

CHAPITRE 6. ALGORITHMES AR RAPIDES.

Ce chapitre est consacré à la présentation des algorithmes qui permettent la résolution rapide des équations de Yule-Walker obtenues au chapitre précédent en (2-47) ou (2-51). Ces algorithmes ne sont pas originaux, leur principe ayant été trouvé pour l'un par MORF, DICKINSON, KAILATH, VIEIRA, 1977, et pour l'autre par SAKAI, 1982. Cependant leur utilisation pour le problème posé ici n'est pas immédiate, du fait que les deux systèmes d'équations (2-47) et (2-51) ne sont ni l'un ni l'autre symétriques. De plus l'algorithme donné par MORF et al., 1977, est écrit pour le cas du signal scalaire. Le passage des formes originales des deux algorithmes aux formes utiles ici n'est pas d'une grande complexité pour qui est quelque peu familier avec le mécanisme de ces algorithmes, cependant il semble utile de décrire ici ces variantes pour épargner ces tâtonnements aux éventuels utilisateurs futurs des méthodes évolutives. L'exposé est repris de GRENIER, 1981-c, pour l'algorithme de Sakai, puis pour celui de Morf et al., de GRENIER, 1982-b, où figure le code Fortran de cette variante.

Le chapitre est divisé en une exposition constructive de l'algorithme de Morf et al., un résumé de l'ordonnancement temporel de cet algorithme, l'exposé de la variante de celui de Sakai, puis en conclusion une comparaison entre le concept de rang de déplacement et celui d'indice d'évolution m qui apparaît dans la méthodologie présente.

1. Construction de l'algorithme.

En définissant les deux vecteurs Y_t et X_t comme suit, on réécrira (2-47) sous la forme:

$$(2-64) \quad X_t = \begin{bmatrix} Y_{t-q-1} \\ \vdots \\ Y_{t-q-p} \end{bmatrix} \quad \text{et} \quad Y_t = \begin{bmatrix} Y_{t-1} \\ \vdots \\ Y_{t-p} \end{bmatrix}$$

$$(2-65) \quad \theta^T \sum_{t=p}^T Y_t X_t^T = [v_1 \dots v_p]$$

que l'on notera encore:

$$(2-66) \quad \theta^T R_p = v^T$$

Il est utile aussi de définir les matrices X_p et Y_p par:

$$(2-67) \quad X_p = [X_p \dots X_T] \quad \text{et} \quad Y_p = [Y_p \dots Y_T]$$

Les matrices X_p et Y_p sont toutes deux des matrices de Toëplitz par bloc, chaque bloc étant une colonne à $(m+1)$ lignes:

$$(2-68) \quad Y_p = \begin{bmatrix} Y_{p-1} & & & Y_{T-p+1} & Y_T \\ & \ddots & & & \\ & & & & \\ Y_0 & & Y_{p-1} & & Y_{T-p+1} \end{bmatrix}$$

L'équation (2-65) correspond à la méthode "de covariance", tandis que celle "de corrélation" s'obtiendrait par un allongement de l'intervalle de sommation, et un prolongement de Y_p et X_p par des zéros au-delà de $[0, T]$.

Dans la méthode "de corrélation", la matrice R_p est de Toëplitz. Dans le cas scalaire, la résolution des équations de Yule-Walker a été donnée par LEVINSON, 1946, et elle correspond à un coût de calcul proportionnel à p^2 alors que la résolution du système de dimension p requiert ordinairement de l'ordre de p^3 opérations. De cette réduction d'un ordre de grandeur du nombre d'opérations vient le terme d'algorithmes rapides dont on désigne cet ensemble de méthodes fondées sur la structure des matrices de Toëplitz.

Quand la matrice est de Toëplitz par blocs, ce qui correspond au modèle autorégressif vectoriel, il existe une généralisation de l'algorithme de Levinson, due à WHITTLE, 1963, pour qui le second membre de l'équation restait particulier, et extrait de R_p . Le cas général, second membre quelconque a été résolu par WIGGINS, ROBINSON, 1965.

Pour ce qui est de la méthode "de covariance", la matrice R_p n'étant plus elle-même de Toëplitz, il a fallu attendre une date plus récente pour que soit reconnu le fait que la structure de R_p en tant que produit de deux matrices de Toëplitz permettait encore l'obtention d'un algorithme rapide. Cette observation a permis à MORF, DICKINSON, KAILATH, VIEIRA, 1977, d'élaborer la solution du cas scalaire sans second membre, étendue au cas d'un second membre quelconque par FRIEDLANDER, KAILATH, MORF, LJUNG, 1978. Nous allons voir comment se résout le cas vectoriel non-symétrique avec second membre, en élaborant progressivement la solution dans un ordre qui éclaire la façon dont MORF et al, 1977, ont pu parvenir à la leur.

