Properties of the Multidimensional Generalized Discrete Fourier Transform

PAOLO CORSINI AND GRAZIANO FROSINI

Abstract—**In this work the generalized discrete Fourier transform (GFT), which includes the DFT as a particular case, is considered. Two pairs of fast algorithms for evaluating a multidimensional GFT are given (T-algorithm, F-algorithm, and T'-algorithm, F'-algorithm). It is shown that in the case of the DFT of a vector, the T-algorithm represents a form of the classical FFT algorithm based on a decimation in time, and the F-algorithm represents a form of the classical FFT algorithm based on decimation in frequency. Moreover, it is shown that the T'-algorithm and the T-algorithm involve exactly the same arithmetic operations on the same data. The same property holds for the F'-algorithm and the F-algorithm. The relevance of such algorithms is discussed, and it is shown that the T'-algorithm and the F'-algorithm are particularly advantageous for evaluating the DFT of large sets of data.**

Index Terms—**Fast algorithms, fast Fourier transform, generalized discrete Fourier transform, multidimensional processing, signal processing.**

I. Introduction

LET us consider the problem of evaluating the one-dimensional discrete Fourier transform (DFT) of a vector E having a large number of elements, on the hypothesis that the working memory of the available processor is not sufficient to handle the vector as a whole. Such a situation can arise in several applications [1]–[4], such as Fourier transform spectroscopy or musical sound analysis. In this case it is convenient to fracture E into a matrix F^2 and to separately process single columns and single rows [5]. Such two-dimensional processing of a one-dimensional vector can also be useful in evaluating the DFT of staggered blocks [6], [7]. Unfortunately, the one-dimensional DFT of E is not obtainable simply by evaluating the two-dimensional DFT of F^2, but proper "twiddle factors" must be introduced between column transforms and row transforms [5]. Then the classical two-dimensional processing of the vector E consists of

1) evaluating the one-dimensional DFT of each column of matrix F^2;
2) multiplying, term by term, the matrix obtained after point 1) by a matrix of twiddle factors;
3) evaluating the one-dimensional DFT of each row of the matrix obtained after point 2).

Due to the presence of the twiddle factors, such two-dimensional processing, even if each column DFT and each row DFT is evaluated by means of a fast algorithm (FFT algorithm), is more complex than an FFT algorithm applied directly to the vector E.

In a previous work [8] the generalized discrete Fourier transform (GFT), which includes the DFT as a particular case, has been introduced and two fast algorithms for the GFT computation have been given. Moreover, two procedures have been presented that allow us to obtain the one-dimensional DFT of E by evaluating a proper two-dimensional GFT of F^2. On the hypothesis that the number of elements of E is a power of two, it has also been shown [9] that the two procedures presented in [8], provided that each column GFT and each row GFT are computed by means of a proper fast algorithm, involve exactly the same arithmetic operations on the same data as the classical FFT algorithms based on decimation in time and on decimation in frequency, respectively.

In this work the characteristics of the multidimensional GFT are further investigated, and some general results are derived that include as particular cases the properties previously given in [8] and [9]. More precisely, the obtained results can be schematized in the following points.

1) Two algorithms (T-algorithm and F-algorithm) are given that allow us to obtain an α-dimensional GFT of an α-dimensional array E^α by evaluating a σ-dimensional GFT of a proper σ-dimensional array F^σ ($\sigma > \alpha$). It is shown that in the case of the DFT of a vector E the T-algorithm and the F-algorithm represent, respectively, a form of the FFT algorithm based on decimation in time and a form of the FFT algorithm based on decimation in frequency. It follows that the classical FFT algorithms can be thought of as consisting of the evaluation of proper multidimensional GFT's.

2) Two other algorithms (T'-algorithm and F'-algorithm) for evaluating an α-dimensional GFT of an α-dimensional array E^α are given, which consist of: a) in reordering the elements of E^α in a τ-dimensional array F^τ ($\alpha < \tau < \sigma$) having arbitrary dimensions, and b) in computing by means of the T-algorithm or the F-algorithm successive one-dimensional GFT's along every coordinate of F^τ. It is shown that the T-algorithm and the T'-algorithm, as well

Manuscript received April 7, 1978; revised December 5, 1978 and June 19, 1979. This work was supported by the Convention of Selenia S.p.A. and Consiglio Nazionale delle Ricerche, Pisa, Italy.
P. Corsini is with the Dipartimento Sperimentale di Elettrotecnica ed Elettronica, Facoltà di Ingegneria, Università di Pisa, Pisa, Italy.
G. Frosini is with the Istituto di Elaborazione della Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy.

0018-9340/79/1100-0819$00.75

as the F-algorithm and the F'-algorithm, involve exactly the same arithmetic operations on the same data.

From the previous points it follows that the one-dimensional DFT of a vector E can be obtained by reordering the elements of E in an array F^τ having arbitrary dimensions, and by computing by means of the T-algorithm or the F-algorithm successive one-dimensional GFT's along every coordinate of F^τ. This τ-dimensional processing involves exactly the same arithmetic operations on the same data as an FFT algorithm, but presents the advantage of handling a number of elements at a time that can be optimized with respect to the dimensions of the working area.

II. Generalized Discrete Fourier Transform

In this section the one-dimensional GFT and the multidimensional GFT are precisely defined.

Definition 1: Let

$$E = \{e_t\}$$

where $t = 0, 1, \cdots, T-1$, and

$$G = \{g_z\}$$

where $z = 0, 1, \cdots, T-1$ are two vectors of T complex numbers, and let a and b be two constants. The vector G is said to be the *one-dimensional generalized discrete Fourier transform* (*GFT*) *of the vector E with time parameter a and frequency parameter b* [briefly, the one-dimensional GFT of (E, a, b)] if

$$g_z = \sum_{t=0}^{T-1} e_t W[T]^{(t+a)(z+b)} \tag{1}$$

where $W[T] = \exp(-2\pi\sqrt{-1}/T)$.

Note that, as a particular case, the one-dimensional GFT of E with both parameters equal to zero coincides with the one-dimensional DFT of E.

If relation (1) holds, then the elements of E can be obtained again from the elements of G by means of the inverse relation

$$e_t = \frac{1}{T} \sum_{z=0}^{T-1} g_z W[T]^{-(z+b)(t+a)}. \tag{2}$$

In fact, it results that

$$\begin{aligned}\frac{1}{T} \sum_{z=0}^{T-1} g_z W[T]^{-(z+b)(t+a)} &= \frac{1}{T} \sum_{z=0}^{T-1} \sum_{x=0}^{T-1} e_x W[T]^{(x+a)(z+b)} W[T]^{-(z+b)(t+a)} \\ &= \frac{1}{T} \sum_{x=0}^{T-1} \left(e_x W[T]^{b(x-t)} \sum_{z=0}^{T-1} W[T]^{z(x-t)} \right).\end{aligned}$$

The second summation is equal to zero for $x \neq t$ and is equal to T for $x = t$, so that relation (2) holds.

Definition 2: Let

$$F^\sigma = \{f^\sigma_{n_1, n_2, \cdots, n_\sigma}\}$$

$$H^\sigma = \{h^\sigma_{k_1, k_2, \cdots, k_\sigma}\}$$

$$\Phi^\sigma = \{a_1^\sigma = a_1^\sigma(n_2, n_3, \cdots, n_\sigma), a_2^\sigma = a_2^\sigma(k_1, n_3, \cdots, n_\sigma), \cdots, a_\sigma^\sigma = a_\sigma^\sigma(k_1, k_2, \cdots, k_{\sigma-1})\}$$

$$\Psi^\sigma = \{b_1^\sigma = b_1^\sigma(n_2, n_3, \cdots, n_\sigma), b_2^\sigma = b_2^\sigma(k_1, n_3, \cdots, n_\sigma), \cdots, b_\sigma^\sigma = b_\sigma^\sigma(k_1, k_2, \cdots, k_{\sigma-1})\}$$

(where $n_u, k_u = 0, 1, \cdots, N_u - 1$ for $u = 1, 2, \cdots, \sigma$) are, respectively, two σ-dimensional arrays (having the same dimensions) of $\prod_{u=1}^{\sigma} N_u$ complex numbers, and two vectors of σ functions of $\sigma - 1$ integers. The array H^σ is said to be the *σ-dimensional GFT of the array F^σ with time parameter vector Φ^σ and frequency parameter vector Ψ^σ* [briefly, the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$] if

$$h_{k_1, k_2, \cdots, k_\sigma} = \sum_{n_\sigma=0}^{N_\sigma-1} \left(\cdots \left(\sum_{n_2=0}^{N_2-1} \left(\sum_{n_1=0}^{N_1-1} f^\sigma_{n_1, n_2, \cdots, n_\sigma} W[N_1]^{(n_1+a_1)(k_1+b_1)} \right) \cdot W[N_2]^{(n_2+a_2)(k_2+b_2)} \right) \cdots \right) W[N_\sigma]^{(n_\sigma+a_\sigma)(k_\sigma+b_\sigma)}.$$

Note that the calculation of the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$, i.e., of the array H^σ, can be obtained in σ steps in the ith of which, $i = 1, 2, \cdots, \sigma$, a σ-dimensional array $D[i]^\sigma$ is processed ($D[1]^\sigma = F^\sigma$) and a σ-dimensional array $D[i+1]^\sigma$ is produced in such a way that $H^\sigma = D[\sigma+1]^\sigma$. To be precise, in the ith step, for every value combination of $k_1, \cdots, k_{i-1}, n_{i+1}, \cdots, n_\sigma$, say $k_1^*, \cdots, k_{i-1}^*, n_{i+1}^*, \cdots, n_\sigma^*$, the one-dimensional GFT of the vector

$$\{d[i]^\sigma_{k_1^*, \cdots, k_{i-1}^*, n_i, n_{i+1}^*, \cdots, n_\sigma^*}\}, \qquad n_i = 0, 1, \cdots, N_{i-1}$$

with parameters

$$a_i^\sigma(k_1^*, \cdots, k_{i-1}^*, n_{i+1}^*, \cdots, n_\sigma^*)$$

$$b_i^\sigma(k_1^*, \cdots, k_{i-1}^*, n_{i+1}^*, \cdots, n_\sigma^*)$$

is evaluated, so obtaining the vector

$$\{d[i+1]^\sigma_{k_1^*, \cdots, k_{i-1}^*, k_i, n_{i+1}^*, \cdots, n_\sigma^*}\}, \qquad k_i = 0, 1, \cdots, N_{i-1}$$

where $d[\mu]^\sigma_{k_1, \cdots, k_{\mu-1}, n_\mu, \cdots, n_\sigma}$ is the $(k_1, \cdots, k_{\mu-1}, n_\mu, \cdots, n_\sigma)$th element of $D[\mu]^\sigma$.

