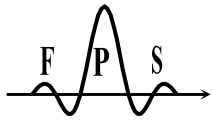


Sumário

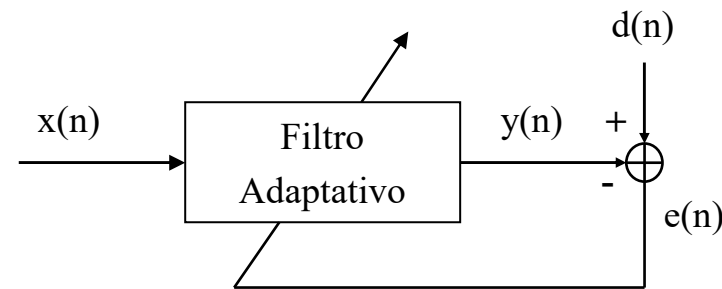
- *Filtragem adaptativa numa perspetiva prática*
 - *conceito e exemplos*
 - *algoritmos de gradiente*
 - *filtragem ótima de Wiener*
 - *o método do gradiente mais negativo*
 - *o algoritmo LMS*



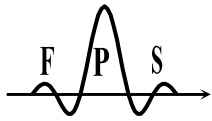
Filtros adaptativos

- Conceito

- um filtro adaptativo é um filtro cujos coeficientes são ajustados, de forma adaptativa, em função de objetivos ou condições variáveis no tempo e traduzidos num sinal de erro



- a aplicação típica consiste em suprimir ou modelizar certas componentes (possivelmente indesejáveis), representadas por $x(n)$ e projetadas num sinal $d(n)$, de acordo com algum critério estatístico incidindo sobre o sinal de erro $e(n)$
- o critério típico e prático para a adaptação dos coeficientes do filtro e otimização do seu desempenho é a minimização do valor médio quadrático do sinal de erro

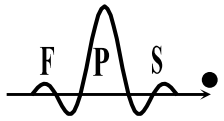


- Conceito (*cont.*)

- para entradas estacionárias e usando o critério de minimização de erro médio quadrático, o filtro ótimo é único e designa-se por filtro de *Wiener*; para entradas não estacionárias, os filtros de *Kalman* são mais adequados e eficientes (mas também implicam maior complexidade)
- os filtros adaptativos podem ser do tipo FIR ou IIR, porém são quase sempre preferidas estruturas FIR já que, como se sabe, estas são intrinsecamente estáveis

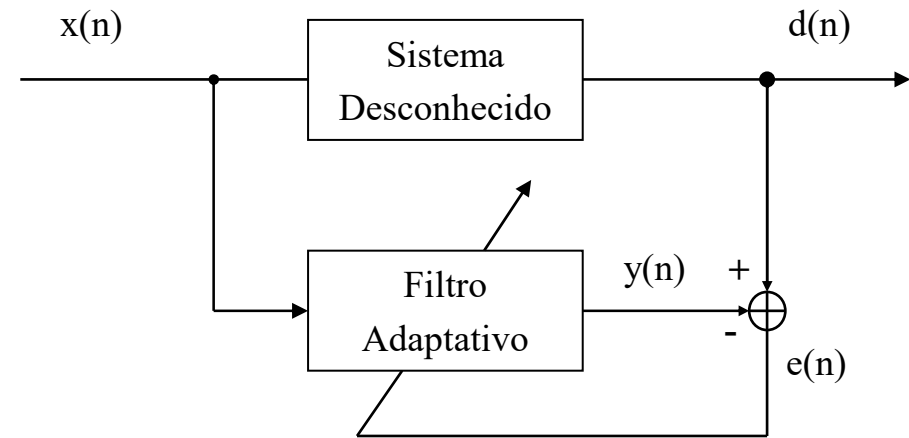
- Exemplos

- os filtros adaptativos são usados em diversas aplicações e contextos, tudo dependendo da forma como a resposta desejada do sistema adaptativo é caracterizada e extraída. Há essencialmente quatro tipos de aplicações: identificação, modelização inversa, predição, cancelamento de interferência.



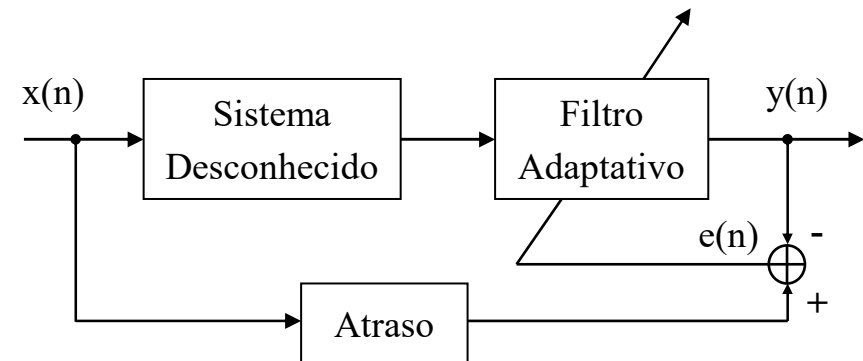
Identificação

- nestas aplicações, o filtro adaptativo é usado para fornecer um modelo linear que representa a melhor aproximação, de acordo com algum critério, a um sistema desconhecido



