



UNE FORME GÉNÉRIQUE POUR LES ALGORITHMES ADAPTATIFS DE TYPE GRADIENT STOCHASTIQUE A CHARGE DE CALCUL RÉDUITE ET CONVERGENCE AMÉLIORÉE

Katia HILAL

Mohsen MONTAZERI

*Pierre DUHAMEL**

CRPE (Unité mixte FRANCE TELECOM-CNET/CNRS), 38-40 rue du Général Leclerc
92131 Issy-Les-Moulineaux, France

*ENST/SIG 46, rue Barrault 75013 Paris, France

e-mail: duhamel@sig.enst.fr

RÉSUMÉ

Nous proposons une formulation générale permettant, à partir d'un algorithme de gradient stochastique, de développer une version équivalente à faible charge de calcul et une version à convergence améliorée. Les formules dérivées sont générales et s'appliquent aussi bien à l'algorithme LMS qu'aux algorithmes d'égalisation aveugle de type Bussgang [1]. Ces résultats reposent sur une écriture explicite de la récursivité de tout algorithme adaptatif.

ABSTRACT

A general derivation of block algorithms which are exactly equivalent to the sample by sample ones while allowing a noticeable reduction of their arithmetic complexity is provided. This result relies on an explicit writing of the inherent recursiveness of any adaptive algorithm. This block implementation is also shown to allow the derivation of new algorithms of the same type with improved convergence behavior. These results are applied to classical algorithms, but hold for the whole class of algorithms which is considered.

1 INTRODUCTION

Les résultats de cette communication sont généraux, et s'appliquent aussi en annulation d'écho acoustique. Ils seront cependant dérivés dans un contexte d'égalisation de canaux de communication, qui présente la plus grande généralité: Lorsque l'on connaît un algorithme pour des variables complexes, il est simple de le spécialiser pour des données réelles, alors que le contraire est difficile.

Il a été montré [1] que l'équation d'adaptation de beaucoup d'algorithmes utilisés en égalisation autodidacte était de la forme:

$$H_{n+1} = H_n - \mu_n X_n^* (y_n - g[y_n]) \quad (1)$$

où y_n est la sortie du filtre à l'instant n ($y_n = X_n^t H_n$), $g[y]$ une fonction généralement non-linéaire, dépendant de l'algorithme, μ_n le pas d'adaptation (généralement petit) à l'instant n et X^* le complexe conjugué de X . Cette équation caractérise des algorithmes dits "Bussgang". La non-linéarité de $g[y]$ est nécessaire si le canal à égaliser n'est pas à minimum de phase [2], ce qui est vrai dans la plupart des applications. La table (1) fournit les expressions de $g[y]$ pour les algorithmes suivants: Gradient stochastique (LMS) [3], Decision-Directed (DD) [4], Sato [5], Godard (CMA) [6]. Notons que l'algorithme LMS appartient à la famille considérée en dépit de la linéarité de la fonction correspondante: $g[y_n] = a_n$, a_n étant le signal de référence.

Les résultats seront établis à partir de la forme générique (1), qui permet d'aller assez loin dans les calculs. Les algorithmes spécifiques sont alors obtenus en remplaçant $g[y]$ par sa valeur précise. Des simplifications sont ensuite nécessaires, mais l'essentiel du problème est alors résolu. La méthodologie sera illustrée par les algorithmes LMS et CMA, pour lesquels on retrouve des variantes d'algorithmes déjà proposés [7,8,9,10].

2 La structure récursive des algorithmes adaptatifs

Les résultats présentés ici reposent sur une réécriture des algorithmes de type "Bussgang" en deux parties: D'une part le filtrage du signal d'entrée par un filtre à coefficients fixes pendant N échantillons et d'autre part une correction des résultats ainsi obtenus à l'aide d'un filtre récursif, dont les coefficients dépendent de la corrélation du signal.

L'équation d'adaptation d'un algorithme de gradient stochastique de type "Bussgang" se réécrit en fonction du gradient instantané $e_n = y_n - g[y_n]$:

$$H_{n+1} = H_n - \mu_n X_n^* e_n \quad (2)$$

où e_n , que nous appellerons "erreur" en référence à l'algorithme LMS est donné pour les algorithmes les plus fréquents par:

- $e_n^{\text{LMS}} = y_n - a_n$ pour l'algorithme LMS.



