HUITIEME COLLOQUE SUR LE TRAITEMENT DU
SIGNAL ET SES APPLICATIONS

NICE du 1er au 5 JUIN 1981

EVALUATION DE LA PRECISION DANS LE TRAITEMENT DIGITAL DU SIGNAL

BOIS P. (1) et VIGNES J. (1) (2)

(1) Institut Français du Pétrole, 92506 RUEIL-MALMAISON
(2) Université Pierre et Marie Curie, 75230 PARIS CEDEX 06

RESUME

Durant cette dernière décennie, le traitement du signal a connu un développement considérable du fait de son informatisation et dans un futur proche son rôle deviendra encore plus important avec l'utilisation des microprocesseurs. Pour faire progresser les techniques de traitement digital du signal, il ne suffit pas d'élaborer des logiciels ou de les microprogrammer mais il faut également faire une étude de la précision des résultats numériques obtenus. En effet, le traitement digital du signal nécessite la mise en oeuvre d'algorithmes qui demandent l'exécution d'un grand nombre d'opérations arithmétiques portant sur des données expérimentales entachées d'erreurs. Ainsi, les résultats fournis par ces algorithmes sont toujours affectés d'une double erreur : erreurs dues aux données et erreurs dues à l'arithmétique à précision limitée de la machine (virgule flottante). Ces erreurs peuvent dans certains cas devenir très grandes et même rendre les résultats non significatifs. Une analyse de ces erreurs s'impose donc. Nous proposons ici une méthode générale et le logiciel correspondant pour évaluer automatiquement la précision locale dans les résultats du traitement digital du signal. Nous donnons des exemples d'application de cette méthode dans le cas de trois opérations de base du traitement du signal : la transformation de Fourier, la convolution et la corrélation ainsi que les filtres récursifs.

SUMMARY

In the last decade, signal processing has undergone considerable development as the result of computerization, and its role in the near future will become even more important with the use of microprocessors. For progress in digital signal processing techniques, it is not sufficient merely to work out software or to microprogram it, but investigations must also be made on the accuracy of the digital results obtained. Indeed, digital signal processing requires the implementation of algorithms involving the execution of a great many arithmetic operations using experimental data containing errors. Therefore, the results provided by such algorithms always contain a dual error, i.e. errors due to the data and errors due to the limited-precision arithmetic of the computer (floating point). In some cases these errors may become quite large and even make the results non significant. An analysis must thus be made of these errors. A general method is proposed here along with the corresponding software for automatically evaluating local accuracy in the results of digital signal processing. Application examples of this method are given in the case of three signal-processing operations, i.e. the Fourier transform, convolution and correlation, and recursive filtering.

I - INTRODUCTION

Le traitement du signal utilise une panoplie d'algorithmes numériques qui constituent les outils de calculs et parmi lesquels on peut citer la Tranformation de Fourier Rapide (T.F.R.), la convolution, la corrélation, les filtres récursifs (F.R.), etc. Avec le traitement de systèmes ayant des dimensions de plus en plus grandes, ces algorithmes demandent un nombre élevé de calculs très complexes dans $\mathbb{R}$ et $\mathbb{C}$. Les erreurs inhérentes à l'arithmétique de l'ordinateur (virgule flottante) engendrent, sur les résultats fournis par l'utilisation de ces outils de calculs, des erreurs qui peuvent, dans certains cas, devenir très importantes et qui nécessitent d'être analysées. Nous proposons ici une méthode et son logiciel associé écrit en langage Fortran, qui permet d'évaluer automatiquement la propagation des erreurs de calculs et des erreurs de données et donc de fournir la précision locale de chacun des résultats.

