ETUDE DES EFFETS DE QUANTIFICATION DANS LES ALGORITHMES MCR NUMERIQUEMENT STABLES

Ahmed BENALLAL, André GILLOIRE

Centre National d'études des Télécommunications
CNET LAA/TSS/CMC BP40 22301 LANNION Cedex FRANCE

Résumé

Nous étudions les effets de la quantification des signaux et des variables internes sur la stabilité et la précision numérique des algorithmes des moindres carrés rapides (MCR) numériquement stables appliqués au filtrage adaptatif transverse. Nous comparons d'abord les performances des algorithmes MCR numériquement stables simulés en virgule fixe et en virgule flottante simple précision. Les simulations ont été réalisées dans le contexte de l'annulation d'écho acoustique. Nous montrons que ces algorithmes restent stables lorsqu'ils sont implantés en virgule fixe, et que la dégradation de l'erreur quadratique moyenne de filtrage en sortie provient essentiellement de la quantification de la partie filtrage transverse. En nous basant sur une approche statistique, nous présentons ensuite une analyse théorique des effets de la quantification dans la partie filtrage avec l'hypothèse que les erreurs dues à la quantification du gain d'adaptation (gain de Kalman) sont faibles par rapport à celles dues à la quantification du filtre transverse.

Abstract

We analyse the effects of the quantization of signals and internal variables on the stability and on the numerical precision of fast recursive least squares numerically stable algorithms (noted MCR) for adaptive transversal filtering. We first use simulations to compare the performances of the numerically stable MCR algorithms in fixed point and in floating point arithmetic, in the context of acoustic echo cancellation. We show that those algorithms keep the stability property when they are implemented in fixed point, and that the degradation of the mean square output filtering error results mainly from the quantization of the transversal filtering part in the algorithm. Then, using a statistical appproach, we present a theoretical analysis of the quantization effects in the filtering part, assuming that the errors due to the quantization of the Kalman gain are small in comparison with those due to the quantization of the transversal filter.

1. Introduction

De nouveaux algorithmes des moindres carrés transversaux rapides numériquement stables en virgule flottante simple précision (32 bits) ont été développés récemment [1], [2]. Plusieurs études portant sur l'implantation en virgule fixe des algorithmes MCR ont été publiées auparavant, par exemple [4], [5], [6] pour les algorithmes transversaux et [5], [8], [9] pour les algorithmes en treillis. La dimension N des filtrages n'y excède guère les 20 points; de plus, certains auteurs ont étudié la virgule fixe précision limitée (16 bits) sur des algorithmes MCR numériquement instables en arithmétique virgule flottante simple ou double précision; les résultats entachés par l'accumulation des erreurs numériques ne sont alors valables qu'à très court terme. L'étude que nous présentons a un caractère plus général, car elle est basée sur des algorithmes numériquement stables et des filtres de grande taille. On en trouvera une description détaillée dans [3].

L'arithmétique en virgule fixe précision limitée (typiquement 16 bits) se distingue de l'arithmétique en virgule flottante simple précision 32 bits (1 bit de signe + 23 bits de précision + 7 bits d'exposant) par une dynamique de codage réduite et par une quantification plus grossière des variables de l'algorithme. Si on connaît la dynamique de celles-ci, on peut les cadrer correctement, et la différence porte essentiellement sur l'importance du bruit de quantification. Pour des raisons pratiques, nous assimilerons virgule flottante simple précision et précision infinie.

Dans tout ce qui suit, l'arithmétique est en complément à deux e les calculs sont faits avec arrondi: l'erreur d'arrondi est d moyenne nulle alors que la troncature introduit un biais négati très néfaste sur les variables de l'algorithme [3].