As an example, let us consider for $\sigma = 2$ the two matrices

$$F^2 = n_1 \downarrow \begin{array}{c} \xrightarrow{n_2} \\ \left| \begin{array}{cccc} f^2_{0,0} & f^2_{0,1} & \cdots & f^2_{0,N_2-1} \\ f^2_{1,0} & f^2_{1,1} & \cdots & f^2_{1,N_2-1} \\ \vdots & \vdots & & \vdots \\ f^2_{N_1-1,0} & f^2_{N_1-1,1} & \cdots & f^2_{N_1-1,N_2-1} \end{array} \right| \end{array}$$

and

$$H^2 = k_1 \downarrow \begin{array}{c} \xrightarrow{k_2} \\ \left| \begin{array}{cccc} h^2_{0,0} & h^2_{0,1} & \cdots & h^2_{0,N_2-1} \\ h^2_{1,0} & h^2_{1,1} & \cdots & h^2_{1,N_2-1} \\ \vdots & \vdots & & \vdots \\ h^2_{N_1-1,0} & h^2_{N_1-1,1} & \cdots & h^2_{N_1-1,N_2-1} \end{array} \right| \end{array}$$

and the two parameter vectors

$$\Phi^2 = \{a_1^2(n_2), a_2^2(k_1)\}$$
$$\Psi^2 = \{b_1^2(n_2), b_2^2(k_1)\}.$$

Let H^2 be the two-dimensional GFT of (F^2, Φ^2, Ψ^2). Such a matrix can be obtained in the following two steps.

1) Evaluate for every value of n_2, say n_2^*, the one-dimensional GFT of the n_2^*th column of F^2 with parameters $a_1^2(n_2^*)$ and $b_1^2(n_2^*)$, so obtaining an intermediate matrix $D[2]^2 = \{d[2]_{k_1,n_2}^2\}$.

2) Evaluate, for every value of k_1, say k_1^*, the one-dimensional GFT of the k_1^*th row of $D[2]^2$ with parameters $a_2^2(k_1^*)$ and $b_2^2(k_1^*)$, so obtaining the final matrix H^2.

For the sake of clarity, first we will introduce fast algorithms for the evaluation of the GFT of a vector (Sections III and IV), and then we will extend the obtained results to the case of the GFT of a multidimensional array (Section V).

III. T-Algorithm and F-Algorithm for the Computation of the GFT of a Vector

The aim of this section is to show how the one-dimensional GFT of a vector E with given parameters a and b can be obtained by regarding the elements of E as reordered in a proper multidimensional array, and by evaluating a multidimensional GFT of such an array with proper parameter vectors.

Definition 3: Let Δ^σ be the set of integers $\{N_1, N_2, \cdots, N_\sigma\}$. The σ-dimensional array F^σ of $\prod_{u=1}^{\sigma} N_u$ elements is the Δ^σ-*horizontal rearrangement* of the vector E of T elements if $N_1 N_2 \cdots N_\sigma = T$, and

$$f_{n_1,n_2,\cdots,n_\sigma}^\sigma = e_t \qquad \text{for} \qquad t = \sum_{u=1}^{\sigma}\left(\prod_{v=u+1}^{\sigma} N_v\right) n_u.$$

Likewise, the vector G of T elements is the *vertical rearrangement* of the σ-dimensional array H^σ of $\prod_{u=1}^{\sigma} N_u$ elements if $T = N_1 N_2 \cdots N_\sigma$, and

$$g_z = h_{k_1,k_2,\cdots,k_\sigma}^\sigma \qquad \text{for} \qquad z = \sum_{u=1}^{\sigma}\left(\prod_{v=1}^{u-1} N_v\right) k_u.$$

As an example, the $N_1 \times N_2$ matrix F^2 is the $\{N_1, N_2\}$-horizontal rearrangement of the vector E of T elements if $N_1 N_2 = T$ and $f_{n_1,n_2}^2 = e_t$ for $t = N_2 n_1 + n_2$, that is, if

$$F^2 = n_1 \Big\downarrow \overset{\displaystyle n_2 \longrightarrow}{\begin{vmatrix} e_0 & e_1 & \cdots & e_{N_2-1} \\ e_{N_2} & e_{N_2+1} & \cdots & e_{2N_2-1} \\ \vdots & \vdots & \cdots & \vdots \\ e_{(N_1-1)N_2} & e_{(N_1-1)N_2+1} & \cdots & e_{N_1N_2-1} \end{vmatrix}}.$$

Likewise, the vector G of T elements is the vertical rearrangement of the $N_1 \times N_2$ matrix H^2 if $T = N_1 N_2$ and $g_z = h_{k_1,k_2}^2$ for $z = k_1 + N_1 k_2$, that is, if

$$G = \{h_{0,0}^2, h_{1,0}^2, \cdots, h_{N_1-1,0}^2, h_{0,1}^2, h_{1,1}^2, \cdots, h_{N_1-1,1}^2, \cdots, h_{0,N_2-1}^2, h_{1,N_2-1}^2, \cdots, h_{N_1-1,N_2-1}^2\}.$$

Definition 4: The parameter vectors Φ^σ and Ψ^σ are the Δ^σ-*projections of type T* of a and b if

$$a_u^\sigma = \begin{cases} 0 & \text{if } 1 \le u < \sigma \\ a & \text{if } u = \sigma \end{cases}$$
$$b_u^\sigma = \frac{\sum_{x=1}^{u-1}\left(\prod_{v=1}^{x-1} N_v\right) k_x + b}{\prod_{x=1}^{u-1} N_x} \qquad u = 1, 2, \cdots, \sigma. \tag{3}$$

Likewise, the parameter vectors Φ^σ and Ψ^σ are the Δ^σ-*projections of type F* of a and b if

$$a_u^\sigma = \frac{\sum_{x=u+1}^{\sigma}\left(\prod_{v=x+1}^{\sigma} N_v\right) n_x + a}{\prod_{x=u+1}^{\sigma} N_x} \qquad u = 1, 2, \cdots, \sigma.$$
$$b_u^\sigma = \begin{cases} b & \text{if } u = 1 \\ 0 & \text{if } 1 < u \le \sigma \end{cases}$$

As an example, for $\sigma = 2$ the parameter vectors Φ^2 and Ψ^2 are the $\{N_1, N_2\}$-projections of type T of a and b if

$$\Phi^2 = \{0, a\}$$
$$\Psi^2 = \{b, (k_1 + b)/N_1\}.$$

Likewise, the parameter vectors Φ^2 and Ψ^2 are the $\{N_1, N_2\}$-projections of type F of a and b if

$$\Phi^2 = \{(n_2 + a)/N_2, a\}$$
$$\Psi^2 = \{b, 0\}.$$

The following theorem gives two fast algorithms (T-algorithm and F-algorithm) for the evaluation of a one-dimensional GFT of a vector. This theorem will be generalized in Section V to the case of a multidimensional GFT of a multidimensional array, and the proof of the generalized theorem can be found in the Appendix.

Theorem 1: Let us consider the vector E of T elements and the parameters a and b, and let us suppose that the set of integers $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ is such that $T = N_1 N_2 \cdots N_\sigma$. The vector G obtained after the following algorithms (T-algorithm and F-algorithm) is the one-dimensional GFT of (E, a, b).

T-Algorithm on (E, a, b, Δ^σ):

1) Reorder the elements of E in a σ-dimensional array F^σ in such a way that F^σ is the Δ^σ-horizontal rearrangement of E.

2) Evaluate the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$ where Φ^σ and Ψ^σ are the Δ^σ-projections of type T of a and b, so giving a new σ-dimensional array H^σ.

3) Reorder the elements of H^σ in a vector G in such a way that G is the vertical rearrangement of H^σ.

F-Algorithm on (E, a, b, Δ^σ):

1) Reorder the elements of E in a σ-dimensional array F^σ in such a way that F^σ is the Δ^σ-horizontal rearrangement of E.

2) Evaluate the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$ where Φ^σ and Ψ^σ are the Δ^σ-projections of type F of a and b, so giving a new σ-dimensional array H^σ.

3) Reorder the elements of H^σ in a vector G in such a way that G is the vertical rearrangement of H^σ.

It should be noted that in applying the T-algorithm and the F-algorithm, the parameter vectors Φ^σ and Ψ^σ are evaluated once for all and can be utilized for different data. As a particular case, if the time parameter a is equal to zero, then in the T-algorithm all the elements of the resulting parameter vector Φ^σ are identically equal to zero. Likewise, if the frequency parameter b is equal to zero, then in the F-algorithm all the elements of the resulting parameter vector Ψ^σ are identically equal to zero. Moreover, the T-algorithm and the F-algorithm contain all the information for the treatment of indexing and programming.

