Acta Cryst. (1992). A48, 415–418

Fast Fourier Transform Algorithms for Real and Symmetric Data

BY M. AN

Center for Large Scale Computation, City University of New York, New York, NY 10036-8099, *USA*

C. LU

Department of Computer and Information Sciences, Towson State University, Towson, MD 21204, *USA*

E. PRINCE*

Materials Science and Engineering Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, *USA*

AND R. TOLIMIERI†

Center for Large Scale Computation, City University of New York, New York, NY 10036-8099, *USA*

(*Received* 13 *June* 1991; *accepted* 19 *December* 1991)

Abstract

Procedures are described for computing the Fourier transforms of one-dimensional periodic sequences of real numbers and sequences that contain Hermitian and translational symmetry. Transforms from real sequences to Hermitian sequences and back are particularly efficient if the number of grid points in a period is two or four times an odd number. If the relation between points that are separated by half of a period is a change of sign or complex conjugate, periods that are a power of two times an odd number are also favorable for constructing algorithms that minimize redundant computations and complex multiplications.

Introduction

The computation of electron-density maps from structure factors and the inverse problem of structure factors from density maps require the computation of the Fourier transforms of functions defined at the nodes of a three-dimensional lattice that is a sublattice of the crystal lattice. These three-dimensional transforms can be computed by the row–column method, which is a sequence of one-dimensional transforms, utilizing procedures known as fast Fourier transforms, or FFTs. While Fourier transforms are, in general, computations involving complex numbers, electron densities are real quantities, and the structure factors have Hermitian symmetry, that is $F(-\mathbf{h}) = F(\mathbf{h})^*$. These facts may be used to reduce the required amount of computation and, further, as was shown by Ten Eyck (1973), space-group symmetry can be exploited to achieve additional reduction in computation. The effect of space-group symmetry sometimes introduces translational symmetry into the one-dimensional sequences whose transforms must be computed. For example, a 2_1 screw axis passing through the origin causes the values of the function at points separated by one half of a period in the direction parallel to the axis to be complex conjugates of one another. In this paper we describe algorithms that exploit various types of one-dimensional symmetry to improve the efficiency of FFTs.

Fast Fourier transform algorithms

Consider a periodic function defined at the nodes of a one-dimensional lattice by the values x_j for $j = 0, 1, \ldots, N-1$, with $x_{j+nN} = x_j$, where n and N are integers. Its discrete Fourier transform, or DFT, is defined for $k = 0, 1, \ldots, N-1$ by

$$X_k = \sum_{j=0}^{N-1} x_j w_N^{jk}, \tag{1}$$

* To whom all correspondence should be addressed.

† This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by the Air Force Office of Scientific Research under contract no. F49620-89-0-0020. The United States Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright and notation hereon.

0108-7673/92/040415-04$06.00

where $w_N = \exp(-2\pi i/N)$ and $i = \sqrt{-1}$. If x_j and X_k are treated as the elements of column vectors, $\mathbf{x}$ and $\mathbf{X}$, this may be written in matrix notation as

$$\mathbf{X} = \mathbf{F}_N \mathbf{x}, \tag{2}$$

where $\mathbf{F}_N$ is the matrix

$$\mathbf{F}_N = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & w_N & w_N^2 & \cdots & w_N^{N-1} \\ 1 & w_N^2 & w_N^4 & \cdots & w_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & w_N^{N-1} & w_N^{2(N-1)} & \cdots & w_N^{(N-1)^2} \end{pmatrix}.$$

This matrix is dense and complex, and the matrix multiplication in (2) requires $(N-1)^2$ complex multiplications. It is always possible, however, to express it as a product of sparse matrices, with the result that the actual number of operations is approximately proportional to $N \log N$. Such factorizations are the basis for constructing fast-Fourier-transform algorithms.

The sparse factors of $\mathbf{F}_N$ have various forms depending on whether N is prime or composite and, if it is composite, whether it has factors that are relatively prime, that is have no common factors larger than 1. The most efficient algorithm for a given N depends on computer architecture, but all are designed to minimize time-consuming operations, particularly complex multiplications, each of which requires four real multiplications and two real additions or (Blahut, 1985) three real multiplications and three real additions. Particular algorithms have been described by Cooley & Tukey (1965), Good (1958), Thomas (1963), Kolba & Parks (1977), Burrus & Eschenbacher (1979), Temperton (1985), Agarwal & Cooley (1987), Rader (1968) and Winograd (1978). An extensive summary of methods is given by Tolimieri, An & Lu (1989).

