INFORME

PROGRAMACION DE LA TRANSFORMADA RAPIDA DE FOURIER DE BASE 4 Y 8 PARA PROCESAMIENTO EN TIEMPO REAL

Presentado

por

César La Hoz

Ronald Woodman

RADIO OBSERVATORIO DE JICAMARCA

ENERO 1975

INFORME

PROGRAMACION DE LA TRANSFORMADA RAPIDA DE FOURIER DE BASE 4 Y 8 PARA PROCESAMIENTO EN TIEMPO REAL

Por César La Hoz

0. EXTRACTO

Se informa sobre la implementación de la transformada Rápida de Fourier para su aplicación en el estimado de funciones de densidad espectral de ecos ionosféricos en tiempo real. Se hace una descripción del algoritmo básico mostrándose que la eficiencia de la TRF es proporcional por lo menos a $N\log_r N$ en lugar de N^2. Se describen los criterios que conducen a una estructura óptima del programa, estableciéndose las relaciones que dan el número de operaciones exactas en el cálculo de la TRF de base arbitraria. Se informa que el programa de base 8 admite un ancho de banda en tiempo real de 6 kHz, ó el procesamiento espectral de 15 alturas de ecos ionosféricos con un periodo entre-pulsos de 2.5 mseg.

1. INTRODUCCION.

Siendo la Transformada Rápida de Fourier (1) un algoritmo sumamente eficiente en evaluar la Transformada Discreta de Fourier (2,3) se ha hallado la conveniencia de su uso para el cálculo de funciones de densidad espectral en programas de tiempo real que procesan ecos ionosféricos obtenidos en las instalaciones del Radio Observatorio de Jicamarca.

Esta comunicación informa sobre la implementación de dos programas optimizados de la TRF, uno con iteraciones de base 4, y otro de base 8, codificados en lenguaje ensamblador para la computadora Datacraft de Jicamarca.

2. EL ALGORITMO BASICO.

La TRF es conveniente para registros cuya longitud N es un número altamente compuesto, $N = \Pi r_i = r_1 r_2 \ldots r_m$. El tiempo de computación es proporcional a $N(r_1+r_2+\ldots+r_m)$, comparado con N^2 para el caso de la evaluación convencional.

La situación más conveniente es cuando las r_1 son todas iguales, esto es, $N = r^m$, y el tiempo de cálculo es entonces proporcional a $N\log_r N = Nm$. Este es el caso que consieraremos.

Existen dos formas canónicas de la TRF, la versión original de Cooley-Tukey (1), y la Sande-Tukey (4). En el presente informe desarrollaremos la versión S-T. Convenientemente abordadas ambas conducen a programas igualmente eficientes. Sin embargo, recientemente se ha visto que la forma C-T puede ser más eficiente para datos entrelazados como los que se obtienen con un radar de alcance como el de Jicamarca. Este punto será discutido en un próximo informe.

Consieraremos la evaluación de la TDF compleja

$$X(j) = \sum_{k=0}^{N-1} A(k)W_N^{jk} \qquad j=0,1,2,..,N-1 \quad k=0,1,2,..,N-1$$

donde $W_N=\exp(2\ i/N)$, i es la unidad imaginaria, y W_N la enésima raíz de la unidad.

Pongamos $N=r^m$, entonces podemos escribir

$$j=j_{m-1}r^{m-1} + j_{m-2}r^{m-2} +\ldots+ j_1 r + j_0$$

$$j=k_{m-1}r^{m-1} + k_{m-2}r^{m-2} +\ldots+ k_1 r + k_0$$

donde $j_{m-1},j_{m-2},\ldots,j_0,k_{m-1}k_{m-2},\ldots,k_0 = 0,1,\ldots,r-1$
lo cuál equivale a hacer un mapeo biunívoco entre j y k y los conjuntos $(j_{m-1},j_{m-2},\ldots,j_0)$ y $(k_{m-1},k_{m-2},\ldots,k_0)$ respectivamente.

