



**UN ALGORITHME RAPIDE POUR LE CALCUL DE L'ESTIMATEUR DES
MOINDRES CARRÉS D'UN MODELE AUTOREGRESSIF DE MOYENNE NON NULLE**

PHAM Dinh Tuan

Laboratoire de Modélisation et Calcul, C.N.R.S., Université de Grenoble, B.P. 53x, 38041 Grenoble Cedex

RÉSUMÉ

Ce papier fournit un algorithme général pour la résolution rapide et récursive des systèmes d'équations linéaires rencontrés dans l'estimation d'un modèle autorégressif de moyenne non nulle, par la méthode des moindres carrés et moindres carrés progressifs-rétrogrades

1. INTRODUCTION

Le modèle autorégressif (AR) joue un rôle important en traitement du signal (modélisation, prévision linéaire, estimation spectrale, ...). Toutefois, la plupart des méthodes d'estimation autorégressive ont été conçues pour un modèle centré. Si le signal a une moyenne non nulle, l'approche habituelle est de soustraire la série observée de leur moyenne arithmétique et puis procéder comme si on a affaire d'un signal centré. Or la moyenne arithmétique n'est pas le meilleur estimateur (au sens de l'erreur moyenne quadratique) de la moyenne d'un signal AR et sa performance peut être très mauvaise quand le spectre du signal a un grand pic près de l'origine et la taille de l'échantillon n'est pas grande (voir [6] où la performance de la moyenne arithmétique est comparée avec celle du meilleur estimateur linéaire sans biais de la moyenne du signal). Cette mauvaise performance peut être expliquée par le phénomène bien connu de fuite spectrale. En effet, la moyenne arithmétique est la transformation de Fourier discrète de la série observée en la fréquence zéro. Or, quand le spectre du signal a un grand pic, une part de puissance associée à celui-ci est répartie en des fréquences voisines, en particulier la fréquence zéro si le pic est proche de zéro. Dans [6], l'auteur a fourni une formule simple pour calculer le meilleur estimateur linéaire sans biais de la moyenne du signal, *quand les coefficients du modèle AR sont connus*. Cette situation n'étant pas réaliste, on adopte ici une approche globale où la moyenne et les coefficients du modèle AR sont estimés simultanément. Ceci est possible si on se sert de la méthode des moindres carrés (MC) ou des moindres carrés progressifs-rétrogrades (MCPR): Nous montrerons que les algorithmes pour la résolution des problèmes MC et MCPR, développées par Morf *et al.* [4] et Marple [3] peuvent être généralisées pour résoudre les mêmes problèmes dans le cas d'un modèle AR de moyenne non nulle. Le nouveau algorithme peut être vu comme une extension d'un algorithme général de l'auteur [5], basé sur le concept de distance de Toeplitz, semblable à la notion du rang de déplacement ([1], [2]).

ABSTRACT

This paper provides a general algorithm for solving quickly and recursively the linear systems of equations arising in the estimation for an autoregressive model with non zero mean, by the methods of least squares and forward-backward least squares.

Rappelons que le modèle AR d'ordre p de moyenne non nulle m s'écrit:

$$\sum_{i=0}^p a_i(x_{t-i} - m) = e_t, \quad (a_0 = 1),$$

où e_t est un bruit blanc (i.e. une suite de variables aléatoires indépendantes centrées identiquement distribuées). En posant $\mu = -(\sum_{i=0}^p a_i)m$, on peut réécrire ce modèle comme

$$\sum_{i=0}^p a_i x_{t-i} + \mu = e_t.$$

L'équation du modèle est alors linéaire en les paramètres a_1, \dots, a_p, μ et on peut donc les estimer par la méthode MC ou MCPR, puis en déduire l'estimateur de m par la relation $m = -\mu / (\sum_{i=0}^p a_i)$. Soient x_1, \dots, x_n un échantillon de n observations provenant du modèle, la méthode MC minimise

$$\sum_{t=p+1}^n \left(\sum_{i=0}^p a_i x_{t-i} + \mu \right)^2,$$

tandis que la méthode MCPR minimise

$$\sum_{t=p+1}^n \left(\sum_{i=0}^p a_i x_{t-i} + \mu \right)^2 + \sum_{t=n-p}^n \left(\sum_{i=0}^p a_i x_{t+i} + \mu \right)^2,$$

