

DIXIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

459



NICE du 20 au 24 MAI 1985

ETUDE DES EFFETS DE QUANTIFICATION SUR LES ALGORITHMES EN TREILLIS ADAPTATIFS.

JEAN-FRANCOIS AGNEL (*) - JEAN-YVES LUCAS

THOMSON-CSF AVS-RCM/GEM 178, boulevard GABRIEL PERI 92242 MALAKOFF CEDEX
ENST - DEPARTEMENT "SYSTEMES et COMMUNICATIONS" - CNRS UA 820 46, rue BARRAULT 75634 PARIS CEDEX 13
(*) membre du GRECO 69 "SARTA"

RESUME

SUMMARY

L'existence d'algorithmes rapides pour les traitements adaptatifs du signal, d'une part, et les progrès réalisés dans le domaine des circuits intégrés, d'autre part, rendent possible aujourd'hui l'implantation de tels algorithmes sur des microprocesseurs ou des circuits spécialisés, et ce pour des applications temps réel. Cependant, l'utilisateur est confronté aux problèmes liés à la quantification, aux propagations d'erreur, ...

Dans cette communication, nous étudions, par simulations, les effets de la quantification sur les algorithmes adaptatifs en treillis (algorithmes en moindres carrés exacts et algorithmes en gradient) lorsque les signaux analysés et les variables internes sont codés sur des mots-machine de longueurs limitées. L'étude des algorithmes montre que l'implantation est caractérisée par trois paramètres, dont nous examinons les influences respectives: nombre de bits pour le codage des coefficients de réflexion, pour les variables de vraisemblance, pour le signal d'entrée. Ce dernier paramètre détermine en particulier les nombres de bits nécessaires aux codages des résidus et de leurs énergies.

Notre approche consiste à comparer les performances d'un algorithme implanté en virgule fixe à celles obtenues avec une programmation en virgule flottante simple précision, nous servant de référence. A l'aide de quantités telles que l'énergie de l'erreur et la blancheur du résidu, nous pouvons mesurer, par rapport à la référence, les dégradations des performances introduites par la quantification.

Les simulations nous permettent d'évaluer la robustesse des treillis adaptatifs aux effets de quantification et mettent en évidence une configuration optimale d'implantation, compromis entre un nombre de bits peu élevé pour le codage des diverses variables, et une dégradation des performances faible.

En plus du coût opératoire, de la vitesse de convergence, .., la dégradation des performances liée à l'implantation est un nouvel outil pour la comparaison des algorithmes de filtrage adaptatif en treillis.

Fast algorithms in adaptive signal processing and new improvement in VLSI domain allow implementation of such algorithms on microprocessors or specialized circuits for real time applications. However, engineers have to face many problems: quantization, error propagation,...

In this communication, we study, by means of simulations, the effects of quantization in the adaptive lattice algorithms (least squares and gradient) when signal and internal variables are implemented on finite binary words.

We compare the performances of a fixed point implemented algorithm with these of a floating point implemented version used as a reference. The tool of the comparison is a measure of the degradations introduced by quantization. The measure is performed on prediction error and residual whiteness.

The simulations point an optimal implementation configuration which realizes a compromise between a low number of bits and small performance degradations.

Besides number of operations, convergence rate,... the performance degradations linked with an implementation is a new tool for the comparison of adaptive lattice algorithms.



1.0 INTRODUCTION

Dans de nombreuses applications de traitement du signal, l'évolution des caractéristiques spectrales du signal nécessite l'utilisation d'algorithmes adaptatifs permettant de suivre l'évolution de ses propriétés. L'utilisateur est généralement amené à supposer que l'évolution du signal est suffisamment lente pour permettre l'utilisation d'algorithmes temporellement récurrents, plus souples d'utilisation qu'une modélisation non-stationnaire du signal. Diverses approches peuvent être envisagées pour l'optimisation des paramètres du modèle. Elles correspondent d'une part à la structure du filtre (transverse ou treillis) et d'autre part à la technique d'actualisation des coefficients (algorithmes en moindres carrés exacts, gradient,...).

