

Tratamiento numérico de señales

En el campo de las comunicaciones detectamos claramente la tendencia del procesamiento de datos. Ya hace años la modulación de pulsos ha servido para una mayor utilización del canal conductor de la información mientras la fuente seguía análoga. Mientras tanto la técnica digital ha desplazado también la técnica análoga en el campo del procesamiento. Antes el filtrado, amplificación, multiflexado y amortiguación eran procesos netamente análogos. Actualmente se busca diseñar una red telefónica basada en la generación digital en la fuente, en la transferencia digital y finalmente en el procesamiento digital de la señal.

La transformación discreta de Fourier juega un papel importante en el análisis, en el diseño y en la implementación de algoritmos y sistemas de procesamiento digital de datos. Su significado se expresa en el creciente número de procedimientos para realizar tal información.

Este trabajo pretende expresar en forma sencilla en qué consiste el método.

ANDRES DOCKX
 Ingeniero Electrónico
 Universidad de Amberes, Bélgica
 Ingeniero Física Nuclear
 Universidad de Bruselas, Bélgica
 Profesor Asociado
 Facultad de Ingeniería
 Universidad Nacional

INTRODUCCION

Los avances obtenidos por los ingenieros al producir sistemas de procesamiento de datos generalmente provienen del desarrollo de dispositivos electrónicos. Esto no fue el caso con las aplicaciones que se refieren al análisis de datos y de señales mediante la transformación de Fourier discreta (TFD).

En esta aplicación los avances no salieron de la electrónica sino de las matemáticas.

La TFD es un método útil para extraer la información contenida en muchas clases de ondas como son las ondas sísmicas, electroencefalogramas o señales que provienen del espacio extraterrestre. Ya fueron investigados muchos métodos para definir el contenido energético correspondiente a cada frecuencia. Un método obvio consiste en usar un banco de filtros, cuyas salidas seleccionan una determinada componente del espectro frecuencial. Tiene el inconveniente de todo método análogo, falta de resolución y flexibilidad.

Las técnicas digitales son mejores pero difíciles de explicar y costosas en tiempo de computador. En el año 1965, dos científicos Cooley y tuckey desarrollaron un método para acelerar el cálculo de las componentes frecuenciales de una onda de amplio espectro. El método fue bautizado como transformada de Fourier rápida (TFR) porque redujo apreciablemente el tiempo computacional requerido para definir las componentes espectrales. En aquel instante ya fue interesante utilizar el computador en un amplio rango de problemas de diseño, en aplicaciones como el análisis de datos, en el procesamiento de señales. Para el ingeniero que se ocupa del filtrado de datos o de la distribución espectral de la potencia dentro de una señal eléctrica, el método mencionado es de gran importancia.

Esto condujo al desarrollo de programas TFR para computadores de propósito general. Pero aún con la TFR tales computadoras resultaron impotentes cuando las aplicaciones aportaban mu-

chos datos o cuando se requerían los resultados con urgencia. Por ejemplo el procesamiento de señales en tiempo real exige una rápida ejecución. Tal, que se ha tratado de construir procesadores de propósito especial, organizadas para esta aplicación.

Desde el punto de vista del diseño, la TRF permite el uso de métodos especiales que requieren menor tiempo de ejecución. Por ejemplo, porque el método presenta una secuencia aritmética siempre igual, la velocidad de la secuencia aritmética puede adaptarse a la memoria que le permite una simultaneidad entre las operaciones aritméticas y las transferencias de la memoria. Además ambas unidades pueden manejar la parte real e imaginaria en paralelo. Antes de entrar en el estudio del método revisamos algunos aspectos teóricos.

Serie de Fourier.

Tomaremos como punto de partida la serie de Fourier de funciones periódicas con período

$$T = \frac{2\pi}{\omega_0} \quad \text{Sea } n \text{ el número armónico.}$$

Los coeficientes de Fourier son:

$$a_n = 2/T \int_T f(t) \cos n\omega_0 t \, dt \quad (1)$$

$$b_n = 2/T \int_T f(t) \sin n\omega_0 t \, dt \quad (2)$$

$$a_0 = 1/T \int_T f(t) \, dt \quad (3)$$

donde: $a_n = a(n\omega_0)$ y $b_n = b(n\omega_0)$

La función periódica está conformada por la serie siguiente:

$$f_{T(t)} = a_0 + \sum_{n=1}^{\infty} (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t) \quad (4)$$

La expresión (4) se conoce comunmente como serie de Fourier o la forma trigonométrica. Existe una forma más concisa, conocida como la forma exponencial. Hagamos la siguiente substitución:

$$c_n = \frac{a_n - j b_n}{2} = 1/t \int f(t) e^{-jn\omega_0 t} \, dt \quad (5)$$

la expresión (4) se reduce a:

$$f_{T(t)} = a_0 + \sum_n c_n e^{jn\omega_0 t} \quad (6)$$

donde: $n = \dots -2, -1, +1, +2, \dots$

La deducción se hace en la siguiente forma, la ecuación (6) se transforma como:

$$f_{T(t)} = a_0 + 1/2 \sum_n (a_n - j b_n) (\cos n\omega_0 t + j \sin n\omega_0 t)$$

$$f_{T(t)} = a_0 + 1/2 \sum_n (a_n \cos n\omega_0 t + b_n \sin n\omega_0 t - j b_n \cos n\omega_0 t + j a_n \sin n\omega_0 t)$$

Teniendo en cuenta que n asume tanto un valor positivo como negativo y recordando que la función coseno es una función par y la función seno impar, podemos escribir:

$$\begin{aligned} a_n \cos n\omega_0 t &= a_{-n} \cos n\omega_0 t \\ b_n \sin n\omega_0 t &= b_{-n} \sin n\omega_0 t \\ b_n \cos n\omega_0 t &= -b_{-n} \cos n\omega_0 t \\ a_n \sin n\omega_0 t &= -a_{-n} \sin n\omega_0 t \end{aligned}$$

Definiendo: $c_0 = 1/T \int f(t) \, dt$

$$f_{T(t)} = \sum_n c_n e^{jn\omega_0 t} \quad (7)$$

$$n = \dots -2, -1, 0, 1, 2$$

La figura 1 muestra los componentes espectrales (las amplitudes espectrales) de una función (onda) periódica.