pour estimer les paramètres a_1, \dots, a_p et μ . Un calcul simple montre que les estimateurs MC et MCPR sont solutions du système

$$[a_0 \dots a_p \mu] \mathbf{R}_p = [v \ 0 \dots 0]$$

où



$$\mathbf{R}_p = \begin{bmatrix} x_{p+1} & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_1 & \cdots & x_{n-p} \\ 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_{p+1} & \cdots & x_1 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_n & \cdots & x_{n-p} & 1 \end{bmatrix}$$

dans le cas de la méthode MC et

$$\mathbf{R}_p = \begin{bmatrix} x_{p+1} & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_1 & \cdots & x_{n-p} \\ 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_{p+1} & \cdots & x_1 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_n & \cdots & x_{n-p} & 1 \end{bmatrix} \\ + \begin{bmatrix} x_{n-p} & \cdots & x_1 \\ \vdots & \ddots & \vdots \\ x_n & \cdots & x_{p+1} \\ 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_{n-p} & \cdots & x_n & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_1 & \cdots & x_{p+1} & 1 \end{bmatrix}$$

dans le cas de la méthode MCPR.

En pratique, l'ordre du modèle est généralement inconnu et il est intéressant d'ajuster un grand nombre de modèle afin de les comparer. Par suite, on cherche des méthodes récursives qui calculent les estimateurs d'un modèle à partir de ceux des modèles d'ordre inférieur. On considère donc une suite de matrices \mathbf{R}_m , $m = 0, 1, \dots$ définie comme précédemment avec m à la place de p . On s'intéresse à la résolution des systèmes linéaires $\mathbf{a}_m \mathbf{R}_m = [v_m \ 0 \ \dots \ 0]$, \mathbf{a}_m étant un vecteur de dimension $m+2$ dont la premier composant est égale un. Pour cela, on exploitera le fait que les matrices \mathbf{R}_m possèdent la propriété intéressante suivante. Soient \mathbf{R}_m^f et \mathbf{R}_m^b les matrices obtenues à partir de \mathbf{R}_m en supprimant l'avant dernière ligne et l'avant dernière colonne et en supprimant la première ligne et la première colonne, respectivement. Alors, il est facile d'établir les relations de récurrence:

$$\mathbf{R}_m^f = \mathbf{R}_{m-1} - \begin{bmatrix} y_m \\ \vdots \\ y_1 \\ \mathbf{u} \end{bmatrix} [y_m^* \ \dots \ y_1^* \ \mathbf{u}^*], \quad (1.2)$$

$$\mathbf{R}_m^b = \mathbf{R}_{m-1} - \begin{bmatrix} z_1 \\ \vdots \\ z_m \\ \mathbf{u} \end{bmatrix} [z_1^* \ \dots \ z_m^* \ \mathbf{u}^*], \quad (1.3)$$

où $y_i = x_i$, $z_i = x_{n+1-i}$, $\mathbf{u} = 1$ dans la méthode MC et $y_i = [x_i \ x_{n+1-i}] = z_i$, $\mathbf{u} = [1 \ 1]$ dans la méthode MCPR et $*$ désigne la transposition. Ces relations sont semblables à celles considérées dans [5] (à part le fait que les définitions de \mathbf{R}_m^f et \mathbf{R}_m^b sont différentes). L'algorithme présenté dans ce papier peut donc être généralisé pour la présente situation. Ici, on définit le rang de déplacement de \mathbf{R}_m comme le rang de $\mathbf{R}_m^b - \mathbf{R}_m^f$, qui est égale à 2 dans le cas de la méthode MC et à 4 dans le cas de la méthode MCPR.

2. FORMULATION DU PROBLEME ET NOTATIONS

Les discussions précédentes montrent que les problèmes MC et MCPR conduisent à la suite de matrices:

$$\mathbf{R}_m = \begin{bmatrix} \mathbf{R}_m(0, 0) & \cdots & \mathbf{R}_m(0, m+1) \\ \vdots & \ddots & \vdots \\ \mathbf{R}_m(m+1, 0) & \cdots & \mathbf{R}_m(m+1, m+1) \end{bmatrix}$$

et les système d'équations linéaires associé

$$\mathbf{a}_m \mathbf{R}_m = [\mathbf{a}_{0m} \ \dots \ \mathbf{a}_{m+1,m}] \mathbf{R}_m = [v_m \ 0 \ \dots \ 0]. \quad (2.1)$$