II - METHODE D'ANALYSE AUTOMATIQUE DES ERREURS DE CALCULS ET DE DONNEES DANS LES ALGORITHMES FINIS EXACTS

II.1 - Classification des outils du traitement du signal

Du point de vue mathématique, les outils de traitement du signal cités précédemment entrent dans la catégorie des *algorithmes finis exacts*, c'est-à-dire des algorithmes qui en un nombre fini de calculs fournissent des résultats exacts $r \subset \mathbb{R}$. Du point de vue informatique, lorsque ces traitements sont mis en oeuvre sur ordinateur, ils sont d'une part exécutés avec l'arithmétique à précision limitée de la machine qui, au niveau de chaque opération, engendre une erreur de calculs et d'autre part utilisent des données expérimentales entachées d'erreurs. Les résultats qu'ils fournissent sont ainsi entachés d'une double erreur de calculs et de données. La méthode de Permutation-Perturbation [1,2] dont nous allons brièvement exposer les idées de base, nous permet de faire exécuter automatiquement par l'ordinateur l'analyse de l'influence de ces erreurs et de fournir pour chaque résultat du traitement du signal le nombre de chiffres décimaux significatifs exacts.

II.2 - Méthode de Permutation-Perturbation

Considérons une procédure algébrique P que nous supposons pour simplifier ne fournir qu'un résultat unique $r \in \mathbb{R}$. Du fait que l'arithmétique des ordinateurs (virgule flottante) ne respecte pas les règles de l'arithmétique exacte, telle l'associativité de l'addition, nous obtiendrons un ensemble de résultats R_i tous différents mais aussi représentatifs du résultat exact r, selon l'ordre d'exécution des opérations contenues dans l'algorithme. Ainsi, à une procédure algébrique P correspond tout un ensemble de C_{op} procédures informatiques $\{\mathcal{P}_i\}_{i=1}^{i=C_{op}}$. Elle sont les images informatiques de la procédure algébrique et représentent toutes les permutations possibles des opérations permutables contenues dans la procédure P, c'est ce qu'on appelle la *méthode de Permutation*.

En particulier, considérons l'une de ces procédures informatiques $\mathcal{P}_j$ et supposons qu'elle nécessite l'exécution de k opérations arithmétiques (affectation, +, −, ×, :, fonct.). Chacune de ces opérations donne deux résultats informatiques, l'un par excès et l'autre par défaut, représentant aussi légitimement le résultat numérique exact. Ainsi pour les k opérations que comporte $\mathcal{P}_j$, il existe donc 2^k résultats informatiques représentant aussi légitimement le résultat de la procédure correspondante. C'est ce que l'on appelle la *méthode de Perturbation*.

En appliquant cette méthode à chacune des procédures informatiques images $\mathcal{P}_i$ fournies par la méthode de Permutation, on obtient un ensemble $\mathcal{R}$ de résultats:

$$\{\mathcal{R} / R_i \in \mathbb{F}\} ,$$

où chacun des R_i représente aussi légitimement le résultat algébrique unique $r \in \mathbb{R}$, $\mathbb{F}$ étant l'ensemble des nombres virgule flottante représentables en machine.

Le cardinal de l'ensemble $\mathcal{R}$ est défini par :

$$\text{Card } \mathcal{R} = C_{op}\, 2^k . \qquad (1)$$

C'est ce qu'on appelle la *méthode Permutation-Perturbation*. Il a été montré que :

- la moyenne $\overline{R}$ des R_i est la meilleure approximation du résultat r [3,4i],
- le nombre de chiffres décimaux significatifs exacts de $\overline{R}$ est donné par [2] :

$$C = \text{Log}_{10} \frac{|\overline{R}|}{\delta} , \qquad (2)$$

où δ est l'écart-type de la population des R_i.

II.3 - Le logiciel de Permutation-Perturbation

Il a été montré [3] que le nombre de chiffres décimaux significatifs exacts C du résultat peut être évalué à partir d'un sous-ensemble de $\mathcal{R}$ contenant seulement trois éléments $R_i \in \mathcal{R}$, i = 1, 2, 3. Ces trois éléments R_i sont obtenus en faisant exécuter trois fois par l'ordinateur la même procédure image $\mathcal{P}_i$. Chaque exécution de la procédure $\mathcal{P}_i$ consiste à perturber aléatoirement (par défaut ou par excès) le résultat de chaque opération arithmétique contenue dans la procédure $\mathcal{P}_i$. Dans la pratique, cette perturbation est faite à l'aide d'une fonction écrite en Fortran. Cette fonction perturbe aléatoirement une valeur réelle $X \in \mathbb{F}$ en additionnant un 0 ou un 1 au bit de poids le plus faible de la mantisse. Cette fonction dépend de l'ordinateur utilisé et de la codification des valeurs numériques en machine.

