

## 6. Filtrage adaptatif

### 6.1. Introduction

Les filtres tels que nous les avons vus jusqu'à présent sont en général utilisés dans des applications où on connaît la bande de fréquence utile ainsi que la fréquence principale. Ces filtres servent à améliorer le rapport signal sur bruit sous l'hypothèse où la bande de fréquence du bruit est supérieure à celle du signal. Dans ce cas, un filtre passe-bande centré sur la fréquence principale du signal permettra d'extraire le signal.

Les filtres de Wiener développés à partir de concepts temporels et non fréquentiels sont conçus pour minimiser l'erreur quadratique moyenne entre leur sortie et une sortie désirée. Ils sont dits optimum au sens du critère de l'erreur quadratique moyenne et nous verrons que dans ce cas les coefficients du filtres sont liés à la fonction d'autocorrélation du signal d'entrée et à l'intercorrélacion entre les signaux d'entrée et de sortie désirée.

Quand les fonctions d'auto et d'intercorrélacion ne sont pas connues (cas le plus courant), alors on va approcher le filtre optimal de Wiener en utilisant une boucle de retour et un algorithme de minimisation : c'est ce que l'on appelle le filtrage adaptatif. Dans ce cas, on remplacera la connaissance des fonctions de corrélacion par une phase d'apprentissage permettant de modifier itérativement la réponse impulsionnelle du filtre.

### 6.2. Filtres de Wiener

#### 6.2.1 Problème d'estimation linéaire

La figure 5.1 illustre un problème courant d'estimation linéaire.  $x(n)$  correspond au signal qui nous intéresse mais n'est pas directement accessible. Seul  $y(n)$  l'est et  $y(n)$  est obtenu après passage de  $x(n)$  dans un système linéaire suivi de l'addition d'un bruit.

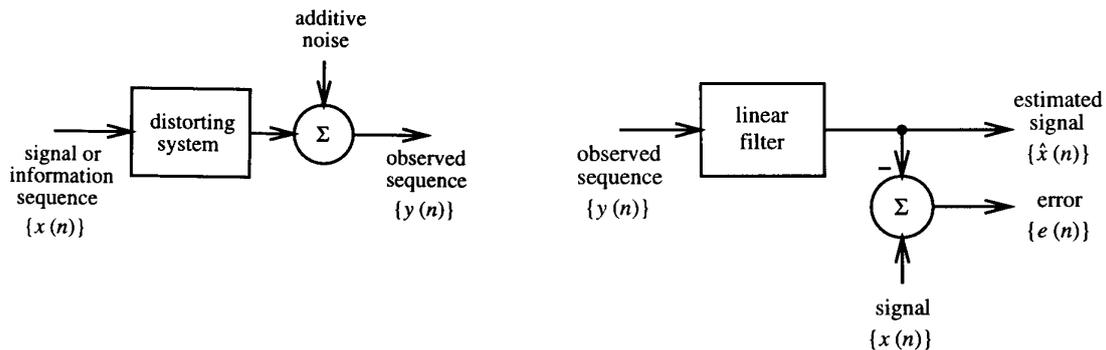


Figure 5.1 : Schéma général d'un problème d'estimation linéaire.

Le problème qui se pose est comment retrouver  $x(n)$  à partir de  $y(n)$ . Une solution consiste à filtrer  $y(n)$  de tel sorte que la sortie  $\hat{x}(n)$  soit la plus proche possible de  $x(n)$ . On peut mesurer la qualité de l'estimation par  $e(n)$  défini par :

$$e(n) = x(n) - \hat{x}(n) \quad (\text{Eq. 5.1})$$

Evidemment, plus  $e(n)$  sera faible, plus l'estimation sera bonne. On cherche donc un filtre qui minimisera l'erreur. Il est pratique de chercher à minimiser  $e^2(n)$  car c'est une fonction quadratique facilement dérivable. Par ailleurs, étant donné que les signaux intéressants sont

aléatoires, la fonction coût qui sera à minimiser est l'erreur quadratique moyenne (MSE<sup>1</sup>) définie par :

$$\xi(n) = E(e^2(n)) \quad (\text{Eq. 5.2})$$

Le filtre optimal de Wiener correspond au filtre qui minimisera la MSE.

### 6.2.2 Filtre de Wiener de type FIR

On se limitera ici au calcul des filtres FIR. Selon les mêmes principes, on peut calculer des filtres IIR. C'est ce qui sera vu dans la suite du cours avec les modèles ARMA utilisés en codage de parole.

Appelons  $\mathbf{h}$ , le filtre que nous recherchons et  $N$  la longueur de sa réponse impulsionnelle donnée avec une notation matricielle par :

$$\mathbf{h} = [h_0 \quad h_1 \quad \dots \quad h_{N-1}]^T$$

Le signal estimé  $\hat{x}(n)$  peut alors s'écrire

$$\hat{x}(n) = \sum_{i=0}^{N-1} h_i y(n-i)$$

ou encore en introduisant la notation matricielle pour  $\mathbf{y}(n)$

$$\hat{x}(n) = \mathbf{h}^T \mathbf{y}(n) \Leftrightarrow \hat{x}(n) = \mathbf{y}^T(n) \mathbf{h} \quad (\text{Eq. 5.3})$$

avec

$$\mathbf{y}(n) = [y(n) \quad y(n-1) \quad \dots \quad y(n-(N-1))]^T$$

En faisant l'hypothèse que les signaux  $x(n)$  et  $y(n)$  sont stationnaires, et si on introduit l'équation 5.3 dans l'équation 5.2, on arrive à la fonction coût suivante :

$$\begin{aligned} \xi &= E\left[ (x(n) - \mathbf{h}^T \mathbf{y}(n))^2 \right] \\ \Leftrightarrow \xi &= E\left[ x^2(n) - 2\mathbf{h}^T \mathbf{y}(n)x(n) + \mathbf{h}^T \mathbf{y}(n)\mathbf{y}^T(n)\mathbf{h} \right] \\ \xi &= E\left[ x^2(n) \right] - 2\mathbf{h}^T \Phi_{yx} + \mathbf{h}^T \Phi_{yy} \mathbf{h} \end{aligned} \quad (\text{Eq. 5.4})$$

où  $\Phi_{yy}$  est une matrice d'autocorrélation de taille  $N \times N$  définie par :

$$\Phi_{yy} = E\left[ \mathbf{y}(n)\mathbf{y}^T(n) \right] \quad (\text{Eq. 5.5})$$

et où  $\Phi_{yx}$  est un vecteur d'intercorrélations de taille  $N$  défini par :

$$\Phi_{yx} = E\left[ \mathbf{y}(n)x(n) \right] \quad (\text{Eq. 5.6})$$

L'équation 5.4 montre que pour un filtre FIR, la fonction coût MSE dépend de la réponse impulsionnelle  $\mathbf{h}$ . Pour en obtenir le minimum, il suffit de chercher les conditions d'annulation de la dérivée de la fonction coût par rapport aux variables qui sont les  $N$  points de la réponse impulsionnelle du filtre.