Nous comparons les performances en virgule fixe et en virgul flottante d'un algorithme MCR particulier: le "Fast Transversa Filter" dans une version stabilisée désignée ici par FTFS1. Le je de paramètres de stabilisation est: $\mu^y = \mu^\beta = \mu^b = 1$ [1], [3] D'autres jeux de paramètres donnent des performances prati quement identiques. Une étude similaire a été faite sur l'algo rithme de Kalman rapide stabilisé à l'aide du paramètre $\mu^k = 1$ [3]

Les caractéristiques étudiées sont la stabilité numérique, la pré cision numérique et l'erreur quadratique moyenne de filtrage e sortie. Par stabilité numérique nous entendons un fonctionnemen à long terme de l'algorithme, pendant lequel les variables resten proches de leurs valeurs optimales. Nous définissons la précisio numérique d'un algorithme comme le nombre de bits nécessaire pour que les performances soient acceptables en pratique. L'indice de performance global est l'évolution temporelle de l'énergie de l'erreur de filtrage normalisée par celle du signal à modéliser y_t (signal d'écho):

$$J(t) = 10\,\mathrm{Log}_{10}\left(\frac{<\bar{\varepsilon}^2_{N,t}>}{<y_t^2>}\right) \qquad (1)$$

où $<\cdot>$ symbolise une moyenne temporelle sur 128 ou 256 échantillons consécutifs.

2. Dynamique des Variables

La connaissance a priori des caractéristiques statistiques du signal d'entrée permet une prévision approximative de la dynamique des variables internes de l'algorithme, mais le plus simple est d'évaluer expérimentalement cette dynamique par des simulations en virgule flottante. L'analyse statistique des variables clés de l'algorithme FTFS1 a été faite pour 3 types de signaux d'entrée: bruit blanc, bruit USASI (à spectre moyen de parole) et signal de parole, et pour deux systèmes identifiés: une réponse impulsionnelle synthétique de 32 points et une réponse impulsionnelle de salle tronquée à 256 points. Les résultats détaillés sont présentés dans [3]. L'ensemble des variables de l'algorithme en virgule fixe est codé sur 16 bits; le nombre de bits des parties entière et fractionnaire de chaque variable est choisi selon la dynamique correspondante observée.

La constante d'initialisation E_0 a été choisie supérieure ou égale à l'énergie du signal d'entrée x_t, ce qui permet de coder les régimes transitoire et permanent sur le même nombre de bits.

Notons enfin que les simulations montrent que les algorithmes MCR ne sont pas très robustes à des saturations excessives des valeurs de grandes amplitudes.

3. Stabilité numérique

La stabilité numérique en virgule fixe a été testée au moyen de simulations sur environ 10^6 échantillons, soit 62.5 secondes de signal. Les simulations ont porté sur les deux réponses impulsionnelles mentionnées plus haut, convoluées soit avec le bruit blanc, soit avec le bruit USASI. Lors de ces tests, aucune perturbation externe (bruit en sortie) n'a été ajoutée au signal y_t.

Les résultats obtenus (voir [3]) montrent l'efficacité de la méthode de stabilisation numérique que nous avons proposée dans [1] et [3]: l'algorithme FTFS1 reste stable en virgule fixe 16 bits alors que le FTF 7N standard en flottant diverge très rapidement. Des résultats similaires ont été obtenus avec l'algorithme de Kalman rapide.

4. Précision Numérique

Pour un nombre de bits donné, la précision numérique des algorithmes MCR dépend du facteur d'oubli, du nombre de coefficients à identifier et de l'énergie du signal d'entrée. Elle est mesurée en régime asymptotique par le palier de la courbe de performance $J(t)$.

a) Influence du nombre de bits de codage

Les performances en virgule fixe sont très proches de celles en virgule flottante si les variables sont bien cadrées et si la longueur des mots en virgule fixe est égale à la taille de la mantisse flottante. En fait, il suffit d'identifier les variables qui doivent être codées avec une plus grande précision que les autres pour que la précision de l'algorithme en virgule fixe se rapproche de la précision en virgule flottante. Notre approche expérimentale consiste à comparer les performances obtenues par simulation en utilisant une quantification par arrondi appliquée sélectivement à certaines variables. Nous distinguons les deux parties de l'algorithme:

- ***Partie prédiction aller/retour (désignée par AR):*** elle fournit le gain de Kalman $C_{N,t}$ utilisé dans l'adaptation de la partie filtrage.