L'utilisation de cette fonction permet donc d'obtenir, pour chaque résultat d'opération arithmétique, la valeur par défaut (addition de 0) ou la valeur par excès (addition de 1) du résultat exact de l'opération arithmétique et cela si l'ordinateur travaille en arithmétique virgule flottante normalisée avec troncature. Si l'ordinateur utilise une arithmétique virgule flottante arrondie, la perturbation se fait aléatoirement et d'une manière pondérée en ajoutant − 1, 0 et + 1 au bit de poids le plus faible de la mantisse.

Après avoir fait exécuter trois fois la même procédure $\mathcal{P}_i$ ainsi perturbée, on obtient trois images $R_i \in \mathcal{R}$, i = 1,2,3 représentant aussi légitimement les uns que les autres le résultat r de la procédure algébrique. A partir de ces trois éléments, on calcule leur moyenne $\overline{R}$ qui, comme nous l'avons dit au § II.2, représente le mieux le résultat exact r. Avec l'équation (2), on en déduit le nombre de chiffres décimaux significatifs exacts de $\overline{R}$.

Le logiciel que l'on vient de décrire permet donc de faire l'analyse automatique de la propagation des erreurs de calculs dues à l'arithmétique de l'ordinateur. De plus, il est très facile avec ce logiciel de faire en même temps l'analyse automatique de l'influence des erreurs de données sur les résultats de l'algorithme fournis par la machine. Pour ce faire, il suffit dans le cas où les amplitudes du signal d_i, i = 1, ..., N_{exp} (N_{exp} étant le nombre de points d'expérience) sont entachés d'une erreur relative de mesure ε_1, de faire 3 exécutions avec chaque fois des données différentes définies par :

$$d_{ij} = d_i (1 + \theta_{ij} \varepsilon_i) \qquad \begin{matrix} i=1, \ldots, N_{exp} \\ j=1,2,3 \end{matrix} \qquad (3)$$

θ_{ij} étant un nombre aléatoire uniformément réparti entre - 1 et + 1.

Ainsi le logiciel permet-il de fournir le nombre de chiffres décimaux significatifs exacts C du résultat par l'équation (2), résultat qui tient compte d'une part des erreurs de calculs et d'autre part des erreurs de données.

Remarque

L'utilisation de ce logiciel nécessite trois exécutions de la procédure, ce qui tend à allonger d'autant le temps machine. Ceci semble onéreux, mais n'est pas plus coûteux que l'utilisation de la double précision qui n'est qu'un pis aller sans aucune garantie dans bien des cas.

III - APPLICATION DU LOGICIEL DE PERMUTATION-PERTURBATION AUX METHODES NUMERIQUES DU TRAITEMENT DU SIGNAL

Le logiciel présenté ci-dessus peut facilement s'adapter à toutes les méthodes numériques du traitement du signal. Toutefois, nous nous limitons à présenter ici son application à la T.F.R. discrète, à la convolution et corrélation et aux F.R.

III.1 - Evaluation de la précision locale de la T.F.R.

Du fait de la nature discrète du signal, on utilise les algorithmes de la T.F.R. : Cooley Tukey [5], Gentleman Sande [6], ... A cause de la structure de ces algorithmes pour lesquels l'ordre d'exécution est figé, seule la méthode de Perturbation est utilisée.

III.1.1 - Cas où le signal est connu sans erreur (ε_i=0, i=1,2,...,N_{exp})

Comme nous l'avons dit au § II.3, la perturbation de l'algorithme consiste à utiliser des fonctions écrites en langage Fortran qui s'appliquent dans le domaine réel. Pour perturber dans l'algorithme de la T.F.R. les opérations complexes, on a recours aux mêmes fonctions mais qui s'appliquent séparément sur la partie réelle et sur la partie imaginaire du nombre complexe à perturber.

