



**ALGORITHMES DE MOINDRES CARRÉS RAPIDES A BASE  
DE ROTATIONS POUR LE FILTRAGE NUMERIQUE  
ADAPTATIF**

*M. Terré \**, *M. Bellanger \*\**

\* Thomson CSF / CNI, 46 quai Alphonse Le Gallo, BP 407, 92103 Boulogne Billancourt, France.

\*\* Conservatoire National des Arts et Métiers, 292 rue St-Martin, 75141 Paris Cedex 03, France.

RÉSUMÉ

ABSTRACT

Cet article rappelle d'abord l'algorithme des moindres carrés basé sur la décomposition QR de la matrice des échantillons du signal. Il souligne les problèmes inhérents à l'intégration directe de cet algorithme sur Silicium et propose une modification de ce dernier afin de le rendre "systolique". Le nouvel algorithme obtenu est alors très proche du treillis normalisé et l'article développe les équivalences entre les variables propagées par ces deux algorithmes.

This article reviews the fast least squares algorithm based on the technique of QR decomposition. It points out problems of direct implementation and proposes a modification of the algorithm in order to give a systolic version. This new algorithm is very close to the normalized lattice and the article emphasizes equivalencies between variables propagated by the two algorithms.

**I. INTRODUCTION**

Les systèmes adaptatifs utilisés dans les terminaux de télécommunications font généralement appel à des algorithmes simples du type gradient. C'est le cas notamment des égaliseurs de canaux, des annuleurs d'écho ou encore des dispositifs de prédiction dans la compression de signaux. Avec l'accroissement des moyens de calcul, des méthodes plus évoluées sont envisagées et de nombreux travaux ont été consacrés à la conception d'algorithmes du type moindres carrés, pour lesquels des versions rapides, c'est à dire avec un nombre d'opérations proportionnel au nombre N de coefficients du filtre, ont été élaborées [1,2].

Les algorithmes des moindres carrés rapides sont liés à la structure des filtres adaptatifs et il faut distinguer la structure transversale associée à un algorithme global c'est à dire qui fournit directement la sortie du filtre à N coefficients, et les structures modulaires où l'algorithme fournit les sorties de tous les filtres adaptatifs d'ordre 1 à N. Les structures modulaires sont évidemment plus compliquées, mais elles présentent des avantages qui peuvent être déterminants, en particulier elles sont plus faciles à maîtriser car elles peuvent être numériquement stables et peuvent conduire à une dynamique réduite des variables internes, ce qui permet leur mise en oeuvre sur des machines de précision limitée.

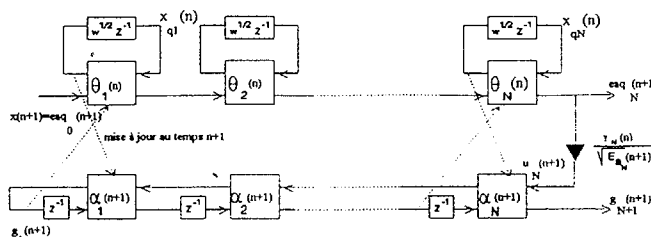
Deux types de structures modulaires ont été plus particulièrement étudiées, les structures en treillis et les structures à rotations, qui peuvent toutes les deux être normalisées, c'est à dire mises en oeuvre sans élévation au carré des variables donc avec une dynamique de ces variables qui est celle des signaux d'entrée et non le double.

Les démarches qui conduisent à ces deux types de structures apparaissent souvent comme différentes, un objectif du présent article est de montrer qu'il existe une équivalence entre les variables utilisées dans les deux cas et qu'en fait, l'approche à rotations est la plus générale, le treillis n'en étant qu'un cas particulier.

L'article est organisé de la manière suivante. L'algorithme des moindres carrés rapides est d'abord rappelé. Ensuite, il est montré que ce dernier peut être modifié pour être réalisable par une structure systolique. Enfin, un certain nombre de changements de variables qui permettent d'obtenir l'algorithme du treillis normalisé, sont exposés.

**II. ALGORITHME DES MOINDRES CARRÉS  
RAPIDES PAR DECOMPOSITION QR**

La propagation des erreurs de prédiction à travers l'algorithme des moindres carrés rapides par décomposition QR [3] est présentée sur la figure 1.