La dérivée de la fonction coût par rapport au  $j^{\text{ème}}$  point de la réponse impulsionnelle est donnée par :

<sup>1</sup> On utilise la notation anglaise MSE pour Mean Square Error

$$\frac{\partial \xi}{\partial h_j} = E \left[ \frac{\partial}{\partial h_j} \{e^2(n)\} \right] = E \left[ 2e(n) \frac{\partial e(n)}{\partial h_j} \right]$$

En substituant dans cette équation  $e(n)$  par les équations 5.1 et 5.3, on obtient l'expression suivante :

$$\frac{\partial \xi}{\partial h_j} = E \left[ 2e(n) \frac{\partial}{\partial h_j} \{x(n) - \mathbf{h}^T \mathbf{y}(n)\} \right]$$

En utilisant le fait que la sortie du filtre  $\mathbf{h}^T \mathbf{y}(n)$  peut s'écrire comme une somme de  $N$  produits dont un seul contient le terme  $h_j$ , on arrive à l'expression suivante :

$$\begin{aligned} \frac{\partial \xi}{\partial h_j} &= E \left[ 2e(n) \frac{\partial}{\partial h_j} \{h_j y(n-j)\} \right] \\ \Leftrightarrow \frac{\partial \xi}{\partial h_j} &= E[-2e(n)y(n-j)] \end{aligned}$$

On cherche les conditions d'annulation de cette équation pour tous les  $j=\{0, \dots, N-1\}$ . Ceci nous donne un ensemble de  $N$  équations qui peut être écrit de façon matricielle en introduisant le vecteur gradient  $\nabla$  :

$$\nabla = \begin{bmatrix} \partial \xi / \partial h_0 \\ \partial \xi / \partial h_1 \\ \vdots \\ \partial \xi / \partial h_j \\ \vdots \\ \partial \xi / \partial h_{N-1} \end{bmatrix} = -2E \begin{bmatrix} y(n)e(n) \\ y(n-1)e(n) \\ \vdots \\ y(n-j)e(n) \\ \vdots \\ y(n-N+1)e(n) \end{bmatrix} = -2E \left\{ \begin{bmatrix} y(n) \\ y(n-1) \\ \vdots \\ y(n-j) \\ \vdots \\ y(n-N+1) \end{bmatrix} \cdot e(n) \right\} = -2E[y(n)e(n)]$$

En utilisant les équations 5.1 et 5.3 pour remplacer  $e(n)$  on obtient :

$$\nabla = -2E[y(n)(x(n) - \mathbf{y}^T(n)\mathbf{h})] = -2E[y(n)x(n)] + 2E[y(n)\mathbf{y}^T(n)]\mathbf{h}$$

qui devient en introduisant la matrice d'autocorrélation et le vecteur d'intercorrélations :

$$\nabla = -2\Phi_{yx} + 2\Phi_{yy} \mathbf{h} \quad (\text{Eq. 5.7})$$

La réponse impulsionnelle optimale  $\mathbf{h}_{opt}$  est celle qui annule cette équation d'où :

$$\boxed{\Phi_{yy} \mathbf{h}_{opt} = \Phi_{yx}} \quad (\text{Eq. 5.8})$$

Le filtre ainsi défini est appelé filtre FIR de Wiener. Il permet d'obtenir une erreur quadratique minimale entre  $x(n)$  et son estimé  $\hat{x}(n)$  donnée par :

$$\xi_{min} = E[x^2(n)] - \mathbf{h}_{opt}^T \Phi_{yx} \quad (\text{Eq. 5.9})$$

### 6.2.3 Application à l'égalisation de canal

Un problème classique rencontré en traitement du signal pour les télécoms est illustré sur la figure 5.2. Une séquence aléatoire de densité de probabilité uniforme est appliquée à l'entrée d'un canal. Un bruit blanc  $\eta(n)$  s'ajoute à la sortie du canal pour donner le signal observable  $y(n)$ .

Le canal peut être modélisé par sa fonction de transfert en  $z$ ,  $C(z) = \sum_i c_i z^{-i}$ .

Notre objectif est de construire un filtre avec une fonction de transfert  $H(z)$  tel que sa sortie nous donne une bonne estimation de  $x(n)$ . Il est naturellement acceptable d'obtenir notre estimé avec un certain retard  $d$  de telle sorte que ce que l'on estime correspond à  $x(n-d)$ . Ce problème est connu sous le nom d'égalisation de canal dans le domaine des télécommunications ou encore sous le nom de déconvolution en traitement d'images. Les filtres de Wiener nous apporte une solution à ce problème que nous allons préciser.

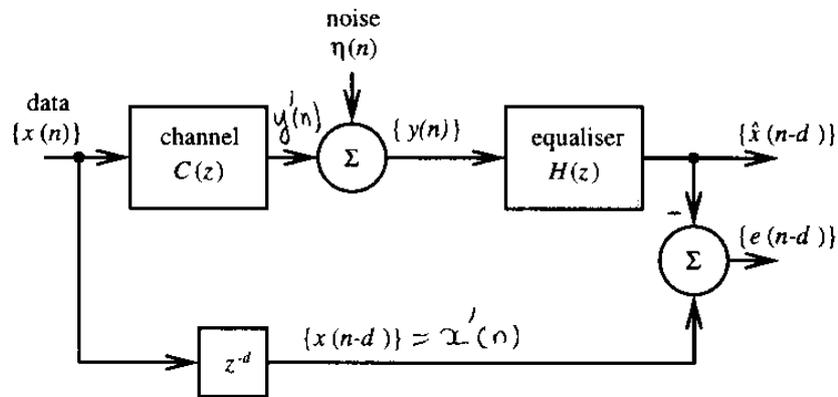


Figure 5.2 : Schéma général d'un problème d'égalisation de canal.

Pour simplifier, nous introduirons les trois notations suivantes  $x'(n)$ ,  $e'(n)$  et  $y'(n)$  respectivement définies par :

$$\begin{aligned} x'(n) &= x(n-d) \\ e'(n) &= x(n-d) - \hat{x}(n-d) \\ y(n) &= y'(n) + \eta(n) \end{aligned} \quad (\text{Eq. 5.10})$$

Le bruit additif et le signal sont considérés comme décorrélés entre eux. Cette hypothèse est généralement vérifiée en pratique. Le filtre de Wiener qui minimise la MSE est alors défini par :

$$\Phi_{yy} \mathbf{h}_{opt} = \Phi_{yx'}$$

où

$$\Phi_{yy} = E[y(n)y^T(n)]$$

et

$$\Phi_{yx'} = E[y(n)x'(n)]$$

Etant donné que les processus sont considérés comme stationnaires et ergodiques, la matrice d'autocorrélation  $\Phi_{yy}$  peut être déduite de la fonction d'autocorrélation  $\phi_{yy}(m)$  donnée par :

$$\begin{aligned} \phi_{yy}(m) &= E[y(n)y(n+m)] \\ \Leftrightarrow \phi_{yy}(m) &= E[\{y'(n) + \eta(n)\}\{y'(n+m) + \eta(n+m)\}] \quad (\text{Eq. 5.11}) \\ \phi_{yy}(m) &= \phi_{y'y'}(m) + \phi_{\eta\eta}(m) + \phi_{y'\eta}(m) \end{aligned}$$

Comme  $y'(n)$  est une combinaison linéaire des échantillon de l'entrée  $x(n)$  i.e.

$$y'(n) = \sum_{i=0}^{\infty} c_i x(n-i)$$

et que  $x(n)$  et  $\eta(n)$  sont décorrélés, il en découle que  $y'(n)$  et  $\eta(n)$  sont décorrélés d'où :

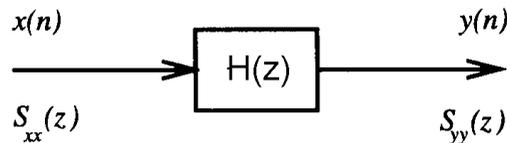
$$\phi_{\eta y'}(m) = E[\eta(m)]E[y'(n+m)] = 0 \text{ pour des processus à moyenne nulle.}$$

Par ailleurs, puisque  $\eta(n)$  est un bruit blanc, il a la propriété suivante :

$$\phi_{\eta\eta}(m) = \sigma_{\eta}^2 \delta(m)$$

Et finalement, l'équation 5.11 prend la forme suivante :

$$\phi_{yy}(m) = \phi_{y'y'}(m) + \sigma_{\eta}^2 \delta(m) \quad (\text{Eq. 5.12})$$



Les transformées en Z des fonctions d'autocorrélation de deux signaux liés par un système linéaire comme dans la figure ci-dessus sont reliées de la façon suivante :

$$S_{yy}(z) = H(z)H(z^{-1})S_{xx}(z)$$

En utilisant cette propriété pour le cas qui nous intéresse, on obtient la relation suivante :

$$S_{y'y'}(z) = C(z)C(z^{-1})\sigma_x^2$$

Par transformée inverse, on obtient :

$$\phi_{y'y'}(m) = Z^{-1}[S_{y'y'}(z)] \quad (\text{Eq. 5.13})$$

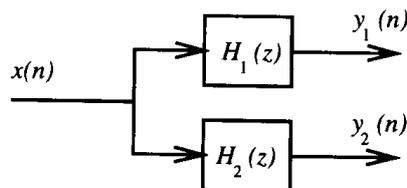
Cette équation introduite dans 5.12 permet d'accéder à  $\phi_{yy}(m)$ .