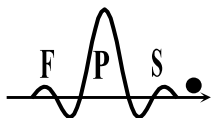
• Modelização inversa (igualização)

- nestas aplicações o filtro adaptativo tem por objetivo fornecer um modelo inverso que representa a melhor aproximação, de acordo com algum critério, a um sistema desconhecido



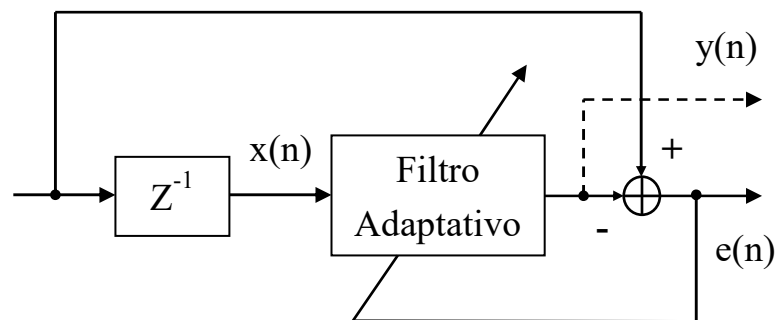
Exemplos de aplicações:

- descorrelação preditiva
- igualização adaptativa



Predição

- nestas aplicações, o filtro adaptativo tem por objetivo fornecer a melhor predição para a evolução de um sinal com base no conhecimento do seu passado

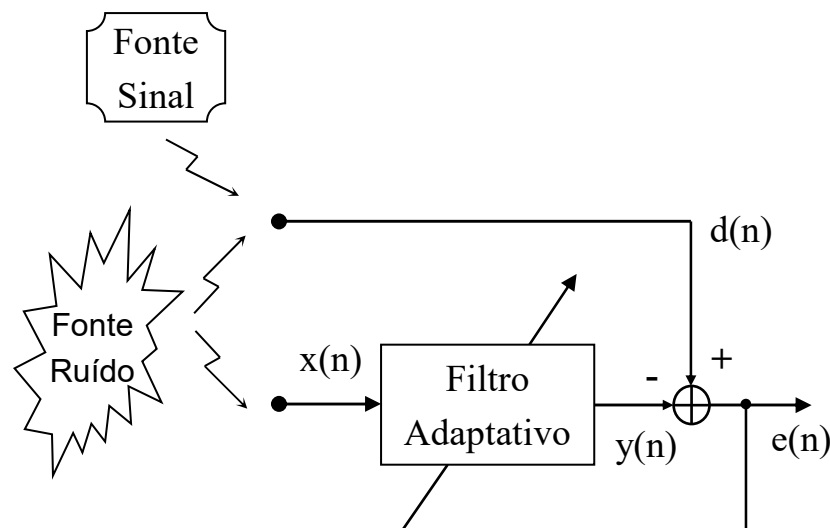


Exemplos de aplicações:

- codificação linear preditiva (LPC)
- ADPCM
- análise espectral autoregressiva (AR)

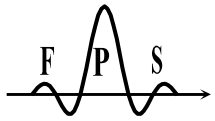
Cancelamento de interferência

- nestas aplicações o filtro adaptativo tem por objetivo cancelar uma interferência desconhecida e contida em $d(n)$ mas refletida também no sinal de referência $x(n)$. O objetivo é simplesmente subtrair o “ruído” do sinal principal, de forma adaptativa, de modo a melhorar, à saída, a relação sinal-ruído.



Exemplos de aplicações:

- cancelamento adaptativo de ruído
- cancelamento de eco



Dado que o primeiro e último casos servem de base ao trabalho de laboratório, retiram-se a seguir algumas conclusões pertinentes a partir da análise aos sinais e operações implicados. Considerando que o sinal $d(n)=s(n)+x'(n)$ contém uma parcela de informação útil a preservar $s(n)$ e também uma componente de ruído a eliminar $x'(n)$ e que este é também projetado em $x(n)$, tomam-se, como hipóteses realistas, as seguintes:

$$E[s(n)x'(n-k)] = 0, \quad \forall k$$

$$E[s(n)x(n-k)] = 0, \quad \forall k$$

$$E[x(n)x'(n-k)] = p(+k)$$

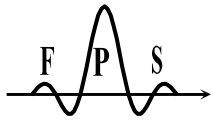
ou seja, $s(n)$ é descorrelacionado com qualquer uma das duas representações do ruído e estas, por sua vez, exibem uma correlação cruzada dada por $p(k)$ para uma distância k .

A saída do filtro adaptativo $y(n)$ e o sinal de erro $e(n)$ são dados por:

$$y(n) = \sum_{k=0}^{N-1} h_n(k)x(n-k)$$

$$e(n) = d(n) - y(n) = s(n) + x'(n) - y(n)$$

o que coloca em evidência que o sinal $s(n)$ faz parte do sinal de erro. Porém, dadas as hipóteses anteriores, minimizar o erro médio quadrático de $e(n)$ equivale a minimizar o erro médio quadrático do ruído indesejado dado por $e(n)-s(n)=x'(n)-y(n)$, o que significa que $s(n)$ não é afetado pelo processo de filtragem adaptativa.