Pour plus de généralité, nous nous plaçons dans le cas où les \mathbf{R}_m sont des matrices bloc, plus précisément, les $\mathbf{R}_m(i, j)$ sont des matrices de dimensions $d \times d$ pour $(i, j) \in \{1, \dots, m\}^2$, $1 \times d$ pour $i = m+1, j \in \{1, \dots, m\}$, $d \times 1$ pour $i \in \{1, \dots, m\}, j = m+1$ et 1×1 pour $i = j = m+1$. Les $\mathbf{a}_{0m}, \dots, \mathbf{a}_{mm}$ sont alors des matrices de dimension $d \times d$, avec \mathbf{a}_{0m} la matrice identité, et $\mathbf{a}_{m+1,m}$ est un vecteur colonne de dimension d . Ce cadre général englobe le cas des modèles AR multi-canaux. D'autre part, pour traiter le cas des données complexes, nous acceptons que les matrices \mathbf{R}_m soient complexes, mais pour simplifier, nous nous limitons au cas où elles sont hermitiennes (quoique il est possible de construire un algorithme général pour le cas des matrices non hermitiennes, comme dans [5]). Nous supposons que les matrices \mathbf{R}_m satisfont les relations de récurrence (1.2) et (1.3) de l'introduction, mais avec y_i, z_i des matrices $d \times r$ quelconques et \mathbf{u} un r -vecteur ligne quelconque, non nécessairement de la forme indiquée dans cette introduction. Comme auparavant, les notations \mathbf{R}_m^f et \mathbf{R}_m^b désignent les matrices obtenues à partir de \mathbf{R}_m en supprimant les blocs $\mathbf{R}_m(i, m), \mathbf{R}_m(m, i), i = 0, \dots, m+1$, et en supprimant les blocs $\mathbf{R}_m(i, 0), \mathbf{R}_m(0, i), i = 0, \dots, m+1$, respectivement. Aussi, $*$ désigne maintenant la transposée conjuguée.

Note: Les relations (1.2) et (1.3) impliquent que les matrices \mathbf{R}_m doivent être de la forme

$$\begin{bmatrix} \mathbf{T}_0 & \cdots & \mathbf{T}_m & \mathbf{t} \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{T}_m^* & \cdots & \mathbf{T}_0 & \mathbf{t} \\ \mathbf{t}^* & \cdots & \mathbf{t}^* & t_m \end{bmatrix} - \begin{bmatrix} y_1 \cdots y_m \\ 0 \cdots \vdots \\ \vdots \cdots y_1 \\ 0 \cdots 0 \\ \mathbf{u} \cdots \mathbf{u} \end{bmatrix} \begin{bmatrix} y_1 & 0 \cdots 0 & \mathbf{u}^* \\ \vdots & \ddots & \vdots \\ y_m \cdots y_1 & 0 & \mathbf{u}^* \end{bmatrix} \\ - \begin{bmatrix} 0 \cdots 0 \\ z_1 \cdots \vdots \\ \vdots \cdots 0 \\ z_m \cdots z_1 \\ \mathbf{u} \cdots \mathbf{u} \end{bmatrix} \begin{bmatrix} 0 & z_1 \cdots z_m & \mathbf{u}^* \\ \vdots & \ddots & \vdots \\ 0 \cdots 0 & z_1 & \mathbf{u}^* \end{bmatrix},$$

où $\mathbf{T}_0, \dots, \mathbf{T}_m$ sont des matrices $d \times d$, \mathbf{t} est un d -vecteur colonne et t_m est un scalaire ne différant de $m(\mathbf{u}\mathbf{u}^*)$ que par une constante.

Notre algorithme fait intervenir non seulement les équations (2.1) mais aussi les "équations rétrogrades":

$$\mathbf{b}_m \mathbf{R}_m = [\mathbf{b}_{mm} \ \dots \ \mathbf{b}_{0m} \ \mathbf{b}_{m+1,m}] \mathbf{R}_m = [0 \ \dots \ 0 \ \mathbf{w}_m \ 0], \quad (2.2)$$

avec \mathbf{b}_{0m} la matrice identité. On aura aussi besoin des "matrices de gain":

$$\mathbf{c}_m = [\mathbf{c}_{1m} \ \dots \ \mathbf{c}_{mm} \ \mathbf{c}_{m+1,m}] = [z_1^* \ \dots \ z_m^* \ \mathbf{u}^*] \mathbf{R}_{m-1}^{-1} \quad (2.3)$$