Afin d'accéder au filtre de Wiener, il reste à calculer le vecteur d'intercorrrelation  $\Phi_{yx'}$ . Si le filtre de Wiener possède N coefficients, le vecteur  $\Phi_{yx'}$  aura N éléments de la forme  $E[y(n-m)x'(n)]$ , où  $0 \leq m < N$ . On peut noter que le processus étant stationnaire :

$$E[y(n-m)x'(n)] = E[y(n)x'(n+m)] = \phi_{yx'}(m)$$

Par ailleurs, comme  $x'(n)$  et  $\eta(n)$  sont décorrélés, on a :

$$\phi_{yx'}(m) = E[y(n)x'(n+m)] = E[(y'(n) + \eta(n))x'(n+m)] = \phi_{y'x'}(m)$$



Les transformées en Z des fonctions d'auto et d'intercorrrelation de trois signaux liés par deux système linéaire en parallèle comme sur la figure ci-dessus sont reliées de la façon suivante :

$$TZ(\phi_{y_2 y_1}(n)) = H_1(z)H_2(z^{-1})TZ(\phi_{xx}(n))$$

$$\Leftrightarrow S_{y_2 y_1}(z) = H_1(z)H_2(z^{-1})S_{xx}(z)$$

En utilisant cette propriété et l'analogie avec la Figure 5.2, on obtient donc la relation suivante :

$$S_{y'x'}(z) = z^{-d} C(z^{-1}) \cdot \sigma_x^2$$

d'où

$$\phi_{y'x'}(m) = Z^{-1} [S_{y'x'}(z)] = \phi_{yx'}(m) \quad (\text{Eq. 5.14})$$

A partir des relations (5.14), (5.13) et (5.12) on peut calculer la réponse impulsionnelle du filtre de Wiener comme nous allons le faire dans l'exemple ci-après.

### 6.2.4 Exercice d'application

Soit un canal de communication modélisable par la fonction de transfert suivante :

$$C(z) = 0.5 + z^{-1}$$

A son entrée, on a un signal aléatoire avec une densité de probabilité uniforme, une moyenne nulle et une variance de 1. A la sortie du canal s'ajoute un bruit blanc de moyenne nulle, de variance 0.1 et décorrélé du signal d'entrée.

Donnez le filtre de Wiener de longueur 3 qui peut être mis en place avec un retard de 1 pour égaliser ce canal. Quelle est la valeur minimale de la MSE qui sera obtenue. Concluez quant à l'intérêt de ce filtre.

(...)

### Réponses

$$h^T = [0.69 \quad 0.13 \quad -0.05] \quad E[e^2] = 0.24$$

Le minimum de MSE est supérieur à la variance du bruit, le filtre n'est pas utile ! Pour avoir un filtre de Wiener efficace, il faut en augmenter le retard toléré et la longueur du filtre de Wiener.

## 6.3. Algorithmes pour le filtrage adaptatif

### 6.3.1. Introduction

La mise en œuvre d'un filtre (estimateur) optimal de Wiener demande la connaissance des caractéristiques du signal, du bruit et de la fonction de transfert du canal. Cela implique également que ces caractéristiques soient stables au cours du temps, ce qui n'est pas le cas en pratique.

Le filtrage adaptatif a pour objet d'approcher ces filtres optimaux. Pour cela, les coefficients de la réponse impulsionnelle du filtre sont adaptés en fonction de l'erreur par une boucle de retour comme le montre la figure ci-dessous.

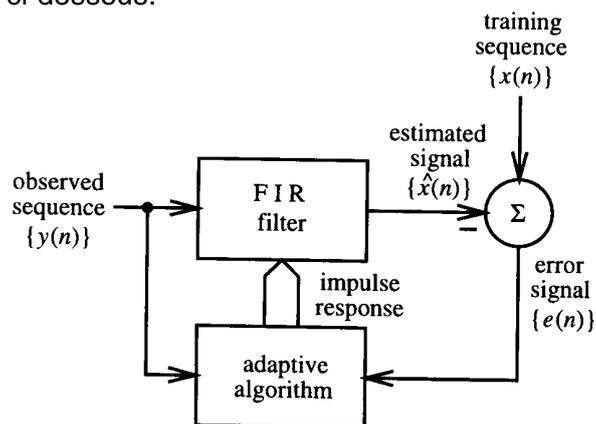


Figure 5.2 : Schéma général d'un système de filtrage adaptatif.

Cette adaptation nécessite une séquence d'apprentissage et une stratégie de mise à jour des coefficients du filtre dont l'objectif est la minimisation d'une erreur. Pour cela, on utilisera des algorithmes d'optimisation. Le détail de ces algorithmes dépasse le cadre du traitement du signal, mais on donnera ici les grandes lignes de deux approches largement utilisées en filtrage adaptatif : le RLS et le LMS.

La réponse impulsionnelle d'un filtre adaptatif est donc variable dans le temps. Elle dépend du signal reçu, de la séquence d'apprentissage et de l'algorithme d'optimisation utilisé. Ces filtres peuvent être de type IIR ou FIR.

Le signal estimé  $\hat{x}(n)$  s'écrit de la façon suivante :

$$\hat{x}(n) = \sum_{i=0}^{N-1} h_i(n-1)y(n-i) \quad (\text{Eq. 5.15})$$

$$\Leftrightarrow \hat{x}(n) = \mathbf{h}^T(n-1)\mathbf{y}$$

Ce qui signifie qu'un point à l'instant  $n$  est calculé en utilisant la réponse impulsionnelle du filtre calculée le coup précédent par l'algorithme d'optimisation.

### 6.3.2 Algorithme RLS

Sachant que les propriétés statistiques nous sont inconnues, on ne va pas chercher à minimiser  $E[e^2(n)]$  mais une somme finie d'erreur au carré donnée par :

$$\xi = \sum_{k=0}^n (x(k) - \hat{x}(k))^2 \quad (\text{Eq. 5.16})$$

Quand cette fonction coût est minimisée en utilisant une réponse impulsionnelle  $h(n)$  associée à  $\hat{x}(n)$ , on obtient l'estimée des moindres carrés.

La réponse impulsionnelle est donc fonction des échantillons disponibles et non pas d'une moyenne statistique générale. Par analogie avec Wiener, elle est donnée par la relation :

$$\mathbf{R}_{yy}(n)\mathbf{h}(n) = \mathbf{r}_{yx}(n) \quad (\text{Eq. 5.17})$$

où

$$\mathbf{R}_{yy}(n) = \sum_{k=0}^n \mathbf{y}(k)\mathbf{y}^T(k) \quad (\text{Eq. 5.18})$$

et

$$\mathbf{r}_{yx}(n) = \sum_{k=0}^n \mathbf{y}(k)x(k) \quad (\text{Eq. 5.19})$$

La réponse impulsionnelle du filtre est donc à modifier à chaque nouvel échantillon. Pour limiter le nombre des calculs, on passe par une équation récursive :

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{R}_{yy}^{-1}(n)\mathbf{y}(n)e(n) \quad (\text{Eq. 5.20})$$

où

$$e(n) = x(n) - \mathbf{h}^T(n-1)\mathbf{y}(n) \quad (\text{Eq. 5.21})$$

et où

$$\mathbf{R}_{yy}^{-1}(n) = \mathbf{R}_{yy}^{-1}(n-1) - \frac{\mathbf{R}_{yy}^{-1}(n-1)\mathbf{y}(n)\mathbf{y}^T(n)\mathbf{R}_{yy}^{-1}(n-1)}{\left(1 + \mathbf{y}^T(n)\mathbf{R}_{yy}^{-1}(n-1)\mathbf{y}(n)\right)} \quad (\text{Eq. 5.22})$$

Ces trois équations sont connues sous le nom de l'algorithme RLS. Le détail de cet algorithme est donné ci-dessous.