- $e_n^{DD} = y_n - \hat{y}_n$ pour l'algorithme "DD".
- $e_n^{CMA2-2} = y_n(|y_n|^2 - 1)$ pour le CMA 2-2.
- $e_n^{CMA1-2} = y_n - \frac{y_n}{|y_n|}$ pour le CMA 1-2.

Réécrivons l'équation (2) aux instants successifs $i = 0, 1, \dots, n$, en reportant successivement H_{i-1} dans l'expression du filtre suivant H_i , et en écrivant l'équation correspondante sur les sorties du filtre:

$$H_n = H_0 - \sum_{i=0}^{n-1} \mu_i X_i^* e_i \quad (3)$$

$$y_n = X_n^t H_n = X_n^t H_0 - \sum_{i=0}^{n-1} \mu_i X_n^t X_i^* e_i \quad (4)$$

ce qui correspond à Fig.(1): l'entrée est filtrée par H_0 , puis par une partie récursive qui commence par calculer l'"erreur" dont les valeurs successives sont pondérées par $\mu_i s_i(n)$. Les coefficients $s_i(n)$ sont des estimés du $i^{\text{ème}}$ coefficient de corrélation à l'instant n :

$$s_i(n) = X_n^t X_{n-i}^* \quad (5)$$

Notons que, dans le cas du LMS, le calcul de l'"erreur" nécessite la connaissance d'un signal de référence $g[y_n] = a_n$, non dessiné. Cette structure fournit donc les mêmes sorties $\{y_n\}$ que l'algorithme initial, ainsi que les erreurs successives $\{e_n\}$ correspondantes.

3 Algorithmes par blocs équivalents aux algorithmes initiaux

Bien que la méthode s'applique en toute généralité, nous nous restreignons au cas d'un pas d'adaptation fixe μ , afin d'alléger les notations.

Supposons que, comme c'est le cas dans beaucoup d'applications pratiques (égalisation, annulation d'écho), il ne soit pas utile de calculer en permanence les coefficients du filtre, mais seulement la sortie correspondante. Le paragraphe précédent a déjà fourni une manière d'obtenir la sortie du filtre adaptatif en évitant de calculer les coefficients. Cependant, comme le degré de la partie récursive augmente avec n , cette procédure ne peut être utilisée avec profit que sur des temps courts. Il nous faut donc être capable de calculer la valeur des coefficients que le filtre aurait eu à la fin de cette période pour pouvoir itérer la procédure.

Ce filtre H_N s'obtient facilement en accumulant les accroissements sur tout le bloc:

$$H_N = H_0 - \sum_{i=0}^{N-1} \mu_i X_i^* e_i \quad (6)$$

où e_i est l'"erreur" définie dans le paragraphe 2

On aura donc un algorithme par blocs au comportement strictement équivalent à la structure échantillon par échantillon en itérant la procédure suivante:

1. Faire fonctionner la structure de la Fig.(1) pendant N échantillons. Ceci fournit les mêmes échantillons de sortie et d'erreur que l'algorithme initial.

2. calculer H_N à partir de H_0 en accumulant les accroissements successifs (6).
3. Initialiser la structure avec H_N au lieu de H_0 , incrémenter l'indice temps de N , et itérer.

La charge de calcul n'a cependant pas diminué, au contraire, et seule la prise en compte des éléments suivants permettra de la réduire:

- 1) Le calcul de la partie récursive comporte d'abord un filtrage RIF à coefficients fixes:

$$Y_n^f = X_n^t H_{n-N+1} \quad (7)$$

qui peut se calculer par différents algorithmes rapides, à FFT courte si l'on tolère des retards importants, ou par "running FIR" sinon. La partie purement récursive nécessite le calcul des $s_i(n)$ (5). Une réduction de complexité ne pourra être obtenue que si ces estimés des coefficients de corrélation sont calculés eux-mêmes de manière récursive:

$$s_i(n) = s_i(n-1) + x_n x_{n-i}^* - x_{n-L} x_{n-i-L}^* \quad (8)$$

Globalement, le calcul effectué de cette manière sera beaucoup moins coûteux que le calcul de N itérations de l'algorithme initial.