Les propriétés des coefficients de réflexion vis à vis de l'ordre de modélisation, de la quantification, des critères de stabilité rendent le filtre en treillis plus attrayant que la structure transverse associée aux coefficients de prédiction. De plus, l'utilisation des techniques récursives conduit à des algorithmes plus souples d'utilisation, dont la stabilité n'est pas altérée par l'introduction d'un facteur d'oubli.

L'ensemble de ces algorithmes a aujourd'hui un coût opératoire comparable, en $\alpha\pi$ et les progrès réalisés dans le domaine des circuits intégrés rendent possible leur implantation sur des microprocesseurs ou des circuits spécialisés, et ce pour des applications temps réel. Cependant, l'utilisateur est confronté aux problèmes liés à la quantification, aux propagations d'erreur,... qui dégradent leurs performances.

Dans cet article, nous étudions, par simulations, les effets de la quantification sur les algorithmes adaptatifs en treillis (algorithmes en moindres carrés exacts et algorithmes en gradient) lorsque les signaux analysés et les variables internes sont codés sur des mots-machine de longueurs limitées.

Notre approche consiste à comparer les performances d'un algorithme implanté en virgule fixe à celles obtenues avec une programmation en virgule flottante simple précision, nous servant de référence. Il convient pour cela de définir une mesure des performances d'un algorithme. La structure treillis étant étroitement liée à la prédiction linéaire et au blanchiment, il semble naturel d'utiliser, pour évaluer cette performance, l'énergie de l'erreur de prédiction et une grandeur décrivant son pouvoir blanchissant: la platitude spectrale du résidu. A l'aide de ces quantités, nous pouvons mesurer, par rapport à la référence, la dégradation des performances introduite par la quantification.

Les simulations effectuées nous permettent d'évaluer la robustesse des treillis adaptatifs aux effets de quantification et mettent en évidence une configuration optimale d'implantation, compromis entre un nombre de bits peu élevé pour le codage des diverses variables, et une dégradation des performances faible.

En plus du coût opératoire, de la vitesse de convergence, .., la dégradation des performances liée à l'implantation est un nouvel outil pour la comparaison des algorithmes de filtrage adaptatif en treillis.

2.0 PRESENTATION des ALGORITHMES:

L'ensemble des algorithmes de filtrage récursif en treillis a largement été étudié dans la littérature,

en particulier par Friedlander /2/. Les simulations que nous avons effectuées ont porté sur quatre algorithmes: deux algorithmes exacts, avec fenêtre antérieure -FA- (pre-windowed) et sur fenêtre glissante -FG- (covariance lattice), et deux gradients -G1- et -G2- respectivement à 1 et 2 jeux de coefficients de réflexion. Afin d'illustrer notre démarche pour le codage des variables, nous donnons ici la formulation de l'algorithme avec fenêtre antérieure (sans facteur d'oubli). Les autres algorithmes sont présentés dans les références /1/ et /2/

Algorithme avec Fenêtre Antérieure

Initialisation:

$$(1) \begin{matrix} e_{o,t} = y_t \\ r_{o,t} = y_t \end{matrix} \quad (2) \begin{matrix} E_{o,t} = E_{o,t-1} + y_t^2 \\ R_{o,t} = R_{o,t-1} + y_t^2 \end{matrix} \quad (3) g_{o,t} = 1$$

Récurrance sur l'ordre:

ré-actualisation de l'intercorrrelation:

$$(4) D_{n,t} = D_{n,t-1} + e_{n,t} r_{n,t-1} / g_{n,t-1}$$

coefficients de réflexion:

$$(5) K_{n,t}^* = D_{n,t} / E_{n,t} \quad K_{n,t}^r = D_{n,t} / R_{n,t-1}$$

$$(6) \text{ vraisemblance: } g_{n+1,t} = g_{n,t} - r_{n,t}^2 / R_{n,t}$$

résidus et énergies:

$$(7) \begin{matrix} e_{n+1,t} = e_{n,t} - K_{n,t}^* r_{n,t-1} \\ r_{n+1,t} = r_{n,t-1} - K_{n,t}^r e_{n,t} \end{matrix}$$

$$(8) \begin{matrix} E_{n+1,t} = E_{n,t} - K_{n,t}^* D_{n,t} \\ R_{n+1,t} = R_{n,t-1} - K_{n,t}^r D_{n,t} \end{matrix}$$

3.0 DYNAMIQUE et CODAGE des VARIABLES:

Dans tous les algorithmes étudiés ici, le signal y_t est, après éventuellement une normalisation, une variable entière codée sur un nombre de bits NBY.

