

# Synthèse des filtres RIF à phase linéaire par limitation de l'espace de recherche

## Synthesis of Filters FIR with Linear Phase by Limitation of the Space of Search

**Benaoumeur Boulerial\*, Mohamed Faouzi Belbachir\*\***

\*Institut des Télécommunications, route d'Es-Sénia, Oran.

e-mail : benboulerial@yahoo.com

\*\*Laboratoire « Signaux, Systèmes, et Données », Département d'Électronique, Faculté Génie Électrique, U.S.T.O, BP. 1505, Oran El M'Naouer.

e-mail : mf\_belbachir@yahoo.fr

Manuscrit reçu le 21 juin 2004

### Résumé

Dans cet article nous présentons une nouvelle méthode dans la conception des filtres numériques à coefficients discrets. Cette méthode limite géométriquement l'espace de recherche des valeurs des coefficients du filtre pour ensuite extraire la solution discrète optimale. Cette démarche permet d'éviter d'une part les erreurs d'arrondi (inévitables par d'autres méthodes), et d'autre part les erreurs de calcul-machine commises en utilisant les algorithmes de calcul des filtres tel que par exemple l'algorithme de Parks et MacClellan (PMC). Notre méthode a plusieurs avantages, elle permet de trouver la solution optimale globale directement implémentable. Plusieurs exemples de filtres discrets en virgule fixe sont donnés. Les résultats obtenus, qui sont plus performants que ceux obtenus par l'algorithme PMC même en virgule flottante, démontrent l'efficacité de notre méthode.

### 1. Introduction

L'algorithme de Parks et MacClellan (PMC) [1] est le plus utilisé dans la conception des filtres numériques RIF à phase linéaire. L'amplitude du filtre obtenu est optimale au sens de Tchebycheff alors que sa phase est exactement linéaire. Les valeurs des coefficients sont calculées à l'aide d'ordinateurs de mots machine de longueur  $L_{ord} \geq 32$  bits.

Lors de la mise en œuvre de ces filtres, les coefficients sont implantés dans des processeurs de signaux de longueur de mots  $L_{proc}$  généralement plus petite que  $L_{ord}$  ( $L_{proc} \leq L_{ord}$ ). Cette opération se répercute souvent par une dégradation importante des performances du filtre.

Afin de pallier à ce problème, de nombreux travaux ont été consacrés à la recherche des coefficients discrets directement implémentables sur le processeur [2][3][4][9]. En d'autres termes, la recherche des coefficients s'effectue dans l'ensemble discret  $E_{proc}^N$  des valeurs représentables dans le processeur ( $N$ : la longueur du filtre).

La méthode par arborescence utilisée dans [3] et appelée « ARBO » effectue une recherche exhaustive du filtre optimal dans  $E_{proc}^N$ . Elle assure l'optimalité globale de la solution au prix d'une très grande complexité. À titre d'exemple, la conception d'un filtre RIF à phase linéaire de longueur  $N$  pour un processeur de mot machine  $L_{proc}$ ,  $(2^{L_{proc}})N$  filtres sont auscultés par cet algorithme pour délivrer la solution optimale. On montre que la méthode devient inutilisable pour  $L_{proc} \cdot N > 64$ . Par conséquent pour augmenter la vitesse de convergence il est nécessaire de limiter l'espace de recherche.

Dans cet article une nouvelle démarche (LOCARBO) est proposée permettant de réduire l'espace de recherche  $E_{proc}^N$  en un espace réduit  $Er_{proc}^N$  avec  $Er_{proc}^N \subset E_{proc}^N$ . Pour cela une recherche géométrique utilisant des conditions obtenues à partir du gabarit du filtre est utilisée. Celle-ci permet une réduction drastique de l'espace de recherche et donc une augmentation importante de la vitesse de convergence. Le filtre optimal est alors obtenu en effectuant une recherche exhaustive dans l'espace  $Er_{proc}^N$ .

Nous donnerons dans la suite de cet article à la section 2 la position du problème de la synthèse des filtres numériques RIF à phase linéaire et coefficients de longueur finie. La méthode proposée sera exposée dans les trois sections 3, 4, et 5. Nous terminerons par la section 6 où nous trouverons des exemples.

