

Module 4: Processus stochastiques à temps discret: analyse au second ordre

1 Notions importantes sur les processus stochastiques à temps discret

Ce module est une réplique du module précédent, mais dans lequel le temps t continu est remplacé par un paramètre $n \in \mathbb{Z}$ entier. En fait, il est même plus simple de traiter un processus stochastique à temps discret, qui n'est rien d'autre qu'une suite de variable aléatoires. Nous appellerons, par abus de langage, le paramètre n également "temps", bien que ce paramètre soit en fait simplement le numéro de l'échantillon $X(n)$ du processus $X(\cdot)$, qui est effectivement souvent le temps discrétisé, mais qui peut aussi être par exemple l'indice du pixel d'une image.

1.1 Processus stochastiques à valeurs réelles et temps discret

En échantillonnant le processus $X(\cdot)$ aux temps $n_1, n_2, \dots, n_m \in \mathbb{Z}$, on obtient m variables aléatoires $X_1 = X(n_1), X_2 = X(n_2), \dots, X_m = X(n_m)$. Le processus stochastique est spécifié par la collection de toutes les fonctions de répartition jointes du m ème ordre obtenues pour tout m et tous les instants $n_1, \dots, n_m$:

$$F_{X_1 \dots X_m}(x_1, \dots, x_m; n_1, \dots, n_m) = P(X(n_1) \leq x_1; \dots; X(n_m) \leq x_m).$$

La densité de probabilité jointe du m ème ordre est tout simplement

$$f_{X_1 \dots X_m}(x_1, \dots, x_m; n_1, \dots, n_m) = \frac{\partial^m F_{X_1 \dots X_m}(x_1, \dots, x_m; n_1, \dots, n_m)}{\partial x_1 \dots \partial x_m}.$$

Les moments d'ordre 1 et 2 du processus sont sa *moyenne*

$$\mu_X(n) = E[X(n)] = \int_{-\infty}^{+\infty} x f_{X(n)}(x; n) dx \quad (1)$$

et sa *fonction d'auto-corrélation*

$$R_X(n_1, n_2) = E[X(n_1)X(n_2)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 x_2 f_{X(n_1)X(n_2)}(x_1, x_2; n_1, n_2) dx_1 dx_2. \quad (2)$$

On suppose que ces deux grandeurs sont toujours finies pour tout n, n_1, n_2 . La *fonction d'auto-covariance* est

$$C_X(n_1, n_2) = E[(X(n_1) - \mu_X(n_1))(X(n_2) - \mu_X(n_2))] = R_X(n_1, n_2) - \mu_X(n_1)\mu_X(n_2). \quad (3)$$

Lorsque $n_1 = n_2 = n$, $C_X(n, n) = \text{VAR}[X(n)]$ est la variance de $X(n)$ tandis que $R_X(n, n) = E[X^2(n)]$ est la puissance moyenne de $X(n)$. A partir des m v.a. $X(n_1), X(n_2), \dots, X(n_m)$, on

peut construire une *matrice de corrélation* de ce processus comme suit

$$\Phi_X = \begin{bmatrix} R_X(n_1, n_1) & R_X(n_1, n_2) & R_X(n_1, n_3) & \dots & R_X(n_1, n_m) \\ R_X(n_2, n_1) & R_X(n_2, n_2) & R_X(n_2, n_3) & \dots & R_X(n_2, n_m) \\ R_X(n_3, n_1) & R_X(n_3, n_2) & R_X(n_3, n_3) & \dots & R_X(n_3, n_m) \\ \vdots & \vdots & \vdots & & \vdots \\ R_X(n_m, n_1) & R_X(n_m, n_2) & R_X(n_m, n_3) & \dots & R_X(n_m, n_m) \end{bmatrix} \quad (4)$$

et de manière similaire une matrice de covariance

$$\Gamma_X = \begin{bmatrix} C_X(n_1, n_1) & C_X(n_1, n_2) & C_X(n_1, n_3) & \dots & C_X(n_1, n_m) \\ C_X(n_2, n_1) & C_X(n_2, n_2) & C_X(n_2, n_3) & \dots & C_X(n_2, n_m) \\ C_X(n_3, n_1) & C_X(n_3, n_2) & C_X(n_3, n_3) & \dots & C_X(n_3, n_m) \\ \vdots & \vdots & \vdots & & \vdots \\ C_X(n_m, n_1) & C_X(n_m, n_2) & C_X(n_m, n_3) & \dots & C_X(n_m, n_m) \end{bmatrix}. \quad (5)$$

Dans le cas de processus stochastiques multiples, on peut encore définir la *cross-corrélation* et la *cross-covariance* respectivement par

$$R_{XY}(n_1, n_2) = E[X(n_1)Y(n_2)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 y_2 f_{X(n_1)Y(n_2)}(x_1, y_2; n_1, n_2) dx_1 dy_2. \quad (6)$$

et

$$C_{XY}(n_1, n_2) = E[(X(n_1) - \mu_X(n_1))(Y(n_2) - \mu_Y(n_2))] = R_{XY}(n_1, n_2) - \mu_X(n_1)\mu_Y(n_2). \quad (7)$$

Les deux processus sont dits *orthogonaux* si $R_{XY}(n_1, n_2) = 0$ pour tout n_1, n_2 et *non corrélés* si $C_{XY}(n_1, n_2) = 0$ pour tout n_1, n_2 .

1.2 Quelques processus stochastiques à temps discret particuliers

1.2.1 Sinusoïde à phase aléatoire

Ce processus est identique à sa contrepartie continue:

$$X(n) = a \sin(2\pi f_0 n + \Phi) \quad (8)$$

où Φ est une v.a. uniformément distribuée sur l'intervalle $[0, 2\pi]$, et a et f_0 deux constantes. On calcule que $\mu_X(n) = E[X(n)] = 0$ et

$$R_X(n_1, n_2) = \frac{a^2}{2} \cos(2\pi f_0(n_1 - n_2)). \quad (9)$$

1.2.2 Processus gaussien

$X(n)$ est un processus gaussien si les v.a. $X_1 = X(n_1), X_2 = X(n_2), \dots, X_m = X(n_m)$ sont des v.a. gaussiennes multivariées pour tout m et tous les temps $n_1, \dots, n_m$. Il est donc entièrement spécifié par sa moyenne et sa fonction de covariance. Pour rappel, sa densité jointe de probabilité s'écrit

$$f_{\mathbf{X}}(\mathbf{x}; n_1, \dots, n_m) = \frac{1}{\sqrt{(2\pi)^m |\det(\Gamma_X)|}} e^{-\frac{1}{2} (\mathbf{x} - \mu_X)^T \Gamma_X^{-1} (\mathbf{x} - \mu_X)} \quad (10)$$

où

$$\begin{aligned} \mathbf{X} &= [X(n_1) \dots X(n_m)]^T \\ \mathbf{x} &= [x_1 \dots x_m]^T \\ \mu_X &= [E[X(n_1)] \dots E[X(n_m)]]^T \end{aligned}$$

1.2.3 Bruit blanc

Contrairement au cas continu, l'impulsion de Dirac est bien définie à l'origine, et a pour conséquence le fait que la variance du bruit blanc à temps discret σ_N^2 est finie. $N(n)$ est un bruit blanc si sa moyenne est nulle et si sa fonction d'auto-corrélation (ou d'auto-covariance) s'écrit

$$R_N(n_1, n_2) = \begin{cases} \sigma_N^2 & \text{si } n_1 = n_2 \\ 0 & \text{si } n_1 \neq n_2. \end{cases} \quad (11)$$

Un bruit blanc est donc une suite de variables aléatoires centrées non corrélées.

1.2.4 Processus MA, AR, ARMA, ARIMA

Soit $U(n)$ un processus stochastique à temps discret, appelé *processus d'innovation*. On supposera par défaut que $U(n)$ est un bruit blanc de variance σ_U^2 . Le processus

$$X(n) = \sum_{k=0}^m b_k U(n-k) \quad (12)$$

est appelé *moyenne mobile* ("Moving Average", MA) d'ordre m de U , pondérée par les coefficients b_k , $0 \leq k \leq m$. Le processus calculé récursivement à partir de U et de p valeurs initiales $X(0), \dots, X(p-1)$

$$X(n) = - \sum_{k=1}^p a_k X(n-k) + U(n) \quad (13)$$

est appelé *processus auto-régressif* (AR) d'ordre p . Les coefficients $a_1, \dots, a_p$ doivent être choisis de telle manière que ce système soit stable (cfr cours de circuits et systèmes ou de traitement

numérique des signaux). La combinaison de ces deux processus donne un processus ARMA (m, p):

$$X(n) = - \sum_{k=1}^p a_k X(n-k) + \sum_{k=0}^m b_k U(n-k). \quad (14)$$

Ces processus sont fort utilisés dans le traitement des séries temporelles. Enfin, par analogie avec l'intégration en temps continue, on appelle processus ARIMA ("Autoregressive Integrated Moving Average") le processus $Y(n) = Y(n-1) + X(n)$, où $X(n)$ est le processus ARMA (14).