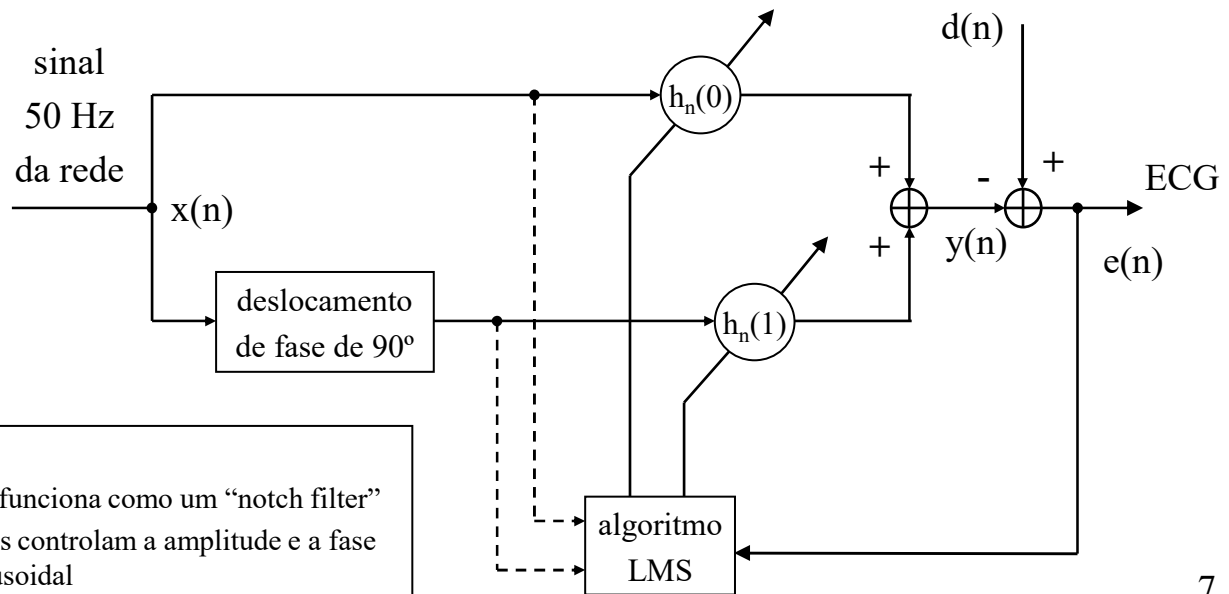


Como casos limite de operação do filtro adaptativo temos duas situações, a primeira em que $y(n)=x'(n)$ e a segunda em que $E[d(n)x(n-k)]=0, \quad \forall k$.

No primeiro caso a operação de filtragem é efetiva, cancelando todo o ruído. No segundo caso, o sinal de referência é completamente descorrelacionado com o sinal primário, fazendo com que o filtro adaptativo se “auto-desligue”, produzindo $y(n)=0$, o que não afeta o sinal principal.

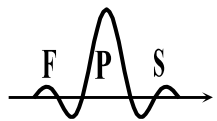
– EXEMPLO 1

- Cancelamento da interferência sinusoidal de rede (50 Hz) num eletrocardiograma usando somente dois coeficientes



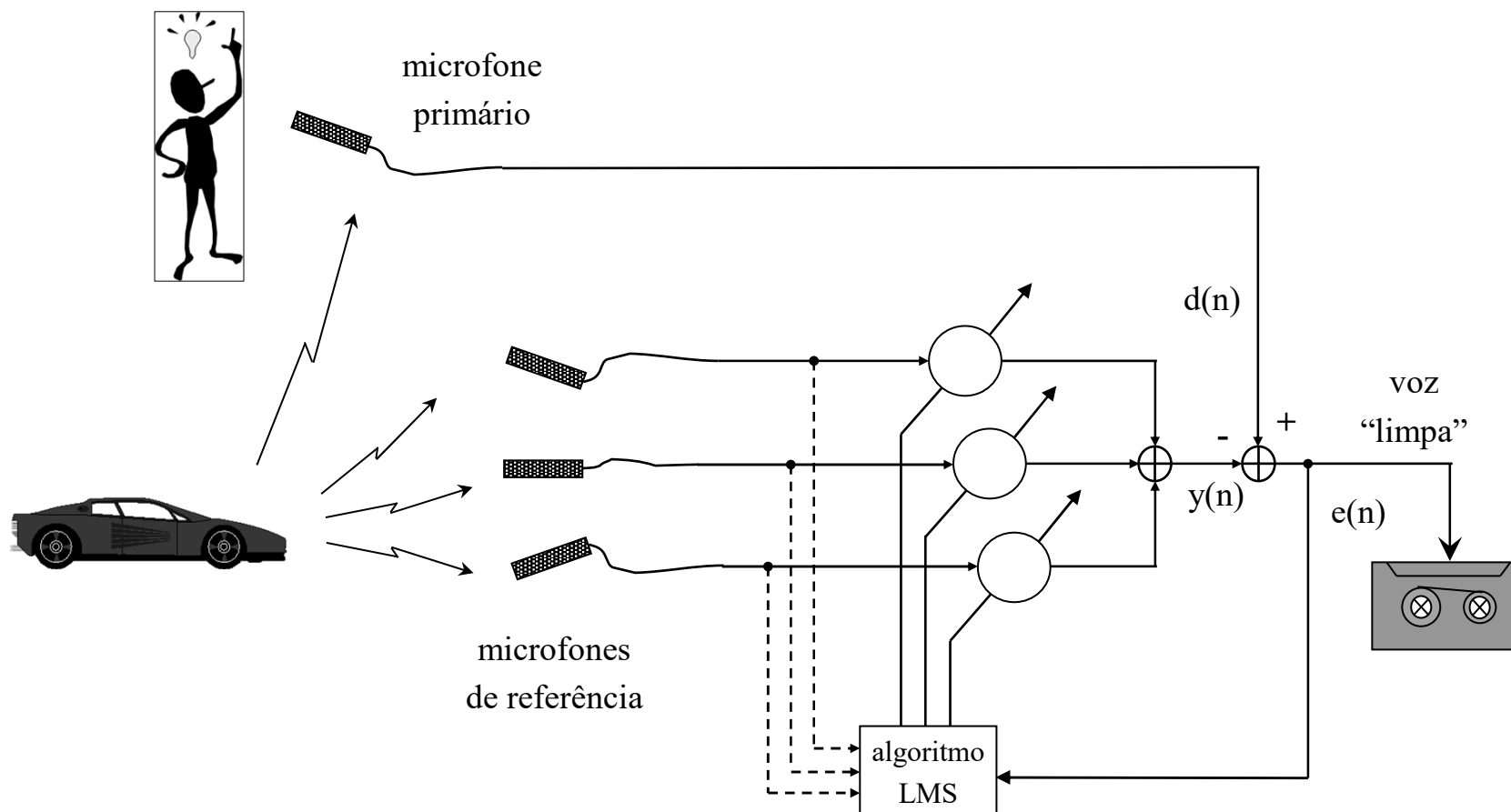
NOTAS:

- o filtro adaptativo funciona como um “notch filter”
- os dois coeficientes controlam a amplitude e a fase da interferência sinusoidal



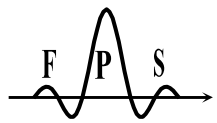
– EXEMPLO 2

- Redução / cancelamento de ruído acústico em sinais de voz (é possível melhorar a relação sinal/ruído em mais do que 10 dB)



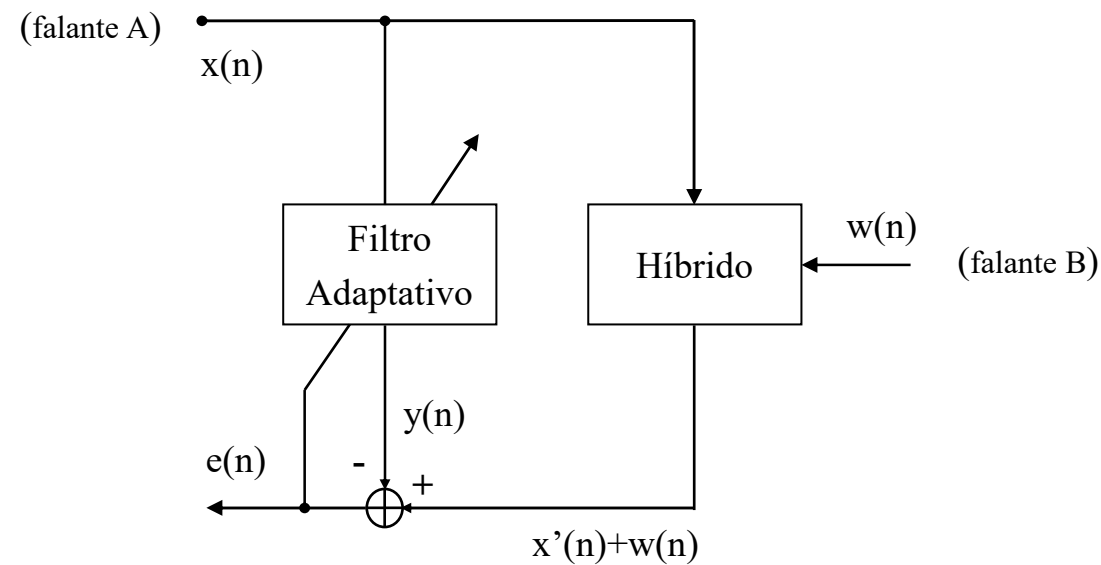
NOTAS:

- o algoritmo de adaptação pode ser o "LMS" ou outro alternativo
- os microfones de referência devem situar-se suficientemente longe do falante de modo a captarem quase exclusivamente só o ruído



– EXEMPLO 3

- Cancelamento de eco em circuitos telefónicos

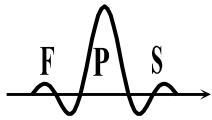


NOTAS:

- o objectivo do filtro adaptativo é sintetizar uma réplica do eco, perto do seu ponto de geração, e subtraí-la do sinal de retorno
- idealmente, o filtro adaptativo aproxima a função de transferência de retorno do eco