2) Le calcul du filtre qui sera utilisé au bloc suivant (9) est du même type qu'un filtrage (produit d'une matrice Toeplitz par un vecteur), et les algorithmes rapides s'appliquent également. Comme la matrice X_n est commune à ces deux étapes, une partie des calculs est commune aux deux algorithmes rapides, et n'est donc pas répétée.

$$H_n = H_{n-N+1} - \mu X_n^* E_n \quad (9)$$

L'utilisation de ces techniques permet une réduction significative de la complexité de calcul, de l'ordre d'un facteur 4 pour des filtres de longueur $L = 256$ et une taille de bloc $N = 32$, ce qui n'introduit qu'un délai faible, acceptable dans beaucoup d'applications. Les algorithmes obtenus sont des variantes de [7], [8]. L'algorithme DD, très proche du LMS, pourra être l'objet des mêmes gains en complexité que celui-ci.

L'important est que cette réduction de complexité repose uniquement sur l'existence de récurrences sur les coefficients de corrélation. Les récurrences décrites par (8) sont mathématiquement exactes, mais engendrent une accumulation du bruit de calcul. On préférera donc, dans une utilisation pratique, utiliser une estimation de la corrélation à facteur d'oubli proche de 1 plutôt que celle à fenêtre rectangulaire donnée ici. On voit d'une part que ce type d'algorithme rapide "exact" existe pour TOUTE la classe considérée, et d'autre part que de nombreux autres algorithmes devraient aussi en bénéficier: Le principe de la réécriture sous forme d'une structure récursive est tout à fait général, seule l'existence de récurrences simples sur les coefficients de cette structure est à vérifier.

4 Algorithmes à convergence accélérée

La réécriture par blocs décrite au paragraphe précédent permet également d'améliorer la convergence des algorithmes



initiaux: on dispose des erreurs observées de l'algorithme initial sur une fenêtre de N échantillons, et on peut alors raffiner l'adaptation en minimisant l'erreur observée sur un bloc:

$$J_n^{\text{block}} = \frac{1}{N} \sum_{i=n-N}^{n-1} J_i \quad (10)$$

où J_i est l'approximation stochastique correspondant à l'algorithme échantillon par échantillon initial.

Conceptuellement, l'algorithme comporte deux étapes: 1) Le calcul de la version "rapide" de l'algorithme initial, conformément à (9). 2) La correction du filtre ainsi obtenu en minimisant (gradient stochastique) la somme quadratique des N dernières évaluations du critère. Globalement, on aura:

$$H_n = H_{n-N+1} - \mu \mathbf{X}_n^* e_n - \mu_{\text{block}} \nabla J_n^{\text{block}} \quad (11)$$

A titre d'exemple, on obtient pour le CMA:

$$J_n^{\text{block}} = \frac{1}{N} \sum_{i=n-N+1}^n (|y_i|^2 - 1)^2 \quad (12)$$

Ce critère dépendant de variables complexes, le calcul de sa dérivée est lourd. Ce calcul se trouve grandement simplifié par les résultats de [11], où il est montré que la direction de plus grande pente de $J(H, H^*)$ par rapport à H est fournie par la dérivée de J par rapport à H^* , H et H^* étant considérés comme indépendants. D'une manière générale, le gradient du critère bloc J^{block} s'écrit:

$$\nabla J_n^{\text{block}} = \frac{1}{N} (\tilde{\mathbf{D}}F + \mathbf{D}K) \quad (13)$$

F et K étant deux vecteurs qui se déduisent en dérivant le critère par blocs correspondant à l'algorithme, \mathbf{D} et $\tilde{\mathbf{D}}$ étant définis comme suit:

$$\begin{aligned} \mathbf{D} &= \left[\frac{\partial y_{n-N+1}}{\partial H^*}, \frac{\partial y_{n-N+2}}{\partial H^*}, \dots, \frac{\partial y_n}{\partial H^*} \right] \\ \tilde{\mathbf{D}} &= \left[\frac{\partial y_{n-N+1}^*}{\partial H^*}, \frac{\partial y_{n-N+2}^*}{\partial H^*}, \dots, \frac{\partial y_n^*}{\partial H^*} \right] \end{aligned}$$

Dans le cas du CMA, on a:

$$\begin{aligned} \nabla J_n^{\text{block}} &= \frac{2}{N} \sum_{i=n-N+1}^n [y_i (|y_i|^2 - 1) \frac{\partial y_n^*}{\partial H^*} \\ &+ y_i^* (|y_i|^2 - 1) \frac{\partial y_n}{\partial H^*}] \end{aligned}$$

et les vecteurs F et K sont respectivement le vecteur d'"erreur" E_n et son complexe conjugué E_n^* .