La derivación posterior (ver Apéndice I) conduce a un conjunto de m ecuaciones recursivas de la forma

$$A_p(j_0,j_1,\ldots,j_{p-1},k_{m-p-1},\ldots,k_0)=$$

$$\left[\sum_{k_{m-p}} A_{p-1}9j_0,j_1,\ldots,j_{p-2},k_{m-p},\ldots,k_0)W_r^{j_{p-1}k_{m-p}}\right] \tag{1}$$

$$.W_N^{j_{p-1}r^{p-1}(k_{m-p-1}r^{m-p-1}+\ldots+k_0)} \qquad p=1,..,m\ A_0=A$$

La última recursión nos da

$$X(j_{m-1},j_{m-2},\ldots,j_0) = A_m(j_0,j_1,\ldots,j_{m-1})$$

lo cual significa que el resultado, después de m iteraciones, se halla en el arreglo A_m en orden de 'r-bits' invertidos.

La recursión (1) la interpretamos como desarrollos de Fourier de r puntos (la expresión entre corchetes), multiplicados por un factor de rotación W_N^u. Estas dos operaciones son conocidas como las mariposas o núcleos de la TRF. Nótese que en cada iteración , cada mariposa emplea r puntos una sola vez para evaluar otros tantos nuevos. Esto significa que el cálculo puede hacerse in situ con el empleo de a los más r direcciones temporales de memoria. Nótese también què si r=2,4,8 se logran importantes ahorros en la evaluación de cada transformada de r puntos, y la única operación compleja (4 multiplicaciones reales y 2 sumas) sería la multiplicación por el factor de rotación. Más adelante se darán los resultados acerca del conteo de operaciones complejas.

3.CONSIDERACIONES GENERALES EN PROGRAMAR LA TRF.

El esfuerzo del presente trabajo está encaminado a la obtención de un algoritmo lo más veloz posible en la evaluación de estimados espectrales. Este planteamiento determina los criterios que deben seguirse en la programación. Por ejemplo se ha considerado aceptable mejorar el tiempo a costa de un incremento en el uso de memoria. Así mismo toda la aritmética se realiza con punto fijo.

Los principales problemas asociados con la codificación del programa se relacionan con el direccionamiento de los datos, y con la generación de los exponentes del factor de rotación.

Existen tres lazos principales, el más exterior cuenta el número de recursiones y se realiza m veces. En el lazo intermedio se calculan los exponentes del factor de rotación. En el lazo más interior se realiza la aritmética de mariposas que consiste en el cálculo de N/r transformadas de r puntos en cada iteración, y la multiplicación por el factor de rotación. Cada etapa se divide en grupos, de tal manera que cada grupo requiere del mismo factor de rotación.

Se han eliminado en forma especial las multiplicaciones por $W_N^0 = 1$. No se realizan cálculos de las funciones trigonométricas, y más bien se las obtiene de una tabla mínima llenada previamente.

El esquema descrito corresponde a la estructura óptima del algoritmo desde el punto de vista de la programación, con miras a minimizar el tiempo de cálculo.

4. MARIPOSAS DE BASE 4 Y 8.

Los bloques elementales de la TRF lo constituyen las llamadas mariposas de base r que consisten en transformadas de r puntos, más una multiplicación por el factor de rotación. En los casos de r=2,4 la aritmética consiste sólo de sumas y restas, y para el caso de $r=8$ de sumas, restas y cuatro multiplicaciones reales debido al factor de $W_8 = \exp(\Pi i/4) = 1/\ 2(1+i)$. Estas mariposas conducen a mejoras considerables por el ahorro de multiplicaciones, que es la operación más lenta. Singleton (5,6) ha informado que estas mejoras son del orden de 25% y 33% para bases de 4 y 8 respectivamente, en comparación a la base 2.

El desarrollo de una mariposa de base 4 es

$$A_0 = \big((x_0 + x_2) + (x_1 + x_3)\big)\ W_4^{\,0}$$

$$A_2 = \big((x_0 + x_2) - (x_1 + x_3)\big)\ W_4^{u2}$$

$$A_1 = \big((x_0 - x_2) + i(x_1 - x_3)\big)\ W_4^{u1}$$

$$A_3 = \big((x_0 - x_2) - i(x_1 - x_3)\big)\ W_4^{u3}$$

El orden de operaciones corresponde a un mínimo en la aritmética, que es de 4 sumas, 4 restas y tres multiplicaciones complejas.