Le problème est donc de résoudre un système du type $u.R_p=v$ où u et v sont décomposés en blocs lignes à $(m+1)$ colonnes et R_p est décomposé en blocs carrés à $(m+1)$ lignes et colonnes. Les blocs de u seront $u_1 \dots u_2$ et ceux de v , $v_1 \dots v_2$. L'idée de LEVINSON, 1946, pour résoudre un tel système est double. Elle consiste tout d'abord à le plonger dans un problème plus vaste comprenant la recherche de quantités A_i et B_i , ici matrices carrées à $(m+1)$ lignes et colonnes telles que:

$$(2-69) \quad \begin{bmatrix} I & A_1 & \dots & A_{p-1} \\ B_{p-1} & & \dots & B_1 & I \end{bmatrix} R_p = \begin{bmatrix} E_p^+ & 0 & \dots & 0 \\ 0 & \dots & 0 & E_p^- \end{bmatrix}$$

Ces variables intermédiaires sont, lorsque $X_p=Y_p$ les matrices des modèles

autorégressifs directs et rétrogrades (forward, backward) pour le vecteur Y_t . Le second trait de génie de LEVINSON, 1946, a été de rendre la recherche d'une solution, récursive en passant par les solutions intermédiaires de $k=1$ à $k=p-1$ de:

$$(2-70) \quad \begin{bmatrix} u_1 & u_2 & \dots & \cdot & u_k \\ I & A_1 & \dots & \cdot & A_{k-1} \\ B_{k-1} & \cdot & \dots & B_1 & I \end{bmatrix} R_k = \begin{bmatrix} v_1 & \dots & v_k \\ E_k^+ & 0 & \dots & 0 \\ 0 & \dots & 0 & E_k^- \end{bmatrix}$$

REMARQUE: Si dans (2-70), les v_i sont bien les blocs de v tels qu'ils figurent dans le second membre de $u.R_p=v$, les u_i , A_i et B_i du terme gauche dépendent tous de l'indice k de R_k . Par exemple le u_1 de la solution d'ordre k n'est pas le u_1 de la solution d'ordre $k+1$. Il faudrait donc deux indices pour repérer les u_i , A_i , B_i : leur indice courant et un second indice indiquant l'ordre du système dont ils sont extraits. Ce second indice sera, pour alléger la notation, négligé dans ce qui suit car les u_i , A_i , B_i ne seront jamais employés seuls, et il suffit de se référer à la valeur maximale du premier indice dans les équations matricielles pour connaître la valeur du second indice.

Comment passer de la solution d'ordre k à celle d'ordre $k+1$? MORF et al., 1977, ont montré que deux étapes successives étaient nécessaires. Commençons par la seconde qui constitue le coeur de l'algorithme de Levinson, et supposons que nous ayons pu modifier les solutions de (2-70) à l'ordre k pour vérifier le système (2-71) suivant:

$$(2-71) \begin{bmatrix} u_1^* & u_2^* & \dots & u_k^* & 0 \\ I & A_1^* & \dots & A_{k-1}^* & 0 \\ 0 & B_{k-1}^* & \dots & B_1^* & I \end{bmatrix} R_{k+1} = \begin{bmatrix} v_1 & \dots & v_k & v_{k+1}^* \\ E_k^{*+} & 0 \dots 0 & \Delta_k^+ \\ \Delta_k^- & 0 \dots 0 & E_k^{*-} \end{bmatrix}$$

Dans ce système, les $(k-1)$ blocs de zéros au second membre fixent les A_i^* et B_i^* , puis les E^* et Δ en découlent, de même les v_i fixent les u_i^* , puis v_{k+1}^* en découle. Si nous obtenions $\Delta_k^+ = \Delta_k^- = 0$ et $v_{k+1}^* = v_{k+1}$, nous aurions les solutions à l'ordre $k+1$. Notre but est donc de forcer ces conditions. Ceci s'obtient pour Δ par un choix de matrices K_{k+1}^+ , K_{k+1}^- :

$$(2-72) \quad K_{k+1}^+ = -\Delta_k^+ (E_k^{*+})^{-1} \quad \text{et} \quad K_{k+1}^- = -\Delta_k^- (E_k^{*-})^{-1}$$