Note also that the arithmetic operations required by the T-algorithm and by the F-algorithm are those involved in point 2), since the reorderings relative to points 1) and 3) are simply obtained by a proper indexing. Point 2) requires the evaluation of T/N_i one-dimensional GFT's of sets of N_i elements for $i = 1, 2, \cdots, \sigma$. Then, if we denote with $O(N_i)$ the number of arithmetic operations involved in the evaluation of a GFT of a set of N_i elements, it follows that the number of arithmetic operations required by the T-algorithm or by the F-algorithm is $\sum_{i=1}^{\sigma} (T/N_i)O(N_i)$. As a particular case, let us consider the T-algorithm for evaluating the GFT of a vector of $T = 2^\sigma$ elements with time parameter equal to zero. This algorithm requires the evaluation of $(T/2) \log_2 T$ one-dimensional GFT's of two elements with time parameter equal to zero. Since the evaluation of a GFT of two elements x_0 and x_1 with parameters 0 and ξ involves the computation

$$y_0 = x_0 + x_1 W[2]^\xi, \qquad y_1 = x_0 - x_1 W[2]^\xi$$

it follows that in this case the T-algorithm requires $T \log_2 T$ complex additions and $(T/2) \log_2 T$ complex multiplications, i.e., it has the same complexity as an FFT algorithm.

Likewise, let us consider the F-algorithm for evaluating the GFT of a vector $T = 2^\sigma$ elements with frequency parameter equal to zero. This algorithm requires the evaluation of $(T/2) \log_2 T$ one-dimensional GFT's of two elements with frequency parameter equal to zero. Since the evaluation of a GFT of two elements x_0 and x_1 with parameters μ and 0 involves the computation

$$y_0 = x_0 + x_1, \qquad y_1 = (x_0 - x_1)W[2]^\mu$$

it follows that in this case also the F-algorithm has the same complexity as an FFT algorithm.

As an example, let us evaluate the one-dimensional DFT of a vector E having $T = 2^4$ elements by using the T-algorithm on $(E, 0, 0, \{2, 2, 2, 2\})$. This consists of the following.

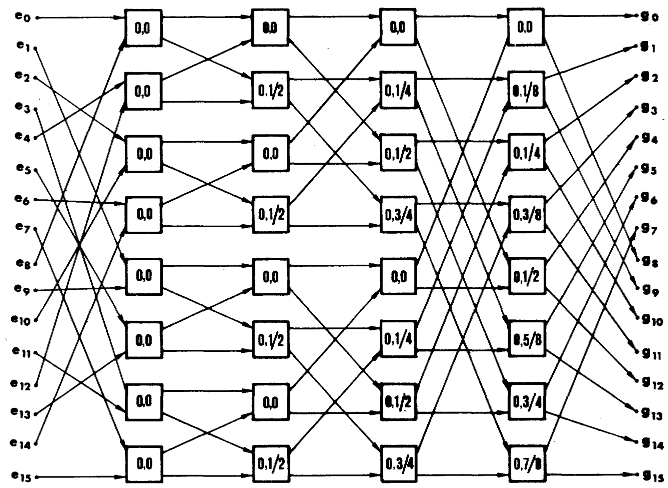

Fig. 1. Flow graph of the T-algorithm on $(E, 0, 0, \{2, 2, 2, 2\})$ where E is a vector of $T = 2^4$ elements.

1) Reordering the elements of E in a four-dimensional array F^4 (having all the dimensions equal to two) by means of the relation

$$f^4_{n_1,n_2,n_3,n_4} = e_t \qquad \text{for } t = 8n_1 + 4n_2 + 2n_3 + n_4.$$

2) Evaluating the four-dimensional GFT of F^4 with time parameter vector $\Phi^4 = \{0, 0, 0, 0\}$ and frequency parameter vector $\Psi^4 = \{0,\ k_1/2,\ (k_1 + 2k_2)/4,\ (k_1 + 2k_2 + 4k_3)/8\}$. This is accomplished in four steps, and in each step eight one-dimensional GFT's of two elements are performed, all having their time parameter equal to zero. In such a way a final four-dimensional array H^4 (having all the dimensions equal to two) is obtained.

3) Reordering the elements of H^4 in a vector G according to the relation

$$g_z = h^4_{k_1,k_2,k_3,k_4} \qquad \text{for } z = k_1 + 2k_2 + 4k_3 + 8k_4.$$

The flow graph of this algorithm is given in Fig. 1, where each block represents the computation of a GFT of two elements, the parameters being written into the block. Each column of blocks represents one step for evaluating the four-dimensional GFT of F^4.

Likewise, let us evaluate the one-dimensional DFT of a vector E of $T = 2^4$ elements by using the F-algorithm on $(E, 0, 0, \{2, 2, 2, 2\})$. This consists of points 1) and 3) relative to the previous example, and of the following point 2).

2) Evaluating the four-dimensional GFT of F^4 with time parameter vector $\Phi^4 = \{(4n_2 + 2n_3 + n_4)/8,\ (2n_3 + n_4)/4,\ n_4/2,\ 0\}$ and frequency parameter vector $\Psi^4 = \{0, 0, 0, 0\}$. This is accomplished in four steps, and in each step eight one-dimensional GFT's of two elements are performed, all having their frequency parameter equal to zero.

The flow graph of this algorithm is given in Fig. 2.

Let us now compare in the case of a DFT the T-algorithm and the F-algorithm with the classical FFT algorithms. By examining the flow graph in Fig. 1, and by taking into account the way a GFT of two elements with time parameter equal to zero is evaluated, it is easy to verify that for the given example the T-algorithm represents a form of the classical FFT algorithm based on decimation in time [10].

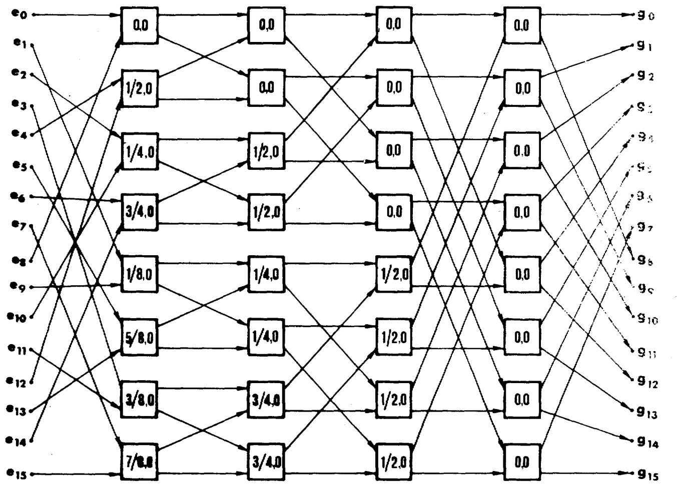

Fig. 2. Flow graph of the F-algorithm on $(E, 0, 0, \{2, 2, 2, 2\})$ where E is a vector of $T = 2^4$ elements.

Likewise, it is easy to verify that for the example given in Fig. 2 the F-algorithm represents a form of the classical FFT algorithm based on decimation in frequency [11]. This fact is true, in general, as stated in the following theorem. The theorem proof can be found in the Appendix.

Theorem 2: Let us consider the vector E of T-elements, and let us suppose that the set of integers $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ is such that $T = N_1 N_2 \cdots N_\sigma$. The T-algorithm on $(E, 0, 0, \Delta^\sigma)$ represents a form of the classical FFT algorithm in mixed radix based on decimation in time. Likewise, the F-algorithm on $(E, 0, 0, \Delta^\sigma)$ represents a form of the classical FFT algorithm in mixed radix based on decimation in frequency.

From the previous theorem it results that the classical FFT algorithms can also be viewed as consisting of the evaluation of proper multidimensional GFT's.

IV. T'-Algorithm and F'-Algorithm

In this section two other algorithms for computing the one-dimensional GFT of a vector are presented, which are useful for computing the DFT of large sets of data.

From now on, besides the notations previously introduced, the following entities are considered. That is,

$$F^\tau = \{f^\tau_{p_1, p_2, \cdots, p_\tau}\}$$

$$H^\tau = \{h^\tau_{q_1, q_2, \cdots, q_\tau}\}$$

$$\Phi^\tau = \{a_1^\tau = a_1^\tau(p_2, p_3, \cdots, p_\tau),\ a_2^\tau = a_2^\tau(q_1, p_3, \cdots, p_\tau), \cdots,\ a_\tau^\tau = a_\tau^\tau(q_1, q_2, \cdots, q_{\tau-1})\}$$

$$\Psi^\tau = \{b_1^\tau = b_1^\tau(p_2, p_3, \cdots, p_\tau),\ b_2^\tau = b_2^\tau(q_1, p_3, \cdots, p_\tau), \cdots,\ b_\tau^\tau = b_\tau^\tau(q_1, q_2, \cdots, q_{\tau-1})\}$$

(where $p_r, q_r = 0, 1, \cdots, P_r - 1$ for $r = 1, 2, \cdots, \tau$) will denote, respectively, two τ-dimensional arrays (having the same dimensions) of $\prod_{r=1}^{\tau} P_r$ complex numbers and two parameter vectors of τ functions of $\tau - 1$ integers.