Siguiendo el mismo lineamiento anterior, la mariposa de base 8 es

$$A_0 = \big(\{(x_0+x_4) + (x_2+x_6)\} + \{(x_1+x_5) + (x_3+x_7)\}\big)\ W_8^0$$

$$A_4 = \big(\{(x_0+x_4) + (x_2+x_6)\} - \{(x_1+x_5) + (x_3+x_7)\}\big)\ W_8^{u4}$$

$$A_2 = \big(\{(x_0+x_4) - (x_2+x_6)\} + \{(x_1+x_5) - (x_3+x_7)\}i\big)\ W_8^{u2}$$

$$A_6 = \big(\{(x_0+x_4) - (x_2+x_6)\} - \{(x_1+x_5) - (x_3+x_7)\}i\big)\ W_8^{u6}$$

$$A_1 = \left(\{(x_0-x_4) + (x_2-x_6)i\} + \{(x_1-x_5) + (x_3-x_7)\}W_8^1\right)W_8^{u1}$$

$$A_5 = \left(\}(x_0-x_4) + (x_2-x_6)i\} - \{(x_1-x_5) + (x_3-x_7)\}W_8^1\right)W_8^{u5}$$

$$A_3 = \left(\{(x_0-x_4) - (x_2-x_6)i\} + \{(x_1-x_5) - (x_3-x_7)\}\ iW_8^1\right)W_8^{u3}$$

$$A_7 = \left(\{(x_0-x_4) - (x_2-x_6)i\} - \{(x_1-x_5) - (x_3-x_7)\}\ iW_8^1\right)W_8^{u7}$$

Este esquema corresponde a 10 sumas y 10 restas complejas, cuatro multiplicaciones reales, y siete multiplicaciones complejas. El orden de las operaciones corresponde también a una secuencia óptima para el cálculo.

5. CONTANDO OPERACIONES DE UNA TRF DE BASE ARBITRARIA

Suponemos como antes que $N=r^m$.

- Número de iteraciones:

$$NL_1 = m$$

- Número de cálculos de factor de rotación:

$$NL_2 = (r^{m-1}-1) + (r^{m-2}-1) + \dots + (r-1)$$
$$= \frac{1}{r-1}(N-1)-m$$

- Número de mariposas ú operaciones complejas:

$$NL_3 = 1(r^{m-1}-1)+r(r^{m-2}-1)+r^2(r^{m-3}-1)+ \dots +r^{m-2}(r-1)$$
$$= \frac{1}{r(r-1)}\left(\{m(r-1)-r\ r^m-r\right)$$

- Número de multiplicaciones por $W_N^0 = 1$:

$$NL_{11} = 1 + r + r^2 = \dots + r^{m-1}$$
$$= \frac{1}{r-1}(N-1)$$

Estos resultados corresponden a cada uno de los la zos de los programas que se han implementado. Así NL_1 viene a ser el número de veces que se repite el lazo exterior.

Se ha verificado que el programa desarrollado de base 8 es 50% más eficiente que un programa de base 2, ambos codificados con los mismos criterios.

6. PERFORMANCE DE LOS PROGRAMAS DESARROLLADOS PARA N = 64 $4^3 = 8^2$.

-El programa de base 4 FFT4.

Con las relaciones del acápite anterior se ha calculado exactamente que este programa tarda 9.668 mseg. en hallar la transformada de 64 puntos complejos. Ocupa 231 direcciones de memoria, sin contar la tabla de funciones trigonométricas. Esta última ocupa 63 palabras adicionales.

-El programa de base 8 FFT8.

Este programa tarda 8.031 mseg, y ocupa 474 palabras de memoria, más 63 palabras para la tabla trigonométrica. Comparado con un programa descrito por Greenwald, el nuestro es casi tres veces más rápido, y comparado con el de M. Ierkic para la misma computadora y con iteraciones de base 2, es dos veces más rápido.

El ancho de banda de tiempo real de FFT8 aplicado a la evaluación de funciones de densidad espectral con integración en doble precisión entera es aproximadamente de 6kHz, lo cuál implica que se podrían procesar aproximadamente 15 alturas de ecos ionosféricos con un período entre-pulsos (IPP) de 2.5 mseg, ó 400 Hz.