**figure 1: Propagation des erreurs de prédiction à travers l'algorithme des moindres carrés récursifs rapides QR**



Les opérations de rotation de la figure 1 sont représentées avec la convention ci après (fig 2):

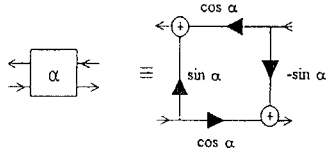


figure 2: Convention pour la représentation des rotations

Les relations de l'algorithme sont rappelées table 1.

Prédiction:	
$\begin{bmatrix} e_{aqN}(n+1) \\ X_{qN}(n+1) \end{bmatrix}$	$= Q_{\alpha}(n) \begin{bmatrix} x(n+1) \\ w^{1/2} X_{qN}(n) \end{bmatrix}$
$E_{aN}(n+1)$	$= wE_{aN}(n) + e_{aqN}^2(n+1)$
$\begin{bmatrix} 0 \\ \sqrt{E_{b0}(n+1)} \end{bmatrix}$	$= Q_{\alpha}(n+1) \begin{bmatrix} X_{qN}(n+1) \\ \sqrt{E_{aN}(n+1)} \end{bmatrix}$
$u_N(n+1)$	$= \gamma_N(n) \frac{e_{aqN}(n+1)}{\sqrt{E_{aN}(n+1)}}$
$\begin{bmatrix} g_{N+1}(n+1) \\ G_N(n+1) \end{bmatrix}$	$= Q_{\alpha}(n+1) \begin{bmatrix} G_N(n) \\ u_N(n+1) \end{bmatrix}$
$\begin{bmatrix} \gamma_N(n+1) \\ G_N(n+1) \end{bmatrix}$	$= Q_{\alpha}(n+1) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$
Filtrage:	
$\begin{bmatrix} e_{qN}(n+1) \\ Y_{qN}(n+1) \end{bmatrix}$	$= Q_{\alpha}(n+1) \begin{bmatrix} y(n+1) \\ w^{1/2} Y_{qN}(n) \end{bmatrix}$
$\varepsilon_N(n+1)$	$= \gamma_N(n+1) e_{qN}(n+1)$

table 1: Relations de l'algorithme des moindres carrés récursifs rapides QR

En analysant la figure 1 il apparaît que la mise à jour au temps  $n+1$  des rotations  $\theta_i(n)$  ne peut se faire qu'après l'obtention des termes  $g_i(n+1)$ . Or ceux ci dépendent de la mise à jour des rotations  $\alpha_i(n+1)$  et cette mise à jour s'effectue, dans cet algorithme, dans le sens des ordres décroissants. Ainsi, c'est la rotation d'angle  $\alpha_N$  qui est mise à jour en premier puis il en découle la mise à jour de la rotation  $\alpha_{N-1}$  et ainsi de suite. Or la mise à jour de la rotation  $\alpha_N(n+1)$  ne peut elle même être réalisée qu'une fois la rotation  $\theta_N(n)$  effectuée. Il en découle qu'il est nécessaire d'effectuer toutes les rotations  $\theta_i(n)$  et  $\alpha_i(n+1)$  pour tous les ordres avant de pouvoir mettre à jour la rotation  $\theta_1(n+1)$  et commencer l'étape de prédiction pour un échantillon du temps  $n+2$ :  $e_{aq0}(n+2)$ .

Si l'algorithme est réalisé par un processeur séquentiel traditionnel, cette particularité est sans importance. Par contre, si une implantation directe de l'algorithme sur Silicium est envisagée, ce phénomène interdit l'alimentation de l'algorithme par un flot de données continue, où le matériel dédié aux rotations d'ordre  $i$  pourrait travailler sur des données d'indice temporel  $n+1$  tandis que le matériel dédié aux ordres  $i+1$  travaillerait sur des données d'indice  $n$ . L'algorithme est alors dit non systolique et son implantation, fortement sous optimale, n'est guère envisageable.

La suite de cet article propose une modification de l'algorithme afin de résoudre ce problème d'organisation des

récurrences et d'aboutir ainsi à un nouvel algorithme que l'on peut qualifier d'algorithme des moindres carrés récursifs rapides QR systolique.

### III. SYSTOLISATION DE L'ALGORITHME DES MOINDRES CARRÉS RECURSIFS RAPIDES QR

La systolisation proposée ici étant dictée par des objectifs d'implantation directe de l'algorithme sur un ASIC, elle va aussi intégrer un étape de normalisation qui permet de contrôler la dynamique des variables internes et d'optimiser leur représentation sur une arithmétique à virgule fixe.

