SEPTIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 28 MAI au 2 JUIN 1979

APPLICATION DES TRANSFORMEES NUMERIQUES AU TRAITEMENT DU SIGNAL

André BERTHON

Société d'Etudes et Conseils AERO - 3 avenue de l'Opéra - 75001 PARIS

RESUME

Les transformées numériques généralisent la propriété de la transformée de FOURIER (convolution ⟷ multiplication) à des séquences dont les éléments appartiennent à des anneaux finis (entiers modulo M). L'élément de matrice $e^{2i\pi \frac{mn}{N}}$ de la TFD est remplacé par α^{mn} avec $\alpha^N = 1$; (M,N,α) ne pouvant être choisis arbitrairement. On passe en revue les avantages de ces transformées (calcul rapide de convolutions par des algorithmes du type FFT, absence d'erreurs d'arrondi, remplacement des multiplications par des décalages lorsque α est une puissance de 2, etc.) et leurs limitations (le choix de la transformée dépend du problème considéré, on doit prendre des précautions pour éviter les ambiguïtés).

Associées aux algorithmes spécialisés qui s'appliquent aux séquences courtes, ces transformées permettent d'espérer des gains de temps considérables dans les calculs de corrélations. On examine leur application au traitement des signaux sonar, en particulier pour les calculs de spectres et d'interspectres.

SUMMARY

Number theoretic transforms generalize the property of the discrete Fourier Transform of mapping convolution into pointwise multiplication for arrays with elements belonging to finite rings (residue classes modulo M). The matrix element $e^{2i\pi \frac{mn}{N}}$ of the DFT is replaced by α^{mn} with $\alpha^N = 1$; (M,N,α) cannot be chosen arbitrarily. The advantages of these transforms (fast computation of convolutions via FFT-type algorithms, no roundoff errors, replacement of multiplications by shifts when α is a power of two, etc.) are reviewed together with their limitations (choice of transform depending on sequence length, care required to avoid ambiguities).

In association with specialized algorithms suited for short arrays these transforms allow to hope for considerable time reduction in computing correlation. Applications to sonar signal processing are discussed, in particular to the computation of spectra and cross-spectra.

APPLICATION DES TRANSFORMEES NUMERIQUES AU TRAITEMENT DU SIGNAL

1. INTRODUCTION

La théorie de la complexité de calcul, qui vise à déterminer le nombre minimum d'opérations nécessaire pour effectuer un calcul algébrique donné (en fait il s'agit presque toujours du nombre de multiplications) et à fournir les algorithmes optimaux, a fait ces dernières années des progrès très rapides [1]. Les types de calcul concernés sont principalement la convolution et la Transformation de FOURIER Discrète ou TFD qui intéressent le traitement du signal. On peut ainsi réduire le nombre de multiplications nécessaire pour certaines TFD d'un facteur 10 par rapport à l'algorithme de Transformée de FOURIER Rapide (FFT) sans augmenter sensiblement le nombre des additions [2]. Le premier pas au-delà de la FFT a été l'introduction des transformées de MERSENNE et de FERMAT par RADER [3], qui a montré qu'on pouvait, dans certains cas, ne faire aucune multiplication sinon par des puissances de 2, ce qui se ramène à des décalages.

A l'heure actuelle, celui qui veut effectuer la convolution cyclique ou la transformation de FOURIER de séquences de N nombres en optimisant le volume des calculs dispose d'une grande variété de moyens, en voici une liste, certainement non définitive et dans un ordre arbitraire :

1. Remplacer une TFD par une convolution et $2N$ multiplications par des nombres fixes suivant la méthode de BLUESTEIN [4].

2. Remplacer une convolution par deux TFD, une TFD inverse et N multiplications afin d'utiliser un algorithme du type FFT.

3. Ecrire la TFD comme produit de convolution à $(N-1)$ termes si N est premier [5].

4. Remplacer une convolution de longueur N, si $N = N_1 N_2$ avec N_1 et N_2 premiers entre eux, par une convolution à deux dimensions [6].

5. Calculer une convolution en effectuant les calculs dans un anneau fini $\mathbb{Z}_M$, c'est-à-dire modulo l'entier M ; c'est l'utilisation des "Number Theoretic Transforms" ou transformées aux racines primitives, que nous appellerons simplement transformées arithmétiques.

6. Calculer une convolution ou une TFD à l'aide des transformées polynomiales introduites par NUSSBAUMER [7].

7. Calculer les TFD et les convolutions courtes à l'aide des algorithmes de WINOGRAD [2], qui ramènent à des algorithmes de convolution pour lesquels le nombre de multiplications est égal au minimum théorique $2N-m$, m étant le nombre de diviseurs de N (on ne compte pas les multiplications par des constantes).