Cette quantité paramètre la dynamique du signal et conditionne le codage d'un certain nombre de variables de l'algorithme, directement liées au signal: les résidus (7), leurs énergies (8) et leur intercorrrelation (4). Les coefficients de réflexion (5) et la variable associée à la vraisemblance (6) (intervenant uniquement dans les algorithmes exacts) sont des grandeurs comprises entre -1 et +1 (en tronquant les coefficients de réflexion lors d'éventuels dépassements) et peuvent être considérés comme des réels codés respectivement sur NBK et NBG bits.

Les performances des algorithmes seront donc fonction des trois paramètres d'implantation NBY, NBK et NBG.

Le facteur d'oubli a été pris égal à 0,9921 soit 127/128 et codé sur NBL=7 bits, ce qui a l'avantage de permettre un codage sans erreur et de représenter le facteur d'oubli par un mot ne comportant que des 1. La longueur de la fenêtre glissante a été prise égale à 127 points, le pas des gradients égal à 1.

3.1 Codage des Energies Résiduelles: - Suivant les algorithmes, le calcul des énergies diffère mais dans tous les cas, ce sont les variables qui ont la plus grande dynamique. En effet, elles se présentent comme une somme cumulée, homogène au carré du signal. Prenons l'exemple de l'énergie résiduelle directe; pour les treillis exacts, nous avons:

$$\text{FA: } E_{o,t} = \lambda E_{o,t-1} + y_t^2 \quad (2)$$

$$\text{FG: } E_{o,t} = E_{o,t-1} + y_t^2 - y_{t-1}^2$$

avec dans les deux cas:

$$E_{n+1,t} = E_{n,t} - K_{n,t}^* D_{n,t}$$

Et pour les gradients:

$$G1: E_{n,t} = \lambda E_{n,t-1} + e_{n,t}^2 + r_{n,t}^2$$

$$G2: E_{n,t} = \lambda E_{n,t-1} + e_{n,t}^2$$

Si NBY est le nombre de bits du signal, y_t vérifie:

$$0 \leq y_t \leq 2^{NBY-1}$$

d'où:

$$0 \leq y_t^2 \leq 2^{(2NBY-1)-1}$$

La somme cumulée s'effectue soit sur une fenêtre de longueur finie soit sur une fenêtre de longueur infinie mais pondérée par un oubli exponentiel. L'équivalence R_1 /1/ entre une longueur fixe et un oubli exponentiel consiste à écrire l'égalité des fenêtres en amplitude. Elle conduit à la relation:

$$\lambda = (L-1)/L$$

Prenons $L=2^{NBL}$ où NBL représente le nombre de bits nécessaires au codage de l'oubli exponentiel. Ainsi, la somme cumulée sur une fenêtre équivalente à L vérifie les inégalités:

$$0 \leq E_t \leq 2^{(2NBY-1)-1} \cdot 2^{NBL}$$

c'est-à-dire une dynamique de $2NBY+NBL-2$ bits, plus un bit de signe (pour l'intercorrélacion) soit un codage sur $2NBY+NBL-1$ bits.

Nous avons fait ce calcul pour l'énergie en entrée du treillis, la somme cumulée portant sur le signal. Il s'agit de savoir si le codage des énergies à chaque étage peut être réalisé sur le même nombre de bits. Pour cela, ré-écrivons la relation de passage de l'ordre n à l'ordre n+1. L'égalité (8) devient, compte tenu de la définition du coefficient de réflexion (5):

$$E_{n+1,t} = E_{n,t} (1 - K_{n,t}^r K_{n,t}^e)$$

où le coefficient $K_{n,t}^r \cdot K_{n,t}^e$ est positif et inférieure à 1; ainsi, nous avons:

$$E_{n,t} \leq E_{0,t}$$

de sorte que les énergies à chaque étage du treillis peuvent être codées sur des mots de même longueur que l'énergie en entrée.