Initialise by setting:

$$\mathbf{R}_{yy}(0) = \frac{1}{\delta} I_N, \quad \delta \text{ is a small positive number}$$

$$\mathbf{h}(0) = \mathbf{0}$$

For each time sample  $n = 1, 2, \dots$  do the following:

$$\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$$

$$e(n) = x(n) - \hat{x}(n)$$

$$\mathbf{R}_{yy}^{-1}(n) = \frac{1}{\alpha} \left( \mathbf{R}_{yy}^{-1}(n-1) - \frac{\mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1)}{\left( \alpha + \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \right)} \right)$$

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{R}_{yy}^{-1}(n) \mathbf{y}(n) e(n)$$

*Détail de l'algorithme RLS*

### 6.3.3. Algorithme LMS

D'autres techniques d'optimisation basées sur la descente en gradient peuvent être mise en œuvre. L'algorithme LMS (Least Mean Squares) dont le détail est donné ci-dessous est souvent utilisé dans les systèmes de filtrage adaptatif. On trouvera des détails concernant cette approche dans [1].

Initialise by setting:

$$\mathbf{h}(0) = \mathbf{0}$$

For each time sample  $n = 1, 2, \dots$  do the following:

$$\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$$

$$e(n) = x(n) - \hat{x}(n)$$

$$\mathbf{h}(n) = \mathbf{h}(n-1) + 2 \mu \mathbf{y}(n) e(n)$$

*Détail de l'algorithme LMS*

### 6.3.4. Comparaison

Le problème qui se pose est celui du choix d'un algorithme d'optimisation. Ce choix va être guidé par le nombre d'opérations nécessaires à chaque étape pour mettre à jour les coefficients et par la vitesse de convergence de l'algorithme, c'est à dire la longueur de la séquence d'apprentissage nécessaire pour obtenir un filtre adapté.

Si une méthode répondait à ces deux critères simultanément, elle serait systématiquement utilisée.

L'algorithme LMS nécessite moins de calcul à chaque étape mais converge plus lentement que le RLS comme le montre la figure ci-dessous. C'est donc l'application qui va déterminer le choix de l'algorithme en fonction de la puissance de calcul disponible !

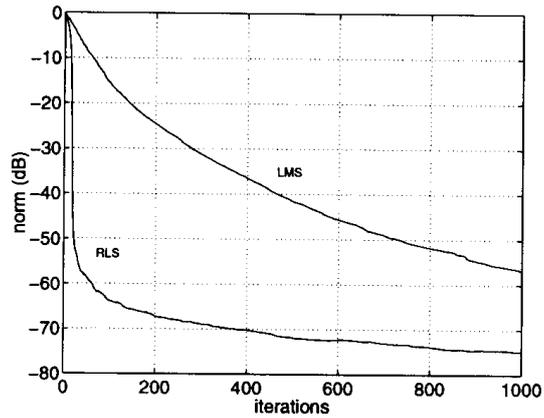


Figure 5. : Erreur (dB) vs Nombre d'itérations pour la comparaison des vitesses de convergence des algorithmes RLS et LMS

## 6.4. Applications

### 6.4.1. Introduction

### 6.4.2. Suppression de bruit

### 6.4.3. Annulation d'écho

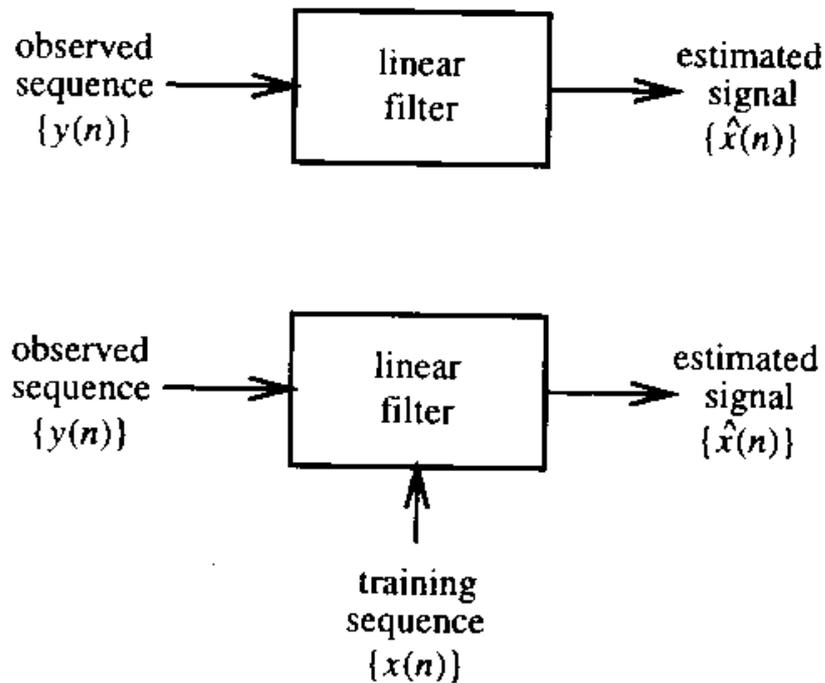
### 6.4.4. Egalisation de canal

## 6.5. Conclusion

## 6.6. Bibliographie

[1] B. Mulgrew, P. Grant, J. Thompson, "Digital Signal Processing, Concepts and Applications", MacMillan Press, 1999,356p.

## Filtre de Wiener



- Objectif

$$\text{Minimiser } \varepsilon = E(e^2(n))$$

- Synthèse de filtre temporelle

$$\Phi_{yy} \mathbf{h}_{\text{opt}} = \Phi_{yx}$$

où

$$\Phi_{yy} = E(Y(n) \cdot Y(n)^T), \text{ Matrice d'autocorrélation (N} \times \text{N)}$$

$$\Phi_{yx} = E(Y(n) \cdot x(n)), \text{ Vecteur d'intercorrélation (N)}$$

## Application du filtre de Wiener à l'égalisation de canal

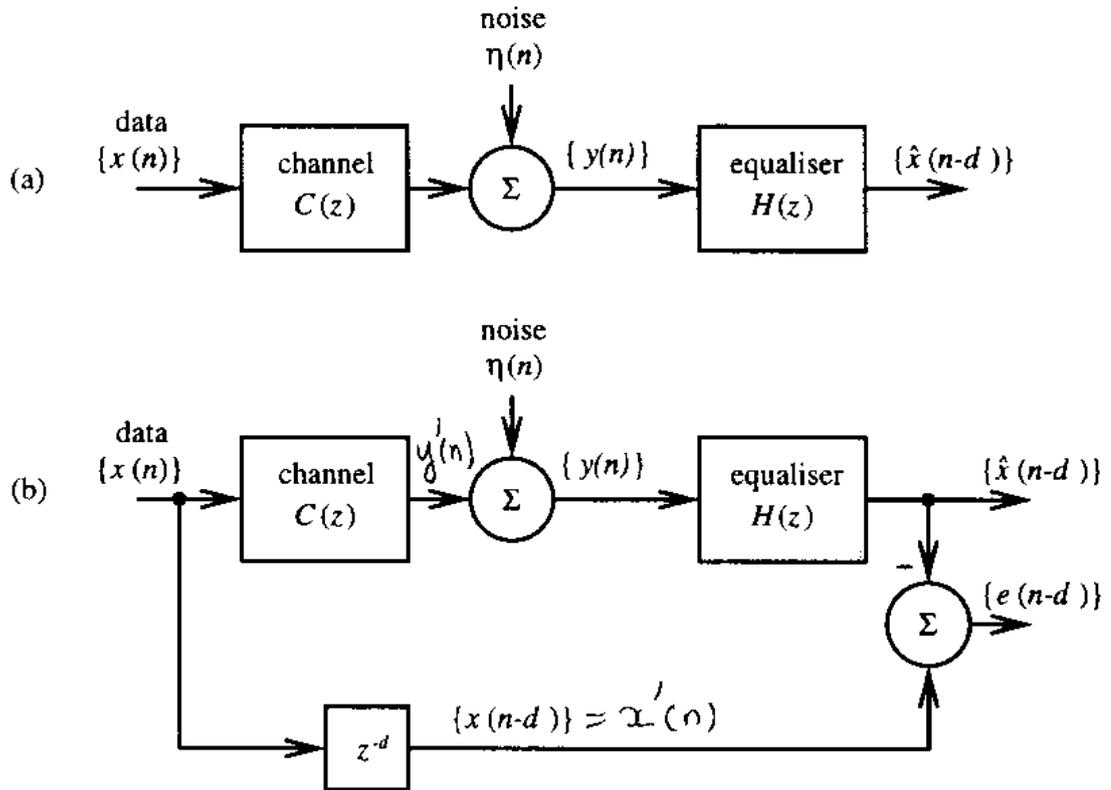


Figure 8.6 The Wiener equaliser: (a) an estimation problem; (b) alternative formulation.