Ces méthodes reposent principalement sur les propriétés des anneaux de classes résiduelles d'entiers ou de polynômes qui sont succinctement rappelées ci-dessous en 2 et 3.

Leur principale limitation pratique vient de ce que la possibilité de les utiliser et de les combiner entre elles dépend de façon capricieuse de l'entier N, étant fonction de sa décomposition en facteurs premiers. Dans la suite nous envisagerons seulement les applications des transformées arithmétiques qui offrent, malgré les limitations qui apparaitront, une relative souplesse d'emploi ; leur attrait réside surtout dans le fait qu'on remplace la plupart des multiplications générales par des multiplications par les puissances d'un même entier, qui peuvent avoir une structure beaucoup plus simple ; mais pour en tirer pleinement parti il faut évidemment disposer d'un matériel spécialisé.

2. GENERALISATIONS DE LA TFD

Si x et y sont des suites de N nombres pris dans un anneau A, le produit de convolution cyclique de x par y est la suite z définie par :

$$(1)\quad (x * y)(n) = z(n) = \sum_{m=0}^{N-1} x(m)\, y(n-m)$$

Dans cette expression les indices sont calculés modulo N, autrement dit n et m sont considérés comme éléments de l'anneau des classes d'entiers modulo N, noté $\mathbb{Z}_N$.

Les transformations que l'on va considérer associent à une suite x la transformée $\hat{x}$ définie par :

$$(2)\quad \hat{x}(n) = \sum_{m=0}^{N-1} \alpha^{mn} x(m)$$

où de nouveau le produit nm est calculé modulo N ; la TFD correspond au cas où $A = \mathbb{C}$, $\alpha = \exp(2i\pi/N)$. On sait qu'elle possède la propriété dite de convolution cyclique :

$$(3)\quad z = x * y \iff \hat{z}(n) = \hat{x}(n)\hat{y}(n) \qquad n = 0,1,\ldots,N-1$$

Remarquons que, plus généralement, si A est un anneau et G un groupe additif fini, le produit de convolution cyclique entre deux fonctions x et y de G dans A n'est autre que la multiplication de l'algèbre du groupe $A[G]$, qui est le A-module libre sur G muni du produit :

$$(4)\quad \Big(\sum_{g\in G} x(g)\,g\Big)\times\Big(\sum_{h\in G} y(h)\,h\Big) = \sum_{u\in G}\Big(\sum_{g\in G} x(g)\,y(u-g)\Big)\,u$$

Le produit de convolution habituel, où $G = \mathbb{Z}_N$, est aussi isomorphe à la multiplication dans l'anneau $A[X]/(X^N-1)$ des polynômes à une indéterminée à coefficients dans A modulo le polynôme X^N-1. En effet, si à la suite $x(n)$ correspond le polynôme :

$$(5)\quad P_x = \sum_{n=0}^{N-1} x(n) X^n$$

on a bien :

$$(6)\quad z = x * y \iff P_x(X)\,P_y(X) = P_z(X) \mod (X^N-1)$$

Cette remarque sert de base à de nombreux algorithmes de calculs de convolution, pratiques pour N assez petit. Peut-on en obtenir d'autres qui généralisent le calcul de convolution par FFT ?

On voit immédiatement que la transformation (2) possède la propriété (3) parce que la "racine" α vérifie $\alpha^N = 1$. Si l'anneau A est le corps des complexes, on démontre qu'il n'y a pas d'autre transformation inversible vérifiant (3) que la TFD. Mais on peut se placer dans un anneau A différent de $\mathbb{C}$, à condition évidemment que les calculs effectués dans A permettent d'obtenir le résultat désiré dans $\mathbb{C}$. Ce qui sera souvent possible si A est un anneau fini possédant un élément unité. On a alors les résultats suivants [8,9]:

. Un anneau fini est la somme directe d'anneaux locaux A_i (parmi les anneaux Z_M, les anneaux locaux sont ceux pour lesquels M n'a qu'un facteur premier). La transformation (2) est un isomorphisme de $A[G]$ dans A^N si et seulement si elle se projette en un isomorphisme de $A_i[G]$ dans A_i^N pour chaque composante.

. Dans un anneau local un tel isomorphisme existe si et seulement si l'exposant m de G (le plus petit entier tel que $mg = 0$ pour tout élément $g \in G$) est inversible dans l'anneau, et s'il existe une racine primitive α d'ordre m de l'unité. Alors la transformation (2) est inversible.