1.3 Stationnarité

Les notions de stationnarités sont identiques à celles vues dans le cas de processus à temps continu. Un processus stochastique $X(n)$ est *stationnaire* ou plus précisément *stationnaire au sens strict* (SSS) ssi toutes ses propriétés statistiques sont indépendantes de l'origine des temps. En d'autres termes, quels que soient $m \in \mathbb{N}_0$, $n_1, \dots, n_m \in \mathbb{Z}$ et $c \in \mathbb{Z}$ ¹,

$$F_{X(n_1) \dots X(n_m)}(x_1, \dots, x_m; n_1, \dots, n_m) = F_{X(n_1+c) \dots X(n_m+c)}(x_1, \dots, x_m; n_1+c, \dots, n_m+c).$$

Un processus stochastique $X(n)$ est *stationnaire au sens large* (WSS pour "wide sense stationary"), ou encore stationnaire du second ordre, ssi sa moyenne ne dépend pas du temps n et sa fonction d'auto-corrélation (ou de covariance) ne dépend que de la différence entre les deux instants auxquels elle est évaluée:

$$\begin{aligned} \mu_X(n) &= \mu_X \\ R_X(n_1, n_2) &= R_X(n_1 - n_2) \end{aligned}$$

Les conséquences et propriétés de ces deux types de processus sont donc les mêmes que dans le cas continu. Elles ont pour conséquence que la matrice de corrélation d'ordre m d'un processus WSS, qui peut s'écrire

$$\Phi_X = \begin{bmatrix} R_X(0) & R_X(1) & R_X(2) & \dots & R_X(m-1) \\ R_X(1) & R_X(0) & R_X(1) & \dots & R_X(m-2) \\ R_X(2) & R_X(1) & R_X(0) & \dots & R_X(m-3) \\ \vdots & \vdots & \vdots & & \vdots \\ R_X(m-1) & R_X(m-2) & R_X(m-3) & \dots & R_X(0) \end{bmatrix}, \quad (15)$$

possède les propriétés suivantes: P3. Elle est symétrique. Ceci est une conséquence directe de la parité de $R_X(k)$. P4. Elle est semi-définie positive. Ceci est une conséquence du fait que $|R_X(k)| \leq R_X(0) = E[X^2(n)]$. P5. Elle est une matrice de Toeplitz. Ceci est une conséquence du fait que $R_X(n, n-k) = R_X(k)$.

¹Si le processus est restreint aux $n \in \mathbb{N}$ (au lieu de $n \in \mathbb{Z}$), il faut prendre $n_1, \dots, n_m, c \in \mathbb{N}$.

1.4 Ergodisme par rapport à la moyenne

Soit $X(\cdot)$ un processus stationnaire au sens large. La moyenne *d'ensemble* de ce processus

$$\mu_X = E[X(n)]$$

peut être estimée par la moyenne *temporelle* d'une de ses réalisations $X(n)$

$$\langle X(n) \rangle_m = \frac{1}{2m+1} \sum_{n=-m}^m X(n).$$

Cet estimateur est non biaisé car

$$E[\langle X(n) \rangle_m] = \frac{1}{2m+1} \sum_{n=-m}^m E[X(n)] = \frac{1}{2m+1} \sum_{n=-m}^m \mu_X = \mu_X.$$

Pour que $\langle X(n) \rangle_m \rightarrow \mu_X$ quand $m \rightarrow \infty$ au sens des moindres carrés (ou encore en moyenne quadratique), il faut encore que

$$\text{VAR}[\langle X(n) \rangle_m] = E[(\langle X(n) \rangle_m - \mu_X)^2] \rightarrow 0$$

quand $m \rightarrow \infty$. Dans ce cas, le processus est *ergodique* par rapport à sa moyenne. En faisant un développement similaire à celui effectué dans le cas continu, on peut établir la condition nécessaire et suffisante d'ergodisme de $X(n)$ par rapport à sa moyenne:

Théorème 1 *Un processus WSS $X(n)$ est ergodique par rapport à sa moyenne si et seulement si*

$$\lim_{m \rightarrow \infty} \frac{1}{2m+1} \sum_{k=-2m}^{2m} C_X(k) \left(1 - \frac{|k|}{2m+1}\right) = 0. \quad (16)$$

Une condition plus facilement vérifiable, mais seulement suffisante, est la suivante.

Corollaire 1 *Un processus WSS $X(n)$ est ergodique par rapport à sa moyenne si*

$$\lim_{k \rightarrow \infty} C_X(k) = 0. \quad (17)$$

Le processus est ergodique par rapport à sa fonction d'auto-corrélation lorsque

$$\langle X(n)X(n-k) \rangle_m = \frac{1}{2m+1} \sum_{n=-m}^m X(n)X(n-k)$$

tend vers $R_X(k)$ au sens des moindres carrés quand $m \rightarrow \infty$.

1.5 Densité spectrale de puissance

1.5.1 Définitions et propriétés

Soit $X(n)$ un processus discret stationnaire au sens large. Sa *densité spectrale de puissance* $S_X(f)$ est la transformée de Fourier discrète de sa fonction d'auto-corrélation:

$$S_X(f) = \sum_{k=-\infty}^{+\infty} R_X(k) e^{-j2\pi f k}. \quad (18)$$

A nouveau, celle-ci est bien définie si

$$\sum_{k=-\infty}^{\infty} |R_X(k)| < \infty.$$

Comme $S_X(f)$ est périodique (de période 1), on peut considérer les seules fréquences du domaine $-1/2 \leq f \leq 1/2$. La fonction d'auto-corrélation peut être obtenue par la transformée inverse

$$R_X(k) = \int_{-1/2}^{1/2} S_X(f) e^{j2\pi f k} df. \quad (19)$$

On peut également étendre la définition (18) à la variable z pour obtenir la transformée en z de $R_X(k)$:

$$\hat{S}_X(z) = \sum_{k=-\infty}^{+\infty} R_X(k) z^{-k}. \quad (20)$$

La densité spectrale s'obtient en posant $z = e^{j2\pi f}$:

$$\hat{S}_X(e^{j2\pi f}) = S_X(f).$$

La puissance moyenne de $X(n)$ est

$$E[X^2(n)] = R_X(0) = \int_{-1/2}^{1/2} S_X(f) df$$

Si X est réel, sa densité spectrale de puissance possède les propriétés suivantes: P6. Comme $R_X(k)$ est une fonction paire, sa transformée de Fourier est réelle et est une fonction paire de f . P7. $S_X(f) \geq 0$ pour toute fréquence f . La *densité spectrale mutuelle de puissance* (en anglais: "cross-power spectral density") de deux processus X et Y est la transformée de Fourier de leur fonction de cross-corrélation:

$$S_{XY}(f) = \sum_{k=-\infty}^{+\infty} R_{XY}(k) e^{-j2\pi f k}. \quad (21)$$

1.5.2 Exemples de densités spectrales

- **Sinusoïde à phase aléatoire.** La densité spectrale du processus (8) est

$$S_X(f) = \frac{A^2}{2} \sum_{k=-\infty}^{+\infty} \cos(2\pi f_0 k) e^{-j2\pi f k} = \frac{A^2}{4} (\delta(f - f_0) + \delta(f + f_0)). \quad (22)$$

- **Bruit blanc.** Un bruit blanc $N(n)$ a une densité spectrale constante sur toute la bande de fréquences $-1/2 \leq f \leq 1/2$ qui vaut

$$S_N(f) = \sigma_N^2$$

car sa fonction d'auto-corrélation est

$$R_N(k) = \sigma_N^2 \bar{\delta}(k) = \begin{cases} \sigma_N^2 & \text{si } k = 0 \\ 0 & \text{si } k \neq 0 \end{cases}$$

où $\bar{\delta}(k)$ est l'impulsion de Dirac à temps discret, valant 1 en $k = 0$ et 0 ailleurs.