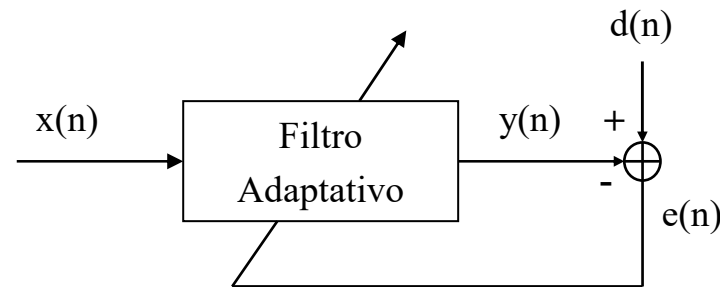
Nota introdutória sobre o algoritmo LMS, o que é afinal ?

trata-se de um algoritmo de gradiente estocástico, no sentido em que itera o peso de cada coeficiente de um filtro transversal (FIR) na direção do gradiente negativo da função amplitude quadrada de um sinal de erro, em ordem ao peso desse coeficiente; detalhes seguem nos 'slides' seguintes ...



Algoritmos de Gradiente

- Filtragem de *Wiener*
 - dado o sistema adaptativo já ilustrado anteriormente:



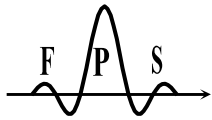
será:
$$e(n) = d(n) - y(n) = d(n) - \sum_{k=0}^{N-1} h_n(k)x(n-k)$$

admitindo as hipóteses anteriormente consideradas e admitindo vetores complexos, será para a função de erro a minimizar:

$$J = E\{e(n)e^*(n)\} = E\left\{\left[d(n) - \sum_{k=0}^{N-1} h_n(k)x(n-k)\right]\left[d^*(n) - \sum_{\ell=0}^{N-1} h_n^*(\ell)x^*(n-\ell)\right]\right\}$$

$$J = E\left\{\underbrace{d(n)^2}_{\sigma_d^2}\right\} - \sum_{\ell=0}^{N-1} \underbrace{h_n^*(\ell)}_{p(+\ell)} E\{d(n)x^*(n-\ell)\} - \sum_{k=0}^{N-1} \underbrace{h_n(k)}_{p^*(+k)} E\{d^*(n)x(n-k)\} + \sum_{k=0}^{N-1} \sum_{\ell=0}^{N-1} \underbrace{h_n(k)h_n^*(\ell)}_{r(\ell-k)} E\{x(n-k)x^*(n-\ell)\}$$

sendo σ^2 a variância de $d(n)$ cuja média se supõe nula, $p(\ell)$ e $p^*(k)$ representam a correlação cruzada entre o sinal $d(n)$ e a entrada $x(n)$ e $r(\ell-k)$ representa a correlação da entrada para uma distância $\ell-k$.



Abreviadamente será:

$$J = \sigma_d^2 - \sum_{\ell=0}^{N-1} h_n^*(\ell) p(+\ell) - \sum_{k=0}^{N-1} h_n(k) p^*(+k) + \sum_{k=0}^{N-1} \sum_{\ell=0}^{N-1} h_n(k) h_n^*(\ell) r(\ell - k)$$

o que revela que quando os sinais $x(n)$ e $d(n)$ são conjuntamente estacionários, a função do erro quadrático médio é uma função de 2ª ordem dos coeficientes do filtro FIR. Mostra-se que [Simon Haykin, “Adaptive Filter Theory”, Prentice-Hall, 1991] que se este filtro possuir N coeficientes, a superfície de erro é uma superfície hiperparabolóide de $N+1$ dimensões com um mínimo global e sem mínimos locais.

O mínimo global ocorre em J_{MIN} onde se verifica que o vetor gradiente é identicamente nulo, ou seja:

$$\nabla_k(J) = 0, \quad k = 0, 1, \dots, N-1$$

Assumindo coeficientes $h(k)$ reais, o vetor gradiente, concretizado na expressão J produz:

$$\nabla_k(J) = \frac{\partial J}{\partial h(k)}$$

$$\nabla_k(J) = -2p(+k) + \sum_{\ell=0}^{N-1} h_n(\ell) r(\ell - k) + \sum_{s=0}^{N-1} h_n(s) r(k - s) = -2p(+k) + 2 \sum_{\ell=0}^{N-1} h_n(\ell) r(\ell - k)$$

pelo que se conclui que os coeficientes ótimos são os que verificam:

$$\sum_{\ell=0}^{N-1} h_n(\ell) r(\ell - k) = p(+k)$$

a solução deste sistema de equações é conhecida como a solução de Wiener-Hopf que fornece o filtro ótimo (filtro de Wiener).