III.1.2 - Cas où le signal est connu avec une précision relative ($\varepsilon_i \neq 0$, i=1,2,..., N_{exp})

Les erreurs de données doivent ici être associées à la propagation des erreurs de calculs et pour ce faire les trois exécutions de l'algorithme de la T.F.R. perturbé doivent être faites avec les données définies par l'équation (3).

III.1.3 - Résultats numériques [4]

a - Le signal est connu exactement (ε_i=0, i=1,2, ..., N_{exp})

Le signal est constitué par une suite de nombres aléatoires uniformément répartis entre - 1 et + 1. Nous avons considéré les transformées de Fourier de 1024 valeurs ainsi obtenues. Nous ne rapportons dans le tableau I qu'un échantillon des valeurs des parties réelles et imaginaires. Notons que l'on peut très aisément évaluer la précision de chaque module et phase de la transformée mais ceci n'est pas présenté ici.

Le tableau I donne les valeurs des parties réelles et imaginaires en double précision (1ère ligne) considérées comme exactes par rapport à la simple précision (2ème ligne) et le nombre de chiffres décimaux significatifs exacts C fourni par notre logiciel. Il ressort du tableau I que C est parfaitement évalué par notre méthode.

b - Le signal est donné avec une précision relative ε_i connue ($\varepsilon_i \neq 0$, i = 1,2, ..., N_{exp})

Le signal est constitué de 256 échantillons provenant d'une trace sismique tirée d'une étude du Bassin Parisien. Pour simplifier, nous supposons que tous les ε_i sont égaux à 10^{-4}. Le tableau 2 nous fournit pour un échantillon de ces 256 valeurs le nombre de chiffres significatifs exacts des parties réelles et imaginaires.

L'application du logiciel de Permutation-Perturbation s'applique très facilement à toutes les transformations linéaires discrètes telles celles de Walsh, de Paley, de Hadamard, de Haar, etc.

III.2 - Evaluation de la précision locale dans le calcul de la convolution (corrélation)

Le calcul de la convolution (corrélation) peut se faire par deux méthodes différentes : soit directement soit par l'intermédiaire de la transformation de Fourier.

III.2.1 - Méthode directe

Considérons le produit de convolution $f(\ell)$ de deux fonctions échantillonnées $f_1(i) \in \mathbb{R}$, i=1,...,N_1 et $f_2(j) \in \mathbb{R}$ j=1, ..., N_2, avec $N = N_1 + N_2 - 1$;

$$f(\ell) = \sum_{k=\text{Max}(1,\, \ell-N_2+1)}^{k=\text{Min}(N_1,\ell)} f_1(k)\, f_2(\ell-k+1), \quad \ell=1,\ldots,N. \qquad (4)$$

L'application de la méthode Permutation-Perturbation à la relation (4) permet d'obtenir le nombre de chiffres décimaux significatifs exacts C_ℓ de chacune des valeurs $f(\ell)$ par la relation :

$$C_\ell = \log_{10} \frac{|\overline{f(\ell)}|}{\delta_\ell} \qquad (5)$$

$\overline{f(\ell)}$ étant la moyenne des trois valeurs fournies par le logiciel décrit dans le § II.2 et δ_ℓ étant leur écart-type.

Un très grand nombre de calculs de convolution nous a permis de constater le parfait accord entre les nombres de chiffres décimaux significatifs donnés par (5) et le nombre de chiffres décimaux exacts (double précision).