FFTs for real and Hermitian sequences

If the sequence x_j is composed of real numbers only, then its Fourier-transform sequence X_k has Hermitian symmetry, so that $X_{N-k} = X_k^*$. Conversely, if the sequence x_j has Hermitian symmetry, X_k is real. A sequence of N complex numbers that has Hermitian symmetry contains redundant information. Ten Eyck (1973) described, for even values of N, a procedure for reducing the real-to-Hermitian and Hermitian-to-real transforms to transforms of complex sequences of length $N/2$. In the real-to-Hermitian case, this procedure, which is based on the Cooley–Tukey (Cooley & Tukey, 1965) factorization, defines two real sequences, **r** and **s**, each with length $N/2$, by $r_j = x_{2j}$ and $s_j = x_{2j+1}$, and defines a complex sequence, **y**, of length $N/2$ by $y_j = r_j + is_j$. Then $Y_k = R_k + iS_k$,

$$R_k = R_{N/2-k}^* = (1/2)(Y_k + Y_{N/2-k}^*) \tag{3}$$

and

$$S_k = S_{N/2-k}^* = (-i/2)(Y_k - Y_{N/2-k}^*). \tag{4}$$

Finally,

$$X_k = X_{N-k}^* = R_k + w_N^k S_k \tag{5}$$

and

$$X_{N/2-k} = X_{N/2+k}^* = R_k - w_N^k S_k. \tag{6}$$

This procedure works for any even value of N, and the Hermitian-to-real case is its inverse, in which the transform of a complex sequence of length $N/2$ contains the real sequence in its real and imaginary parts. In both cases, however, the factor w_N^k multiplying S_k (commonly called a 'twiddle factor') requires $N/2$ complex multiplications in computing the transform. If, however, N is of the form $4n + 2$, that is twice an odd number, then an alternative procedure, based on the Good–Thomas (Good, 1958; Thomas, 1963) factorization, avoids complex multiplication. In this case, (1) can be written

$$X_k = \sum_{j=0}^{N/2-1} x_{2j} w_N^{2jk} + x_{N/2+2j} w_N^{(N/2+2j)k}, \tag{7}$$

which reduces to

$$X_k = \sum_{j=0}^{N/2-1} [x_{2j} + (-1)^k x_{N/2+2j}] w_{N/2}^{jk}. \tag{8}$$

Let $\mathbf{x}$ be a real sequence and let $r_j = x_{2j}$ and $s_j = x_{N/2+2j}$. Then $\mathbf{X}$, $\mathbf{R}$ and $\mathbf{S}$ are all Hermitian and, for $k = 0, 1, \ldots, (N/2-1)/2$,

$$X_{2k} = X_{N-2k}^* = R_{2k} + S_{2k} \tag{9}$$

and

$$X_{N/2+2k} = X_{N/2-2k}^* = R_{2k} - S_{2k}. \tag{10}$$

If one sets $\mathbf{y} = \mathbf{r} + i\mathbf{s}$, $\mathbf{Y} = \mathbf{R} + i\mathbf{S}$, and $\mathbf{R}$ and $\mathbf{S}$ are given in terms of $\mathbf{Y}$ in the same way as in (3) and (4). Thus the transform of a sequence of N real numbers is formed from the transform of a sequence of $N/2$ complex numbers in which the imaginary part of each is displaced by $N/2$ in the real sequence from its real part. Note that there is no complex multiplication in this twofold reduction in the length of the sequence.

The transform from Hermitian to real is the inverse process. Let $r_j = x_j + x_{N/2-j}^*$, $s_j = (-1)^j(x_j - x_{N/2-j}^*)$ and $y_j = r_j + is_j$. Then $Y_k = R_k + iS_k$ and, because **r** and s are Hermitian, R_k is real and iS_k is imaginary. Thus X_{2k} is found in the real part of Y_k and $X_{N/2+2k}$ is found in the imaginary part of Y_k.

A similar procedure can be used if N is four times an odd number. In this case the equations that are analogous to (7) and (8) are

$$X_k = \sum_{j=0}^{N/2-1} x_{2j}w_N^{2jk} + x_{N/4+2j}w_N^{(N/4+2j)k} \quad (11)$$

and

$$X_k = \sum_{j=0}^{N/2-1} [x_{2j} + (-i)^k x_{N/4+2j}]w_{N/2}^{jk}. \quad (12)$$

Let $r_j = x_{2j}$ and $s_j = x_{N/4+2j}$. Then

$$X_k = X_{N-k}^* = R_k + (-i)^k S_k. \quad (13)$$

If we again set $\mathbf{y} = \mathbf{r} + i\mathbf{s}$, $\mathbf{R}$ and $\mathbf{S}$ can be found from $\mathbf{Y}$ by (3) and (4). The transform of a real sequence with period N can be computed from the transform of a complex sequence with period $N/2$ in which the imaginary part of each element is displaced by $N/4$ in the real sequence from its real part. Again, no complex multiplication is required.