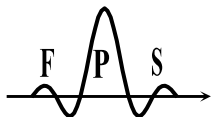


Ilustração de superfície hiperparabolóide (suave e convexa) quando o número de coeficientes é 2

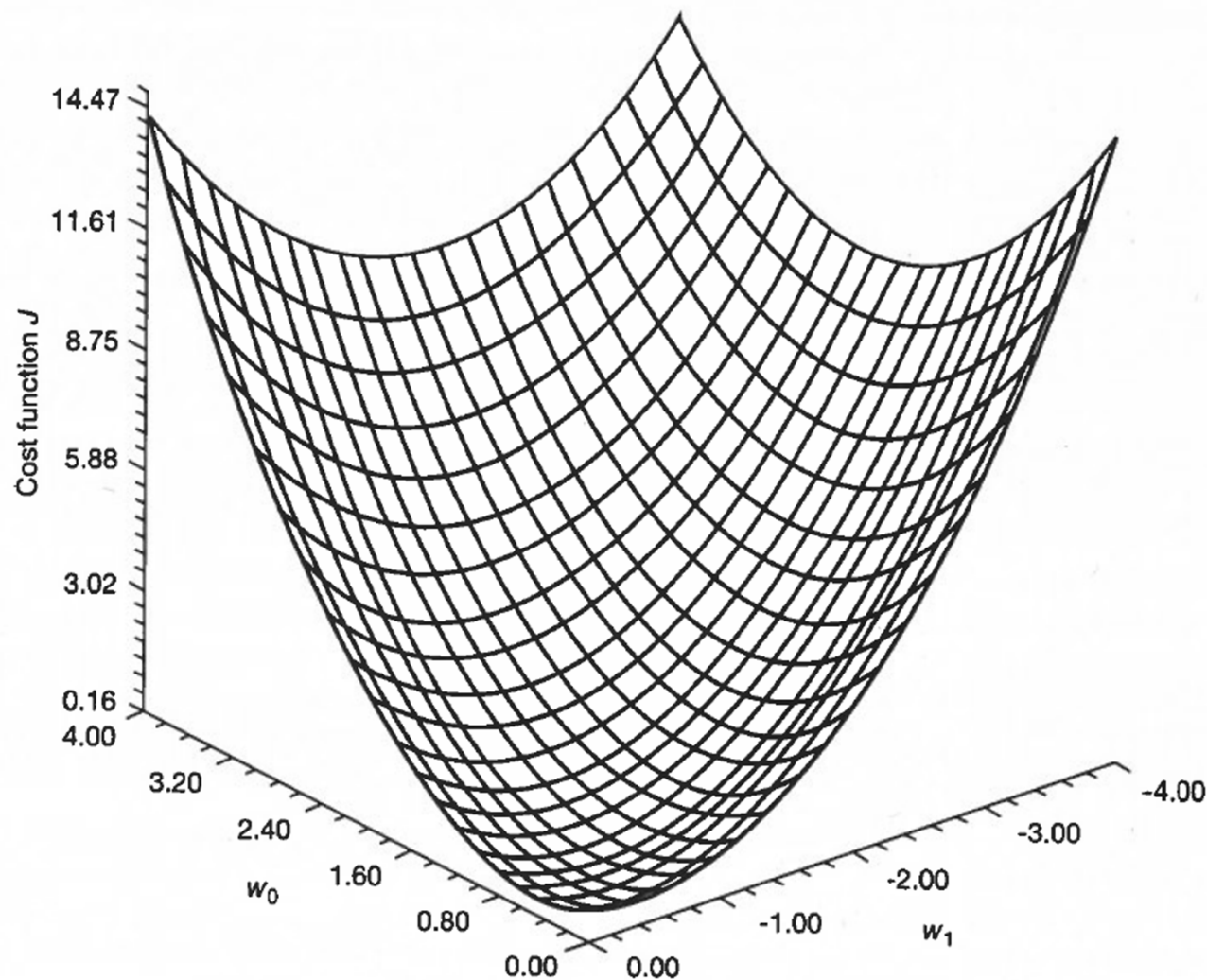
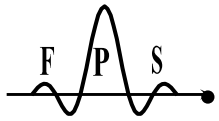
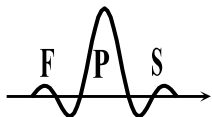


Figure 5.6 Error-performance surface of the two-tap transversal filter described in the numerical example.