1.6 Réponse d'un système linéaire à des signaux aléatoires

1.6.1 Analyse harmonique

Soit un système linéaire et invariant dans le temps dont la réponse impulsionnelle est $h(k)$. Sa fonction de transfert est la transformée de Fourier de sa réponse impulsionnelle, c'est-à-dire

$$H(f) = \sum_{l=-\infty}^{+\infty} h(l) e^{-j2\pi f l}$$

qui est bien définie si

$$\sum_{l=-\infty}^{+\infty} |h(l)| < \infty.$$

Si l'entrée du système $X(n)$ est un processus stationnaire au sens large, la sortie du système $Y(n)$ est un processus stochastique donné par le produit de convolution

$$Y(n) = \sum_{l=-\infty}^{+\infty} h(l) X(n - l).$$

De manière similaire à celle suivie pour un processus continu, on établit que sa moyenne vaut

$$E[Y(n)] = H(0) \mu_X \quad (23)$$

et est donc indépendante du temps et que ce processus est stationnaire au sens large, sa fonction d'auto-corrélation étant

$$R_Y(k) = \sum_{l=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} h(l) h(m) R_X(k + l - m).$$

En prenant la transformée de Fourier discrète des deux membres de cette équation, on retrouve le même théorème que dans le cas continu:

Théorème 2 *La densité spectrale de la réponse $Y(n)$ d'un système linéaire et invariant dans le temps dont la réponse impulsionnelle $h(n)$ est absolument sommable et dont la transmittance est $H(f)$, à un processus $X(n)$ WSS de densité spectrale $S_X(f)$ est*

$$S_Y(f) = |H(f)|^2 S_X(f). \quad (24)$$

Ce théorème s'écrit également, dans le domaine des z , comme

$$\hat{S}_Y(z) = \hat{H}(z) \hat{H}(1/z) \hat{S}_X(z), \quad (25)$$

où la fonction de transfert $H(z)$ est définie par

$$\hat{H}(z) = \sum_{l=-\infty}^{+\infty} h(l) z^{-l} \quad (26)$$

On a donc $H(f) = \hat{H}(e^{2\pi j f})$. On peut également calculer la densité spectrale mutuelle $S_{YX}(f)$. On trouve

$$S_{YX}(f) = H(f) S_X(f) \quad (27)$$

$$S_{XY}(f) = H^*(f) S_X(f). \quad (28)$$

1.6.2 Exemples

- **Processus ARMA.** Le processus ARMA (m, p) de l'équation (14) revient à filtrer le processus d'innovation $U(n)$ par un filtre dont la transformée en z est

$$\hat{H}_{ARMA}(z) = \frac{\sum_{k=0}^m b_k z^{-k}}{1 + \sum_{k=1}^p a_k z^{-k}} = \frac{\hat{B}(z)}{\hat{A}(z)} \quad (29)$$

où

$$\begin{aligned} \hat{A}(z) &= 1 + \sum_{k=1}^p a_k z^{-k} \\ \hat{B}(z) &= \sum_{k=0}^m b_k z^{-k}. \end{aligned}$$

Par conséquent, sa densité spectrale est

$$S_{X_{ARMA}}(f) = \sigma_U^2 \frac{\hat{B}(e^{2\pi j f}) \hat{B}(e^{-2\pi j f})}{\hat{A}(e^{2\pi j f}) \hat{A}(e^{-2\pi j f})} = \sigma_U^2 \frac{|\sum_{k=0}^m b_k e^{-2\pi j f k}|^2}{|1 + \sum_{k=1}^p a_k e^{-2\pi j f k}|^2} \quad (30)$$

Un processus MA est un processus ARMA particulier pour lequel $\hat{A}(z) = 1$ (par conséquent la fonction de transfert du filtre n'a que des zéros et pas de pôles); un processus AR est un processus ARMA particulier pour lequel $\hat{B}(z) = 1$ (par conséquent la fonction de transfert du filtre n'a que des pôles et pas de zéros).

- **Factorisation spectrale** On vient de voir que la densité spectrale d'un processus ARMA est directement factorisable en un produit de la transmittance d'un filtre LTI par son complexe conjugué. En fait cette propriété est vraie pour une grande catégorie de processus. En effet, on peut montrer, en développant la fonction d'une variable complexe $\ln \hat{S}_X(z)$ en série de Laurent, que si

$$\int_{-1/2}^{1/2} |\ln S_X(f)| df < \infty$$

alors il existe une fonction $\hat{G}(z)$ telle que

$$\hat{S}_X(z) = \hat{G}(z)\hat{G}(z^{-1}) \quad (31)$$

En remplaçant z par $e^{2\pi jf}$ dans cette expression, on a également

$$S_X(f) = G(f)G^*(f). \quad (32)$$

1.7 Filtre optimal de Wiener

Deux problèmes d'estimation très fréquents en traitement du signal sont la *prédiction* d'un signal à partir d'échantillons précédents, et le *filtrage* d'un signal bruité. Les deux applications qui suivent illustrent ces deux problèmes, et peuvent être toutes deux vues comme un cas particulier du filtre de Wiener que nous allons étudier dans cette section. Un *filtre de Wiener*, de transmittance $H(f)$ et de réponse impulsionnelle $h(k)$, transforme linéairement une entrée $X(n)$ en une sortie

$$Y(n) = \sum_{k=-\infty}^{\infty} h(k)X(n-k). \quad (33)$$

Le problème consiste à le choisir de telle manière que la sortie $Y(n)$ soit aussi proche que possible d'une réponse désirée $D(n)$. Le critère d'erreur est celui des *moindres carrés*: en désignant l'erreur par la variable aléatoire $e(n)$

$$e(n) = D(n) - Y(n), \quad (34)$$

on cherche le filtre qui minimise l'erreur quadratique moyenne

$$\varepsilon = E[e^2(n)] = E[(D(n) - Y(n))^2]. \quad (35)$$

Dans le cas du *filtrage d'un signal bruité* (figure 1(a)), le signal d'entrée $X(n) = S_{ideal}(n) + N(n)$ est un signal $S_{ideal}(n)$ corrompu par du bruit additif $N(n)$, et la réponse désirée est $D(n) = S_{ideal}(n)$. Dans le cas de la *prédiction* (figure 1(b)), les échantillons passés $X(n-1), \dots, X(n-N)$ sont présentés à l'entrée du filtre, et la réponse désirée est la valeur présente (ou future) du signal d'entrée $X(\cdot)$: $D(n) = X(n)$. Les processus $X(n)$ et donc $Y(n)$ sont supposés être stationnaires au sens large. A partir de (33), cette erreur peut être développée pour donner

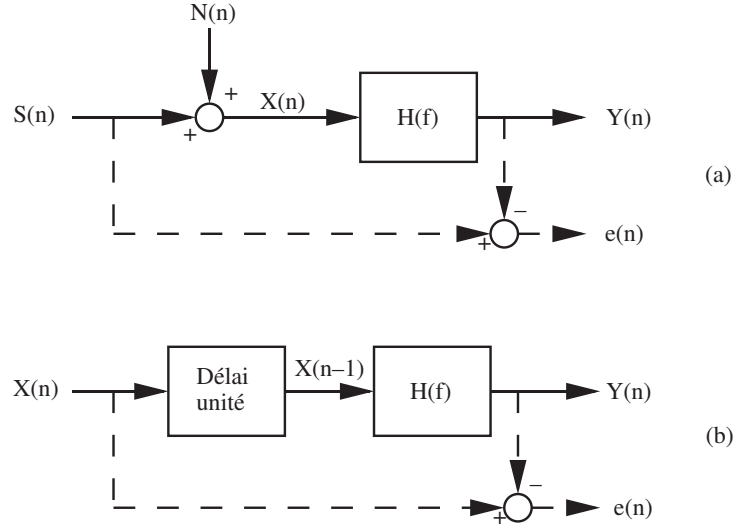


Figure 1: Filtre optimal de Wiener: filtrage d'un signal bruité (a) et prédiction (b)

$$\begin{aligned}
\varepsilon &= E[D(n)^2] - 2E[D(n)Y(n)] + E[Y(n)^2] \\
&= E[D(n)^2] - 2 \sum_{k=-\infty}^{\infty} h(k)E[D(n)X(n-k)] + \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k)h(l)E[X(n-k)X(n-l)] \\
&= R_D(0) - 2 \sum_{k=-\infty}^{\infty} h(k)R_{DX}(k) + \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k)h(l)R_X(k-l).
\end{aligned}$$