Pour un signal codé sur 8 bits avec $NBL=7$, nous pouvons prendre des mots de 22 bits pour les énergies. Or les simulations effectuées nous indiquent que par rapport aux $2NBY-1$ bits du carré du signal, l'accumulation nécessite 5 bits supplémentaires (ce pour un oubli égal à $127/128$), ce qui donne un codage des énergies sur $2NBY+4$ bits soit, pour $NBY=8$, 20 bits pour les énergies et l'intercorrélacion. Nous avons, en prenant 22 bits, surestimé, d'un facteur 4, l'effet d'accumulation en considérant le cas critique où sur la totalité de la fenêtre le signal est en saturation.

3.2 Codage des Résidus: - Deux points sont à mentionner. Les équations livrant les résidus, du type de la relation (7), montrent que le caractère récursif de l'algorithme vis à vis de l'ordre peut entraîner un phénomène d'accumulation sur les résidus nécessitant dans le pire des cas p bits de garde pour un ordre égal à p. Cependant, les variables que nous appelons "énergie" et qui constituent des sommes cumulées d'énergie résiduelle sont codées au maximum

sur $2NBY+NBL-1$ bits; par conséquent, la dynamique des résidus permet un codage sur $NBY+NBL/2$ bits soit pour $NBL=7$ un nombre de bits égal à $NBY+3$. Ainsi, par rapport au nombre de bits utilisés pour le codage du signal, nous sommes amenés à introduire des bits de garde pour le codage des résidus. Ce nombre de bits est majoré par $NBL/2$ qui est indépendant de l'ordre du filtre.

D'autre part, l'énergie résiduelle $E_{n,t}$ diminuant pour n croissant, les résidus correspondants ont en moyenne une dynamique de plus en plus faible, de sorte que l'estimation des coefficients d'ordre supérieur est erronée. Dans nos simulations, pour un signal codé sur 8 bits, avec un treillis comportant 6 cellules, le dernier résidu avait une dynamique de 4 bits, ce qui rendait impossible une mesure fine de ses caractéristiques, énergie ou blancheur, et par conséquent des performances de l'algorithme. Pour limiter cette imprécision, nous introduisons dans l'ensemble des algorithmes un facteur d'échelle correspondant à un décalage de NER bits sur les résidus et ayant en quelque sorte les effets d'une virgule flottante. Nous introduisons, par le biais de ce décalage, NER bits de précision sur les résidus. Ce facteur peut être optimisé pour chaque étage du filtre; sa valeur dépend de l'ordre et résulte d'un compromis entre précision et longueur de registre.

Cette variable est, après NBY, NBK et NBG, le quatrième paramètre de l'implantation de l'algorithme.

4.0 PRESENTATION des SIGNAUX de TEST:

Nous avons mis en oeuvre les algorithmes paramétrés par NBY, NBK, NBG, NER sur divers types de signaux:

- a. des signaux synthétiques autorégressifs: dans un premier temps, les signaux ont été obtenus comme la sortie des modèles autorégressifs d'ordre 2 proposés par Friedlander /2/ avec pour entrée une séquence de bruit pseudo-blanc. Les modèles ont pour fonction de transfert:

$$A_1(z) = 1 - 1,6 z^{-1} + 0,95 z^{-2}$$

$$A_2(z) = 1 - 1,8 z^{-1} + 0,95 z^{-2}$$

Dans un second temps, les simulations ont été faites sur 20 modèles autorégressifs d'ordre 6 dont les pôles étaient tirés aléatoirement dans une couronne définie par:

$$0,8 < \rho < 0,97 \quad \text{et} \quad \pi/16 < \theta < 15\pi/16$$

Les résultats ont été obtenus sur 25 réalisations de bruit d'entrée.

- b. des signaux réels de fouillis de sols.

Dans ces simulations, nous avons fourni suffisamment d'échantillons à l'algorithme pour que ne se pose pas le problème de la convergence préalable. La mesure des performances est effectuée après un temps de convergence de 200 points.