Let us now consider the vector E of T elements and the parameters a and b, and let us suppose that the two sets of integers $\Delta^\tau = \{P_1, P_2, \cdots, P_\tau\}$ and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ are such that $T = P_1 P_2 \cdots P_\tau$ and $P_r = N_{\delta_r+1} N_{\delta_r+2} \cdots N_{\delta_{r+1}}$, $r = 1, 2, \cdots, \tau$ where $\delta_1 = 0$, $\delta_{r+1} > \delta_r$ and $\delta_{\tau+1} = \sigma$. By taking into account Definition 2, from the T-algorithm and from the F-algorithm it follows that the one-dimensional GFT of (E, a, b) can also be obtained according to the following two algorithms (T'-algorithm and F'-algorithm).

T'-Algorithm on $(E, a, b, \Delta^\tau, \Delta^\sigma)$:

1) Reorder the elements of E in a τ-dimensional array F^τ in such a way that F^τ is the Δ^τ-horizontal rearrangement of E.

2) Evaluate the τ-dimensional GFT of $(F^\tau, \Phi^\tau, \Psi^\tau)$, say H^τ, where Φ^τ and Ψ^τ are the Δ^τ-projections of type T of a and b, according to a τ-step procedure (see note of Definition 2), in the ith step of which, $i = 1, 2, \cdots, \tau$, a τ-dimensional array $D[i]^\tau$ is processed ($D[1]^\tau = F^\tau$) and a τ-dimensional array $D[i+1]^\tau$ is produced in such a way that $H^\tau = D[\tau + 1]^\tau$. To be precise, in the ith step, for every value combination of $q_1, \cdots, q_{i-1}, p_{i+1}, \cdots, p_\tau$, say $q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*$, the one-dimensional GFT of the vector

$$\{d[i]^\tau_{q_1^*, \cdots, q_{i-1}^*, p_i, p_{i+1}^*, \cdots, p_\tau^*}\}, \quad p_i = 0, 1, \cdots, P_{i-1}$$

with parameters

$$a_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$
$$b_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$

is evaluated by means of the T-algorithm on

$$(\{d[i]^\tau_{q_1^*, \cdots, q_{i-1}^*, p_i, p_{i+1}^*, \cdots, p_\tau^*}\},\ a_i^\tau,\ b_i^\tau,\ \{N_{\delta_i+1}, N_{\delta_i+2}, \cdots, N_{\delta_{i+1}}\})$$

so obtaining the vector

$$\{d[i+1]^\tau_{q_1^*, \cdots, q_{i-1}^*, q_i, p_{i+1}^*, \cdots, p_\tau^*}\}, \quad q_i = 0, 1, \cdots, P_{i-1}$$

where $d[\mu]^\tau_{q_1, \cdots, q_{\mu-1}, p_\mu, \cdots, p_\tau}$ is the $(q_1, \cdots, q_{\mu-1}, p_\mu, \cdots, p_\tau)$th element of $D[\mu]^\tau$.

3) Reorder the elements of the τ-dimensional array H^τ in a vector G in such a way that G is the vertical rearrangement of H^τ.

F'-Algorithm on $(E, a, b, \Delta^\tau, \Delta^\sigma)$:

1) Reorder the elements of E in a τ-dimensional array F^τ in such a way that F^τ is the Δ^τ-horizontal rearrangement of E.

2) Evaluate the τ-dimensional GFT of $(F^\tau, \Phi^\tau, \Psi^\tau)$, say H^τ, where Φ^τ and Ψ^τ are the Δ^τ-projections of type F of a and b, according to a τ-step procedure (see note of Definition 2), in the ith step of which, $i = 1, 2, \cdots, \tau$, a τ-dimensional array $D[i]^\tau$ is processed ($D[1]^\tau = F^\tau$) and a τ-dimensional array $D[i+1]^\tau$ is produced in such a way that $H^\tau = D[\tau+1]^\tau$. To be precise, in the ith step, for every value combination of $q_1, \cdots, q_{i-1}, p_{i+1}, \cdots, p_\tau$, say $q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*$, the one-dimensional GFT of the vector

$$\{d[i]^\tau_{q_1^*, \cdots, q_{i-1}^*, p_i, p_{i+1}^*, \cdots, p_\tau^*}\}, \quad p_i = 0, 1, \cdots, P_{i-1}$$

with parameters

$$a_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$
$$b_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$

is evaluated by means of the F-algorithm on

$$(\{d[i]^\tau_{q_1^*, \cdots, q_{i-1}^*, p_i, p_{i+1}^*, \cdots, p_\tau^*}\}, a_i^\tau, b_i^\tau, \{N_{\delta_i+1}, N_{\delta_i+2}, \cdots, N_{\delta_{i+1}}\})$$

so obtaining the vector

$$\{d[i+1]^\tau_{q_1^*, \cdots, q_{i-1}^*, q_i, p_{i+1}^*, \cdots, p_\tau^*}\}, \quad q_i = 0, 1, \cdots, P_{i-1}$$

where $d[\mu]^\tau_{q_1, \cdots, q_{\mu-1}, p_\mu, \cdots, p_\tau}$ is the $(q_1, \cdots, q_{\mu-1}, p_\mu, \cdots, p_\tau)$th element of $D[\mu]^\tau$.

3) Reorder the elements of the τ-dimensional array H^τ in a vector G in such a way that G is the vertical rearrangement of H^τ.

As a particular case, let us consider the problem of evaluating the one-dimensional DFT of a vector E [i.e., the one-dimensional GFT of $(E, 0, 0)$] where E is constituted by $T = P_1 P_2$ elements and $P_1 = N_1 N_2 \cdots N_{\delta_2}$, $P_2 = N_{\delta_2+1} N_{\delta_2+2} \cdots N_\sigma$. Such a DFT can be evaluated by using the T'-algorithm, i.e., by the following.

1) Reordering the elements of E in a matrix F^2 according to the relation $f^2_{p_1,p_2} = e_t$ for $t = P_2 p_1 + p_2$.

2) Evaluating the two-dimensional GFT of F^2 with time parameter vector $\Phi^2 = \{0, 0\}$ and frequency parameter vector $\Psi^2 = \{0, q_1/P_1\}$ according to the following two-step procedure.

a) Calculate the one-dimensional DFT of the p_2th column of F^2, $p_2 = 0, 1, \cdots, P_2 - 1$, by means of the T-algorithm on $(\{p_2$th column of $F^2\}, 0, 0, \{N_1, N_2, \cdots, N_{\delta_2}\})$, so giving an intermediate matrix $D[2]^2$.

b) Calculate the one-dimensional GFT of the q_1th row of $D[2]^2$ with parameters 0 and q_1/P_1, $q_1 = 0, 1, \cdots, P_1 - 1$, by means of the T-algorithm on $(\{q_1$th row of $D[2]^2\}, 0, q_1/P_1, \{N_{\delta_2+1}, N_{\delta_2+2}, \cdots, N_\sigma\})$, so giving a matrix H^2.

3) Reordering the elements of H^2 in the vector G according to the relation

$$g_z = h^2_{q_1,q_2} \qquad \text{for } z = q_1 + P_1 q_2$$

so obtaining the DFT of E.

Likewise, under the previous hypotheses, the one-dimensional DFT of a vector E can be evaluated by using the F'-algorithm that consists of points 1) and 3) relative to the T'-algorithm, and of the following point 2).

2) Evaluating the two-dimensional GFT of F^2 with time parameter vector $\Phi^2 = \{p_2/P_2, 0\}$ and frequency parameter vector $\Psi^2 = \{0, 0\}$, according to the following two-step procedure.

a) Calculate the one-dimensional GFT of the p_2th column of F^2 with parameters p_2/P_2 and 0, $p_2 = 0, 1, \cdots, P_2 - 1$, by means of the F-algorithm on $(\{p_2$th column of $F^2\}, p_2/P_2, 0, \{N_1, N_2, \cdots, N_{\delta_2}\})$, so giving an intermediate matrix $D[2]^2$.

b) Calculate the one-dimensional DFT of the q_1th row of $D[2]^2$, $q_1 = 0, 1, \cdots, P_1 - 1$, by means of the F-algorithm on $(\{q_1$th row of $D[2]^2\}, 0, 0, \{N_{\delta_2+1}, N_{\delta_2+2}, \cdots, N_\sigma\})$, so giving a matrix H^2.

As an example, let us evaluate the DFT of a vector E of 16 elements by means of the T'-algorithm on $(E, 0, 0, \{4, 4\}, \{2, 2, 2, 2\})$. This consists of: 1) reordering the elements of E in a 4×4 matrix F^2; 2a) evaluating the DFT of each column of F^2, so giving another 4×4 matrix $D[2]^2$, 2b) evaluating a GFT of each row of $D[2]^2$ with proper parameters, so giving another 4×4 matrix H^2; and 3) reordering the elements of H^2 in the final vector G. Each column transform and each row transform is performed by means of the T-algorithm on sets of 2^2 elements, that is, by rearranging each column and each row in a 2×2 matrix. The flow graph of the T'-algorithm for this example is shown in Fig. 3.

Let us consider in this figure the main blocks on the left. The p_2th block from the top to the bottom ($p_2 = 0, 1, 2, 3$) contains the flow graph of the T-algorithm on $(\{p_2$th column of $F^2\}, 0, 0, \{2, 2\})$, used for computing the DFT of the pertinent column of F^2. Let us consider the main blocks on the right. The q_1th block from the top to the bottom ($q_1 = 0, 1, 2, 3$) contains the flow graph of the T-algorithm on $(\{q_1$th row of $D[2]^2\}, 0, q_1/4, \{2, 2\})$, used for computing the GFT of the pertinent row of $D[2]^2$ with parameters 0 and $q_1/4$.