$$\Phi_{yy} \mathbf{h}_{\text{opt}} = \Phi_{yx'}$$

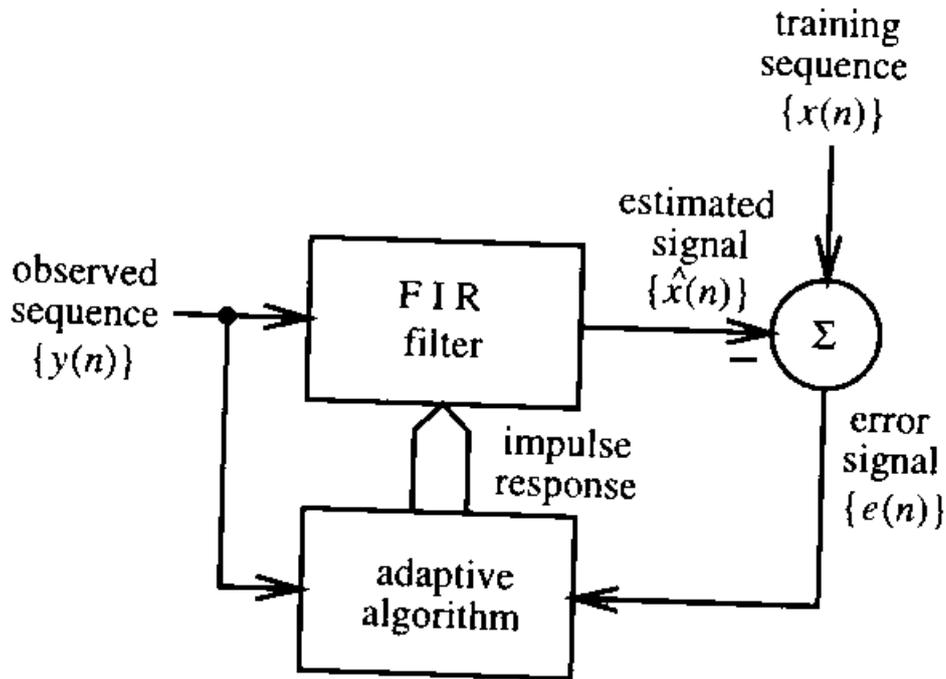
$$\phi_{yy}(m) = E(y(n).y(n+m))$$

...

- $\phi_{yy}(m) = \phi_{y'y'}(m) + \sigma_{\eta}^2 \delta(m)$
- $\phi_{y'y'}(m) = Z^{-1}(S_{y'y'}(z))$  avec  $S_{y'y'}(z) = C(z)C(z^{-1})\sigma_x^2$
- $\phi_{yx'}(m) = \phi_{y'x'} = Z^{-1}(S_{y'x'}(z))$  avec  $S_{y'x'}(z) = z^{-d}.C(z^{-1})\sigma_x^2$

## Filtrage adaptatif

└ Approcher le filtre optimal (MSE) par une boucle de retour



Algorithmes d'optimisation

$$\varepsilon = \sum_{k=0}^n (x(k) - \hat{x}(k))^2$$

- RLS : Recursive Least Squares
- LMS : Least Mean Square (Gradient stochastique, Descente en gradient)

RLS converge plus rapidement que LMS mais plus coûteux en nombre d'opérations

## Algorithmes d'optimisation récursifs

**Table 8.1** *RLS algorithm summary.*

<p>Initialise by setting:</p> $\mathbf{R}_{yy}(0) = \frac{1}{\delta} I_N, \quad \delta \text{ is a small positive number}$ $\mathbf{h}(0) = \mathbf{0}$ <p>For each time sample <math>n = 1, 2, \dots</math> do the following:</p> $\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$ $e(n) = x(n) - \hat{x}(n)$ $\mathbf{R}_{yy}^{-1}(n) = \frac{1}{\alpha} \left( \mathbf{R}_{yy}^{-1}(n-1) - \frac{\mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1)}{\left( \alpha + \mathbf{y}^T(n) \mathbf{R}_{yy}^{-1}(n-1) \mathbf{y}(n) \right)} \right)$ $\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{R}_{yy}^{-1}(n) \mathbf{y}(n) e(n)$
---

**Table 8.2** *LMS algorithm.*

<p>Initialise by setting:</p> $\mathbf{h}(0) = \mathbf{0}$ <p>For each time sample <math>n = 1, 2, \dots</math> do the following:</p> $\hat{x}(n) = \mathbf{h}^T(n-1) \mathbf{y}(n)$ $e(n) = x(n) - \hat{x}(n)$ $\mathbf{h}(n) = \mathbf{h}(n-1) + 2 \mu \mathbf{y}(n) e(n)$
--

## Applications

- 3 modes de fonctionnement

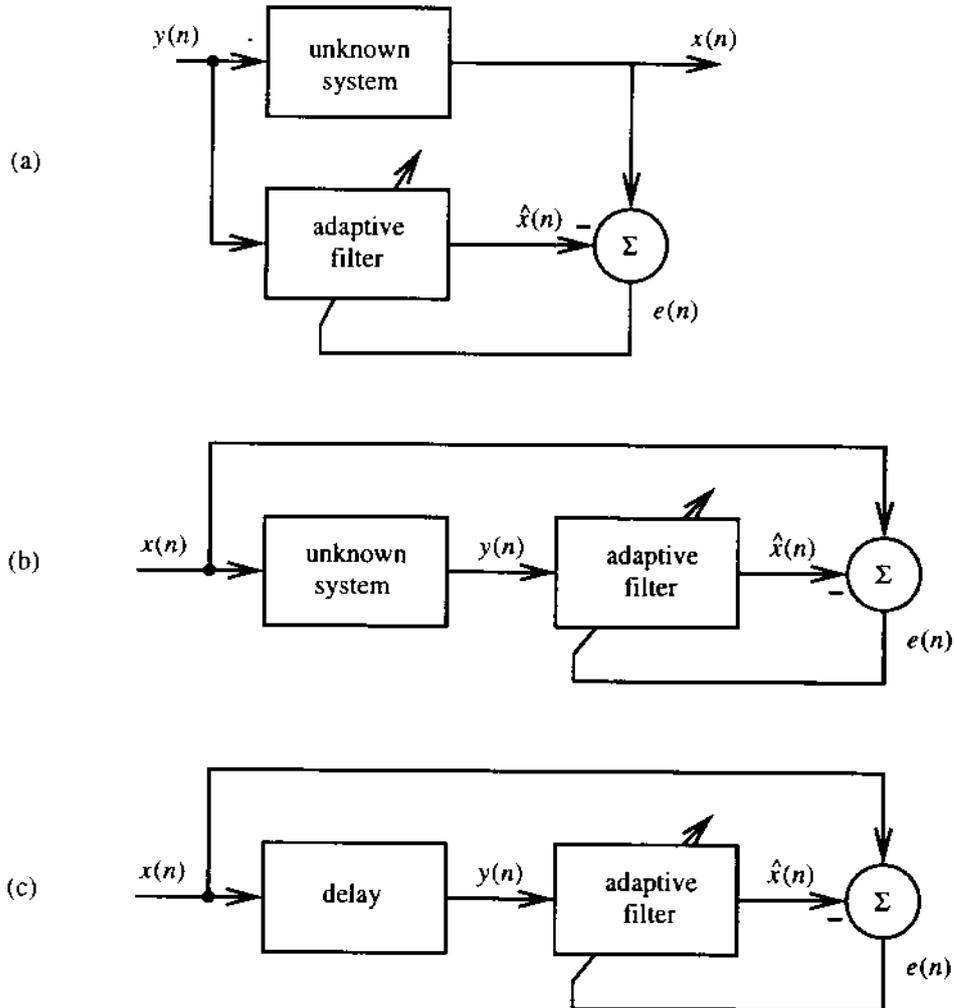


Figure 8.13 Modes of operation of the adaptive filter: (a) direct system modelling; (b) inverse system modelling; (c) linear prediction.