$$\gamma_m = \mathbf{I} - \underline{\mathbf{c}}_m \begin{bmatrix} z_1 \\ \vdots \\ z_m \\ \mathbf{u} \end{bmatrix} = \mathbf{I} - [z_1^* \dots z_m^* \mathbf{u}^*] \mathbf{R}_{m-1}^{-1} \begin{bmatrix} z_1 \\ \vdots \\ z_m \\ \mathbf{u} \end{bmatrix} \quad (2.4)$$

et

$$\underline{\mathbf{d}}_m = [\underline{\mathbf{d}}_{mm} \dots \underline{\mathbf{d}}_{1m} \underline{\mathbf{d}}_{m+1,m}] = [y_m^* \dots y_1^* \mathbf{u}^*] \mathbf{R}_{m-1}^{-1} \quad (2.5)$$

$$\delta_m = \mathbf{I} - \underline{\mathbf{d}}_m \begin{bmatrix} y_m \\ \vdots \\ y_1 \\ \mathbf{u} \end{bmatrix} = \mathbf{I} - [y_m^* \dots y_1^* \mathbf{u}^*] \mathbf{R}_{m-1}^{-1} \begin{bmatrix} y_m \\ \vdots \\ y_1 \\ \mathbf{u} \end{bmatrix}, \quad (2.6)$$

\mathbf{I} désignant la matrice identité. On définit les matrices $\tilde{\mathbf{a}}_m, \tilde{\mathbf{v}}_m, \tilde{\mathbf{c}}_m$ et $\tilde{\gamma}_m$ de la même façon avec \mathbf{R}_m et \mathbf{R}_{m-1} remplacées par \mathbf{R}_{m+1}^f et \mathbf{R}_m^f . Les matrices $\tilde{\mathbf{h}}_m, \tilde{\mathbf{w}}_m, \tilde{\mathbf{d}}_m$ et $\tilde{\delta}_m$ sont définies similairement, à partir de \mathbf{R}_{m+1}^b et \mathbf{R}_m^b .

3. ALGORITHME

Il est composé de quatre groupes de relations de récurrence ayant une structure semblable:

Récurrence "en temps" pour les gains:

$$\tilde{\mathbf{c}}_m = \underline{\mathbf{c}}_m + \mathbf{h}_m \delta_m^{-1} \underline{\mathbf{d}}_m, \quad \tilde{\gamma}_m = \gamma_m - \mathbf{h}_m \delta_m^{-1} \mathbf{h}_m^* \quad (3.1)$$

$$\tilde{\underline{\mathbf{d}}}_m = \underline{\mathbf{d}}_m + \mathbf{h}_m^* \gamma_m^{-1} \underline{\mathbf{c}}_m, \quad \tilde{\delta}_m = \delta_m - \mathbf{h}_m^* \gamma_m^{-1} \mathbf{h}_m \quad (3.2)$$

où

$$\begin{aligned} \mathbf{h}_m &= [z_1^* \dots z_m^* \mathbf{u}^*] \underline{\mathbf{d}}_m^* \\ &= [z_1^* \dots z_m^* \mathbf{u}^*] \mathbf{R}_{m-1}^{-1} \begin{bmatrix} y_m \\ \vdots \\ y_1 \\ \mathbf{u} \end{bmatrix} = \underline{\mathbf{c}}_m \begin{bmatrix} y_m \\ \vdots \\ y_1 \\ \mathbf{u} \end{bmatrix}. \end{aligned}$$

Récurrence "en temps" pour les coefficients AR progressifs et "en ordre" pour les gains:

$$\tilde{\mathbf{a}}_m = \mathbf{a}_m + \mathbf{e}_m \tilde{\delta}_m^{-1} [0 \ \tilde{\underline{\mathbf{d}}}_m], \quad \tilde{\mathbf{v}}_m = \mathbf{v}_m - \mathbf{e}_m \tilde{\delta}_m^{-1} \mathbf{e}_m^* \quad (3.3)$$

$$\underline{\mathbf{d}}_{m+1} = [0 \ \tilde{\underline{\mathbf{d}}}_m] + \mathbf{e}_m^* \mathbf{v}_m^{-1} \mathbf{a}_m, \quad \delta_{m+1} = \tilde{\delta}_m - \mathbf{e}_m^* \mathbf{v}_m^{-1} \mathbf{e}_m \quad (3.4)$$