Ces matrices dans lesquelles on reconnaît des corrélations partielles servent à former les A_i et B_i de la solution d'ordre $k+1$:

$$(2-73) \quad \begin{bmatrix} I & A_1 & \dots & A_k \\ B_k & & \dots & B_1 & I \end{bmatrix} = \begin{bmatrix} I & K_{k+1}^+ \\ K_{k+1}^- & I \end{bmatrix} \begin{bmatrix} I & A_1^* & \dots & A_{k-1}^* & 0 \\ 0 & B_{k-1}^* & \dots & B_1^* & I \end{bmatrix}$$

Ces nouvelles matrices, grâce au choix (2-72) sont bien les solutions cherchées. Pour ce qui est de forcer l'égalité $v_{k+1}^* = v_{k+1}$, ceci s'obtient en considérant la matrice E_{k+1}^- issue de (2-73) et le vecteur-ligne L_{k+1} :

$$(2-74) \quad E_{k+1}^- = E_k^- + K_{k+1}^- \Delta_k^+$$

$$(2-75) \quad L_{k+1} = (v_{k+1} - v_{k+1}^*) (E_{k+1}^-)^{-1}$$

La solution sur les u_i s'obtient alors comme:

$$(2-76) \quad [u_1 \dots u_{k+1}] = [u_1^* \dots u_k^* \ 0] + L_{k+1} [B_k \dots B_1 \ I]$$

L'étape qui vient d'être décrite est due à LEVINSON, 1946, et constituait l'essentiel de son algorithme, voire même sa totalité, car dans le cas qu'il

considérât, la matrice R_p étât de Toëplitz, et le passage de l'équation (2-70) à l'équation augmentée (2-71) étât trivial: $A_i=A_i^*$, $B_i=B_i^*$, $u_i=u_i^*$, ceci provenant du fait que R_k se retrouve à la fois dans le coin supérieur gauche et dans le coin inférieur droit de R_{k+1} . Dans le cas plus général de la méthode de covariance, R_k n'est plus de Toëplitz, mais s'obtient par le produit des deux matrices de Toëplitz X_k et Y_k . L'inclusion de R_k dans R_{k+1} se transforme en une relation mise en évidence par MORF et al., 1977. Soient R_k^+ et R_k^- les coins supérieur gauche et inférieurs droit de R_{k+1} obtenus par suppression d'une ligne-bloc et d'une colonne-bloc.

$$(2-77) \quad R_{k+1} = \left[\begin{array}{c|c} R_k^+ & \\ \hline & R_k^- \end{array} \right]$$

De l'égalité $R_{k+1} = Y_{k+1} X_{k+1}^T$ découlent directement les relations suivantes:

$$(2-78) \quad R_k^+ = R_k - \begin{bmatrix} Y_{k-1} \\ \cdot \\ \cdot \\ Y_0 \end{bmatrix} [X_{k-1}^T \dots X_0^T]$$

$$(2-79) \quad R_k^- = R_k - \begin{bmatrix} Y_T \\ \cdot \\ \cdot \\ Y_{T-k+1} \end{bmatrix} [X_T^T \dots X_{T-k+1}^T]$$

Ces deux relations constituent le sésame de l'algorithme de MORF et al., 1977, ainsi que de la variante développée ici. Pour le voir il suffit de décomposer le système (2-71) augmenté, en ses deux sous-systèmes exprimés sur R_k^+ et R_k^- :

$$(2-80) \quad \begin{bmatrix} u_1^* & \dots & u_k^* \\ I & A_1^* & \dots & A_{k-1}^* \end{bmatrix} R_k^+ = \begin{bmatrix} v_1 & \dots & v_k \\ E_k^{*+} & 0 & \dots & 0 \end{bmatrix}$$

$$(2-81) \quad [B_{k-1}^* \dots B_1^* I] R_k^- = [0 \dots 0 E_k^-]$$

Le passage de (2-70) à (2-71) peut s'amorcer par:

$$(2-82) \quad \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} R_k^+ = \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} R_k^- \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} \begin{bmatrix} Y_{k-1} \\ \cdot \\ \cdot \\ Y_0 \end{bmatrix} [X_{k-1}^T \dots X_0^T]$$

Ceci devient après définition de v_k et ϵ_k^+ , et en tenant compte de (2-70):

$$(2-83) \quad \begin{bmatrix} v_k \\ \epsilon_k^+ \end{bmatrix} = \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} \begin{bmatrix} Y_{k-1} \\ \cdot \\ \cdot \\ Y_0 \end{bmatrix}$$

$$(2-84) \quad \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} R_k^+ = \begin{bmatrix} v_1 & \dots & v_k \\ E_k^{*+} & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} v_k \\ \epsilon_k^+ \end{bmatrix} [X_T^T \dots X_{T-k+1}^T]$$

