual a  $N_{SD}$ , entonces :

$$\mu_k = \frac{\mu_k}{L} \quad (4.87)$$

además se agregan dos condiciones, las cuales se verifican en cada iteración. Esto es :

Si  $\mu_k > \mu_{max}$  entonces  $\mu_k = \mu_{max}$  (4.88)

y si  $\mu_k < \mu_{min}$  entonces  $\mu_k = \mu_{min}$  (4.89)

En caso de que ninguna de las dos condiciones mencionadas inicialmente se satisfagan, el factor de convergencia no se modifica. La  $\mu_{max}$  es escogida generalmente como :

$$\mu_{max} = \frac{1}{Nx^2(n)} \quad (4.90)$$

La habilidad de rastreo de este algoritmo es pobre, especialmente cuando hay cambios grandes en las estadísticas de la señal de referencia. Lo anterior es debido a que el paso se incrementa gradualmente después de que un cambio abrupto ocurre en la señal de referencia.

#### 4.9.- Algoritmo VSS (Variable Step Size)

Otro algoritmo recientemente propuesto es el llamado algoritmo VSS[4.2]. Este se basa en el hecho de que si el algoritmo es convergente,  $\overline{e^2(n)}$ , donde  $e(n)$  es el error residual, decrece con el tiempo. Así el algoritmo VSS usa un factor de convergencia  $\alpha$  dado por :

$$\alpha(n) = \frac{1}{n} \sum_{k=1}^n e^2(k) \quad (4.91)$$

donde  $\alpha(n)$  puede ser estimado en forma recursiva<sup>1</sup>.

El algoritmo operando en esa forma podría ser inestable al inicio si :

$$e^2(n) > 2 \quad (4.92)$$

de acuerdo a (4.60), y podría tener una respuesta lenta a cambios abruptos en las estadísticas de las señales de interés. Para evitar estos problemas se define el factor de convergencia  $\alpha(n)$  de la siguiente manera :

$$\alpha'(n) = (1 - \varepsilon)\alpha(n-1) + \varepsilon e^2(n) \quad (4.93)$$

si  $\alpha'(n) > \alpha_{\max}$  entonces  $\alpha(n) = \alpha_{\max}$  (4.94)

y si  $\alpha'(n) < \alpha_{\min}$  entonces  $\alpha(n) = \alpha_{\min}$  (4.95)

En caso de que no se satisfaga ninguna de las dos condiciones :

$$\alpha(n) = \alpha'(n) \quad (4.96)$$

$\alpha_{\max}$  y  $\alpha_{\min}$  son dos cantidades seleccionadas de antemano, los valores usados fueron :

$$\alpha_{\max} = 1 \text{ y } \alpha_{\min} = 0.05, 0.1 \quad (4.97)$$

---

<sup>1</sup>ver Apéndice I



el algoritmo se inicializa con :

$$\alpha(0) = \alpha_{\max} \quad (4.98)$$

Los valores de  $\alpha_{\min}$  dependerán del nivel de cancelación del filtro adaptable.

#### 4.10.-Algoritmo en Bloque FLMS (Fast Least Mean Squares)

Una alternativa al algoritmo LMS tradicional, es el algoritmo en bloque FLMS, la idea es modificar los coeficientes del filtro adaptable al final de cada bloque, de esta manera es posible reducir la complejidad computacional de los filtros adaptables FIR. Sin embargo el algoritmo de bloque tiene una baja velocidad de convergencia.

En este algoritmo se transforma la señal de entrada al dominio de la frecuencia antes del filtrado adaptable, la transformación se basa en la Transformada Rápida de Fourier FFT [4.24].

Sea un filtro adaptable FIR cuya señal de salida está dada por

$$y(n) = X^T(n)W(n) \quad (4.99)$$

donde

$$W(n) = [w_0, w_1, \dots, w_{N-1}]^T \quad (4.100)$$

es el vector de coeficientes y

$$X(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T \quad (4.101)$$

es el vector de entrada.

Se supone que la señal de referencia se encuentra dada por

$$d(n) = H^T X(n) + r(n) \quad (4.102)$$

donde H es el vector de coeficientes del sistema a ser identificado y  $r(n)$  es una señal que no puede ser identificada por  $y(n)$ .

El vector de coeficientes será modificado usando un algoritmo basado en la búsqueda del gradiente, el cual está dado por

$$W(k+1) = W(k) + 2\mu\nabla(k) \quad (4.103)$$

donde el gradiente  $\nabla(k)$  es estimado usando la relación

$$\nabla(k) = \sum_{j=0}^{N-1} e(kl+j) x(kl+j) \quad N > 0 \quad (4.104)$$

donde el error de salida  $e(kl+j)$  estará dado por

$$e(kl+j) = d(n) - X^T(k) W(k)$$

Usando la ecuación (4.142) el vector de coeficientes es modificado solamente una vez cada  $N$  períodos de muestreo, es decir una vez cada bloque, donde la longitud del bloque es igual a  $N$ .

Debido a que los coeficientes del filtro permanecen constantes durante  $N$  períodos de muestreo, la convolución dada por (4.138) podría implementarse usando la Transformada Rápida de Fourier (FFT). Así usando el método de convolución rápida con 50 % de traslape, al vector de coeficientes se le agregan  $N$  ceros y se calcula una FFT de  $2N$  puntos. Así si denotamos  $C(k)$  como el vector de coeficientes en el dominio de la frecuencia este estará dado por :

$$C(k) = \text{FFT}[W(k), 0, \dots, 0] \quad (4.105)$$

N ceros

Ahora se define una matriz  $B(k)$  como :

$$B(k) = \text{diag}\{\text{FFT}[x(kl-N), \dots, x(kl-1), x(kl), \dots, x(kl+N-1)]\} \quad (4.106)$$

Así la salida del filtro  $y(kN), y(kN+1), \dots$ , estará dada como :

$$Y(k) = \text{Ultimos } N \text{ términos de } [\text{IFT}(B(k)C(k))] \quad (4.107)$$



La convergencia de este algoritmo esta dada por [4.25] :

$$0 < \mu < \frac{2}{N \lambda_{\max}} \quad (4.150)$$

El algoritmo FLMS realiza un procesamiento en bloque, los coeficientes del filtro adaptable se modifican solo al final de cada bloque. Se hace uso de la FFT para calcular las convoluciones lineales requeridas durante el filtrado y adaptación. Este algoritmo reduce la complejidad computacional de los filtros adaptables RIF [4.20], Sin embargo tiene una baja velocidad de convergencia.

## 4.11.- Conclusiones

Se han expuesto varios algoritmos para filtrado adaptable ,ellos se dividen en dos ramas, es decir algoritmo de búsqueda del gradiente (LMS) y algoritmo RLS. Dada la complejidad computacional del algoritmo RLS, ya que se requiere del cálculo de la matriz inversa  $R^{-1}(n)$  en cada iteración, y por otra parte en cancelación de ecos se usan filtros adaptables con  $N=100$ , se requerirían 10000 multiplicaciones por iteración. Por este motivo se seleccionó al algoritmo LMS para trabajar en éste proyecto.

Aunque considerable esfuerzo ha sido realizado también para reducir la complejidad computacional de los filtros adaptables .En este sentido se ha propuesto el algoritmo de procesamiento en bloque FLMS, sin embargo éste tiene en general una baja velocidad de convergencia Por lo anterior se seleccionó para éste proyecto al algoritmo LMS.

Dada la característica del algoritmo LMS, de usar un tamaño de paso fijo, si se usa un paso grande se tiene una rápida convergencia del algoritmo, sin embargo el error de salida es grande; ahora bien si se usa un paso pequeño, el error de salida es pequeño, pero la rapidez de convergencia es lenta. Por lo anterior resultó interesante desarrollar algoritmos que usaran mas de un paso para la adaptación, para de esta forma obtener velocidades de convergencia altas al inicio y altos niveles de cancelación al alcanzar la convergencia. .En particular se trabajó en los algoritmos VSS-Doble paso y VSS-AC los cuales se describen en el siguiente capítulo.

En este capítulo se describieron dos algoritmos de paso variable VSLMS [4.1] y VSS [4.2], como propuestas existentes que superan al algoritmo LMS. En particular dado que el algoritmo VSS supera al VSLMS, se usó al VSS [4.2] para comparar con el algoritmo VSS-CC propuesto en éste proyecto.

## 4.12.- Referencias

- [4.1] Harris,R.W., Chabries,D.M.,and Bishop,F.A., "A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316,Apr.1986.
- [4.2] Raymond ,H., Kwong,E., and Johnston,W., "A Variable Step Size LMS Algorithm ",IEEE Transactions on Signal Processing,vol.40 No.7 pp. 1633-1642,July 1992.
- [4.3] Carayannis,G.,Manolakis,D., and Kaloptsidis, N., "Fast Kalman Type Algorithms for Sequential Signal Processing,"Proc 1983 IEEE Internat.Conf Acoust.Speech, Signal Processing,pp 186-189.
- [4.4] Lawrence,R.E.,and H Kaufman(1971), "The Kalman Filter for the equalization of a digital communication channel",IEEE Trans.Commun. Technol.,vol.COM-19,pp 1137-1141, 1971.
- [4.5] Mulgrew,B., "Kalman Filter Techniques in Adaptive Filtering ",IEEE Proc(London)vol. 134.pt.F,pp 239-243.
- [4.6] Bershad,N, and Macchi,O. "Comparison of RLS and LMS algoritihms for tracking a chirped signal",Proc.ICASSP,Glasgow, Scotland, pp 896-899.1989.
- [4.7] Eweda,E. "RLS/LMS/ sign algorithms comparison",IEEE Trans. on Signal Processing, pp 2937-2944,Nov. 1994.
- [4.8] Widrow,B., McCool,J.M.,Larimore,M.G.,and Johnson,C.R., "Stationary and Non-Stationary Learning Characteristics of the LMS Adaptive Filter",Proceedings IEEE, Vol. 64,No 8,pp.1151-1161, August 1976.
- [4.9] Weinstein, E., "Stability Analysis of LMS Adaptive Filters",submitted for publication, 1983.
- [4.10] Bershad, N.J, "Analysis of the normalized LMS algorithm with Gaussian inputs",IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-34, pp 793-806, 1986.
- [4.11] Hsia,T.C. "Convergence analysis of LMS and NLMS adaptive algorithms", Proc,ICASSP, Boston, pp. 667-670, 1983.

- [4.12] Clark,G.A.,Mitra,S.K.,and Parker,S.R.,"Block Implementation of Adaptive Digital Filters",IEE Trans.,Vol.CAS-28,No 6,pp 584-592,June 1981.
- [4.13] Solo,V.,"The limiting behavior of LMS", IEEE Trans. Acoust. Speech Signal Process., vol. 37, pp. 1909-1922, 1989.
- [4.14] Widrow,B.,Mc.Cool,J.,Ball,M.,"The Complex LMS Algoritihm",Proceedings IEEE,vol.63,No 4, pp 719-720,April 1975.
- [4.15] Perez,H., Amano,F., "Acoustic Echo Cancellation Using Multirate Techniques", IEICE Transactions, vol. E74, no 11 pp 3559-3568, Nov1991.
- [4.16] Widrow,B. 1966 Adaptive Filters I: Fundamentals,Rep.SEL-66-126 (TR6764-6)Stanford Electronics Laboratories,Stanford Calif.
- [4.17] Zeidler,J.R. (1990), "Performance Analysis of LMS Adaptive Prediction Filters", scheduled for publication in Fall 1990, Proc. IEEE, vol. 78.
- [4.18] Sharpe,S.M.,and Nolte,L.W.,"Adaptive MSE estimation",Proc. ICASSP, Atlanta, Ga.,pp 518-521, 1981.
- [4.19] Widrow,B.,and McCool,J.M.,"A Comparison of Adaptive Algorithms Based on the Methods of Steepest Descent and Random Search",IEEE trans.,Vol.AP-24,No 5,pp 615-637, September 1976.
- [4.20] Ferrara,E.R.,"Fast Implementation of LMS Adaptive Filters", IEEE Trans.vol. ASSP-28,No 4,pp 474-475, August 1980.
- [4.21] Weckler,G.P.,and Walby,M.D.,"Programable Transversal Filters:Design Tradeoffs",Proceedings 5th International Conference on CCD's CCD-79, Centre for Industrial Consultancy and Liaison, University of Edinburgh, pp 211-221, 1979.
- [4.22] Weiss, A., and Mitra, D.,"Digital Adaptive Filters: Conditions for Convergence, Rates of Convergence, Effects of Noise and Errors Arising from the Implementation,"IEEE Trans. Information Theory, Nov 1979, Vol.IT-25, No 6,pp 637-652.
- [4.23] Samson, C.,"A unified treatment of fast Kalman algorithms for identification",Inte.J.Control,vol.35,pp 909-934, 1982.

[4.24] Brigham, E.O., The Fast Fourier Transform, Prentice Hall, Englewood Cliffs, NJ, 1974.

[4.25] Cowan, C.F.N., Grant, P.M., Adaptive Filters, Prentice Hall, Englewood Cliffs, N.J., 1985.



## V.-ALGORITMOS NLMS DE PASO VARIABLE PROPUESTOS VSS-Doble paso y VSS-AC

En éste capítulo se presentan los algoritmos de paso variable propuestos en el desarrollo de éste proyecto de investigación, tal como se fueron desarrollando. Se inicia con el algoritmo VSS-Doble paso, el siguiente algoritmo es el VSS-AC. Estos algoritmos muestran el desarrollo de éste proyecto de investigación, ellos superan al algoritmo NLMS; ambos usan dos tamaños de paso  $\alpha_1$  (paso grande) y  $\alpha_2$  (paso pequeño). Para la prueba de los algoritmos se llevaron a cabo simulaciones, bajo las mismas condiciones, esto es : número de coeficientes del filtro adaptable FIR (128 derivaciones), señal de voz como señal de entrada (se usaron 2 archivos, voz1 y voz2); como ruido aditivo se usó ruido blanco (se usaron 2 archivos, ruido blanco1 y ruido blanco2).

### 5.1.1.-Algoritmo VSS-Doble paso

El algoritmo NLMS usa un paso  $\alpha$  constante, la selección del paso refleja un compromiso entre el error de salida y la velocidad de convergencia. Por ejemplo si se usa un paso pequeño, el error de salida es pequeño; sin embargo presenta una velocidad de convergencia baja. Por otra parte si se usa un paso grande la velocidad de convergencia se incrementa, pero se tiene un error de salida grande.

En el algoritmo VSS-Doble paso [5.1] , la idea es usar dos pasos, el ajuste del paso es controlado por el cuadrado del error de salida. Un error de salida grande hace que el paso a usar en el algoritmo sea grande ( $\alpha_1$ ), de esta manera se tendrá una velocidad de convergencia rápida. Por otra parte cuando el error de salida ha disminuído, el algoritmo usará un paso pequeño  $\alpha_2$ , haciendo que el error de salida sea menor. En la figura 5.1 se muestra el algoritmo propuesto.

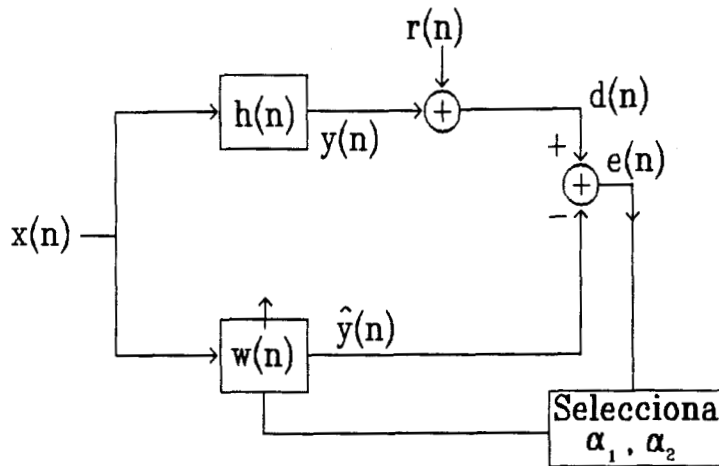


Figura 5.1.-Algoritmo VSS-Doble paso.