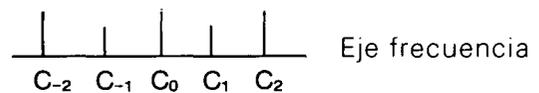


FIGURA 1. Componentes espectrales de una onda periódica.

INTEGRAL DE FOURIER

La función no periódica se caracteriza por una no-repetición de un determinado ciclo, lo que matemáticamente corresponde a un período infinito.

La distancia frecuencial entre las armónicas tiende a cero. Sea tal distancia $\Delta\omega$ infinitesimal De donde:

$$\begin{aligned} \Delta\omega &= (n + 1)\omega_0 - n\omega_0 = \omega_0 = \frac{2\pi}{T} \\ \frac{2}{T} &= \frac{\Delta\omega}{\pi} & \Delta f &= \frac{1}{T} \end{aligned}$$

Si las frecuencias se agrupan podemos considerar una función continua de frecuencia, es decir, el espectro de líneas pasa a ser un espectro análogo, lo que conduce a las igualdades siguientes:

$$a_{(n\omega_0)} = a_{(\omega)}; b_{(n\omega_0)} = b_{(\omega)}; c_{(n\omega_0)} = c_{(\omega)}$$

Así: (1), (2), (5) se transforman en:

$$a_{(\omega)} = \Delta\omega/\pi \int f(t) \cos\omega t \, dt$$

$$b_{(\omega)} = \Delta\omega/\pi \int f(t) \sin\omega t \, dt$$

$$c_{(\omega)} = \Delta\omega/2\pi \int f(t) e^{-j\omega t} \, dt$$

La transformada de Fourier se define como:

$$F_{(\omega)} = \lim_{\Delta f \rightarrow 0} c_{(\omega)}/\Delta f \quad \text{tal que}$$

$$F_{(\omega)} = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} \, dt \quad (8)$$

Aplicando el límite T

$$\rightarrow \infty (\Delta f \rightarrow 0): f(t) = \lim_{\Delta f \rightarrow 0} f_{T(t)} = \int_{-\infty}^{\infty} F_{(\omega)} e^{j\omega t} \, df \quad (9)$$

Discretización de las integrales de Fourier

La transformación de Fourier tal como fue definida en las ecuaciones (8) y (9) no tiene la forma adecuada para un tratamiento numérico. Esto proviene por una parte de la existencia de

una variable continua (análoga) que representa la frecuencia y por otra parte de una necesidad de hacer intervenir un número infinito de muestras. En vista de la considerable importancia en el tratamiento de señales, será necesario ponerlas bajo una forma utilizable. Esta forma recibe el nombre de la transformación de Fourier discreta, que se torna atractiva por la existencia de un algoritmo particularmente rápido y eficaz. Además la representación numérica de una señal análoga por un muestreo periódico guarda la correlación y la convolución puesto que la TFD también evalúa funciones de correlación y productos de convolución análogos. La figura 2 muestra cómo una señal análoga se muestrea. En la expresión (8) reemplazamos las variables continuas:

$$t = k\Delta t$$

$$T = K\Delta t$$

El reemplazo de la variable continua

$$\omega = n\Delta\omega$$

$$f = n\Delta f$$
(10)

Donde Δf es un incremento utilizado sobre el eje frecuencial. Las frecuencias discretas $n\Delta f$ se llaman frecuencias armónicas de la transformación de Fourier discreta (TFD).

Al sustituir los límites $-\infty, \infty$, por los límites cero hasta T_{seg} y la integral por una suma conseguimos:

$$F_{(n\Delta\omega)} = \Delta t \sum_k f_{(k\Delta t)} e^{-j2\pi n(k\Delta t)/K\Delta t}$$

Al poner $F_{(n\Delta\omega)} = F_{(n)}$; $f_{(k\Delta t)} = f_{(k)}$

simplificamos la notación tal

$$F_{(n)} = \Delta t \sum_k f_{(k)} e^{-2\pi nk/K}$$
(11)

La figura 2 indica el muestreo de una señal aperiódica de duración finita.

La función $F(w)$ es regular con período unitario y en general es una función compleja de la variable w . La muestra k de la señal se da por la relación inversa.

$$f_{(k)} = \Delta f \sum_n F_{(n)} e^{j2\pi nk/K} = 1/K \sum_n F_{(n)} e^{j2\pi nk/K}$$
(12)

donde $n = 0, 1, 2, \dots, (K - 1)$

Esta transformación es la inversa a la síntesis de la señal muestreando su espectro frecuencial. Las relaciones (11) y (12) son las transformadas de Fourier discretas.

Vale la pena recalcar la dualidad de los dominios de tiempo y frecuencia a través de la transformación de Fourier. El muestreo de una señal análoga genera una repetición periódica de la transformada $F(w)$ de la señal por dualidad, el muestreo de $F(w)$ introduce la repetición periódica de la función de tiempo de modo que:

$$f_{(k)} = f_{(k+K)} = f_{(k+2K)} = \text{etc.}$$
(13)

Esto permite descubrir una diferencia de fondo entre las transformadas de Fourier para señales análogas y numéricas.

