

## Reconstruction en temps réel de signaux à échantillonnage non périodique.

---

### *Real-Time Reconstruction of Non Uniformly Sampled Signals.*

par Othmane BENSAOUD, Jacques OKSMAN

École Supérieure d'Électricité – Service des Mesures  
Plateau de Moulon  
F-91192 Gif-sur-Yvette cedex

#### Résumé

Nous proposons dans cet article des méthodes de traitement statistique ou d'analyse numérique pour reconstituer en temps réel un signal échantillonné non uniformément. Notre approche est essentiellement basée sur une modélisation du signal à court terme suivie d'une prédiction utilisant le modèle obtenu.

Cet article comprend trois parties. Une première concerne l'utilisation d'un filtre de Kalman pour une reconstitution à partir d'un modèle donné. La deuxième partie correspond à des méthodes d'analyse numérique, nous y présentons un algorithme de prédiction-corrrection. Enfin, la troisième partie présente deux modélisations, l'une par un développement de Taylor et l'autre par les paramètres AR du signal.

**Mots clés :** échantillonnage non uniforme, prédiction, lissage, prédiction-corrrection, temps réel.

#### Abstract

*The purpose of this paper is the presentation of methods of signal processing or numerical analysis for the real-time reconstruction of non uniformly sampled signal. Our approach is essentially based on the short term modelling of a signal and its prediction using the obtained model.*

*This paper includes three parts. In the first one we use a Kalman filter with a given model. The second part corresponds to numerical analysis methods; we present a prediction-correction algorithm. In the last part, two examples of modelisation are presented. The first one uses Taylor development and the second one uses an AR model.*

**Key words :** nonuniform sampling, prediction, smoothing, prediction-corrrection, real-time.

## 1. Introduction

### 1.1. CADRE « UTILITAIRE »

La plupart des signaux sont aujourd'hui représentés par des séries temporelles obtenues par discrétisation périodique du temps. Les avantages de ce type de représentation ne sont plus à vanter : seule la dimension « amplitude » importe (le temps est uniquement représenté par un indice, entier incrémenté à chaque période), les matériels de traitement (typiquement des systèmes séquentiels synchrones à horloge périodique) sont relativement simples et, de plus, la quasi totalité des outils théoriques qui existent dans le domaine du traitement numérique du signal sont basés sur le principe d'un échantillonnage périodique.

À l'exception de certains travaux de Cauchy et d'autres mathématiciens, le premier élément qui a conduit historiquement à l'étude des signaux à échantillonnage non périodique (en abrégé

S.E.N.P.) a été la nécessité de tenir compte des effets de gigue (« jitter ») de l'horloge dans le cas des oscilloscopes « à échantillonnage ». En effet, le sous-échantillonnage qu'effectuent ces appareils (puisque la période d'échantillonnage est supérieure à celle du signal) exacerbe l'influence de la gigue (qui doit être comparée à la période du signal et non plus à celle de l'horloge).

Dans les techniques actuelles, il est cependant de nombreux cas dans lesquels apparaissent des exemples de S.E.N.P. Par exemple, on sait que, pour un pas de quantification  $q$  donné, il n'est pas utile d'avoir une période d'échantillonnage plus courte que  $\frac{q}{V}$ ,  $V$  étant la vitesse maximale du signal.

Il est alors évident que la connaissance des instants  $t_k$  de franchissement d'un incrément  $q$  et du sens de variation du signal est suffisante pour connaître toutes ces valeurs avec une incertitude  $q$ . Ce procédé est ainsi utilisé dans le cadre de la modulation  $\Delta$ .

Dans le domaine du contrôle/commande des procédés industriels, on rencontre aussi, de plus en plus fréquemment, des réseaux de capteurs/actionneurs dotés d'une capacité de traitement locale.

Les S.E.N.P. peuvent alors apparaître dans plusieurs cas. Le premier est celui des réseaux asynchrones, c'est-à-dire de la plupart des réseaux « classiques » (tels que ceux que l'on trouve dans les applications informatiques). Les techniques d'accès CSMA/CD, utilisées dans le populaire réseau ETHERNET, supposent l'existence de collisions et donc de délais aléatoires sur l'émission. Les réseaux de « terrain » n'utilisent donc pas ces solutions pourtant éprouvées, au profit de solutions dans lesquelles le temps est centralisé et distribué par un gestionnaire (ordinateur de procédé), au prix du transfert d'informations inutiles que sont les interrogations périodiques des capteurs. D'un autre côté, la périodicité de ces interrogations impose que les temps de réponse des capteurs soient limités au maximum. Ceci est incompatible avec la notion de « capteur intelligent », qui est censé effectuer un traitement complexe dont la durée ne peut être supposée constante. Il est en effet évident que les instants d'acquisition des grandeurs physiques et de transfert vers le gestionnaire d'un réseau sont toujours distincts, que le capteur effectue son traitement après ou avant la demande périodique. Même dans le cas d'interrogations périodiques, il y a alors apparition d'un S.E.N.P., sauf si l'on admet un retard pur égal à une période d'échantillonnage.

Si l'on considère maintenant un cas « idéal » purement périodique, on va encore rencontrer des S.E.N.P., dans tous les cas où il y a disparition des échantillons, soit par suite d'erreurs de transmission, soit par suite d'une indisponibilité temporaire de la mesure (ex. : les « trous » dans un signal Doppler, disparitions brusques du signal liées à des variations de la rétrodiffusion).

On peut aussi imaginer que l'on crée volontairement des signaux à échantillonnage non périodique, soit pour éviter des synchronismes entre l'horloge d'échantillonnage et le signal (cas des mesures de puissance, par exemple) soit que l'on pense à des techniques de compression de données basées sur le principe suivant : un capteur n'émettra un échantillon que lorsque ce sera « nécessaire » (par exemple au sens de l'erreur maximale que commettrait un algorithme de reconstitution exécuté par le récepteur).

En résumé, le traitement des S.E.N.P. apparaît utile, et il est donc nécessaire de développer les outils correspondants, en particulier pour en effectuer une simple reconstitution, si possible en temps réel, afin de recouvrir le domaine général des mesures et de la commande industrielle [1][2][3][4].

## 1.2. PRÉSENTATION

On est amené à considérer deux cas bien précis d'apparition des signaux à échantillonnage non périodique, auxquels correspondront deux types de traitements.

Dans le premier cas, les instants d'échantillonnage sont quelconques, les écarts entre échantillons peuvent prendre leur valeurs dans un ensemble continu. Dans le deuxième, en revanche, il existe une « période d'échantillonnage cachée » et les temps correspondant aux instants d'échantillonnage ne peuvent prendre que

des valeurs quantifiées (fig. 1.1). Nous travaillerons dans le cadre de ce dernier cas qui correspond, par exemple, à des signaux subissant des pertes dues à des erreurs de transmission, tandis qu'un exemple de signal à temps continu pourrait être celui d'une modulation  $\Delta$ .

On suppose donc que les intervalles de temps  $d_n$  ( $d_n = t_n - t_{n-1}$ ) sont des multiples entiers de  $h$  et qu'ils sont bornés. Nous avons alors :  $d_n = k_n h$ ,  $k_n - 1$  étant le nombre de mesures manquantes entre deux instants d'échantillonnage.

Nous ferons également l'hypothèse que le bruit de mesure est nul, ce qui est réaliste dans le domaine de l'instrumentation.

Dans le but de reconstruire un signal échantillonné avec un pas non uniforme, on se propose d'utiliser des algorithmes qui, pour les prédicteurs, connaissant le signal et son modèle jusqu'à l'instant  $t_n$ , permettent de prédire celui-ci aux instants  $t_n + \alpha h$ , (avec  $t_n$  : instants d'échantillonnage,  $\alpha$  : entier naturel,  $h$  : période d'échantillonnage cachée (fig. 1.1)), et qui, pour les algorithmes de lissage, connaissant le signal et son modèle jusqu'à l'instant  $t_n$  permettent de le lisser jusqu'à l'instant d'échantillonnage précédent.