En procédant de même sur les B_i , on obtient:

$$(2-85) \quad [B_{k-1}^* \dots B_1^* I] R_k^- = [0 \dots 0 E_k^{*-}] - [\epsilon_k^-] [X_T^T \dots X_{T-k+1}^T]$$

où on a posé:

$$(2-86) \quad [\epsilon_k^-] = [B_{k-1}^* \dots B_1^* I] \begin{bmatrix} Y_T \\ \cdot \\ \cdot \\ Y_{T-k+1} \end{bmatrix}$$

La comparaison de (2-84) avec (2-80) et de (2-85) avec (2-81) montre que les A_i^* , B_i^* et u_i^* peuvent s'obtenir par simple correction des A_i , B_i et u_i pourvu que l'on possède les quantités c_i , d_i telles que:

$$(2-87) \quad \begin{bmatrix} d_0 & \dots & d_{k-1} \\ c_{k-1} & \dots & c_0 \end{bmatrix} R_k = \begin{bmatrix} X_{k-1}^T & \dots & X_0^T \\ X_T^T & \dots & X_{T-k+1}^T \end{bmatrix}$$

Le passage à R_k^+ et R_k^- se fait en introduisant f_k et g_k tels que:

$$(2-88) \quad f_k = [d_0 \dots d_{k-1}] \begin{bmatrix} Y_{k-1} \\ \vdots \\ Y_0 \end{bmatrix} \quad \text{et} \quad g_k = [c_{k-1} \dots c_0] \begin{bmatrix} Y_T \\ \vdots \\ Y_{T-k+1} \end{bmatrix}$$

Ces relations jointes à celles reliant R_k , R_k^+ et R_k^- permettent de réécrire

(2-87) comme:

$$(2-89) \quad [d_0 \dots d_{k-1}] R_k^- = (1-f_k) [X_{k-1}^T \dots X_0^T] \\ [c_{k-1} \dots c_0] R_k^- = (1-g_k) [X_T^T \dots X_{T-k+1}^T]$$

Les deux équations précédentes permettent de corriger (2-84) pour obtenir le second membre voulu par (2-80), et corriger (2-85) pour obtenir (2-81), en effet:

$$(2-90) \quad \left(\begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} + \begin{bmatrix} v_k \\ \epsilon_k^+ \end{bmatrix} (1-f_k)^{-1} [d_0 \dots d_{k-1}] \right) R_k^+ = \begin{bmatrix} v_1 & \dots & v_k \\ E_k^+ & 0 & \dots & 0 \end{bmatrix}$$

$$(2-91) \quad ([B_{k-1} \dots B_1 \ I] + [\epsilon_k^-] (1-g_k)^{-1} [c_{k-1} \dots c_0]) R_k^- = [0 \dots 0 \ E_k^-]$$

La première ligne de (2-90) donne bien les u_i^* , mais les A_i et B_i corrigés par (2-90) et (2-91) ne respectent pas la condition de normalisation $A_0 = B_0 = I$. On aurait même précisément:

$$(2-92) \quad A_0 = I + \epsilon_k^+ (1-f_k)^{-1} d_0 \quad \text{et} \quad B_0 = I + \epsilon_k^- (1-g_k)^{-1} c_0$$

Il faut donc normaliser par l'inverse de A_0 et l'inverse de B_0 . Ces inverses se calculent d'ailleurs simplement par l'usage du lemme d'inversion matricielle qui s'écrit pour quatre matrices M , N , U , V :

$(M+UNV)^{-1} = M^{-1} - M^{-1}U(VM^{-1}U+N^{-1})^{-1}VM^{-1}$. Ce lemme appliqué à A_0 et B_0 donne:

$$(2-93) \quad \begin{cases} (A_0)^{-1} = I - \epsilon_k^+ (d_0 \epsilon_k^+ + 1 - f_k)^{-1} d_0 \\ (B_0)^{-1} = I - \epsilon_k^- (c_0 \epsilon_k^- + 1 - g_k)^{-1} c_0 \end{cases}$$

L'introduction des c_i et d_i a donc permis de compléter la première étape de l'algorithme mais la solution complète n'en est que repoussée puisqu'il faut maintenant trouver le moyen de transférer les c_i, d_i de la solution d'ordre k à ceux de la solution d'ordre $k+1$. Ceci n'est plus qu'un problème de combinaisons de matrices entre elles, calquées sur celles qui viennent d'être décrites, aussi ne seront-elles pas reprises ici (voir GRENIER, 1982-b, pp 13-15).