Ainsi, en normalisant les échantillons en entrée par leur énergie:  $\sqrt{E_{b0}(n+1)}$ , l'équation de mise à jour des rotations  $\alpha(n+1)$ :

$$\begin{bmatrix} 0 \\ \sqrt{E_{b0}(n+1)} \end{bmatrix} = Q_{\alpha}(n+1) \begin{bmatrix} X_{qN}(n+1) \\ \sqrt{E_{aN}(n+1)} \end{bmatrix}$$

va devenir:

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} = Q_{\alpha}(n+1) \begin{bmatrix} X_{qN}(n+1) \\ \sqrt{E_{aN}(n+1)} \end{bmatrix}$$

$X_{qN}$  et  $E_{aN}$  correspondant alors aux données normalisées.

Sa résolution sous forme inverse:

$$Q_{\alpha}^{-1}(n+1) \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} X_{qN}(n+1) \\ \sqrt{E_{aN}(n+1)} \end{bmatrix} \quad (1)$$

permet de mettre à jour les angles  $\alpha_i(n+1)$  dans le sens des ordres croissants. Il en découle une mise à jour ordonnée croissante des composantes  $g_i(n+1)$  du vecteur  $G_N(n+1)$  à travers la relation:

$$Q_{\alpha}(n+1) \begin{bmatrix} G_i(n) \\ u_{i+1}(n+1) \end{bmatrix} = \begin{bmatrix} g_{i+1}(n+1) \\ G_i(n+1) \end{bmatrix}$$

avec:

$$u_{i+1}(n+1) = \gamma_{i+1}(n) \frac{e_{aq(i+1)}(n+1)}{\sqrt{E_{a(i+1)}(n+1)}} \quad (2)$$

Les problèmes d'ordre des itérations rencontrés avec la version de base de l'algorithme sont alors résolus et cette nouvelle version est systolique.

La normalisation d'entrée peut être formalisée par l'introduction d'une rotation d'angle  $\psi$ :

$$\begin{bmatrix} 0 \\ \sqrt{E_{b0}(n+1)} \end{bmatrix} = \begin{bmatrix} \cos \psi(n+1) & -\sin \psi(n+1) \\ \sin \psi(n+1) & \cos \psi(n+1) \end{bmatrix} \begin{bmatrix} x(n+1) \\ \sqrt{wE_{b0}(n)} \end{bmatrix} \quad (3)$$

L'écriture de la conservation de la norme par rotation conduit alors à:

$$\sin \psi(n+1) = \frac{x(n+1)}{\sqrt{E_{b0}(n+1)}} \quad \text{et} \quad \cos \psi(n+1) = \frac{\sqrt{wE_{b0}(n)}}{\sqrt{E_{b0}(n+1)}}$$

Dès lors la normalisation de l'algorithme consiste simplement à remplacer  $x(n+1)$  par  $\sin \psi(n+1)$  et  $\sqrt{w}$  par  $\cos \psi(n+1)$ .

Les équations de l'algorithme systolique sont présentées table 2.

Normalisation :

$$E_{b0}(n+1) = wE_{b0}(n) + x^2(n+1)$$

$$\sin \psi(n+1) = \frac{x(n+1)}{\sqrt{E_{b0}(n+1)}}, \quad \cos \psi(n+1) = \frac{wE_{b0}(n)}{\sqrt{E_{b0}(n+1)}}$$

Initialisation :

$$\gamma_0(n+1) = 1, \quad E_{a0}(n+1) = 1$$

$$e_{aq0}(n+1) = \sin \psi(n+1), \quad g_1(n+1) = \sin \psi(n+1), \quad e_{q0}(n+1) = y(n+1)$$

Prédiction :