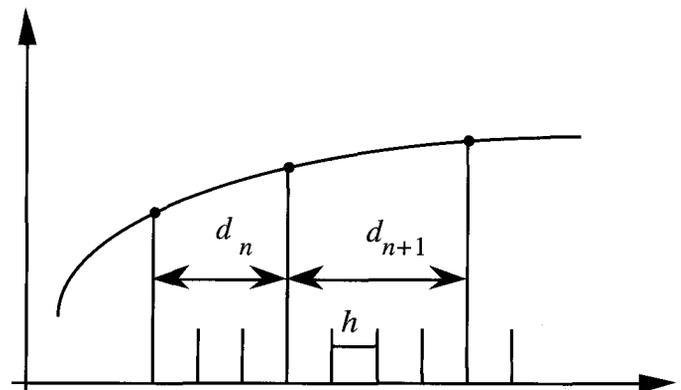


Figure 1.1.

Nous présentons dans ce qui suit deux algorithmes permettant la reconstitution du signal en utilisant les notions de prédiction et de lissage. Le premier algorithme, que nous appelons prédicteur-lisseur de Kalman est un algorithme qui utilise un filtre de Kalman, d'abord en tant que prédicteur et ensuite en tant que lisseur. Le deuxième algorithme, que nous appelons prédicteur correcteur, est utilisé en analyse numérique dans le cadre de la résolution des équations différentielles. Le filtre de Kalman est ensuite utilisé pour lisser le signal reconstruit.

## 2. Techniques fondées sur le filtre de Kalman

Dans cette technique, l'idée de départ est d'utiliser un filtre de Kalman en tant que prédicteur. En effet, puisque nous ne disposons pas de mesures entre deux instants d'échantillonnage,

nous ne pouvons faire que des prédictions de l'état en utilisant le modèle du signal. L'arrivée de la mesure à l'instant d'échantillonnage nous permet d'améliorer la dernière prédiction mais pas les précédentes. Nous avons pensé à améliorer ces prédictions en utilisant un lissage par un filtre de Kalman. C'est donc l'association de ces deux idées que nous appelons algorithme prédictif-lisseur de Kalman.

Nous représentons le signal dans l'espace d'état par les équations suivantes

$$\begin{aligned} \mathbf{x}_{t_n+1} &= A_{t_n} \mathbf{x}_{t_n} + \mathbf{v}_{t_n} \\ y_{t_n} &= H_{t_n} \mathbf{x}_{t_n} \end{aligned}$$

$\mathbf{x}_{t_n}$  représente le vecteur d'état à l'instant  $t_n$ ,  $A_{t_n}$  et  $H_{t_n}$  représentent respectivement la matrice de transition et la matrice des observations au même instant. Nous présenterons par la suite des modélisations du signal nous permettant d'identifier les éléments de ces matrices.  $\mathbf{v}_{t_n}$  est un vecteur qui représente le bruit d'état, autrement dit  $\mathbf{v}_{t_n}$  représente les erreurs de modélisation du signal. Ce bruit est supposé indépendant et de moyenne nulle.

À partir de l'équation d'état et des conditions initiales, nous allons essayer de reconstruire les parties manquantes du signal reçu. Le fait de ne pas avoir de mesures pendant un certain temps ne permet pas au filtre de Kalman [5] d'améliorer la matrice de covariance de l'erreur de prédiction a priori, qui aura tendance à prendre de grandes valeurs dès que l'intervalle de temps  $d_n$  deviendra grand. De ce fait apparaît le problème de la « crédibilité » à accorder aux valeurs prédites. En particulier, la détermination de la valeur maximale des intervalles  $d_n$  qui permet une bonne reconstitution du signal apparaît difficile à obtenir, si ce n'est par simulation.

Le filtre de Kalman choisi utilise, comme on va le voir, les anciennes prédictions pour en construire une autre; plus l'on fait de prédictions plus l'on s'écarte du modèle du signal et plus l'erreur de prédiction augmente. Nous modéliserons ce phénomène en considérant que la matrice de covariance du bruit d'état, notée  $Q_{t_n}$  augmente linéairement avec le temps. Nous prendrons dans ce cas :

$$Q_{t_n+i} = E[\mathbf{v}_{t_n+i} \mathbf{v}_{t_n+i}^T] = \lambda i Q_0$$

$Q_0$  dépend du modèle,  $i$  étant l'indice temporel (en nombre de périodes d'échantillonnage cachées). Le scalaire  $\lambda$  et la matrice  $Q_0$  représentent respectivement le facteur de pondération servant à limiter l'augmentation de la variance du bruit d'état et la matrice de covariance initiale du bruit d'état.

## 2.1. ALGORITHME DE PRÉDICTION

Supposons qu'à l'instant  $t_n$  on connaisse  $\hat{\mathbf{x}}_{t_n}$  et  $\hat{P}_{t_n}$  respectivement estimation du vecteur d'état à l'instant  $t_n$  connaissant le signal aux instants précédents et estimation de la matrice de covariance de l'erreur de prédiction a priori. On veut estimer l'état jusqu'à l'instant  $t_{n+1}$  alors qu'on ne dispose d'aucune mesure entre les instants  $t_n$  et  $t_{n+1}$ . On commence par faire une prédiction

de l'état à l'instant  $t_n + 1$  :

$$\hat{\mathbf{x}}_{t_n+1|t_n} = A_{t_n} \hat{\mathbf{x}}_{t_n}$$

La matrice de covariance de l'erreur de prédiction est donnée par :

$$\hat{P}_{t_n+1,t_n} = A_{t_n} \hat{P}_{t_n} A_{t_n}^T + Q_{t_n}$$

Puisque nous ne disposons pas de mesure à l'instant  $t_n + 1$ , nous ne pouvons pas faire d'estimation de l'état à cet instant et par conséquent nous ne pouvons pas améliorer  $\hat{P}_{t_n+1}$ .

La prédiction de l'état à l'instant  $t_n + 2$  est donnée par :

$$\hat{\mathbf{x}}_{t_n+2|t_n+1} = A_{t_n+1} \hat{\mathbf{x}}_{t_n+1}$$

De même, la matrice de covariance de l'erreur de prédiction à l'instant  $t_n + 2$  est donnée par :

$$\hat{P}_{t_n+2,t_n+1} = A_{t_n+1} \hat{P}_{t_n+1} A_{t_n+1}^T + Q_{t_n+1}$$

Or le calcul, utilisant les équations de Kalman, de la prédiction de l'état à l'instant  $t_n + 2$  (c'est-à-dire  $\hat{\mathbf{x}}_{t_n+2|t_n+1}$ ) nécessite la connaissance de  $\hat{\mathbf{x}}_{t_n+1}$  qui à son tour nécessite la connaissance de l'écart entre la mesure prédite et la mesure effective; or cette dernière n'est disponible qu'à chaque instant d'échantillonnage  $t_n$ . L'idée alors est de prendre :

$$\hat{\mathbf{x}}_{t_n+1} = \hat{\mathbf{x}}_{t_n+1|t_n}$$

De même le calcul de  $\hat{P}_{t_n+2,t_n+1}$ , nécessite la connaissance de  $\hat{P}_{t_n+1}$ , qui est calculée en considérant que l'on a :

$$\hat{P}_{t_n+1} = \hat{P}_{t_n+1,t_n}$$

Le passage de la prédiction à l'estimation permet d'améliorer la prédiction et par suite de diminuer la matrice de covariance de l'erreur de prédiction a priori, mais, comme nous ne faisons pas d'estimation, nous égalisons la matrice de covariance de l'erreur de prédiction a priori et la matrice de covariance de l'erreur d'estimation pour tenir compte de l'erreur faite sur l'estimation du vecteur d'état. Cette considération dépend du modèle du signal et du modèle des écarts entre échantillons.