### 2. Position du problème

D'une manière générale, la réponse en fréquence  $H(e^{j2\pi f})$  d'un filtre RIF de longueur  $N$  s'exprime comme suit :

$$H(e^{j2\pi f}) = \sum_{n=1}^N h(n)e^{-j(n-1)2\pi f} \quad (1)$$

Sans perte de généralités, nous considérerons que le cas du filtre RIF à phase linéaire, avec  $N$  pair, l'amplitude  $H(f)$  s'exprime alors par [5] : (le terme de la phase linéaire étant omis)

$$H(f) = \sum_{n=1}^{N/2} 2h(n)\cos\left(2\pi f\left(\frac{N+1}{2} - n\right)\right) \quad (2)$$

Dans le cas d'un filtre passe-bas, les spécifications en amplitude sont données par :

$$\forall f \in [0, f_p] \quad 1 - \delta_p \leq H(f) \leq 1 + \delta_p \quad (3)$$

$$\forall f \in [f_a, 0.5] \quad -\delta_a \leq H(f) \leq \delta_a \quad (4)$$

où  $f_p$  et  $f_a$  sont les fréquences respectivement à la fin de la bande passante et au début de la bande atténuée, et avec  $\delta_p$  et  $\delta_a$  les tolérances dans les bandes passante et atténuée.

Le problème que nous voulons résoudre consiste à rechercher le vecteur des coefficients  $\mathbf{h} = [h(1), h(2), \dots, h(N/2)] \in E_{\text{proc}}^{N/2}$  qui correspond au filtre optimal au sens de l'erreur de Tchebycheff.

De nombreux travaux ont été entrepris pour la résolution de ce problème, nous citerons entre autres [2][4][6]. À notre connaissance seule les méthodes utilisant la recherche exhaustive [7][8], comme dans ARBO [3], assurent l'optimalité globale des résultats au prix d'une grande complexité. À l'aide d'un exemple nous illustrerons, dans la section suivante, l'approche utilisée.

### 3. Étude d'un exemple

Considérons le cas simple d'un filtre passe-bas RIF à phase linéaire, symétrique avec  $N = 4$ ,  $f_p = 0.318$ ,  $f_a = 0.371$ ,  $\delta_a = 0.455$ ,  $\delta_p = 0.455$  et  $L_{\text{proc}} = 5$  en représentation virgule fixe. La méthode d'ARBO consiste à chercher le meilleur arrangement  $[h(1), h(2)] \in E^2$  espace des coefficients discrets, qui minimise l'erreur de Tchebycheff. Chaque coefficient  $h(n)$  peut prendre 32 valeurs, il s'en suit que ARBO doit ausculter  $32^2 = 1024$  cas (ou filtres) pour trouver l'optimum. Ces différents cas sont représentés par des points dans la figure 1.