De la misma forma que el algoritmo NLMS, el algoritmo propuesto adapta los coeficientes del filtro a partir de la minimización del promedio de la potencia del error de salida, es decir :

$$E[e^2(n)] \quad (5.1)$$

con

$$e(n) = d(n) - X^T(n) W(n) \quad (5.2)$$

La versión normalizada del algoritmo propuesto es :

$$W(n+1) = W(n) + \beta e(n) X(n) \quad (5.3)$$

donde

$$\beta = \frac{\alpha}{|X(n)|^2} \quad (5.4)$$

$X(n)$  es el vector de entrada,  $W(n)$  es el vector de coeficientes del filtro adaptable, y  $0 < \alpha < 2$  es el factor que controla la estabilidad y velocidad de convergencia. Utilizando la figura de mérito de un filtro adaptable ec.(4.120) del capítulo anterior se tiene :

$$ERLE = \frac{S}{R} + 10 \log_{10} \left( \frac{2}{\alpha} - 1 \right) \quad (5.5)$$

El algoritmo VSS-Doble paso usa 2 valores de  $\alpha$ , al principio el algoritmo usa

$$\alpha_1 = 1 \quad (5.6)$$

así el ERLE resulta como :

$$\text{ERLE} = \frac{S}{R} \quad (5.7)$$

en éste caso el comportamiento del algoritmo propuesto es similar al algoritmo NLMS, sin embargo cuando

$$\alpha_2 = 0.1 \quad (5.8)$$

el ERLE queda :

$$\text{ERLE} = \frac{S}{R} + 10 \log_{10}(19) \quad (5.9)$$

$$\text{ERLE} = \frac{S}{R} + 12.78 \text{ dBs} \quad (5.10)$$

La ec.(5.10) muestra una ganancia de aproximadamente 12 dBs en el ERLE comparado con el algoritmo NLMS .

### 5.1.2.-Resultados de la simulación del Algoritmo VSS-Doble paso

En la figura 5.2 se muestra la gráfica de resultados del algoritmo VSS-Doble paso, para entrada de voz1 y ruido aditivo blanco.

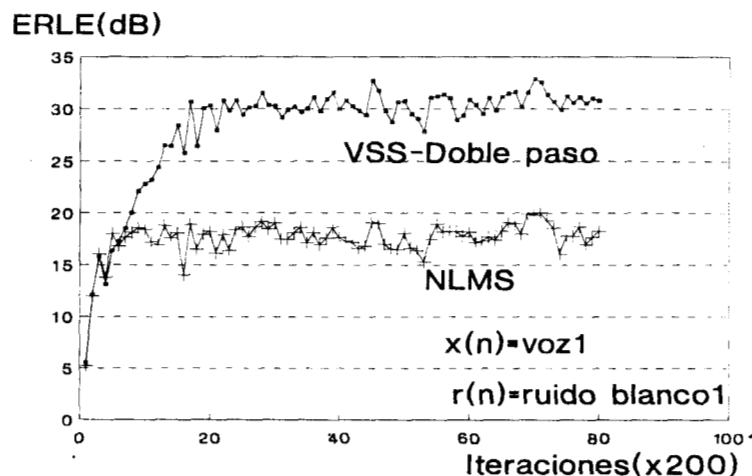


Figura 5.2.-Comparación de los algoritmos NLMS y VSS-Doble paso, para una relación S/R de 15 dBs, para entrada de voz1 y como ruido aditivo, ruido blanco1.

La figura 5.2 muestra la velocidad de convergencia de ambos algoritmos. El algoritmo VSS-Doble paso inicia con  $\alpha_1 = 1$ , tal como el algoritmo NLMS, cuando el cuadrado del error de salida es igual a un umbral propuesto  $c_1$ , el algoritmo cambia a  $\alpha_2 = 0.1$ , obteniendo así un ERLE mayor (12 dBs más que el algoritmo NLMS). El valor del umbral  $c_1$  se selecciona en base a la ecuación (4.120):

$$\text{ERLE} = 10 \log_{10} \frac{E[y^2(n)]}{E[r^2(n)]} + 10 \log_{10} \left[ \frac{2}{\alpha} - 1 \right]$$

cuando  $\alpha = 1$  el nivel de cancelación logrado por el algoritmo NLMS es igual a la relación:

$$\text{ERLE} = 10 \log_{10} \frac{E[y^2(n)]}{E[r^2(n)]}$$

como para  $\alpha < 1$  se obtiene un aumento en el ERLE, así para que el algoritmo VSS-Doble paso use  $\alpha_2$ ,  $c_1$  puede determinarse de acuerdo al nivel de cancelación logrado por el algoritmo NLMS (en la figura 5.2 se tiene un ERLE  $\approx 15$  dBs), por lo tanto  $c_1$  debe estar cercano a:

$$c_1 \approx \frac{1}{\text{antilog}(15/10)} \approx 0.0316$$

Así la velocidad de convergencia del algoritmo VSS-Doble paso es similar a la del algoritmo NLMS, obteniendo un ERLE  $\approx 30$  dBs.

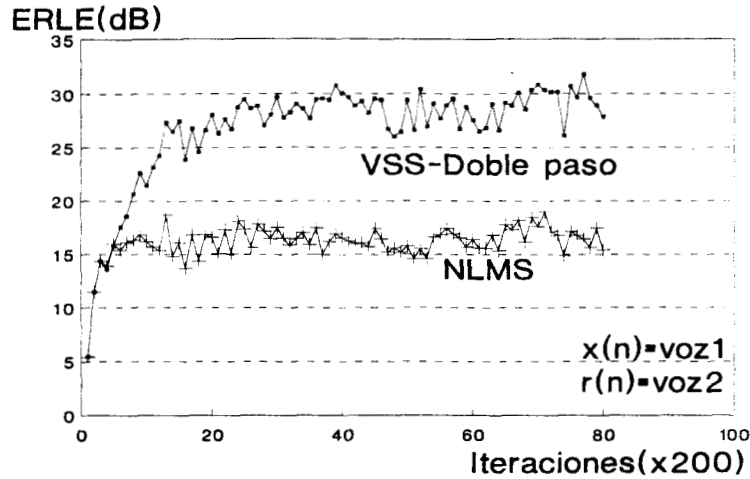


Figura 5.3.-Comparación de los algoritmos NLMS y VSS-Doble paso. para entrada de voz1 y ruido aditivo voz2, la relación S/R es de 17 dBs.

La figura 5.3 es similar a la figura 5.2, con la diferencia de que la entrada es voz1 y el ruido aditivo es voz2 . El algoritmo VSS-Doble paso usa  $\alpha_1 = 1$  y  $\alpha_2 = 0.1$ , mientras que el algoritmo NLMS usa  $\alpha = 1$ .

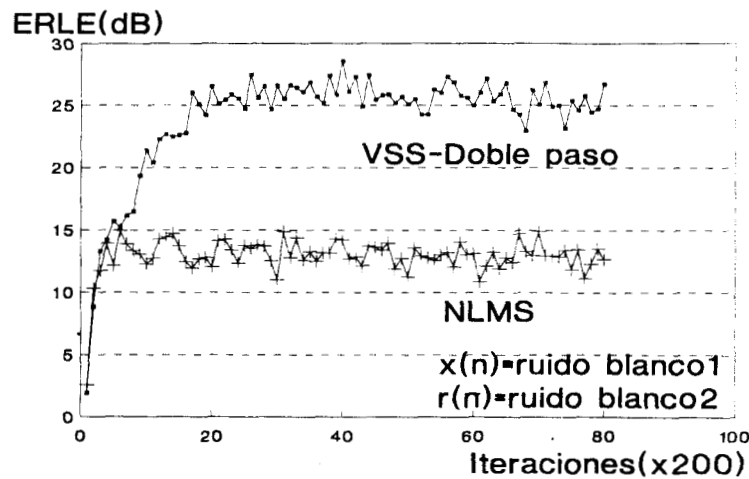


Figura 5.4.-Comparación de los algoritmos NLMS y VSS-Doble paso. para entrada de ruido blanco1 y como ruido aditivo ruido blanco2, la relación S/R es de 13 dBs.

La figura 5.4 también es similar a la figura 5.2, con la diferencia de que la entrada es ruido blanco1 y el ruido aditivo es ruido blanco2. El algoritmo VSS-Doble paso usa  $\alpha_1 = 1$  y  $\alpha_2 = 0.1$ , mientras que el algoritmo NLMS usa  $\alpha = 1$ .

Una prueba interesante de un algoritmo es la modificación de la respuesta a impulso del sistema a seguir, esto en la práctica ocurre cuando se dispone de un teléfono fijo y el usuario se aleja una cierta distancia del mismo. A continuación se tiene una prueba de este tipo.

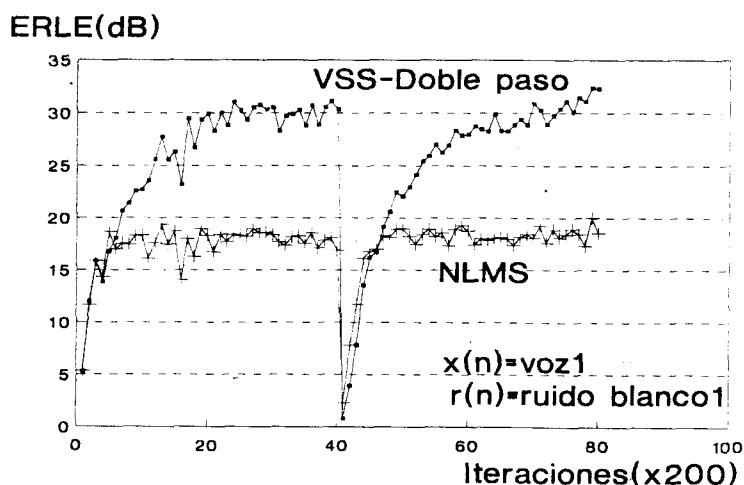


Figura 5.5.-Comparación de los algoritmos VSS-Doble paso y NLMS, la respuesta a impulso del sistema cambia súbitamente, en ambas situaciones el algoritmo propuesto obtiene un ERLE mayor que el algoritmo NLMS.

La figura 5.5 muestra la convergencia de los algoritmos también, pero en este caso la respuesta a impulso del sistema ha sido cambiada, el algoritmo VSS-Doble paso estaba usando el factor  $\alpha_2 = 0.1$  cuando repentinamente ocurre el cambio de respuesta a impulso, después de esto el algoritmo nuevamente usa  $\alpha_1 = 1$ , de esta forma el algoritmo VSS-Doble paso converge a la misma velocidad que el algoritmo NLMS y así de nuevo cuando el cuadrado del error de salida es igual al umbral  $c_1$ , el algoritmo propuesto cambia a  $\alpha_2 = 0.1$ , obteniendo así un error de salida pequeño. La relación S/R es de 18 dB al inicio.

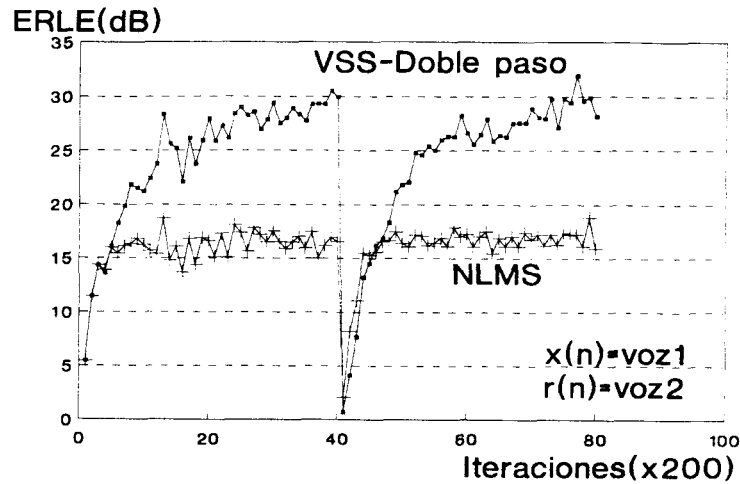


Figura 5.6.-Comparación de los algoritmos VSS-Doble paso y NLMS, la respuesta a impulso del sistema cambia súbitamente, el algoritmo propuesto obtiene un ERLE mayor que el algoritmo NLMS .

La figura 5.6 es similar a la fig. 5.5, con entrada igual a voz1 y ruido aditivo igual a voz2, en éste caso se tienen 17 dBs al inicio, en ambos casos el algoritmo VSS-Doble paso mejora al algoritmo NLMS.

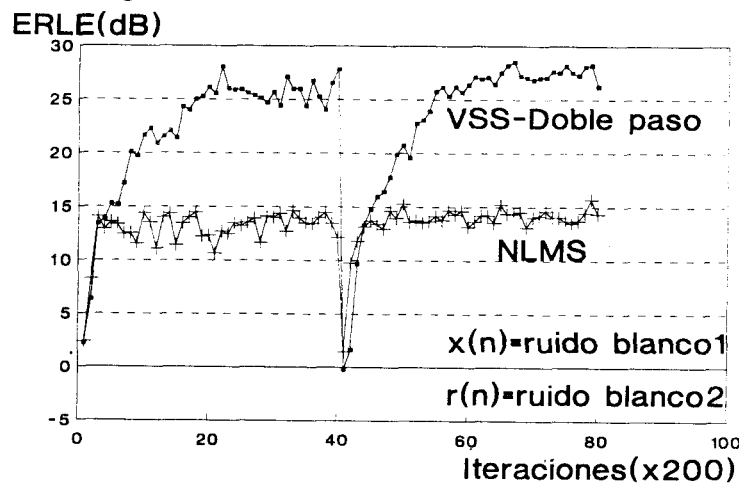


Figura 5.7.-Similar a la fig.5.6, la respuesta a impulso del sistema cambia súbitamente, el algoritmo VSS-Doble paso obtiene un ERLE mejor que el algoritmo NLMS.

En la figura 5.7 se comparan los algoritmos VSS-Doble paso y NLMS, con entrada ruido blanco1 y como ruido aditivo ruido blanco2 , en éste caso se tienen 13 dBs de

relación S/R al inicio. El algoritmo VSS-Doble paso mejora al algoritmo NLMS por 10 dBs.

Otra forma de ver el comportamiento de un algoritmo es modificando la respuesta a impulso del sistema desconocido ( $h[n]$ ), como es el caso de las figuras 5.5-5.7. Consideremos que para evaluar el algoritmo, esta variación puede ser representada por la suma de una senoide a  $h[n]$ ; de esta manera es posible observar a una muestra de la respuesta a impulso del filtro adaptable (por ejemplo  $w[50]$ ) contra la muestra equivalente de la respuesta a impulso del sistema desconocido ( $h[50]$ ), en las siguientes figuras se aprecia esto.

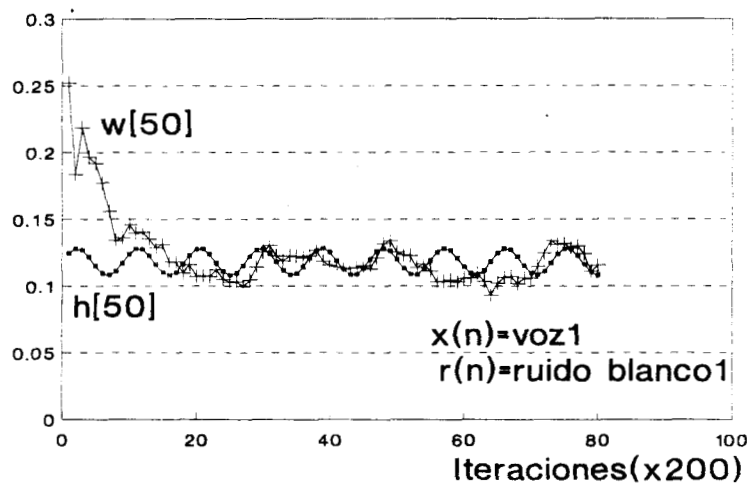


Figura 5.8.-La muestra  $w[50]$  de la respuesta a impulso del filtro adaptable es comparada con la muestra  $h[50]$  de la respuesta a impulso del sistema desconocido, usando el algoritmo VSS-Doble paso para una relación S/R de 18 dBs.La entrada es voz1 y el ruido aditivo es ruido blanco1.