As before, the transform from Hermitian to real is the inverse process. Let $r_j = x_j + x_{N/2-j}^*$, $s_j = i^j(x_j - x_{N/2-j}^*)$ and $y_j = r_j + is_j$. Then $\mathbf{r}$ and $\mathbf{s}$ are Hermitian, R_k is real, iS_k is imaginary, X_{2k} is found in the real part of Y_k and $X_{N/4+2k}$ is found in the imaginary part of Y_k.

Translational symmetry

The discrete Fourier transform, as defined in (1), assumes that the function is periodic, that it is defined at evenly spaced points and that the sum is over a complete period. A three-dimensional Fourier transform can be separated into a sequence of one-dimensional transforms in which the input coefficients for the second dimension are the output coefficients for the first dimension, and the input for the third dimension is the output from the second dimension. When the space group contains screw axes or is based on a centered lattice, the inputs to the second- and third-dimensional transforms contain relationships like $x_{N/2+j} = -x_j$, $x_{N/2+j} = x_j^*$ or $x_{N/2+j} = -x_j^*$, which causes the full period to contain redundant information. Ten Eyck (1973) gives a procedure that is applicable to these cases, but it depends on a transformation that is singular at $j = 0$ and will tend to be ill conditioned for small values of j/N. As in the case of real and Hermitian sequences, however, if N is twice an odd number, the Good–Thomas factorization allows a straightforward solution to the problem. In this case it is apparent that a sequence consisting of only the even-numbered elements of $\mathbf{x}$ contains a complete set of information and is itself periodic. This principle can be extended to the case of $N = 2^k M$, where M is an odd number and k is a positive integer.

As an illustration, consider the case of $M = 5$, $k = 2$, so that $N = 20$. The one-dimensional sequence can be arranged in the two-dimensional array

$$\begin{matrix} x_0 & x_4 & x_8 & x_{12} & x_{16} \\ x_5 & x_9 & x_{13} & x_{17} & x_1 \\ x_{10} & x_{14} & x_{18} & x_2 & x_6 \\ x_{15} & x_{19} & x_3 & x_7 & x_{11} \end{matrix}.$$

The rows and columns of this array contain periodic subsets of the full sequence and, if $x_{10+j} = x_j^*$, the array can be rewritten

$$\begin{matrix} x_0 & x_4 & x_8 & x_2^* & x_6^* \\ x_5 & x_9 & x_3^* & x_7^* & x_1 \\ x_0^* & x_4^* & x_8^* & x_2 & x_6 \\ x_5^* & x_9^* & x_3 & x_7 & x_1^* \end{matrix}.$$

With the row–column method, the transform of the full sequence can be written as the consecutive transforms of the rows and columns of this array. Because the transform of the complex conjugate of a sequence is the complex conjugate of the mirror image of the transform of the sequence, the intermediate array resulting from the row transforms, which we denote by $\mathcal{X}_j$, is

$$\begin{matrix} \mathcal{X}_0 & \mathcal{X}_4 & \mathcal{X}_8 & \mathcal{X}_{12} & \mathcal{X}_{16} \\ \mathcal{X}_5 & \mathcal{X}_9 & \mathcal{X}_{13} & \mathcal{X}_{17} & \mathcal{X}_1 \\ \mathcal{X}_0^* & \mathcal{X}_{16}^* & \mathcal{X}_{12}^* & \mathcal{X}_8^* & \mathcal{X}_4^* \\ \mathcal{X}_5^* & \mathcal{X}_1^* & \mathcal{X}_{17}^* & \mathcal{X}_{13}^* & \mathcal{X}_9^* \end{matrix}.$$

Clearly only the transforms of the first two rows are needed to fill in this complete array, and the transforms of the first three columns of this array are sufficient to complete the transform of $\mathbf{x}$, in which $X_{20-2k} = X_{2k}^*$ and $X_{20-2k-1} = -X_{2k+1}^*$. If $x_{N/2+j} = -x_j^*$, the sign relations are reversed, so that $X_{N-2k} = -X_{2k}^*$ and $X_{N-2k-1} = X_{2k+1}^*$. If $x_{N/2+j} = -x_j$, then $X_{2k} = 0$ for all k.

Discussion

Although FFT routines based on factorizations of the $N \times N$ discrete Fourier-transform matrix can be written for any value of N, the routines for some values of N are relatively more efficient than for others, and the proportionality to $N \log N$ is only approximate. It is widely believed that N must be a product of small prime numbers. In fact, a popular reference on numerical methods (Press, Flannery, Teukolsky & Vetterling, 1986) recommends that, if the data are defined over a period whose size is not a power of two, they be filled with zeros up to the next-higher power of two. Actually, in the crystallographic case, where the transforms are between real and Hermitian data and where partial transforms contain symmetry, there is a positive advantage to periods for which

N is a small power of two times an odd number and, further, there is also an advantage if that odd number is a product of several different prime numbers rather than a high power of a small one. A library of efficient FFT routines for a wide range of values of N is being developed (An, Lu, Prince & Tolimieri, 1992).