Le cas où le groupe G n'est pas cyclique et se décompose donc en somme directe de n groupes cycliques, correspond à une convolution multidimensionnelle ; le cas $n = 2$ se rencontre notamment dans le traitement d'images. Nous nous limiterons à la dimension 1, G est alors le groupe Z_N et son exposant est $m = N$. Remarquons que si l'entier N est composite et s'écrit $N_1 N_2$, N_1 et N_2 étant premiers entre eux, Z_N est isomorphe à la somme directe de Z_{N_1} et Z_{N_2}. Cette propriété permet de remplacer une convolution longue par une cascade de convolutions courtes, et ne dépend pas de l'anneau A.

Reste le choix de A. Les premiers cas étudiés ont été ceux des corps finis, anneaux d'entiers modulo un nombre premier [3] ou corps de GALOIS généraux [10]. La généralisation à des anneaux non d'intégrité est apparue nécessaire [11] pour augmenter les possibilités de choix, notamment en ce qui concerne la longueur de séquence N. Une méthode générale consiste à considérer une extension algébrique finie du corps des rationnels et à prendre pour A une classe de résidus de l'anneau des entiers de ce corps [9]. En particulier les extensions quadratiques sont intéressantes pour traiter des convolutions de séquences complexes [12]. Cependant nous nous limiterons au cas où A est un anneau d'entiers modulo M.

3. TRANSFORMATIONS ARITHMETIQUES DANS Z_M

Appelons transformation arithmétique (TA) la transformation (2) dans l'anneau Z_M ; elle est définie par la donnée de M, N, et α ; la longueur de séquence N doit vérifier les conditions énoncées en 2, c'est-à-dire, dans le cas présent, être un diviseur de $(p_i - 1)$ pour tous les facteurs premiers p_i du module M ; α est une racine d'ordre N de l'unité et la TA admet une transformation inverse :

$$(7) \quad x(n) = N^{-1} \sum_{m=0}^{N-1} \alpha^{-mn} \hat{x}(m)$$

N^{-1} étant l'inverse de N modulo M. Le produit de convolution cyclique (1) peut donc se calculer à l'aide de deux TA et d'une TA inverse en vertu de (3). Pour qu'on ait avantage à procéder ainsi il faut évidemment que le nombre d'opérations nécessaire pour effectuer une TA soit très inférieur à N^2 ; c'est le cas, exactement dans les mêmes conditions que pour la TFD ordinaire, c'est-à-dire si l'entier N est composite. D'autre part, l'utilisation d'une TA plutôt que d'une FFT pour calculer la convolution n'est intéressante que si les opérations à effectuer sont moins coûteuses. Pour gagner du temps par rapport au calcul par FFT - qu'on suppose programmé en nombres entiers - il faut que les multiplications par les puissances de α soient plus simples que par les puissances de $\exp(2i\pi/N)$. On cherche donc pour α des entiers s'écrivant à l'aide d'un petit nombre de digits binaires ainsi que leurs puissances. Alors une multiplication se ramène à un petit nombre de décalages et d'additions. D'autre part, la TA exige de calculer modulo M, on peut certes attendre la fin des calculs - sauf dépassement de capacité - pour effectuer les N divisions euclidiennes par M dont les restes sont les éléments $x(n)$ cherchés. Mais il y a avantage à ce que M ait lui-même une structure de bits simple telle que l'arithmétique modulo M soit facile à programmer ou mieux à réaliser dans un calculateur spécialisé. D'où les deux choix de M les plus populaires, celui des nombres de MERSENNE et celui des nombres de FERMAT :

$$(8) \quad M_q = 2^q - 1 \quad q \text{ premier} \qquad F_n = 2^b + 1 \qquad b = 2^n$$

L'arithmétique modulo M_q est simplement l'arithmétique à q bits binaires dans laquelle le bit de dépassement est reporté en première position. On montre que pour $N = 2q$ la racine $\alpha = -2$ est utilisable mais, q étant premier, on ne peut employer d'algorithmes rapides. On a donc éliminé les multiplications générales mais non réduit le nombre des additions. En revanche, les nombres de FERMAT (qui ne sont pas premiers pour $n > 4$) permettent de choisir $N = 2b = 2^{n+1}$ avec $\alpha = 2$. La transformée ne nécessite plus que $n+1/2$ additions modulo F_n et autant de décalages par point de calcul. La conception de matériels réalisant de manière efficace ces opérations a été étudiée [13,14]. La transformée de FERMAT serait donc idéale si la longueur des mots requis ($b + 1$ bits) n'était sensiblement la moitié de la longueur de séquence désirée N (le quart si on prend pour α, au lieu de 2, une racine carrée de 2). Il existe plusieurs méthodes pour profiter quand même des avantages de la transformée de FERMAT dans le calcul d'une convolution longue [15,16,6], notamment en la remplaçant par la convolution de deux séquences à deux dimensions dont chaque dimension est de l'ordre de $\sqrt{N}$. Ainsi pour une convolution cyclique à 1 024 points comme on en rencontre fréquemment en traitement du signal, AGARWAL et BURRUS dans un article déjà ancien [11] annoncent un gain de temps de 40 % - sur un ordinateur - par rapport au calcul par FFT. Ce chiffre peut certainement être amélioré notablement si l'on emploie un matériel spécialisé.