The flow graph of the F'-algorithm on $(E, 0, 0, \{4, 4\}, \{2, 2, 2, 2\})$ is given in Fig. 4.

The following theorem establishes an equivalence relation between the T-algorithm and the T'-algorithm and between the F-algorithm and the F'-algorithm. Such a theorem will be generalized in the next section to the case of a multidimensional GFT of a multidimensional array, and the proof of the generalized theorem can be found in the Appendix.

Theorem 3: Let us consider the vector E of T elements and the parameters a and b, and let us suppose that the two sets of integers $\Delta^\tau = \{P_1, P_2, \cdots, P_\tau\}$ and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ are such that $T = P_1 P_2 \cdots P_\tau$ and $P_r = N_{\delta_r+1} N_{\delta_r+2} \cdots N_{\delta_{r+1}}$, $r = 1, 2, \cdots, \tau$, where $\delta_1 = 0$, $\delta_{r+1} > \delta_r$, and $\delta_{\tau+1} = \sigma$. The T-algorithm on (E, a, b, Δ^σ) and the T'-algorithm on $(E, a, b, \Delta^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data. Likewise, the F-algorithm on (E, a, b, Δ^σ) and the F'-algorithm on $(E, a, b, \Delta^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data.

From Theorems 2 and 3 it follows that the previous algorithms for evaluating the DFT of E involve exactly the same arithmetic operations on the same data as the FFT algorithms based on decimation in time and on decimation in frequency, respectively.

Let us now illustrate the differences between the T-algorithm and the T'-algorithm. Since the two algorithms involve exactly the same arithmetic operations on the same data, such algorithms differ only in the order in which the operations are performed. As a consequence of this fact, when the dimensions of the working area are not sufficient to handle the vector as a whole, the T'-algorithm involves fewer transfers from storage to working area than the T-algorithm. In order to explain this property, consider the case of a DFT of a vector E of T elements, and suppose that $T = N^\sigma$. Moreover, suppose that the working area is sufficient to handle P elements at most, where $P = N^\delta$ and $T = P^\tau$. By applying the T-algorithm, we must evaluate a

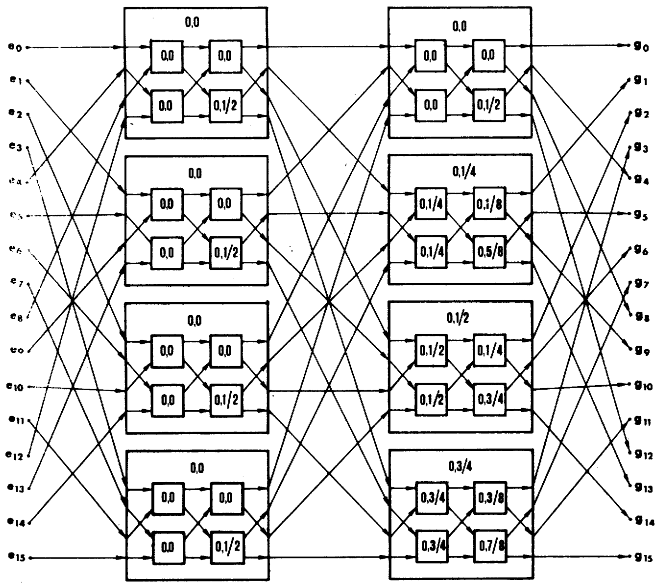

Fig. 3. Flow graph of the T'-algorithm on $(E, 0, 0, \{4, 4\}, \{2, 2, 2, 2\})$ where E is a vector of 16 elements.

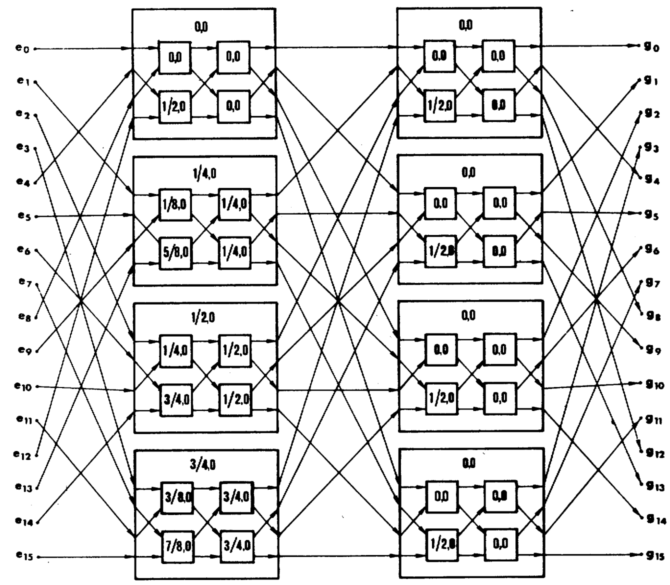

Fig. 4. Flow graph of the F'-algorithm on $(E, 0, 0, \{4, 4\}, \{2, 2, 2, 2\})$ where E is a vector of 16 elements.

σ-dimensional GFT of a σ-dimensional array having all the dimensions equal to N. That is, we must perform σ steps, and at every step we must evaluate T/N one-dimensional GFT's of sets of N elements. One transfer from storage to working area is needed for loading data involved in the computation of P/N one-dimensional GFT, so that T/P transfers are required at every step, for a total of $\sigma T/P$ transfers. By applying the T'-algorithm, we must evaluate a τ-dimensional GFT of a τ-dimensional array having all the dimensions equal to P. That is, we must perform τ steps, and at every step we must evaluate T/P one-dimensional GFT's of sets of P elements. Then T/P transfers from storage to working area are required at every step, for a total of $\tau T/P$ transfers. Since usually $\tau \ll \sigma$, it follows that in the T'-algorithm the number of transfers is reduced. The previous reasoning can be verified by inspecting the flow graph of the T-algorithm in Fig. 1 and the flow graph of the T'-algorithm in Fig. 3. Similar considerations hold for the F-algorithm and the F'-algorithm.

In the literature some algorithms have been presented for evaluating the one-dimensional DFT of large sets of data [1]–[3]. Such algorithms require the same number of transfers from storage to working area as the T'-algorithm or the F'-algorithm, but are more complex. In fact, let us consider as an example the method presented in [2] for evaluating the one-dimensional DFT of a vector of $T = 2^\sigma$ elements, on the hypothesis that the working area is sufficient to handle $P = 2^{\sigma/2}$ elements at most. Such a method involves the rearrangement of E in a $P \times P$ matrix F^2 and the three following computational steps:

1) evaluate the DFT's along the first coordinate of F^2,

2) multiply term by term the matrix obtained after point 1) by a $P \times P$ matrix of twiddle factors,

3) evaluate the DFT's along the second coordinate of the matrix obtained after point 2).

Both points 1) and 3) involve the evaluation of P DFT's of P elements so that the previous method requires $T \log_2 T$ complex additions and $(T/2) \log_2 T + T$ complex multiplications, that is, T complex multiplications more than the T'-algorithm and the F'-algorithms.

V. Fast Algorithms for the GFT Computation of a Multidimensional Array

In this section the obtained results are generalized to the case of a multidimensional GFT of a multidimensional array. From now on, besides the notations previously introduced, the following entities are considered. That is,

$$E^\alpha = \{e^\alpha_{t_1, t_2, \cdots, t_\alpha}\}$$

$$G^\alpha = \{g^\alpha_{z_1, z_2, \cdots, z_\alpha}\}$$

$$\Phi^\alpha = \{a^\alpha_1 = a^\alpha_1(t_2, t_3, \cdots, t_\alpha), a^\alpha_2 = a^\alpha_2(z_1, t_3, \cdots, t_\alpha), \cdots, a^\alpha_\alpha = a^\alpha_\alpha(z_1, z_2, \cdots, z_{\alpha-1})\}$$

$$\Psi^\alpha = \{b^\alpha_1 = b^\alpha_1(t_2, t_3, \cdots, t_\alpha), b^\alpha_2 = b^\alpha_2(z_1, t_3, \cdots, t_\alpha), \cdots, b^\alpha_\alpha = b^\alpha_\alpha(z_1, z_2, \cdots, z_{\alpha-1})\}$$

(where $t_s, z_s = 0, 1, \cdots, T_s - 1$ for $s = 1, 2, \cdots, \alpha$) will denote, respectively, two α-dimensional arrays (having the same dimensions) of $\prod_{s=1}^{\alpha} T_s$ complex numbers and two parameter vectors of α functions of $\alpha - 1$ integers.