La transformada directa o inversa de una función muestreada es una función periódica cuyo período es el inverso del período de muestreo. El olvido de este detalle puede introducir errores graves. La relación (13) indica que si la duración de la señal aperiódica $f_{(k)}$ se limita a K , cada período de la señal muestreada es una réplica exacta de $f_{(t)}$.

Si esta duración es superior a K hay una sobreposición y no podemos extraer exactamente $f_{(t)}$ a partir de las muestras $f_{(k)}$.

Transformaciones rápidas

Desde su descubrimiento en 1965 por Cooley y Tuckey, el algoritmo de la TFR ha revolucionado,

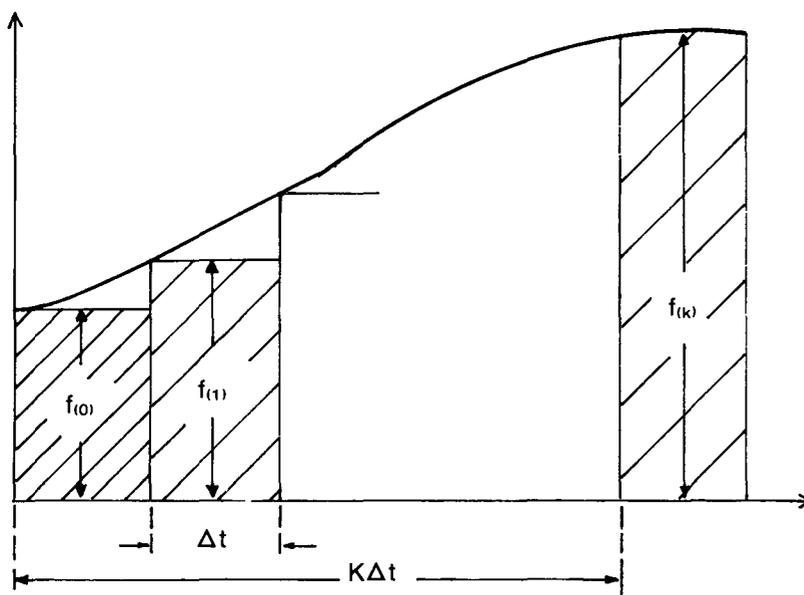


FIGURA 2. Muestreo de una señal aperiódica de duración finita.

debido a su eficacia, el tratamiento numérico de las señales. La Transformada de Fourier Discreta —TFD— puede considerarse como el producto de una matriz llamada de transformación por un vector formado por las muestras de una señal. La eficacia de la TFR proviene de la redundancia de los elementos de la matriz de transformación. Esta redundancia no es cualquiera; ella posee una estructura bien determinada. Luego se puede aprovechar esta estructura para sintetizar otras matrices de transformación que tienen redundancias similares, generalizando en esta forma el algoritmo de TFR. Esta estructura obtenida gracias al producto de Kronecker (producto tensorial) fue utilizado por la primera vez por Sylvester en 1867. Desde entonces el método fue mejorado y generalizado de tal manera que constituye hoy en día una herramienta eficaz en el tratamiento numérico de las señales. Las bases de la TFD fueron establecidas en 1939. La mayoría de los algoritmos de los cálculos actuales ya existían en una época cuando todavía no existían los computadores.

En las teorías modernas las señales son consideradas como vectores en un espacio multidimensional. Se podría formar un vector mediante las muestras de una señal $f_{(k)}$, sobre la cual se puede aplicar una transformación (una matriz). El producto directo entre dos matrices es otra manera de multiplicación matricial: $A \otimes B$. También se llama producto de Kronecker. Si A es del orden $m \times n$ y B es del orden $p \times q$, la matriz $C = A \otimes B$ es una matriz $mp \times nq$.

Consideramos un ejemplo:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \otimes \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{11}b_{12} & a_{12}b_{11} & a_{12}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & a_{12}b_{21} & a_{12}b_{22} \\ a_{21}b_{11} & a_{21}b_{12} & a_{22}b_{11} & a_{22}b_{12} \\ a_{21}b_{21} & a_{21}b_{22} & a_{22}b_{21} & a_{22}b_{22} \end{bmatrix}$$

El teorema de Good es muy útil para redactar algoritmos de cálculo rápido. Dichos algoritmos se deducen de un diagrama de flujo al indicar esquemáticamente las operaciones por efectuar. Para ilustrar la utilización del teorema de Good en la redacción de algoritmos, tomaremos 2 matrices A^1 y A^0 de orden 2×2 . El teorema de Good permite escribir la relación matricial siguiente:

Sea el vector X^2 relacionado con X^0 según la transformación:

$$X^2 = [A^1 \otimes A^0] X^0$$

$$\begin{bmatrix} x_1^2 \\ x_2^2 \\ x_3^2 \\ x_4^2 \end{bmatrix} = \begin{bmatrix} a_{11}^1 a_{11}^0 & a_{11}^1 a_{12}^0 & a_{12}^1 a_{11}^0 & a_{12}^1 a_{12}^0 & x_1^0 \\ a_{11}^1 a_{21}^0 & a_{11}^1 a_{22}^0 & a_{12}^1 a_{21}^0 & a_{12}^1 a_{22}^0 & x_2^0 \\ a_{21}^1 a_{11}^0 & a_{21}^1 a_{12}^0 & a_{22}^1 a_{11}^0 & a_{22}^1 a_{12}^0 & x_3^0 \\ a_{21}^1 a_{21}^0 & a_{21}^1 a_{22}^0 & a_{22}^1 a_{21}^0 & a_{22}^1 a_{22}^0 & x_4^0 \end{bmatrix}$$