- ***Partie filtrage et adaptation du filtre (désignée par MA) :*** elle se résume aux deux équations suivantes:

$$\bar{\varepsilon}_{N,t} = y_t - H^T_{N,t-1} X_{N,t} \qquad (2)$$

$$H_{N,t} = H_{N,t-1} - \tilde{C}_{N,t} \gamma_{N,t} \bar{\varepsilon}_{N,t} \qquad (3)$$

Notons que le calcul de la partie prédiction est indépendant de la partie filtrage.

Dans un premier temps, nous avons identifié les deux réponses impulsionnelles (N=32 et N=256) à l'aide de l'algorithme FTFS1 implanté en virgule flottante simple précision et en virgule fixe 16 bits. Ces résultats nous serviront de référence.

Quatre essais avec quantification par arrondi sélective ont été effectués:

- ***Essai 1:*** partie AR en virgule fixe et partie MA en virgule flottante.

- ***Essai 2:*** partie AR en virgule flottante et partie MA en virgule fixe.

- ***Essai 3:*** l'ensemble des variables scalaires et vectorielles en flottant sauf le vecteur $H_{N,t}$ de la partie MA en virgule fixe.

- ***Essai 4:*** l'ensemble des variables scalaires et vectorielles en virgule fixe sauf $H_{N,t}$ en flottant.

Nous ne présentons ici que les résultats correspondant à N=256. On observe pour N=32 des résultats analogues, bien que la dégradation entre virgule flottante et virgule fixe soit alors beaucoup plus faible.

Sur la figure 1 les essais 1 et 2 sont comparés à la simulation de l'intégralité de l'algorithme soit en virgule flottante simple précision soit en virgule fixe 16 bits. Le résultat de l'essai 1 (quantification de la partie AR) est très proche de celui de la simulation intégrale en virgule flottante (courbes du bas de la figure), alors que le résultat de l'essai 2 (quantification de la partie MA) est très proche de celui de la simulation intégrale en virgule fixe 16 bits (courbes du haut). Ceci montre que c'est la partie MA (et non la partie AR) qui est responsable de la dégradation observée en virgule fixe. Ce résultat peut être observé même pour un codage de la partie AR sur moins de 16 bits. Par conséquent, le calcul du gain de Kalman apparaît robuste vis-à-vis de la quantification.

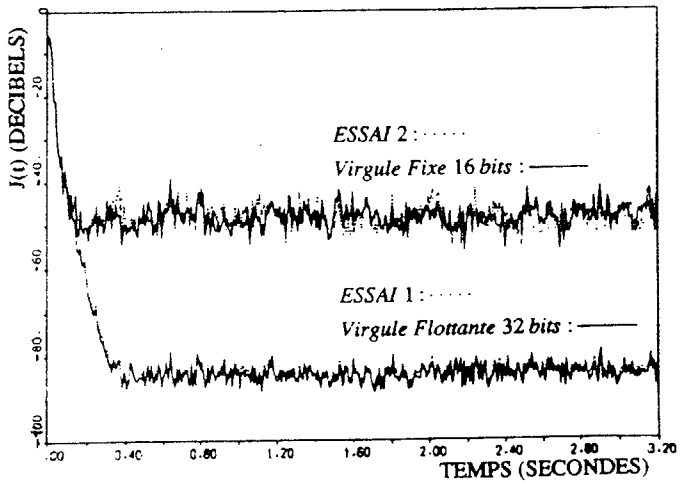

Figure 1: Test de la précision numérique en virgule fixe
N=256 $\lambda = 0.9987$ Eo=1. x_t = bruit USASI (Fech=16 kHz)

Les essais 3 et 4 confirment les résultats précédents en mettant en évidence l'influence de la quantification du vecteur $H_{N,t}$ de la partie MA sur la performance globale de l'algorithme. La quantification de $H_{N,t}$ seul sur 16 bits donne un résultat équivalent à celui obtenu en quantifiant l'ensemble des variables de l'algorithme. L'essai 4, qui réalise l'opération inverse de l'essai 3, confirme ce résultat. La courbe de performance (non montrée ici) est pratiquement identique à celle de l'essai 1.