Definition 5: Let us consider the two sets of integers $\Delta^\alpha = \{T_1, T_2, \cdots, T_\alpha\}$ and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$. The σ-dimensional array F^σ of $\prod_{u=1}^{\sigma} N_\sigma$ elements is the *Δ^σ-horizontal rearrangment* of the α-dimensional array F^α of $\prod_{s=1}^{\alpha} T_s$ elements if: a) $N_{\gamma_s+1} N_{\gamma_s+2} \cdots N_{\gamma_{s+1}} = T_s, s = 1, 2,$

$\cdots, \alpha$, with $\gamma_1 = 0$, $\gamma_{s+1} > \gamma_s$ and $\gamma_{\alpha+1} = \sigma$, and b)

$$f^{\sigma}_{n_1,n_2,\cdots,n_\sigma} = e^{\alpha}_{t_1,t_2,\cdots,t_\alpha} \tag{4a}$$

for

$$t_s = \sum_{u=\gamma_s+1}^{\gamma_{s+1}} \left(\prod_{v=u+1}^{\gamma_{s+1}} N_v \right) n_u, \qquad s = 1, 2, \cdots, \alpha. \tag{4b}$$

Likewise, the α-dimensional array G^α of $\prod_{s=1}^{\alpha} T_s$ elements is the Δ^α-*vertical rearrangement* of the σ-dimensional array H^σ of $\prod_{u=1}^{\sigma} N_u$ elements if: a) $T_s = N_{\gamma_s+1} N_{\gamma_s+2} \cdots N_{\gamma_{s+1}}$, $s = 1, 2, \cdots, \alpha$, with $\gamma_1 = 0$, $\gamma_{s+1} > \gamma_s$, and $\gamma_{\alpha+1} = \sigma$, and b)

$$g^{\alpha}_{z_1,z_2,\cdots,z_\alpha} = h^{\sigma}_{k_1,k_2,\cdots,k_\sigma} \tag{5a}$$

for

$$z_s = \sum_{u=\gamma_s+1}^{\gamma_{s+1}} \left(\prod_{v=\gamma_s+1}^{u-1} N_v \right) k_u, \qquad s = 1, 2, \cdots, \alpha. \tag{5b}$$

Definition 6: The parameter vectors Φ^σ and Ψ^σ are the Δ^σ-*projections of type T* of the parameter vectors Φ^α and Ψ^α if: a) $N_{\gamma_s+1} N_{\gamma_s+2} \cdots N_{\gamma_{s+1}} = T_s$, $s = 1, 2, \cdots, \alpha$, with $\gamma_1 = 0$, $\gamma_{s+1} > \gamma_s$, and $\gamma_{\alpha+1} = \sigma$, and b) under the transformation of coordinates defined by relations (4b) and (5b) it is

$$a^{\sigma}_{\gamma_s+u} = \begin{cases} 0 & \text{if } 1 \le u < \gamma_{s+1} - \gamma_s \\ a^{\alpha}_s & \text{if } u = \gamma_{s+1} - \gamma_s \end{cases}$$
$$b^{\sigma}_{\gamma_s+u} = \frac{\sum_{x=\gamma_s+1}^{\gamma_s+u-1} \left(\prod_{v=\gamma_s+1}^{x-1} N_v \right) k_x + b^{\alpha}_s}{\prod_{x=\gamma_s+1}^{\gamma_s+u-1} N_x} \tag{6}$$

for $s = 1, 2, \cdots, \alpha$, $u = 1, 2, \cdots, \gamma_{s+1} - \gamma_s$.

Likewise, the parameter vectors Φ^σ and Ψ^σ are the Δ^σ-*projections of type F* of the parameter vectors Φ^α and Ψ^α if: a) $N_{\gamma_s+1} N_{\gamma_s+2} \cdots N_{\gamma_{s+1}} = T_s$, $s = 1, 2, \cdots, \alpha$, with $\gamma_1 = 0$, $\gamma_{s+1} > \gamma_s$, and $\gamma_{\alpha+1} = \sigma$, and b) under the transformation of coordinates defined by relations (4b) and (5b) it is

$$a^{\sigma}_{\gamma_s+u} = \frac{\sum_{x=\gamma_s+u+1}^{\gamma_{s+1}} \left(\prod_{v=x+1}^{\gamma_{s+1}} N_v \right) n_x + a^{\alpha}_s}{\prod_{x=\gamma_s+u+1}^{\gamma_{s+1}} N_x}$$
$$b^{\sigma}_{\gamma_s+u} = \begin{cases} b^2_s & \text{if } u = 1 \\ 0 & \text{if } 1 < u \le \gamma_{s+1} - \gamma_s \end{cases}$$

for $s = 1, 2, \cdots, \alpha$, $u = 1, 2, \cdots, \gamma_{s+1} - \gamma_s$.

Theorem 4: Let us consider the α-dimensional array E^α of $\prod_{s=1}^{\alpha} T_s$ elements and the parameter vectors Φ^α and Ψ^α, and let us suppose that the two sets of integers $\Delta^\alpha = \{T_1, T_2, \cdots, T_\alpha\}$ and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ are such that $T_s = N_{\gamma_s+1} N_{\gamma_s+2} \cdots N_{\gamma_s+1}$, $s = 1, 2, \cdots, \alpha$, where $\gamma_1 = 0$ $\gamma_{s+1} > \gamma_s$ and $\gamma_{\alpha+1} = \sigma$. The α-dimensional array G^α obtained after the following algorithms (T-algorithm and F-algorithm) is the α-dimensional GFT of $(E^\alpha, \Phi^\alpha, \Psi^\alpha)$.

T-Algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$*:*

1) Reorder the elements of E^α in a σ-dimensional array F^σ in such a way that F^σ is the Δ^σ-horizontal rearrangment of E^α.

2) Evaluate the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$ where Φ^σ and Ψ^σ are the Δ^σ-projections of type T of Φ^α and Ψ^α, so giving a new σ-dimensional array H^σ.

3) Reorder the elements of H^σ in an α-dimensional array G^α in such a way that G^α is the Δ^α-vertical rearrangment of H^σ.

F-Algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$*:*

This algorithm differs from the T-algorithm previously described only for the fact that in point 2) the parameter vectors Φ^σ and Ψ^σ are the Δ^σ-projections of type F of Φ^α and Ψ^α.

Still referring to the α-dimensional array E^α of $\prod_{s=1}^{\alpha} T_s$ elements and to the parameter vectors Φ^α and Ψ^α, let us now suppose that the three sets of integers $\Delta^\alpha = \{T_1, T_2, \cdots, T_\alpha\}$, $\Delta^\tau = \{P_1, P_2, \cdots, P_\tau\}$ and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ are such that: a) $T_s = P_{v_s+1} P_{v_s+2} \cdots P_{v_s+1}$, $s = 1, 2, \cdots, \alpha$, where $v_1 = 0$, $v_{s+1} > v_s$, and $v_{\alpha+1} = \tau$, and b) $P_r = N_{\delta_r+1} N_{\delta_r+2} \cdots N_{\delta_{r+1}}$, $r = 1, 2, \cdots, \tau$, where $\delta_1 = 0$, $\delta_{r+1} > \delta_r$ and $\delta_{\tau+1} = \sigma$. By taking into account Definition 2, from the T-algorithm and from the F-algorithm it follows that the α-dimensional GFT of $(E^\alpha, \Phi^\alpha, \Psi^\alpha)$ can also be obtained according to the following two algorithms (T'-algorithm and F'-algorithm).

T′-Algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$*:*

1) Reorder the elements of E^α in a τ-dimensional array F^τ in such a way that F^τ is the Δ^τ-horizontal rearrangement of E^α.

2) Evaluate the τ-dimensional GFT of $(F^\tau, \Phi^\tau, \Psi^\tau)$, say H^τ, where Φ^τ and Ψ^τ are the Δ^τ-projections of type T of Φ^α and Ψ^α, according to a τ-step procedure (see note of Definition 2), in the ith step of which, $i = 1, 2, \cdots, \tau$, a τ-dimensional array $D[i]^\tau$ is processed ($D[1]^\tau = F^\tau$) and a τ-dimensional array $D[i+1]^\tau$ is produced in such a way that $H^\tau = D[\tau+1]^\tau$. To be precise, in the ith step, for every value combination of $q_1, \cdots, q_{i-1}, p_{i+1}, \cdots, p_\tau$, say $q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*$, the one-dimensional GFT of the vector

$$\{d[i]^\tau_{q_1^*,\cdots,q_{i-1}^*,p_i,p_{i+1}^*,\cdots,p_\tau^*}\}, \quad p_i = 0, 1, \cdots, P_{i-1}$$

with parameters

$$a_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$
$$b_i^\tau(q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)$$

is evaluated by means of the T-algorithm on

$$(\{d[i]^\tau_{q_1^*,\cdots,q_{i-1}^*,p_i,p_{i+1}^*,\cdots,p_\tau^*}\}, a_i^\tau, b_i^\tau, \{N_{\delta_i+1}, N_{\delta_i+2}, \cdots, N_{\delta_{i+1}}\})$$

so obtaining the vector

$$\{d[i+1]^\tau_{q_1^*,\cdots,q_{i-1}^*,q_i,p_{i+1}^*,\cdots,p_\tau^*}\}, \quad q_i = 0, 1, \cdots, P_{i-1}$$

where $d[\mu]^\tau_{q_1,\cdots,q_{\mu-1},p_\mu,\cdots,p_\tau}$ is the $(q_1, \cdots, q_{\mu-1}, p_\mu, \cdots, p_\tau)$th element of $D[\mu]^\tau$.

3) Reorder the elements of the τ-dimensional array H^τ in an α-dimensional array G^α in such a way that G^α is the Δ^α-vertical rearrangement of H^τ.

F'-Algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$:

This algorithm differs from the T'-algorithm previously described only for the fact that in point 2) Φ^τ and Ψ^τ are the Δ^τ-projections of type F of Φ^α and Ψ^α, and successive one-dimensional GFT's along every coordinate of $D[i]^\tau$ are evaluated by means of the F-algorithm.

Theorem 5: Let us consider the α-dimensional array E^α of $\prod_{s=1}^{\alpha} T_s$ elements and the parameter vectors Φ^α and Ψ^α, and let us suppose that the three sets of integers $\Delta^\alpha = \{T_1, T_2, \cdots, T_\alpha\}$, $\Delta^\tau = \{P_1, P_2, \cdots, P_\tau\}$, and $\Delta^\sigma = \{N_1, N_2, \cdots, N_\sigma\}$ are such that: a) $T_s = P_{v_s+1}\, P_{v_s+2} \cdots P_{v_{s+1}}$, $s = 1, 2, \cdots, \alpha$, where $v_1 = 0$, $v_{s+1} > v_s$, and $v_{\alpha+1} = \tau$, and b) $P_r = N_{\delta_r+1} N_{\delta_r+2} \cdots N_{\delta_{r+1}}$, $r = 1, 2, \cdots, \tau$, where $\delta_1 = 0$, $\delta_{r+1} > \delta_r$, and $\delta_{\tau+1} = \sigma$. The T-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$ and the T'-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data. Likewise, the F-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$ and the F'-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data.