L'algorithme une fois complété se caractérise donc par l'emploi des variables auxiliaires A_i et B_i prédicteurs directs et rétrogrades, et par l'introduction des c_i et d_i qui sont les antécédents pour R_p des conditions finales et initiales de la variable X_t . Il se décompose en deux phases: la première est un passage des matrices R_k aux matrices augmentées R_{k+1} qui s'accompagne des corrections sur les A_i, B_i, u_i pour les adapter sans augmentation de leur ordre aux matrices d'ordre suivant. Dans la seconde phase, l'ordre des matrices R est inchangé, mais ce sont les variables A_i, B_i, u_i qui passent à l'ordre suivant. La présentation qui vient d'être faite a suivi la construction de l'algorithme en introduisant les quantités au fur et à mesure de leur nécessité. L'ordre chronologique s'en est vu bouleversé. Pour le rétablir, l'algorithme va maintenant être résumé, ce qui permettra de justifier le résultat annoncé au chapitre précédent: seules les matrices de corrélations définies par (2-48) sont à calculer, même dans la méthode "de covariance".

2. Déroulement temporel de l'algorithme.

La première étape se résume dans ce qui suit où $A=B$ est à lire comme: "affecter à A la valeur B". Les quantités u_i, A_i, B_i, c_i, d_i sont supposées connues à l'ordre k .

$$\text{Calculer: } \begin{bmatrix} f_k & h_k \\ l_k & g_k \end{bmatrix} = \begin{bmatrix} d_0 & \dots & d_{k-1} \\ c_{k-1} & \dots & c_0 \end{bmatrix} \begin{bmatrix} Y_{k-1} & Y_T \\ \cdot & \cdot \\ \cdot & \cdot \\ Y_0 & Y_{T-k+1} \end{bmatrix}$$

Calculer $v_k, \epsilon_k^+, \epsilon_k^-$:

$$\begin{bmatrix} v_k \\ + \\ \epsilon_k^- \end{bmatrix} = \begin{bmatrix} u_1 & u_2 & \dots & u_k \\ I & A_1 & \dots & A_{k-1} \end{bmatrix} \begin{bmatrix} Y_{k-1} \\ \cdot \\ \cdot \\ Y_0 \end{bmatrix} \quad \text{et} \quad [\epsilon_k^-] = [B_{k-1} \dots B_1 \quad I] \begin{bmatrix} Y_T \\ \cdot \\ \cdot \\ Y_{T-k+1} \end{bmatrix}$$

Corriger les A_i, B_i :

$$[A_1^* \dots A_{k-1}^*] = [A_1 \dots A_{k-1}] + \epsilon_k^+ (1-f_k)^{-1} [d_1 \dots d_{k-1}]$$

$$[B_{k-1}^* \dots B_1^*] = [B_{k-1} \dots B_1] + \epsilon_k^- (1-g_k)^{-1} [c_{k-1} \dots c_1]$$

Préparer la normalisation des A_i^*, B_i^* :

$$\rho_k^+ = d_0 \epsilon_k^+, \quad \rho_k^- = c_0 \epsilon_k^-, \quad e_k^+ = d_0 E_k^+, \quad e_k^- = d_0 E_k^-$$

$$[a_1 \dots a_{k-1}] = (\rho_k^+ + 1 - f_k)^{-1} d_0 [A_1^* \dots A_{k-1}^*]$$

$$[b_{k-1} \dots b_1] = (\rho_k^- + 1 - g_k)^{-1} c_0 [B_{k-1}^* \dots B_1^*]$$

Normaliser:

$$[A_1^* \dots A_{k-1}^*] = [A_1^* \dots A_{k-1}^*] - \epsilon_k^+ [a_1 \dots a_{k-1}]$$

$$[B_{k-1}^* \dots B_1^*] = [B_{k-1}^* \dots B_1^*] - \epsilon_k^- [b_{k-1} \dots b_1]$$

$$E_k^{*+} = E_k^+ - \epsilon_k^+ (\rho_k^+ + 1 - f_k)^{-1} e_k^+$$

$$E_k^{*-} = E_k^- - \epsilon_k^- (\rho_k^- + 1 - g_k)^{-1} e_k^-$$

Corriger les u_i, d_i, c_i :

$$\begin{bmatrix} u_1^* & \dots & u_k^* \\ d_0^* & \dots & d_{k-1}^* \\ c_{k-1}^* & \dots & c_0^* \end{bmatrix} = \begin{bmatrix} 1 & v_k(1-f_k)^{-1} & 0 \\ 0 & 1 & h_k(1-g_k)^{-1} \\ 0 & l_k(1-f_k)^{-1} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 & \dots & u_k \\ d_0 & \dots & d_{k-1} \\ c_{k-1} & \dots & c_0 \end{bmatrix}$$