Cette erreur est minimisée en annulant toutes ses dérivées partielles par rapport aux coefficients $h(k)$

$$\frac{\partial \varepsilon}{\partial h(k)} = -2R_{DX}(k) + 2 \sum_{l=-\infty}^{\infty} h(l)R_X(k-l) = 0$$

d'où, pour $-\infty \leq k \leq \infty$,

$$\sum_{l=-\infty}^{\infty} h(l)R_X(k-l) = R_{DX}(k). \quad (36)$$

La solution $\{h(k)\}$ de cette équation est le filtre de Wiener. En prenant la transformée de Fourier des deux membres de cette équation, celle-ci devient

$$H(f)S_X(f) = S_{DX}(f)$$

dont on déduit la transmittance du filtre de Wiener

$$H(f) = \frac{S_{DX}(f)}{S_X(f)}. \quad (37)$$

Un tel filtre n'est cependant pas causal, ce qui empêche le traitement en temps réel. Dans le cas causal, il faut imposer $h(k) = 0$ pour $k \leq 0$ et les équations (36) deviennent, avec cette fois $0 \leq k \leq \infty$ seulement,

$$\sum_{l=0}^{\infty} h(l)R_X(k-l) = R_{DX}(k) \quad (38)$$

et sont appelées *équations de Wiener-Hopf* ou *équations normales*. Elles sont plus difficiles à transposer dans le plan fréquentiel, car il faut alors calculer la transformée de Fourier de la fonction

$$[R_{DX}(k)]_+ = \begin{cases} R_{DX}(k) & \text{si } k \geq 0 \\ 0 & \text{si } k < 0 \end{cases}$$

et non plus celle de $R_{DX}(k)$ (qui est $S_{DX}(f)$).

2 Application 1: Restoration d'image par la méthode de Wiener

Dans de nombreuses applications (images spatiales, médicales, ...) la prise d'image s'accompagne d'une dégradation de celle-ci (due par exemple aux turbulences atmosphériques, au mouvement relatif entre l'objet photographié et la caméra, au mauvais réglage de la caméra...). L'opération visant à rétablir la qualité initiale de l'image s'appelle *restauration d'image*. Avant de passer au traitement d'une image qui est un signal bi-dimensionnel, prenons le cas d'une "image" uni-dimensionnelle. Une grande partie des dégradations que subissent les images peuvent être modélisées par 9

$$X(n) = \sum_{i=-\infty}^{\infty} b(i)S(n-i) + N(n) \quad (39)$$

où $S(n)$ est l'image originale idéale, $X(n)$ l'image dégradée, $N(n)$ un bruit additif, $b(i)$ la réponse impulsionnelle du filtre modélisant la dégradation (mouvement de la caméra...) dont la transmittance est $B(f)$. Le système de restauration, montré à la figure 2(a), est formé d'un filtre $H(f)$ qui dont la sortie $Y(n)$ doit être aussi proche que possible de l'image originale $S(n)$. Une première solution dite du *filtrage inverse* consiste à prendre

$$H(f) = H_{inv}(f) = 1/B(f) \quad (40)$$

de sorte que

$$\tilde{Y}(f) = H_{inv}(f)\tilde{X}(f) = \frac{\tilde{X}(f)}{B(f)} = \tilde{S}(f) + \frac{\tilde{N}(f)}{B(f)}. \quad (41)$$

où $\tilde{Y}(f)$, $\tilde{X}(f)$, $\tilde{S}(f)$ et $\tilde{N}(f)$ sont les transformées de Fourier (à ne pas confondre avec les densités spectrales) respectivement de $Y(n)$, $X(n)$, $S(n)$ et $N(n)$. Un gros inconvénient de cette méthode réside dans le fait que la transmittance $B(f)$ peut avoir une très faible valeur pour certaines valeurs de f , ce qui donnera une valeur du terme $\tilde{N}(f)/B(f)$ beaucoup plus grande que $\tilde{S}(f)$, et empêchera pratiquement de recouvrer l'image originale. D'autre part, si l'image n'est corrompue que par un bruit additif, $H_{inv}(f) = 1/B(f) = 1$, et le filtrage inverse ne modifie plus du tout l'image dégradée. Pour pouvoir à la fois réduire le niveau du bruit $N(n)$ et compenser

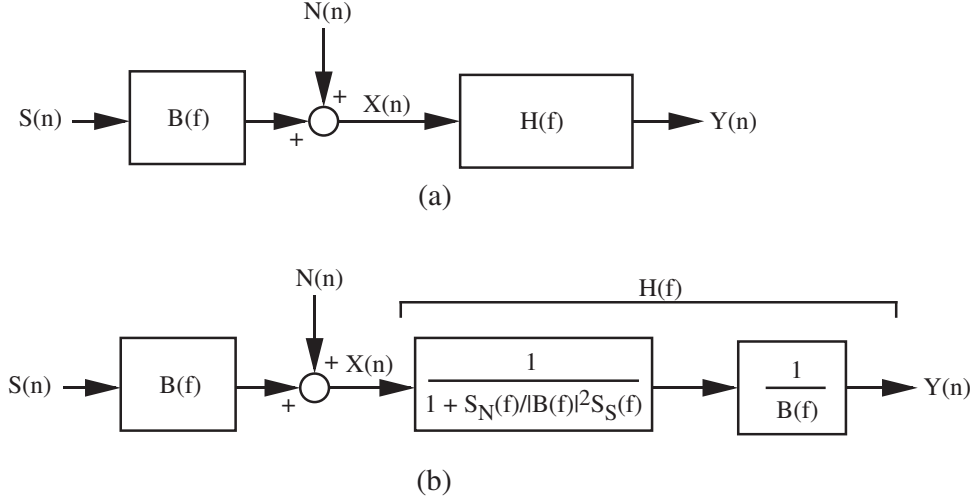


Figure 2: Système de restauration d'image (a) et filtre de Wiener pour la restauration d'image (b).

la dégradation $B(f)$, une autre méthode de restauration cherche à minimiser l'erreur quadratique moyenne entre $S(n)$ et $Y(n)$, et constitue donc un cas particulier de *filtrage de Wiener*. Dans notre cas, le signal idéal défini à la section précédente est $D(n) = S(n)$. Par conséquent, si le bruit $N(n)$ est de moyenne nulle et indépendant de $S(n)$, on calcule aisément que les densités spectrales de puissance sont

$$\begin{aligned} S_{DX}(f) &= S_{SX}(f) = B^*(f)S_S(f) \\ S_X(f) &= |B(f)|^2 S_S(f) + S_N(f) \end{aligned}$$

si bien que la transmittance du filtre de Wiener non causal obtenue par (37) est

$$H(f) = \frac{B^*(f)S_S(f)}{|B(f)|^2 S_S(f) + S_N(f)}$$

ou encore

$$H(f) = \frac{1}{B(f)} \frac{|B(f)|^2 S_S(f)}{|B(f)|^2 S_S(f) + S_N(f)} \quad (42)$$

ce qui permet de représenter le filtre de Wiener comme une cascade de deux filtres (figure 2 (b)): le premier diminue le niveau de bruit, le second est le filtre inverse corrigeant la dégradation. Si l'image n'est pas bruitée ($N(n) = 0$), ce filtre se réduit donc au filtre inverse $H_{inv}(f) = 1/B(f)$. Au contraire, si $B(f) = 1$, le filtre devient le filtre de Wiener réduisant un bruit additif corrompant un signal $S(n)$. On étend facilement ce développement au cas bidimensionnel qui est celui du traitement d'images, les équations (39) et (42) devenant

$$X(n_1, n_2) = \sum_{i_1=-\infty}^{\infty} \sum_{i_2=-\infty}^{\infty} b(i_1, i_2) S(n_1 - i_1, n_2 - i_2) + N(n_1, n_2) \quad (43)$$