où

$$\mathbf{e}_m = \mathbf{a}_m [y_{m+1}^* \dots y_1^* \mathbf{u}^*]^*.$$

Récurrence "en temps" pour les coefficients AR rétrogrades et "en ordre" pour les gains:

$$\tilde{\mathbf{h}}_m = \underline{\mathbf{h}}_m + \mathbf{f}_m \tilde{\gamma}_m^{-1} \tilde{\underline{\mathbf{c}}}_m, \quad \tilde{\mathbf{w}}_m = \mathbf{w}_m - \mathbf{f}_m \tilde{\gamma}_m^{-1} \mathbf{f}_m^* \quad (3.5)$$

$$\underline{\mathbf{c}}_{m+1} = \tilde{\underline{\mathbf{c}}}_m + \mathbf{f}_m^* \mathbf{w}_m^{-1} \underline{\mathbf{h}}_m, \quad \gamma_{m+1} = \tilde{\gamma}_m - \mathbf{f}_m^* \mathbf{w}_m^{-1} \mathbf{f}_m \quad (3.6)$$

où $\tilde{\underline{\mathbf{c}}}_m = [\tilde{\mathbf{c}}_{1m} \dots \tilde{\mathbf{c}}_{mm} \ \mathbf{0}^t \ \tilde{\mathbf{c}}_{m+1,m}]$ et

$$\mathbf{f}_m = \underline{\mathbf{h}}_m [z_1^* \dots z_m^* \mathbf{u}^*]^*.$$

Récurrence "en ordre" pour les coefficients AR

$$\underline{\mathbf{a}}_{m+1} = \tilde{\underline{\mathbf{a}}}_m - \pi_{m+1} \tilde{\mathbf{w}}_m^{-1} [0 \ \tilde{\underline{\mathbf{h}}}_m], \quad \mathbf{v}_{m+1} = \tilde{\mathbf{v}}_m - \pi_{m+1} \tilde{\mathbf{w}}_m^{-1} \pi_{m+1}^* \quad (3.7)$$

$$\underline{\mathbf{h}}_{m+1} = \tilde{\underline{\mathbf{h}}}_m - \pi_{m+1}^* \tilde{\mathbf{v}}_m^{-1} \tilde{\underline{\mathbf{a}}}_m, \quad \mathbf{w}_{m+1} = \tilde{\mathbf{w}}_m - \pi_{m+1}^* \tilde{\mathbf{v}}_m^{-1} \pi_{m+1} \quad (3.8)$$

où $\tilde{\underline{\mathbf{a}}}_m = [\tilde{\mathbf{a}}_{0m} \dots \tilde{\mathbf{a}}_{mm} \ \mathbf{0} \ \tilde{\mathbf{a}}_{m+1,m}]$ et

$$\pi_{m+1} = \tilde{\underline{\mathbf{a}}}_m \mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{I} \\ 0 \end{bmatrix} = [\mathbf{I} \ \mathbf{0} \dots \mathbf{0}] \mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \vdots \\ \tilde{\underline{\mathbf{h}}}_m^* \end{bmatrix}.$$

Cette étape (et seulement celle-ci) nécessite la connaissance de $\mathbf{R}_{m+1}(0, m+1), \dots, \mathbf{R}_{m+1}(m, m+1), \mathbf{R}_{m+1}(m+2, m+1)$ ou de $\mathbf{R}_{m+1}(0, 1), \dots, \mathbf{R}_{m+1}(0, m+2)$. Ces matrices, excepté $\mathbf{R}_{m+1}(0, m+1)$, peuvent être calculées récursivement par (1.2) ou (1.3).