A ce stade, la première étape est terminée. La seconde étape doit commencer par le calcul de $\Delta_k^+, \Delta_k^-, v_{k+1}^*$. On constate que ces trois quantités ne dépendent que de la première et la dernière colonne-bloc de R_{k+1} , dont on exclut d'ailleurs le bloc situé sur la diagonale. De plus dans chaque de ces colonnes, un seul bloc n'a pas encore été utilisé: le dernier bloc de la première colonne ou le premier de la dernière colonne. Chacun se calcule par une somme sur tous les échantillons dans la mesure où chaque matrice R_k est une matrice de covariance évaluée sur l'intervalle (k, T) . Ces deux derniers blocs sont donc bien des blocs de "corrélation" comme en (2-48). Les autres blocs s'obtiennent par simple correction de leur valeur à l'ordre précédent, conformément à (2-78) et (2-79).

La seconde phase se résume à: Calculer Δ_k^+, Δ_k^- et v_{k+1}^* :

$$\begin{bmatrix} v_{k+1}^* \\ \Delta_k^+ \end{bmatrix} = \begin{bmatrix} u_1^* & \dots & u_k^* \\ I & A_1^* & \dots & A_{k-1}^* \end{bmatrix} \times \text{dernière colonne de } R_{k+1}$$

$$[\Delta_k^-] = [B_{k-1}^* \dots B_1^* I] \times \text{première colonne de } R_{k+1}$$

Calculer les gains K_k^+, K_k^- :

$$K_k^+ = -\Delta_k^+ (E_k^-)^{-1} \quad \text{et} \quad K_k^- = -\Delta_k^- (E_k^+)^{-1}$$

Calculer les A_i, B_i :

$$\begin{bmatrix} A_1 \dots A_k \\ B_k \dots B_1 \end{bmatrix} = \begin{bmatrix} I & K_k^+ \\ K_k^- & I \end{bmatrix} \begin{bmatrix} A_1^* \dots A_{k-1}^* & 0 \\ 0 & B_{k-1}^* \dots B_1^* \end{bmatrix}$$

Calculer les E_{k+1}^+ , E_{k+1}^- :

$$\begin{bmatrix} E_{k+1}^+ \\ E_{k+1}^- \end{bmatrix} = \begin{bmatrix} I & K_k^+ \\ K_k^- & I \end{bmatrix} \begin{bmatrix} E_k^+ \\ E_k^- \end{bmatrix}$$

Calculer les u_i :

$$L_k = (v_{k+1}^* - v_{k+1}^*) (E_k^-)^{-1}$$

$$[u_1 \dots u_{k+1}] = [u_1^* \dots u_k^* \ 0] + L_k [B_k \dots B_1 \ I]$$

Corriger les c_i et d_i :

$$w_k^+ = [d_0^* \dots d_{k-1}^*] \times \text{première colonne de } R_{k+1}$$

$$w_k^- = [c_{k-1}^* \dots c_1^*] \times \text{dernière colonne de } R_{k+1}$$

$$[d_0 \dots d_k] = [0 \ d_0^* \dots d_{k-1}^*] + (X_k^T - w_k^+) (E_{k+1}^+)^{-1} [I \ A_1 \dots A_k]$$

$$[c_k \dots c_0] = [c_{k-1}^* \dots c_0^* \ 0] + (X_{T-k}^T - w_k^-) (E_{k+1}^-)^{-1} [B_k \dots B_1 \ I]$$

3. Variante de Sakai.

Le second type d'algorithmes rapide qui peut s'utiliser pour la résolution des équations de Yule-Walker du modèle non-stationnaire, est dû à SAKAI, 1982. Celui-ci l'a d'ailleurs élaboré en vue de l'identification des modèles autorégressifs périodiques dont PAGANO, 1978, avait montré qu'ils se ramenaient à un type particulier de modèle autorégressif vectoriel, ce qui constitue un cas compris dans le cadre plus général développé ici. SAKAI, 1982, considère ainsi un modèle vectoriel dont les coefficients sont répartis dans un trapèze A:

$$(2-94) \ A = \begin{bmatrix} 1 & a_1(0) & & a_k(0) & 0 & 0 \\ & \diagdown & & \diagdown & \diagdown & \diagdown \\ & & 0 & 1 & a_1(m) & \\ & & & & & a_k(m) \end{bmatrix}$$