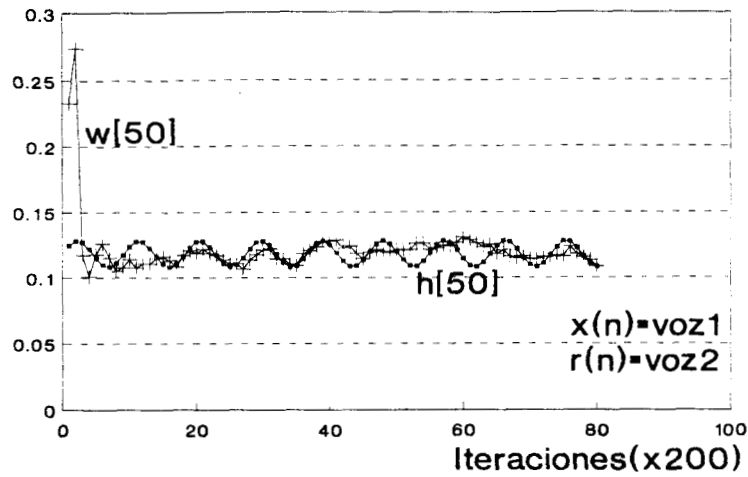


Figura 5.9.-La muestra  $w[50]$  de la respuesta a impulso del filtro adaptable es comparada con la muestra  $h[50]$  de la respuesta a impulso del sistema desconocido, usando el algoritmo VSS-Doble paso para una relación S/R de 17 dBs.La entrada es voz1 y el ruido aditivo es voz2.

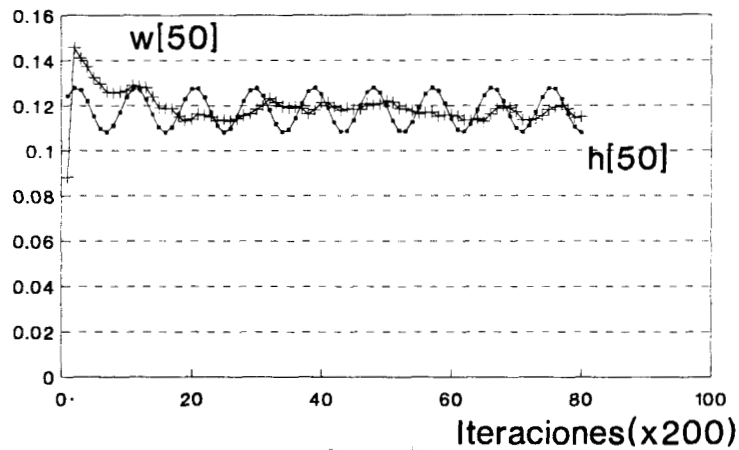


Figura 5.10.-La muestra  $w[50]$  de la respuesta a impulso del filtro adaptable es comparada con la muestra  $h[50]$  de la respuesta a impulso del sistema desconocido, usando el algoritmo VSS-Doble paso para una relación S/R de 13 dBs.La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

En las figuras 5.8, 5.9 y 5.10 la respuesta a impulso del sistema desconocido  $h[n]$  está variando con el tiempo. El algoritmo VSS-Doble paso hace que la muestra  $w[50]$  se aproxime a  $h[50]$ . Para la variación de  $h[n]$  con el tiempo, una senoide le es sumada, es decir:

$$h(n) = \begin{cases} 1 & \text{para } n = 65 \\ \left[ \frac{\sin(n\pi/c_x)}{(n\pi/c_x)} + A \sin(n\pi n_3/64) \right] & \text{para } n \neq 65 \end{cases} \quad (5.11)$$

donde  $c_x$  = Frecuencia de corte del filtro desconocido

$n$  = Variable indicadora de la variación de la respuesta a impulso del sistema desconocido

$n_3$  = Variable indicadora de la variación de la senoide para cada bloque de muestras a obtener

En la ec.(5.11) se usa  $n = 65$  para obtener la muestra central de la respuesta a impulso del sistema desconocido.

### 5.2.1- Algoritmo VSS-AC

A diferencia del algoritmo VSS-Doble paso de la sección anterior, el algoritmo VSS-AC [5.2] busca el umbral para cambiar entre  $\alpha_1$  y  $\alpha_2$ . El algoritmo VSS-AC se basa en el hecho de que cuando hay un error de salida grande, el algoritmo usa un paso grande  $\alpha_1$  para obtener un rastreo rápido; cuando el error de salida es pequeño el algoritmo usará un paso chico  $\alpha_2$ , dando lugar a una disminución en el error (fig. 5.11). Este algoritmo se comparó con el algoritmo NLMS, obteniendo un mejor desempeño del algoritmo propuesto.

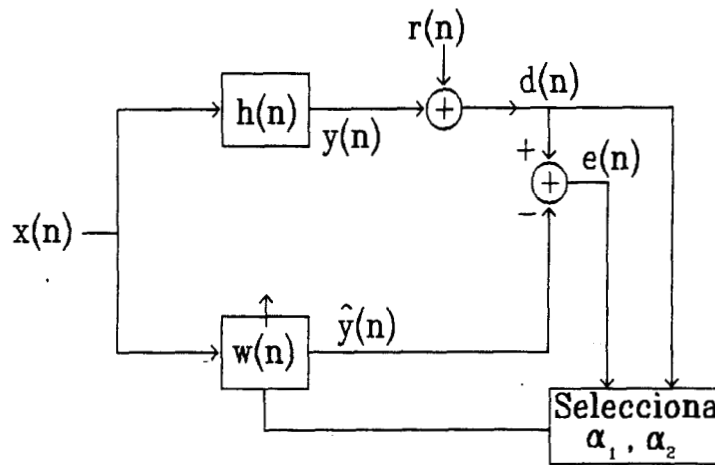


Figura 5.11.- Algoritmo VSS-AC.

El umbral  $c$  permite seleccionar entre  $\alpha_1$  y  $\alpha_2$ , si el cuadrado del error de salida es menor que  $c$ , el algoritmo usa  $\alpha_2$  (alfa pequeña). Considerando el error residual en la figura 5.11:

$$e(n) = d(n) - X^T(n)W(n) \quad (5.12')$$

El umbral en este algoritmo se obtiene a partir de la correlación entre la señal deseada  $d$  y el error  $e$ , es decir :

$$c(n+1) = \lambda_1 c(n) + \varepsilon \cdot u(n) \quad (5.12)$$

$$u(n) = E [d(n) e(n+m)] \Big|_{m=0} \quad (5.13)$$

donde  $\lambda_1$  es el factor de olvido<sup>1</sup> en la memoria y  $\varepsilon$  es el inverso de la memoria, se usan los valores

$$\lambda_1 = 0.999, \varepsilon = 0.001 \quad (5.14)$$

La ecuación (5.13) se puede escribir como :

$$E [d(n) e(n+m)] \Big|_{m=0} \approx E [(y(n)+r(n))(y(n)+r(n)-\hat{y}(n))] \quad (5.15)$$

y

$$E [y(n) r(n)] = E [r(n) y(n)] = E [r(n)\hat{y}(n)] = 0$$

Así

$$E [d(n)e(n+m)] \Big|_{m=0} = E [y^2(n)] - E [y(n)\hat{y}(n)] + E [r^2(n)] \quad (5.16)$$

Al inicio en la ecuación (5.12) la correlación  $c(n+1)$  es grande, así el algoritmo al comparar el error  $e$  con el umbral  $c$ , selecciona el paso  $\alpha_1$  (ya que no se cumple que  $e < c$ ). Cuando el algoritmo converge :

$$E [y^2(n)] \approx E [y(n)\hat{y}(n)]$$

entonces la ec. (5.16) resulta :

$$E [d(n) e(n+m)] \Big|_{m=0} \approx E [r^2(n)]$$

de esta forma  $c(n+1)$  es grande (cercana al nivel de ruido), así el algoritmo usa  $\alpha_2$ .

---

<sup>1</sup>ver apéndice I

### 5.2.2.-Resultados de la simulación del Algoritmo VSS-AC

La figura 5.12 muestra el ERLE de los algoritmos NLMS ( $\alpha = 1$  y  $\alpha = 0.05$ ) y VSS-AC; se obtiene una mejora en el ERLE de aproximadamente 13 dBs en el algoritmo VSS-AC respecto al algoritmo NLMS con  $\alpha=1$  (en éste caso se tiene una relación S/R igual a 18 dBs), con respecto al algoritmo NLMS ( $\alpha = 0.05$ ), el algoritmo VSS-AC tiene una velocidad de convergencia mayor, al final ambos algoritmos obtienen un ERLE de 32 dBs.

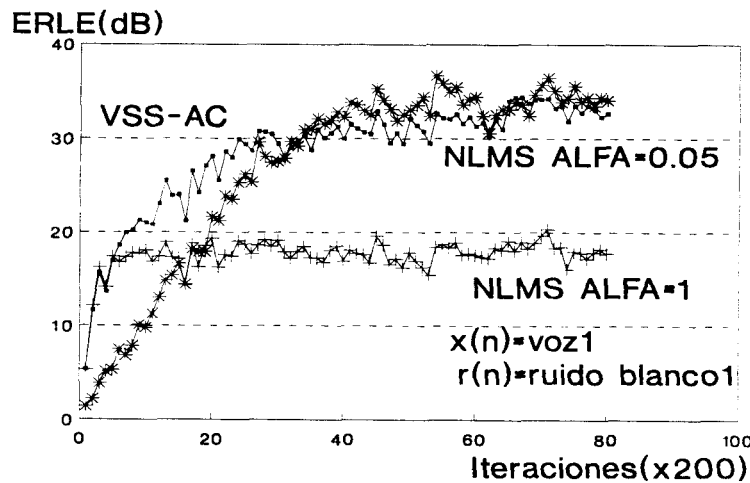


Figura 5.12.-Respuesta de los algoritmos NLMS( $\alpha=0.05,1$ ) y VSS-AC, para una relación S/R de 18 dBs.La entrada es voz1 y el ruido aditivo es ruido blanco1.

Las figuras 5.13 y 5.14 son similares a la figura 5.12, con las diferencias :

En la fig. 5.13 la entrada es voz1 y el ruido aditivo es voz2.

En la fig. 5.14 la entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

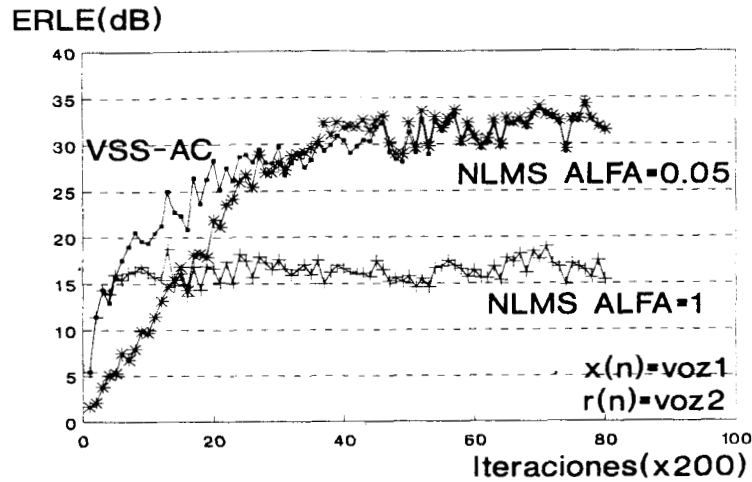


Figura 5.13.-Respuesta de los algoritmos NLMS( $\alpha=0.05,1$ ) y VSS-AC, para una relación S/R de 15 dBs.La entrada es voz1 y el ruido aditivo es voz2.

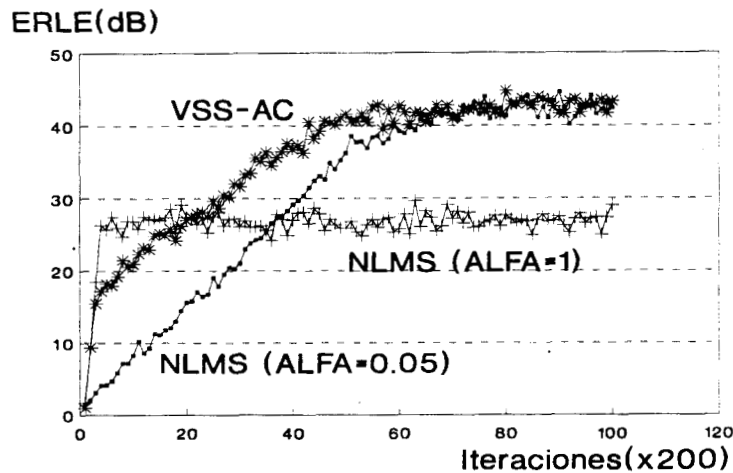


Figura 5.14.-Respuesta de los algoritmos NLMS ( $\alpha=0.05,1$ ) y VSS-AC, para una relación S/R de 28 dBs.La entrada es ruido blanco 1 y el ruido aditivo es ruido blanco2

La figura 5.15 es similar a la figura 5.14, se comparan los algoritmos NLMS ( $\alpha = 1$  y  $\alpha = 0.05$ ) y VSS-AC . La relación S/R es de 13 dBs, se obtiene una mejora de 16 dBs del algoritmo VSS-AC respecto al algoritmo NLMS ( $\alpha = 1$ ). Con respecto al algoritmo NLMS ( $\alpha = 0.05$ ), el algoritmo VSS-AC converge mas rápido, al final ambos convergen a 30 dBs.

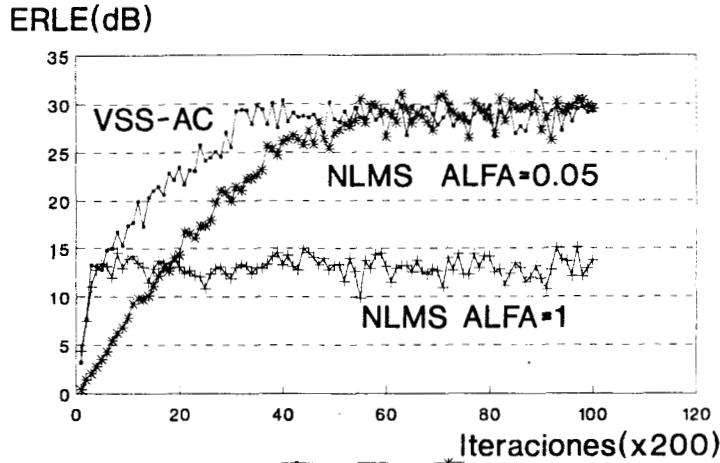


Figura 5.15.-Comparación de los algoritmos NLMS ( $\alpha=0.05,1$ ) y VSS-AC, para una relación S/R de 13 dBs.La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

La figura 5.16 muestra la convergencia de los algoritmos, para cuando la respuesta a impulso del sistema  $h(n)$  es cambiada, es decir el algoritmo NLMS ( $\alpha = 1$  y  $\alpha = 0.05$ ) y el algoritmo VSS-AC estaban trabajando, cuando de repente cambia  $h(n)$ , después de esto el algoritmo VSS-AC usa la alfa grande ( $\alpha_1$ ) así converge más rápido que como lo hacía antes del cambio. Cuando el cuadrado del error de salida del filtro adaptable es igual al umbral  $c$ , el algoritmo cambia a la alfa chica ( $\alpha_2$ ) obteniendo un ERLE mayor (en éste caso el nivel de ruido es de 16 dBs). Puede observarse que el algoritmo VSS-AC supera al algoritmo NLMS ( $\alpha = 1$  y  $\alpha = 0.05$ ), en un mayor ERLE y una mayor velocidad de convergencia respectivamente.La señal de entrada es voz1 y el ruido aditivo es voz2.