$$i = 0 \rightarrow N:$$

$$\begin{cases} e_{aq(i+1)}(n+1) = \cos \theta_{i+1}(n) e_{aqi}(n+1) - \sin \theta_{i+1}(n) \cos \psi(n+1) x_{q(i+1)}(n) & (4) \\ x_{q(i+1)}(n+1) = \sin \theta_{i+1}(n) e_{aqi}(n+1) + \cos \theta_{i+1}(n) \cos \psi(n+1) x_{q(i+1)}(n) & (5) \\ E_{a(i+1)}(n+1) = E_{ai}(n+1) - x_{q(i+1)}^2(n+1) \\ \sin \alpha_{i+1}(n+1) = \frac{x_{q(i+1)}(n+1)}{\sqrt{E_{ai}(n+1)}}, \quad \cos \alpha_{i+1}(n+1) = \sqrt{1 - \sin^2 \alpha_{i+1}(n+1)} \\ u_{(i+1)}(n+1) = \gamma_{(i+1)}(n) \frac{e_{aq(i+1)}(n+1)}{\sqrt{E_{a(i+1)}(n+1)}} \\ g_{i+2}(n+1) = \cos \alpha_{i+1}(n+1) g_{i+1}(n) - \sin \alpha_{i+1}(n+1) u_{i+1}(n+1) & (6) \\ \sin \theta_{i+1}(n+1) = \frac{g_{i+1}(n+1)}{\gamma_i(n+1)}, \quad \cos \theta_{i+1}(n+1) = \sqrt{1 - s^2 \theta_{i+1}(n+1)} \\ \gamma_{i+1}(n+1) = \gamma_i(n+1) \cos \theta_{i+1}(n+1) \end{cases}$$

Filtrage

$$i = 0 \rightarrow N:$$

$$\begin{cases} e_{q(i+1)}(n+1) = \cos \theta_{i+1}(n+1) e_{qi}(n+1) - \sin \theta_{i+1}(n+1) w^{1/2} y_{q(i+1)}(n) & (7) \\ y_{q(i+1)}(n+1) = \sin \theta_{i+1}(n+1) e_{qi}(n+1) + \cos \theta_{i+1}(n+1) w^{1/2} y_{q(i+1)}(n) & (8) \end{cases}$$

$$e_N(n+1), \quad \gamma_N(n+1), \quad e_{qN}(n+1)$$

table 2: Equations de l'algorithme des moindres carrés récursifs rapides QR systolique

La figure 3 représente la propagation des erreurs de prédiction avec le nouvel algorithme.

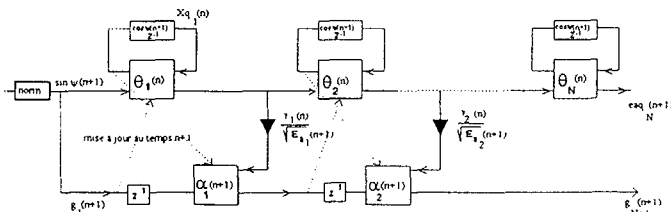


figure 3: Propagation des erreurs de prédiction dans l'algorithme des moindres carrés récursifs rapides QR systolique

La boucle de retour de la rotation  $\alpha_N$  vers la rotation  $\alpha_1$ , présente dans la version de base de l'algorithme (cf fig 1) et responsable du caractère non systolique de cette version, a bien disparu dans cette nouvelle structure. Au regard de cette dernière, il est immédiat que rien n'interdit aux rotations d'indice  $i$  ( $\theta_i$  et  $\alpha_i$ ) de traiter une donnée d'indice temporel  $n+1$  tandis qu'en parallèle, les rotations d'indice  $i+1$  traitent des échantillons du temps  $n$ . Ce nouvel algorithme est donc constitué d'une répétition de cellules élémentaires ( $\theta_i, \alpha_i$ ) pouvant fonctionner en parallèle, il est donc bien systolique.

Cette nouvelle structure fait apparaître deux voies d'erreurs ( $e_{aqi}, g_{i+1}$ ), des cellules de retard ( $Z^{-1}$ ) sur une des voies et des liens croisés entre ces deux voies. Enfin, l'aspect modulaire de l'algorithme a déjà été souligné. Tous ces caractères évoquent fortement les structures d'algorithmes en treillis et le paragraphe suivant va mettre à jour les équivalences entre les variables propagées par

l'algorithme QR rapide systolique et celles propagées par l'algorithme du treillis normalisé.

#### IV. EQUIVALENCE ENTRE L'ALGORITHME DES MOINDRES CARRÉS RECURSIFS RAPIDES QR SYSTOLIQUE ET L'ALGORITHME DU TREILLIS NORMALISE

En effectuant quelques changements de variables dans les itérations de l'algorithme QR rapide systolique obtenu précédemment (table 2), il est possible de regrouper différemment les itérations et de constater qu'on aboutit alors rigoureusement à l'algorithme du treillis normalisé [4].