a) Annulation d'écho

b) Egalisation de Canal

c) Suppression de bruit large bande / bande étroite

## Suppression de bruit large bande / bande étroite

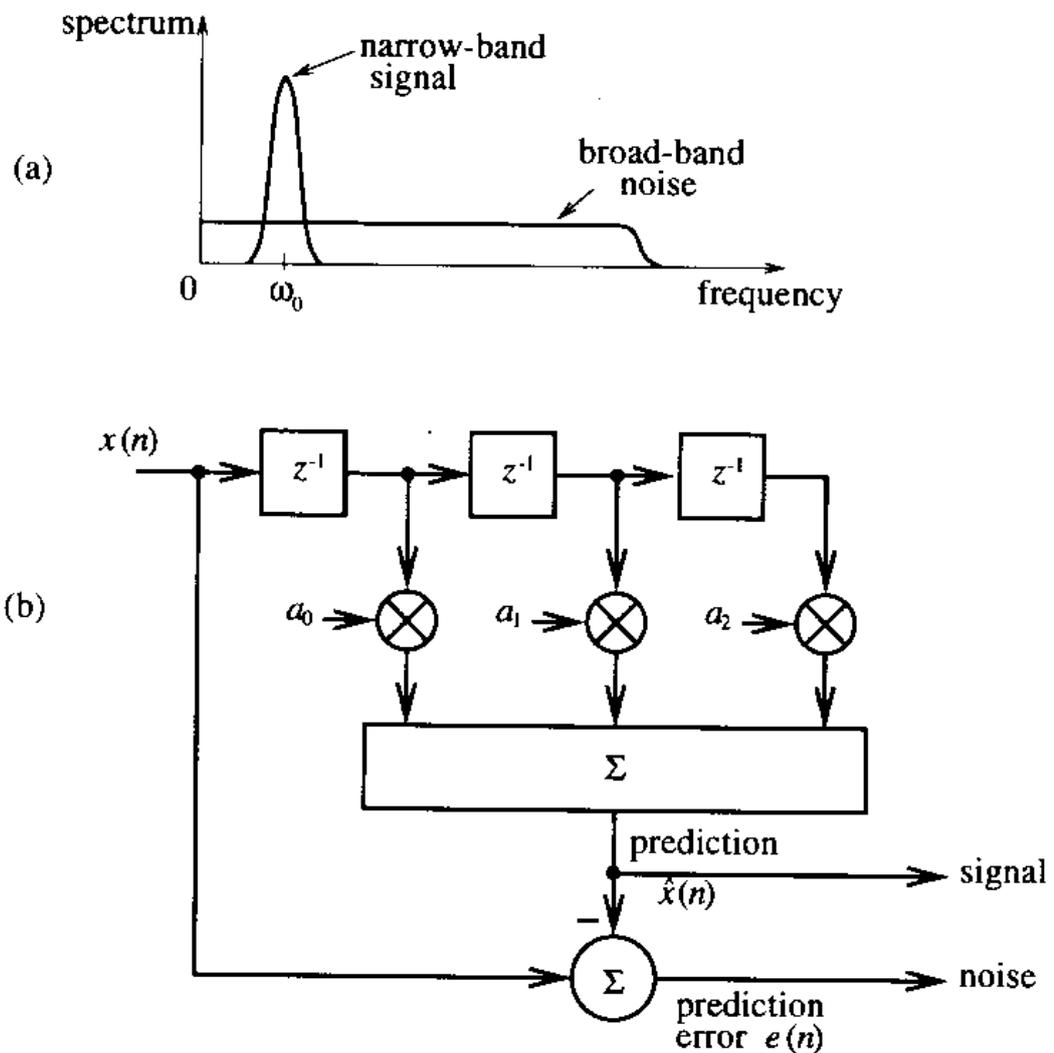
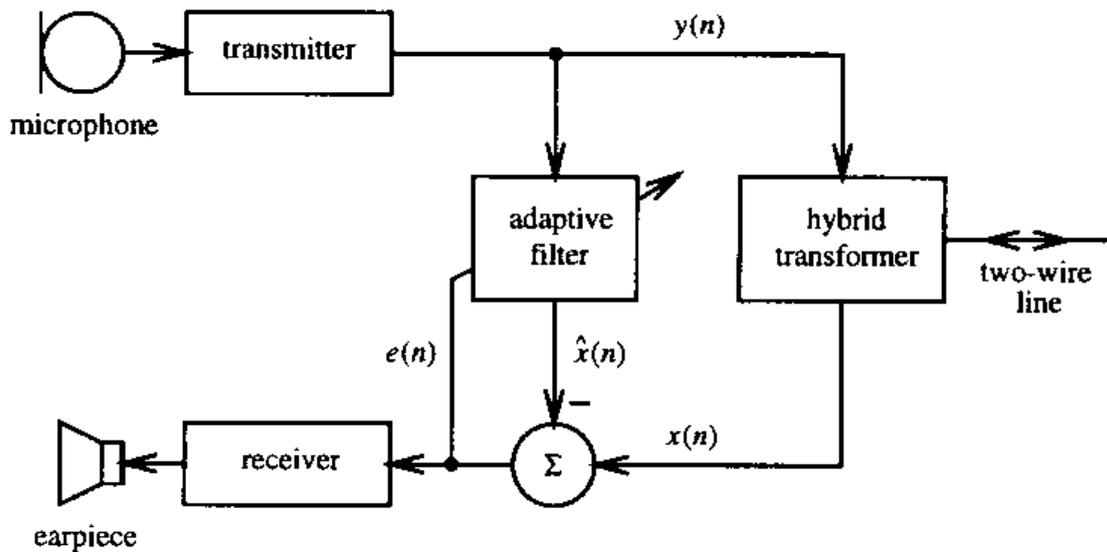


Figure 8.14 Adaptive line enhancer: (a) signal spectrum; (b) enhancer configuration.

- Système connu  $\Pi$  filtre passe-bande
- Idée !  
Signal fortement corrélé facile à prédire ! pas le bruit !
- Apprentissage  
Seul le signal d'entrée est nécessaire : il sert d'entrée et d'apprentissage !

## Annulation d'écho



**Figure 8.17** Application of adaptive echo cancellation in a telephone handset.

- Pas trop grave en parole
- Très grave en modem data
  
- Apprentissage  
Emission d'une séquence aléatoire type avant de fonctionner en émission / réception

## Egalisation de canal

- Transmission de séquence binaire aléatoire sur un canal
- Filtre de Wiener irréaliste
- Décision binaire à prendre  $y(n) = \pm 1$  ?

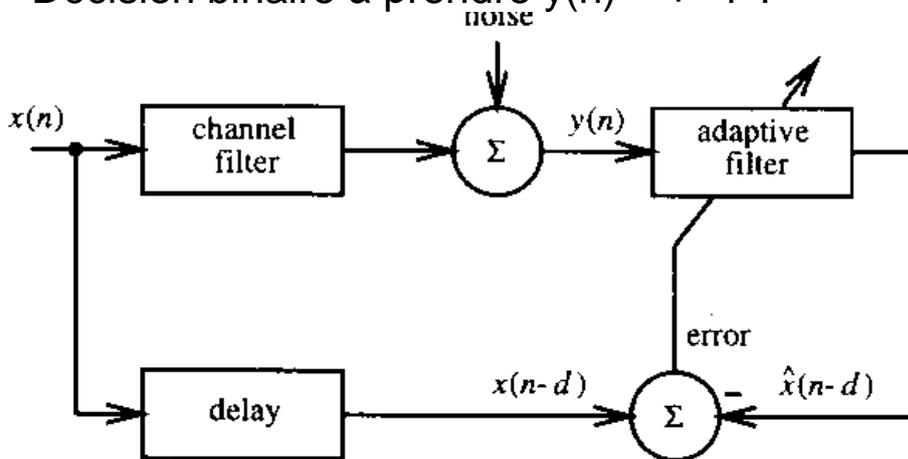


Figure 8.18 The adaptive equaliser.