La prédiction de ce vecteur se fait jusqu'à l'instant  $t_{n+1} - 1$ . À l'instant suivant, c'est-à-dire  $t_{n+1}$  la mesure effective est disponible, le vecteur d'état est alors estimé en utilisant les équations suivantes, non optimisées en terme de stabilité numérique (en particulier puisque les erreurs de mesure sont négligées), mais qui s'avèrent expérimentalement satisfaisantes, y compris à long terme :

$$\begin{aligned} \hat{\mathbf{x}}_{t_n+1} &= \hat{\mathbf{x}}_{t_n+1|t_n+k_n-1} + \mathbf{K}_{t_n+1} (y_{t_n+1} - H_{t_n+1} \hat{\mathbf{x}}_{t_n+1|t_n+k_n-1}) \\ \mathbf{K}_{t_n+1} &= \hat{P}_{t_n+1,t_n+k_n-1} H_{t_n+1}^T (H_{t_n+1} \hat{P}_{t_n+1,t_n+k_n-1} H_{t_n+1}^T)^{-1} \end{aligned}$$

## 2.2. ALGORITHME DE PRÉDICTION-LISSAGE

Dans un souci d'améliorer les valeurs données par les prédictions du vecteur d'état, nous avons tout simplement pensé les lisser, ce qui améliore considérablement les performances de la reconstitution.

### 2.2.1. Notations

On note ici par  $\hat{x}_{t|t_{n+1}}$ , le vecteur d'état lissé à l'instant  $t$ , et par  $\hat{x}_t$  le vecteur d'état prédit à l'instant  $t$ , avec  $t$  variant de  $t_{n+1}$  à  $t_n$  avec un pas égal à  $-h$  (nous étendons ici cette notation au cas de l'échantillonnage non périodique).

### 2.2.2. Lissage

À partir de l'algorithme de prédiction, nous pouvons calculer l'estimée de l'état à l'instant où la mesure arrive, c'est-à-dire à l'instant  $t_{n+1}$ . Cette estimée va nous permettre d'avoir une valeur lissée de l'état à l'instant  $t_{n+1} - 1$  en utilisant l'équation itérative de Kalman [5] :

$$\hat{x}_{t|t_{n+1}} = \hat{x}_t + C_t(\hat{x}_{t+1|t_{n+1}} - A_{t_{n+1}} \hat{x}_t)$$

$$C_t = \hat{P}_{t,t} A_{t_{n+1}}^T (\hat{P}_{t,t} A_{t_{n+1}}^T + Q_t)^{-1}$$

L'algorithme de lissage se résume en une équation récurrente selon les indices décroissants dans laquelle  $\hat{x}_{t|t_{n+1}}$  est calculé à partir de  $\hat{x}_{t+1|t_{n+1}}$  et des conditions initiales  $\hat{x}_{t_{n+1}|t_{n+1}}$ , qui représentent la « mesure » à l'instant d'échantillonnage  $t_{n+1}$ .  $C_t$  représente le gain de Kalman dans le cadre du lissage. Concrètement, on profite de la connaissance du nouvel échantillon obtenu à l'instant  $t_{n+1}$  pour améliorer l'estimation entre  $t_n$  et  $t_{n+1}$ .

### 2.3. REMARQUES

- En pratique, l'arrivée d'une mesure permet de prédire le signal en utilisant l'algorithme de prédiction de Kalman et ce jusqu'à l'arrivée d'une nouvelle mesure, qui permet alors de lisser le signal prédit en utilisant l'algorithme de lissage de Kalman. La mise en œuvre de la méthode est très simple, ses performances dépendent essentiellement de la qualité de la modélisation du signal. L'algorithme prédicteur-lisseur de Kalman introduit dans le traitement un retard dont la valeur est cependant toujours inférieure à  $(d_n)_{max}$ , intervalle maximal entre échantillons. Les courbes (cf. 5.2) montrent un exemple de signal ainsi reconstitué.
- L'étude théorique de la convergence de cet algorithme est très complexe puisque dans le cas où le nombre de mesures qui manquent est supérieur à un, le nombre de cas à considérer devient très important. Il ne nous a pas paru possible d'étudier analytiquement cette convergence. C. CHEN [6] a démontré la convergence d'un algorithme de Kalman dans le cas où une seule mesure manque.

## 3. Reconstitution par prédiction-corrrection

Les algorithmes de prédiction-corrrection sont des algorithmes à pas multiples et sont très utilisés en analyse numérique [7] [8] [9]. Toute combinaison d'algorithmes de prédiction et correction peut être mise sous une forme canonique matricielle, que nous présenterons en termes d'équations d'état. Cette représentation présente les avantages suivants :

- elle est complètement générale et par conséquent permet de formuler une théorie unifiée des algorithmes à pas multiples,
- elle est très adaptée à la programmation,
- elle permet sans difficulté de changer la taille du pas de traitement et l'ordre du modèle.

### 3.1. ALGORITHME PRÉDICTEUR-CORRECTEUR

Pour reconstituer le signal nous choisissons une approche polynomiale, assimilant la courbe entre deux échantillons à un polynôme de degré  $k$ . Pour cela nous utilisons un prédicteur-correcteur de forme générale suivante :

$$x_{t_{n+1}} = \sum_{i=1}^p a_i x_{t_n-i+1} + h \sum_{i=0}^p b_i f(x_{t_n-i+1}, t_n - i + 1) \quad (1)$$

avec :

$$f(x_t, t) = x_t^{(1)}$$

On notera par  $x_{t_{n+1}}^{[j]}$  respectivement  $x_{t_{n+1}}^{(m)}$  la  $j^{\text{ème}}$  correction et la dérivée  $m^{\text{ème}}$  de  $x_{t_{n+1}}$ .

Si  $b_0 = 0$ , l'équation (1) devient l'équation d'un prédicteur que l'on met sous la forme vectorielle suivante :

$$x_{t_{n+1}}^{[0]} = \mathbf{a}x_{t_n} + h\mathbf{b}f_{t_n} \quad (2)$$

avec :

$$\mathbf{a} = [a_1 \quad a_2 \quad \dots \quad a_p]$$

$$\mathbf{b} = [b_1 \quad b_2 \quad \dots \quad b_p]$$

$$x_{t_n} \begin{bmatrix} x_{t_n} \\ x_{t_n-1} \\ \vdots \\ x_{t_n-p} \end{bmatrix} \quad f_{t_n} = \begin{bmatrix} f(x_{t_n}, t_n) \\ f(x_{t_n-1}, t_n - 1) \\ \vdots \\ f(x_{t_n-p}, t_n - p) \end{bmatrix}$$

Si  $b_0 \neq 0$ , l'équation (1) devient l'équation d'un correcteur que l'on écrit comme précédemment :

$$x_{t_{n+1}}^{[j+1]} = \mathbf{a}'x_{t_n} + h\mathbf{b}'x_{t_n}^{(1)} + hb_0f(x_{t_{n+1}}^{[j]}, t_n + 1) \quad (3)$$

avec :

$$\mathbf{a}' = [a'_1 \quad a'_2 \quad \dots \quad a'_p]$$

$$\mathbf{b}' = [b'_1 \quad b'_2 \quad \dots \quad b'_p]$$

$$\mathbf{x}_{t_n}^{(1)} = \begin{bmatrix} x_{t_n}^{(1)} \\ x_{t_n-1}^{(1)} \\ \vdots \\ x_{t_n-p}^{(1)} \end{bmatrix}$$