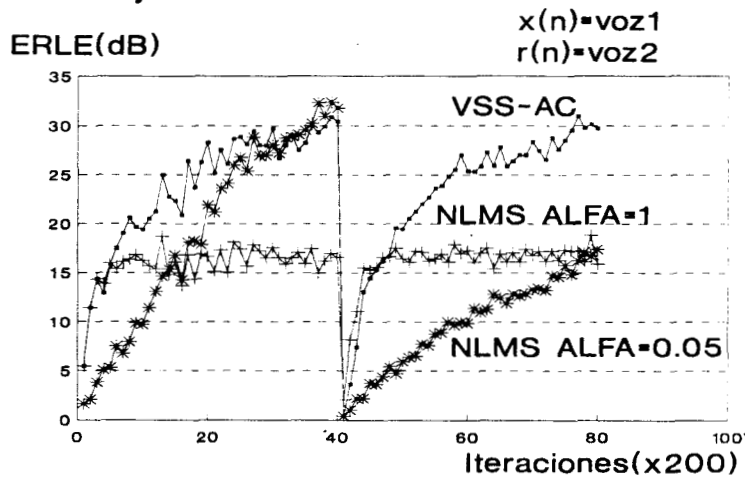


Figura 5.16.-Resultados de los algoritmos NLMS ( $\alpha=0.05,1$ ) y VSS-AC, la respuesta a impulso del sistema  $h(n)$  cambia súbitamente.La señal de entrada es voz1 y el ruido aditivo es voz2.

Las figuras 5.17 y 5.18 son similares a la figura 5.16, con las diferencias:

En la figura 5.17 la entrada es voz1 y el ruido aditivo es ruido blanco1.

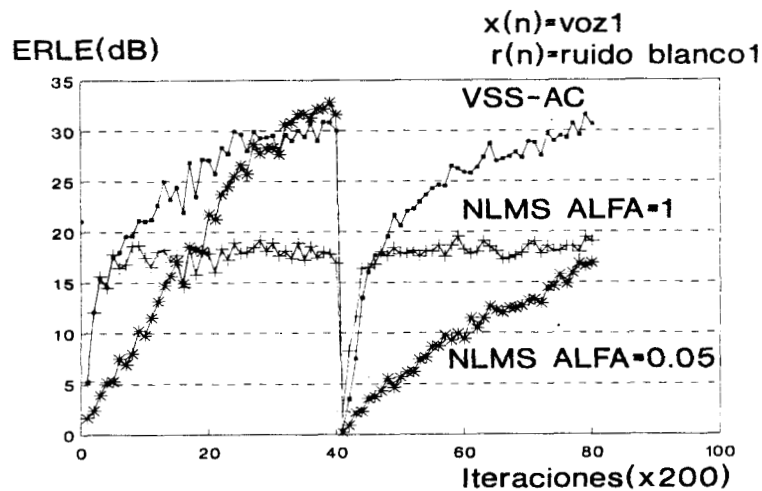


Figura 5.17.-Resultados de los algoritmos NLMS ( $\alpha=0.05,1$ ) y VSS-AC, la respuesta a impulso del sistema  $h(n)$  cambia súbitamente.La señal de entrada es voz1 y el ruido aditivo es ruido blanco1.

Otra forma de ver el comportamiento de un algoritmo es cuando se dispone de un teléfono fijo y el usuario se aleja un cierta distancia del aparato, lo anterior da lugar a una modificación en la respuesta a impulso del sistema desconocido ( $h[n]$ ). Consideremos que para evaluar el algoritmo, esta variación puede ser representada por la suma de una senoide a  $h[n]$ ; de esta manera es posible observar a una muestra de la respuesta a impulso del filtro adaptable (por ejemplo  $w[50]$ ) contra la muestra equivalente de la respuesta a impulso del sistema desconocido ( $h[50]$ ).Para la variación de  $h[n]$  con el tiempo, una senoide le es sumada, de la misma forma como en la ec.(5.11), en las figuras 5.18 y 5.19 .



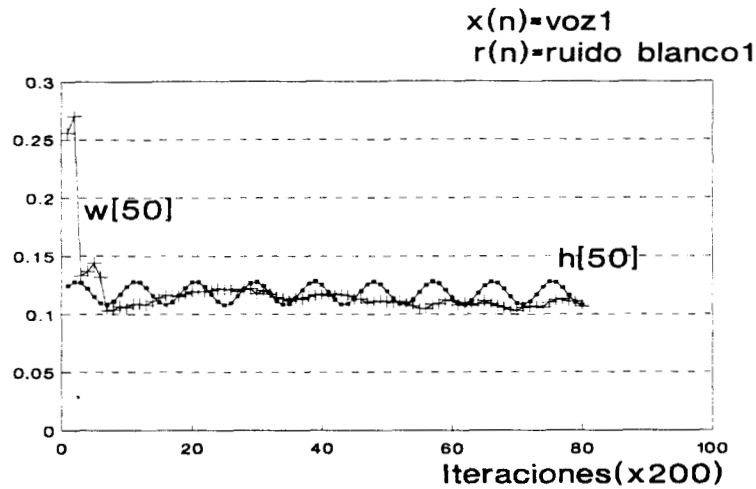


Figura 5.18.-La muestra  $w[50]$  de la respuesta a impulso del filtro adaptable es comparada con la muestra  $h[50]$  de la respuesta a impulso del sistema desconocido, usando el algoritmo VSS-AC para una relación S/R de 18 dBs.La entrada es voz1 y el ruido aditivo es ruido blanco1.

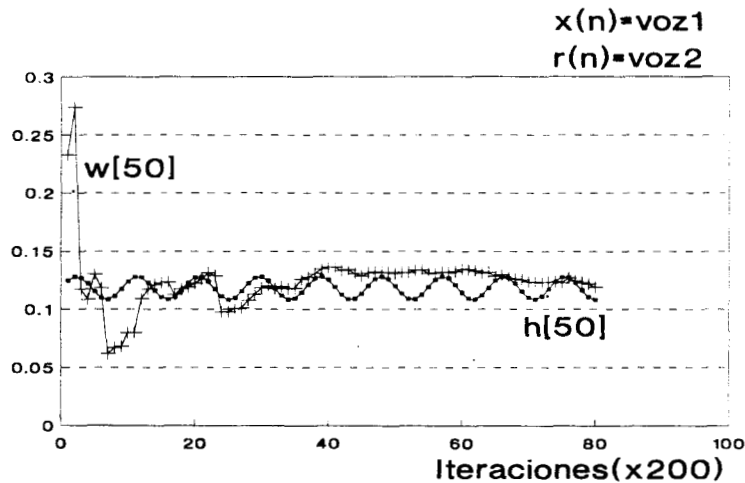


Figura 5.19.-La muestra  $w[50]$  de la respuesta a impulso del filtro adaptable es comparada con la muestra  $h[50]$  de la respuesta a impulso del sistema desconocido, usando el algoritmo VSS-AC para una relación S/R de 16 dBs.La entrada es voz1 y el ruido aditivo es voz2.

En las figuras 5.18 y 5.19 la respuesta a impulso del sistema desconocido  $h[n]$  está variando con el tiempo. El algoritmo VSS-AC hace que la muestra  $w[50]$  se aproxime a  $h[50]$ .

### 5.3.-Conclusiones

Los dos algoritmos propuestos en este capítulo, el VSS-Doble paso y VSS-AC superan al algoritmo normalizado NLMS, obteniendo un mejor ERLE en ambos casos. En el caso del algoritmo VSS-Doble paso se define al umbral  $c$  para cambio de paso en función del valor del 2o paso ( $\alpha_2$ ). Por otra parte el algoritmo VSS-AC usa dos pasos también, solo que el umbral de decisión está en función de la correlación entre la señal deseada y el error de salida. Dado que los algoritmos propuestos usan dos tamaños de paso  $\alpha_1$  y  $\alpha_2$ , resulta interesante considerar un algoritmo que varíe el tamaño del paso, de esta forma sería posible obtener un mejor ERLE; en el siguiente capítulo se describe el algoritmo VSS-CC [5.3], que hace esto.

## 5.4.-Referencias

- [5.1] Casco,F., Perez,H., "A Two Step Size LMS Algorithm", Proceedings of the International Conference on Signal Processing, vol 1, pp 543-546, October 1993, Beijing China.
- [5.2] Casco,F.,Pérez,H.,López,M.,and Marcelín,R., " A two Step Size NLMS Adaptive Filter Algorithm", IEEE Singapore International Conference on Communication Systems. ICCS'94, 14-18 November 1994, pp 814-819.
- [5.3] Casco,F.,Pérez,H.,Nakano,M., and López,M., "A Variable Step Size (VSS-CC) NLMS Algorithm", IEICE Trans.Fundamentals,vol.E78-A, No 8 August, 1995,pp 1004-1009.
- [5.4] Nagumo,J., and Noda,A., "A learning method for system identification IEEE Trans. Automat. Contr., vol. AC-12, no 3, pp 282-287, June 1967.
- [5.5] Mikhael,W., F.Hu.L.Kasovsky, G.Kang, and L. Francen, "Adaptive Filters with individual adaptation of parameters", IEEE Trans. Circuits Syst.,vol.CAS-33,no 7, pp 677-685, July 1986.
- [5.6] Harris,R.W., Chabries,D.M.,and Bishop,F.A., "A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316,Apr.1986.
- [5.7] Perez,H., Amano,F., "Acoustic Echo Cancellation Using Multirate Techniques", IEICE Transacctions, vol. E74, no 11 pp 3559-3568, Nov1991.
- [5.8] Casco,F.,Pérez,H.,López,M., "A three step size LMS algorithm", Proceedings of the 1993 IEICE FALL CONFERENCE, pp 1-84, 1-85 September 5 to 8 1993, Sapporo, Hokkaido Institute of Technology.
- [5.9] Haykin,S., Adaptive Filter Theory, Editorial Prentice Hall 1991.

## CAPITULO VI.-ALGORITMO NLMS DE PASO VARIABLE PROPUESTO VSS-CC

En este capítulo se presenta el algoritmo de paso variable VSS-CC [6.3] .Se describe su funcionamiento y se muestran los resultados obtenidos en las simulaciones del algoritmo.Las condiciones para la prueba de dicho algoritmo, son similares a las usadas en la prueba de los algoritmos presentados en el capítulo V, esto es : número de coeficientes del filtro adaptable FIR (128 derivaciones), señal de voz como señal de entrada (se usaron 2 archivos, voz1 y voz2); como ruido aditivo se usó ruido blanco (se usaron 2 archivos, ruido blanco1 y ruido blanco2).

### 6.1.-Algoritmo VSS-CC

Este algoritmo, es útil en estructuras canceladoras de eco ó ruido, el ajuste del paso  $\alpha$  es controlado por medio de la correlación entre el error de salida  $e(n)$  y la salida del filtro adaptable  $\hat{y}(n)$ . Al inicio  $e(n)$  y  $\hat{y}(n)$  estan correlacionados, lo cual hace que el algoritmo use una  $\alpha$  grande, proporcionando una velocidad de convergencia rápida. Cuando el algoritmo converge, la correlación producirá una  $\alpha$  pequeña, así el error de salida disminuirá. Para derivar este algoritmo tambien se considera la versión normalizada del algoritmo LMS :

$$W(n+1) = W(n) + \beta e(n) X(n) \quad (6.1)$$

donde

$$\beta = \frac{\alpha}{|X(n)|^2} \quad (6.2)$$

$W(n)$  es el vector de coeficientes del cancelador de ecos,  $X(n)$  es el vector de entrada,  $e(n)$  es el eco residual,  $\alpha$  es un escalar que controla la estabilidad y velocidad de convergencia ( $0 < \alpha < 2$ ) y  $|X(n)|^2$  es el valor normalizado del vector de entrada  $X(n)$  ;en la figura 6.1 se muestra el algoritmo propuesto.

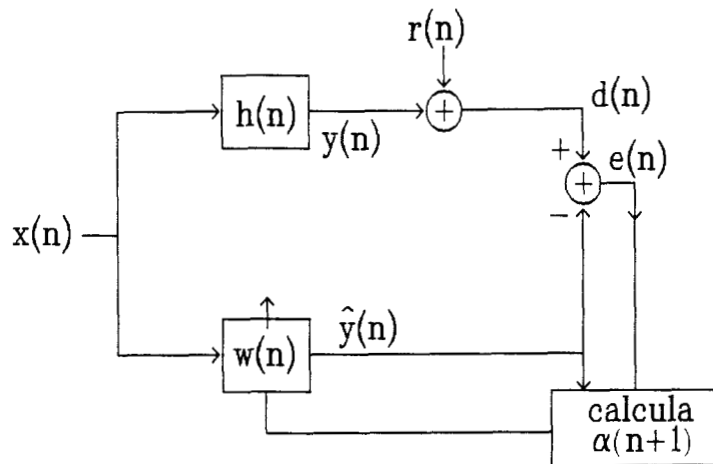


Figura 6.1.-Algoritmo VSS-CC.

El eco residual  $e(n)$  está dado por :

$$e(n) = d(n) - W^T(n) X(n) \quad (6.3)$$

$W(n)$  es el vector de coeficientes del filtro adaptable,  $X(n)$  es el vector de la señal de entrada y  $d(n)$  es la señal deseada. Después de convergencia la figura de mérito ec.(5.5) es:

$$ERLE = \frac{S}{R} + 10 \text{LOG}_{10} \left( \frac{2-\alpha}{\alpha} \right) \quad (6.4)$$

donde la S/R entre la señal de eco  $y(n)$  y el ruido aditivo  $r(n)$  (figura 6.1) está dada por :

$$\frac{S}{R} = 10 \text{LOG}_{10} \frac{\overline{y^2(n)}}{\overline{r^2(n)}} = 10 \text{LOG}_{10} u^2(n) \quad (6.5)$$

y

$$y(n) = X^T(n) H(n) \quad (6.6)$$

es la señal de eco,  $X^T(n)$  es el vector transpuesto de  $X(n)$  y  $\overline{u^2(n)}$  es el estimado de potencia de  $u(n)$  en un tiempo corto, en este caso  $u(n)$  representa a  $y(n)$  y  $r(n)$ . La ec.(6.4) indica que un ERLE igual a la relación S/R cuando el factor de convergencia  $\alpha$  es igual a 1. En el algoritmo NLMS, un factor de convergencia igual a 1 proporciona la velocidad de convergencia más rápida. Si la relación S/R es grande, un factor de convergencia igual a 1 es la mejor selección, sin embargo cuando la

relación S/R disminuye, el ERLE también disminuye a menos que un factor de convergencia más pequeño sea usado.

El algoritmo VSS-CC usa una  $\alpha$  variable la cual es 1 ó está cerca de éste valor, cuando empieza la adaptación, entonces al inicio en la ec.(6.4) si

$$\alpha_1 = 1 \quad (6.7)$$

así el ERLE resulta como :

$$\text{ERLE} = \frac{S}{R} \quad (6.8)$$

en éste caso el comportamiento del algoritmo propuesto es similar al algoritmo NLMS, sin embargo cuando

$$\alpha_2 = 0.05 \quad (6.9)$$

el ERLE queda :

$$\text{ERLE} = \frac{S}{R} + 10 \log_{10}(39) \quad (6.10)$$

$$\text{ERLE} = \frac{S}{R} + 16 \text{ dBs} \quad (6.11)$$

La ec.(6.11) muestra una ganancia de 16 dBs en el ERLE comparado con el algoritmo NLMS . El algoritmo VSS-CC usa una  $\alpha$  variable cuyo valor es estimado por un criterio de correlación ( $R_{e\hat{y}}$ ) entre el error de salida ( $e(n)$ ) y la salida del filtro adaptable ( $\hat{y}(n)$ ), donde :

$$R_{e\hat{y}} = E[e(n) \hat{y}(n+m)]_{m=0} \quad (6.12)$$

es el primer elemento de la correlación cruzada entre  $e(n)$  y  $\hat{y}(n)$ , en la ecuación (6.12)  $E[e(n) \hat{y}(n+m)]_{m=0}$  es aproximadamente :

$$E[e(n) \hat{y}(n+m)]_{m=0} \cong E[e(n) \hat{y}(n)]$$

$$\begin{aligned} &\cong E[(y(n) - \hat{y}(n) + r(n))\hat{y}(n)] \\ &\cong E[y(n)\hat{y}(n)] - E[\hat{y}^2(n)] + E[r(n)\hat{y}(n)] \end{aligned} \quad (6.13)$$

Al inicio en la ecuación (6.13) el valor de la correlación  $E[e(n)\hat{y}(n)]$  es grande, así que el algoritmo usa una  $\alpha$  grande. Cuando el algoritmo converge ( $y(n) \cong \hat{y}(n)$ ), la ecuación (6.13) es aproximadamente :

$$E[e(n)\hat{y}(n)] \cong E[r(n)\hat{y}(n)] \quad (6.14)$$

De esta forma el algoritmo usa una  $\alpha$  pequeña, así se obtiene un error residual menor. Cuando el algoritmo ha convergido, el valor de  $\alpha(n+1)$  es pequeño debido a que  $r(n)$  y  $\hat{y}(n)$  en la ecuación (6.14) no están correlacionadas (aún con  $r(n)$  grande).