El diagrama de flujo de la figura 3 representa gráficamente los cálculos de las distintas etapas de la transformación. La primera columna de nodos a la izquierda representa las muestras de la señal. La segunda columna representa el resultado del primer producto entre la matriz y el vector. Las ramas que unen los nodos llevan el coeficiente multiplicador. Los valores asociados a la última

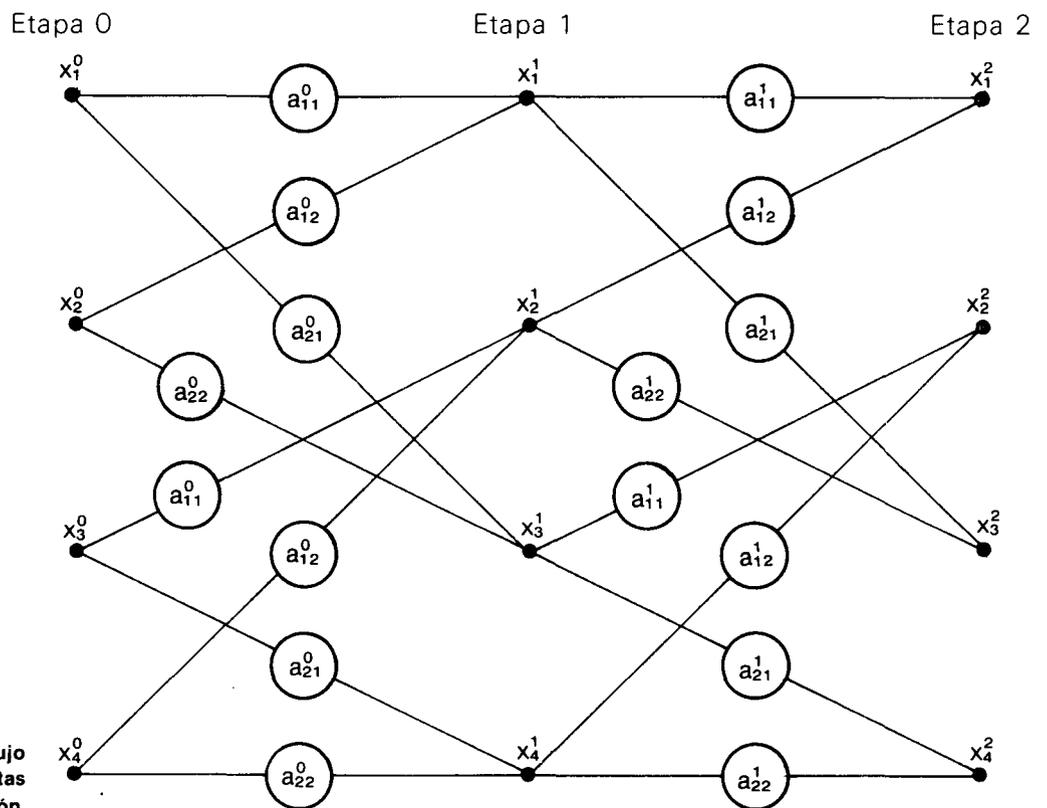


FIGURA 3. Diagrama de flujo de los cálculos de las distintas etapas de transformación.

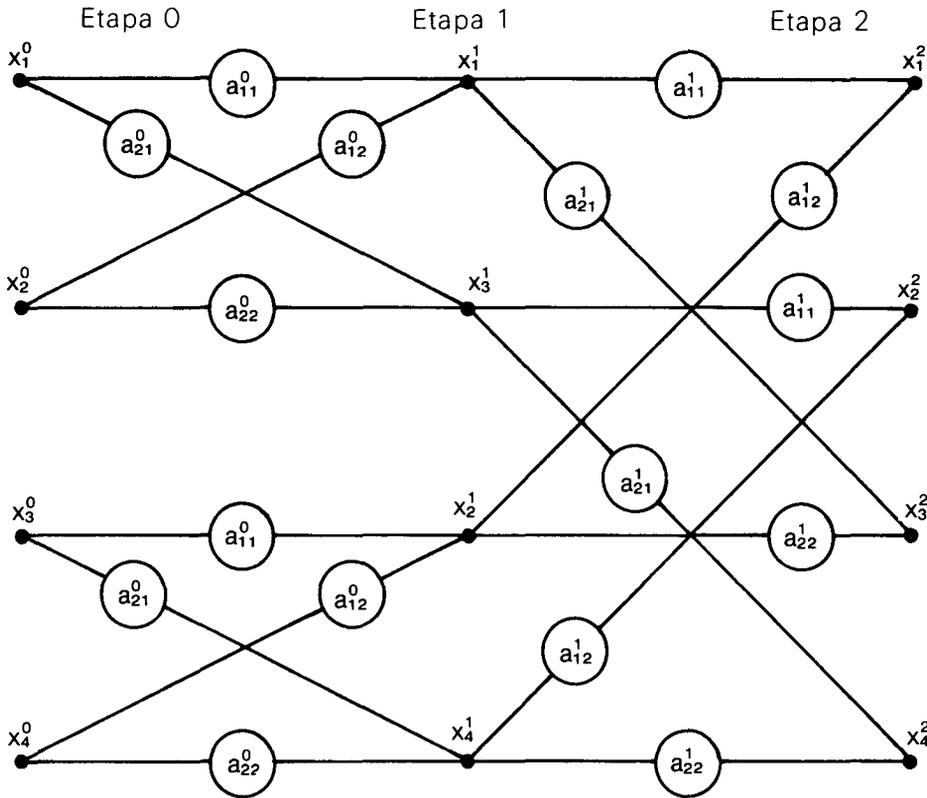


FIGURA 4. Gráfica de flujo modificado.