La figure 2 compare les résultats en virgule fixe pour trois précisions différentes de codage (11, 17 et 23 bits) du vecteur $H_{N,t}$. La partie AR et l'erreur de filtrage sont codées sur 16 bits. Il est clair que la plus grande part de la dégradation est due à la quantification du filtre transverse principal.

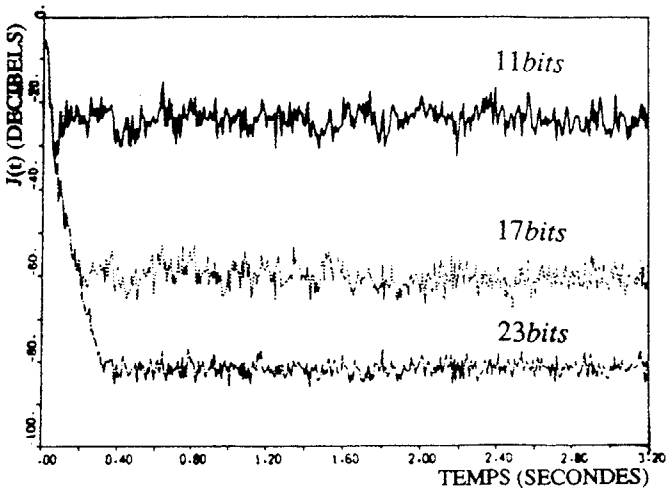

Figure 2 : Influence du nombre de bits de codage du filtre
N=256 $\lambda = 0.9987$ Eo=1. x_t = bruit USASI

On vérifie que le codage de ce dernier sur 23 bits rend la performance globale équivalente à celle obtenue en virgule flottante (voir figure 1).

b) Influence du nombre de paramètres à identifier

En virgule fixe 16 bits, le passage de N=32 à N=256 apporte une nette dégradation: l'EQM est augmentée de 17 dB pour N=32 et de 37 dB pour N=256, par rapport à l'EQM en arithmétique flottante. En annulation d'écho acoustique (typiquement N > 500), le codage de $H_{N,t}$ sur plus de 16 bits peut donc être nécessaire pour atteindre une atténuation de l'écho satisfaisante (> 30 dB).

c) Influence du facteur d'oubli

En précision infinie, il est connu qu'un facteur d'oubli proche de 1 diminue l'EQM de filtrage. Notre étude théorique montre qu'en virgule fixe une augmentation du facteur d'oubli entraîne une augmentation de l'EQM due à la quantification du vecteur filtre $H_{N,t}$ et réduit en même temps les contributions dues à la quantification des signaux d'entrée et de l'erreur de filtrage (formule (17)). Il existe donc un choix de λ qui assure un compromis entre les contributions des différentes erreurs de quantification à l'erreur finale (voir formule (18)).

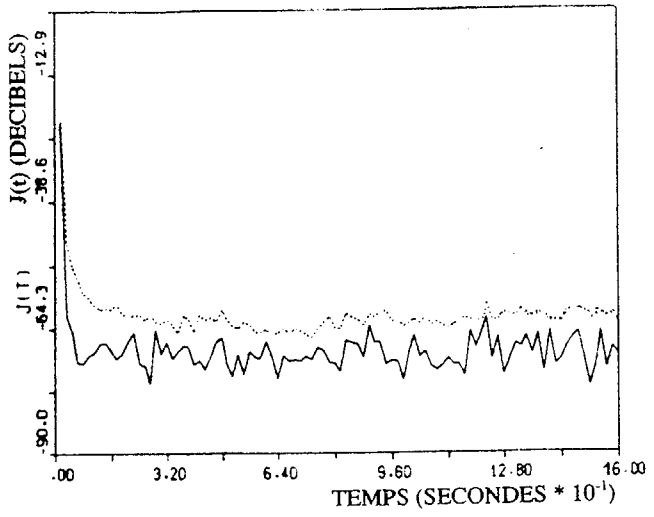

Figure 3 : Influence du **facteur d'oubli** en virgule fixe
$N = 32, \lambda = 0.99$ (——), $\lambda = 0.9989$ (· · ·)

La figure 3 montre un cas où le passage de λ=0.99 à λ=0.9989 dégrade les performances en virgule fixe. Cette tendance en fonction de λ peut être différente, notamment en fonction de la puissance du bruit additif en sortie.