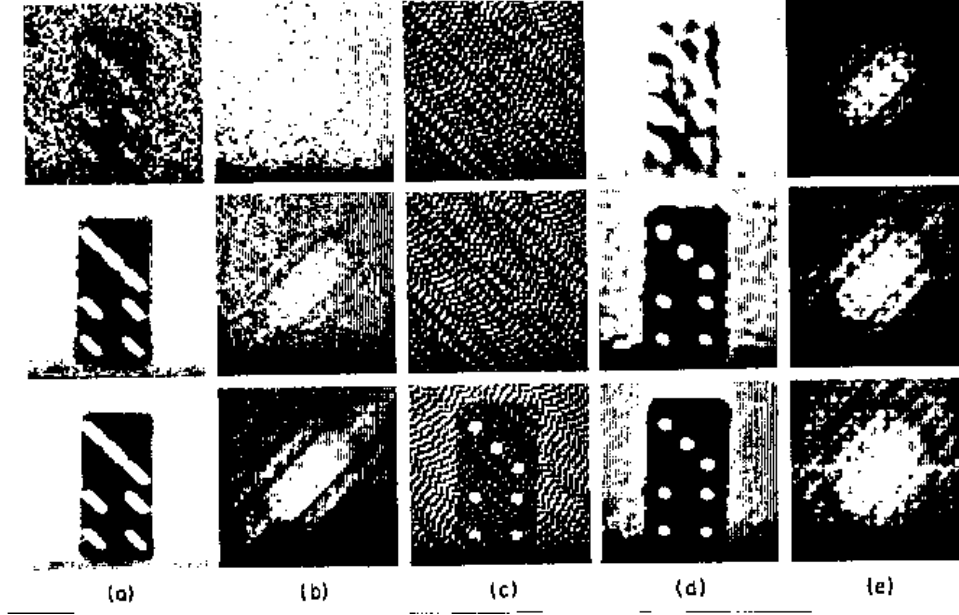


Figure 3: Restoration de trois images

et

$$H(f_1, f_2) = \frac{1}{B(f_1, f_2)} \frac{|B(f_1, f_2)|^2 S_S(f_1, f_2)}{|B(f_1, f_2)|^2 S_S(f_1, f_2) + S_N(f_1, f_2)}. \quad (44)$$

Le filtrage de Wiener présente encore plusieurs défauts. L'un d'entre eux est le choix du critère des moindres carrés pour l'erreur, qui est commode pour les manipulations mathématiques, mais n'est pas le critère de l'oeil humain pour juger de la qualité d'une image. Des variations du type

$$H(f_1, f_2) = \frac{1}{B(f_1, f_2)} \left(\frac{|B(f_1, f_2)|^2 S_S(f_1, f_2)}{|B(f_1, f_2)|^2 S_S(f_1, f_2) + \alpha S_N(f_1, f_2)} \right)^\beta$$

où α et β sont deux paramètres ajustables sont proposées. D'autre part, les caractéristiques d'une image ne sont pas stationnaires, et une implémentation adaptative (en anglais algorithme LMS pour Least Mean Square) est possible. Enfin, une connaissance a priori des spectres $S_S(f)$ et $S_N(f)$ est nécessaire. En utilisant une méthode des moindres carrés avec contrainte, cette connaissance (ou estimation) complète des spectres n'est plus indispensable. La figure 3(a) présente trois images $X(n_1, n_2)$ dégradées par un mouvement de la caméra à vitesse constante, et corrompues par des bruits gaussiens avec des rapports signal/bruit respectivement égaux à 1, 10 et 100. La figure 3(b) montre les trois spectres $S_X(f_1, f_2)$ correspondants. Les figures 3(c) et (d) montrent les images restaurées $Y(n_1, n_2)$ respectivement par la méthode du filtrage inverse (c) et celle du filtrage de Wiener (d), ainsi que le spectre $S_Y(f_1, f_2)$ des trois images restaurées par filtrage de Wiener.

3 Application 2: Codage DPCM

Le codage du signal de parole (c'est-à-dire la numérisation des valeurs des échantillons du signal par un certain nombre de bits) peut s'effectuer de deux manières différentes: soit en reproduisant directement la forme d'onde du signal de parole, c'est le *codage direct de l'onde* ("waveform coding"), soit en utilisant une connaissance a priori du modèle de production de la parole, c'est le *codage de source* ("source coding"). Dans cette section, nous décrivons sommairement une technique appartenant à la première catégorie, la section suivante présentera une méthode de codage de source.

3.1 Codage MIC ou PCM et bruit de quantification

La manière la plus simple de coder les échantillons $X(n)$ d'un signal de parole est la quantification immédiate de chaque échantillon sur un certain nombre de bits. Cette méthode, appelée *Modulation par Impulsions Codées* (MIC) ou *Pulse Code Modulation* (PCM) ne tire cependant aucun parti de la redondance du signal, et produit des débits binaires élevés. Pour un signal de parole, le débit est typiquement de 64 kbps (kbits/sec). Elle a par contre l'avantage d'une très grande simplicité et donc d'une très grande rapidité d'encodage. La quantification la plus habituelle consiste à arrondir la valeur de $X(n)$ à la valeur $i\Delta$ la plus proche, où Δ est le pas de quantification, et où i est un entier (quantification uniforme, figure 4(a)). Le nombre de niveaux de quantification est toujours fini: c'est la puissance de 2 donnée par le nombre de bits de codage. La plage (domaine) de valeurs que le signal quantifié $X_q(n) = Q(X(n))$ peut prendre est donc limitée. Si le domaine du signal analogique X n'est pas limité à la même plage de valeurs, il y peut y avoir des dépassements importants, qui mènent à des distorsions ("overload distortion"). Ici nous négligeons cette source d'erreur, et nous supposons que le domaine de X_q est infini ($i \in \mathbb{Z}$). Il reste alors les erreurs dues aux arrondis, qui sont décrites par le processus stochastique $q(n) = X(n) - X_q(n) = X(n) - Q(X(n))$ (figure 4(b)). Ce dernier est le *bruit de quantification* (ou plus précisément, le bruit de granularité: "granular noise"). Moyennant certaines hypothèses (fréquence d'échantillonnage du signal à temps continu pas trop élevée, pas de quantification suffisamment petit, signal analogique non périodique), on peut supposer que les erreurs entre échantillons différents sont indépendantes. Le domaine de $q(n)$ est limité à l'intervalle $[-\Delta/2, \Delta/2]$. La seconde hypothèse qu'on peut assez raisonnablement faire est que $q(n)$ prend toutes les valeurs dans $[-\Delta/2, \Delta/2]$ de manière équiprobable, et suit donc une loi uniforme dans cet intervalle. Sa moyenne est donc nulle et sa variance vaut $\sigma_q^2 = \Delta^2/12$. Par conséquent, le bruit de quantification est généralement considéré comme un bruit blanc, de fonction d'auto-corrélation

$$R_q(k) = \sigma_q^2 \delta(k) = \begin{cases} \sigma_q^2 = \frac{\Delta^2}{12} & \text{si } k = 0 \\ 0 & \text{si } k \neq 0 \end{cases}$$

La qualité du codage PCM est souvent mesurée par le rapport de la variance σ_X^2 du signal utile X sur la variance du bruit de quantification q . Ce rapport Signal sur Bruit (Signal to Noise Ratio, SNR) $SNR = \sigma_X^2 / \sigma_q^2$ et est souvent exprimé en dB.

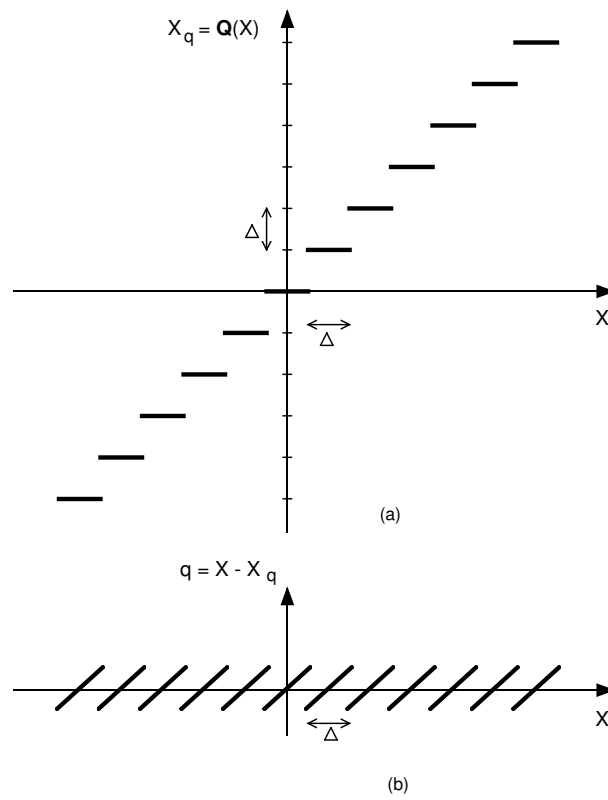
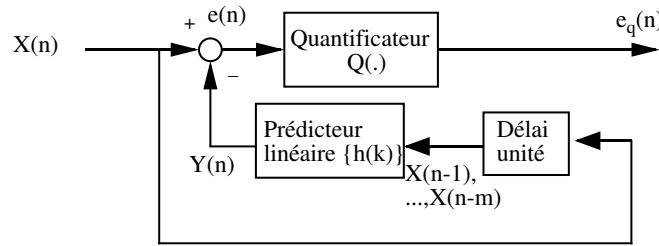


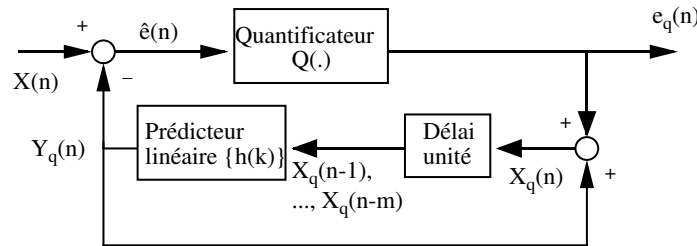
Figure 4: Fonction de quantification: $X_q = Q(X)$ est la v.a. quantifiée (a) et $q = X - X_q$ est le bruit de quantification résultant (b).