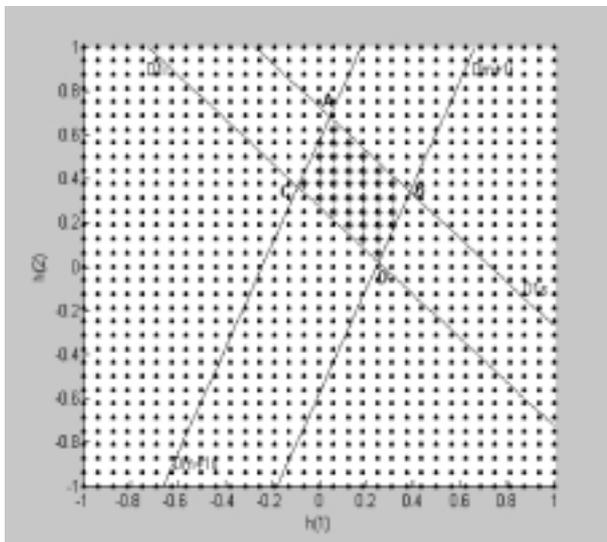
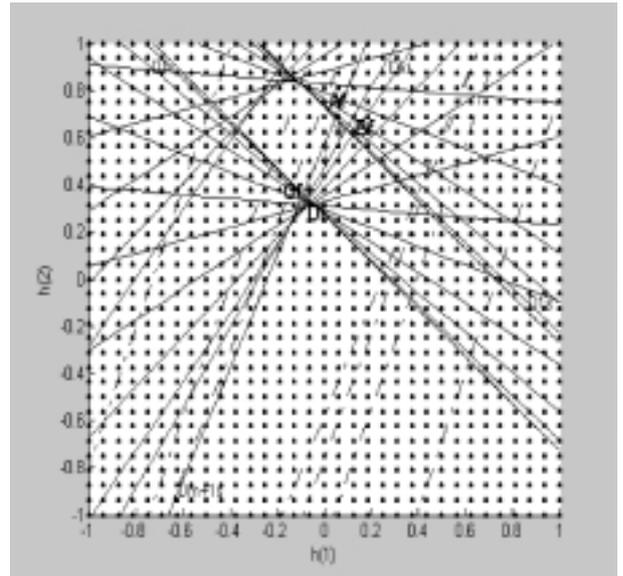
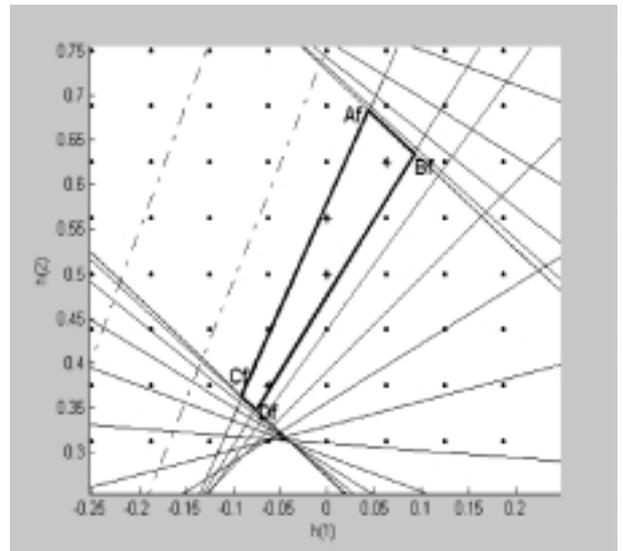


Figure 1. Réduction de l'espace de recherche  $E^2$  à l'aide des spécifications (3) et (4) pour  $f = 0$  et  $f = 0.371$ .  
Notons que nous passons de 1024 à 38 cas à ausculter.

La méthode que nous proposons LOCARBO consiste à réduire le nombre de cas à ausculter pour trouver l'optimum. Les expressions (3) et (4) par exemple pour  $f = 0$  et  $f = f_a = 0.371$  correspondent dans  $E^2$ , à un parallélogramme que nous notons par  $ABCD$ , (figure 1) qui contient les solutions admissibles indiquées par \* et au nombre de 38. Nous notons que nous passons de 1024 à simplement 38 cas. Ce nombre peut être réduit davantage en considérant d'autres fréquences. En effet les spécifications du gabarit doivent être vérifiées



(a)



(b)

Figure 2. Réduction de l'espace de recherche  $E^2$  à  $E^2r$  en considérant les spécifications d'amplitude pour  $f_i \in F_{BP} \cup F_{BA}$ .

dans tout l'intervalle  $[0, f_p] \cup [f_a, 0.5]$ . Si nous considérons celles-ci aux fréquences<sup>1</sup>  $f_i \in F_{BP} = [0.0318 \ 0.0636 \ 0.0954 \ 0.1272 \ 0.1590 \ 0.1908 \ 0.2226 \ 0.2544 \ 0.2862 \ 0.3180]$  dans la bande passante et  $f_i \in F_{BA} = [0.3710 \ 0.4033 \ 0.4355 \ 0.4678 \ 0.5]$  dans la bande atténuée, les solutions admissibles sont à l'intérieur du quadrilatère  $A_f B_f C_f D_f$  (voir figure 2.a). Nous constatons que le nombre de solutions s'est réduit à 4 (voir figure 2.b).

À partir de cet exemple, nous observons que l'on peut réduire d'une manière importante le nombre de cas à ausculter pour la recherche de l'optimum (de 1024 à 4) et cela simplement à partir des conditions tirées du gabarit.

<sup>1</sup> Les  $4N$  fréquences  $f_i$  sont choisies approximativement équidistantes dans  $[0, f_p] \cup [f_a, 0.5]$ ,  $f_p$  et  $f_a$  incluses.

Un certain nombre de remarques s'imposent :

- partant de l'espace initial, le quadrilatère  $ABCD$ , (contenant 38 solutions admissibles), et en utilisant les spécifications pour les fréquences  $f_i \in F_{BP} \cup F_{BA}$  on a abouti au quadrilatère  $A_f B_f C_f D_f$  (contenant 4 solutions admissibles)

- Nous observons que les fréquences choisies ne permettent pas toutes de réduire l'espace de recherche.