columna provienen de la segunda multiplicación matricial. El diagrama proviene de la aplicación del teorema de Good. Constatamos que la geometría es idéntica para cada etapa lo que es importante para las implementaciones prácticas. Se repite una misma subestructura de cálculo en orden secuencial de una etapa a otra, cambiando únicamente los coeficientes de multiplicación. La estructura de las ramas muestra que el acceso a los datos y la memorización de los resultados en cada etapa pueden efectuarse secuencialmente. Al efectuar los cálculos indicados por el diagrama, se pueden utilizar dos líneas de registros: una línea para memorizar los resultados; una otra para memorizar los datos usados en el cálculo de la etapa. Al fin de cada etapa el papel de cada línea de registros se invierte, es decir, los resultados de una etapa son los datos de la etapa siguiente.

Lo que es importante en el diagrama de flujo, son las ramas que unen los nodos y los coeficientes multiplicativos. No importa cual sea la disposición de los nodos, los resultados son correctos si se respetan las ramas y los coeficientes entre dos

nodos dados. En consecuencia el gráfico de flujo es modificable con el fin de economizar memorias y de acelerar los cálculos. Por ejemplo, si las filas 2 y 3 se intercambian en la matriz de (14), obtenemos los resultados en el orden $X_1^1, X_3^1, X_2^1, X_4^1$. Al efectuar esta permutación de filas se deben intercambiar las segunda y tercera columna de la matriz en (15). El gráfico de flujo modificado está en la figura 4.

La matriz correspondiente al diagrama de flujo de la figura 4 está dada por:

$$\begin{bmatrix} x_1^2 \\ x_2^2 \\ x_3^2 \\ x_4^2 \end{bmatrix} = \begin{bmatrix} a_{11}^1 & a_{12}^1 & 0 & 0 \\ a_{21}^1 & a_{22}^1 & 0 & 0 \\ 0 & 0 & a_{11}^1 & a_{12}^1 \\ 0 & 0 & a_{21}^1 & a_{22}^1 \end{bmatrix} \begin{bmatrix} a_{11}^0 & a_{12}^0 & 0 & 0 \\ a_{21}^0 & a_{22}^0 & 0 & 0 \\ 0 & 0 & a_{11}^0 & a_{12}^0 \\ 0 & 0 & a_{21}^0 & a_{22}^0 \end{bmatrix} \begin{bmatrix} x_1^0 \\ x_2^0 \\ x_3^0 \\ x_4^0 \end{bmatrix}$$

Desde el punto de vista de la velocidad del algoritmo y de una utilización eficaz de la memoria las consecuencias del citado cambio son importantes. En el algoritmo correspondiente al diagrama de flujo modificado se pueden calcular los resultados de una etapa dos a dos. La figura 5

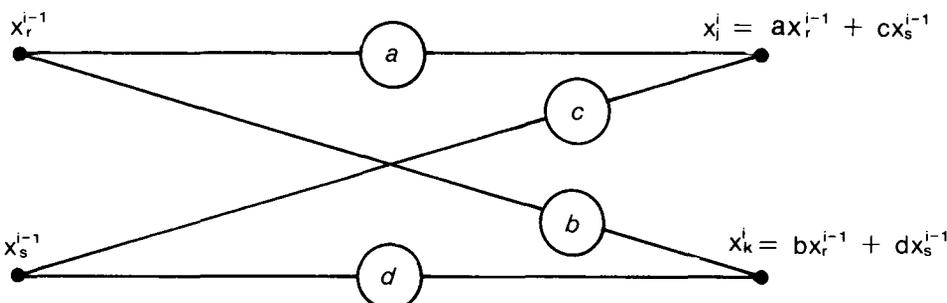


FIGURA 5. Estructura básica conocida como operación mariposa.

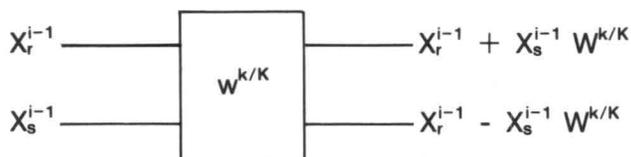


FIGURA 6. Bloque operativo.

muestra la estructura básica conocida como operación mariposa.

Los cálculos que intervienen en la operación mariposa pueden ejecutarse simultáneamente si hay dos unidades aritméticas, o en dos pasos si se memorizan temporalmente los datos en dos registros auxiliares. Los datos de una etapa tomados dos a dos sólo se utilizan una vez para cada operación mariposa. En consecuencia los dos resultados correspondientes pueden ser almacenados en los registros usados para los datos. Un algoritmo que tiene una característica semejante se llama cálculo in situ.

Transformación de Fourier rápida

La TFR juega no solo un papel importante en el tratamiento digital de señales sino también existe ya la posibilidad de usarla para las señales análogas o más aún en las funciones continuas en numerosos dominios científicos. La terminología utilizada en la designación TFR puede conducir a confusión. Aunque se llama transformación, ello no difiere de la TFD; es simplemente un medio (un algoritmo) eficaz, económico y elegante para calcular la TFD.