Par analogie avec les méthodes d'intégration des équations différentielles, nous choisissons d'utiliser une prédiction à partir de l'équation (2), puis de l'améliorer (« correction ») en utilisant l'équation (3). L'expérience montre qu'il est possible de ne faire qu'une seule correction, on aura donc :

$$\hat{x}_{t_n+1} = x_{t_n+1}^{[1]} = x_{t_n+1}^{[0]} - b_0 \left[ \lambda x_{t_n} + h \delta x_{t_n}^{(1)} - h f(x_{t_n+1}^{[0]}, t_n + 1) \right]$$

avec :

$$\lambda = [\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_p]$$

$$\delta = [\delta_1 \quad \delta_2 \quad \dots \quad \delta_p]$$

et

$$\lambda_i = \frac{a_i - a'_i}{b_0} \quad \delta_i = \frac{b_i - b'_i}{b_0}$$

L'étude de la stabilité de ces prédicteurs linéaires [10], montre que ceux ci deviennent instables dès que leur ordre devient supérieur ou égal à cinq, ce qui nous amène à considérer des ordres faibles et donc des traitements locaux. Nous avons choisi le prédicteur d'Adams-Bashforth [7] d'ordre trois dont les coefficients sont déterminés de sorte que la courbe qui passe entre deux échantillons soit un polynôme de degré trois :

$$x_{t_n+1}^{[0]} = x_{t_n} + h \left[ \frac{23}{12} f(x_{t_n}, t_n) - \frac{16}{12} f(x_{t_n-1}, t_n - 1) + \frac{5}{12} f(x_{t_n-2}, t_n - 2) \right]$$

Le correcteur que nous utilisons est celui d'Adams-Moulton d'ordre quatre :

$$x_{t_n+1}^{[1]} = x_{t_n} + h \left[ \frac{9}{24} f(x_{t_n+1}^{[0]}, t_n + 1) + \frac{19}{24} f(x_{t_n}, t_n) - \frac{5}{24} f(x_{t_n-1}, t_n - 1) + \frac{1}{24} f(x_{t_n-2}, t_n - 2) \right]$$

L'utilisation d'un développement de Taylor (cf. 4.2) permet d'écrire, en faisant intervenir les coefficients binomiaux :

$$x_{t_n} = \sum_{i=1}^p C_p^i (-1)^{i+1} x_{t_n-i} \quad (4)$$

Comme on va le voir, l'application du filtre de Kalman permettra d'une part de lisser les informations prédites et corrigées, d'autre part, d'estimer  $x_{t_n-i}^{(1)}$ , puis  $x_{t_n}^{(1)}$ , en utilisant l'équation (4), et par suite d'avoir une approximation de  $f(x_{t_n}, t_n)$ . On approchera de plus  $f(x_{t_n+1}^{[0]}, t_n + 1)$  par  $\frac{x_{t_n+1}^{[0]} - x_{t_n}^{[0]}}{h}$ .

On constate que le prédicteur et le correcteur ne font intervenir que l'information  $x_{t_n}$  à l'instant  $t_n$  et les valeurs de la dérivée

première aux instants  $t_n$ ,  $t_n - 1$  et  $t_n - 2$ . En conséquence, nous choisissons le vecteur d'état  $\mathbf{x}_{t_n}$  selon :

$$\mathbf{x}_{t_n} = \begin{bmatrix} x_{t_n} \\ h x_{t_n}^{(1)} \\ h x_{t_n-1}^{(1)} \\ h x_{t_n-2}^{(1)} \end{bmatrix}$$

En réécrivant les équations (2) et (4) sous forme d'équations d'état, on obtient :

$$A_{t_n} = \begin{bmatrix} 1 & \frac{23}{12} & -\frac{16}{12} & \frac{5}{12} \\ 0 & 3 & -3 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

et

$$H_{t_n} = [1 \quad 0 \quad \dots \quad 0]$$

### 3.2. MÉTHODE DE RECONSTITUTION

La matrice  $A_{t_n}$  permet de prédire l'information; cette prédiction peut être corrigée grâce à l'équation (3). Au nouvel instant d'échantillonnage, un filtre de Kalman utilise alors la mesure pour lisser les informations prédites et corrigées. Ceci introduit dans la reconstitution le même retard que précédemment.

### 3.3. UTILISATION DU VECTEUR DE NORDSIECK

La matrice de transition dépend du choix du vecteur d'état. L'utilisation du vecteur d'état de Nordsieck [7], présente comme on va le voir l'avantage d'obtenir une matrice de transition triangulaire supérieure de Pascal, ce qui permet une diminution du temps de calcul, tout en gardant les mêmes performances de reconstitution, en comparaison avec la matrice de transition obtenue à partir de l'ancien vecteur d'état.

Le vecteur de Nordsieck est défini comme suit :

$$\mathbf{z}_{t_n} = \begin{bmatrix} x_{t_n} \\ h x_{t_n}^{(1)} \\ \frac{h^2}{2!} x_{t_n}^{(2)} \\ \frac{h^3}{3!} x_{t_n}^{(3)} \end{bmatrix}$$

nous avons alors :

$$z_{t_n+1} = D_{t_n} \mathbf{x}_{t_n+1}$$

$$z_{t_n+1} = D_{t_n} A_{t_n} \mathbf{x}_{t_n}$$

$$z_{t_n+1} = D_{t_n} A_{t_n} D_{t_n}^{-1} z_{t_n}$$

avec :

$$D_{t_n} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & h & 0 & 0 \\ 0 & 0 & \frac{h^2}{2!} & 0 \\ 0 & 0 & 0 & \frac{h^3}{3!} \end{bmatrix}$$

et donc :

$$\begin{aligned} z_{t_{n+1}} &= N_{t_n} z_{t_n} + v_{t_n} \\ y_{t_n} &= H_{t_n} z_{t_n} \end{aligned}$$

$$N_{t_n} = D_{t_n} A_{t_n} D_{t_n}^{-1} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

et

$$H_{t_n} = [1 \quad 0 \quad \dots \quad 0]$$

$N_{t_n}$  est la nouvelle matrice de transition, elle est triangulaire supérieure de Pascal.

## 4. Identification des éléments de la matrice de transition

L'algorithme de reconstitution précédent nécessite la connaissance des matrices de transition et d'observation dont la détermination se fait à partir du modèle du signal. Nous présentons dans ce qui suit les deux modélisations utilisées.

### 4.1. MODÉLISATION PAR DÉVELOPPEMENT DE TAYLOR

Pour représenter l'évolution à court terme d'un signal dont les observations (échantillons) sont peu perturbées (cas d'un signal de mesure) il est normal d'utiliser une approximation locale, la plus naturelle étant celle de Taylor.

On considère l'équation d'état qui s'écrit sous la forme suivante :

$$\begin{aligned} x_{t_{n+1}} &= A_{t_n} x_{t_n} + v_{t_n} \\ y_{t_n} &= H_{t_n} x_{t_n} \end{aligned}$$

Le vecteur d'état  $x_{t_n}$  est constitué par l'observation et ses  $p$  premières dérivées ( $p \leq 5$ ).