7. RECETAS DE USO.

```
CALL FFT4(X,CST,IX,N)     ó
CALL FFT8(X,CST,IX,N)
```

donde X es un arreglo entero-complejo que va a ser transformado. La parte real e imaginaria ocupa posiciones consecutivas. El resultado es devuelto en el mismo arreglo en orden de 'r-bits' (r=4,8) invertidos.

CST es una tabla de cosenos-senos intercalados compatible con el valor de N, que ha sido llenada previamente. Esta tabla debe ser en punto fijo.

IX es un apuntador que cuando es positivo se efectúa síntesis de Fourier; cuando es negativo se efectúa análisis de Fourier.

N Dimensión de arreglo X.

APENDICE I

Supongamos que $N = r^m$, y consideramos la evaluación de la Transformada Discreta de Fourier compleja

$$X(j) = \sum_{k=0}^{N-1} (A(k)W_N^{jk} \qquad j=0,1,2,..,N-1 \; k=0,1,2,..,N-1$$

donde $W_n = \exp(2\ i/N)$, la enésima raíz principal de la unidad.

Pongamos $N = r^m$, entonces podemos escribir

$$j=j_{m-1}r^{m-1} + j_{m-2}r^{m-2} +\ldots+ j_1 r + j_0 \qquad (2)$$

$$k=k_{m-1}r^{m-1} + k_{m-2}r^{m-2} +\ldots+ k_1 r + k_0 \qquad (3)$$

donde $j_{m-1}, j_{m-2},\ldots, j_0, k_{m-1}, k_{m-2},\ldots,k_0 = 0,1,\ldots,4-1$ (3)

lo que equivale a hacer un mapeo biunívoco entre j y k y los conjuntos $(j_{m-1}, j_{m-2},\ldots, j_0)$ y $(k_{m-1}, k_{m-2},\ldots,k_0)$ respectivamente.

Poniendo (2) y (3) en (1)

$$X\ (j_{m-1}, j_{m-2},\ldots,j_0) = \sum_{k_0} \sum_{k_1} \cdots \sum_{k_{m-1}} A(k_{m-1}, k_{m-2},\ldots,k_0)W_N^{jk} \qquad (4)$$

Notamos que:

$$W_N^{jk} = W_N^{j_0 k}\ W_N^{(j_{m-1}r^{m-1} + j_{m-2}r^{m-2}+\ldots+j_1 r)k}$$

Sustituyendo en (4)

$$X(\ldots\ldots) = \sum_{k_0} \sum_{k_1} \cdots \sum_{k_{m-2}} \left[\sum_{k_{m-1}} A(k_{m-1},k_{m-2},\ldots,k_0) W_N^{j_0(k_{m-1}r^{m-1}+\ldots+k_0)} \right]$$

$$.W_N^{(j_{m-1}r^{m-1}+\ \ldots\ +\ j_1 r)k} \qquad (5)$$

Si ponemos $A_1(j_0,k_{m-2},k_{m-3},\ldots,k_0) = \sum_{K_{m-1}} A(\ldots\ldots) W_N^{j_0(k_{m-1}r^{m-1}+\ldots+k_0)}$

Sustituyendo en (5)

$$X(\ldots\ldots) = \sum_{k_0} \sum_{k_1} \cdots \sum_{k_{m-2}} A_1(j_0,k_{m-2},\ldots,k_0) W^{(j_{m-1}r^{m-1}+\ldots+j_1 r)k}$$

Notemos que:

$$W_N^{(j_{m-1}r^{m-1}+\ldots+j_1 r)k} = W_N^{j_1 rk}\ W_N^{(j_{m-1}r^{m-1}+\ldots+j_2 r^2)k}$$

$$= W_N^{j_1 r(k_{m-2}r^{m-2}+\ldots+k_0)}\ W^{(j_{m-1}r^{m-1}+\ldots+j_2 r^2)k}$$

debido a que W_N^{1N} para 1 cualquier entero.

