



INSTABILITE ET STABILITE DES ALGORITHMES DES MOINDRES CARRES TRANSVERSAUX RAPIDES EXCITES PAR LA PAROLE

Ahmed BENALLAL* et André GILLOIRE**

* Département Electronique - I.N.E.S. de BLIDA - BLIDA, ALGERIE

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

Résumé

Les algorithmes des moindres carrés transversaux rapides numériquement stables (MCRNS) avec des signaux stationnaires à l'entrée, divergent avec des signaux non-stationnaires comme la parole. Pour remédier à ces divergences, nous proposons deux techniques qui permettent le fonctionnement continu normal des algorithmes MCRNS excités par le signal de parole, dans le contexte du filtrage adaptatif transverse. La première est une méthode de réinitialisation utilisable avec la plupart des algorithmes MCRNS; la seconde combine la régularisation de la variance de l'erreur de prédiction aller et le rappel à zéro des prédicteurs aller-retour, et conduit à un algorithme MCRNS particulier de type "Fast Transversal Filter" (FTF).

Abstract

Fast least squares transversal algorithms which are numerically stable (MCRNS algorithms) with stationary inputs, diverge with non-stationary input signals like speech. We present two techniques which allow the continuous, normal operation of those algorithms for transversal adaptive filtering, when they are excited by speech signals. The first one is a reinitialization method which can be used with a large number of fast transversal algorithms; the second one combines the regularization of the forward prediction error variance and a leakage operation on the forward and backward predictors, and it leads to a particular algorithm of the Fast Transversal Filter (FTF) type.

1. Introduction

Plusieurs algorithmes des moindres carrés transversaux rapides numériquement stables (MCRNS) en présence de signaux d'entrée stationnaires ont été proposés récemment [1-4]. Certains de ces algorithmes restent stables même lorsqu'ils sont implantés en virgule fixe 16 bits [3]. Cependant, l'utilisation de signaux d'entrée non-stationnaires comme la parole fait apparaître des problèmes de divergence dans ces algorithmes, même lorsqu'ils sont implantés en virgule flottante simple précision (32 bits).

Nous étudions le comportement des algorithmes MCRNS par simulations sur des signaux de parole, dans le contexte du filtrage adaptatif transverse. Le rôle des algorithmes MCRNS est alors de fournir à la partie filtrage un vecteur gain d'adaptation (ou gain de Kalman) pour adapter le filtre transverse:

$$H_{N,t} = H_{N,t-1} - \bar{\epsilon}_{N,t} C_{N,t} \quad (1)$$

$$\bar{\epsilon}_{N,t} = y_t - H_{N,t-1}^T X_{N,t} \quad (2)$$

où y_t est le signal à modéliser, $\bar{\epsilon}_{N,t}$ est l'erreur de filtrage et $C_{N,t}$ ($= \gamma_{N,t} \tilde{C}_{N,t}$) désigne le gain d'adaptation. Les variables $X_{N,t}$, $\gamma_{N,t}$ et $\tilde{C}_{N,t}$ désignent respectivement le vecteur signal d'entrée, la variable de vraisemblance et le gain de Kalman dual. Le gain d'adaptation est calculé, indépendamment de la partie filtrage, par un algorithme MCRNS qui effectue une analyse par prédiction linéaire aller-retour sur le signal de parole x_t .

Après la description des phénomènes de divergence observés, nous présentons deux techniques qui permettent un fonctionnement continu des algorithmes MCRNS avec les signaux de parole à l'entrée, dans le contexte du filtrage: une méthode de réinitialisation pour les algorithmes MCRNS qui divergent en se bloquant sur les paramètres déjà identifiés, et un algorithme de type "Fast Transversal Filter" (FTF) qui utilise la régularisation de la variance des erreurs de prédiction aller et le rappel à zéro des prédicteurs aller-retour, technique déjà utilisée avec l'algorithme de Kalman rapide, plus complexe [5]. Ces deux techniques conduisent à des algorithmes MCRNS optimaux en ce qui concerne le niveau de l'erreur de filtrage asymptotique. Néanmoins, la modification de l'algorithme de type FTF que nous proposons ralentit la vitesse de convergence.