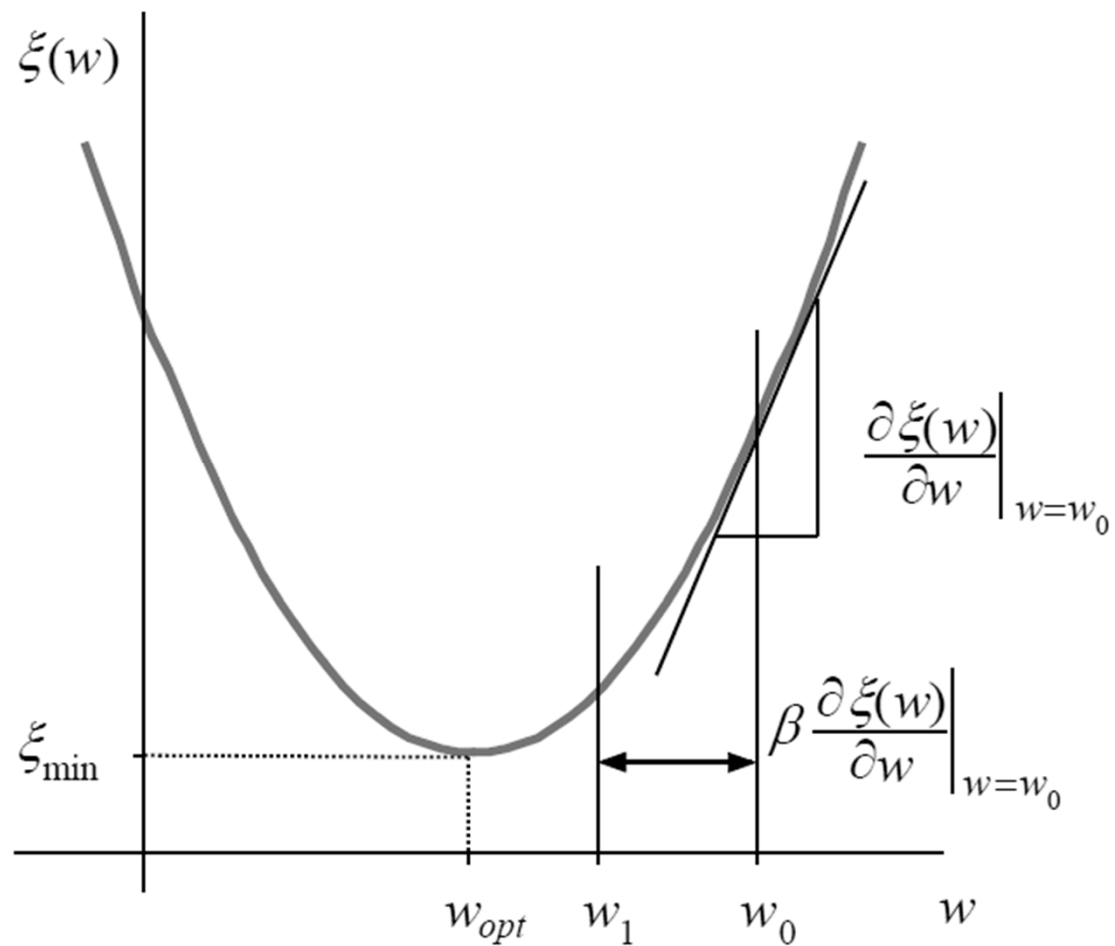


O método do gradiente

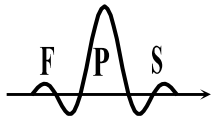
- Em vez de usarmos o procedimento determinístico anterior que é aplicável a sinais estacionários para os quais se conhecem previamente as estatísticas associadas, podemos, não conhecendo estas, optar por um método iterativo de aproximação ao filtro ótimo de Wiener.
- O método iterativo de adaptação mais conhecido que converge para a solução ótima (filtro de Wiener), é o método de gradiente mais negativo (“steppest gradient” = *direction of the negative of the gradient vector*) que referiremos simplesmente pelo método do gradiente e que consiste no seguinte procedimento:
 1. Parte-se de um valor inicial para os coeficientes do filtro, tipicamente o vetor nulo,
 2. Calcula-se o vetor gradiente, determinado em ordem aos coeficientes do filtro no instante n ,
 3. Estimam-se os coeficientes do filtro para a próxima iteração modificando a estimativa atual no sentido oposto ao do vetor gradiente,
 4. Retorna-se ao ponto 2 e repete-se o processo.



- Ilustração do método do gradiente:



$$\underline{w}_{k+1} = \underline{w}_k - \beta \hat{\underline{\nabla}}_k$$



- De acordo com este procedimento, será então:
$$h_{n+1}(k) = h_n(k) + \frac{1}{2} \mu [-\nabla_k [J(n)]]$$
 em que μ é uma constante real e positiva que traduz o *passo de adaptação* do algoritmo.

Como vimos anteriormente,
$$\nabla_k [J(n)] = -2p(+k) + 2 \sum_{\ell=0}^{N-1} h_n(\ell) r(\ell - k)$$
 porém, como não conhecemos as estatísticas, podemos optar por uma expressão alternativa usando a definição de $p(k)$ e de $r(\ell - k)$ e admitindo dados e coeficientes reais:

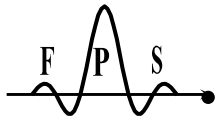
$$\nabla_k [J(n)] = -2E\{d(n)x(n-k)\} + 2 \sum_{\ell=0}^{N-1} h_n(\ell) E\{x(n-k)x(n-\ell)\}$$

$$\nabla_k [J(n)] = -2E\left\{x(n-k) \left[d(n) - \sum_{\ell=0}^{N-1} h_n(\ell) x(n-\ell) \right]\right\} = -2E\{x(n-k)e(n)\}$$

Este resultado traduz o princípio da ortogonalidade: para a solução ótima, $x(n-k)$ e $e(n)$ são ortogonais. Como corolário desta afirmação, resulta que para a solução ótima, $y(n)$ e $e(n)$ são ortogonais.

Será então:
$$h_{n+1}(k) = h_n(k) + \mu E\{x(n-k)e(n)\}$$

Esta é a solução do método iterativo de gradiente mais negativo que evolui suavemente para a solução ótima (*i.e.* para o filtro de Wiener).