On a donc :

$$x_{t_n} = \begin{bmatrix} x_{t_n} \\ x_{t_n}^{(1)} \\ \vdots \\ x_{t_n}^{(p)} \end{bmatrix}$$

On utilise un développement de Taylor, qui permet de représenter l'évolution de  $x_{t_n}$  et de ses  $p$  premières dérivées, sous la forme suivante :

$$\begin{cases} x_{t_{n+i}} \approx x_{t_{n+(i-1)}} + h x_{t_{n+(i-1)}}^{(1)} + \frac{h^2}{2!} x_{t_{n+(i-1)}}^{(2)} \\ \quad + \dots + \frac{h^p}{p!} x_{t_{n+(i-1)}}^{(p)} \\ x_{t_{n+i}}^{(1)} \approx x_{t_{n+(i-1)}}^{(1)} + h x_{t_{n+(i-1)}}^{(2)} + \frac{h^2}{2!} x_{t_{n+(i-1)}}^{(3)} \\ \quad + \dots + \frac{h^{p-1}}{(p-1)!} x_{t_{n+(i-1)}}^{(p)} \\ \dots \\ x_{t_{n+i}}^{(p)} \approx x_{t_{n+(i-1)}}^{(p)} \end{cases} \quad (5)$$

Il est évident que l'indice  $i$  varie de 1 à  $k_n - 1$ . On rappelle que  $k_n$  et  $h$  représentent respectivement le nombre de mesures manquantes entre deux instants d'échantillonnage et la période de traitement. De plus, on considère que les dérivées de  $x_{t_{n+i}}$  d'ordre supérieur à  $p$  sont nulles (c'est-à-dire prises en compte dans le bruit d'état). Ce qui donne en comparant avec les équations d'état :

$$A_{t_n} = \begin{bmatrix} 1 & h & \frac{h^2}{2!} & \dots & \frac{h^p}{p!} \\ 0 & 1 & h & \dots & \frac{h^{p-1}}{(p-1)!} \\ 0 & 0 & 1 & \dots & \frac{h^{p-2}}{(p-2)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}$$

$$H_{t_n} = [1 \quad 0 \quad \dots \quad 0]$$

### 4.2. MODÉLISATION PAR DÉVELOPPEMENT DE TAYLOR MODIFIÉ

Dans ce cas le vecteur d'état  $x_{t_n}$  est constitué par les échantillons à l'instant  $t_n$  et aux instants précédents. Nous considérons le même développement de Taylor que précédemment; on utilise aussi le fait que les dérivées d'ordre supérieur à  $p$  sont nulles. Ceci va nous permettre de remplacer, de proche en proche, toutes les dérivées et de ne garder à la fin qu'une seule équation de la forme :

$$x_{t_n} = f(x_{t_{n-1}}, x_{t_{n-2}}, \dots, x_{t_{n-p}})$$

On commence par appliquer la décomposition de Taylor en utilisant  $x_{t_n}$  à l'ordre zéro :

$$\begin{cases} x_{t_n} \approx x_{t_{n-1}} + h x_{t_{n-1}}^{(1)} + \frac{h^2}{2!} x_{t_{n-1}}^{(2)} \\ \quad + \dots + \frac{h^p}{p!} x_{t_{n-1}}^{(p)} \\ x_{t_{n-1}} \approx x_{t_{n-2}} + h x_{t_{n-2}}^{(1)} + \frac{h^2}{2!} x_{t_{n-2}}^{(2)} \\ \quad + \dots + \frac{h^p}{p!} x_{t_{n-2}}^{(p)} \\ \dots \\ x_{t_{n-(p-1)}} \approx x_{t_{n-p}} + h x_{t_{n-p}}^{(1)} + \frac{h^2}{2!} x_{t_{n-p}}^{(2)} \\ \quad + \dots + \frac{h^p}{p!} x_{t_{n-p}}^{(p)} \end{cases} \quad (6)$$

et à l'ordre  $p$  :

$$\begin{cases} x_{t_n-1}^{(p)} \approx x_{t_n-2}^{(p)} \\ x_{t_n-(p-1)}^{(p)} \approx x_{t_n-p}^{(p)} \end{cases}$$

En reportant les conditions précédentes dans (5) et (6) et en posant :

$$\Delta^p x_{t_n} = \Delta^{p-1}(\Delta x_{t_n}) \quad \text{et} \quad \Delta x_{t_n-i} = x_{t_n-i} - x_{t_n-(i+1)}$$

on obtient :

$$\Delta^p x_{t_n} = \sum_{i=0}^p C_p^i (-1)^i x_{t_n-i} = 0$$

ce qui conduit enfin à

$$x_{t_n} = \sum_{i=1}^p C_p^i (-1)^{i+1} x_{t_n-i}$$

La comparaison avec les équations d'état permet de choisir :

$$A_{t_n} = \begin{bmatrix} C_p^1 & -C_p^2 & C_p^3 & \dots & (-1)^{p+1} \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & & & & \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

$$H_{t_n} = [1 \quad 0 \quad \dots \quad 0]$$

### Remarque :

La modélisation par développement de Taylor modifiée présente l'avantage d'avoir une matrice de transition ayant beaucoup plus d'éléments nuls que celle de la première modélisation, ce qui permet de diminuer le temps de calcul. Cependant les deux modélisations permettent les mêmes performances en terme de reconstitution du signal.

## 4.3. IDENTIFICATION DES PARAMÈTRES ET RECONSTITUTION D'UN SIGNAL AR

### 4.3.1. Introduction

Dans le but de reconstruire un signal autorégressif sans connaître les paramètres de son modèle, on se propose d'identifier ces paramètres à chaque instant d'échantillonnage et de reconstruire le signal avec un retard maximum égal à  $(d_n)_{max}$ . Pour cela, nous utilisons une combinaison de deux algorithmes récurrents : l'algorithme des moindres carrés récursifs (MCR) et l'algorithme prédicteur-lisseur de Kalman.

En effet, à chaque instant d'échantillonnage, nous pouvons identifier un vecteur des paramètres en utilisant l'algorithme MCR

[11][12], sur une fenêtre glissante contenant  $N$  points. Nous considérons que le vecteur des paramètres reste constant entre deux instants d'échantillonnage ce qui nous permet de reconstruire le signal en utilisant l'algorithme prédicteur-lisseur de Kalman. Cela veut dire que la matrice  $A_{t_n}$  utilisée dans cet algorithme va varier en fonction du vecteur des paramètres identifiés. Cette méthode peut être utilisée pour identifier des paramètres constants ou variables dans le temps.

### 4.3.2. Algorithmes MCR et prédicteur-lisseur de Kalman

On considère un signal AR représenté par son équation aux différences écrite sous forme condensée :

$$x_{t_n} + \theta_{t_n}^T x_{t_n} = b_{t_n}$$

en posant :

$$\begin{aligned} \theta_{t_n} &= [a_1 \quad a_2 \quad \dots \quad a_p]^T \\ \mathbf{x}_{t_n} &= [x_{t_n-1} \quad x_{t_n-2} \quad \dots \quad x_{t_n-p}]^T \end{aligned}$$

qui sont respectivement le vecteur de régression des observations ou vecteur des paramètres à identifier et le vecteur des observations passées. Dans ces observations on trouve les mesures vraies données à chaque instant d'échantillonnage et les estimées des mesures manquantes entre deux instants d'échantillonnage. Ces estimées sont données par l'algorithme prédicteur-lisseur de Kalman.  $b_{t_n}$  représente un élément d'une séquence aléatoire de moyenne nulle et de variance  $\sigma^2$ .

### 4.3.3. Exposé de la méthode

La méthode utilise l'algorithme MCR à chaque instant d'échantillonnage pour identifier le vecteur des paramètres sur un horizon d'observation  $N$  borné, tout en privilégiant classiquement les valeurs les plus récentes par un facteur d'oubli. On note par  $X_{t_n}$  la matrice des observations et par  $\mathbf{K}_{t_n}$  le gain d'adaptation de l'algorithme, avec :

$$X_{t_n} = \begin{bmatrix} \mathbf{x}_{t_n}^T \\ \mathbf{x}_{t_n-1}^T \\ \vdots \\ \mathbf{x}_{t_n-N}^T \end{bmatrix} = \begin{bmatrix} x_{t_n-1} & x_{t_n-2} & \dots & x_{t_n-p} \\ x_{t_n-2} & x_{t_n-3} & \dots & x_{t_n-(p-1)} \\ \vdots & \vdots & \ddots & \vdots \\ x_{t_n-N-1} & x_{t_n-N-2} & \dots & x_{t_n-N-p} \end{bmatrix}$$

On rappelle les notations classiques de l'algorithme MCR :

$$\begin{aligned} O_{t_n} &= (X_{t_n}^T X_{t_n})^{-1} \\ \mathbf{K}_{t_n+1} &= O_{t_n} \mathbf{x}_{t_n+1} (1 + \mathbf{x}_{t_n+1}^T O_{t_n} \mathbf{x}_{t_n+1})^{-1} \\ O_{t_n+1} &= (\mathbf{I} - \mathbf{K}_{t_n+1} \mathbf{x}_{t_n+1}^T) O_{t_n} \\ \hat{\theta}_{t_n+1} &= \hat{\theta}_{t_n} + \mathbf{K}_{t_n+1} (\mathbf{x}_{t_n+1} - \hat{\theta}_{t_n}^T \mathbf{x}_{t_n}) \end{aligned}$$

Dans la pratique, on initialise  $\theta_{t_n}$  et  $O_{t_n}$  à des valeurs arbitraires. L'instabilité numérique potentielle des formules précédentes ne joue pas, du fait des faibles horizons d'observation utilisés.