Notre propos dans la section suivante est de généraliser cette approche.

#### 4. Étude du cas général

##### 4.1. Espace initial

D'une manière générale pour la conception d'un filtre de longueur  $N$ , on peut montrer que l'espace initial correspond à un polyèdre convexe  $P$ . Celui-ci est limité par  $N$  hyperplans et possède  $Ns$  sommets, avec  $Ns = 2^{N/2}$ . Ces hyperplans (supérieur et inférieur selon (3) et (4)) sont définis par les équations suivantes (5) et (6) :

$$\sum_{n=1}^{N/2} 2h(n) \cos\left(2\pi f_i \left(\frac{N+1}{2} - n\right)\right) = 1 + \delta_p$$

$$\sum_{n=1}^{N/2} 2h(n) \cos\left(2\pi f_i \left(\frac{N+1}{2} - n\right)\right) = 1 - \delta_p \quad (5)$$

$$f_i \in [0, f_p] \quad i = 1 \dots k$$

$$\sum_{n=1}^{N/2} 2h(n) \cos\left(2\pi f_i \left(\frac{N+1}{2} - n\right)\right) = \delta_a$$

$$\sum_{n=1}^{N/2} 2h(n) \cos\left(2\pi f_i \left(\frac{N+1}{2} - n\right)\right) = -\delta_a \quad (6)$$

$$f_i \in [f_a, 0.5] \quad i = k+1 \dots N/2$$

et que nous groupons dans l'équation (7)

$$\sum_{n=1}^{N/2} 2h(n) \cos\left(2\pi f_i \left(\frac{N+1}{2} - n\right)\right) = \Delta_q^i \quad (7)$$

$$\text{où} \quad \Delta_q^i = \begin{cases} 1 + (-1)^q \delta_p ; i = 1 \dots k \\ (-1)^q \delta_a ; i = k+1 \dots N/2 \end{cases}$$

$$\text{et} \quad q = \begin{cases} 0. & \text{hyperplan supérieur} \\ 1. & \text{hyperplan inférieur} \end{cases}$$

( $k$  est choisi tel que les fréquences  $f_i \in [0, f_p] \cup [f_a, 0.5]$  soient approximativement équidistantes).

Chaque sommet du polyèdre  $P$  est à l'intersection de  $N/2$  hyperplans. Le calcul des coordonnées des sommets  $[h_{Ns}]$  est effectué en résolvant  $Ns$  systèmes d'équations formées chacun par des équations du type (7) et pour  $N/2$  fréquences différentes. La solution est donnée par

$$[h_{Ns}] = 0.5[A]^{-1} \cdot [B]$$

avec

$$[h_{Ns}] = \begin{bmatrix} h_1(1) & \dots & h_{N_s}(1) \\ \dots & \dots & \dots \\ h_1(N/2) & \dots & h_{N_s}(N/2) \end{bmatrix}$$

$$[A] = \begin{bmatrix} \cos\left(2\pi f_1 \left(\frac{N+1}{2} - 1\right)\right) & \dots & \cos\left(\frac{2\pi f_1}{2}\right) \\ \dots & \dots & \dots \\ \cos\left(2\pi f_{N/2} \left(\frac{N+1}{2} - 1\right)\right) & \dots & \cos\left(\frac{2\pi f_{N/2}}{2}\right) \end{bmatrix}$$

et

$$[B] = \begin{bmatrix} \Delta_0^1 & \Delta_0^1 & \Delta_0^1 & \Delta_0^1 & \dots & \Delta_1^1 & \Delta_1^1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \Delta_0^{N-1/2} & \Delta_0^{N-1/2} & \Delta_0^{N-1/2} & \Delta_1^{N-1/2} & \dots & \Delta_1^{N-1/2} & \Delta_1^{N-1/2} \\ \Delta_0^{N/2-1} & \Delta_0^{N/2-1} & \Delta_1^{N/2-1} & \Delta_1^{N/2-1} & \dots & \Delta_1^{N/2-1} & \Delta_1^{N/2-1} \\ \Delta_0^{N/2} & \Delta_1^{N/2} & \Delta_0^{N/2} & \Delta_1^{N/2} & \dots & \Delta_0^{N/2} & \Delta_1^{N/2} \end{bmatrix}$$

Discussion sur le choix de la tolérance  $\delta$ : le polyèdre  $P$  ainsi obtenu est dépendant de  $\delta$ . La complexité de l'algorithme en dépend fortement

du choix de celle-ci. Par expérience nous avons pris  $\delta = \frac{\delta_{PMC} + \delta_{PMC_a}}{2}$ .

avec  $\delta_{PMC}$ : erreur de PMC

et  $\delta_{PMC_a}$ : erreur de PMC dont les coefficients ont été arrondis.