d) Influence de l'énergie du signal d'entrée

Nous avons observé dans les simulations en virgule fixe que l'EQM varie en raison directe de l'énergie du signal d'entrée. En précision infinie, l'erreur de filtrage asymptotique ne dépend que de l'énergie du bruit non mesurable; ceci a été vérifié par des simulations en virgule flottante. Comme le montre notre analyse théorique, cette dépendance de l'EQM en virgule fixe vis-à-vis de l'énergie du signal est due principalement à la quantification du filtre $H_{N,t}$. L'analyse théorique donne des valeurs optimales de l'énergie du signal qui minimisent l'EQM en virgule fixe, mais ces valeurs sont trop faibles par rapport aux signaux utilisés en pratique.

5. Analyse théorique des effets de la quantification

Les simulations montrent que la dégradation de l'EQM est due essentiellement à la quantification du filtre $H_{N,t}$. Dans un but de simplicité, l'analyse théorique a donc été effectuée avec l'hypothèse que le gain de Kalman est calculé avec une précision infinie. Les résultats que nous présentons sont plus généraux que ceux obtenus ailleurs [7], car ils tiennent compte de la quantification du signal de référence x_t et du signal désiré y_t.

Suivant [10], [11], les erreurs de quantification instantanées sont supposées indépendantes entre elles et indépendantes des divers signaux de l'algorithme. Chaque erreur est modélisée par un bruit blanc de moyenne nulle et de variance $2^{-2b}/12$, où b est le nombre de bits de la partie fractionnaire. Les calculs se font sans débordements et seule la multiplication introduit une erreur. On supposera de plus les signaux x_t et y_t stationnaires et centrés.

$\hat{Z}_t$ désigne la valeur calculée en précision finie à l'instant t d'une variable théorique Z_t. L'erreur due à une simple quantification, à l'instant t, est notée δZ_t (vecteur) ou $\delta_z(t)$ (scalaire); sa variance est $\sigma^2_{Q_Z}$. L'accumulation jusqu'à l'instant t des erreurs successives est désigné par ΔZ_t.

Comme le calcul du gain de Kalman est supposé sans erreur, les deux équations de base à analyser sont (2) et (3). Nous avons largement utilisé le principe de la moyenne [8]: un processus $y(t)$ lentement variable devant un processus $x(t)$ qui lui est corrélé, est considéré comme indépendant de $x(t)$.

a) Algorithme en précision finie

En désignant par $Q_Z[Z]$ un quantificateur uniforme de la variable Z, les équations (2) et (3) s'écrivent:

$$\hat{\varepsilon}_{N,t} = Q_\varepsilon[Q_y[y_t] - \hat{H}^T_{N,t-1}\hat{X}_{N,t}] \tag{4}$$

$$\hat{H}_{N,t} = \hat{H}_{N,t-1} - Q_H[C_{N,t}\hat{\varepsilon}_{N,t}] \tag{5}$$

En ne retenant que les erreurs du premier ordre, l'algorithme en précision finie s'écrit:

$$\hat{\varepsilon}_{N,t} = \bar{\varepsilon}_{N,t} - (\Delta H^T_{N,t-1}X_{N,t} + H^T_{N,t-1}\delta X_{N,t} + \delta_\varepsilon(t) - \delta_y(t)) \tag{6}$$

$$\begin{aligned}\hat{H}_{N,t} &= H_{N,t} + \Delta H_{N,t} \\ &= H_{N,t} + (I_N + C_{N,t}X^T_{N,t})\Delta H_{N,t-1} + C_{N,t}H^T_{N,t-1}\delta X_{N,t} \\ &\quad + C_{N,t}(\delta_\varepsilon(t) - \delta_y(t)) - \delta_H(t)\end{aligned} \tag{7}$$

b) Calcul de l'erreur quadratique moyenne de filtrage

L'erreur quadratique moyenne de filtrage en précision finie s'obtient à partir de (6), en prenant en compte l'hypothèse d'indépendance des bruits de quantification:

$$EQM_{PF} = E\{\hat{\varepsilon}^2_{N,t}\}$$

$$= E\{\bar{\varepsilon}^2_{N,t}\} + E\{(\Delta H^T_{N,t-1} X_{N,t})^2\} + E\{(H^T_{N,t-1}\delta X_{N,t})^2\} + E\{\delta_y(t)^2 + \delta_\varepsilon(t)^2\}$$

$$= A + B + C + D \qquad (8)$$

Le terme D traduit une simple quantification de l'erreur de filtrage et du signal y_t:

$$D = \sigma^2_{Qy} + \sigma^2_{Q\varepsilon} \qquad (9)$$

où on a supposé que le produit scalaire intervenant dans (4) est quantifié après accumulation de la somme.

Le terme A représente l'EQM en précision infinie. Pour le calculer, on modélise y_t par:

$$y_t = H^T_{opt} X_{N,t} + \varepsilon_{opt,t} \qquad (10)$$

où $\varepsilon_{opt,t}$ est le bruit de sortie non mesurable, supposé blanc de variance $E_{\min}$, et on utilise la solution $H_{N,t}$ des moindres carrés exacts:

$$H_{N,t} = R^{-1}_{N,t} \sum_{k=1}^{t} \lambda^{t-k} X_{N,k} y_k \qquad (11)$$

En utilisant en plus la blancheur du bruit additif, le caractère lentement variable de la matrice $R_{N,t}$ (λ proche de 1) par rapport à la matrice $X_{N,t}X^T_{N,t}$ et en se plaçant en régime asymptotique, on obtient:

$$A = E\{\bar{\varepsilon}^2_{N,t}\} \approx \left(1 + \frac{1-\lambda}{1+\lambda} N\right) E_{\min} \qquad (12)$$

Le terme C traduit l'influence de la quantification du signal d'entrée x_t présent dans la mémoire de l'algorithme. Les hypothèses d'indépendance des bruits de quantification donnent:

$$C = trace\{E(H_{N,t-1} H^T_{N,t-1})\sigma^2_{Qx}\} \qquad (13)$$

Posant $trace(R^{-1}_{N,XX}) = trace(E(X_{N,t}X^T_{N,t})) = N/\sigma^2_x$, et avec les hypothèses de variation lente de $R_{N,t}$ et de blancheur de $\varepsilon_{opt,t}$, on obtient, pour t grand:

$$C = \left(|H_{opt}|^2 + N \cdot \frac{1-\lambda}{1+\lambda} \cdot \frac{E_{\min}}{\sigma^2_x}\right)\sigma^2_{Qx} \qquad (14)$$

où $|\cdot|$ désigne la norme euclidienne d'un vecteur.

Le calcul du terme B est plus complexe. B traduit l'effet de la quantification de l'algorithme d'adaptation, et fait intervenir l'ensemble des quantificateurs. En première approximation, on suppose que les erreurs sur les composantes du vecteur $\hat{H}_{N,t}$ sont indépendantes et qu'elles admettent la même variance:

$$B \approx trace\{E(\Delta H_{N,t-1}\Delta H^T_{N,t-1}) R_{N,XX}\}$$

$$\approx N \cdot \sigma^2_x \cdot \sigma^2_{\Delta H} \qquad (15)$$

où $\sigma^2_{\Delta H}$ est la variance asymptotique du bruit dans les coefficients du filtre. En utilisant le principe de la moyenne et λ proche de 1, et en faisant l'hypothèse que x_t est une séquence blanche, on obtient:

$$\sigma^2_{\Delta H} = \frac{\sigma^2_{Q_H}}{1-\lambda^2} + \frac{1-\lambda}{1+\lambda} \cdot \frac{\sigma^2_{Q_y} + P\sigma^2_{Q_{\tilde{\varepsilon}}}}{\sigma^2_x} + \frac{1-\lambda}{1+\lambda} \cdot \frac{\sigma^2_{Q_x}}{\sigma^2_x} \cdot \left(|H_{opt}|^2 + \frac{1-\lambda}{1+\lambda} \cdot \frac{E_{\min}}{\sigma^2_x} \cdot N\right) \qquad (16)$$