3.2 Codage DPCM

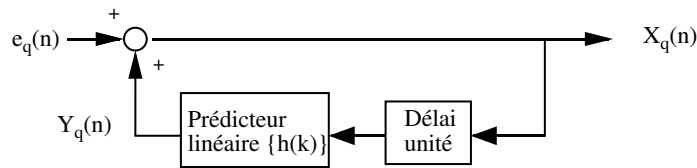
Une méthode permettant de compresser l'information consiste à exploiter la redondance entre les échantillons en calculant une estimation $Y(n)$ de chaque échantillon $X(n)$ à partir des m échantillons précédents $X(n-1), \dots, X(n-m+1)$, comme montré par le schéma de l'encodeur de la figure 5(a). L'erreur résultante $e(n) = X(n) - Y(n)$ est alors encodée sur un nombre fini de niveaux de quantification (bits), de sorte que le signal quantifié et effectivement transmis est $e_q(n) = Q(e(n)) = e(n) - q(n)$ où $q(n)$ est l'erreur de quantification. A la réception (figure 5(c)), le signal reconstruit est $X_q(n) = Y_q(n) + e_q(n)$, où $Y_q(n)$ est la valeur obtenue en soumettant le même filtre de prédiction linéaire aux entrées $X_q(n-1), \dots, X_q(n-m)$. Remarquons qu'en l'absence de quantification ($q(n) = 0$), le signal X_q est bien identique à X . L'intérêt de ce codage



(a) (Pseudo-)Codeur DPCM (prédiction à partir des échantillons non quantifiés)



(b) Codeur DPCM (prédiction à partir des échantillons quantifiés)



(c) Décodeur DPCM

Figure 5: Codeur (versions simplifiée (a) et réelle (b)) et décodeur DPCM (c).

vient du fait que la plage de valeurs prises par $e(n)$ est plus réduite que celle du signal $X(n)$ lui même. En effet, le filtre de prédiction sera conçu de manière à minimiser la variance de $e(n)$, et donc $\sigma_e^2 \ll \sigma_X^2$. Par conséquent, pour un nombre donné de niveaux de quantifications (bits), la

variance de l'erreur de quantification σ_q^2 résultant de la quantification de e sera beaucoup plus faible que dans le cas PCM, où l'erreur résultait de la quantification du signal X lui-même. De façon duale, pour un niveau de qualité donné, le nombre de niveaux de quantifications (de bits) à utiliser sera beaucoup plus faible: c'est ce qui permet la compression du signal. La figure 6 montre les résultats obtenus par codage PCM et DPCM. La compression ainsi réalisé permet d'atteindre des débits de 16 kbps. Avant de décrire le filtre de prédiction linéaire dans la section

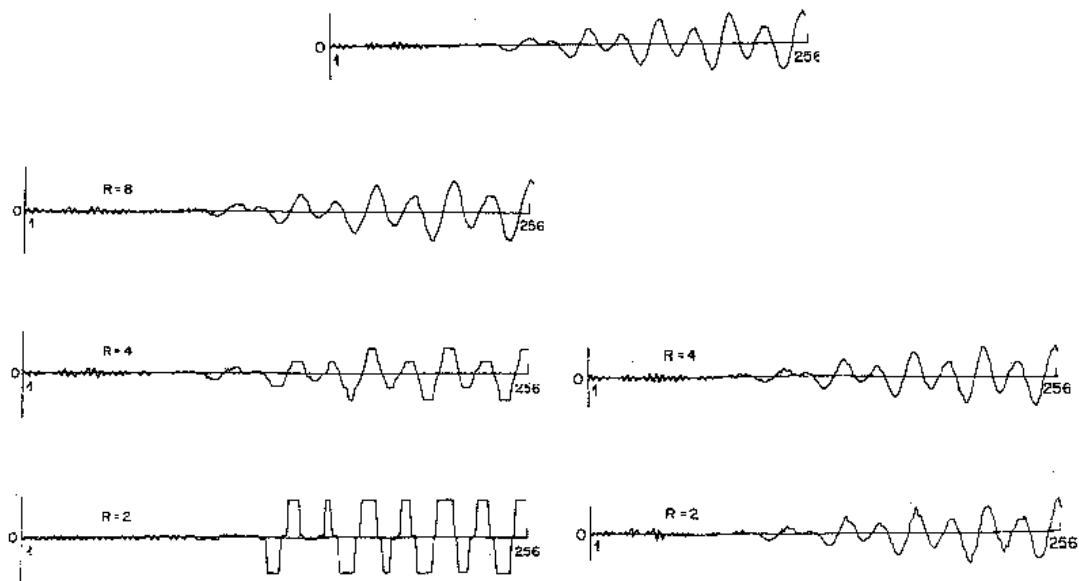


Figure 6: Codage PCM (à gauche) et DPCM (à droite) du signal de parole donné en haut. R est le nombre de bits. (d'après Jayant et Noll)

suivante, notons que la prédiction basée sur les échantillons quantifiés $X_q(n-1), \dots, X_q(n-m+1)$ (donc sur le signal de sortie) est préférée à la celle basée sur les échantillons de départ $X(n-1), \dots, X(n-m+1)$ (i.e., sur le signal d'entrée), de sorte que le prédicteur se trouve dans la boucle de rétroaction autour du quantificateur (comme à la figure 5(b)) au lieu d'être placé en aval (comme à la figure 5(a)). Les erreurs de quantification $q(n-1), \dots, q(n-m+1)$ sont dès lors prises en compte par le prédicteur pour la reconstruction du signal $X_q(n)$, ce qui évite l'accumulation d'erreurs dans ce signal.

3.3 Filtre de prédiction linéaire

On suppose que le signal $X(n)$ est un processus WSS. On veut calculer une estimation $Y(n)$ de l'échantillon $X(n)$ du signal à partir des m échantillons $X(n-1), \dots, X(n-m)$ par un filtre

linéaire

$$Y(n) = \sum_{k=1}^m h(k)X(n-k) \quad (45)$$

Il faudra alors déterminer les coefficients $h(k)$ ($1 \leq k \leq m$) de ce filtre de telle manière que l'erreur quadratique moyenne

$$\varepsilon = E[e^2(n)] = E[(X(n) - Y(n))^2]$$

soit minimisée. Les coefficients du filtre de prédiction $h(k)$ sont appelés *coefficients de prédiction*. On remarque que ce problème de prédiction est à nouveau le problème de filtrage de Wiener représenté à la figure 1, dans lequel la réponse désirée est $D(n) = X(n)$, si bien que dans ce cas

$$R_{DX}(k) = R_X(k)$$

pour $1 \leq k \leq m$. Le filtre est à réponse impulsionnelle finie car $h(k) \neq 0$ seulement si $1 \leq k \leq m$. Les équations de Wiener-Hopf (38) deviennent alors

$$\sum_{l=1}^m h(l)R_X(k-l) = R_X(k) \quad (46)$$

avec $1 \leq k \leq m$, et prennent également le nom d'*équations de Yule-Walker*. En notant

$$\begin{aligned} \mathbf{r}_X &= [R_X(1) \ R_X(2) \ \dots \ R_X(m)]^T \\ \mathbf{h} &= [h(1) \ h(2) \ \dots \ h(m)]^T \end{aligned}$$

et en utilisant (15), elles s'écrivent sous la forme matricielle

$$\Phi_X \mathbf{h} = \mathbf{r}_X \quad (47)$$

Le calcul de leur solution

$$\mathbf{h} = \Phi_X^{-1} \mathbf{r}_X \quad (48)$$

nécessite l'inversion de la matrice de corrélation, et par conséquent la mise au point d'algorithmes efficaces pour réaliser cette opération: ce sera l'objet des cours avancés de traitement du signal (applied DSP), qui utiliseront les propriétés particulières de la matrice de corrélation pour cette inversion. Ainsi, comme cette matrice est symétrique, on peut la diagonaliser par une matrice orthogonale Q (i.e., telle que $Q^{-1} = Q^T$), de sorte que

$$\Phi_X = Q^T \Lambda Q.$$

Dans ce contexte du traitement du signal, cette décomposition s'appelle la *décomposition de Karhunen-Loève*, elle consiste en fait à décorrélérer les échantillons du signal de parole $X(n)$. On peut encore écrire cette relation sous la forme spectrale

$$\Phi_X = \sum_{i=1}^m \lambda_i \mathbf{q}_i \mathbf{q}_i^T$$

où les $\mathbf{q}_i$ sont les vecteurs-colonnes de Q :

$$Q = [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_m].$$

L'inverse de Φ_X est directement calculable par

$$\Phi_X^{-1} = Q\Lambda^{-1}Q^T = \sum_{i=1}^m \lambda_i^{-1} \mathbf{q}_i \mathbf{q}_i^T$$