Estudiando las expresiones (11) y (12) vemos cómo aparece un número complejo representado por su forma exponencial. Corresponde a un factor de magnitud unitaria y de ángulo: $-\frac{n2\pi k}{K}$

Existe una relación de simetría porque:

$$\frac{n2\pi k}{K} = \frac{2\pi n(k + 0.5k)}{K} = \frac{n2\pi k + n\pi}{K}$$

Para componentes pares (n par) las muestras k y (k + 0.5K) son paralelos para componentes impares (n impar) las muestras k y (k + 0.5K) son antiparalelas, lo que implica que en el primer caso las entradas se suman y en el segundo se restan. La implementación de un algoritmo debe tener en cuenta dicha propiedad; con el fin de reducir las operaciones desarrollamos la ecuación (11) poniendo $\omega = e^{-i2\pi}$

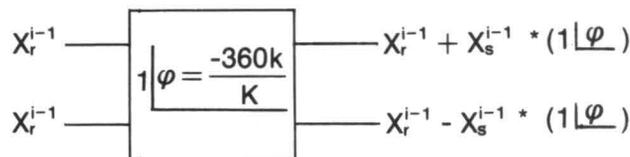


FIGURA 7. Bloque operativo con notación polar.

$$F_{(n)} = \Delta t [f_{(0)}W^0 + f_{(1)}W^{n/k} + f_{(2)}W^{2n/k} + \dots + f_{(k-1)}W^{(k-1)n/k}]$$

Esta suma grande se deja repartir en dos sumas sobre la mitad de las muestras que pueden desarrollarse en paralelo.

$$F_{(n)} = \Delta t [\sum_k f_{(2k)} W^{-2nk/k} + W^{n/k} \sum_k f_{(2k+1)} W^{-2nk/k}] \tag{15}$$

donde: k = 0, 1, 2, 3, ..., 0.5 K - 1

Se ha comprobado que el cálculo se reduce a 0.5K lgK multiplicaciones y a K lgK sumas, lo que significa una importante mejora (lg es el logaritmo en base 2). La separación de las muestras pares e impares permite calcular las sumatorias parciales simultáneamente.

Implementación de una TFR

Se ha experimentado con transformadas discretas con gama amplia de número de muestras: K = 1024, 512, 256, 128, 64, 32. Deseamos sin embargo continuar con muestra DFT de 8 puntos (8 muestras) porque permite resaltar los hechos importantes con medios sencillos. Las operaciones representadas en la figura 5 son repetitivas, puesto que el operador mariposa es básico para la implementación. La figura 6 representa el bloque operativo que relaciona las salidas con las entradas.

Las entradas y salidas son en general numerosas y complejas que son ponderadas por el factor complejo 1/-360k/K como lo visualiza la figura 7. La figura 7 muestra gráficamente las operaciones básicas que debe ejecutar el procesador básico, no describe tan sólo el algoritmo de TFR sino que caracteriza al punto de partida de un método de cálculo.

La figura 8 muestra cómo se vuelve a utilizar una y otra vez para entregar los resultados aplicando las relaciones entre entradas y salidas de cada

TABLA I
Angulo de giro para cada armónica

	f ₍₀₎	f ₍₁₎	f ₍₂₎	f ₍₃₎	f ₍₄₎	f ₍₅₎	f ₍₆₎	f ₍₇₎	Δφ
C ₀	0	0	0	0	0	0	0	0	0
C ₁	0	-45	-90	-135	-180	-225	-270	-315	-45
C ₂	0	-90	-180	-270	-360	-90	-180	-270	-90
C ₃	0	-135	-270	-305		etc.			-135
C ₄	0	-180	-360	-440		etc.			-180
C ₅	0	-225	-450		etc.				-225
C ₆	0	-270	-540		etc.				-315
C ₇	0	-315	-630		etc.				0