##### 4.2. Recherche de l'espace minimal par réduction de l'espace initial

La théorie mathématique ne permet pas encore de déduire d'une manière analytique, directement les hyperplans intéressants qui limitent l'espace minimal admissible. Pour cette raison, nous utilisons dans la recherche de l'espace minimal une méthode itérative. Dans cette partie nous effectuons des réductions successives sur l'espace initial fermé obtenu en (4.1).

Il est important de remarquer que les sommets  $[h_{Ns}]$  ainsi obtenus sont déduits d'un gabarit discrétisé sur  $N/2$  points. Aussi, l'algorithme que nous avons conçu, partant du polyèdre initial  $P$ , réduit celui-ci en considérant itérativement les conditions obtenus à partir de  $4N$  autres fréquences équidistantes dans  $[0, f_p] \cup [f_a, 0.5]$ .

L'algorithme que nous avons élaboré procède comme suit :

- Définir le polyèdre convexe de départ  $P(h_1, h_2, h_3, \dots, h_{N_s})$
- Pour chaque hyperplan défini par les nouvelles fréquences  $f_i \in [0, f_p] \cup [f_a, 0.5]$  et  $i = 1 \dots 4N$ , procéder comme suit :

si l'hyperplan coupe le polyèdre initial ( $h_1, h_2, h_3, \dots, h_{N_s}$ ) alors le nouvel polyèdre devient ( $h_1, h_B, h_C, \dots, h_{N_s}$ ) (en supposant que  $h_B$  et  $h_C$  sont les sommets dus à l'intersection de  $P$  et de l'hyperplan. Les sommets  $h_2$  et  $h_3$  n'appartiennent plus au polyèdre  $P$ ), autrement considérer un nouvel hyperplan.

#### 5. Recherche de la solution optimale dans l'espace réduit

Après la première étape (4) qui a consisté à réduire l'espace admissible  $E_{proc}^N$ , une seconde opération est effectuée pour la recherche de la solution optimale discrète contenue dans l'espace réduit  $E_{proc}^{N/2}$ . Pour cela

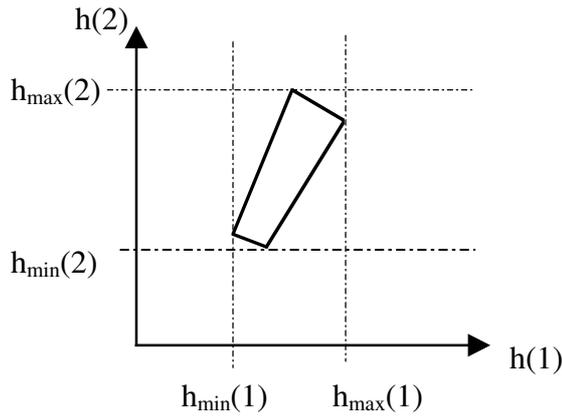


Figure 3. Projection de l'espace.

des projections de cet espace sont effectuées sur tous les axes pour déterminer les valeurs minimales et maximales des coefficients (fig.3). En fonction des intervalles  $[h_{\min}(n) \ h_{\max}(n)]$  correspondant à chaque coefficient et du pas de quantification désiré, la solution discrète optimale contenue dans l'espace est trouvée à l'aide de la recherche exhaustive.

### 6. Exemples

1) Un ensemble de filtres avec  $L_{\text{proc}}N \leq 64$  ont été conçus avec succès à l'aide de la méthode LOCARBO. Les filtres obtenus ont toujours été identiques à ceux délivrés par ARBO : ce qui conforte l'optimalité de LOCARBO. À titre d'exemple, nous présentons 4 filtres dont les caractéristiques figurent dans le tableau 1.

Tableau 1. Caractéristiques des filtres choisis.