O algoritmo LMS

- O algoritmo anterior presume a obtenção de estatísticas durante o procedimento iterativo e oferece a garantia de convergência suave para o filtro de Wiener. Podemos contudo ter, a partir do algoritmo anterior, uma outra aproximação à solução ótima baseada no gradiente “instantâneo” (dito *estocástico*), em que se elimina o operador de média:

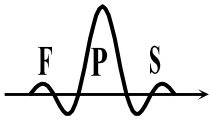
$$h_{n+1}(k) = h_n(k) + \mu x(n-k)e(n)$$

Dada a sua simplicidade, esta solução é muito popular e constitui o conhecido algoritmo de filtragem adaptativa LMS (de *Least Mean Squares*). Porém, o custo da simplicidade traduz-se em ruído de gradiente no cálculo recursivo de cada coeficiente do filtro adaptativo. Este ruído de gradiente pode conduzir (assumindo convergência):

1. Ou a uma solução aleatória do filtro adaptativo em torno do filtro ótimo de Wiener,
2. Ou a um deslocamento fixo do filtro adaptativo relativamente ao filtro ótimo (desalinhamento), comprometendo portanto o seu desempenho.

Estes aspetos são controlados através do passo de adaptação μ : um passo mais pequeno dá lugar a um menor ruído de gradiente o que minimiza o problema de desalinhamento, mas implica também, em contra-partida, uma convergência mais lenta para a solução ótima.

Para saber mais: há outros algoritmos de filtragem adaptativa que possuem melhor desempenho do que o LMS, e.g., o NMLS e o *Recursive Least Squares* (RLS).



- Ilustração do impacto do passo de adaptação μ

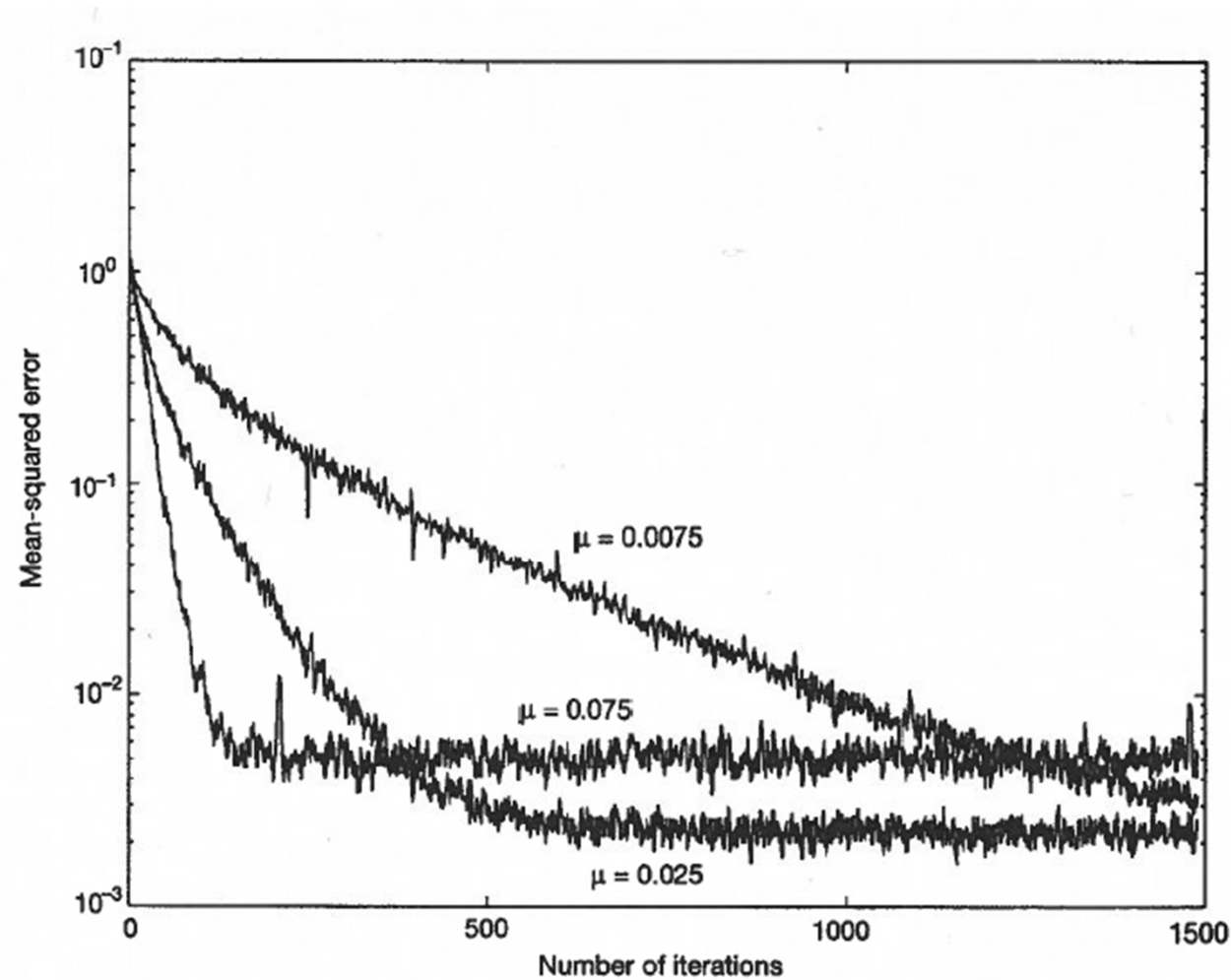


Figure 9.23 Learning curves of the LMS algorithm for adaptive equalizer with the number of taps $M = 11$, fixed eigenvalue spread, and varying step-size parameter μ .