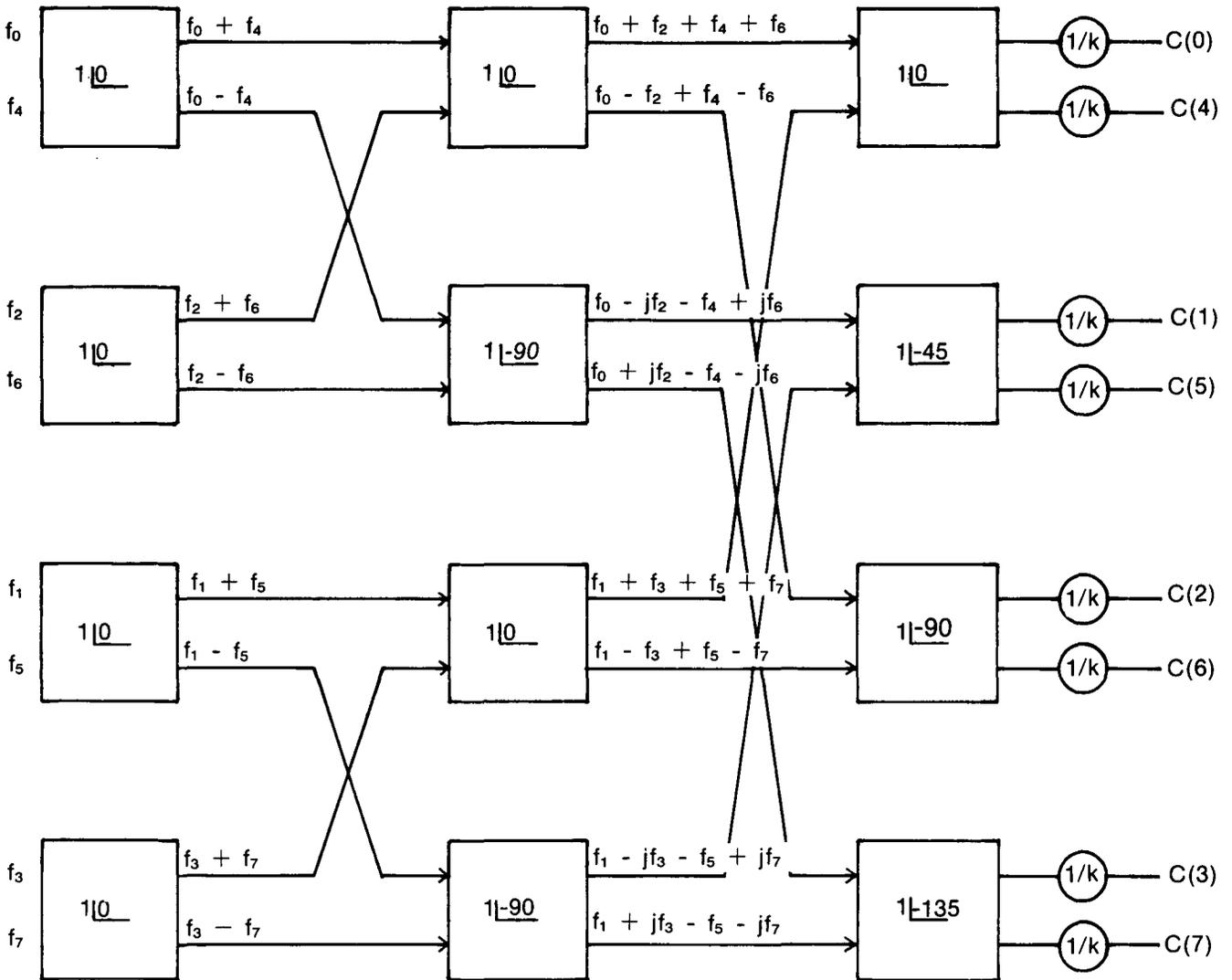


FIGURA 8. Esquema de conexión de bloques operativos.

unidad. También observamos una interesante regularidad en los giros que existe en cada armónica. La tabla 1 muestra el ángulo para cada muestra y armónica.

Para un análisis computarizado de funciones periódicas el concepto $C(nw^0)$ es más familiar al ingeniero electricista ya que en la figura 8 aparecen las salidas espectrales $C(n)$.

La versión discretizada se origina en la forma análoga (11) como:

$$C(nw_0) = 1/k\Delta t \sum f(k \Delta t)e^{-j2\pi n/k} \Delta t$$

$$C(n) = 1/k \sum f(k) e^{-j\pi n /k}$$

En muestra ejemplo el número de muestras $K = 8$ y las salidas son:

$$C_{(0)} = 1/8[f_{(0)} + f_{(1)} + f_{(2)} + f_{(3)} + f_{(4)} + f_{(5)} + f_{(6)} + f_{(7)}]$$

$$C_{(4)} = 1/8[f_{(0)} - f_{(1)} + f_{(2)} - f_{(3)} + f_{(4)} - f_{(5)} + f_{(6)} - f_{(7)}]$$

$$C_{(1)} = 1/8[f_{(0)} + f_{(1)} \underline{-45} - jf_{(2)} + f_{(3)} \underline{-135} - f_{(4)} + f_{(5)} \underline{-225} + jf_{(6)} + f_{(7)} \underline{45}]$$

$$C_{(5)} = 1/8[f_{(0)} + f_{(1)} \underline{135} - jf_{(2)} + f_{(3)} \underline{45} - f_{(4)} + f_{(5)} \underline{-45} + jf_{(6)} + f_{(7)} \underline{225}]$$

$$C_{(2)} = 1/8 [f_{(0)} + jf_{(1)} - f_{(2)} + jf_{(3)} + f_{(4)} - jf_{(5)} - f_{(6)} + jf_{(7)}]$$

$$C_{(6)} = 1/8[f_{(0)} + jf_{(1)} - f_{(2)} - jf_{(3)} + f_{(4)} + jf_{(5)} - f_{(6)} - jf_{(7)}]$$

$$C_{(3)} = 1/8 [f_{(0)} + f_{(1)} \underline{-135} + jf_{(2)} + f_{(3)} \underline{-45} - f_{(4)} + f_{(5)} \underline{45} - jf_{(6)} + f_{(7)} \underline{135}]$$

$$C_{(7)} = 1/8 [f_{(0)} + f_{(1)} \underline{45} + jf_{(2)} + f_{(3)} \underline{135} - f_{(4)} + f_{(5)} \underline{225} - jf_{(6)} + f_{(7)} \underline{-45}]$$

Para comprender un poco más cuales operaciones debe cumplir el operador mariposa sobre los números complejos simplificamos la notación de la figura 7 al designar las entradas X, Y.