On peut également montrer que:

$$\begin{aligned} \nabla J_n^{\text{block}} &= \mathbf{X}_n^* \mathbf{B}_n^\dagger E_n + \tilde{\mathbf{G}}_n (\mathbf{I} - \mathbf{B}_n^\dagger) E_n \\ &+ \mathbf{G}_n (\mathbf{I} - \mathbf{B}_n^\dagger) E_n^* \end{aligned} \quad (14)$$

où les matrices \mathbf{G}_n et $\tilde{\mathbf{G}}_n$ caractérisent l'opérateur $g[\cdot]$ par l'intermédiaire de ses dérivées par rapport à H^* durant tout le bloc de taille N :

$$\begin{aligned} \mathbf{G}_n &= \left[\frac{\partial g[y_{n-N+1}]}{\partial H^*}, \frac{\partial g[y_{n-N+2}]}{\partial H^*}, \dots, \frac{\partial g[y_n]}{\partial H^*} \right] \\ \tilde{\mathbf{G}}_n &= \left[\frac{\partial g^*[y_{n-N+1}]}{\partial H^*}, \frac{\partial g^*[y_{n-N+2}]}{\partial H^*}, \dots, \frac{\partial g^*[y_n]}{\partial H^*} \right] \end{aligned}$$

et où $\mathbf{B}_n = (\mathbf{I} + \mu \mathbf{S}_n)^{-1}$, \mathbf{S}_n étant la matrice triangulaire inférieure de dimension $N \times N$ avec des zéros sur la diagonale et dont l'élément i, j est égal au coefficient de corrélation $s_i(n - N + j + 1)$ de la section 3.

Cette équation s'applique à tous les algorithmes de type "Busgang", et la charge de calcul nécessaire à son évaluation, bien que raisonnable est néanmoins importante. On peut montrer qu'elle se réduit à des expressions très simples pour les algorithmes LMS, Sato, et DD. Dans le cas du CMA, une approximation raisonnable (négliger les termes en μ^2 par rapport aux termes en μ), permet également de réduire les calculs.

Appliquons cette technique aux algorithmes classiques: dans le cas du LMS, \mathbf{G}_n et $\tilde{\mathbf{G}}_n$ sont nulles, et on a alors:

$$H_n = H_{n-N+1} - \mu X_n^* e_n - \frac{\mu_{\text{block}}}{N} \mathbf{X}_n^* \mathbf{B}_n^\dagger E_n$$

qui est une version légèrement différente du *Nested LMS* proposé par Benesty *et al* [7]. La non linéarité, dans le cas des algorithmes DD et Sato est en marche d'escalier (voir table (1)), donc sa dérivée est nulle, sauf en un nombre fini de points. Dans ce cas également, on prendra \mathbf{G}_n and $\tilde{\mathbf{G}}_n$ égales à zéro, et l'algorithme sera très proche du *Nested LMS* (sauf le calcul de l'erreur).

Dans le cas général, on doit exprimer les matrices \mathbf{G}_n et $\tilde{\mathbf{G}}_n$ en fonction des données disponibles, et pour cela, on dispose des relations générales suivantes:

$$\mathbf{G}_n = \mathbf{D}\mathbf{A} + \tilde{\mathbf{D}}\mathbf{C} \quad (15)$$

$$\tilde{\mathbf{G}}_n = \tilde{\mathbf{D}}\mathbf{A}^* + \mathbf{D}\mathbf{C}^* \quad (16)$$