On a essayé de trouver un compromis entre la taille de la fenêtre  $N$  et le temps de calcul. Plus la fenêtre est grande plus l'on connaît le passé du signal, ce qui permet une bonne identification des paramètres, le temps de calcul quant à lui augmentant. La solution retenue [6] est  $N = 3p$ ,  $p$  étant l'ordre du modèle AR.

Le vecteur des paramètres identifié  $\hat{\theta}_{t_n}$  est introduit dans la matrice de transition  $A_{t_n}$  du modèle d'état. Dans le cas d'un signal AR d'ordre deux nous aurons ainsi :

$$A_{t_n} = \begin{bmatrix} -\hat{a}_{1t_n} & -\hat{a}_{2t_n} \\ 1 & 0 \end{bmatrix}$$

ce qui permet, en utilisant l'algorithme prédicteur-lisseur, de reconstruire le signal entre deux instants d'échantillonnage en gardant  $\hat{\theta}_{t_n}$  constant pendant ce temps.

## 5. Étude comparative des méthodes

Nous avons essayé ces méthodes sur deux types de signaux : des signaux « AR » et des signaux « basse fréquence », ces derniers étant obtenus en filtrant un bruit blanc par un filtre passe-bas de fréquence de coupure réduite égale à 0.05. Ces signaux sont échantillonnés de la façon suivante :  $x_{t_n}$  est obtenu par :

$$x_{t_n} = x(t_{n-1} + d_n)$$

expression dans laquelle  $t_{n-1}$  représente l'instant d'échantillonnage de  $x_{t_{n-1}}$ ,  $d_n$  étant une variable aléatoire de loi uniforme sur l'intervalle  $[0, 2 \cdot Dtmoy]$ .  $Dtmoy$  est donc l'intervalle de temps moyen entre deux échantillons successifs.

Pour évaluer les performances de ces algorithmes, nous utilisons deux erreurs, une locale et une autre globale. L'erreur locale, notée  $Erq_{moy}$  est par définition la racine carrée de l'erreur quadratique moyenne entre deux instants d'échantillonnage :

$$Erq_{moy} = \sqrt{\frac{1}{t_{n+1} - t_n} \sum_{i=t_n}^{t_{n+1}} (x_{rec} - x)^2}$$

Le nombre de points utilisés pour calculer cette erreur dépend de  $t_{n+1} - t_n$ , temps séparant deux instants d'échantillonnage. La taille du vecteur contenant les erreurs locales sera donc plus petite que la taille du signal. L'erreur globale est une erreur relative scalaire, notée  $Errel$ , qui exprime le rapport entre l'écart type de l'erreur de reconstitution globale et l'écart type du signal à reconstruire.

À titre d'exemple, nous effectuons une comparaison avec les A-splines récurrentes [14], méthode récente qui, à notre connaissance, est la seule qui corresponde à la problématique que nous envisageons. Cette méthode consiste à approcher entre  $t_n$  et  $t_{n+1}$  le signal par un polynôme de degré 3 assurant en  $t_n$  les continuités du signal et de sa dérivée. Elle autorise une formulation

récurrente et donne des résultats tout à fait estimables quand les échantillons ne sont pas trop éloignés. On peut donc légitimement la considérer comme une référence, tout en notant qu'elle ne permet pas la prédiction.

Pour un  $Dtmoy$  fixé, nous appliquons l'échantillonneur décrit ci-dessus sur cent signaux AR d'ordre deux différents et cent signaux passe bas différents. Pour chaque signal reconstruit nous calculons l'erreur globale et, à partir de la reconstitution des cent signaux, nous calculons une erreur globale moyenne. Nous répétons l'opération pour différents  $Dtmoy$ .

### 5.1. TABLEAU DE SIMULATION

Dans le tableau, on lira :

P.L.K = Prédicteur-lisseur de Kalman,

P.C. = Prédicteur-correcteur,

A-S.R. = A-splines récurrentes,

Nb. d'éch. réels = Nombre d'échantillons réels.

Le nombre d'échantillons reconstruits est égal à 200.

On constatera que les deux méthodes que nous proposons donnent toutes les deux des résultats de bonne qualité, les écarts sont faibles entre les chiffres, et dépendent plus des signaux que des méthodes.

### 5.2. REMARQUES

L'algorithme prédicteur-lisseur de Kalman donne de meilleurs résultats que les deux autres algorithmes dans le cas des signaux AR. Ceci vient du caractère purement statistique de cet algorithme. Par contre dans le cas des signaux basse-fréquence le prédicteur-correcteur se comporte, de façon générale, mieux que les deux autres algorithmes. Le lissage que nous utilisons dans l'algorithme prédicteur-lisseur de Kalman et le prédicteur-correcteur permet de corriger les erreurs de prédiction et de donner de bons résultats.

Enfin, on peut aussi noter que l'erreur globale moyenne varie inversement au nombre d'échantillons quelle que soit la méthode utilisée.

### 5.3. FIGURES

Dans ce paragraphe, nous présentons les résultats de reconstitution d'un signal AR d'ordre deux échantillonné avec une période qui suit une loi de distribution uniforme (fig. 5.1.). L'intervalle de temps moyen qui sépare deux instants d'échantillonnage est égal à 8h. Sur la figure les croix désignent les instants d'échantillonnage réel.

Les algorithmes utilisés pour reconstituer le signal AR sont le prédicteur-lisseur de Kalman utilisant une modélisation par

$\left\{ \frac{D_{moy}}{\text{Nb. d'éch. réels}} \right\}$	Erreur globale (signal AR)			Erreur globale (signal basse-fréquence)		
	PL.K.	P.C.	A-S.R.	PL.K.	P.C.	A-S.R.
$\left\{ \frac{8}{55} \right\}$	0,0751	0,0861	0,0849	0,0842	0,0762	0,0774
$\left\{ \frac{10}{45} \right\}$	0,102	0,110	0,108	0,112	0,102	0,106
$\left\{ \frac{12}{36} \right\}$	0,128	0,136	0,131	0,181	0,175	0,222
$\left\{ \frac{14}{31} \right\}$	0,130	0,151	0,150	0,233	0,225	0,274
$\left\{ \frac{16}{28} \right\}$	0,150	0,176	0,171	0,264	0,243	0,275

développement de Taylor d'ordre deux (fig. 5.2.) et le prédicteur-correcteur utilisant une modélisation par vecteur de Nordsieck d'ordre quatre (fig. 5.3.). Nous avons reconstruit le signal à partir de la connaissance de 30% seulement du signal original. Sur les figures le signal original est en trait plein, le signal reconstruit en pointillé. Les figures 5.2.a, 5.3.a, etc. représentent le graphe de l'erreur locale, la valeur de l'erreur globale y est affichée en commentaire.