La multiplicación y la suma: $X + Y * 1|\varphi$ comprende en realidad dos operaciones:

$$ReX + jI_m X + (ReY + jI_m Y)(\cos\varphi + jsen\varphi) \quad (4,a)$$

Relación de salida para la parte real:

$$ReX + \cos\varphi ReY - \text{sen}\varphi I_m Y \quad (4,b)$$

Relación de salida para la parte imaginaria:

$$I_m Y + \text{sen}\varphi ReY + \cos\varphi I_m Y \quad (4,c)$$

En igual forma para la salida abajo tenemos:

$$ReX - (\cos\varphi ReY - \text{sen}\varphi I_m Y) \quad (4,d)$$

$$I_m X - (\text{sen}\varphi ReY + \cos\varphi I_m Y) \quad (4,c)$$

TABLA II
Secuencia de operaciones del computador

Ciclos de máquina	Operación	Comentario
2	$ReX \rightarrow R_0$	Parte real de X al registro R_0
2	$ImX \rightarrow R_1$	
2	$ReY \rightarrow R_2$	
2	$ImY \rightarrow R_3$	
1	$(R_0) \rightarrow R_4$	Traslado entre registros
1	$(R_1) \rightarrow R_5$	
1	$(R_0) + (R_2) \rightarrow R_0$	$ReX + ReY \rightarrow R_0$
1	$(R_1) + (R_3) \rightarrow R_1$	$ImX + ImY \rightarrow R_1$
1	$-(R_2) \rightarrow R_2$	Cambiar el signo del contenido
1	$(R_2) + (R_4) \rightarrow R_2$	$ReX - ImY \rightarrow R_2$
1	$-(R_3) \rightarrow R_3$	$-ImY \rightarrow R_3$
1	$(R_3) + (R_5) \rightarrow R_3$	$ImX - ImY \rightarrow R_2$

En total 16 ciclos de máquina.

El esfuerzo computacional depende del número de muestras de la función original y del espectro que queremos obtener. Sin el FFT los coeficientes de Fourier deben ser calculados para cada frecuencia por separado. Cada coeficiente requiere la suma de **K** números reales y **K** imaginarios, de acuerdo de (15) cada término es el producto del valor muestreado y del peso trigonométrico **W**. Si deseamos una resolución espectral fina sobre una banda amplia, el trabajo sería enorme porque la rata muestral debe ser al menos el doble de la máxima frecuencia contenida en el espectro de la señal original. El número de muestras es proporcional al inverso de la resolución frecuencial.

Por ejemplo; para determinar el espectro de la potencia de una señal que posee una anchura de banda de 1 Kc con una resolución de dos ciclos, la rata de muestreo debe ser mínimo igual a 2 Kc (teorema de Nyquist). En tal caso, la transformada requiere la formación y suma de millones de datos.

Muchas aplicaciones actuales requieren una entrega rápida de datos cuando se trabaja en tiempo real. Distintas empresas han desarrollado procesadores especiales para este fin. El microprocesador es un chip-CMOS de 16×1 bits en el cual se efectúan operaciones programadas entre los distintos registros.

La tabla II muestra la codificación y el desarrollo secuencial del primer paso del cálculo. Los símbolos y las operaciones se refieren a las ecuaciones (4,b) hasta (4,e) teniendo en cuenta que el factor inicial trigonométrico $\phi = 0$ lo que permite acelerar al menos al iniciarse el cálculo es ahorrar tiempo. Los registros del microprocesador se indican por R_1 hasta r_5 .

El contenido del registro se codifica como (R).

TABLA III
Cálculo de A y B

Ciclos de máquina	Operación	Comentario
2	$ReW \rightarrow R_{30}$	
2	$ImW \rightarrow R_{31}$	
1	$(Re) \rightarrow R_4$	$ReY \rightarrow R_4$
1	$(R_3) \rightarrow R_5$	$ImY \rightarrow R_5$
18	$(R_4) * (R_{30}) \rightarrow R_4$	$ReW * ReY \rightarrow R_4$
18	$(R_5) * (R_{31}) \rightarrow R_5$	$ImW * ImY \rightarrow R_5$
1	$(R_4) - (R_5) \rightarrow R_4$	$ReW * ReY - ImW * ImY$
18	$(R_2) * (R_{31}) \rightarrow R_2$	$ReY * ImW \rightarrow R_2$
18	$(R_3) * (R_{30}) \rightarrow R_3$	$ImY * ReW \rightarrow R_3$
1	$(R_3) + (R_2) \rightarrow R_3$	$(ImY * ReW + ReY * ImW) \rightarrow R_3$
1	$(R_0) \rightarrow R_2$	$ReX \rightarrow R_2$
1	$(R_1) \rightarrow R_5$	$ImX \rightarrow R_1$
1	$(R_0) + (R_4) \rightarrow R_0$	$ReA \rightarrow R_0$
1	$(R_1) + (R_3) \rightarrow R_1$	$ImA \rightarrow R_1$
1	$(R_2) - (R_4) \rightarrow R_2$	$ReB \rightarrow R_2$
1	$-(R_3) \rightarrow R_3$	Complementación
1	$(R_3) + (R_5) \rightarrow R_3$	$ReB \rightarrow R_3$

En total 87 ciclos de máquina.

La tabla III muestra la codificación y la frecuencia temporal del paso general de cálculo. El propósito consiste en calcular ecuaciones de la forma

$$A = X + YW \quad B = X - YW$$

En realidad $W = W^{k/K}$ pero para simplificar la expresión, se hizo abstracción de la potencia de **W**.

CONCLUSION

Se aprecia en el artículo la utilización de la Transformada Rápida de Fourier para el procesamiento de señales. Hemos analizado distintas formas, tanto matemáticas como prácticas para reducir el tiempo de cálculo. Además la figura 8 muestra que los cálculos pueden ser paralelos (simultáneas) porque ningún bloque operatorio de una determinada columna requiere datos de su propia columna.

La existencia de procesadores preformados para tal fin han acelerado apreciablemente el desarrollo computacional.

Campos como el procesamiento de la imagen vía computador, el análisis y síntesis del lenguaje, el radar gozan ahora de un arma potente, nunca antes soñada con los medios análogos.

BIBLIOGRAFIA

- Signalverarbeitung Elektrisches Nachrichtenwesen Band 59 N° 3, 1985.
- Cap Instruction Set Specification ATC Systems. 1984.