	$N$	$L_{\text{proc}}$	$f_p$	$f_a$
$F1$	4	8	0.235	0.255
$F2$	4	8	0.235	0.355
$F3$	8	6	0.235	0.355
$F4$	8	8	0.235	0.355

Considérons le tableau 2 qui donne les temps nécessaires pour ARBO et LOCARBO pour la conception de ces filtres. Nous observons que LOCARBO présente une complexité réduite : le temps nécessaire pour LOCARBO peut être  $10^7$  fois plus petit que celui nécessité par ARBO.

Tableau 2. Les filtres obtenus par ARBO et LOCARBO sont identiques et optimaux, cependant le temps nécessaire pour leur calcul diffère selon la méthode utilisée.

	$F1$ (mn)	$F2$ (mn)	$F3$ (mn)	$F4$ (mn)
ARBO	0.040	0.04	9.401	4492
LOCARBO <sup>2</sup>	1.0e-03	1.1e-04	1.4e-04	1.8e-03

<sup>2</sup> (Les valeurs indiquées de LOCARBO dépendent de la tolérance  $\delta$  choisi)

Et à titre illustratif nous donnons ci-dessous l'espace réduit pour le filtre  $F4$ .

$$\begin{aligned}
 & -0.015625000 \leq h(1) \leq 0 \\
 & -0.148437500 \leq h(2) \leq -0.132812500 \\
 & 0.093750000 \leq h(3) \leq 0.117187500 \\
 & 0.484375000 \leq h(4) \leq 0.515625000
 \end{aligned}$$

C'est ce qui confirme une fois de plus la rapide convergence de l'algorithme LOCARBO.

2) Dans une deuxième étape, nous avons conçu avec succès des filtres avec  $L_{\text{proc}}N > 64$ , les temps nécessaires sont de quelques minutes. Rappelons que dans ce contexte ARBO n'est plus exploitable. Nous nous sommes intéressés à l'étude de l'évolution de l'erreur de Tchebycheff en fonction du nombre de bits  $L_{\text{proc}}$  ( $L_{\text{proc}} = 10 \dots 20$ ). Les caractéristiques du filtre passe-bas étudié sont :  $N = 12$ ,  $f_p = 0.318$  et  $f_a = 0.371$  avec une fonction de pondération identique dans les deux bandes. Les résultats sont donnés dans la figure 4 où l'erreur représentée par « + », de valeur égale à 0.15729 est celle obtenue par PMC (Matlab 6) sur un Pentium IV. Nous observons qu'à partir de  $L_{\text{proc}} = 13$  (les coefficients sont donnés ci-dessous), la méthode LOCARBO délivre des filtres meilleurs que ceux obtenus par PMC sur MATLAB avec un mot machine de 32 bits. Ceci est dû à l'efficacité de LOCARBO. Nous n'effectuons aucun calcul sur les coefficients du filtre, les valeurs discrètes sont directement recherchées à l'intérieur de l'espace réduit. Par contre pour PMC, les coefficients sont obtenus après calculs et sont donc entachés par le bruit machine.

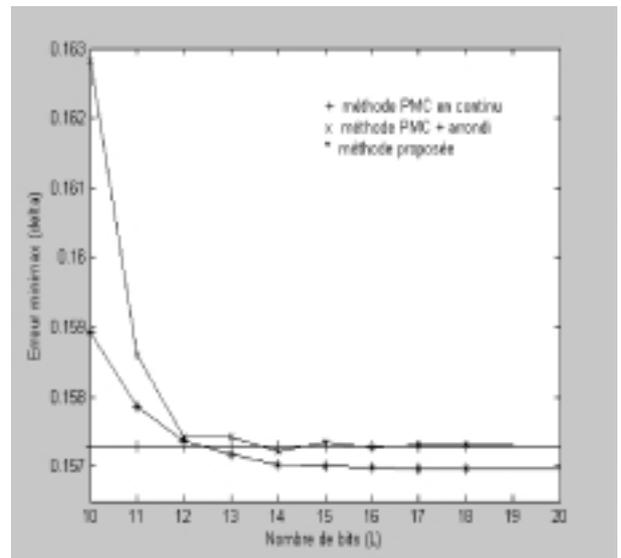


Figure 4. Evolution et comparaison de l'erreur minimax du filtre ( $N = 12$ ,  $f_p = 0.318$  et  $f_a = 0.371$ ) obtenue par LOCARBO (\*), PMC(+) et PMC + arrondi (x) pour  $L_{\text{proc}} = 10, 11, \dots, 20$ .