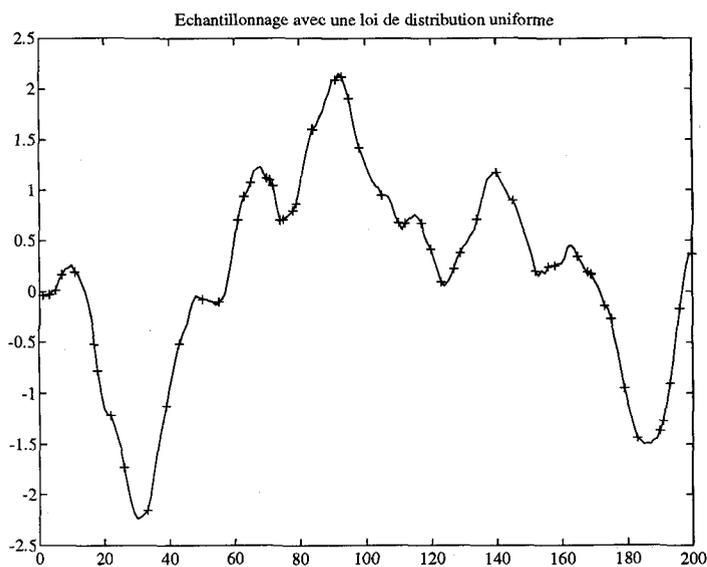


Figure 5.1

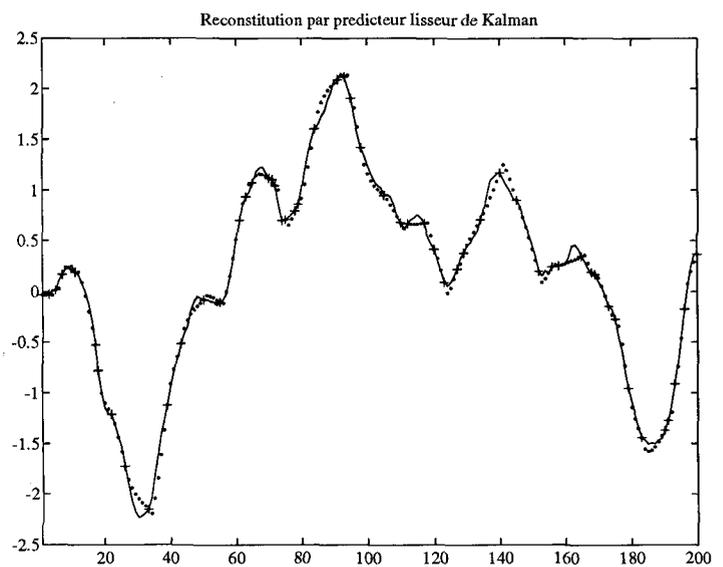


Figure 5.2

À titre de comparaison, la figure 5.4 montre la reconstitution du signal AR par les A-splines récurrentes.

La figure 5.5 représente les courbes d'identification des paramètres du signal AR en utilisant l'algorithme MCR et prédicteur-lisseur de Kalman. Ce résultat montre que nous pouvons identifier dans ces conditions des signaux dans lesquels manque une forte proportion d'échantillons (50% d'échantillons manquant).

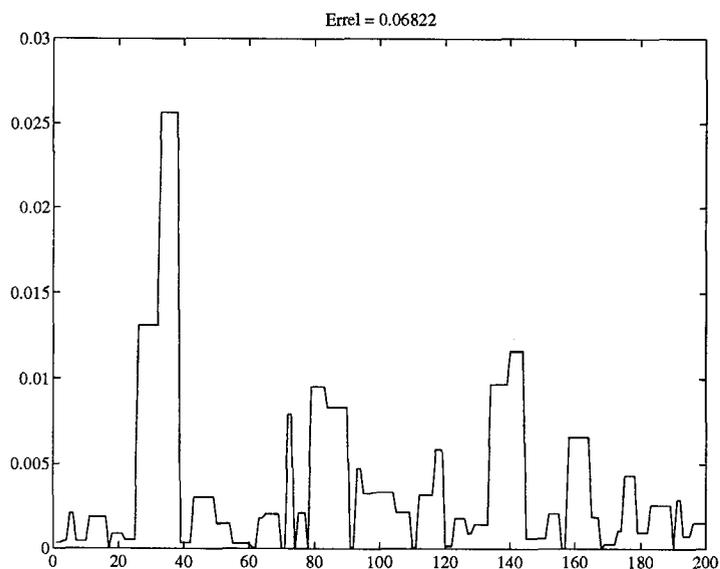


Figure 5.2a

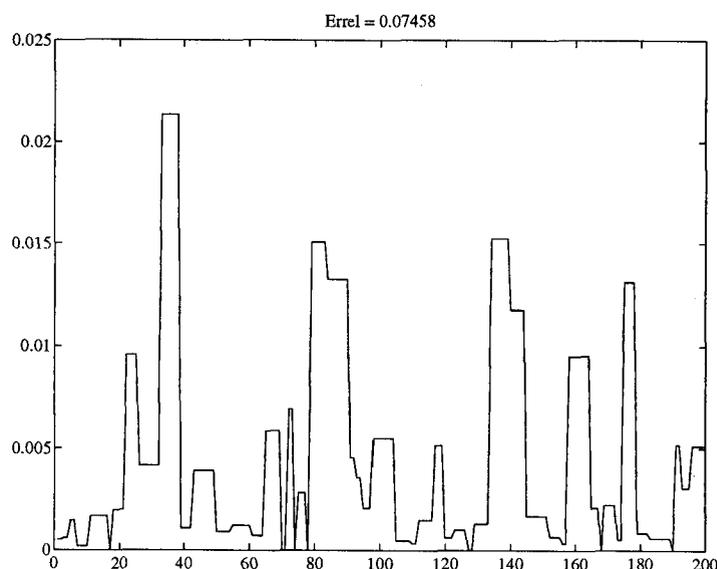


Figure 5.3a

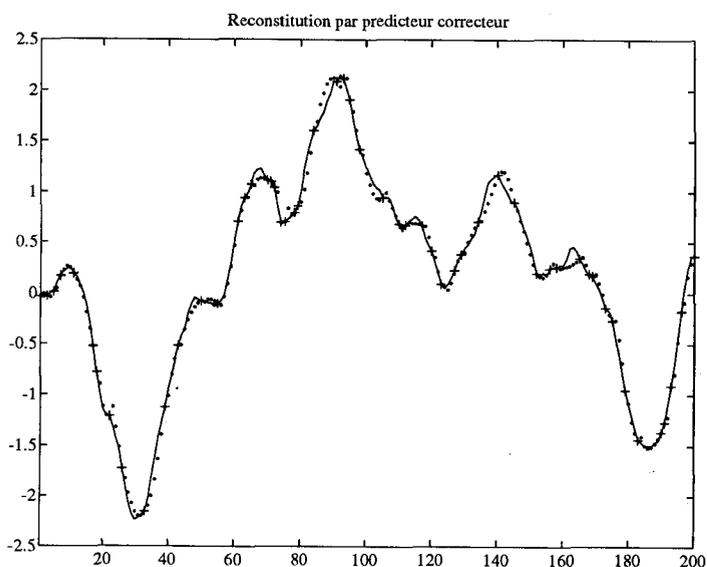


Figure 5.3

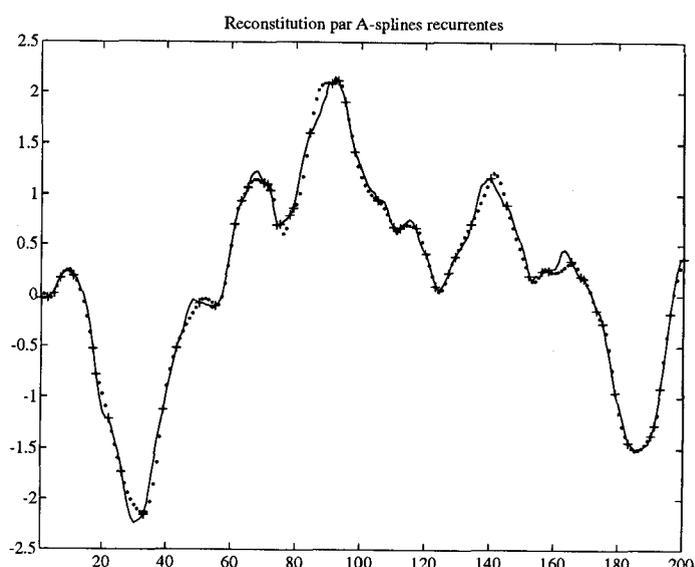


Figure 5.4

## 6. Conclusions

Les techniques de traitement statistique et d'analyse numérique ainsi que leur association aboutissent à des résultats de simulation qui montrent une bonne qualité de la reconstitution des signaux. Bien évidemment, le « lissage » introduit un retard pur obligatoire égal au plus grand intervalle entre échantillons ce qui n'est toutefois pas trop pénalisant dans beaucoup d'applications potentielles.