III.2.2 - Méthode utilisant la transformation de Fourier

L'application du théorème de Plancherel nous permet de calculer la convolution par l'algorithme décrit ci-après :

$$f(\ell) = \mathcal{R}_e \{\mathcal{F}^{-1} (\mathcal{P})_\ell\}, \ \ell=1, \ldots, N, \qquad (6)$$

où $\mathcal{R}_e(Z)$ est la partie réelle de Z,

$\mathcal{F}(g)$ et $\mathcal{F}^{-1}(g)$ sont respectivement les transformées de Fourier directe et inverse de la fonction échantillonnée $g \subset \mathcal{C}$,

$(\mathcal{P})_k \subset \mathcal{C}$ est le produit terme à terme défini par

$$(\mathcal{P})_k = \mathcal{F}(f_1^o(k)).\ \mathcal{F}(f_2^o(k)), \quad k=1,\ldots,2Q_1 \ (Q_1=\text{Max}(N_1,N_2)) \qquad (7)(7)$$

$f_1^o(k) \subset \mathbb{R}$ et $f_2^o(k) \subset \mathbb{R}$ sont les fonctions échantillonnées $f_1(k)$ et $f_2(k)$ complétées par des zéros [7].

L'application de la méthode de Permutation-Perturbation nous permet aisément après trois exécutions de

TABLEAU I

I	Partie Réelle	C	Partie Imaginaire	C
2	+ 0,366137517001258 + 0,366137517001246	13	- 0,561254705938367 - 0,561254705938396	13
26	+ 1,26424525194931 + 1,26424525194939	14	- 1,19052559282246 - 1,19052559282251	13
50	+ 0,287946474953282 + 0,287946474953140	12	+ 0,161401215411010 + 0,161401215411210	12
64	+ 0,106386392532203 + 0,106386392532240	13	+ 0,00562989107693890 + 0,00562989107691075	11
86	+ 0,0778238509708866 + 0,0778238509708533	12	- 0,0170886969526247 - 0,0170886969526520	12
95	- 0,00123151342464212 - 0,00123151342462464	11	+ 0,000902203869012741 + 0,000902203868994977	9
112	- 0,0147206139918765 - 0,0147206139921346	11	+ 0,00311536803333468 + 0,00311536803316650	10
126	+ 0,00482681016395592 + 0,00482681016395312	12	- 0,0157898115801398 - 0,0157898115801226	12

TABLEAU 2

I	Partie réelle	C	Partie imaginaire	C
2	+ 0,366137	6	- 0,56125	5
26	+ 1,26424	6	- 1,190526	7
50	+ 0,28794	5	+ 0,16140	5
64	+ 0,10639	5	+ 0,00563	3
86	+ 0,07782	4	- 0,01709	4
95	- 0,0012	2	+ 0,00090	2
112	- 0,01472	4	+ 0,00312	3
126	+ 0,00483	3	- 0,01579	4

l'algorithme d'évaluer le nombre de chiffres décimaux significatifs exacts C de chacune des valeurs $f(\ell)$. Un grand nombre de calculs de convolution a été réalisé et a montré l'efficacité de cette méthode.

Remarquons que la *fonction de corrélation* $\rho(\ell)$ des deux séries discrètes $f_1(i)$ et $f_2(j)$ précédemment définies est donnée par :

$$\rho(\ell) = \sum_{k=\mathrm{Max}(1,\ell-N_2+1)}^{\mathrm{Min}(\ell,N_1)} f_1(k)\, f_2(k-\ell+N_2),\ \ell=1,\ldots,N\ . \qquad (8)$$

$\rho(\ell)$ a une forme analogue au produit de convolution (4), à ceci près que dans cette dernière, la fonction $f_2(j)$ est "renversée" sur l'ensemble des $j \in N$. Ainsi le calcul de la corrélation s'effectue d'une manière quasi identique dans le cas de la méthode directe. Dans le cas de celle qui utilise la T.F.R., il faut remarquer que la complémentation par des zéros se fait différemment [7] et que $(\mathcal{P})_k$ est donné par :

$$(\mathcal{P})_k = \mathcal{F}(f_1^o(k))\ .\ \mathcal{F}(f_2^o(k))^*\ , \qquad (9)$$

où l'astérisque signifie la quantité complexe conjuguée.

III.2.3 - Résultats numériques

a - Méthode directe

Rappelons que cette méthode est obtenue en calculant à l'ordinateur la formule (4).