Podemos entonces hacer:

$$A_2(j_0,j_1,k_{m-3},\ldots,k_0) = \sum_{km-2} A_1(j_0,k_{m-2},\ldots,k_0) W_N^{j_1 r(k_{m-2}r^{m-2}+\ldots+k_0)}$$

entonces:

$$X(\ldots\ldots) = \sum_{k_0} \sum_{k_1} \cdots \sum_{k_{m-4}} \sum_{k_{m-3}} A_2(j_0,j_1,k_{m-3},\ldots k_0) W_N^{(j_{m-1}r^{m-1}+\ldots+j_2 r^2)k}$$

Continuando en forma similar llegamos a un conjunto de ecua ciones resursivas:

$$A_p(j_0,j_1\dots,j_{p-1},k_{m-p-1},\dots k_0)=\sum_{k_{m-p}} A_{p-1}(j_0,j_1,\dots j_{p-2},k_{m-p},\dots\dots,k_0)$$

que finalmente nos da:

$$X(j_{m-1},j_{m-2},\dots,j_0)=A_m(j_0,j_1,\dots\dots,j_{m-1})$$

lo que significa que el resultado, después de m iteraciones, se halla en el arreglo Am en orden de "r-bits" invertidos.

En (6) podemos separar dos partes:

$$W_N^{j_{p-1}r^{p-1}(k_{m-p}r^{m-p}+k_{m-p-1}r^{m-p-1}+\dots+k_0)}=$$

$$W_N^{j_{p-1}r^{p-1}k_{m-p}r^{m-p}}\,W_N^{j_{p-1}r^{p-1}(k_{m-p-1}r^{m-p-1}+\dots+k_0)}$$

$$=W_r^{j_{p-1}k_{m-p}}\,W_N^{j_{p-1}(k_{m-p-1}r^{m-p-1}+\dots+k_0)}$$

Sust. en (6)

$$A_p(j_0,j_1,\dots,j_{p-1},k_{m-p-1},\dots,k_0)=\left[\sum_{k_{m-p}} A_{p-1}(j_0,j_1,\dots,j_{p-2},k_{m-p},\dots,k_0)W_r^{j_{p-1}j_{m-p}}\right]$$

$$\cdot W_N^{j_{p-1}r^{p-1}(k_{m-p-1}r^{m-p-1}+\dots+k_0)}\quad \begin{matrix} p=1,2,\dots,m \\ A_0=A \end{matrix}$$

Esta última relación la interpretamos como desarrollos de Fourier de r puntos (la expresión entre cohetes), multiplicados por un factor de rotación W_N^1.

Si los desarrollos de Fourier no contienen multiplicaciones complejas efectivas ($r=2,4$), en cada iteración se hacen $r(r^{m-1})$ MC. En m iteraciones se harán $mr^m = N\log_r N$ multiplicaciones.

Si los desarrollos de Fourier no contienen multiplicaciones complejas efectivas $(r=2,4)$, en cada itera ción se hacen $r(r^{m-1})$ MC. En m iteraciones se harán $mr^{m} = N\log_{r}N$ multiplicaciones.

REFERENCIAS

1. J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", Math. Computation, vol. 19, pp. 297-301, April 1965.
2. J. W. Cooley, P. A. W. Lewis and P. D. Welch, "The Finite Fourier Transform", IEEE Trans. Audio and Electroacoustics, vol. AU-17, pp. 77-a5, June 1969.
3. G-AE Subcommitee on Measurement Concepts, "What is the Fast Fourier Transform?", IEEE Trans. Audio and Electroacoustics, vol. AU-15, pp. 45-55, June 1967.
4. W. M. Gentleman and G. Sande, "Fast Fourier Transform for fun and profit", 1966 Fall Joint Computer Conf., AFIPS Proc., vol. 29. Washington, D. C.: Spartan, 1966, pp.563-578.
5. R. C. Singleton, "An Algorithm for Computing the Miexed Radix Fast Fourier Transform", IEEE Trans. Audio and Electroacoustics, vol. AU17, pp. 93-103, June 1969.
6. G. D. Bergland, "A Fast Fourier Transform using Base 8 Iterations", Math. of Computation, vol.22, pp. 275-279, April 1968.
7. R. A. Greenwald, "Doppler Spectrum Analysis using a Minicomputer", NOAA Technical Rep. ERL 241-AL 7, Boulder, Colo., July 1972.
8. M. H. Ierkic, Comunicación Personal, Jicamarca 1975.