VII. Concluding Remarks

In this work two pairs of fast algorithms for computing a multidimensional GFT are presented; they are called the T-algorithm and F-algorithm and the T'-algorithm and F'-algorithm, respectively.

It is shown that in the case of the DFT of a vector the T-algorithm and the F-algorithm represent a form of the classical FFT algorithms in mixed radix based on decimation in time and on decimation in frequency, respectively. Moreover, it is proved that the T-algorithm and the T'-algorithm, as well as the F-algorithm and the F'-algorithm, involve exactly the same arithmetic operations on the same data.

The T'-algorithm and the F'-algorithm are relevant if we want to evaluate an α-dimensional DFT of an α-dimensional array E^α on the hypothesis that single transforms along the coordinates of the data array cannot be handled as a whole by the available processor. As an example, suppose that all the dimensions of E^α are equal to T (where T is a power of N) and that the working area is sufficient to perform a GFT of P elements at most (where T is a power of P and P is a power of N). In this case it is possible to evaluate the α-dimensional DFT of E by using the T'-algorithm or the F'-algorithm which consist of reordering the elements of E^α in an array F^τ having all the dimensions equal to P, and in evaluating by means of the T-algorithm or the F-algorithm successive one-dimensional GFT's (along every coordinate of the τ-dimensional data array) of vectors having P elements each.

In this work the GFT has been utilized as a mechanism for evaluating in a multidimensional way the DFT of a vector or of an array. On the other hand, observe that the GFT can be useful in other applications, such as the evaluation of the DFT of staggered blocks [6], [7] and the evaluation of the DFT of a vector circular shifted or the DFT circular shifted of a vector. Let us consider, for example, the DFT of a vector E right circular shifted of a positions, a being an integer. Such a DFT is given by

$$g_z = \sum_{t=0}^{T-1} e_{|t-a|_T} W[T]^{tz} \qquad z = 0, 1, \cdots, T-1$$

where $|x|_y$ means x modulo y. By putting $|t-a|_T = t'$ we obtain

$$g_z = \sum_{t'=0}^{T-1} e_{t'} W[T]^{(t'+a)z} \qquad z = 0, 1, \cdots, T-1.$$

Then the DFT of E right circular shifted of a positions can be obtained by evaluating the GFT of $(E, a, 0)$. Likewise, it can easily be verified that the DFT right circular shifted of b positions (b being an integer) of a vector E can be obtained by evaluating the GFT of $(E, 0, b)$. Remember that the F-algorithm for evaluating the GFT of a vector with frequency parameter equal to zero and the T-algorithm for evaluating the GFT of a vector with time parameter equal to zero have the same complexity as an FFT algorithm (see Section III). Then, if we want to evaluate the DFT of a vector circular shifted, or the DFT circular shifted of a vector, it is convenient to directly perform the GFT instead of the phase shift method given in [12] which consists of the evaluation of the DFT of E and of T further multiplications.

Appendix

Proof of Theorem 4: The proof is made for the T-algorithm. The case of the F-algorithm can be similarly treated.

Point 2) of the T-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$ can be written as

$$\begin{aligned} h^\sigma_{k_1,k_2,\cdots,k_\sigma} &= \sum_{n_\sigma=0}^{N_\sigma-1}\left(\cdots\left(\sum_{n_2=0}^{N_2-1}\left(\sum_{n_1=0}^{N_1-1} f^\sigma_{n_1,n_2,\cdots,n_\sigma} W[N_1]^{(n_1+a_1^\sigma)(k_1+b_1^\sigma)}\right)\right.\right. \\ &\quad \left.\left. \cdot\, W[N_2]^{(n_2+a_2^\sigma)(k_2+b_2^\sigma)}\right)\cdots\right) W[N_\sigma]^{(n_\sigma+a_\sigma^\sigma)(k_\sigma+b_\sigma^\sigma)} \end{aligned} \tag{7}$$

where F^σ is the Δ^σ-horizontal rearrangement of E^α, and Φ^σ and Ψ^σ are the Δ^σ-projections of type T of Φ^α and Ψ^α. By taking into account relation (6), the product

$$W[N_{\gamma_s+1}]^{(n_{\gamma s+1}+a^\sigma_{\gamma s+1})(k_{\gamma s+1}+b^\sigma_{\gamma s+1})} \cdots W[N_{\gamma_s+2}]^{(n_{\gamma s+1}+a^\sigma_{\gamma s+1})(k_{\gamma s+1}+b^\sigma_{\gamma s+1})}$$

can be rewritten as

$$W[T_s]^{(t_s+a_s^\alpha)(z_s+b_s^\alpha)}$$

where

$$t_s = \sum_{u=\gamma_s+1}^{\gamma_{s+1}} \left(\prod_{v=u+1}^{\gamma_{s+1}} N_v\right) n_u$$

and

$$z_s = \sum_{u=\gamma_s+1}^{\gamma_{s+1}} \left(\prod_{v=\gamma_s+1}^{u-1} N_v\right) k_u.$$

By denoting with G^α the Δ^α-vertical rearrangement of H^σ, it follows that relation (7) can be rewritten as

$$g^\alpha_{z_1,z_2,\cdots,z_\alpha} = \sum_{t_\alpha=0}^{T_\alpha-1}\left(\cdots\left(\sum_{t_2=0}^{T_2-1}\left(\sum_{t_1=0}^{T_1-1} e_{t_1,t_2,\cdots,t_\alpha} W[T_1]^{(t_1+a_1^\alpha)(z_1+b_1^\alpha)}\right)\cdot W[T_2]^{(t_2+a_2^\alpha)(z_2+b_2^\alpha)}\right)\cdots\right) W[T_\alpha]^{(t_\alpha+a_\alpha^\alpha)(z_\alpha+b_\alpha^\alpha)}$$

Q.E.D.

Proof of Theorem 2: The proof is made for the T-algorithm and the FFT algorithm based on decimation in time. The case of the F-algorithm and the FFT algorithm based on decimation in frequency can be similarly treated.

The T-algorithm on $(E, 0, 0, \Delta^\sigma)$ involves the computation of the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$, where F^σ is the Δ^σ-horizontal rearrangement of E and Φ^σ and Ψ^σ are the Δ^σ-projections of type T of 0 and 0. From relation (3) it follows that the elements of Φ^σ are all equal to zero, while the elements of Ψ^σ are given by

$$b_u^\sigma = \frac{(k_1 + N_1 k_2 + \cdots + N_1 N_2 \cdots N_{u-2} k_{u-1})}{(N_1 N_2 \cdots N_{u-1})} \qquad u = 1, 2, \cdots, \sigma. \tag{8}$$

The computation of the GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$ consists of σ-steps (see note of Definition 2) in the ith of which the following quantity is evaluated for every value combination of $k_1, \cdots, k_{i-1}, n_{i+1}, \cdots, n_\sigma$, say $k_1^*, \cdots, k_{i-1}^*, n_{i+1}^*, \cdots, n_\sigma^*$:

$$d[i+1]^\sigma_{k_1^*,\cdots,k_{i-1}^*,k_i,n_{i+1}^*,\cdots,n_\sigma^*} = \sum_{n_i=0}^{N_i-1} d[i]^\sigma_{k_1^*,\cdots,k_{i-1}^*,n_i,n_{i+1}^*,\cdots,n_\sigma^*} \cdot W[N_i]^{n_i(k_i+b_i^\sigma)}, \qquad k_i = 0, 1, \cdots, N_i - 1. \tag{9}$$

By taking into account relation (8), the quantity $W[N_i]^{n_i(k_i+b_i^\sigma)}$ can be rewritten as

$$W[T]^{n_i(k_1+N_1k_2+\cdots+N_1N_2\cdots N_{i-1}k_i)N_{i+1}N_{i+2}\cdots N_\sigma}.$$

Therefore, the recursive equation (9) is a different form of the recursive equation (16) given in [10] which represents the Cooley–Tukey FFT algorithm in mixed radix. Q.E.D.

Proof of Theorem 5: The proof is made for the T-algorithm and the T'-algorithm. The case of the F-algorithm and the F'-algorithm can be similarly treated.

Let F^τ be the Δ^τ-horizontal rearrangement of E^α, and let Φ^τ and Ψ^τ be the Δ^τ-projections of type T of Φ^α and Ψ^α. First we will prove that the T-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$ and the T-algorithm on $(F^\tau, \Phi^\tau, \Psi^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data [point a)], and second we will prove that the T-algorithm on $(F^\tau, \Phi^\tau, \Psi^\tau, \Delta^\sigma)$ and the T'-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$ involve exactly the same arithmetic operations on the same data [point b)].