El algoritmo VSS-CC propuesto se muestra en la figura 6.1, la ecuación para estimar  $\alpha$  en dicho algoritmo es :

$$\alpha(n+1) = \begin{cases} \alpha_{MAX} & \text{si } \alpha'(n+1) > \alpha_{MAX} \\ \alpha_{MIN} & \text{si } \alpha'(n+1) < \alpha_{MIN} \\ \alpha'(n+1) & \text{de otra forma} \end{cases} \quad (6.15)$$

y

$$\hat{R}_{e\hat{y}}(n+1) = \lambda_1 \hat{R}_{e\hat{y}}(n) + \varepsilon[e(n)\hat{y}(n)] \quad (6.16)$$

$$\alpha'(n+1) = \frac{\hat{R}_{e\hat{y}}(n+1)}{x^2(n+1)} \quad (6.17)$$

$$\overline{x^2}(n+1) = \lambda_1 \overline{x^2}(n) + \varepsilon[x^2(n)] \quad (6.18)$$

En la ecuación (6.17)  $x(n)$  es la señal de entrada y  $\hat{R}_{e\hat{y}}(n)$  es el valor estimado de  $R_{e\hat{y}}(n)$  de la ecuación (6.12) donde  $\frac{1}{\varepsilon}$  es<sup>1</sup> aproximadamente igual al número de muestras  $N$ , usado para estimar los promedios :

---

<sup>1</sup>ver apéndice I

$$\varepsilon = \frac{1}{N} , \quad \lambda_1 \leq 1 - \frac{1}{N} \quad (6.19)$$

$\alpha_{MAX}$  es igual a 1 para asegurar una convergencia rápida y a  $\alpha_{MIN}$  se le da un valor fijo para evitar factores de convergencia demasiado pequeños ( una vez que la convergencia ha sido lograda), algunos valores adecuados son :

$$\lambda_1 = 0.997 , \quad \varepsilon = 0.0048 \quad (6.20)$$

y

$$e(n) = y(n) - \hat{y}(n) + r(n) \quad (6.21)$$

En la ecuación (6.21)  $\hat{y}(n)$  es el valor estimado de  $y(n)$  y  $r(n)$  es el ruido aditivo. Al principio en la ecuación (6.16) la correlación entre  $e(n)$  y  $\hat{y}(n)$  es grande, debido a que  $(y(n) - \hat{y}(n))$  es grande, en consecuencia  $\alpha$  también es grande. Cuando el algoritmo converge, la correlación  $|R_{e\hat{y}}(0)|$  se debilita, así  $\alpha$  viene a ser más pequeño haciendo que el algoritmo sea más sensible al ruido aditivo (se supone que la señal de entrada y el ruido aditivo no están correlacionados).

## 6.2.-Comparación del Algoritmo VSS-CC con otros Algoritmos de paso variable

Varios algoritmos LMS de paso variable han sido propuestos para superar las limitaciones del algoritmo LMS reduciendo el error residual, sin embargo no se reduce la razón de convergencia significativamente. En esta sección se compara el comportamiento del algoritmo VSS-CC con los algoritmos LMS de paso variable VSLMS y VSS del capítulo anterior.

### VSS-CC vs VSLMS

En el algoritmo VSLMS [6.1] se proponen factores de convergencia individuales, los cuales son calculados para cada peso. Estos factores de convergencia son incrementados si  $M$  cambios de signo suceden, en caso contrario los factores son disminuidos. Si  $M$  cambia el signo del gradiente, ocurre que el paso es disminuido alguna cantidad, es aumentado si durante los  $M$  períodos de muestreo no hay cambio en el signo del gradiente. El paso permanece sin cambiar en cualquier otra situación,



así el algoritmo tiene una buena velocidad de convergencia. Sin embargo su habilidad de rastreo es en general pobre, debido a que el paso se incrementará gradualmente después de que un cambio abrupto ocurra en las características de la señal de referencia; otra desventaja de este algoritmo es que el signo del gradiente depende mucho del ruido aditivo, particularmente para una baja relación de señal a ruido [6.2]. Dado que la referencia del artículo [6.2] reporta tener una mejor convergencia que el algoritmo VSLMS de la referencia [6.1], no se usó al algoritmo VSLMS en las comparaciones con el algoritmo propuesto VSS-CC.

### VSS-CC vs VSS

El algoritmo VSS [6.2], tiene un factor de convergencia que es proporcional a la potencia del error de salida. En este algoritmo el paso se estima como :

$$\alpha(n+1) = \begin{cases} \alpha_{MAX} & \text{si } \alpha'(n+1) > \alpha_{MAX} \\ \alpha_{MIN} & \text{si } \alpha'(n+1) < \alpha_{MIN} \\ \alpha'(n+1) & \text{de otra forma} \end{cases} \quad (6.22)$$

donde

$$\alpha'(n+1) = \lambda_1 \alpha(n) + \varepsilon e^2(n) \quad (6.23)$$

con<sup>2</sup>  $0 < \lambda_1 < 1$ ,  $0 < \varepsilon < 1$ . Aquí  $\alpha_{MAX}$  es útil para asegurar que el algoritmo permanece estable y  $\alpha_{MIN}$  se fija para evitar un factor de convergencia demasiado pequeño, después de que el algoritmo ha convergido. Sin embargo el algoritmo VSS aún puede presentar problemas cuando la relación señal a ruido disminuye significativamente, esto es cuando el ruido aditivo aumenta, en esta situación el paso toma un valor muy cercano a  $\alpha_{MAX}$ , esto puede provocar degradación en el sistema como se indica en la ecuación (6.4).

En el algoritmo propuesto VSS-CC, el paso depende de la correlación entre la salida del filtro adaptable y el error de salida. De esta forma si la respuesta no varía, la correlación será pequeña sin depender de la potencia del ruido aditivo. Esta característica es la diferencia en el mejor comportamiento del algoritmo VSS-CC con respecto a VSS.

---

<sup>2</sup>ver apéndice I

### 6.3.-Resultados de la simulación del Algoritmo VSS-CC

En el algoritmo LMS [6.5],  $\beta$  está dada por la ecuación (6.2) con  $\alpha$  una constante. En el algoritmo LMS de paso variable VSS [6.2]  $\alpha$  es estimada a través de las fórmulas (6.22) y (6.23) con ( $\lambda_1=0.997$  y  $\varepsilon=4.8 \times 10^{-4}$ ). El algoritmo propuesto VSS-CC es comparado con los algoritmos NLMS y VSS de paso variable, en los resultados de simulación se usa el ERLE definido por la ecuación (5.5), éste fue estimado como :

$$ERLE = 10 \log_{10} x_3 \quad (6.24)$$

donde

$$x_3 = \frac{y_2(n+1)}{e_2(n+1)} \quad (6.25)$$

con

$$e_2(n+1) = \lambda_1 e_2(n) + \varepsilon e^2(n) \quad (6.26)$$

$$y_2(n+1) = \lambda_1 y_2(n) + \varepsilon d^2(n) \quad (6.27)$$

donde  $\lambda_1$  y  $\varepsilon$  son iguales a los usados en la ecuación (6.18);  $e(n)$  y  $d(n)$  se muestran en la figura (6.1), así  $x_3$  de la ecuación (6.26) se estimó en forma recursiva.

La respuesta de los tres algoritmos NLMS, VSS y VSS-CC propuesto, se muestra en la figura 6.2.

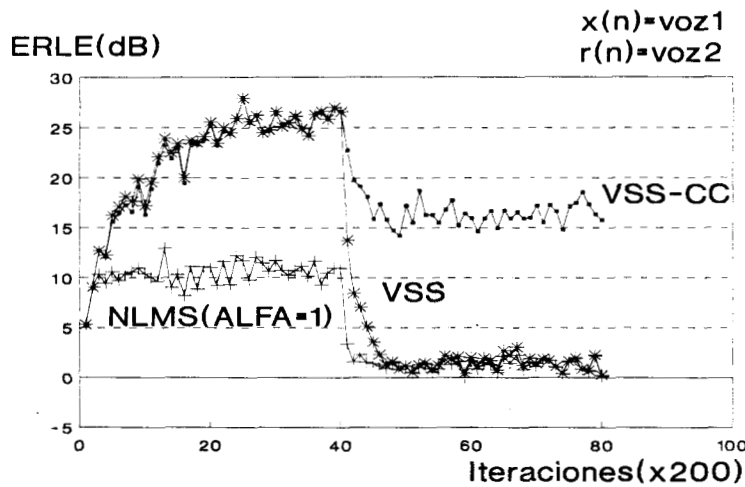


Figura 6.2.-Comparación de los algoritmos NLMS, VSS de paso variable y VSS-CC de paso variable propuesto. Al inicio se tiene una relación S/R de 10 dBs, posteriormente la S/R es de 2 dBs. La entrada es voz1 y el ruido aditivo es voz2.

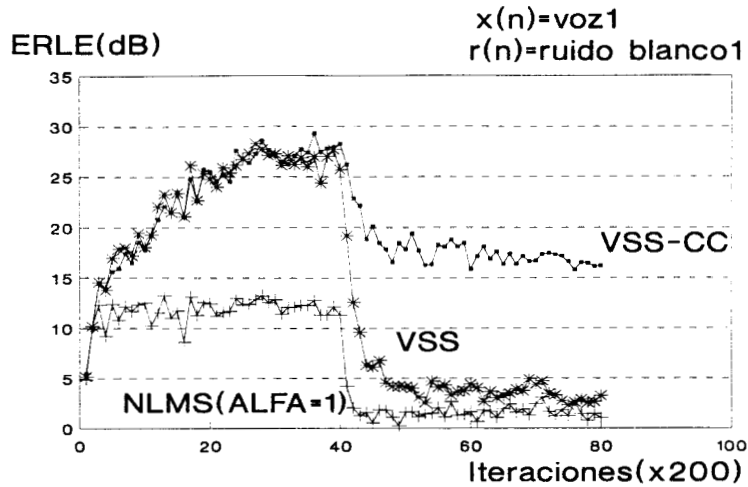


Figura 6.3.-Comparación de los algoritmos NLMS, VSS de paso variable y VSS-CC de paso variable propuesto. Al inicio se tiene una relación S/R de 13 dBs, posteriormente la S/R es de 3 dBs. La entrada es voz1 y el ruido aditivo es ruido blanco1.

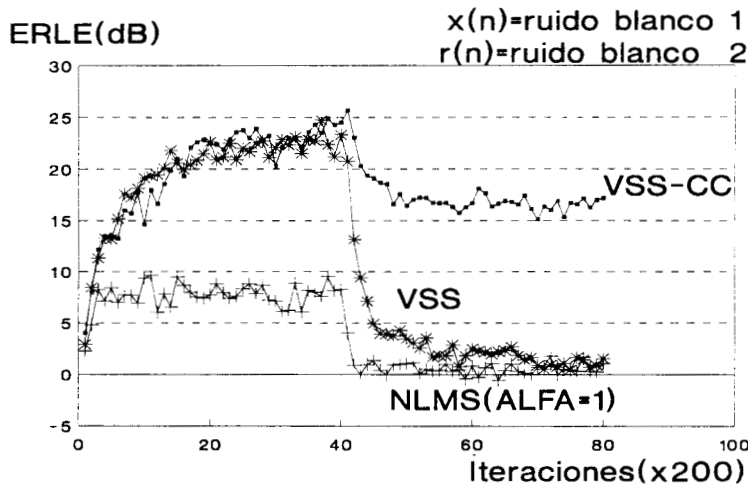


Figura 6.4.-Comparación de los algoritmos NLMS, VSS de paso variable y VSS-CC de paso variable propuesto. Al inicio se tiene una relación S/R de 8 dBs, posteriormente la relación S/R es de 0 dBs. La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

En las figuras 6.2, 6.3 y 6.4 el algoritmo propuesto VSS-CC inicia con una  $\alpha$  grande como lo hacen los algoritmos NLMS y VSS. Al inicio los tres algoritmos tienen el mismo factor de convergencia. Después de varias iteraciones la relación S/R es disminuída y el algoritmo propuesto VSS-CC obtiene un ERLE mayor que el algoritmo VSS, el algoritmo VSS converge de manera similar al algoritmo NLMS , bajo las mismas condiciones.

La figuras 6.5,6.6 y 6.7 muestran la respuesta de los algoritmos NLMS, VSS y VSS-CC, para diferentes condiciones de trabajo

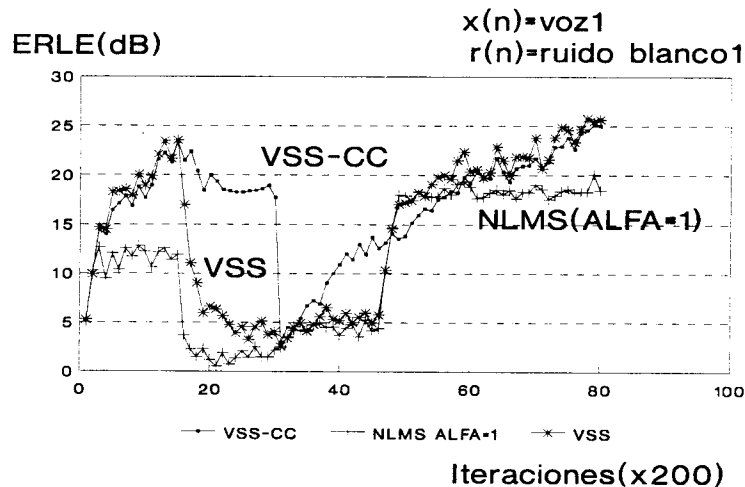


Figura 6.5.-Comparación de los algoritmos NLMS , VSS y VSS-CC para 4 períodos : (I) S/R de 12 dBs, (II) S/R 2 dBs, (III) La respuesta a impulso del sistema es cambiada y S/R de 4 dBs, (IV) S/R de 18 dBs.La entrada es voz1 y el ruido aditivo es ruido blanco1.

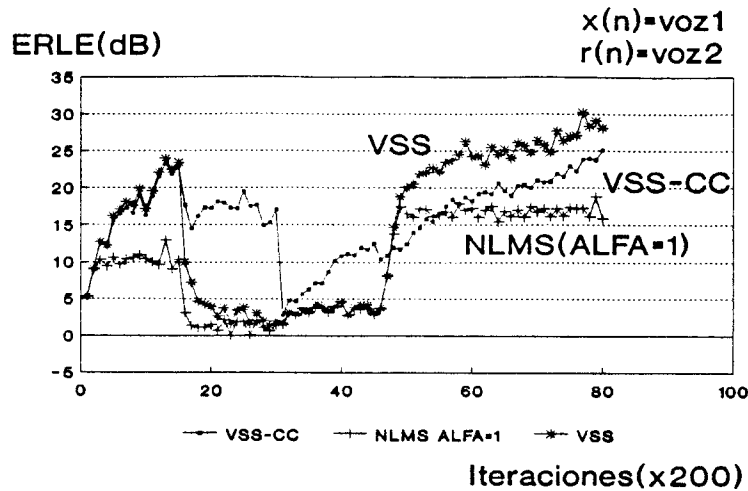


Figura 6.6.-Comparación de los algoritmos NLMS , VSS y VSS-CC para 4 períodos : (I) S/R de 10 dBs, (II) S/R 3 dBs, (III) La respuesta a impulso del sistema es cambiada y S/R de 4 dBs, (IV) S/R de 16 dBs.La entrada es voz1 y el ruido aditivo es voz2.