$$\mathbf{D} = \mathbf{G}_n (\mathbf{I} - \mathbf{B}_n^\dagger) \quad (17)$$

$$\tilde{\mathbf{D}} = \mathbf{X}_n^* \mathbf{B}_n^\dagger + \tilde{\mathbf{G}}_n (\mathbf{I} - \mathbf{B}_n^\dagger) \quad (18)$$

où \mathbf{A} et \mathbf{C} sont des vecteurs découlant de la relation: $\frac{\partial g[y_i]}{\partial H^*} = \frac{\partial y_i}{\partial H^*} a_i + \frac{\partial y_i^*}{\partial H^*} c_i$. Les équations (17) et (18) découlent de la relation récursive entre les sorties du filtre y_n et les fonctions $g[y_n]$.

Tout calcul fait, on obtient pour les CMA 2-2 et 1-2:

$$\begin{aligned} H_n &= H_{n-N+1} - \mu \mathbf{X}_n^* E_n \\ &- \frac{\mu_{\text{block}}}{N} \mathbf{X}_n^* \mathbf{B}_n^\dagger \{ E_n \\ &+ [\mathbf{A} + \mathbf{C}\mathbf{P}(\mathbf{I} - \mathbf{B}_n^\dagger)\mathbf{C}^*] \mathbf{P}^* (\mathbf{I} - \mathbf{B}_n^\dagger) E_n \\ &+ [\mathbf{C} + \mathbf{A}\mathbf{P}^*(\mathbf{I} - \mathbf{B}_n^\dagger)\mathbf{C}] \mathbf{P} (\mathbf{I} - \mathbf{B}_n^\dagger) E_n^* \} \end{aligned} \quad (19)$$

où $\mathbf{P} = [\mathbf{I} - (\mathbf{I} - \mathbf{B}_n^\dagger)\mathbf{A}]^{-1}$.

Notons que le CMA tout comme le Bloc-CMA traditionnels sont des cas particulier de cette équation, lorsque μ_{block} ou μ sont nuls, respectivement.

5 CONCLUSION

Plusieurs variantes d'algorithmes adaptatifs ont récemment été proposées, qui permettaient à la fois une réduction de la charge de calcul et une accélération de la convergence par rapport aux algorithmes initiaux. Nous avons montré que cette approche s'inscrivait dans une démarche générale, et que tout un ensemble d'algorithmes pouvait être "travaillés" de cette manière. Dans [7] on vérifie sur les courbes de convergence du Nested-LMS l'amélioration de



vitesse de convergence par rapport au LMS est sensible et on montre théoriquement la stabilité de cet algorithme. La Fig.(2) montre l'évolution du critère d'"œil ouvert" pour une égalisation par un CMA 1-2. Ce critère mesure l'Interférence Entre Symbole (IES) résiduelle dans le système et descend en dessous de la valeur 1 lorsque l'œil est complètement ouvert (c-à-d lorsque les trajets résiduels sont négligeables par rapport au trajet dominant). La simulation a été effectuée sur un canal à six trajets ayant un retard maximum de 5 temps symbole. Le filtre égaliseur est de longueur 32 et la longueur du bloc est égale à 4. Les courbes montrent clairement que l'on a gagné 20% d'itérations afin d'ouvrir complètement l'œil tout en restant à un niveau d'IES inférieur par rapport au CMA 1-2. Une étude détaillée montrerait que cette approche élargit la gamme des compromis vitesse de convergence / erreur résiduelle possible.