Néanmoins, le calcul des valeurs propres étant complexe en nombre d'opérations, cette solution élégante en théorie est peu efficace en pratique. On préférera l'algorithme de Levinson, pour le calcul rapide de l'inverse de Φ_X , qui tire parti du fait que c'est une matrice de Toeplitz.

4 Application 3: Vocodeur LPC et synthèse de la parole

Une méthode de compression plus radicale que le codage DPCM consiste à ne plus transmettre l'erreur d'estimation due au modèle linéaire de prédiction, mais seulement ces derniers, et de reconstruire le signal à partir de ceux-ci. Cela suppose que le prédicteur linéaire modélise relativement bien le vrai système de production du signal de parole. Cette méthode de codage de source permet de réduire considérablement la redondance du signal de parole (débit d'environ 4.8 kbps (kbits/sec) ou même 2.4 kbps au lieu de 16 kbps), mais au détriment de la qualité de la parole qui n'est que synthétique. Comme ce codage se base sur une modélisation du processus de synthèse de la parole, nous commençons par en exposer un modèle linéaire simple.

4.1 Synthèse de la parole

Un modèle très simplifié de production de la parole, représenté à la figure 7, consiste en un filtre numérique, décrit par ses coefficients a_k ($1 \leq k \leq m$), soumis à une excitation $u(n)$ qui dépend du type de son. Pour les sons *voisés* ou sonores (comme /a/, /i/, /v/), qui résultent d'impulsions périodiques provoquées par les oscillations des cordes vocales, cette excitation sera un train périodique d'impulsions d'amplitude σ . La période de ce train d'impulsions est appelée le *pitch*. Pour les sons *non voisés* (comme /f/, /s/, /t/), qui ne présentent pas de structure périodique, cette excitation est un bruit blanc de moyenne nulle et de variance σ^2 . La figure 8 montre différentes formes d'onde du signal de parole. Si la durée sur laquelle les échantillons du signal de parole $X(n)$ sont pris est suffisamment petite (Durée N_0 de 10 à 30 msec), le signal peut être considéré comme stationnaire. L'idée du codage de source basé sur ce modèle est de ne pas transmettre les échantillons $X(n)$, mais seulement les paramètres du modèle qui peut synthétiser ce signal $X(n)$ pendant une période de durée N_0 . Comme $m \ll N_0$, il en résulterait une compression considérable de l'information. En pratique cependant, on ne dispose pas des valeurs a_k ni de l'excitation $U(n)$. Il faut donc estimer à partir des seuls échantillons $X(n)$ du signal de parole les coefficients a_k .

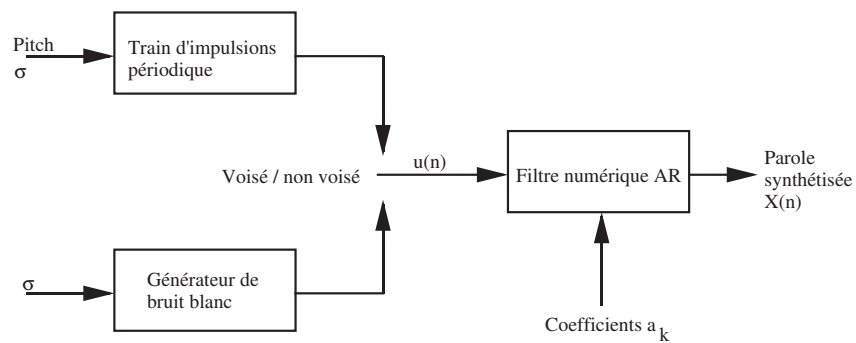


Figure 7: Modèle simplifié de synthèse de la parole.

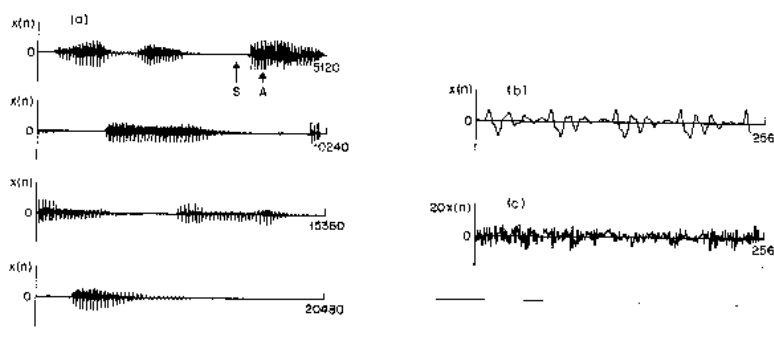


Figure 8: Formes d'onde d'un signal de parole pour une longueur d'une phrase (a) et d'un son voisé (b) et non voisé (c). (d'après Jayant et Noll)

4.2 Vocodeurs

Le filtre de prédiction linéaire développé à la section 3.3 est utilisé pour coder le signal de parole selon la technique de codage de source: on parle d'un *vocodeur* (pour "voice coder"), dont un schéma de principe est donné à la figure 9. L'ordre du filtre est le même que l'ordre du modèle AR du système de production de la parole ($m = p$, nous ne décrivons pas ici la méthode pour déterminer ce paramètre), les coefficients de prédiction $h(k)$ sont des estimées des coefficients $-a_k$, $1 \leq k \leq p$. L'émetteur prend des blocs de N_0 échantillons $X(n)$ au travers d'une fenêtre (typiquement d'une longueur de 10 à 30 msec) qui sont suffisamment courts pour être considérés comme stationnaires. Chaque bloc est alors analysé pour en déterminer les caractéristiques suivantes:

1. son voisé ou non voisé,
2. le pitch,
3. le gain σ ,
4. les coefficients de prédiction $h(k)$.

Ces quatre (groupes de) paramètres sont codés et transmis sur le canal de transmission. A l'arrivée, le récepteur fait le décodage inverse, et utilise ces paramètres dans un synthétiseur de parole du type de celui de la figure 7 pour retrouver le signal de parole. La figure 10 montre deux signaux de parole et leur synthèse par un vocodeur. La compression d'information ainsi réalisée est fort importante, mais la qualité du codage s'en ressent. Pour remédier à cet inconvénient, on peut utiliser la technique de codage de la forme d'onde, qui aboutit néanmoins alors à des débits plus élevés.

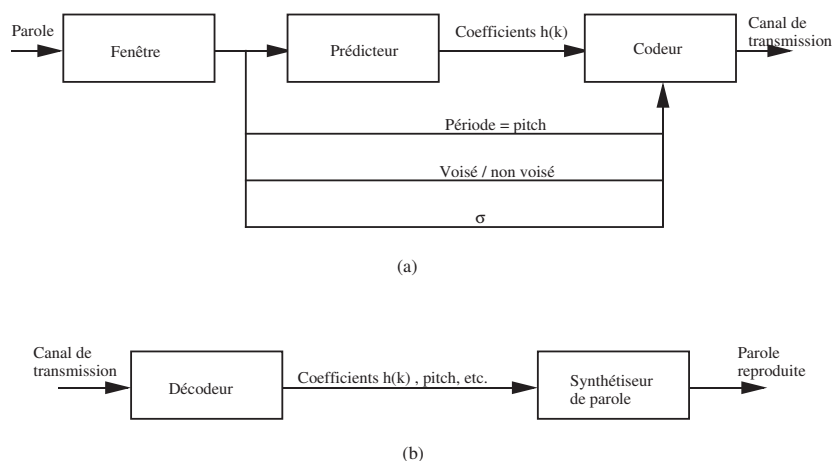


Figure 9: Diagramme bloc d'un vocodeur: émetteur (a) et récepteur (b).