Si on prolonge cette matrice A par un nombre de colonnes nulles tel que

le nombre total des colonnes devienne égal à un multiple du nombre des lignes, on peut considérer une décomposition en blocs de la matrice A : $A=[A_0 A_1 \dots A_p]$ qui la fait lire comme un modèle autorégressif vectoriel dont la normalisation n'est plus $A_0=I$ mais A_0 = matrice triangulaire supérieure avec des 1 sur la diagonale. Une telle normalisation est voisine de celle utilisée par MORF, VIEIRA, LEE, KAILATH, 1977, pour le modèle AR vectoriel. L'obtention des équations de Yule-Walker décrivant l'optimalité d'un tel modèle est immédiate. Si R est la matrice de corrélation du vecteur Y_t , l'optimalité s'écrit:

$$(2-95) \quad \begin{bmatrix} 1 & a_1(0) & & & & \\ & \backslash & & & & \\ & & 1 & a_1(m) & & \\ & & & \backslash & & \\ & & & & a_k(0) & 0 & 0 & 0 & 0 \\ & & & & & \backslash & & & \\ & & & & & & a_k(m) & 0 & 0 \end{bmatrix} \mathbf{R} = \begin{bmatrix} e_0^+(k) & 0 & 0 & g_0^+(k) & ? & ? & ? \\ & \backslash & & & & & \\ & & e_m^+(k) & 0 & 0 & g_m^+(k) & ? & ? \\ & & & \backslash & & & & \\ & & & & & & & \backslash & \\ & & & & & & & & g_m^+(k) & ? & ? \end{bmatrix}$$

Les "?" désignent des éléments dont la valeur est arbitraire, fixée par les $a_i(j)$, eux-mêmes fixés par les zéros du second membre. SAKAI, 1982, ne donne son algorithme que dans le cas de la méthode "de corrélation", avec R symétrique, mais la transposition au cas non-symétrique est immédiate. L'utilisation de cet algorithme pour la résolution d'un système linéaire avec un second membre s'obtient sans difficultés et va être décrite.

A l'équation précédente va être associée l'équation rétrograde, fonction de $b_i(j)$, et les vecteurs-ligne u et v tels que $u.R=v$:

$$(2-96) \quad \begin{bmatrix} b_k(0) & b_1(0) & 1 & 0 & 0 \\ & \backslash & & & \\ & & b_k(m) & b_1(m) & 1 & 0 & 0 \end{bmatrix} \mathbf{R} = \begin{bmatrix} 0 & 0 & e_0^-(k) & ? & ? \\ & \backslash & & & \\ & & ? & g_m^-(k) & 0 & 0 & e_m^-(k) & ? & ? \\ & & & & \backslash & & & & \\ & & & & & & & & \backslash & \\ & & & & & & & & & & g_m^-(k) \end{bmatrix}$$

Le système à résoudre s'écrit:

$$(2-97) \quad [u_0 \dots u_k \ 0 \dots 0] \mathbf{R} = [v_0 \dots v_k \ ? \dots ?]$$

L'algorithme repose sur la constatation que si R est une matrice de Toëplitz par blocs, ces blocs ayant $(m+1)$ lignes et colonnes, on peut effectuer une permutation circulaire sur les lignes de A , en prenant la première ligne, la faisant précéder de $(m+1)$ zéros, le second membre correspondant s'obtient en lui faisant subir la même opération: prendre la première ligne, la remettre sous les autres, décalée de $(m+1)$ éléments, vers la droite, les éléments introduits à gauche ne sont pas nuls, mais n'interviendront pas dans l'algorithme. La solution récursive sur k de l'équation s'écrit en notant $A_i(k)$ la i -ème ligne de A , et $B_i(k)$ la i -ème ligne de B :

$$(2-98) \quad A_i(k) = [1 \ a_1(i) \ \dots \ a_k(i)]$$

$$B_i(k) = [b_k(i) \ \dots \ b_1(i) \ 1]$$

Algorithme:

$$(2-99) \quad \begin{bmatrix} A_i(k+1) \\ B_i(k+1) \end{bmatrix} = \begin{bmatrix} 1 & K_i^+(k) \\ K_i^-(k) & 1 \end{bmatrix} \begin{bmatrix} A_i(k) & 0 \\ 0 & B_{i+1}(k) \end{bmatrix}$$

$$[u_0 \ \dots \ u_{k+1}] = [u_0 \ \dots \ u_k \ 0] + L_k B_i(k+1)$$

avec $K_i^+(k) = -g_1^+(k)(e_{i+1}^-(k))^{-1}$ et $K_i^-(k) = -g_{i+1}^-(k)(e_i^+(k))^{-1}$

$$L_k = (v_{k+1}^* - v_{k+1}^-)(e_{k+1}^-)^{-1}$$

Le terme $g_0^-(k)$ n'est pas défini, mais on l'obtiendra comme $g_0^-(k) = g_{m+1}^-(k)$. La mise à jour des e^+ et e^- se fait par:

$$(2-100) \quad e_i^+(k+1) = e_i^+(k) + K_i^+(k)g_{i+1}^-(k) = e_i^+(k)(1 - K_i^+(k)K_i^-(k))$$

$$e_{i+1}^-(k+1) = e_{i+1}^-(k) + K_i^-(k)g_i^+(k) = e_{i+1}^-(k)(1 - K_i^+(k)K_i^-(k))$$