Nous soulignons l'intérêt de coupler des méthodes de traitement statistique et des méthodes d'analyse numérique dans le cas présent où il est nécessaire de s'intéresser à un signal sur un

horizon, intermédiaire entre le court terme (domaine de l'analyse numérique) et le moyen ou long terme (domaine des modèles AR ou ARMA du traitement du signal).

Nous terminons cette conclusion en citant l'échantillonnage « adaptatif » qui est de plus en plus utilisé et nous pensons que l'utilisation des méthodes de reconstitution temps réel permettra de profiter de tous les avantages d'un tel échantillonnage : en particulier la minimisation de la consommation des systèmes électroniques de mesures et la compression des données dans ces mêmes systèmes.

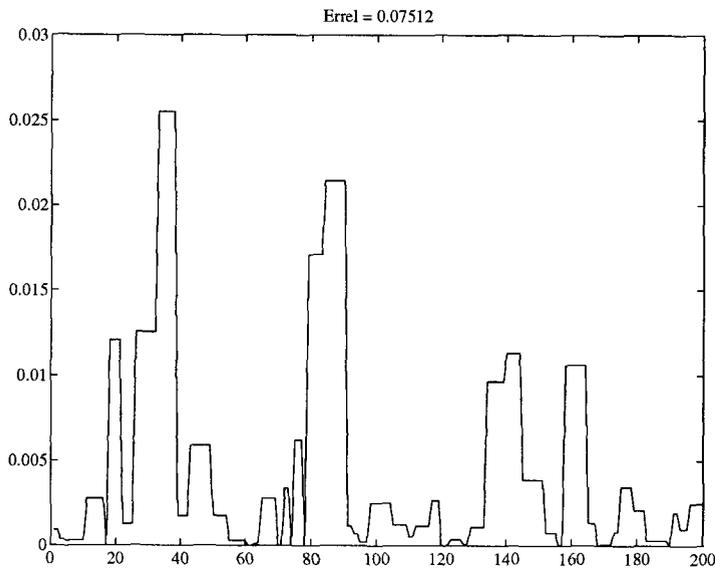


Figure 5.4a

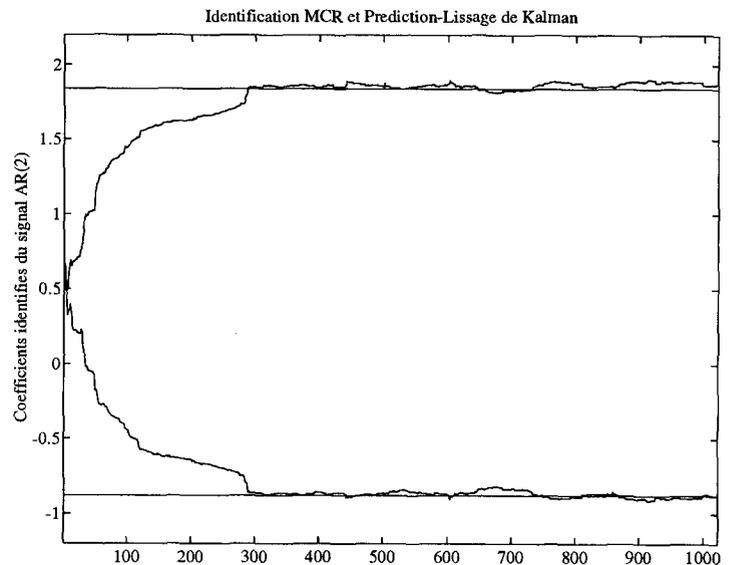


Figure 5.5

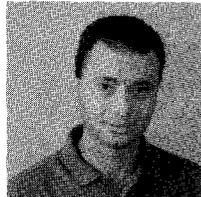
## BIBLIOGRAPHIE

- [1] L. ROMERO, « Reconstitution d'un signal échantillonné avec un pas non uniforme », *Master de l'École Supérieure d'Électricité*, 1990.
- [2] R.J. MARKS ed., « Advanced Topics in Shannon Sampling and Interpolation Theory », *Springer-Verlag*, 1993.
- [3] J.M. BARDET, N. DURAND, « Reconstitution de signaux échantillonnés non-uniformément », rapport interne, Service des Mesures, École Supérieure d'Électricité, 1991.
- [4] E. VAN DER OUDEVAA, J. RENNEBOOG, « Some formulas and applications of non-uniform sampling of bandwidth limited signals », *IEEE Transaction on instrumentation and measurement*, vol. 37, n°3, september 1988, p. 353-357.
- [5] M. LABARRERE, J.P. KRIEF, B. GIMONET, « Le filtrage et ses applications », *CEPADUES*, 1978.
- [6] G. CHEN, « A Simple Treatment For Suboptimal Kalman Filtering in Case of Measurement Data Missing », *IEEE Trans. Aeros. Elec. Sys.*, vol. 26, n°2, mars 1990, p. 413-415.
- [7] O. CHUA, P. LIN, « Computer aided analysis of electronic devices », *PRENTICE-HALL, INC. Englewood Cliffs, New Jersey*, 1975.
- [8] J. STOER, R. BULIRSCH, « Introduction to numerical analysis », *Springer-Verlag*, Second Edition, New York, Heidelberg, Berlin.
- [9] W.H. PRESS, B.P. FLANNERY, S.A. TEUKOISKY, « Numerical recipes, The art of scientific computing », *Cambridge University Press* 86, chapter 3, 1986.
- [10] S.J. OVASKA, « Newton-type predictors – A signal processing oriented viewpoint », *Signal Processing*, n°25, 1991, p. 251-257.
- [11] B. PICINBONO, « Traitement du Signal », *École Supérieure d'Électricité*, 1991.

- [12] M. NAJIM, « Modélisation et identification en traitement du signal », *Masson*, 1988, p. 84-102.
- [13] H. VAEZI-NEJAD, « Détection de défauts d'instrumentations de mesures », Doctorat de l'Université de Nancy I, Janvier 1990.
- [14] A. SEREDINSKI, « Principe de la méthode des A-splines récurrentes pour l'interpolation et la compression des signaux », *Traitement du signal*, vol. 9, n°2, 1992, p. 175-180.

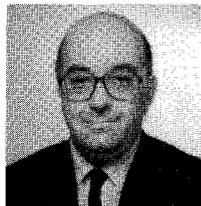
## LES AUTEURS

### Othmane BENSAOUD



Othmane Bensaoud est né en 1963. Après une maîtrise de physique, une maîtrise EEA et un DEA en métrologie électronique et automatique à l'Institut Polytechnique de Lorraine, il termine un Doctorat en traitement du signal à l'École Supérieure d'Électricité. Ses travaux portent sur les signaux à échantillonnage non périodique.

### Jacques OKSMAN



Jacques Oksman, ingénieur SUPÉLEC, est professeur à l'École Supérieure d'Électricité. Il est responsable du groupe « Systèmes de traitement des signaux » du Service des Mesures. Ses domaines de recherche concernent principalement le traitement du signal appliqué aux problèmes de mesures (méthodes d'estimation précises dans un cadre non linéaire, architectures associées).

Manuscrit reçu le 6 mars 1993.