4. UTILISATION DES TRANSFORMEES ARITHMETIQUES

Jusqu'ici nous avons raisonné comme s'il était indifférent de calculer dans l'anneau Z_M ou dans celui des entiers naturels Z. C'est vrai si le résultat final, constitué par les valeurs de $z(n)$ données par (1), est identique lorsqu'on le calcule dans Z et lorsqu'on le calcule modulo M, autrement dit s'il prend au plus M valeurs ; cette condition s'écrit, compte tenu des deux signes possibles :

$$(9) \quad \left| \sum_{m=0}^{N-1} x(m) y(n-m) \right| < \frac{M}{2} \cdot \quad n = 0, 1, \ldots, N-1$$

On voit que ceci impose de choisir M assez grand, non seulement pour limiter les erreurs de quantification mais aussi pour que la dynamique du signal de sortie, s'il s'agit d'un filtrage, n'excède pas $M/2$. Toutefois, si M est un entier composite de la forme $M_1 M_2$ avec M_1 et M_2 premiers entre eux, en faisant les calculs séparément dans Z_{M_1} et dans Z_{M_2} on pourra reconstruire sans ambiguïté le résultat du calcul modulo M (théorème du reste chinois). Si la valeur de M donnée par (9) implique une longueur de mot excessive, ce procédé permet de la réduire. Notons un avantage de la transformée arithmétique sur la transformée de FOURIER pour

les calculs de convolution : elle ne comporte pas d'erreur d'arrondi ou d'erreur de quantification des fonctions trigonométriques.

En traitement du signal on pense généralement à la variable $x(n)$ comme à une valeur instantanée, à $\hat{x}(n)$ comme à une composante spectrale du signal. Avec les transformées arithmétiques la suite transformée n'a aucune interprétation physique de ce genre (mis à part le fait que, si la suite a une période T qui est un diviseur de N, la transformée aura la même structure de raies que le spectre de FOURIER). C'est pourquoi leur domaine d'application naturel est le filtrage ou la corrélation, car l'interprétation des résultats se fait, après passage dans un espace transformé, dans l'espace d'origine. S'il faut un temps T_A pour calculer une transformée arithmétique à N points et un temps T_F pour calculer la transformée de FOURIER discrète de même longueur (ou de longueur peu différente, puisque la structure arithmétique la plus favorable pour N n'est pas toujours la même dans les deux cas), la convolution de deux séquences impliquant essentiellement trois transformations et N multiplications (qui ont la même durée dans les deux cas) on a intérêt à employer la transformée arithmétique dès lors que $T_A < T_F$. Elle n'est dans ce cas, comme la TFD, qu'un intermédiaire de calcul. Les opérations préliminaires : échantillonnage, addition de zéros, quantification, sont les mêmes dans les deux cas.

Il en va différemment si nous considérons le calcul d'un spectre. Peut-on encore utiliser les transformées arithmétiques ? Oui si, au lieu de ramener la convolution à la transformée de FOURIER on fait cette fois l'inverse, suivant la formule bien connue :

$$(10)\quad \hat{x}(n) = e^{-i\pi\frac{n^2}{N}} \sum_{m=0}^{N-1} \left(x(m) e^{-i\pi\frac{m^2}{N}} \right) e^{i\frac{\pi}{N}(n-m)^2}$$