La preuve de l'algorithme est donnée dans l'appendice. Notons que les relations de récurrence (3.1) et (3.2) ne sont définies que pour $m > 0$. Par suite l'algorithme doit être initialisé au pas $m = 0$ à partir de (3.3), avec les valeurs initiales: $\tilde{\mathbf{c}}_{10} = \mathbf{u}^* \mathbf{R}_0(1, 1)^{-1}$, $\tilde{\underline{\mathbf{d}}}_{10} = \tilde{\mathbf{c}}_{10}$, $\tilde{\gamma}_0 = \mathbf{I} - \mathbf{u}^* \mathbf{R}_0(1, 1)^{-1} \mathbf{u}$, $\tilde{\delta}_0 = \tilde{\gamma}_0$ et $(\underline{\mathbf{a}}_0 \ \mathbf{v}_0)$ la solution de $\underline{\mathbf{a}}_0 \mathbf{R}_0 = (\mathbf{v}_0 \ \mathbf{0})$ et $(\underline{\mathbf{h}}_0 \ \mathbf{w}_0) = (\underline{\mathbf{a}}_0 \ \mathbf{v}_0)$. Dans le cas où $\mathbf{R}_0(1, 1)$ de la forme $n\mathbf{u}\mathbf{u}^*$ comme ce qui est souvent le cas, il est facile de calculer explicitement les quantités intervenues dans l'algorithme pour $m = 0$ et donc démarre celui-ci en $m = 1$. On a :

$$\tilde{\mathbf{c}}_{10} = \mathbf{u}^*/(n\mathbf{u}\mathbf{u}^*), \quad \tilde{\gamma}_0 = \mathbf{I} - \mathbf{u}^*\mathbf{u}/(n\mathbf{u}\mathbf{u}^*),$$

$$\mathbf{a}_{10} = \mathbf{R}_0(1,0)/(n\mathbf{u}\mathbf{u}^*), \quad \mathbf{v}_0 = \mathbf{R}_0(0,0) - \mathbf{a}_{10}\mathbf{R}_0(0,1),$$

$$\mathbf{c}_{11} = \mathbf{f}_0^* \mathbf{v}_0^{-1}, \quad \mathbf{c}_{21} = \tilde{\mathbf{c}}_{10} + \mathbf{c}_{11} \mathbf{a}_{10}, \quad \gamma_1 = \tilde{\gamma}_0 - \mathbf{c}_{11} \mathbf{f}_0,$$

$$\mathbf{d}_{11} = \mathbf{e}_0^* \mathbf{v}_0^{-1}, \quad \mathbf{d}_{21} = \tilde{\mathbf{c}}_{10} + \mathbf{d}_{11} \mathbf{a}_{10}, \quad \delta_1 = \tilde{\delta}_0 - \mathbf{d}_{11} \mathbf{e}_0,$$

$$\tilde{\mathbf{a}}_{10} = \mathbf{a}_{10} + \mathbf{e}_0 \mathbf{u}^*/[(n-1)\mathbf{u}\mathbf{u}^*], \quad \tilde{\mathbf{v}}_0 = \mathbf{v}_0 - \tilde{\mathbf{e}}_0 \mathbf{e}_0^*,$$

$$\tilde{\mathbf{h}}_{10} = \mathbf{a}_{10} + \mathbf{f}_0 \mathbf{u}^*/[(n-1)\mathbf{u}\mathbf{u}^*], \quad \tilde{\mathbf{w}}_0 = \mathbf{w}_0 - \tilde{\mathbf{f}}_0 \mathbf{f}_0^*$$

où

$$\mathbf{f}_0 = \mathbf{z}_1 + \mathbf{a}_{10} \mathbf{u}, \quad \mathbf{e}_0 = \mathbf{y}_1 + \mathbf{a}_{10} \mathbf{u}$$

$$\tilde{\mathbf{f}}_0 = \mathbf{z}_1 + \tilde{\mathbf{h}}_{10} \mathbf{u}, \quad \tilde{\mathbf{e}}_0 = \mathbf{y}_1 + \tilde{\mathbf{a}}_{10} \mathbf{u},$$

et enfin

$$\mathbf{a}_{11} = -\pi_1 \tilde{\mathbf{w}}_0^{-1}, \quad \mathbf{a}_{21} = \mathbf{a}_{11} \tilde{\mathbf{h}}_{10}, \quad \mathbf{v}_1 = \tilde{\mathbf{v}}_0 + \mathbf{a}_{11} \pi_1^*,$$

$$\mathbf{b}_{11} = -\pi_1^* \tilde{\mathbf{v}}_0^{-1}, \quad \mathbf{b}_{21} = \mathbf{b}_{11} \tilde{\mathbf{a}}_{10}, \quad \mathbf{w}_1 = \tilde{\mathbf{w}}_0 + \mathbf{b}_{11} \pi_1.$$