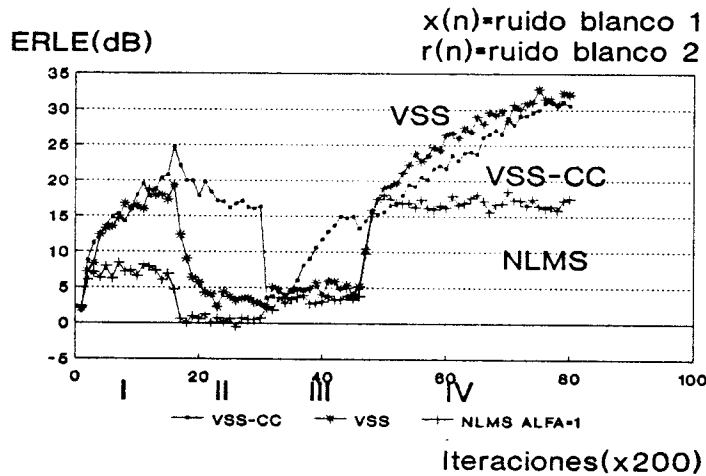


Figura 6.7.-Comparación de los algoritmos NLMS , VSS y VSS-CC para 4 períodos : (I) S/R de 8 dBs, (II) S/R 0 dBs, (III) La respuesta a impulso del sistema es cambiada y S/R de 4 dBs, (IV) S/R de 16 dBs.La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

En las figuras 6.5,6.6 y 6.7 en el período (I) ambos algoritmos VSS y VSS-CC tienen la misma razón de convergencia y obtienen el mismo ERLE, mientras que el algoritmo NLMS converge a 12,10 y 8 dBs respectivamente. Después de varias iteraciones (período II), la relación S/R es cambiada, el algoritmo propuesto VSS-CC obtiene un mayor ERLE que el algoritmo VSS. En el período (III) la respuesta del sistema es cambiada.El algoritmo VSS-CC estaba usando una  $\alpha$  pequeña, cuando de repente ocurre el cambio y la correlación entre  $\hat{y}(n)$  y  $e(n)$  se incrementa y en consecuencia  $\alpha$  también. Lo anterior indica que el algoritmo sigue a la variación de la respuesta a impulso del sistema, aunque la razón de convergencia es menor. Después de esto, el algoritmo usa una  $\alpha$  grande para lograr una convergencia rápida. Finalmente en el período (IV), la relación S/R es cambiada de nuevo ,en este caso ambos algoritmos VSS y VSS-CC obtienen mejor ERLE que el algoritmo NLMS, cabe mencionar que en este caso el algoritmo VSS obtiene un mejor ERLE que el algoritmo VSS-CC (figura 6.6).

Las figuras 6.8, 6.9 y 6.10 muestran también la respuesta de los algoritmos NLMS VSS y VSS-CC , para variaciones de la respuesta a impulso  $h(n)$  a seguir.

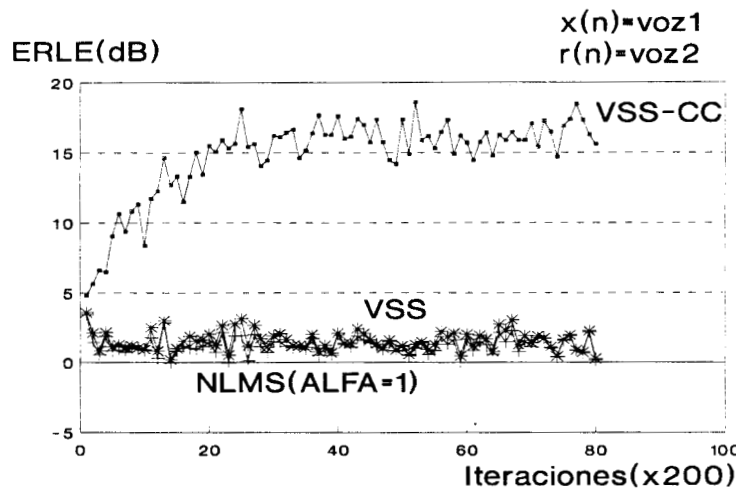


Figura 6.8.-Comparación de los algoritmos NLMS , VSS y VSS-CC, la respuesta a impulso de la trayectoria de eco  $h[n]$ , está variando con el tiempo, para una relación S/R de 2 dBs.La entrada es voz1 y el ruido aditivo es voz2.

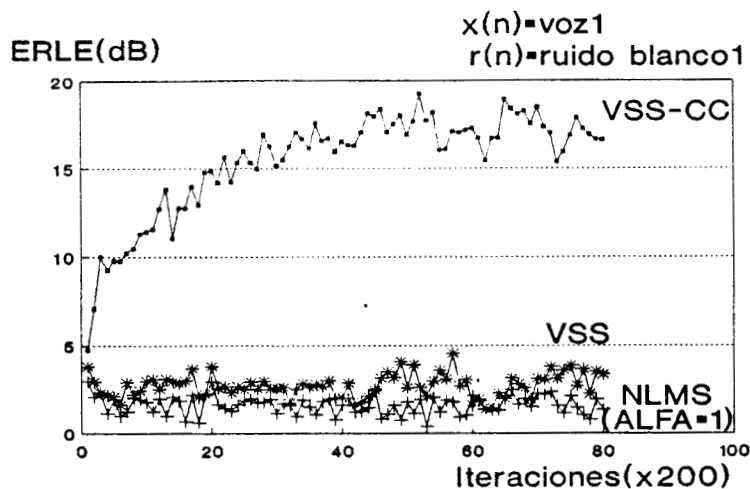


Figura 6.9.-Comparación de los algoritmos NLMS , VSS y VSS-CC, la respuesta a impulso de la trayectoria de eco  $h[n]$ , está variando con el tiempo, para una relación S/R de 3 dBs.La entrada es voz1 y el ruido aditivo es ruido blanco1.

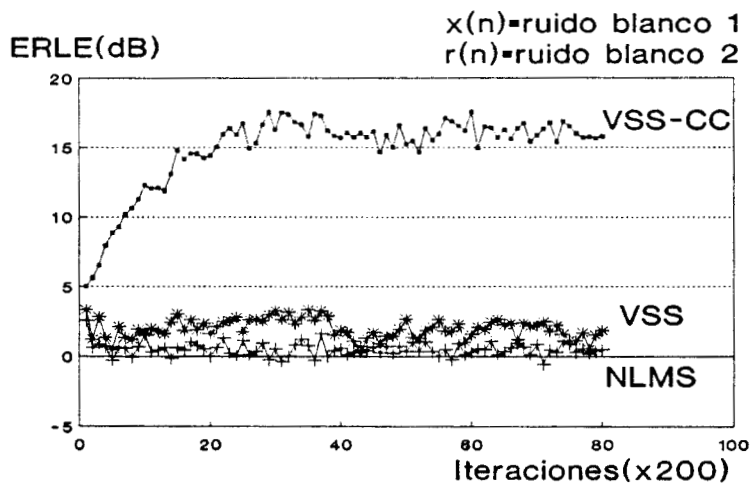


Figura 6.10.-Comparación de los algoritmos NLMS , VSS y VSS-CC, la respuesta a impulso de la trayectoria de eco  $h[n]$ , está variando con el tiempo, para una relación S/R de 0 dBs.La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

En las figuras 6.8,6.9 y 6.10 la respuesta a impulso del sistema desconocido  $h[n]$  está variando con el tiempo.El algoritmo VSS-CC logra un mejor ERLE que los algoritmos VSS y NLMS (la relación S/R es baja).

En las figuras 6.11,6.12 y 6.13 la respuesta a impulso de la trayectoria de eco  $h[n]$  está variando con el tiempo , con una relación S/R baja (condiciones similares a las figuras 6.8,6.9 y 6.10 respectivamente). El algoritmo VSS-CC hace que la muestra  $w[50]$  se aproxime a  $h[50]$ . Para la variación de  $h[n]$  con el tiempo, una senoide le es sumada, de la misma forma como en la ec.(5.11).

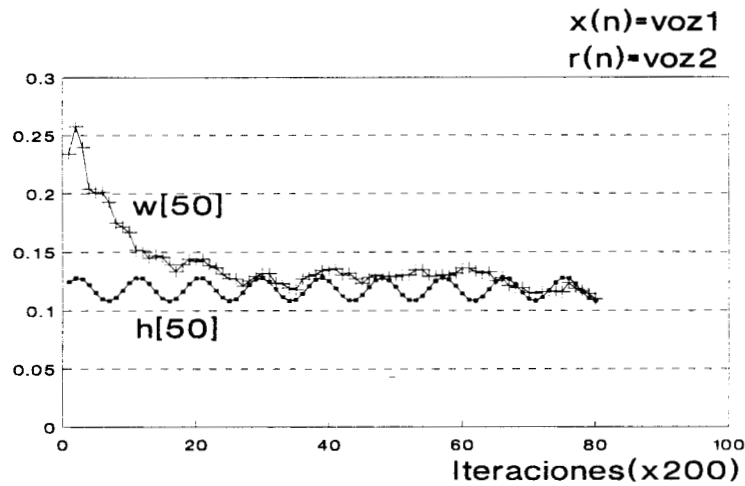


Figura 6.11.-La muestra  $w[50]$  de la respuesta a impulso adaptativa es comparada con la muestra  $h[50]$  , usando el algoritmo VSS-CC.La entrada es voz1 y el ruido aditivo es voz2.

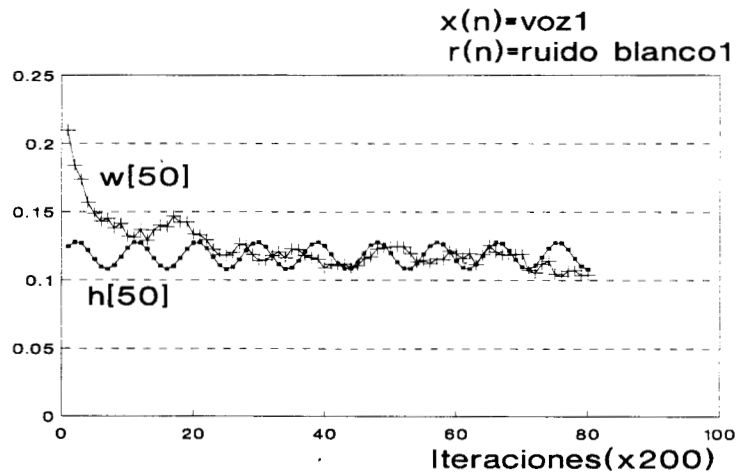


Figura 6.12.-La muestra  $w[50]$  de la respuesta a impulso adaptativa es comparada con la muestra  $h[50]$  , usando el algoritmo VSS-CC . La entrada es voz1 y el ruido aditivo es ruido blanco1.



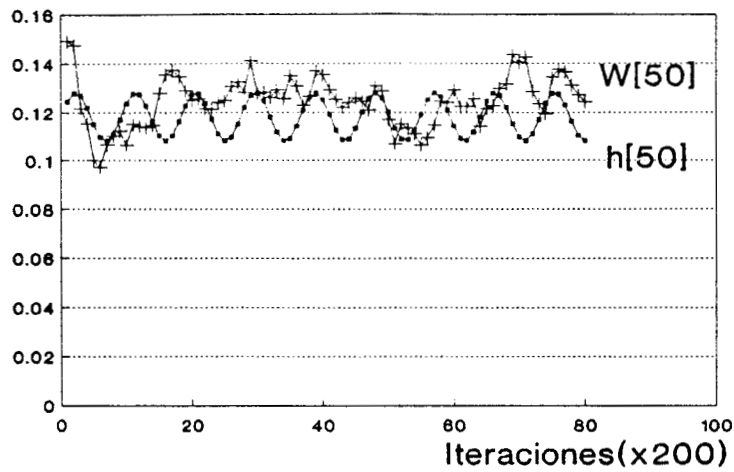


Figura 6.13.-La muestra  $w[50]$  de la respuesta a impulso adaptiva es comparada con la muestra  $h[50]$ , usando el algoritmo VSS-CC. La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

En todas las pruebas del algoritmo VSS-CC  $\alpha_{\text{MAX}} = 1$ ,  $\alpha_{\text{MIN}} = 0.05$ .

#### 6.4.-Comparación del Algoritmo VSS-CC con los algoritmos VSS-Doble paso y VSS-AC.

Son 3 los algoritmos propuestos: VSS-Doble paso, VSS-AC (de paso variable seleccionando el umbral por correlación) y VSS-CC (de paso variable estimando el paso por correlación). Para establecer una comparación entre ellos recurrimos a las tres Tablas mostradas a continuación.

Algoritmo Propuesto	Relación S/R en dBs	ERLE alcanzado por el algoritmo propuesto	Mejora en ERLE (dBs) del algoritmo respecto al NLMS	Figura
VSS-Doble paso ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	17	27	10	5.3
VSS-AC ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	17	30	13	5.12
VSS-CC	2	16	14	6.2

Tabla I.-Comparación de Algoritmos propuestos, la velocidad de convergencia fue de 200 iteraciones por muestra y la referencia en las comparaciones es el algoritmo NLMS ( $\alpha = 1$ ).La entrada es voz1 y el ruido aditivo es voz2.

Algoritmo Propuesto	Relación S/R en dBs	ERLE alcanzado por el algoritmo propuesto	Mejora en ERLE (dBs) del algoritmo respecto al NLMS	Figura
VSS-Doble paso ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	17	30	13	5.2
VSS-AC ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	18	32	12	5.12
VSS-CC	2	16	14	6.3

Tabla II.-Comparación de Algoritmos propuestos, la velocidad de convergencia fue de 200 iteraciones por muestra y la referencia en las comparaciones es el algoritmo NLMS ( $\alpha = 1$ ).La entrada es voz1 y el ruido aditivo es ruido blanco1.

Algoritmo Propuesto	Relación S/R en dBs	ERLE alcanzado por el algoritmo propuesto	Mejora en ERLE (dBs) del algoritmo respecto al NLMS	Figura
VSS-Doble paso ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	13	25	12	5.4
VSS-AC ( $\alpha_1 = 1, \alpha_2 = 0.05$ )	28	42	14	5.14
VSS-CC	0	16	16	6.4

Tabla III.-Comparación de Algoritmos propuestos, la velocidad de convergencia fue de 200 iteraciones por muestra y la referencia en las comparaciones es el algoritmo NLMS ( $\alpha = 1$ ).La entrada es ruido blanco1 y el ruido aditivo es ruido blanco2.

De las tablas I,II y III los valores mostrados estan indicados con respecto al algoritmo NLMS , en donde se obtiene una mejora en el ERLE. Por ejemplo con el algoritmo VSS-Doble paso se obtiene una mejora de 10,13 y 12 dBs respectivamente. Tambien en el algoritmo VSS-AC se obtienen 13,12 y 14 dBs respectivamente por arriba del algoritmo NLMS . Finalmente con el algoritmo VSS-CC es con el que se obtiene un mejor ERLE ( 14,14 y 16 dBs respectivamente).

En la tabla IV se muestra la comparación de los algoritmos VSS y VSS-CC para dos valores de relación S/R.

Algoritmo	Relación S/R en dBs	ERLE alcanzado por el algoritmo	Figura
VSS-	10	25	6.2
VSS-CC	10	25	6.2
VSS	2	2	6.2
VSS-CC	2	16	6.2

Tabla IV.-Comparación de Algoritmos VSS y VSS-CC, para 2 valores de relación S/R (10 y 2 dBs), el algoritmo propuesto VSS-CC supera al algoritmo VSS para la relación S/R de 2 dBs. La entrada es voz1 y el ruido aditivo es voz2.

De la tabla IV se puede ver que el algoritmo VSS-CC supera al algoritmo y VSS [6.2].