5.0 MESURES des PERFORMANCES:

Les performances des algorithmes sont décrites par deux mesures calculées sur le résidu de la modélisation:

- a. l'énergie de l'erreur résiduelle;
- b. la blancheur du résidu, mesurée par le rapport des valeurs propres extrémales de la matrice de corrélation (dynamique spectrale) /1/.

Sur l'ensemble des réalisations de bruit, pour chaque type de signal, ont été calculés la moyenne et



l'écart-type des mesures. Celles-ci sont exprimées en pourcentages de dégradation par rapport au résultat acquis avec l'algorithme flottant simple précision et sont fonction des quatre paramètres de l'implantation.

6.0 RESULTATS des SIMULATIONS sur les MODELES d'ORDRE 2:

La planche 1 illustre les simulations effectuées sur les modèles d'ordre 2 introduits par Friedlander; elle indique les pourcentages de dégradation obtenus avec les divers algorithmes implantés, pour chaque modèle, sur l'énergie et la blancheur du résidu.

6.1 Influence de NBY: - Au vu des simulations, il apparaît que les performances sont d'autant meilleures que le nombre de bits servant au codage du signal est plus grand. Cependant, ce paramètre est généralement imposé par l'expérience et constitue une donnée dont l'utilisateur n'est pas maître: utilisation de convertisseur spécifique dans une chaîne d'acquisition,... Pour la suite des expériences, nous avons fixé NBY égal à 8, ce qui implique un codage des énergies sur 20 bits.

6.2 Facteur d'Echelle NER sur les Résidus: - Ce paramètre a été introduit afin d'augmenter la dynamique des résidus. Les simulations ont montré que l'on a intérêt à augmenter le plus possible NER mais il faut toutefois que ce supplément ne soit pas coûteux (architecture, temps de calcul pour certains types de processeur). Dans nos simulations, sur la seule précision des mesures, il est apparu que NER=4 est une valeur suffisante ce qui donne des registres de 12 bits pour les résidus.

6.3 Influence de NBG: - Ce paramètre n'intervient que pour les algorithmes exacts. Les simulations ont été faites avec NBG = 4 / 8 / 12 / 16 . Pour ces valeurs, l'influence de NBG sur les performances des algorithmes est peu importante en regard des variations obtenues lorsque NBK varie. Nous présentons par la suite les résultats acquis avec NBG=8.

6.4 Influence de NBK: - Les simulations ont été faites avec diverses valeurs de NBK (4, 6, 8, 10, 12, 16). Nous allons examiner successivement les algorithmes exacts et les gradients:

a. algorithmes exacts:

pour l'ensemble des critères, on observe sur les courbes ci-jointes un point de rupture pour NBK=8 au delà duquel les performances ne sont plus modifiées. Les deux algorithmes (FA et FG) réagissent de manière pratiquement identique pour l'ensemble des modèles. Des écarts-types faibles valident les moyennes obtenues, à l'exception de NBK=4 pour lequel on obtient un écart-type sur la mesure relativement élevé en raison de la dispersion de la convergence de l'algorithme.

Pour un nombre très faible de bits sur les coefficients de réflexion, NBK=4, les résultats obtenus avec NBG = 12 ou 16 sont moins bons que ceux acquis avec NBG = 4 ou 8: en effet, on exige une précision beaucoup trop grande pour la vraisemblance (qui traduit l'exactitude du modèle à chaque instant) par rapport à la précision des coefficients de réflexion qui illustrent la convergence: la sensibilité de l'algorithme est trop fine en regard de la mesure de l'exactitude qui en est faite.

b. gradients:

les comportements des deux algorithmes sont très voisins sur l'ensemble des modèles. Nous constatons de très fortes dégradations supérieures

à 10% pour l'énergie et 15% pour la blancheur pour un nombre de bits NBK inférieur à 12. Ce n'est que pour 16 bits que les performances des gradients sont comparables à celles des treillis exacts. Nous observons par ailleurs un très fort écart-type sur la mesure pour les valeurs de NBK inférieures à 8. Ce phénomène est lié à la non-convergence des algorithmes pour un nombre trop faible de bits sur les coefficients de réflexion.