Notons que dans le cas du problème MCPR scalaire les y_i et z_i sont les mêmes (ils sont égales à $[x_i \ x_{n+1-i}]$). On peut alors montrer (par induction) que $\mathbf{R}_m = \mathbf{J}_m \mathbf{R}_m \mathbf{J}_m$ où \mathbf{J}_m est la matrice $(m+1) \times (m+1)$ avec des 1 aux positions $(m+1, m+1), (0, m), (1, m-1), \dots, (m, 0)$ et des 0 ailleurs. Il s'ensuit que $\mathbf{a}_m = \underline{\mathbf{h}}_m \mathbf{J}_m$, $\mathbf{v}_m = \mathbf{w}_m$, $\underline{\mathbf{c}}_m = \underline{\mathbf{d}}_m \mathbf{J}_{m-1}$, $\gamma_m = \delta_m$, ainsi que des relations semblables entre $\tilde{\underline{\mathbf{a}}}_m, \tilde{\mathbf{v}}_m$ et $\tilde{\underline{\mathbf{h}}}_m, \tilde{\mathbf{w}}_m$ et entre $\tilde{\mathbf{c}}_m, \tilde{\gamma}_m$ et $\tilde{\underline{\mathbf{d}}}_m, \tilde{\delta}_m$. Par suite, on a besoin seulement de mettre à jours les $\underline{\mathbf{a}}_m, \mathbf{v}_m, \underline{\mathbf{c}}_m$ et γ_m .

4. QUELQUES EXEMPLES DE SIMULATION

Nous avons appliqué notre algorithme sur des données simulées. Deux modèles AR, d'ordre 2 et 4 respectivement, sont considérés. Pour chaque modèle, nous générons 2500 séries indépendantes de longueur 30 chacune et nous calculons les estimateurs de la moyenne et des coefficients AR par trois méthodes différentes. Dans la méthode 0, la moyenne est *supposée connue* (cette méthode est utilisée comme une référence), dans la méthode 1, la moyenne est estimé par la moyenne arithmétique et dans la méthode 2, la moyenne et les



paramètres AR sont estimés simultanément. Dans tous les cas, le critère MCPR est utilisé. Les biais et les erreurs moyenne quadratiques des estimateurs sont listés dans la table 1. On peut voir dans cette table que la méthode 2 produit des estimateurs nettement meilleurs que la méthode 1 en ce qui concerne la moyenne et marginalement meilleur en ce qui concerne les paramètres autorégressifs. Cela peut être expliqué par le fait que les estimateurs des paramètres AR semblent être peu influencés par les erreurs commises sur l'estimation de la moyenne de la série: même quand on soustrait la série observée de sa vraie moyenne (méthode 0), le gain de performance est faible.

Table 1: Biais (première ligne) et erreur moyenne quadratique (deuxième ligne) des estimateurs deux modèles AR (séries de taille 30, 2500 répétitions).

Paramètres	Méthode 0	Méthode 1	Méthode 2
$a_1 = -1.8831$	0.0251	0.0290	0.0270
	0.05964	0.06362	0.06276
$a_2 = 0.9801$	-0.0245	-0.0247	-0.0228
	0.05814	0.05641	0.05569
moyenne	0	0.0433	-0.0148
	0	3.7437	1.9785
$a_1 = -2.7607$	0.0600	0.0346	0.0664
	0.12461	0.11397	0.13154
$a_2 = 3.8106$	-0.1551	-0.1459	-0.1581
	0.28739	0.28078	0.29096
$a_3 = -2.6535$	0.1564	0.1693	0.1543
	0.28667	0.29754	0.28476
$a_4 = 0.9238$	-0.0685	-0.0952	-0.0625
	0.12563	0.14919	0.11945
moyenne	0	-0.0029	0.0019
	0	1.9151	0.6153

APPENDICE: PREUVE DE L'ALGORITHME

D'après une formule bien connue d'inversion de matrice (voir par ex. [3], p. 68), on obtient à partir de (1.2) et (2.5), (2.6)

$$(\mathbf{R}_m^f)^{-1} = \mathbf{R}_{m-1}^{-1} + \mathbf{d}_m^* \delta_m^{-1} \mathbf{d}_m.$$

On en déduit alors facilement (3.1). Les relations (3.2) se montrent de la même façon, en se servant de (1.3) et (2.3), (2.4). D'autre part, à partir de la définition (2.1) de \mathbf{a}_m et $\tilde{\mathbf{a}}_m$ et de (1.2)

$$(\tilde{\mathbf{a}}_m - \mathbf{a}_m) \mathbf{R}_{m+1}^f = [(\mathbf{v}_m - \tilde{\mathbf{v}}_m) \mathbf{0} \dots \mathbf{0}] + \mathbf{e}_m [y_{m+1}^* \dots y_1^* \mathbf{u}^*].$$