2. Instabilité des algorithmes MCRNS avec la parole à l'entrée

L'étude par simulations que nous avons menée montre qu'aucun des algorithmes MCRNS optimaux au sens des moindres carrés [3], [4], [1] ne présente un fonctionnement stable (c'est-à-dire un fonctionnement sans interruption sur une très longue durée de signal), quel que soit le signal de parole à l'entrée: tous ces algorithmes divergent pour un facteur d'oubli λ inférieur à 1.

Globalement, la cause de ces divergences peut être attribuée au caractère fortement non-stationnaire du signal de parole. La présence de longues périodes de silence rend les variances des erreurs de prédiction aller-retour très faibles devant l'énergie moyenne du signal de parole, et à l'arrivée du signal utile



l'algorithme se trouve mal initialisé. Ce phénomène peut aussi être observé sur des sons voisés très pauvres en harmoniques dont le caractère prévisible rend les variances des erreurs de prédiction très faibles.

Des problèmes liés à l'accumulation des erreurs numériques peuvent aussi surgir à cause des non-stationnarités dans le signal lui-même (transitions entre deux sons). En effet, les algorithmes MCRNS [1], [3] sont basés sur une analyse de la propagation des erreurs numériques au premier ordre, qui suppose que les erreurs numériques introduisent de petites perturbations autour de la solution nominale (théorique). Ceci est suffisant pour combattre l'accumulation des erreurs numériques pour un signal d'entrée x_t stationnaire ou faiblement non-stationnaire. Or avec la parole, après les instants qui suivent une forte non-stationnarité (changement de modèle), la solution (dans la partie prédiction aller/retour) de l'algorithme est momentanément très éloignée de la solution optimale. Avant que l'algorithme ne se rapproche de l'optimum, l'analyse des erreurs au premier ordre et les algorithmes MCR stabilisés qui en découlent peuvent être mis en défaut.

L'algorithme MCRNS de type FTF de la table 1 dans [3] (noté ici FTFS1 pour le choix des paramètres de contrôle de la stabilité $\mu^b = \mu^c = \mu^v = 1$) présente avec les signaux de parole un type de divergence qui, contrairement à celui dû à l'accumulation des erreurs numériques (explosion de la majorité des variables scalaires et vectorielles de l'algorithme non stabilisé), se traduit par une convergence vers 0 des variances des erreurs de prédiction aller/retour, de la variable de vraisemblance $\gamma_{N,t}$ ainsi que du gain d'adaptation $C_{N,t} = \gamma_{N,t} \tilde{C}_{N,t}$. Les vecteurs de prédiction aller/retour $a_{N,t}$ et $b_{N,t}$ ainsi que le filtre transverse principal $H_{N,t}$ perdent leur caractère adaptatif et se bloquent sur les valeurs déjà identifiées. Notons que ce type de divergence peut être observé avec d'autres algorithmes MCRNS décrits dans la littérature.

La figure 1a présente une tranche de signal de parole analysée (échantillonnage à 8 KHz); la figure 1b présente en correspondance la variable de vraisemblance $\gamma_{N,t}$ dans l'algorithme FTFS1. L'ordre N du filtre est égal à 32, le facteur d'oubli λ vaut 0.992, et la constante d'initialisation E_0 est choisie égale à 1.

La divergence se manifeste par un nette diminution de la valeur de la variable de vraisemblance intervenant juste après une non-stationnarité du signal (passage d'une zone voisée très pauvre en harmoniques à une zone voisée très riche en harmoniques). La même étude, conduite sur deux autres algorithmes FTFS [1],[4], montre qu'avant l'instant de divergence (identique) les trois algorithmes donnent presque exactement la même variable de vraisemblance; après cet instant les algorithmes se comportent différemment [7]. La variable de vraisemblance de l'algorithme FTFS1 continue à converger vers 0, conduisant au blocage de la majorité des variables récursives mises en jeu par l'algorithme.