##### Propagation de l'erreur de prédiction avant:

Ainsi, en extrayant  $\cos \psi(n+1) x_{q(i+1)}(n)$  dans (5) pour le remplacer dans (4), il vient:

$$e_{aq(i+1)}(n+1) = \frac{1}{\cos \theta_{i+1}(n)} (e_{aqi}(n+1) - x_{q(i+1)}(n+1) \sin \theta_{i+1}(n))$$

dès lors, en divisant chaque côté par  $\sqrt{E_{a(i+1)}(n+1)}$ , il vient:

$$\frac{e_{aq(i+1)}(n+1)}{\sqrt{E_{a(i+1)}(n+1)}} = \frac{1}{\cos \theta_{i+1}(n)} \frac{1}{\cos \alpha_{i+1}(n+1)} \times \left( \frac{e_{aqi}(n+1)}{\sqrt{E_{ai}(n+1)}} - \sin \alpha_{i+1}(n+1) \sin \theta_{i+1}(n) \right)$$

En effectuant alors les changements de variables suivants:

$$e_{na(i+1)}(n+1) = \frac{e_{aq(i+1)}(n+1)}{\sqrt{E_{a(i+1)}(n+1)}},$$

$$e_{nbi}(n+1) = \sin \theta_{i+1}(n+1), \quad k_{i+1}(n+1) = \sin \alpha_{i+1}(n+1)$$

Cette équation devient:

$$e_{na(i+1)}(n+1) = \frac{1}{\sqrt{1 - k_{i+1}^2(n+1)}} \frac{1}{\sqrt{1 - e_{nbi}^2(n)}} \times (e_{nai}(n+1) - k_{i+1}(n+1) e_{nbi}(n)) \quad (9)$$

##### Propagation de l'erreur de prédiction arrière:

En reportant les changements de variables précédents dans l'équation (6), celle ci devient:

$$g_{i+2}(n+1) = \sqrt{1 - k_{i+1}^2(n+1)} g_{i+1}(n) - \gamma_{i+1}(n) k_{i+1}(n+1) \frac{1}{\sqrt{1 - k_{i+1}^2(n+1)}} \frac{1}{\sqrt{1 - e_{nbi}^2(n)}} \times (e_{nai}(n+1) - k_{i+1}(n+1) e_{nbi}(n))$$

En divisant de part et d'autre par  $\gamma_{i+1}(n+1)$  et en remplaçant  $\frac{\gamma_i(n)}{\gamma_{i+1}(n+1)}$  par  $\frac{1}{\sqrt{1 - e_{nai}^2(n+1)}}$

l'équation devient:



$$e_{nb(i+1)}(n+1) = \frac{1}{\sqrt{1-k_{i+1}^2(n+1)}} \frac{1}{\sqrt{1-e_{nai}^2(n+1)}} \times (e_{nbi}(n) - k_{i+1}(n+1)e_{nai}(n+1)) \quad (10)$$

#### Propagation du coefficient $k$ :

En divisant l'équation (5) par  $\sqrt{E_{ai}(n+1)}$  et en effectuant les changements de variables précédents, celle ci devient:

$$k_{i+1}(n+1) = e_{nai}(n+1)e_{nbi}(n) + \sqrt{1-e_{nai}^2(n+1)}\sqrt{1-e_{nbi}^2(n)}k_{i+1}(n) \quad (11)$$

#### Propagation de l'erreur:

Pour la partie filtrage, il convient d'effectuer les changements de variables suivants:

$$e_{n(i+1)}(n+1) = \frac{e_{q(i+1)}(n+1)}{\sqrt{E_{i+1}(n+1)}} \quad \text{et} \quad k_{f(i+1)}(n+1) = \frac{y_{q(i+1)}(n+1)}{\sqrt{E_i(n+1)}}$$

Dès lors en divisant l'équation (7) par  $\sqrt{E_{i+1}(n+1)}$ , il vient:

$$e_{n(i+1)}(n+1) = \frac{1}{\sqrt{1-k_{f(i+1)}^2(n+1)}} \frac{1}{\sqrt{1-e_{nb(i+1)}^2(n+1)}} \times (e_{ni}(n+1) - k_{f(i+1)}(n+1)e_{nb(i+1)}(n+1)) \quad (12)$$

#### Propagation du coefficient $k_f$ :

En effectuant les changements de variables précédents dans l'équation (8) et en la divisant par  $\sqrt{E_i(n+1)}$ , celle ci devient:

$$k_{f(i+1)}(n+1) = k_{f(i+1)}(n)\sqrt{1-e_{nb(i+1)}^2(n+1)}\sqrt{1-e_{ni}^2(n+1)} + e_{nb(i+1)}(n+1)e_{ni}(n+1) \quad (13)$$

Les équations 9,10,11,12 et 13 forment l'algorithme du treillis normalisé. Ainsi de simples changements de variables ont permis de passer de l'algorithme QR rapide systolique présenté dans cet article, à l'algorithme du treillis normalisé.