Les coefficients du filtre pour  $N = 12$  et  $L_{\text{proc}} = 13$  sont :

$$\begin{aligned}
 h(1) &= -0.0615234375000 = h(12) \\
 h(2) &= -0.0579833984375 = h(11) \\
 h(3) &= 0.1037597656250 = h(10) \\
 h(4) &= -0.1060791015625 = h(9) \\
 h(5) &= -0.0166015625000 = h(8) \\
 h(6) &= 0.5599365234375 = h(7)
 \end{aligned}$$

## 7. Conclusions

La méthode LOCARBO que nous avons proposé dans cet article permet de délivrer des filtres RIF à phase linéaire optimaux en amplitude au sens de Tchebycheff. Comparativement à la méthode par arborescence ARBO, l'algorithme LOCARBO présente une complexité drastiquement réduite et permet de concevoir des filtres avec  $L_{\text{proc}}N > 64$ . Un ensemble d'exemples sont donnés montrant l'efficacité de LOCARBO.

Nous montrons en outre que LOCARBO permet de concevoir des filtres pour  $L_{\text{proc}} \geq 13$  meilleurs que ceux obtenus par PMC à l'aide d'ordinateurs de longueur de mot machine égale à 32 bits.

## Références

- [1] J. H. Mc CLELLAN, T. W. PARKS, L. R. RABINER, « A Computer Program for Designing Optimum FIR Linear Phase Digital Filters », *IEEE Trans. Audio Electroacoust.*, vol. AU-21, p. 506-526, Dec. 1973.
- [2] Y. C. LIM, S. R. PARKER, « FIR Filter Design Over a Discrete Powers-of-Two Coefficient Space », *IEEE Trans. Acoust. Speech, Signal Processing.*, vol. ASSP-31, p. 583-591, June 1983.
- [3] B. BOULERIAL, M. F. BELBACHIR, « Filtrés RIF: Synthèse Directe dans l'Espace Discret des coefficients », *1<sup>st</sup> National Workshop on Signal & Image Processing, NWSIP'98*, 1 Décembre 1998, Sidi Bel-Abbes, Algérie.
- [4] I. PITAS, « Optimisation and Adaptation Of Discrete Valued Digital Filter Parameters By Simulated Annealing », *IEEE Trans. Signal Processing.*, vol. 42, N° 4, p. 860-866, April 1994.
- [5] L. R. RABINER, B. GOLD, « Theory and Application Of Digital Signal Processing », Prentice-Hall, Inc, 1975.
- [6] P. SIOHAN, A. BENSLIMANE, « Synthèse des Filtrés Numériques Non Récursifs à Phase Linéaire et Coefficients de Longueur Finie », *ANN. Telecommun.*, vol 39, N7-8, 1984.
- [7] M. SAKAROVITCH, « Optimisation Combinatoire. Graphes et Programmation Linéaire », Éditions Hermann 1984.
- [8] M. MINOUX, « Programmation mathématique, théorie et algorithmes », tomes 1 et 2, Dunod, 1983.
- [9] A. N. BELBACHIR, B. BOULERIAL, M. F. BELBACHIR, « A New Approach to Finite Wordlength Coefficient FIR Digital Filter Design Using the Branch and Bound Technique », *European Signal Processing Conference "EUSIPCO 2000"*, 04-08 sept.2000, Tampere, Finland.



Benaoumeur **Boulerial**

Benaoumeur Boulerial a reçu le diplôme d'ingénieur de l'EPFL de Lausanne en 1981, et le Magister en 1998 de l'institut des télécommunications d'Oran (Algérie). Depuis 1981, il est enseignant à l'institut des télécommunications d'Oran. Actuellement, il est au laboratoire *Signaux, Systèmes et Données* où il travaille sur la conception des filtres numériques.



Mohamed Faouzi **Belbachir**

Belbachir Mohamed Faouzi a obtenu l'ingénieur (1976), le Magister (1984) et la thèse d'État (1991) à l'USTO (Oran, Algérie). Actuellement il est professeur au département d'Électronique USTO et Directeur du laboratoire *Signaux, Systèmes et Données*. Ses travaux portent sur la conception des filtres numériques, le filtrage et la restauration de séquences d'images.