6.5 Conclusion des Simulations sur les Modèles d'Ordre 2:

- Il semble donc que, sur ces signaux synthétiques, les treillis exacts soient plus robustes que les gradients pour une quantification des coefficients de réflexion sur un nombre faible de niveaux. Pour 8 bits sur le signal, en choisissant un facteur d'échelle sur les résidus de 4 bits (NER), la configuration (NBG,NBK)=(8,8) donne de bonnes performances avec les algorithmes exacts. Il faut prendre NBK supérieur à 12 et en toute sécurité égal à 16 pour les gradients.

Nous allons examiner maintenant si ces lignes générales se confirment sur des modèles synthétiques tirés aléatoirement et sur des signaux réels.

7.0 RESULTATS des SIMULATIONS sur les MODELES ALEATOIRES d'ORDRE 6:

Sur 20 modèles, l'ensemble des résultats précédents est confirmé. Les courbes obtenues ont la même forme bien que l'échelle des ordonnées dépende très étroitement de la nature du signal. Ainsi, nous observons une grande dispersion des courbes en fonction de la position des pôles dans le cercle unité. Quoiqu'il en soit, nous obtenons pour les algorithmes exacts un palier pour NBK ≥ 8 et de fortes dégradations pour un nombre de bits inférieur. Pour les gradients, il faut prendre NBK=16.

8.0 RESULTATS sur SIGNAUX REELS de FOUILLIS:

Le fouillis en signal radar est dû aux diverses réflexions parasites du signal sur le sol, la végétation, la mer, les perturbations atmosphériques,... Il constitue la principale limitation des systèmes de détection. Les résultats présentés dans cette communication entrent dans le cadre plus large de l'étude de l'utilisation des algorithmes récursifs en treillis pour le filtrage du fouillis. Le lecteur pourra consulter sur ce point la référence /1/. Les simulations ont été effectuées sur trois types de fouillis de sol: champs labourés (C), arbres (A) et neige givrée (N). Ces trois types de signaux admettent une représentation par des modèles autorégressifs d'ordre 6 /1/.

Le nombre de bits du signal était fixé par l'expérience à 8. L'influence de NER et de NBG est comparable à celle observée pour les signaux synthétiques, et nous prenons pour la suite NER=4 et NBG=8. Là encore nous constatons une plus forte robustesse des algorithmes exacts par rapport aux gradients (planches 2 et 3).

Pour les premiers, nous observons pour NBK faible de fortes dégradations (8 à 20% pour l'énergie, plus de 40% pour la blancheur) et un comportement asymptotique pour NBK supérieur à 8.

En ce qui concerne les gradients, le seuil de 12 bits (et en tout état de cause 16 bits) est nécessaire pour obtenir des dégradations inférieures à 10%.

9.0 CONCLUSION:

La ressemblance des courbes obtenues pour les signaux

ETUDE DES EFFETS DE QUANTIFICATION SUR LES ALGORITHMES EN TREILLIS ADAPTATIFS.

synthétiques d'ordre 2 ou 6 et pour les signaux réels est remarquable. Les conclusions partielles obtenues pour chaque type de signal se recoupent ce qui donne un caractère plus général aux résultats acquis.

Dans l'introduction, nous nous proposons de comparer les performances de deux types d'algorithmes: moindres carrés exacts et gradients. Ces derniers sont moins précis et convergent en plus d'itération. L'étude effectuée montre qu'ils sont également moins robustes aux effets de troncature. Cependant, leur coût opératoire est moins lourd de sorte que le problème qui se pose schématiquement dès lors est le choix entre peu de calculs sur des mots longs et plus de calculs sur des mots courts. Rappelons que nous avons mesuré les performances après convergence de l'algorithme, ce qui masque le retard à la convergence des gradients.