## **6.5.-Conclusiones**

En este capítulo se ha presentado al algoritmo VSS-CC. En todas las comparaciones realizadas contra los algoritmos NLMS y VSS, el algoritmo VSS-CC propuesto tiene un mejor comportamiento, aún cuando en el período IV de la figura 6.6 para relación S/R de 17 dBs el algoritmo VSS mejora en 4 dBs al algoritmo VSS-CC, el algoritmo propuesto es mejor ya que en la figura 6.8 para condiciones de entrada similares (con una relación S/R de 3 dBs) el algoritmo VSS-CC obtiene 15 dBs mas que el algoritmo VSS.

Básicamente las características del algoritmo VSS-CC son : su velocidad de convergencia ,bajo error residual y buen funcionamiento aún en condiciones de baja relación S/R. Esta última cualidad del algoritmo, lo hace útil para el trabajo bajo condiciones severas de ruido.

## 6.6.-Referencias

[6.1] Harris,R.W., Chabries,D.M.,and Bishop,F.A.,"A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316,Apr.1986.

[6.2] Raymond ,H., Kwong,E., and Johnston,W.,"A Variable Step Size LMS Algorithm ",IEEE Transactions on Signal Processing,vol.40 No.7 pp. 1633-1642,July 1992.

[6.3] Casco,F.,Pérez,H.,Nakano,M., and López,M.,"A Variable Step Size (VSS-CC) NLMS Algorithm", IEICE Trans.Fundamentals,vol.E78-A, No 8 August, 1995,pp 1004-1009.

[6.4] Widrow,B., McCool,J.M.,Larimore,M.G.,and Johnson,C.R.,,"Stationary and Non-Stationary Learning Characteristics of the LMS Adaptive Filter",Proceedings IEEE,Vol. 64,No 8,pp.1151-1161, August 1976.

[6.5] Mikhael,W., F.Hu.L.Kasovsky, G.Kang, and L. Francen, "Adaptive Filters with individual adaptation of parameters", IEEE Trans. Circuits Syst.,vol.CAS-33,no 7, pp 677-685, July 1986.

[6.6] Perez,H., Amano,F., "Acoustic Echo Cancellation Using Multirate Techniques", IEICE Transacctions, vol. E74, no 11 pp 3559-3568, Nov1991.

[6.7] Casco,F.,Pérez,H.,López,M.,"A three step size LMS algorithm",Proceedings of the 1993 IEICE FALL CONFERENCE, pp 1-84, 1-85 September 5 to 8 1993, Sapporo, Hokkaido Institute of Technology.

[6.8] Casco,F., Perez,H., "A Two Step Size LMS Algorithm", Proceedings of the International Conference on Signal Processing, vol 1, pp 543-546, October 1993, Beijing China.

[6.9] Haykin,S., Adaptive Filter Theory, Editorial Prentice Hall 1991.

## VII.-PERSPECTIVAS PARA CONTINUAR EN ESTA LINEA DE INVESTIGACION

La idea central en el algoritmo propuesto fue la de usar mas de un tamaño de paso en el algoritmo NLMS, el cual solo usa un paso constante, de manera que se pudiera mejorar la velocidad de convergencia y disminuir el error de salida del filtro adaptable. Dado que ya existían en la literatura los algoritmos VSLMS[7.8] y VSS [7.9] , los cuales usan un tamaño de paso variable , sin embargo tales algoritmos no funcionan adecuadamente para relaciones S/R bajas.El algoritmo propuesto VSS-CC [7.10] mejora a los algoritmos anteriores.Cabe mencionar que el número de multiplicaciones requeridas para la obtención de las muestras de salida del filtro adaptable fueron  $2N+1$ . A continuación se describen dos posibles extensiones del algoritmo VSS-CC.

### 7.1.-Extensión del algoritmo de paso variable VSS-CC al algoritmo en bloque FLMS

Dado el esfuerzo realizado para reducir la complejidad computacional de los filtros adaptables basados en el algoritmo LMS, como es el caso del algoritmo de procesamiento en bloque (FLMS) (ver sección 4.10 del capítulo IV). Sus características son:

-Los coeficientes del filtro adaptable son modificados solamente al final de cada bloque, así usando la Transformada Rápida de Fourier se reduce la complejidad computacional de éstos filtros.

-baja velocidad de convergencia.

Una variante que puede investigarse es la posibilidad de usar pasos variables en el algoritmo de procesamiento de bloque, para así incrementar la velocidad de convergencia de los mismos.Básicamente la idea sería substituir el parámetro  $\mu$  del bloque "estimación del paso en LMS", por la  $\alpha$  variable del algoritmo VSS-CC (ver la figura 7.1).

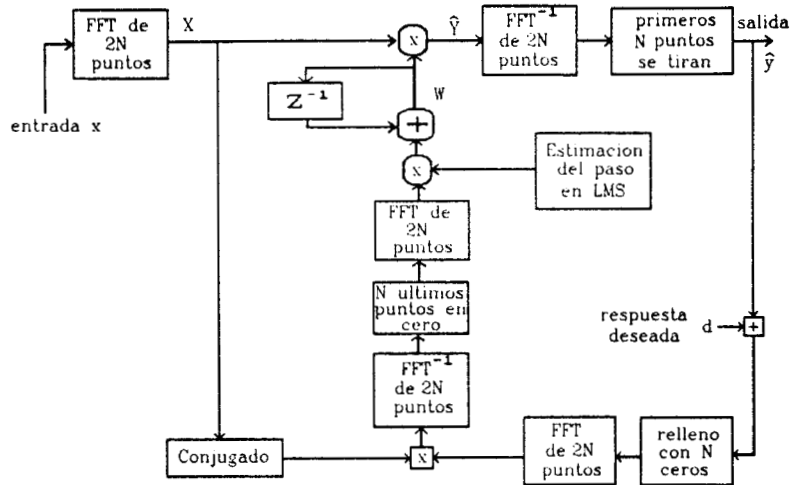


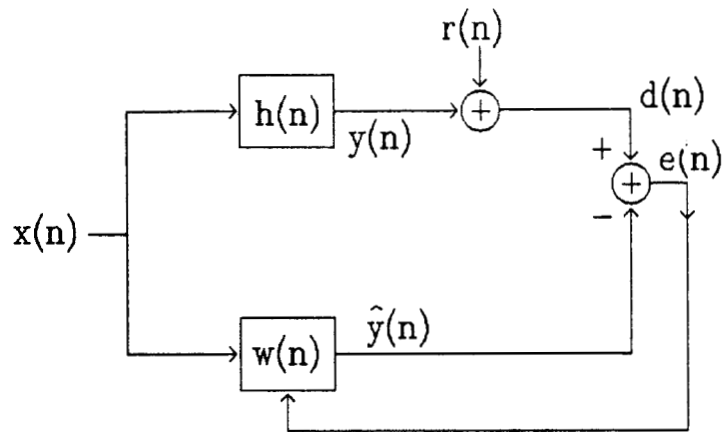
Figura 7.1.-Algoritmo FLMS.

## 7.2.-Otras aplicaciones del algoritmo VSS-CC

El algoritmo VSS-CC es propuesto para su aplicación en cancelación de ecos telefónicos, sin embargo existen otras aplicaciones donde puede aplicarse este algoritmo. A continuación se consideran tres casos mas de aplicación.

### 7.2.1.- Identificación de Sistemas

En muchas aplicaciones un sistema con una estructura desconocida tiene señales de entrada y salida las cuales son observables [7.1-7.3]. Así una forma de obtener cierto conocimiento acerca de la respuesta dinámica del sistema, es aplicar su entrada a un filtro adaptable y usar su salida como señal de referencia ó respuesta deseada del sistema como se muestra en la figura 7.2.



donde :

- $h(n)$ - Respuesta a impulso del sistema desconocido
- $w(n)$ - Respuesta a impulso del filtro adaptable
- $r(n)$ - Ruido aditivo
- $e(n)$ - Señal de error
- $x(n)$ - Señal de entrada

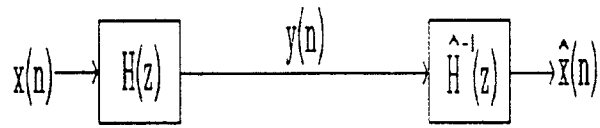
Figura 7.2.-Configuración para identificar un sistema

En la figura 7.2 una vez que el filtro converge, éste desarrolla una respuesta a impulso  $w(n)$ , la cual es la mejor aproximación de la respuesta a impulso del sistema  $h(n)$ , ya que ambos presentan salidas similares cuando tienen ambas la misma señal de entrada.

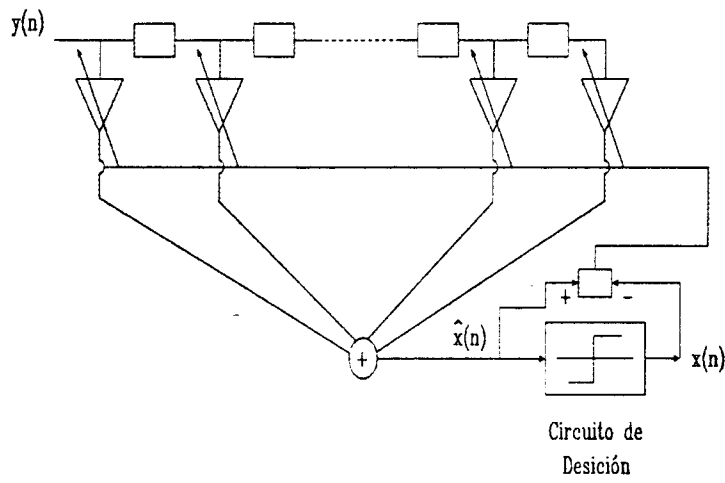
### 7.2.2.-Cancelación de Interferencia Intersímbolo en Transmisión Digital

Este problema ha recibido considerable atención [7.4-7.7], ya que han sido propuestos varios métodos para su solución; uno de los más usados es un filtro adaptable, el cual estima el inverso del canal de comunicación y realiza un filtrado inverso, de esta manera se cancela ó reduce la interferencia intersímbolo (figura 7.3).





(a) Ecuación : Principio Básico



(b) Ecuador Transversal.

Figura 7.3.- Ecuación Adaptiva.

### 7.3.-Conclusiones

Al inicio de este proyecto de investigación, se estudió el algoritmo adaptable LMS [7.2], el cual usa un paso constante, esto refleja un compromiso entre el error de salida del filtro y la velocidad de convergencia, es decir si se selecciona un cierto tamaño de paso grande se tiene una velocidad de convergencia rápida, sin embargo la potencia del error de salida del filtro adaptable es grande. Para mejorar lo anterior varios algoritmos de paso variable han sido propuestos, tal es el caso de los algoritmos VSLMS [7.8] y VSS [7.9]; revisados en las secciones 4.8 y 4.9 respectivamente.

Durante el desarrollo de éste proyecto de investigación, se comenzó a trabajar con el algoritmo VSS-Doble paso; el cual usa por ejemplo dos pasos diferentes, esto es una  $\alpha$  grande y una  $\alpha$  pequeña (por ejemplo  $\alpha = 1$  y  $\alpha = 0.05$ ). En éste algoritmo se usa un umbral  $c_1$  de decisión para seleccionar entre la  $\alpha$  grande y la  $\alpha$  pequeña; éste umbral se le da un valor cercano a la  $\alpha$  pequeña. Este algoritmo es de utilidad en aplicaciones en las que se conoce previamente el nivel de ruido presente en el sistema, o bien cuando el nivel de ruido es constante y puede ser estimado con cierta facilidad.

Para aplicaciones donde el nivel de ruido es desconocido, o no puede ser estimado, se requiere de seleccionar el umbral  $c$  en base al error de salida que se está obteniendo, en el algoritmo VSS-AC se usan dos alfas diferentes (tal como en el algoritmo VSS-Doble paso), y se selecciona entre la  $\alpha$  grande y la  $\alpha$  pequeña con un umbral  $c$  obtenido por la correlación entre la señal deseada y el error de salida del filtro adaptable, de esta manera cuando el umbral  $c$  es igual a la  $\alpha$  pequeña, el algoritmo comienza a trabajar con la  $\alpha$  pequeña.

Por otra parte el algoritmo VSS-CC es útil en aplicaciones donde el nivel de ruido es desconocido y además el nivel de ruido es grande (o variable con el tiempo). Este algoritmo usa un paso variable, el cual se obtiene en función de la correlación entre el error de salida  $e(n)$  y la salida del filtro adaptable  $\hat{y}(n)$ . Al inicio  $e(n)$  y  $\hat{y}(n)$  están correlacionados, así el algoritmo usa una  $\alpha$  grande y se tiene una velocidad de convergencia rápida. Cuando el algoritmo converge, la correlación produce una  $\alpha$  pequeña, así se tiene un aumento en el ERLE. La particularidad de éste algoritmo es precisamente que si la relación S/R presente en el filtro es pequeña, entonces el

algoritmo continúa obteniendo un ERLE aceptable (superando al algoritmo VSS[7.9]).

Dada la particularidad del algoritmo VSS-CC, útil en aplicaciones donde la relación S/R presente en el filtro es pequeña, una variante que puede investigarse (sección 7.1), es el uso de pasos variables para algoritmos de procesamiento en bloque (FLMS), para así incrementar la velocidad de convergencia de los mismos, los coeficientes del filtro adaptable son modificados al final de cada bloque, usando la Transformada Rápida de Fourier se reduce la complejidad computacional de éstos filtros.

Por supuesto el algoritmo VSS-CC puede ser usado en otras aplicaciones, tales como identificación de sistemas e interferencia intersímbolos; éstas son variantes, que muestran un interesante panorama por continuar en esta línea de investigación.

## 7.4.-Referencias

- [7.1] Friedlander, B."System Identification Techniques for Adaptive Signal Processing",Circuits Systems Signal Processing,vol. 1, No 1, pp 3-41, 1982.
- [7.2] Nagumo,J., and Noda,A., "A learning method for system identification IEEE Trans. Automat. Contr., vol. AC-12, no 3, pp 282-287, June 1967.
- [7.3] Sage,A.P.,and Melsa,J.L., System Identification,Academic Press, New York, 1971.
- [7.4] Gersho,A.,and Lim,T.L.,"Adaptive Cancellation of Intersymbol Interference for Data Transmission",Bell System Technical Journal,vol. 60, No 11, pp 1997-2021, November 1981.
- [7.5] Neissen,C.,W.,and Willim,D.K.,"Adaptive Equalizer for Pulse Transmission",IEEE Trans., vol. COM-18,No 4,pp. 377-395,August 1970.
- [7.6] Proakis,J.G.,"Advances in equalization for intersymbol interference", in Advances in Communication Systems, vol. 4, Academic Press, New York, pp. 123-198, 1975.
- [7.7] Reeves,A.H.,"The past, present and future of PCM",IEEE Spectrum,vol. 12. pp 58-63, 1975.
- [7.8] Harris,R.W., Chabries,D.M.,and Bishop,F.A.,"A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316,Apr.1986.
- [7.9] Raymond ,H., Kwong,E., and Johnston,W.,"A Variable Step Size LMS Algorithm ",IEEE Transactions on Signal Processing,vol.40 No.7 pp. 1633-1642,July 1992.

### Publicación realizada

- [7.10] Casco,F.,Pérez,H.,Nakano,M., and López,M.,"A Variable Step Size (VSS-CC) NLMS Algorithm", IEICE Trans.Fundamentals,vol.E78-A, No 8 August, 1995,pp 1004-1009.

## APENDICE I.-Factor de olvido y Memoria usados en este proyecto de investigación

Sea la estimación :

$$\bar{x}(N) = \frac{1}{N} \sum_{k=1}^N x^2(k) \quad (\text{a.1})$$

donde N es el número de muestras a usar en la memoria, para la estimación de

$$\bar{x}(N) \quad (\text{a.2})$$

La estimación anterior puede desarrollarse como :

$$\bar{x}(N) = \frac{1}{N} \sum_{k=1}^N x^2(k)$$

$$\bar{x}(N) = \frac{1}{N} \sum_{k=1}^{N-1} x^2(k) + \frac{1}{N} x^2(N)$$

$$= \frac{1}{N} \frac{N-1}{N-1} \sum_{k=1}^{N-1} x^2(k) + \frac{1}{N} x^2(N)$$

$$= \frac{N-1}{N} \frac{1}{N-1} \sum_{k=1}^{N-1} x^2(k) + \frac{1}{N} x^2(N)$$

$$= \frac{N-1}{N} \bar{x}(N-1) + \frac{1}{N} x^2(N) \quad (\text{a.3})$$

haciendo

$$\varepsilon = \frac{1}{N} \quad (\text{a.4})$$

y

$$\lambda_1 = 1 - \varepsilon \quad (\text{a.5})$$

(a.3) resulta

$$\bar{x}(N) = \lambda_1 \bar{x}(N-1) + \varepsilon x^2(N) \quad (\text{a.6})$$

donde a  $\lambda_1$  se le conoce como factor de olvido y  $\varepsilon$  es el inverso de la memoria.