Cette équation donne

$$[(\mathbf{a}_{1m} - \tilde{\mathbf{a}}_{1m}) \dots (\mathbf{a}_{mm} - \tilde{\mathbf{a}}_{mm})] \mathbf{R}_{m+1}^{fb} = \mathbf{e}_m [y_m^* \dots y_1^* \mathbf{u}^*],$$

$$(\tilde{\mathbf{a}}_m - \mathbf{a}_m) \mathbf{R}_{m+1}^f \tilde{\mathbf{a}}_m^* = \mathbf{v}_m - \tilde{\mathbf{v}}_m + \mathbf{e}_m [y_{m+1}^* \dots y_1^* \mathbf{u}^*] \tilde{\mathbf{a}}_m^*.$$

où \mathbf{R}_{m+1}^{fb} désigne la matrice obtenue à partir de \mathbf{R}_{m+1}^f en enlevant la première bloc-ligne et la première bloc-colonne. Mais, en se servant encore de (1.2) et la formule d'inversion citée plus haut $(\mathbf{R}_{m+1}^{fb})^{-1} = (\mathbf{R}_m^b)^{-1} + \tilde{\mathbf{d}}_m^* \tilde{\delta}_m^{-1} \tilde{\mathbf{d}}_m$ et donc la première des égalités précédente donne $\tilde{\mathbf{a}}_m = \mathbf{a}_m + \mathbf{e}_m \tilde{\delta}_m^{-1} [\mathbf{0} \quad \tilde{\mathbf{d}}_m]$, qui est la première partie de (3.3). En rapportant cette valeur de $\tilde{\mathbf{a}}_m$ dans l'autre égalité, on obtient

$$\mathbf{v}_m - \tilde{\mathbf{v}}_m + \mathbf{e}_m \mathbf{e}_m^* + \mathbf{e}_m [y_m^* \dots y_1^* \mathbf{u}^*] \tilde{\mathbf{d}}_m^* \tilde{\delta}_m^{-1} \mathbf{e}_m^* = 0.$$

Ce qui prouve la seconde partie de (3.3). Les relations (3.5) se démontrent de la même façon.

Les relations (3.4) sont obtenues de l'égalité

$$\mathbf{R}_m^{-1} = \mathbf{a}_m^* \mathbf{v}_m^{-1} \mathbf{a}_m + \begin{bmatrix} 0 & 0 \\ 0 & (\mathbf{R}_m^b)^{-1} \end{bmatrix},$$

qui peut être établie à partir de la définition (2.1) de \mathbf{a}_m . De la même façon, on montre (3.6). Enfin, on peut vérifier directement que les matrices \mathbf{a}_{m+1} , \mathbf{v}_{m+1} et \mathbf{b}_{m+1} , \mathbf{w}_{m+1} , telles qu'elles sont définies par les membres de droite de (3.7) et (3.8), satisfont les équations (2.1) et (2.2).

REFERENCES

[1] FRIEDLANDER, B., MORF, M., KAILATH, T. LJUNG, L. (1979) New inversion formula for matrices classified in terms of their distances to Toeplitz matrices. *Linear Algebra & Appl.* **27**, 31-60.
 [2] KAILATH, T., KUNG, S. Y., MORF, M. (1979) Displacement rank of a matrix. *Bull. Amer. Math. Soc.* **1**, 5, 769-773.
 [3] MARPLE, L. (1987) *Digital Spectral Analysis with Applications*. New-Jersey: Prentice Hall.
 [4] MORF, M., DICKINSON, B., KAILATH, T., VIERA, A. (1977) Efficient solution of covariance equation for linear prediction. *IEEE Trans. Acoust. Speech Signal Processing* **25**, 5, 429-433.
 [5] PHAM, D. T. (1991) Quick solution of least squares equations and inversion of block matrices of low displacement rank. *IEEE Trans. on Signal Processing* **39**, 9, 2122-2124.
 [6] PHAM, D. T., TRAN, L. T. (1992) A note on the best unbiased estimate for the mean of a short autoregressive time series. *J. Econometric Theory* **8**, 1, 120-126.