Cette étude a donc permis de révéler la relative robustesse des algorithmes de filtrage en treillis. Les expériences effectuées nous amènent à retenir une configuration optimale pour l'implantation: pour les treillis exacts, $NER=4$, $NBG=8$, $NBK=8$ et pour les gradients, $NER=4$, $NBK=16$ pour un facteur d'oubli de $127/128$ et un signal codé sur 8 bits. Ces configurations d'implantation donnent des performances équivalentes entre les algorithmes exacts et les versions gradients, cependant le faible nombre de bits nécessaire aux premiers et leur rapidité de convergence nous laissent supposer qu'à l'avenir, pour des processeurs de plus en plus rapides, ils prendront le pas sur les versions approchées. Par contre, la charge en variables des algorithmes exacts impose un accès rapide aux données impliquant l'existence de mémoire interne RAM.

Examinons le coût opératoire des algorithmes et proposons nous d'évaluer leur possibilité d'implantation sur un micro-processeur. Pour cela, calculons en temps de cycle leurs coûts opératoires respectifs: définissons un temps t_a pour l'addition (par exemple 1 temps de cycle), un temps t_m pour la multiplication (par exemple 2 temps de cycle) et un temps $N.t_d$ pour une division nécessitant N bits de précision (par exemple $2N$ temps de cycle). Nous appliquerons à cette estimation un facteur multiplicatif pour tenir compte des transferts d'opérateurs. Généralement, ce coût multiplicatif n'est pas très élevé, compte tenu de l'organisation possible des calculs et des possibilités de mémoire interne dans les spécifications des futurs matériels. Nous prendrons ainsi un coût multiplicatif γ de 1,5.

L'estimation que nous faisons ici ne tient pas compte de la structure particulière éventuelle de la machine, en particulier de l'organisation parallèle de certains calculs, qui permettrait de réduire cette estimation. Nous obtiendrons ainsi un majorant du coût effectif de l'utilisation. L'estimation est faite pour une récurrence en temps et un ordre de modélisation p et tient compte du coût opératoire de l'initialisation.

Nous notons N_k , N_v et N_i les nombres de bits des coefficients de réflexion, des variables liées à la vraisemblance et de l'intercorrélolation des résidus. L'estimation pour chaque algorithme est la suivante:

a. treillis exact avec fenêtre antérieure:
 nous avons $6p+1$ additions, $7p+2$ multiplications, $2p$ divisions avec N_k bits de précision, p divisions avec une précision de N_v bits et p divisions sur N_i bits, c'est-à-dire un coût égal à:

$$\gamma \left[(6p+1)t_a + (7p+2)t_m + (2pN_k + pN_v + pN_i)t_d \right]$$

b. treillis exact sur fenêtre glissante:
 nous avons alors $10p+2$ additions, $10p+2$ multiplications et $2p$ divisions sur N_k , N_v et N_i bits de précision, soit un coût égal à:

$$\gamma \left[(10p+2)t_a + (10p+2)t_m + 2p(N_k + N_v + N_i)t_d \right]$$

c. gradient à un jeu de coefficients de réflexion:
 l'algorithme a $6p$ additions, $7p$ multiplications et p divisions avec N_k bits de précision:

$$\gamma \left[6pt_a + 7pt_m + pN_k t_d \right]$$

d. gradient à deux jeux de coefficients de réflexion:
 l'algorithme a alors $6p$ additions, $8p$ multiplications et $2p$ divisions:

$$\gamma \left[6pt_a + 8pt_m + 2pN_k t_d \right]$$

Nous avons vu qu'il convenait de prendre, pour les algorithmes exacts, $N_k=N_v=8$ et $N_i=20$, et, pour les gradients, $N_k=16$. Nous poursuivons l'estimation en considérant une application sur un micro-processeur semblable au TMS 320 et en fixant $t_a=1$, $t_m=2$ et $t_d=2$. Nous avons ainsi:

algorithme FA:	$\gamma(108p+5)$	FG:	$\gamma(174p+6)$
G1:	$\gamma(52p)$	G2:	$\gamma(86p)$

La technologie actuelle permet un temps de cycle de 200ns sur le TMS 320 commercialisé et les spécifications du matériel nouveau indiquent un temps de 100ns. Sur cette base, avec $\gamma=1.5$, le coût d'une récurrence en temps pour un treillis d'ordre 6 est majoré par:

FA:	98 μs	et	FG:	157,5 μs
G1:	46,8 μs	et	G2:	77,4 μs

Il n'est pas possible de dire si ces algorithmes sont utilisables pour les applications radar en toutes généralités, puisqu'il faut tenir compte essentiellement de la fréquence de récurrence du radar, 1 à 100 kHz, et du nombre d'échantillons analysés. Pour une application "parole" avec un ordre égal à 10, nous aurons:

FA:	163 μs	et	FG:	262 μs
G1:	78 μs	et	G2:	129 μs

La fréquence d'échantillonnage du signal de parole est classiquement de 8kHz, c'est-à-dire une période de 125 μs . Seul l'algorithme G1 semble utilisable pour une telle application mais rappelons que ces chiffres sont des majorants qu'une organisation précise des calculs pourrait diminuer.

Ces exemples nous montrent néanmoins que les algorithmes FA et G2 ont un coût comparable, avec une meilleure convergence pour le premier. L'algorithme G1 est environ deux fois plus rapide et l'algorithme FG deux fois plus long.

REFERENCES:

/1/ JF AGNEL:
 Elimination de Fouillis en Radar et Détection Séquentielle de Cibles. Une Approche par la Reconnaissance des Formes et le Filtrage en Treillis.
 Thèse de Docteur-Ingénieur ENST 1985.

/2/ B FRIEDLANDER:
 Lattice Filters for Adaptive Processing.
 Proc. IEEE. Vol 70. N:8. pp 829-867. 1982.



ETUDE DES EFFETS DE QUANTIFICATION SUR LES ALGORITHMES EN TREILLIS ADAPTATIFS.

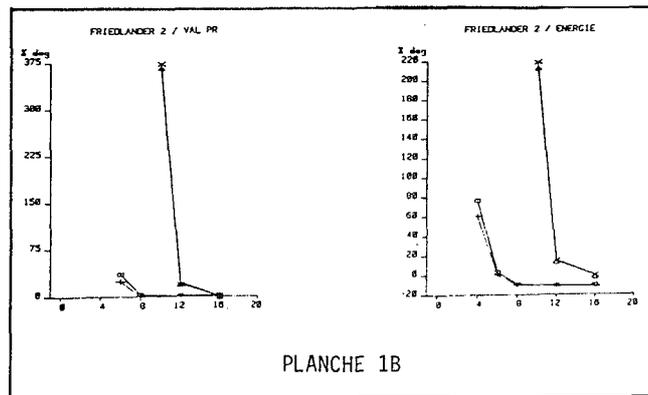
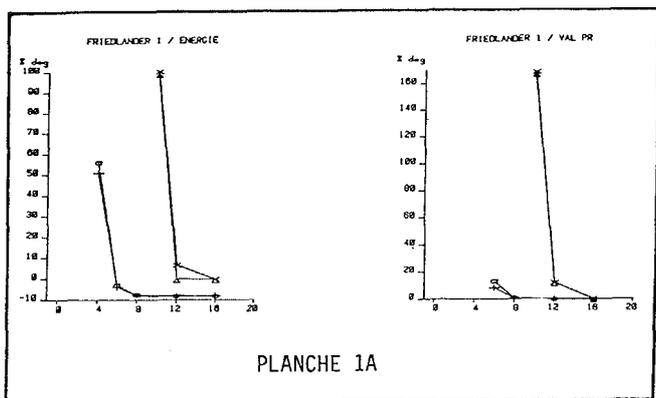
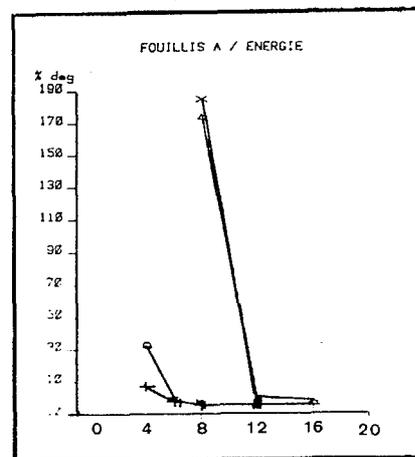


PLANCHE 2



LEGENDE PLANCHES 1 ET 2

- + Fenetre Anterieure
- Fenetre Glissante
- △ Gradient 1
- × Gradient 2

PLANCHE 3