on se pénalise ainsi de $2N$ multiplications complexes (sauf si N est premier, auquel cas on appliquera la méthode de RADER [5]). Il reste une convolution complexe, qui peut se ramener à trois convolutions réelles, dont chacune nécessite deux transformations arithmétiques, puisque l'une des deux séquences est fixe et peut être transformée à l'avance. On voit que, cette fois, la condition pour que l'emploi des transformées arithmétiques soit avantageux serait $T_A < T_F/6$. Quoique ce résultat puisse être amélioré, en particulier par l'emploi de transformées arithmétiques adaptées aux séquences complexes, il illustre bien une des limitations des transformées arithmétiques. Considérons cependant le calcul du spectre croisé de deux signaux. Supposons qu'on l'effectue sur des échantillons de N points mais qu'il soit nécessaire d'accumuler les résultats de P échantillons pour obtenir la précision statistique souhaitée sur les N points du spectre. On a le choix entre : calculer l'intercorrélation entre les deux signaux et en prendre la transformée de FOURIER - ou bien : calculer la transformée de FOURIER de chaque signal en l'accumulant sur P échantillons, puis multiplier les deux spectres l'un par l'autre. La seconde méthode passe pour plus rapide du fait de l'existence des algorithmes FFT. Elle conduit à effectuer $2P$ transformations de FOURIER à N points et N multiplications. Mais si l'on choisit la première méthode, en calculant les intercorrélations à l'aide de transformées arithmétiques il suffira de cumuler les transformées des deux signaux sur P échantillons avant de les multiplier et d'effectuer ensuite la transformation inverse pour obtenir la fonction d'intercorrélation, dont il restera à prendre la transformée de FOURIER. D'où (si le module M est assez grand pour éviter les ambiguïtés) $(2P+1)$ transformations arithmétiques, N multiplications et une transformée de FOURIER. L'emploi des transformées arithmétiques s'avère donc avantageux si :

$$(2P+1)\,T_A < (2P-1)\,T_F$$

Ce résultat montre que l'utilisation des transformées arithmétiques peut faire gagner du temps aussi dans les calculs d'interspectre, et conduira parfois à renverser le point de vue selon lequel il est avantageux de remplacer les convolutions par des transformations de FOURIER.

REFERENCES BIBLIOGRAPHIQUES

[1] WINOGRAD (S). La complexité des calculs numériques.
La Recherche, n° 83, Nov. 1977.

[2] WINOGRAD (S). On Computing the Discrete Fourier Transform.
Math. of Comp. Vol. 32, pp. 175-199, Janv. 1978.

[3] RADER (C.M). Discrete Convolutions via Mersenne Transforms.
IEEE Trans. on Computers, Vol. C-21, pp. 1269-1273, Dec. 1972.

[4] BLUESTEIN (L.O). A Linear Filtering Approach to the Computation of Discrete Fourier Transform.
IEEE Trans. Audio. Electroacoust. Vol. AU-18, pp. 451-455, Dec. 1970.

[5] RADER (C.M). Discrete Fourier Transforms when the Number of Data Samples is Prime.
Proc. IEEE, Vol. 5, pp. 1107-1108, June 1968.

[6] AGARWAL (R.C), COOLEY (J.W). New Algorithms for Digital Convolution.
IEEE Trans. Vol. ASSP-25, pp. 392-410, Oct. 1977.

[7] NUSSBAUMER (H), QUANDALLE (P). Computation of Convolution and Discrete Fourier Transforms by Polynomial Transforms.
IBM J. Res. Develop. Vol. 22, pp. 134-144, March 1978.

[8] NICHOLSON (P.J). Algebraic Theory of Finite Fourier Transforms.
J. of computer and System Sciences, Vol. 5, pp. 524-547 (1971).

[9] DUBOIS (E), VENETSANOPOULOS (A). The Discrete Fourier Transform over Finite Rings with Application to Fast Convolution.
IEEE Trans. on Computers, Vol. C-27, pp. 586-593, Juil. 1978.

[10] POLLARD (J.M). The Fast Fourier Transform in a Finite Field.
Math. of Computation, Vol. 25, pp. 365-374, April 1971.

[11] AGARWAL (R.C), BURRUS (C.S). Number Theoretic Transforms to Implement Fast Digital Convolution.
Proc. IEEE, Vol. 63, pp. 550-560, April 1975.

[12] REED (I.S), TRUONG (T.K). A Fast Computation of Complex Convolution Using a Hybrid Transform.
IEEE Trans. Vol. ASSP-26, pp. 566-570, Dec. 1978.

[13] Mc/ CLELLAN (J.H). Hardware Realization of a Fermat Number Transform.
IEEE Trans. Vol. ASSP-24, pp. 216-225, June 1976.

[14] LEIBOWITZ (L.M). A Simplified Binary Arithmetic for the Fermat Number Transform.
IEEE Trans. Vol. ASSP-24, pp. 356-359, Oct. 1976.

[15] NUSSBAUMER (H). Digital Filtering Using Pseudo Fermat Number Transforms.
IEEE Trans. Vol. ASSP-25, pp. 79-83, Fev. 1977.

[16] AGARWAL (R.C), BURRUS (C.S). Fast One-Dimensional Digital Convolution by Multidimensional Techniques.
IEEE Trans., Vol ASSP-22, pp. 1-10, Fev. 1974.