Finalement, l'EQM en précision finie (8) s'écrit:

$$EQM_{PF} = \left(1 + \frac{1-\lambda}{1+\lambda} \cdot N\right) \cdot E_{\min} + \frac{N\sigma^2_{Q_H}\sigma^2_x}{1-\lambda^2} + \left(1 + \frac{1-\lambda}{1+\lambda} N\right)\left(\sigma^2_{Q_y} + \sigma^2_{Q_{\tilde{\varepsilon}}}\right) + \left(1 + \frac{1-\lambda}{1+\lambda} N\right)\left(|H_{opt}|^2 + \frac{1-\lambda}{1+\lambda} \cdot \frac{E_{\min}}{\sigma^2_x} \cdot N\right) \cdot \sigma^2_{Q_x} \qquad (17)$$

Cette formule peut servir de guide pour l'implantation pratique des algorithmes. En annulant sa dérivée par rapport à λ, on obtient la valeur du facteur d'oubli qui minimise l'EQM_{PF}:

$$\lambda_{opt} = \frac{1 - 0.5\nu}{1 + 0.5\nu} \qquad (18)$$

où

$$\nu = \sqrt{\frac{\sigma^2_{Q_H}\sigma^2_x}{E_{\min} + \sigma^2_{Q_y} + P\sigma^2_{Q_{\tilde{\varepsilon}}} + \sigma^2_{Q_x}|H_{opt}|^2}}$$

Pour obtenir cette dernière expression, on approche $(1 - \lambda^2)$ par $2(1 - \lambda)$ et on néglige les termes pondérés par le produit $E_{\min}\sigma^2_{Q_x}$.

La relation (18) peut être utile en pratique pour ajuster le facteur d'oubli en fonction de la précision des variables, du bruit additif et de la vitesse de convergence désirée.

La bonne correspondance entre résultats théoriques et expérimentaux a été vérifiée sur les deux réponses impulsionnelles (N=32 et N=256) avec un bruit blanc gaussien à l'entrée (voir [3]). Notons que cette analyse théorique reste valable pour les algorithmes MCR transversaux en général.

Références bibliographiques

[1] A.Benallal and A.Gilloire, "A New Method to Stabilize Fast RLS Algorithms Based on a First-order Model of the Propagation of Numerical Errors", IEEE ICASSP'88, New-York.

[2] D.T.M.Slock and T.Kailath, "Numerically Stable Fast RLS Transversal Filters", IEEE ICASSP'88, New-York.

[3] A.Benallal, "Etude des algorithmes des moindres carrés transversaux rapides et application à l'identification de réponses impulsionnelles acoustiques" Thèse Univ. Rennes-1, Jan. 1989.

[4] R.Alcantara, "Implantion d'algorithmes rapides sur des Processeurs de Traitement du Signal" Thèse ENST Paris Sep. 1986.

[5] F.Ling, D.Manolakis and G.Proakis, "Finite Word-length in RLS algorithms with Application to Adaptive Equalization", Ann. Télécomm., 41, n° 5-6, 1986.

[6] J.M.Cioffi, "Limited-Precision Effects in Adaptive Filtering", IEEE Trans. CAS-34, n° 7, July 1987.

[7] S.H.Ardalan and S.T.Alexander, "Fixed-Point Roundoff Error Analysis of Exponentially Windowed RLS Algorithm for Time-Varying Systems", IEEE Trans. ASSP-35, n° 6, June 1987.

[8] C.G.Samson and R.U.Reddy, "Fixed Point Error Analysis of the Normalized Ladder Algorithm", IEEE Trans. ASSP-31, n° 5, Oct.1983.

[9] J.F.Agnel and J-V.Lucas,"Etude des Effets de Quantification sur les Algorithmes en Treillis Adaptatifs", 10° Colloque GRETSI, Mai 1985.

[10] C.Caraisos and B.Liu,"A Roundoff Error Analysis of the LMS Adaptive Algorithm", IEEE Trans. ASSP-32, n° 1, Feb.1984.

[11] H.Dedieu and F.Castanie, "Analyse des Effets de la Quantification dans l'algorithme LMS" 11° Colloque GRETSI, Juin 1987.