Les arrêts d'adaptation (ou blocages) de l'algorithme dépendent du choix du facteur d'oubli: un facteur d'oubli "faible" (par exemple $\lambda = 0.99$) conduit, dans les zones de silence ou de signal très prédictible, à la convergence rapide vers 0 des variances des erreurs de prédiction aller/retour. Ceci augmente le risque de blocage de l'algorithme à l'arrivée d'une non-stationnarité importante. Un facteur d'oubli proche de 1 (par exemple $\lambda = 0.999$) conduit à un meilleur comportement.

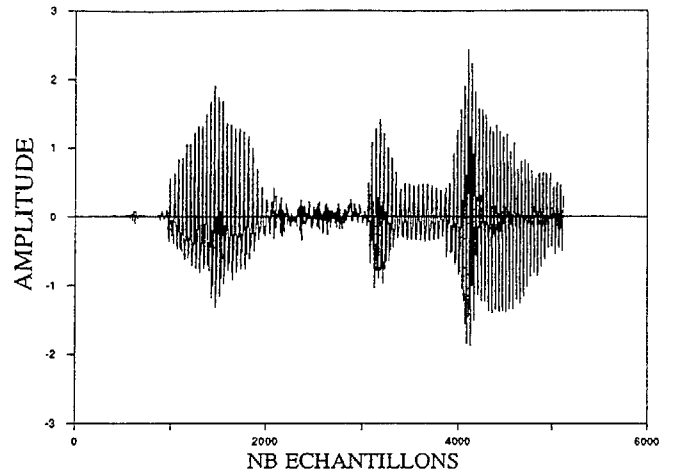


Figure 1a: Tranche de signal de parole (durée 0.75 s)

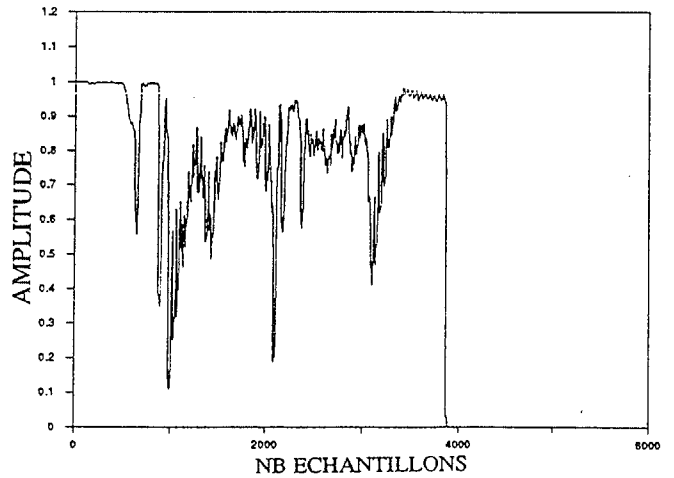


Figure 1b: Variable de vraisemblance (Algorithme FTFS1)

3. Stabilisation par la méthode de réinitialisation

Comme on l'a vu précédemment, la parole provoque une extinction de certains algorithmes MCR numériquement stables sur les paramètres du filtre transverse identifié. La méthode la plus simple permettant de maintenir l'adaptativité du filtrage est la réinitialisation de l'algorithme sur les paramètres $H_{N,t}$ déjà identifiés. Pour cela, on utilise la procédure d'initialisation proposée dans [6]. Le point nouveau dans la technique que nous proposons est d'une part la définition d'un seuil minimal γ_{\min} de la variable de vraisemblance $\gamma_{N,t}$ au-dessous duquel on réapprend le gain de Kalman, et d'autre part qu'on applique la procédure de réinitialisation sur l'échantillon courant. La procédure est la suivante:

si $\gamma_{N,t-1} \leq \gamma_{\min}$ alors:

$$a_{N,t-1} = b_{N,t-1} = \tilde{C}_{N,t-1} = \downarrow 0_N \quad , \quad H_{N,t-1} = H_{N,t-1} \quad (3)$$

$$\alpha_{N,t-1} = E_0 \quad , \quad \beta_{N,t-1} = E_0 \lambda^{-N} \quad , \quad \gamma_{N,t-1} = 1 \quad (4)$$

où $\downarrow 0_N$ représente le vecteur nul de taille N .