References

- [1] J. G. Proakis and C. L. Nikias, "Blind equalization," in *SPIE, Adaptive Signal Processing*, (San Diego), July 22-24 1991, pp. 76-87.
- [2] A. Benveniste, M. Goursat, and G. Ruget, "Robust identification of a nonminimum phase system: Blind adjustment of a linear equalizer in data communications," *IEEE Trans. on Aut. Cont.*, vol. AC-25, no. 3, pp. 385-399, June 1980.
- [3] B. Widrow and J. Maccool, "A comparison of adaptive algorithms based on the methods of steepest descent and random search," *IEEE Trans. on A.P.*, vol. 24, no. 5, pp. 615-637, Sept. 1976.
- [4] R. W. Lucky, "Techniques for adaptive equalization of digital communication systems," *Bell System Technical Journal*, vol. 45, pp. 255-286, Feb. 1966.
- [5] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Trans. on Com.*, pp. 679-682, June 1975.
- [6] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. on Com.*, vol. COM-28, no. 11, pp. 1867-1875, Nov. 1980.
- [7] J. Benesty, L. S. Wen, and P. Duhamel, "A gradient-based adaptive algorithm with reduced complexity, fast convergence and good tracking characteristics," in *ICASSP*, (San Francisco), Mar 23-26, 1992, pp. IV.5-IV.8.
- [8] J. Benesty and P. Duhamel, "A fast constant modulus adaptive algorithm," *IEE Proc. Special issue on adaptive filtering*, vol. 138, no. 4, pp. 379-387, Aug. 1991.
- [9] K. Hilal and P. Duhamel, "Amélioration de la convergence de l'algorithme à Module Constant pour la déconvolution aveugle," in *Treizième colloque GRETSI*, (Juan-les-Pins, France), 16 au 20 Sept 1991, pp. 485-488.
- [10] K. Hilal and P. Duhamel, "A convergence study of the Constant Modulus Algorithm leading to a Normalized-CMA and a Block-Normalized-CMA," in *EUSIPCO, European Sig. Proc. Conf.*, (Brussels), Aug 24-27, 1992, pp. 135-138. and submitted to *IEEE Trans. on Sig. Proc.*
- [11] D. H. Branwood, "A complex gradient operator and its application in adaptive array theory," *IEE proc.*, vol. 130, no. 1, pp. 11-16, Feb. 1983. Pts. F and H.

Algorithme	Non linéarité $g[y]$
DD	\hat{y}
Sato	$\gamma \text{sgn}(y)$
Stop-and-Go	$y - \frac{1}{2} [(f^R + f^I)\hat{e} + (f^R - f^I)\hat{e}]$
Benveniste-Goursat	$y - [k_1\hat{e} + k_2 \hat{e} \hat{e}]$
Godard $q = 2$ (CMA p-2)	$\frac{y}{ y } \{ y + y ^{p-1} - y ^{2p-1} \}$

Table 1: Operateur $g[\cdot]$ pour les différents algorithmes adaptatifs.

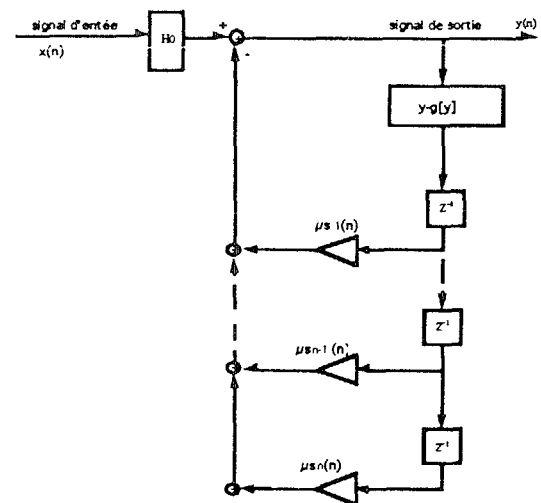


Figure 1: Structure récurrente générale.

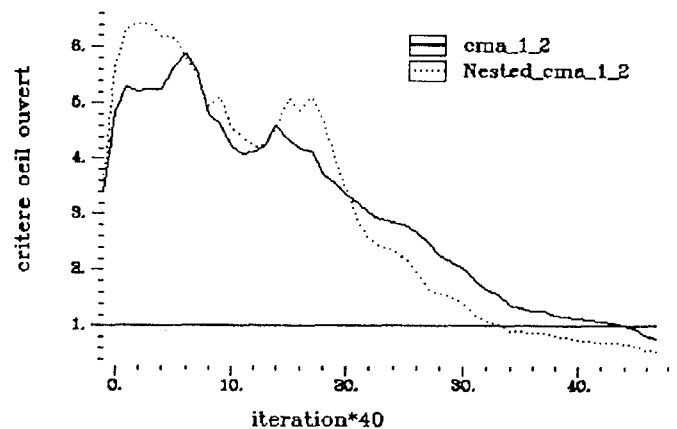


Figure 2: Evolution du critère pour le CMA 1-2 et le Nested CMA 1-2.