- Réglage initial avec séquence mémorisée émetteur / récepteur
- Réglage en continu pour la téléphonie mobile

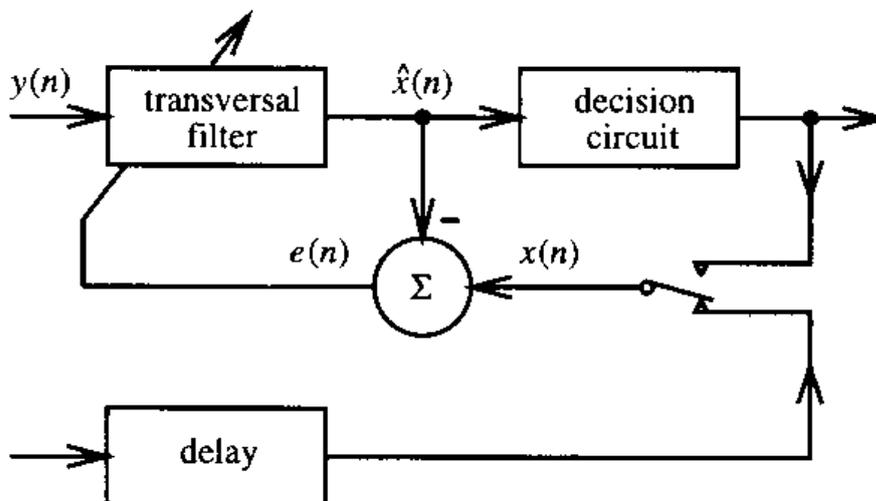
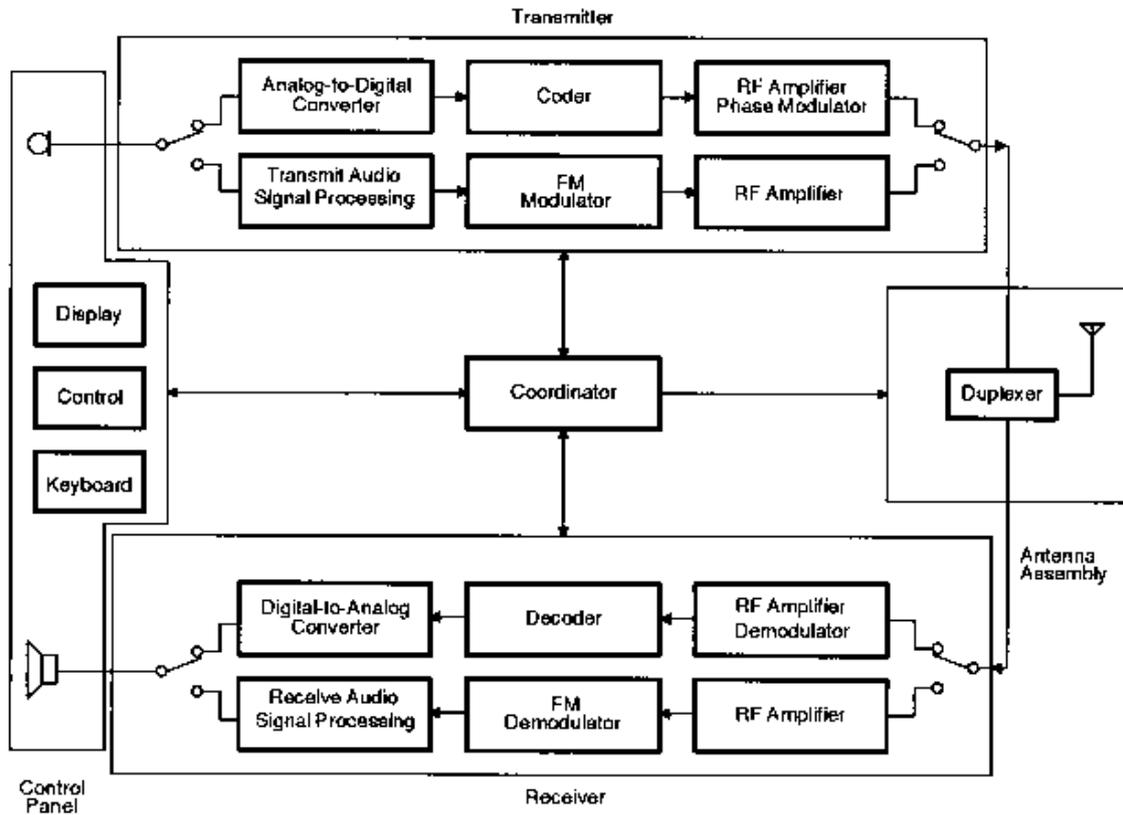


Figure 8.19 Decision directed equaliser design.

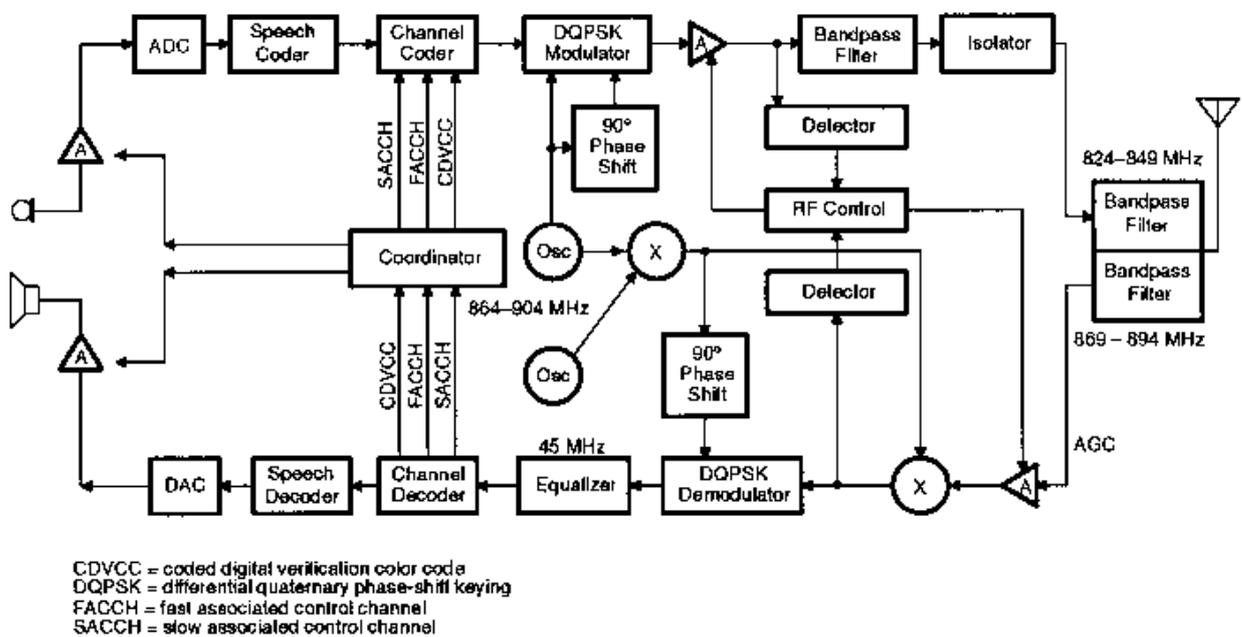
## **Telecommunications Applications with the TMS320C50 DSP**