Les seuls paramètres à fixer sont le seuil minimal γ_{\min} et la constante d'initialisation E_0 des variances des erreurs de prédiction aller/retour. Le choix du seuil γ_{\min} n'est pas très critique, n'importe quel seuil très faible réinitialise l'algorithme mais risque de le laisser dans une période de faible adaptativité plus longtemps. La valeur asymptotique λ^N de la variable de vraisemblance avec les signaux x_t stationnaires [7] peut être utilisée pour le choix de γ_{\min} .

Un facteur d'oubli λ faible conduit à des fluctuations importantes de $\gamma_{N,t}$; on pourra alors choisir par exemple $\gamma_{\min} = \lambda^N/10$. Par contre un facteur d'oubli proche de 1 donne une variable $\gamma_{N,t}$ moins erratique et on pourra choisir un seuil de réinitialisation relativement élevé: $\gamma_{\min} = \lambda^N/5$ par exemple.

La constante E_0 doit être bien choisie pour assurer le redémarrage de l'algorithme. Si on connaît la dynamique du signal d'entrée et son énergie maximale admissible, on pourra fixer E_0 a-priori en fonction de N . Sinon, on peut choisir pour E_0 "le pire des cas" où le blocage se produit au passage d'une zone de silence vers une zone de signal non voisé (qu'on suppose blanc). Pour cela, on estime en permanence l'énergie du signal d'entrée par $E(t)$:

$$E(t) = WE(t-1) + x_t^2 \quad (5)$$

$$E_0 = E(t_t) \quad (6)$$

où t_t est l'instant où $\gamma_{N,t-1}$ devient inférieure à γ_{\min} , et $W=\lambda$.

Ce choix de E_0 est assez pessimiste, mais il assure le redémarrage de l'algorithme. Une constante $W < \lambda$ permet de réduire les valeurs prises par la constante E_0 et de mieux poursuivre les variations du signal d'entrée.

Afin d'illustrer les performances de la méthode de réinitialisation, on présente sur la figure 2 l'évolution temporelle de l'énergie de l'erreur de filtrage (EQM) correspondant à l'identification d'une réponse impulsionnelle de salle tronquée à $N=256$ points. Les paramètres utilisés sont: $\lambda=0.999$, $E_0=1$, $\gamma_{\min}=0.1$, $W=0.995$. Le signal de parole est échantillonné à 8 kHz et l'énergie du bruit en sortie est $E_{\min}=-80$ dB.

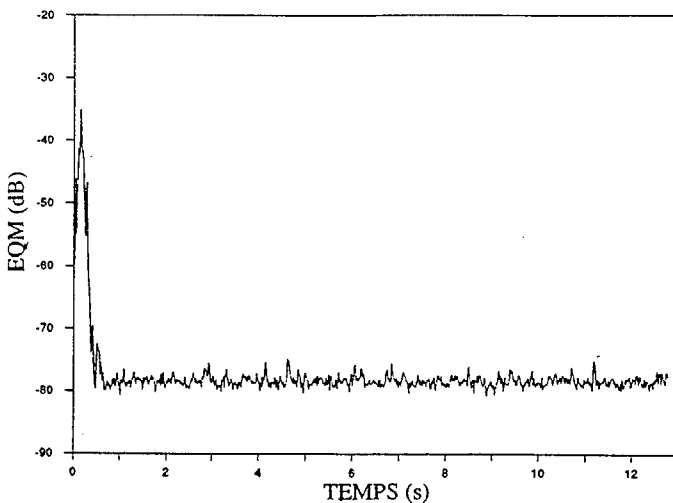


Figure 2: EQM de filtrage (méthode de réinitialisation)

On constate que la méthode de réinitialisation ne dégrade absolument pas les performances de filtrage. L'identification sur $N=32$ points montre aussi un comportement satisfaisant [7]. Notons que cette méthode de réinitialisation peut être utilisée avec un grand nombre d'algorithmes MCR numériquement stables [7].