α) Données exactes

Nous avons considéré le produit de convolution de $N_1 = 1024$ valeurs réparties uniformément entre - 1000 et + 1000 pouvant représenter les échantillons d'un signal avec $N_2 = 128$ valeurs réparties uniformément entre - 100 et + 100 pouvant être les coefficients d'un filtre. Ces valeurs sont supposées être connues exactement. Nous donnons ci-après un tableau qui n'est qu'un échantillonnage des 1151 valeurs de la convolution avec les valeurs exactes (double précision), les valeurs trouvées avec le logiciel Permutation-Perturbation et le nombre de chiffres décimaux significatifs exacts correspondants.

En examinant le tableau 3, nous constatons un excellent accord entre le nombre de chiffres décimaux significatifs exacts C déterminés par la méthode Permutation-Perturbation et le résultat exact donné par la double précision.

β) Données expérimentales

Dans le tableau 4, nous analysons le même produit de convolution que dans le cas des données connues exactement. Les 1024 valeurs uniformément réparties entre - 1000 et + 1000 qui jouent le rôle de signal sont connues à 10^{-3} près et les 128 valeurs uniformément réparties entre - 100 et + 100 et qui sont les coefficients du filtre sont connues à 10^{-6} près.

On constate en examinant le tableau 4 que certaines valeurs (I=699 et 892) de la convolution peuvent être entachées d'erreurs si grandes qu'elles deviennent non significatives.

b - Méthode utilisant la T.F.R.

On pourra comparer les résultats contenus dans les deux tableaux suivants avec ceux présentés dans les deux précédents pour confronter les précisions des deux méthodes.

α) Données exactes

Nous considérons dans le tableau 5 le même produit de convolution que celui qui nous a servi à analyser la précision dans la méthode directe (III.2.3.a) mais nous utilisons ici la méthode mettant en oeuvre le théorème de Plancherel.

En examinant les résultats des tableaux 3 et 5, on constate l'excellente concordance entre les différentes valeurs de C. Dans le § suivant, on comparera les précisions des deux méthodes.

TABLEAU 3

I	Valeur exacte (double précision)	Valeur donnée par la méthode Permutation-Perturbation	C
87	- 132657, 66646473233797	- 132657, 666464731	14
201	- 65293, 288158033897351	- 65293, 2881580375	13
272	- 16126, 120786176954648	- 16126, 1207861821	12
402	- 272375, 26638602794939	- 272375, 266386027	14
512	- 503894, 08230089058096	- 503894, 082300883	14
579	- 14638, 99374016983072	- 14638, 9937401695	14
699	+ 69, 445338374384767925	+ 69, 4453383713067	10
798	- 131334, 54775942861595	- 131334, 547759430	13
892	- 150, 13829499216654996	- 150, 138294987536	10
1051	+ 8755, 2046071648551790	+ 8755, 20460716594	12
1149	+ 41182, 471567278951829	+ 41182, 4715672790	14

TABLEAU 4

I	Valeur donnée par la méthode Permutation-Perturbation	C
87	- 132851	3
201	- 65563	2
272	- 16230	2
402	- 272566	3
512	- 503654	3
579	- 14656	3
699	+ 340	0
798	- 131448	3
892	- 597	0
1051	+ 8646	1
1149	+ 41198	3

TABLEAU 5

I	Valeur donnée par la méthode Permutation-Perturbation	C
87	- 132657, 666464730	14
201	- 65293, 2881580337	14
272	- 16126, 1207861779	13
402	- 272375, 266386025	14
512	- 503894, 082300883	14
579	- 14638, 9937401694	14
699	+ 69, 4453383700838	10
798	- 131334, 547759428	14
892	- 150, 138294991179	11
1051	+ 8755, 20460716635	12
1149	+ 41182, 4715672794	13

TABLEAU 6

I	Valeur donnée par la méthode Permutation-Perturbation	C
87	- 132740	3
201	- 65556	2
272	- 16346	2
402	- 272644	3
512	- 503779	3
579	- 14449	2
699	+ 56	0
798	- 131674	3
892	- 494	0
1051	+ 8340	1
1149	+ 41244	2

β) Données expérimentales

Comme précédemment, on étudie le même produit de convolution où les échantillons de $f_1(\ell)$ sont connus à 10^{-3} et ceux de $f_2(\ell)$ à 10^{-6} près.