- TMS320C50 \* DSP 16 Bits, Virgule fixe, 25 ns/ins.
- Orienté Téléphonie mobile
  - Codeurs de parole
  - Codage de canal : CRC, convolutif, Vitterbi
  - Modulation/Démodulation
  - Egalisation de Canal
  - Reconnaissance de la parole

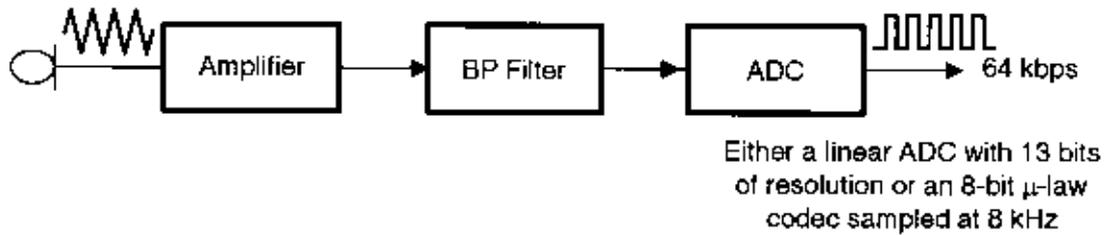
**Figure 1. Functional Components of a Dual-Mode (IS-54) Cellular Phone**



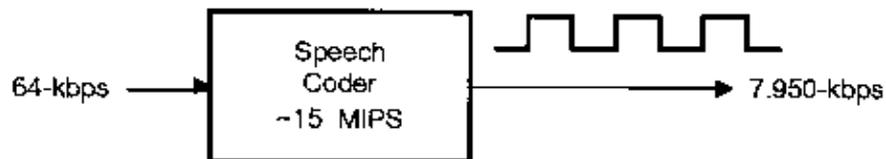
**Figure 2. Functional Blocks of the Digital Portion of a Dual-Mode Phone**



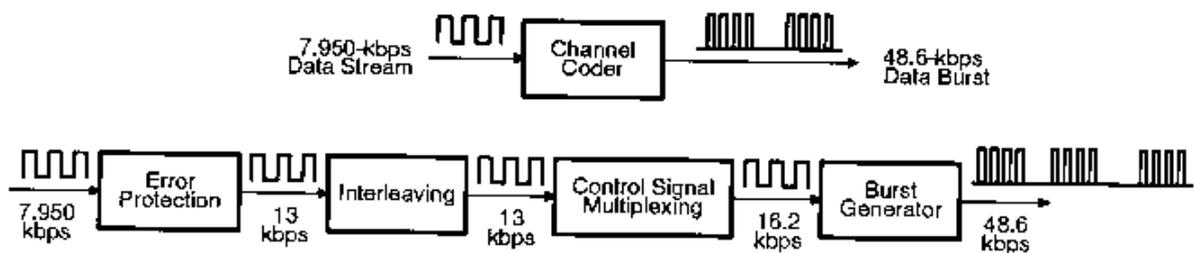
**Figure 3. Front-End Analog Section Converts Audio to a 64-kbps Data Stream**



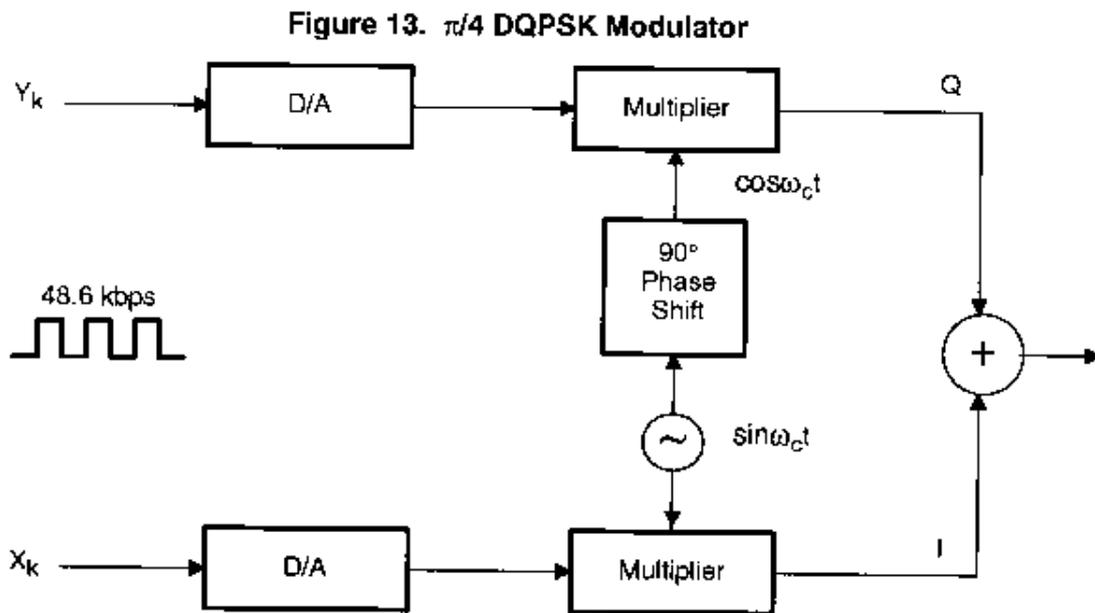
**Figure 4. Full-Rate Speech Coder (VSELP) Reduces a 64-kbps Data Stream to an 8-kbps Data Stream**



**Figure 5. A Channel Coder and Its Functional Components With Associated Data Rates**

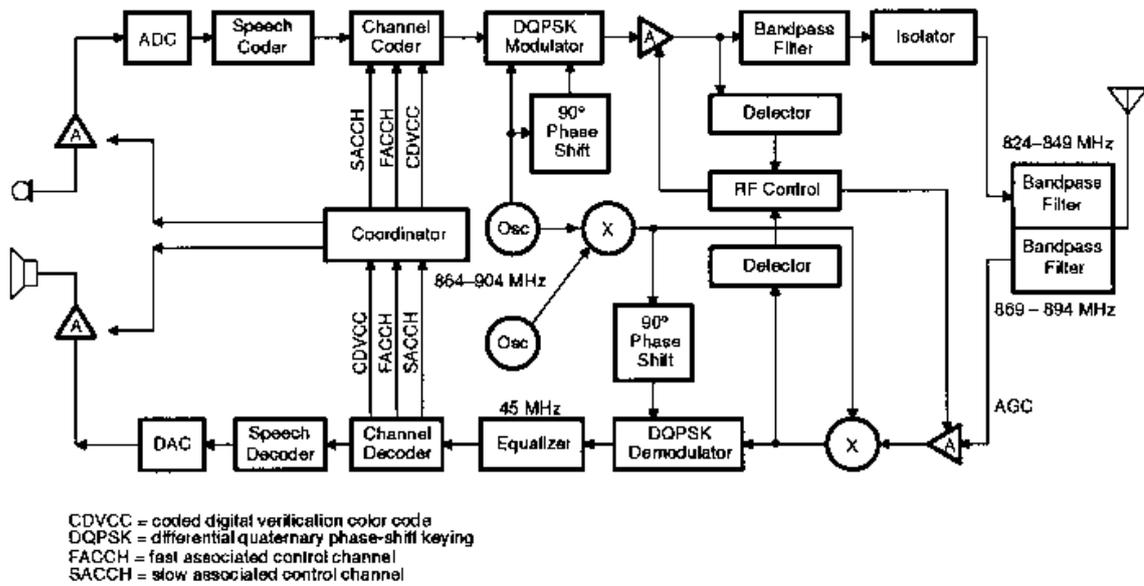


- CRC 7 bits :  $g(x) = 1 + x + x^2 + x^4 + x^5 + x^7$
- Convolutif :  $R=1/2$



- Differential Quaternary Phase Shift Keying  
2 bits  $\rightarrow$  1 symbole, 4 niveaux de modulation
- Translation en fréquence  $\rightarrow$  800 Mhz

Figure 2. Functional Blocks of the Digital Portion of a Dual-Mode Phone



• Egaliseur de canal ?

+ Annulateur d'écho ?

└ Filtrage adapté