4. Régularisation et Rappel à zéro des prédicteurs

Des méthodes de régularisation des variances et de rappel à zéro des prédicteurs ont été testées ailleurs sur l'algorithme de Kalman rapide [5]. Rappelons que la régularisation consiste à ajouter à la variance des erreurs de prédiction aller une constante positive C :

$$\alpha'_{N,t} = \alpha_{N,t-1} + C \quad (7)$$

alors que le rappel à zéro est la multiplication des prédicteurs aller/retour par une constante légèrement inférieure à 1:

$$a'_{N,t} = (1-g)a_{N,t} \quad (8)$$

$$b'_{N,t} = (1-g)b_{N,t} \quad (9)$$

avec $0 < g \ll 1$.

L'algorithme que nous proposons est une extension des méthodes de régularisation à un algorithme de type FTF plus intéressant en complexité de calcul que l'algorithme de Kalman rapide. Le rôle de la constante C est d'éviter la convergence vers 0 des variances aller/retour et de permettre un lissage des fortes discontinuités dues aux passages de non-stationnarités importantes. Le facteur de rappel à zéro g permet un oubli du contenu des prédicteurs aller/retour dans les longues périodes de silence; à l'arrivée du signal utile, l'algorithme effectue une sorte de réinitialisation automatique grâce à la régularisation des variances.

L'application des méthodes de régularisation et de rappel à zéro aux algorithmes MCRNS de type FTF n'est pas triviale. D'une part la seule régularisation de la variance aller entraîne un blocage irréversible de l'algorithme (la variable de vraisemblance converge vers 0), d'autre part la régularisation de la variance retour peut créer des problèmes numériques car l'ajout de la constante C provoque un écart important entre les valeurs calculées et les valeurs théoriques, et les méthodes de stabilisation fondées sur une approximation au premier ordre des erreurs numériques peuvent être mises en défaut.

Pour éviter la convergence vers 0 de $\gamma_{N,t}$, on utilise les deux formules suivantes:

$$\alpha_{N,t} = \lambda\alpha_{N,t-1} + \gamma_{N,t-1}\bar{e}_{N,t}^2 + C \quad (10)$$

$$\gamma_{N+1,t} = \frac{\lambda\alpha_{N,t-1} + C}{\alpha_{N,t}}\gamma_{N,t-1} \quad (11)$$

La régularisation de la variance des erreurs de prédiction retour sera évitée en utilisant un algorithme FTF numériquement stable qui ne calcule pas cette variance [7]. Cet algorithme calcule l'une des erreurs de prédiction retour qui sert à la stabilisation numérique à l'aide de la variance aller:

$$\bar{r}_{N,t}^p = -\lambda^{-N+1}\alpha_{N,t-1}\gamma_{N,t-1}\tilde{C}_{N+1,t}^{N+1} \quad (12)$$

où $\tilde{C}_{N+1,t}^{N+1}$ est la $N+1$ ème composante du gain de Kalman dual d'ordre $N+1$.

Il faut noter que la régularisation et le rappel à zéro ne règlent absolument pas le problème de l'instabilité due à l'accumulation des erreurs numériques dans les algorithmes MCR. Pour la stabilisation on utilise les équations:

$$\bar{r}_{N,t}^b = \bar{r}_{N,t}^c + \mu^b \xi_{N,t} \quad (13)$$

$$\bar{r}_{N,t}^y = \bar{r}_{N,t}^c + \mu^y \xi_{N,t} \quad (14)$$

où la variable indicateur de divergence $\xi_{N,t}$ est donnée par la formule suivante, où $\bar{r}_{N,t}^c$ est l'erreur de prédiction retour a-priori calculée par convolution:

$$\xi_{N,t} = \bar{r}_{N,t}^c + \lambda^{-N+1}\alpha_{N,t-1}\gamma_{N,t-1}\tilde{C}_{N+1,t}^{N+1} \quad (15)$$

Les paramètres μ^y et μ^b sont fixés à 1 [7]. L'algorithme FTF numériquement stable utilisant la régularisation et le rappel à zéro (désigné par FTFSR) complet est donné par les équations suivantes:

• Variables disponibles à l'instant t :

$$a_{N,t-1}, b_{N,t-1}, \tilde{C}_{N,t-1}, \gamma_{N,t-1}, \alpha_{N,t-1}, \beta_{N,t-1}, H_{N,t-1}$$

• Nouvelles informations: x_t et y_t



• Modélisations de x_t et x_{t-N} :

$$\bar{e}_{N,t} = x_t - a_{N,t-1}^T X_{N,t-1} \quad (16)$$

$$\alpha_{N,t} = \lambda \alpha_{N,t-1} + \gamma_{N,t-1} \bar{e}_{N,t}^2 + C \quad (17)$$

$$\gamma_{N+1,t} = \frac{\lambda \alpha_{N,t-1} + C}{\alpha_{N,t}} \gamma_{N,t-1} \quad (18)$$

$$\tilde{C}_{N+1,t} = \begin{bmatrix} 0 \\ \tilde{C}_{N,t-1} \end{bmatrix} - \frac{\bar{e}_{N,t}}{\lambda \alpha_{N,t-1}} \begin{bmatrix} 1 \\ -a_{N,t-1} \end{bmatrix} \quad (19)$$

$$a_{N,t} = (1-g) \{a_{N,t-1} - \bar{e}_{N,t} \gamma_{N,t-1} \tilde{C}_{N,t-1}\} \quad (20)$$

$$\bar{r}_{N,t}^c = x_{t-N} - b_{N,t-1}^T X_{N,t} \quad (21)$$

$$\xi_{N,t} = \bar{r}_{N,t}^c + \lambda^{-N+1} \alpha_{N,t-1} \gamma_{N,t-1} \tilde{C}_{N+1,t}^{N+1} \quad (22)$$

$$\bar{r}_{N,t} = \bar{r}_{N,t}^c + \xi_{N,t} \quad (23)$$

$$\gamma_{N,t} = \frac{\gamma_{N+1,t}}{1 + \gamma_{N+1,t} \bar{r}_{N,t}^c \tilde{C}_{N+1,t}^{N+1}} \quad (24)$$

$$\begin{bmatrix} \tilde{C}_{N,t} \\ 0 \end{bmatrix} = \tilde{C}_{N+1,t} - \tilde{C}_{N+1,t}^{N+1} \begin{bmatrix} -b_{N,t-1} \\ 1 \end{bmatrix} \quad (25)$$

$$b_{N,t} = (1-g) \{b_{N,t-1} - \bar{r}_{N,t} \gamma_{N,t} \tilde{C}_{N,t}\} \quad (26)$$

• Filtrage de y_t :

$$\bar{e}_{N,t} = y_t - H_{N,t-1}^T X_{N,t} \quad (27)$$

$$H_{N,t} = H_{N,t-1} - \bar{e}_{N,t} \gamma_{N,t} \tilde{C}_{N,t} \quad (28)$$

Le bon fonctionnement de l'algorithme FTFSR avec la parole dépend du choix de la constante C: une valeur trop faible peut s'avérer insuffisante pour éviter les blocages de l'algorithme, surtout après de longues périodes de silence. Notons aussi que dans les zones où les erreurs de prédiction deviennent très faibles la variance aller prend la valeur $C/(1-\lambda)$. Par ailleurs, en l'absence de régularisation ($C=0$), la valeur maximale de $\alpha_{N,t}$, obtenue avec un bruit blanc à l'entrée d'énergie σ_x^2 , vaut $\sigma_x^2/(1-\lambda)$. Le choix "du pire des cas" (zone de silence suivie d'une zone de signal blanc) est $C = \sigma_x^2$. Ce dernier choix ralentit beaucoup la vitesse de convergence; on pourra choisir la constante C de $\sigma_x^2/100$ à $\sigma_x^2/10$ selon que λ est plus ou moins proche de 1.

La figure 3 présente le résultat de l'identification de la réponse impulsionnelle de $N=256$ points par l'algorithme FTFSR. Le maximum et le minimum de l'énergie à court terme (256 échantillons) du signal de parole sont $(\sigma_x^2)_{\min} = 0.00002$ et $(\sigma_x^2)_{\max} = 0.89$.