Quels commentaires peut-on faire sur cet algorithme ? Son premier point fort est de ne pas faire appel à une inversion matricielle, opération

couteuse et délicate. Ici les seules inversions sont scalaires. Son coût est pas conséquent légèrement moindre que celui de l'algorithme précédent (dans sa version "corrélation" ou bloc-Toëplitz bien entendu, la version "covariance" précédente étant encore plus couteuse !). En fait son coût dominant est en $p^2 m^3$ comme pour l'algorithme du type "bloc", mais les termes annexes sont moins importants (en $p^2 m^2$, $p m^3$...). Un autre point fort de cet algorithme est qu'il est entièrement vectoriel, ce qui le rend très bien adapté à une implantation sur calculateur vectoriel (array processor).

4. Rang de déplacement, indice d'évolution.

Quelques mots en conclusion de ce chapitre, sur le concept de rang de déplacement, et son lien avec les modèles évolutifs. Les algorithmes rapides sont tous intimement liés à la structure de Toëplitz. Ce fait a été formalisé par FRIEDLANDER, KAILATH, MORF, LJUNG, 1978, 1979: on peut associer à toute matrice R un rang de déplacement noté α , défini par:

$$(2-101) \quad \alpha = \text{Rang}(R - JRJ^T)$$

$$\text{avec } J = \begin{bmatrix} 0 & 1 & & \\ & \backslash & 1 & \\ & & \backslash & 1 \\ & & & \backslash & 0 \end{bmatrix}$$

Une matrice de Toëplitz a un rang de déplacement $\alpha=2$, et pour toute matrice, $2 \leq \alpha \leq M$ si M est la dimension de la matrice. Une matrice de rang de déplacement α s'exprime comme une somme de α produits de matrices de Toëplitz triangulaires, et il en est ainsi de l'inverse de la matrice. Ceci conduisit KAILATH, 1982, à la notion d' α -stationnarité: un signal stationnaire a une matrice de covariance qui est de Toëplitz: $\alpha=2$, donc l'indice α où $\alpha-2$ étant une mesure algébrique de la distance d'une matrice à

la structure de Toëplitz, si cette matrice est la covariance d'un signal, l'indice α devient une mesure de la distance de ce signal à la stationnarité. Il faut hélas noter qu'à ce sens, un signal évolutif, c'est à dire engendré par un modèle ARMA évolutif comme ceux analysés ici, n'a aucune raison d'avoir un rang α faible, et en général $\alpha=M$, la dimension de la matrice de covariance examinée. Pour les signaux évolutifs existe par contre un indice, appelons le "indice d'évolution", qui est m , l'indice de la dernière des fonctions de la famille $f_0(t) \dots f_m(t)$. Il est lié à cette famille et n'a pas de définition intrinsèque. Il a cependant une signification qui l'apparente à l'indice α , car $m=0$ signifie la stationnarité du signal, et m croissant implique une complexité croissante dans l'évolution du signal. D'un point de vue opératoire m a aussi une signification structurale pour la covariance du signal, car il permet de construire un signal issu du précédent dont la covariance moyenne sera Toëplitz par blocs (ou proche de Toëplitz par blocs). Il semble donc que ces deux indices α et m définissent deux notions de distance à la stationnarité qui bien que semblables dans leurs conséquences, en particulier par le fait qu'ils impliquent des algorithmes rapides, ne peuvent pourtant se relier l'un à l'autre. Le fait qu'un des indices soit faible n'implique pas que l'autre le soit.

Cette coexistence de deux indices de non-stationnarité, qui ne sont d'ailleurs pas les seuls possibles, illustre bien l'ambiguïté de l'étude du phénomène de non-stationnarité. Si la stationnarité est définie de façon unique, par la convergence d'un ensemble de propriétés, il suffit par contre de relâcher l'une de ces propriétés pour définir une classe de signaux non-stationnaires, et il y a autant de non-stationnarités différentes que de propriétés possibles.