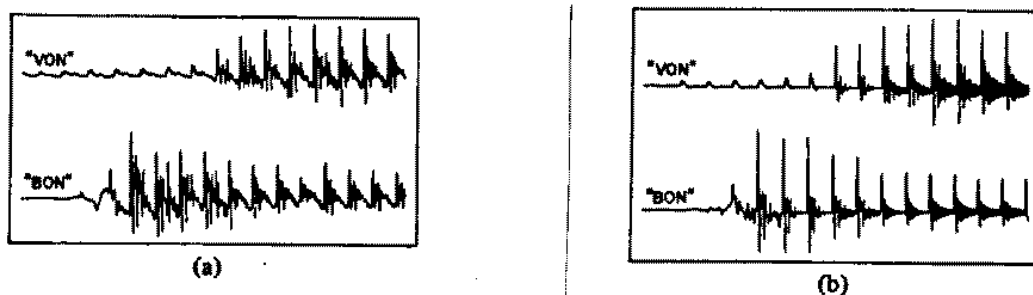


Figure 10: Signal de parole réel (a) et synthèse par vocodeur (b) (d'après Jayant et Noll).

5 Exercices

1. On considère un processus WSS $\{X(n), n \in \mathbb{N}\}$, dont la moyenne est nulle et dont la fonction d'auto-corrélation est

$$R_X(k) = \sigma^2 \alpha^{|k|}$$

avec $|\alpha| < 1$.

- (a) Montrer que ce processus est un processus AR d'ordre 1. Que vaut le coefficient a_1 dans (13) ? Quelle est la variance du bruit blanc $U(n)$?
 - (b) Montrer que ce processus peut être aussi obtenu comme un processus MA d'ordre ∞ . Que valent les coefficients b_k dans (12) ? (Hint: développer $(1 - \alpha z)^{-1}$ en série....)
 - (c) Ce processus est-il ergodique par rapport à sa moyenne ?
2. On considère un processus à moyenne mobile (MA) d'ordre 1

$$Y(n) = X(n) + \beta X(n-1)$$

où $X(n)$ est le processus de l'exercice 1, de moyenne nulle et de fonction d'autocorrélation $R_X(k) = \sigma^2 \alpha^{|k|}$ avec $|\alpha| < 1$.

- (a) Calculer $S_Y(f)$.
 - (b) Pour quelle valeur de β $Y(n)$ est-il un bruit blanc ?
3. Calculer la fonction d'auto-corrélation du processus MA d'ordre m donné par (12).
 4. Soit la suite $X(n), X(n-1), \dots, X(n-m+1)$ formée de m échantillons d'un processus Gaussien discret, stationnaire et de moyenne nulle. On se propose d'étudier les propriétés de deux estimateurs de la fonction d'auto-corrélation du processus $R_X(k)$, à savoir

$$\hat{R}'_X(k) = \begin{cases} \frac{1}{m-|k|} \sum_{n=|k|+1}^m X(n)X(n-k) & \text{pour } |k| < m \\ 0 & \text{pour } |k| \geq m \end{cases} \quad (49)$$

et

$$\hat{R}''_X(k) = \begin{cases} \frac{1}{m} \sum_{n=|k|+1}^m X(n)X(n-k) & \text{pour } |k| < m \\ 0 & \text{pour } |k| \geq m \end{cases} \quad (50)$$

- (a) Ces estimateurs sont dits non biaisés lorsque $E[\hat{R}_X(k)] = R_X(k)$ pour $|k| < m$. Montrer que l'un des deux estimateurs est biaisé au contraire de l'autre.
- (b) Montrer que la variance du premier estimateur (49) est

$$VAR[\hat{R}'_X(k)] = \frac{1}{(m-|k|)^2} \sum_{l=-m+k}^{m-k} (m-k-|l|)(R_X^2(l) + R_X(l+k)R_X(l-k))$$

tandis que celle du second estimateur (50) est

$$VAR[\hat{R}''_X(k)] = \frac{1}{m^2} \sum_{l=-m+k}^{m-k} (m-k-|l|)(R_X^2(l) + R_X(l+k)R_X(l-k)).$$

- (c) Comparer les qualités et les défauts des deux estimateurs (Biais, variance, erreur quadratique moyenne $\varepsilon = E[(\hat{R}_X(k) - R_X(k))^2]$). En particulier, que se passe-t-il lorsque k est variable mais que le nombre d'échantillons m est fixé ? Et lorsque la valeur de m devient très grande ?

5. On étudie la stationnarité des trois processus stochastiques à temps discret

$$\begin{aligned} X_1(n) &= A \sin(2\pi f n + \varphi) & n \geq 1 \\ X_2(n) &= a \sin(2\pi f n + \Phi) & n \geq 1 \\ X_3(n) &= a \sin(2\pi F n + \varphi) & n \geq 1 \end{aligned}$$

où l'amplitude, la fréquence et la phase sont respectivement, une lettre minuscule désignant un réel fixé et une lettre majuscule désignant une variable aléatoire.

Processus	Amplitude	Fréquence	Phase
$\{X_1(n), n \in \mathbb{N}_0\}$	v.a. unif. dans $[-1, 1]$	constante	constante
$\{X_2(n), n \in \mathbb{N}_0\}$	constante	constante	v.a. unif. dans $[0, 2\pi]$
$\{X_3(n), n \in \mathbb{N}_0\}$	constante	v.a. unif. dans $[0, 1]$	constante

- (a) Le(s)quel(s) de ces trois processus est-il (sont-ils) stationnaire(s) au sens large ?
- (b) Calculer les densités de probabilité $f_{X_i(n)}(x; n)$ de chaque v.a. $X_i(n)$, pour $i = 1, 2, 3$ et $n \in \mathbb{N}_0$.
- (c) Calculer $E[X_i^2(n)X_i(m)]$ pour $i = 1, 2, 3$ et $n, m \in \mathbb{N}_0$. (Ce moment d'ordre 3 peut vous être utile pour répondre à la sous-question d)
- (d) Le(s)quel(s) de ces trois processus est-il (sont-ils) stationnaire(s) au sens strict ?
6. Si le signal à l'entrée d'un filtre de Wiener $X(n)$ est un bruit blanc de variance σ_X^2 , quelle est la solution $\{h(k)\}$ des équations de Wiener-Hopf, dans le cas causal, en fonction de $R_{DX}(k)$?

7. Un filtre de Wiener est utilisé pour filtrer un bruit blanc additif $N(n)$, de moyenne nulle et de variance σ_N^2 , corrompant un signal $S(n)$, de moyenne nulle et de fonction d'autocorrélation $R_S(k) = \sigma_S^2 \alpha^{|k|}$, avec $|\alpha| < 1$. Le signal à l'entrée du filtre est donc $X(n) = S(n) + N(n)$. Les deux processus $N(n)$ et $S(n)$ sont indépendants. Quelle est la transmittance $H(f)$ du filtre de Wiener non causal résolvant ce problème ?
8. Montrer que les deux processus $e(n)$ et $X(n)$ définis à la section 3.3 consacrée au filtre de Wiener sont orthogonaux.
9. Soit $\{S(n), n \in \mathbb{Z}\}$ un processus stationnaire au sens faible. On définit le processus $\{X(n), n \in \mathbb{Z}\}$ suivant:

$$X(n) = S(n) + S(n-1), \quad n \in \mathbb{Z}.$$

On désire estimer le signal $D(n) = S(n)$ à partir de l'observation du processus X au moyen du filtre de Wiener

$$Y(n) = \sum_{k \in \mathbb{Z}} h(k) X(n-k), \quad n \in \mathbb{Z}.$$

- (a) Quel est le filtre optimal de Wiener $\{h(k), k \in \mathbb{Z}\}$ qui minimise l'erreur quadratique moyenne

$$\varepsilon = E(e(n)^2), \quad \text{où } e(n) = S(n) - Y(n)?$$

Indication: pour résoudre le problème, utiliser la transformée en z : $H(z) = \sum_{k \in \mathbb{Z}} h(k) z^{-k}$, plutôt que la transformée de Fourier discrète; ça simplifie les notations et donc les calculs!

- (b) Le filtre optimal dépend-il du processus S ?
- (c) Quelle est l'erreur commise ε ? Peut-on dire quelque chose de plus sur $e(n)$ ici?
- (d) Le filtre optimal est-il causal?