Finalement, il apparaît que l'algorithme des moindres carrés récursifs rapides QR systolique et l'algorithme du treillis normalisé sont extrêmement semblables et ne diffèrent que par le nombre de variables propagées. Pour la partie prédiction, le QR propage quatre variables d'un ordre au suivant, ces variables étant:  $e_{aqi}, g_{i+1}, \sqrt{E_{ai}}, \gamma_i$ . L'algorithme du treillis normalisé ne propage que deux variables qui sont  $e_{nai}$  et  $e_{nbi}$ . Or il a été démontré que:

$$e_{nai}(n+1) = \frac{e_{aqi}(n+1)}{\sqrt{E_{ai}(n+1)}} \quad \text{et} \quad e_{nbi}(n+1) = \frac{g_{i+1}(n+1)}{\gamma_i(n+1)}$$

Ainsi, c'est un quotient qui est propagé dans le cas du treillis normalisé alors que c'est un numérateur et un dénominateur dans le cas du QR systolique.

Les changements de variables effectués, qui sont un prolongement de [5], permettent finalement de dresser le tableau d'équivalence suivant (table 3) entre les variables de

l'algorithme QR rapide systolique et celles du treillis normalisé, les notations pour le treillis normalisé sont issues de [2] :

QR systolique	=	Treillis normalisé
$e_{aq(i+1)}(n+1)$	=	$\sqrt{e_{a(i+1)}(n+1)}\varepsilon_{a(i+1)}(n+1)$
$g_{i+1}(n+1)$	=	$\frac{\varepsilon_{bi}(n+1)}{\sqrt{E_{bi}(n+1)}}$
$e_{q(i+1)}(n+1)$	=	$\sqrt{e_{i+1}(n+1)}\varepsilon_{i+1}(n+1)$
$\sqrt{E_{a(i+1)}(n+1)}$	=	$\sqrt{E_{a(i+1)}(n+1)}$
$\gamma_{i+1}(n+1)$	=	$\sqrt{\frac{\varepsilon_{b(i+1)}(n+1)}{E_{b(i+1)}(n+1)}}$
$\sin\theta_{i+1}(n+1)$	=	$e_{nbi}(n+1)$
$\sin\alpha_{i+1}(n+1)$	=	$k_{i+1}(n+1)$
$x_{q(i+1)}(n+1)$	=	$k_{i+1}(n+1)\sqrt{E_{ai}(n+1)}$
$y_{q(i+1)}(n+1)$	=	$k_{f(i+1)}(n+1)\sqrt{E_i(n+1)}$

table 3: Variables équivalentes entre le QR rapide systolique et le treillis normalisé

## V. CONCLUSION

Cet article a proposé une modification de l'algorithme des moindres carrés rapides QR afin de le munir d'un caractère systolique et de le rendre implantable directement sur silicium. Il a ensuite été montré que le QR ainsi modifié ne différait du treillis normalisé que par le nombre de variables propagées.

L'extension de l'algorithme QR systolique au cas des signaux à  $p$  dimensions s'effectue de la même manière que l'extension de l'algorithme non systolique [3] en prenant garde de remplacer l'étape de normalisation par  $p$  étapes de normalisation.

## VI. REMERCIEMENTS

Nous tenons à remercier Monsieur Serge Héthuïn pour son soutien durant ces travaux ainsi que pour sa contribution à la systolisation des algorithmes rapides en vue de leur implantation sur circuits intégrés.

## BIBLIOGRAPHIE

- [1] S. Haykin, "Adaptive Filter Theory," 2nd Ed., Prentice Hall 1991.
- [2] M. Bellanger, "Analyse des signaux et filtrage numérique adaptatif," Masson 1989.
- [3] M. G. Bellanger and P. A. Regalia, "The FLS-QR algorithm for adaptive filtering: The case of multichannel signals," *Signal Processing* 1991, vol 22 pp 115-126.
- [4] C. Samson and V.U. Reddy, "Fixed point error analysis of the normalized ladder algorithm," *IEEE Trans. on ASSP*, Oct. 1983, vol ASSP 31 n°5 pp 1177-1191.
- [5] P.A. Regalia and M. G. Bellanger, "On the duality between fast QR methods and lattice methods in least squares adaptive filtering," *IEEE Trans. on Signal Processing* Apr. 1991, vol SP 39 n°4 pp 879-891.