Comme pour le tableau 4, on constate dans le tableau 6 ue les valeurs I=699 et I=892 ne sont pas significatives et qu'elles correspondent précisément à celles qui ont les valeurs de C les plus faibles dans le cas des données exactes. Ceci est dû au fait que ces valeurs sont faibles par rapport aux autres.

III.2.4 - Comparaison entre la méthode directe et la méthode par la T.F.R.

Il a été montré dans [8] que pour chaque élément $f(\ell)$ le nombre d'opérations arithmétiques réelles varie selon ℓ et que dans le cas particulier du traitement du signal, c'est-à-dire pour $Q_2 \leq \ell \leq N-Q_2$ (Q_2=Min(N_1, N_2), ce nombre est égal

- dans le cas de la méthode directe à :

$$N^1_{op} = 2\ \mathrm{Min}(N_1, N_2)\ , \qquad (10)$$

- dans le cas de la méthode par T.F.R. à :

$$N^2_{op} = 24\ n + 6, \qquad (11)$$

où $n = \mathrm{Max}(n_1,n_2) + 1$, ($N_1=2^{n_1}$ et $N_2=2^{n_2}$). (12)

Si la précision ne tenait compte que du nombre d'opérations, il serait aisé de choisir la méthode nécessitant le moins d'opérations et définie par :

$$N_{op} = \mathrm{Min}(N^1_{op}, N^2_{op})\ . \qquad (13)$$

En réalité, la précision sur chaque valeur $f(\ell)$ résulte du nombre d'opérations mais aussi du conditionnement numérique de ces opérations qui, lui, dépend des valeurs mises en jeu dans les calculs. Aussi, la seule méthode valable pour évaluer la précision exacte des valeurs de $f(\ell)$ est l'utilisation du logiciel décrit ci-dessus car il tient compte à la fois du nombre d'opérations et de leur conditionnement numérique.

Lors de l'examen des résultats contenus dans les tableaux 3 et 5, nous avions constaté une excellente similitude entre les valeurs de C dans la méthode directe et la méthode utilisant les T.F.R. Indépendamment du conditionnement numérique des opérations, cela semble se justifier puisque : N^1_{op} = 256 et N^2_{op} = 270 donc sensiblement égaux.

Dans ces conditions, on peut utiliser l'une ou l'autre méthode, seule le logiciel Permutation-Perturbation qui tient compte du conditionnement numérique des opérations peut nous aider à faire notre choix.

III.3 - Evaluation de la précision locale dans les F.R. [9]

III.3.1 - Formulation directe des F.R.

Considérons la fonction échantillonnée $x_i \in \mathbb{R}$ $i = 1, 2, \ldots, N$ entrant dans le F.R. de coefficients $\{a_k\}_{k=1}^{p} \in \mathbb{R}$ et $\{b_\ell\}_{\ell=0}^{q} \in \mathbb{R}$ pour donner un signal de sortie $y_n \in \mathbb{R}$ défini par :

$$y_n = -\sum_{k=1}^{p} a_k\, y_{n-k} + G \sum_{\ell=0}^{q} \ell\, x_{n-\ell} \qquad (14)$$

où G est le gain du filtre.

L'examen de la formule (14) montre que les F.R. se ramènent à la différence de deux produits de convolution de fonctions échantillonnées. De ce fait, le logiciel relatif à l'estimation de la précision dans la convolution est directement applicable aux F.R.

III.3.2 - Résultats numériques

a) Données exactes

Pour simplifier, les coefficients du filtre $\{a_i\}_{i=1}^{p}$ et $\{b_\ell\}_{\ell=0}^{q}$ sont au nombre de 256 chacun et sont des nombres aléatoires uniformément répartis entre - 0,001 et + 0,001, ainsi que les 512 valeurs du signal d'entrée x. On ne donnera dans le tableau 7 qu'un échantillonnage des 1024 valeurs demandées du signal de sortie y. On a supposé le gain du filtre G = 1.