La idea de usar (a.6), es realizar la estimación de  $\overline{x(n)}$  en forma recursiva.

## BIBLIOGRAFIA

- [1] Bellamy, J., Digital Telephony, editorial John Wiley 1991.
- [2] Bershad,N,and Macchi,O."Comparison of RLS and LMS algoritihms for tracking a chirped signal",Proc.ICASSP,Glasgow, Scotland, pp 896-899.1989.
- [3] Bershad, N.J., and Macchi,O.,"Adaptive Recovery of a Chirped Sinusoid in Noise, Part 2:Performance of the LMS Algorithm," IEEE Trans. Acoust. Speech Signal Process vol. ASSP-39, 1991.
- [4] Bershad, N.J,"Analysis of the normalized LMS algorithm with Gaussian inputs",IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-34, pp 793-806, 1986.
- [5] Brigham,E.O.,The Fast Fourier Transform, Prentice Hall, Englewood Cliffs,NJ,1974.
- [6] Cowan,C.F.N.,Grant,P.M.,Adaptive Filters,Prentice Hall,Englewood Cliffs,N.J.,1985.
- [7] Cowan,C.F.N.,and Mavor,J.,"New Digital Adaptive Filter Implementation Using Distributed-Arithmetic Techniques",Proceedings IEEE, Vol. 128.Pt.F,No 4, pp 225-230. August 1981.
- [8] Carayannis,G.,Manolakis,D., and Kaloptsidis, N.,"Fast Kalman Type Algorithms for Sequential Signal Processing,"Proc 1983 IEEE Internat.Conf Acoust.Speech, Signal Processing,pp 186-189.
- [9] Claasen.T.A.C.M., Melenbrauker.W.F.G.,"Adaptive Techniques for Signal Processing in Communications", IEEE Communications Magazine., November 1985- Vol 23, No 11., pp 8-19.
- [10] Chapman,R.C. Digital Signal Processor, Special Issue,Bell System Technical Journal,vol. 60, No 7, part 2, september 1981.
- [11] Clark,G.A.,Mitra,S.K.,and Parker,S.R.,"Block Implementation of AdaptiveDigital Filters",IEE Trans., Vol.CAS-28,No 6,pp 584-592,June 1981.
- [12] Chazan,D.,Medan,Y,and Shvadron,U.,"Noise cancellation for hearing aids",IEEE Trans. Acoustics Speech Signal Process,vol. ASSP-36.pp 1697-1705, 1988.

- [13] Claasen, T.A.C.M., and Meckenbrauker W.F.G., "Comparison of the Convergence of two algorithms for Adaptive FIR Digital Filters", IEEE Trans., vol. CAS-28, No 6, pp 510-518, June 1981.
- [14] Davila, C.E., "A Stochastic Newton Algorithm with Data-Adaptive Step Size", IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. 38, No 10, pp 1796-1798, October 1990.
- [15] Diniz, P.S.R., "LMS-Newton Adaptive Filtering Algorithm, variable convergence factor". IEEE Trans. on Signal Processing, pp 617-627, Marzo 1995.
- [16] Demytko, N., and Mackechnie, L., "A High Speed Digital Adaptive Echo Canceled", Australian Telecommunication Review, vol. 7, No 1, pp 20-28, 1973.
- [17] Dentino, M., McCool, J., and Widrow, B., "Adaptive Filtering in the Frequency Domain", Proceedings IEEE, vol. 66, No 12, pp 1658-1659, December 1978.
- [18] Eweda, E. "RLS/LMS/ sign algorithms comparison", IEEE Trans. on Signal Processing, pp 2937-2944, Nov. 1994.
- [19] Feintuch, P.L., "An Adaptive Recursive LMS Filter", Proceedings IEEE, vol. 64, No 11, pp 1622-1624, November 1976.
- [20] Ferrara, E.R., "Fast Implementation of LMS Adaptive Filters", IEEE Trans. vol. ASSP-28, No 4, pp 474-475, August 1980.
- [21] Friedlander, B. "System Identification Techniques for Adaptive Signal Processing", Circuits Systems Signal Processing, vol. 1, No 1, pp 3-41, 1982.
- [22] Gersho, A., and Lim, T.L., "Adaptive Cancellation of Intersymbol Interference for Data Transmission", Bell System Technical Journal, vol. 60, No 11, pp 1997-2021, November 1981.
- [23] Goodwin, G.C., and Sin, K.S., Adaptive Filtering Prediction and Control, Prentice Hall, Englewood Cliffs, NJ, 1984.
- [24] Glover, J.R., Jr., "Adaptive noise cancelling applied to sinusoidal interferences", IEEE Trans. Acoust. Speech Signal Process., vol. ASSP-25, pp 484-491, 1977.
- [25] Gritton, C.W.K., D.W. Lin., "Echo Cancellation Algorithms", Proc IEEE ASSP Magazine April 1984, pp 30-37.



- [26] Harris,R.W., Chabries,D.M.,and Bishop,F.A., "A Variable Step (VS) Adaptive Filter Algorithm, IEEE Trans.Acoust.,Speech and Signal Processing,vol. ASSP-34,pp.309-316, Apr.1986.
- [27] Hamming,R.,W.,Digital Filters,2a ed.,Prentice Hall,Englewood Cliffs,NJ,1983.
- [28]Haykin,S., "Adaptive Filters:past,present and future",Proc IMA Conf.math.Signal Process,Warwick,England,1989.
- [29] Haykin,S., Adaptive Filter Theory, Editorial Prentice Hall 1991.
- [30] Herrera ,P. E, Fundamentos de Ingeniería Telefónica, editorial Limusa 1989.
- [31] Hsia,T.C. "Convergence analysis of LMS and NLMS adaptive algorithms", Proc,ICASSP, Boston, pp. 667-670, 1983.
- [32] Honig,M.L.,and Messersmitt,D.G.,Adaptive Filters:Structures,Algorithms and Applications,Boston : Kluwer 1984.
- [33] León-García,A.,Probability and Random Processes for Electrical Engineering, Reading, Mass.,Addison Wesley, 1989.
- [34] Ljung,L., "Analysis of a General Recursive Prediction Algorithm", Automatica , vol. 17, No 1,pp 89-99,January 1981.
- [35] Long,G.,Shwed,D.,and Falconer,David D., "Study of a Pole-Zero Adaptive Echo Canceller",IEEE Trans. on Circuits and Systems, vol CAS-34, No 7, July 1987.
- [36] Lucky,R.W.,Sallz,J.,and Weldon, E.JR.,Principle of Data Communications , McGraw Hill, NewYork, 1968.
- [37] Lawrence,R.E.,and H Kaufman(1971), "The Kalman Filter for the equalization of a digital communication channel",IEEE Trans.Commun. Technol.,vol.COM-19,pp 1137-1141, 1971.
- [38] Makhoul,J., "Linear Prediction:a tutorial review",Proc.IEEE, vol 63, pp 561-580, 1975.
- [39] Mikhael,W., F.Hu.L.Kasovsky, G.Kang, and L. Francen, "Adaptive Filters with individual adaptation of parameters", IEEE Trans. Circuits Syst.,vol.CAS-33,no 7, pp 677-685, July 1986.

- [40] Manolakis,D., "On Adaptive Recursive Digital Filters and their Application to Echo Cancellation in Data Transmision", Nat.Lab.Technical Note Nr. 242/78, 1978(Philips Research).
- [41] Murano,K., ET AL "Echo cancellation and applications", IEEE Commun., vol. 28, pp 49-55, 1990.
- [42] Mulgrew,B., "Kalman Filter Techniques in Adaptive Filtering ", IEEE Proc(London)vol. 134.pt.F, pp 239-243.
- [43] Narayan, S.S., Peterson A.M., and Narashima,M.J., "Transform Domain LMS Algorithm", IEEE Trans. Acoust. Speech Signal Process., vol. ASSP-31, pp 609-615.
- [44] Nagumo,J., and Noda,A., "A learning method for system identification IEEE Trans. Automat. Contr., vol. AC-12, no 3, pp 282-287, June 1967.
- [45] Neissen,C.,W., and Willim,D.K., "Adaptive Equalizer for Pulse Transmission" , IEEE Trans., vol. COM-18, No 4, pp. 377-395, August 1970.
- [46] Ochiai,K., Araseki, T., and Ogihara, T., "Echo Canceller with two Echo Path Models", IEEE Trans.COM-25, No 6, pp 589-595, June 1977.
- [47] Oppenheim Alan V, Applications of Digital Signal Processing, editorial Prentice Hall 1978.
- [48] Oppenheim,A.V., and Schafer,R.W., Digital Signal Processing, Prentice Hall, Englewood Cliffs, NJ, 1975.
- [49] Oppenheim,A.V., Willsky,A.S., and Young,I.T., Signals and Systems, Prentice Hall, 1983.
- [50] Papoulis,A., Probability, Random Variables and Stochastic Processes, Mc Graw Hill, 1985.  
Academic Publishers, 1984.
- [51] Pérez ,H and Amano,F " A new Subband Echo Canceller Structure", Trans. of the IEICE, Vol. E73, pp. 1625-1631, Oct. 1990.
- [52] Perez,H., Amano,F., "Acoustic Echo Cancellation Using Multirate Techniques", IEICE Transacctions, vol. E74, no 11 pp 3559-3568, Nov1991.

- [53] Proakis,J.G., "Advances in equalization for intersymbol interference", in Advances in Communication Systems, vol. 4, Academic Press, New York, pp. 123-198, 1975.
- [54] Proakis,J.G.,and Manolakis,D.G.,Introduction to Digital Signal Processing, Mc Millan 1988.
- [55] Proakis,J.G.,Digital Communications,Mc Graw Hill 1983.
- [56] Rabiner,L.R.,and Schafer,R.W.,Digital Processing of Speech Signals,Prentice Hall 1978.
- [57] Rabiner,L.R.,and Gold,B.,Theory and Application of Digital Signal Processing, Prentice Hall, Englewood Cliffs,NJ, 1975.
- [58] Raymond ,H., Kwong,E., and Johnston,W., "A Variable Step Size LMS Algorithm ",IEEE Transactions on Signal Processing,vol.40 No.7 pp. 1633-1642,July 1992.
- [59] Reeves,A.H., "The past, present and future of PCM",IEEE Spectrum,vol. 12. pp 58-63, 1975.
- [60] Roberts,R.A.,and Mullis,C.T.,Digital Signal Processing,Addison Wesley, 1987.
- [61] Sambur,M.R., "Adaptive noise cancelling for speech signals",IEEE Trans. Acoust. Speech Signal Process, vol. ASSP-26,pp 419-423, 1978.
- [62] Solo, V., "The limiting behavior of LMS", IEEE Trans. Acoust. Speech Signal Process., vol. 37, pp. 1909-1922, 1989.
- [63] Shynk,J.J., "Frequency Domain and Multirate Adaptive Filtering IEEE Signal Processing Magazine, January 1992,pp 14-36.
- [64] Sondhi M, and D.A. Berkley 1980 "Silencing echoes in the telephone network", proc. IEEE, vol 68, pp 948-963.
- [65] Sage,A.P.,and Melsa,J.L., "System Identification",Academic Press, New York, 1971.
- [66] Sondhi, M.M., and Mitra, D., "New Results on the Performance of a well-Known Class of Adaptive Filters,"Procc IEEE, 1976, Vol 64,No 11, pp 1583-1597.

- [67] Samson, C., "A unified treatment of fast Kalman algorithms for identification", *Inte.J.Control*, vol.35, pp 909-934, 1982.
- [68] Sondhi M.M. and A.J.Presti 1966, "A self adaptive echo canceller", *Bell Syst. Tech. J.*, vol 45 pp 1851-1854.
- [69] Sondhi, M.M. 1970 " Closed Loop Adaptive Echo Canceller using Generalized Filter Networks, U.S. Patent 3, 499,999.
- [70] Sondhi M.M. Mar. 1967 " An Adaptive Echo canceller", *Bell System Tech, J*, vol 46 No3 pp 497-511.
- [71] Sharpe,S.M.,and Nolte,L.W., "Adaptive MSE estimation", *Proc. ICASSP*, Atlanta, Ga.,pp 518-521, 1981.
- [72] Treichler,J.R.,Johnson, C.R.,and Larimore,M.G., "Theory and Design of Adaptive Filters", *Wiley-Interscience*, New York 1987.
- [73] Ukrainec,A.,and Haykin,S., "Adaptive Interference Canceller", *Canadian Patent Application 603,935*.1989.
- [74] Voorman,J., "On the convergence of adaptive equalizers", *J.O. Voorman , Nat.Lab.Report Nr.5937*, 1984 (Philips Research).
- [75]Widrow,B.,Glover,J.R.,McCool,J.M.,Kaunitz,J.,Williams,C.S.,Hearn,R.H.,Ziedler,J.R.,Dong,E.,and Goodlin,R.C., "Adaptive Noise Canceling:Principles and Applications", *Proceedings IEEE*, Vol.63,No 12,pp 1692-1716, December 1975.
- [76] Widrow,B.,and McCool,J.M., "A Comparison of Adaptive Algorithms Based on the Methods of Steepest Descent and Random Search", *IEEE trans.*, Vol.AP-24,No 5,pp 615-637, September 1976.
- [77] Widrow,B.,and Stearns,S.D., *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs,NJ, 1984.
- [78] Weckler,G.P.,and Walby,M.D., "Programable Transversal Filters:Design Tradeoffs", *Proceedings 5th International Conference on CCD's CCD-79*, Centre for Industrial Consultancy and Liaison, University of Edinburgh, pp 211-221, 1979.
- [79] Weinstein, E., "Stability Analysis of LMS Adaptive Filters", submitted for publication, 1983.

[80] Weiss, A., and Mitra, D., "Digital Adaptive Filters: Conditions for Convergence, Rates of Convergence, Effects of Noise and Errors Arising from the Implementation," IEEE Trans. Information Theory, Nov 1979, Vol.IT-25, No 6, pp 637-652.

[81] Widrow, B., Mc.Cool, J., Ball, M., "The Complex LMS Algorithm", Proceedings IEEE, vol.63, No 4, pp 719-720, April 1975.

[82] Widrow, B. 1966 Adaptive Filters I: Fundamentals, Rep.SEL-66-126 (TR6764-6) Stanford Electronics Laboratories, Stanford Calif.

[83] Zeidler, J.R. (1990), "Performance Analysis of LMS Adaptive Prediction Filters", scheduled for publication in Fall 1990, Proc. IEEE, vol. 78.

### **Principales Contribuciones a la Teoría del Filtrado Adaptivo**

[84] Casco, F., Pérez, H., López, M., "A three step size LMS algorithm", Proceedings of the 1993 IEICE FALL CONFERENCE, pp 1-84, 1-85 September 5 to 8 1993, Sapporo, Hokkaido Institute of Technology.

[85] Casco, F., Perez, H., "A Two Step Size LMS Algorithm", Proceedings of the International Conference on Signal Processing, vol 1, pp 543-546, October 1993, Beijing China.

[86] Casco, F., Pérez, H., López, M., and Marcelín, R., " A two Step Size NLMS Adaptive Filter Algorithm", IEEE Singapore International Conference on Communication Systems. ICCS'94, 14-18 November 1994, pp 814-819.

[87] Casco, F., Pérez, H., Nakano, M., and López, M., "A Variable Step Size (VSS-CC) NLMS Algorithm", IEICE Trans.Fundamentals, vol.E78-A, No 8 August, 1995, pp 1004-1009.