References

AGARWAL, R. C. & COOLEY, J. W. (1987). *Proc. IEEE*, **75**, 1283–1292.
AN, M., LU, C., PRINCE, E. & TOLIMIERI, R. (1992). *Acta Cryst.* A**48**, 346–349.
BLAHUT, R. E. (1985). *Fast Algorithms for Digital Signal Processing*. Reading, MA: Addison-Wesley.
BURRUS, C. S. & ESCHENBACHER, P. W. (1979). *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-29**, 806–817.
COOLEY, J. W. & TUKEY, J. W. (1965). *Math. Comput.* **19**, 297–301.
GOOD, I. J. (1958). *J. R. Statist. Soc. B*, **20**, 361–372.
KOLBA, D. P. & PARKS, T. W. (1977). *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-25**, 281–294.
PRESS, W. H., FLANNERY, B. P., TEUKOLSKY, S. A. & VETTERLING, W. T. (1986). *Numerical Recipes*. Cambridge Univ. Press.
RADER, C. M. (1968). *Proc. IEEE*, **56**, 1107–1108.
TEMPERTON, C. (1985). *J. Comput. Phys.* **58**, 283–299.
TEN EYCK, L. F. (1973). *Acta Cryst.* A**29**, 183–191.
THOMAS, L. H. (1963). *Using a Computer to Solve Problems in Physics, Applications of Digital Computers*. Boston: Ginn & Co.
TOLIMIERI, R., AN, M. & LU, C. (1989). *Algorithms for Discrete Fourier Transform and Convolution*. New York: Springer.
WINOGRAD, S. (1978). *Math. Comput.* **32**, 175–199.

Acta Cryst. (1992). A**48**, 418-423

Exact Conditional Joint Probability Distribution of a Three-Phase Invariant in Space Group *P*2. I. Derivation of the Fourier Coefficients

BY URI SHMUELI

School of Chemistry, Tel Aviv University, 69978 Tel Aviv, Israel

AND GEORGE H. WEISS

Physical Sciences Laboratory, Division of Computer Sciences and Technology, National Institutes of Health, Bethesda, Maryland 20892, USA

(*Received 20 May 1991; accepted 30 September 1991*)

Abstract

We extend our study of the conditional probability density function (c.p.d.f.) of the three-phase invariant for the space group $P1$ [Shmueli, Rabinovich & Weiss (1989). *Acta Cryst.* A**45**, 361-367] to the monoclinic space group $P2$. A detailed derivation of the characteristic function (and hence Fourier coefficients) of the latter c.p.d.f. is presented in this paper, as well as some simplifications of the resulting expressions.

Introduction

The first exact study of the conditional probability density function (p.d.f.) of a three-phase invariant was presented recently (Shmueli, Rabinovich & Weiss, 1989*a*) in terms of a Fourier representation of the relevant hexavariate p.d.f. The resulting series was then adapted to computer evaluation by suitably partitioning the sums and taking their symmetry into account (Shmueli, Rabinovich & Weiss, 1989*b*). Although the formalism appeared to be extremely complicated it was seen that by properly exploiting the symmetry inherent in the Fourier summations it is possible to reduce the computing efforts sufficiently that conventional mainframes and workstations are able to cope with the relevant computations. The study cited earlier contains a derivation of the general form of the conditional p.d.f. as well as its evaluation for the space group $P1$ - the simplest and, so far, the only example for which noncentrosymmetric direct-methods formalisms have been extensively developed. Our earlier study shows that the general form of the conditional p.d.f. is given by a summation of the form

$$\sum_{\mathbf{u}} C_{\mathbf{u}} Z_{\mathbf{u}} \qquad (1)$$

where $\mathbf{u}$ is a vector of the (six) summation indices and the $C_{\mathbf{u}}$ are coefficients depending on the composition and symmetry of the crystal. The function $Z_{\mathbf{u}}$ depends on the magnitudes of the normalized structure factors and is the same for all the symmetries and compositions. While the conditional p.d.f. for the three-phase invariant in $P1$, in either its approximate (Cochran, 1955) or exact (Shmueli *et al.*, 1989*a*) form, may lead to satisfactory practical algorithms, we believe that it is also desirable to examine the effect of symmetry on this important statistic. To do this, we need only calculate $C_{\mathbf{u}}$ in (1). As pointed out, *e.g.* by Shmueli & Weiss (1985), these Fourier coefficients are just the values of the characteristic function $C(\omega_1, \omega_2, \ldots, \omega_k, \ldots)$ of the p.d.f. at the

0108-7673/92/040418-06$06.00