TABLEAU 7

I	Valeurs exactes (double précision) / Valeurs trouvées par méthode de Permutation-Perturbation	C
177	- 0,159159000280167 10^{-8}	11
	- 0,159159000279216 10^{-8}	
286	+ 0,165058828537699 10^{-6}	13
	+ 0,165058828537725 10^{-6}	
580	- 0,201199908983461 10^{-7}	12
	- 0,201199908983035 10^{-7}	
697	- 0,107548635908671 10^{-5}	14
	- 0,107548635908677 10^{-5}	
898	+ 0,418148284219128 10^{-10}	11
	+ 0,418148284215658 10^{-10}	

On remarque que le nombre de chiffres décimaux significatifs exacts C est d'autant plus petit que les valeurs auxquelles ils correspondent sont elles-mêmes petites.Cette remarque est générale et s'applique également aux tableaux 1, 3 et 5.

On constate également une excellente concordance entre les résultats donnés par la double précision et ceux fournis par la méthode de permutation-perturbation et les valeurs de C.

b) Données expérimentales

Nous reprenons l'exemple précédent mais en supposant que les coefficients du filtre a et b sont connus à 10^{-5} et le signal x à 10^{-3} près. Dans le tableau 8, nous donnons les valeurs y(i) du signal de sortie correspondantes à celles du tableau 7.

TABLEAU 8

I	Valeurs données par la méthode Permutation-Perturbation	C
177	- 0,1 10^{-8}	1
286	+ 0,16 10^{-6}	2
580	- 0,2 10^{-7}	1
697	- 0,107 10^{-5}	3
898	non significatif	0

Comme on peut le constater dans les autres tableaux, l'influence des erreurs de données se manifeste très clairement dans ce dernier tableau au point que la valeur de y (898) trouvée est non significative.

IV - CONCLUSION

Le logiciel que nous avons décrit ici, basé sur la méthode de Permutation-Perturbation, est très facilement adaptable à toutes les méthodes de traitement du signal. Il permet de faire l'analyse automatique d'une part de la propagation des erreurs de calculs dues à l'arithmétique à précision limitée de la machine (virgule flottante normalisée) et d'autre part d'évaluer l'influence des erreurs de données sur les résultats de traitement du signal. Ce logiciel a été utilisé dans un grand nombre de cas, il a toujours donné d'excellents résultats. C'est à l'heure actuelle la seule méthode pratique permettant d'estimer la précision exacte sur tout résultat de traitement du signal.

BIBLIOGRAPHIE

[1] J. Vignes and M. La Porte, Error analysis in computing, Proceedings of IFIP Congress, Stockholm, pp. 610-614, 1974.

[2] J. Vignes, New methods for evaluating the validity of the results of mathematical computations, IMACS vol. 20, no. 4, pp. 227-249, 1978.

[3] M. Maillé, Méthodes d'évaluation de la précision d'une mesure ou d'un calcul numérique, LITP Report Institut de Programmation, Université Pierre et Marie Curie, 1979.

[4] P. Bois and J. Vignes, A software for evaluating local accuracy in the Fourier transform, IMACS, vol. 22, no. 2, pp. 141-150, 1980.

[5] J.W. Cooley and J.W. Tukey, An algorithm for the machine calculation of complex series, Math. Comput., vol. 19, pp. 297-301, 1965.

[6] W.M. Gentleman and G. Sande, Fast Fourier transforms for fun and profit, Proc. Fall Joint Computer, Conf., pp. 563-578, 1966.

[7] E.O. Brigham, The fast Fourier transform, Prentice-Hall, Inc., pp. 201-206, 1974.

[8] P. Bois et J. Vignes, Evaluation de la précision dans le calcul de la convolution et de la corrélation, C.R. Acad. Sci. II, 48, 1981.

[9] J.L. Shanks and T.W. Cairns, Use of digital convolution device to perform recursive filtering and the Cooley-Tukey algorithm, IEEE Trans. on Computers, vol. C-17, no. 10, pp. 943-949, 1968.