Les constantes de régularisation et de rappel à zéro sont $C=0.008$ et $g=0.002$. Les autres paramètres sont les mêmes que ceux de la figure 2.

En comparant avec la figure 2 correspondant à la méthode de réinitialisation, on observe que l'algorithme FTFSR utilisant la régularisation et le rappel à zéro est asymptotiquement optimal pour le filtrage adaptatif. Par contre, la sous-optimalité se manifeste dans la convergence de l'algorithme qui est nettement ralentie par rapport à celle de l'algorithme avec réinitialisation. Ce ralentissement de la vitesse de convergence est dû à l'ajout de la constante de régularisation C à la variance aller, qui affaiblit considérablement le gain d'adaptation de l'algorithme. Pour améliorer la vitesse de convergence (dans le cas du filtrage adaptatif), on peut utiliser l'algorithme modifié proposé dans [8] et [7].

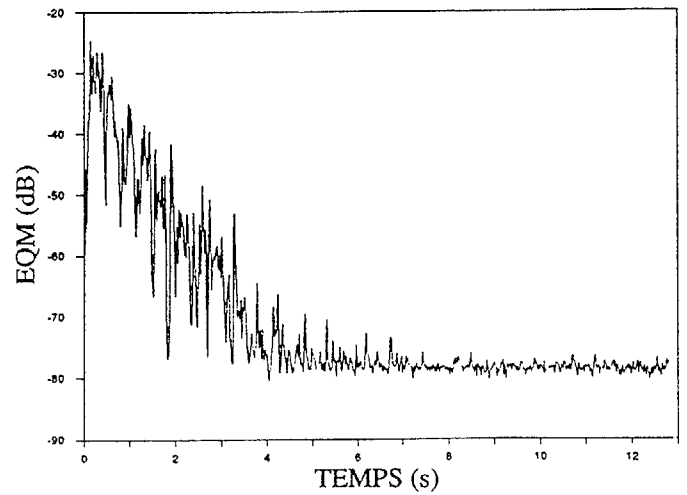


Figure 3: EQM de filtrage (algorithme FTFSR)

4. Conclusion

Nous avons étudié les algorithmes MCRNS quand le signal d'entrée est la parole; nous avons alors observé que les algorithmes MCRNS existant dans la littérature divergent. Ces divergences sont dues essentiellement au caractère fortement non stationnaire de la parole. Pour utiliser ces algorithmes avec le signal de parole, nous proposons deux solutions: la première exploite le fait que la divergence de certains algorithmes MCRNS se traduit par un blocage sur les paramètres déjà identifiés pour réinitialiser l'algorithme chaque fois qu'une tendance au blocage est détectée. La deuxième solution applique la régularisation de la variance aller et le rappel à zéro des prédicteurs aller/retour à un algorithme particulier de type FTF pour éviter les fortes discontinuités de certaines variables de l'algorithme lors de non-stationnarités importantes dans la parole.

Références bibliographiques

- [1] J.L.Botto, "Stabilization of Fast RLS Transversal Filters for Adaptive Filtering", IEEE ICASSP'87, Dallas.
- [2] A.Benallal, A.Gilloire et G.Faucon, "Méthode de stabilisation des algorithmes des Moindres Carrés Rapides appliqués au Traitement de la Parole", 11^e Colloque GRETSI, Juin 1987.
- [3] 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.
- [4] D.T.M.Slock and T.Kailath, "Numerically Stable Fast RLS Transversal Filters", IEEE ICASSP'88, New-York.
- [5] M.G.Bellanger, "Engineering Aspects of Fast RLS Algorithms in Transversal Adaptive Filters", IEEE ICASSP'87, Dallas.
- [6] J.M.Cioffi and T.Kailath, "Fast RLS Transversal Filters for Adaptive Filtering", IEEE Trans. ASSP-32, n° 2, April 1984.
- [7] 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.
- [8] A.Benallal and A.Gilloire, "Improvement of the Convergence Speed and tracking Capability of the Numerically Stable FLS Algorithms for Adaptive Filtering", IEEE ICASSP'89, Glasgow, UK.