Point a): The T-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\sigma)$ involves the evaluation of the σ-dimensional GFT of a σ-dimensional array $F^{\sigma'}$ with parameter vectors $\Phi^{\sigma'}$ and $\Psi^{\sigma'}$, where $F^{\sigma'}$ is the Δ^σ-horizontal rearrangement of E^α, and Φ^{σ} and $\Psi^{\sigma'}$ are the Δ^σ-projections of type T of Φ^α and Ψ^α. Likewise, the T-algorithm on $(F^\tau, \Phi^\tau, \Psi^\tau, \Delta^\sigma)$ involves the evaluation of the σ-dimensional GFT of a σ-dimensional array $F^{\sigma''}$ with parameter vectors $\Phi^{\sigma''}$ and $\Psi^{\sigma''}$, where $F^{\sigma''}$ is the Δ^σ-horizontal rearrangement of F^τ, and $\Phi^{\sigma''}$ and $\Psi^{\sigma''}$ are the Δ^σ-projections of type T of Φ^τ and Ψ^τ. Point a) is proven if we verify that $F^{\sigma'} = F^{\sigma''}$, $\Phi^{\sigma'} = \Phi^{\sigma''}$, and $\Psi^{\sigma'} = \Psi^{\sigma''}$. From Definition 5 it results

$$f^{\sigma'}_{n_1,n_2,\cdots,n_\sigma} = e^\alpha_{t_1,t_2,\cdots,t_\alpha} \tag{10a}$$

for

$$t_s = \sum_{u=\gamma_s+1}^{\gamma_{s+1}} \left(\prod_{v=u+1}^{\gamma_{s+1}} N_v\right) n_u, \qquad s = 1, 2, \cdots, \alpha \tag{10b}$$

where $\gamma_s = \delta_{v_s+1}$. Likewise, from the same definition it results

$$f^\tau_{p_1,p_2,\cdots,p_\tau} = e^\alpha_{t_1,t_2,\cdots,t_\alpha}$$

for

$$t_s = \sum_{u=v_s+1}^{v_{s+1}} \left(\prod_{v=u+1}^{v_{s+1}} P_v\right) p_u, \qquad s = 1, 2, \cdots, \alpha \tag{11}$$

and

$$f^{\sigma''}_{n_1,n_2,\cdots,n_\sigma} = f^\tau_{p_1,p_2,\cdots,p_\tau}$$

for

$$p_r = \sum_{u=\delta_r+1}^{\delta_{r+1}} \left(\prod_{v=u+1}^{\delta_{r+1}} N_v\right) n_u, \qquad r = 1, 2, \cdots, \tau. \tag{12}$$

From relations (11) and (12) we have that

$$f^{\sigma''}_{n_1,n_2,\cdots,n_\sigma} = e^\alpha_{t_1,t_2,\cdots,t_\alpha} \tag{13a}$$

for

$$t_s = \sum_{u=v_s+1}^{v_{s+1}} \left(\prod_{v=u+1}^{v_{s+1}} P_v\right)\left[\sum_{x=\delta_u+1}^{\delta_{u+1}} \left(\prod_{y=x+1}^{\delta_{u+1}} N_y\right) n_x\right] \qquad s = 1, 2, \cdots, \alpha \tag{13b}$$

where $P_v = N_{\delta_v+1} \cdots N_{\delta_{v+1}}$. Relation (13b) can be written as

$$t_s = \sum_{u=v_s+1}^{v_{s+1}} \left[\sum_{x=\delta_u+1}^{\delta_{u+1}} \left(\prod_{y=x+1}^{\delta_{u+1}} N_y\right)\left(\prod_{v=\delta_{u+1}+1}^{\delta_{\gamma_{s+1}}+1} N_v\right)\right] n_x = \sum_{x=\delta_{\gamma_s+1}+1}^{\delta_{\gamma_{s+1}}+1} \left(\prod_{y=x+1}^{\delta_{\gamma_{s+1}}+1} N_y\right) n_x. \tag{14}$$

Therefore, since (14) coincides with (10b), we have that $F^{\sigma'} = F^{\sigma''}$.

In a similar way it can be proved that $\Phi^{\sigma'} = \Phi^{\sigma''}$ and $\Psi^{\sigma'} = \Psi^{\sigma''}$.

Point b): The T-algorithm on $(F^\tau, \Phi^\tau, \Psi^\tau, \Delta^\sigma)$ involves the evaluation of the σ-dimensional GFT of $(F^\sigma, \Phi^\sigma, \Psi^\sigma)$, say H^σ, where F^σ is the Δ^σ-horizontal rearrangement of F^τ and Φ^σ and Ψ^σ are the Δ^σ-projections of type T of Φ^τ and Ψ^τ. Let us evaluate H^σ as explained in the note of Definition 2. Let us put

$$c[i]^\sigma_{k_1,\cdots,k_{\delta_i},n_{\delta_i+1},\cdots,n_\sigma} = \sum_{n_{\delta_i}=0}^{N_{\delta_i}-1}\left(\cdots\left(\sum_{n_1=0}^{N_1-1} f^\sigma_{n_1,\cdots,n_{\delta_i},n_{\delta_i+1},\cdots,n_\sigma}\, W[N_1]^{(n_1+a_1^\sigma)(k_1+b_1^\sigma)}\right)\cdots\right)W[N_{\delta_i}]^{(n_{\delta_i}+a^\sigma_{\delta_i})(k_{\delta_i}+b^\sigma_{\delta_i})}.$$

Obviously, it results

$$\begin{aligned} c[1]^\sigma_{n_1,\cdots,n_\sigma} &= f^\sigma_{n_1,\cdots,n_\sigma} \\ c[\sigma+1]^\sigma_{k_1,\cdots,k_\sigma} &= h^\sigma_{k_1,\cdots,k_\sigma}. \end{aligned} \tag{15}$$

Then we can think of evaluating H^σ in τ steps (without altering the involved arithmetic operations), the ith of which consists of performing, for every value combination of $k_1, \cdots, k_{\delta_i}, n_{\delta_{i+1}+1}, \cdots, n_\sigma$, say $k_1^*, \cdots, k_{\delta_i}^*, n^*_{\delta_{i+1}+1}, \cdots, n_\sigma^*$, the computation

$$\begin{aligned} &c[i+1]^\sigma_{k_1^*,\cdots,k_{\delta_i}^*,k_{\delta_i+1},\cdots,k_{\delta_{i+1}},n^*_{\delta_{i+1}+1},\cdots,n^*_\sigma} \\ &= \sum_{n_{\delta_{i+1}}=0}^{N_{\delta_{i+1}}-1}\left(\cdots\left(\sum_{n_{\delta_i+1}=0}^{N_{\delta_i+1}-1} c[i]^\sigma_{k_1^*,\cdots,k_{\delta_i}^*,n_{\delta_i+1},\cdots,n_{\delta_{i+1}},n^*_{\delta_{i+1}+1},\cdots,n^*_\sigma}\right.\right. \\ &\left.\left.\cdot\, W[N_{\delta_i+1}]^{(n_{\delta_i+1}+a^\sigma_{\delta_i+1})(k_{\delta_i+1}+a^\sigma_{\delta_i+1})}\right)\cdots\right) \\ &\cdot\, W[N_{\delta_{i+1}}]^{(n_{\delta_{i+1}}+a^\sigma_{\delta_{i+1}})(k_{\delta_{i+1}}+b^\sigma_{\delta_{i+1}})}. \end{aligned} \tag{16}$$

The T'-algorithm on $(E^\alpha, \Phi^\alpha, \Psi^\alpha, \Delta^\tau, \Delta^\sigma)$ consists of τ steps, in the ith of which for every value combination of $q_1, \cdots, q_{i-1}, p_{i+1}, \cdots, p_\tau$, say $q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*$, the T-algorithm on

$$(\{d[i]^\tau_{q_1^*,\cdots,q_{i-1}^*,q_i,p_{i+1}^*,\cdots,p_\tau^*}\}, a_i^\tau, b_i^\tau, \{N_{\delta_i+1}, \cdots, N_{\delta_{i+1}}\})$$

is performed where

$$d[1]^\tau_{p_1,\cdots,p_\tau} = f^\tau_{p_1,\cdots,p_\tau}. \tag{17}$$

Then the ith step involves the following computation:

$$\begin{aligned} &g[i](q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)_{q_1^i,\cdots,q_{\lambda_i}^i} \\ &= \sum_{p^i_{\lambda_i}=0}^{N_{\delta_{i+1}}-1}\left(\cdots\left(\sum_{p_1^i=0}^{N_{\delta_i+1}-1} e[i](q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)_{p_1^i,\cdots,p^i_{\lambda_i}}\right.\right. \\ &\left.\left.\cdot\, W[N_{\delta_i+1}]^{(p_1^i+a[i]_1)(q_1^i+b[i]_1)}\right)\cdots\right)\cdot \\ &\qquad W[N_{\delta_{i+1}}]^{(p^i_{\lambda_i}+a[i]_{\lambda_i})(q^i_{\lambda_i}+b[i]_{\lambda_i})} \end{aligned} \tag{18}$$

where $\lambda_i = \delta_{i+1} - \delta_i$ and

$$e[i](q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)_{p_1^i,\cdots,p^i_{\lambda_i}} = d[i]^\tau_{q_1^*,\cdots,q_{i-1}^*,p_i,p_{i+1}^*,\cdots,p_\tau^*}$$

for

$$p_i = \sum_{u=1}^{\lambda_i}\left(\prod_{v=u+1}^{\lambda_i} N_{\delta_i+v}\right) p_u^i \tag{19}$$

and

$$a[i]_u = \begin{cases} 0 & \text{for } 1 \le u < \lambda_i \\ a_i^\tau & \text{for } u = \lambda_i \end{cases} \qquad u = 1, 2, \cdots, \lambda_i \tag{20}$$

$$b[i]_u = \frac{\sum_{x=1}^{u-1}\left(\prod_{v=1}^{x-1} N_{\delta_i+v}\right) q_x^i + b_i^\tau}{\prod_{x=1}^{u-1} N_{\delta_i+x}}$$

and

$$d[i+1]^\tau_{q_1^*,\cdots,q_{i-1}^*,q_i,p_{i+1}^*,\cdots,p_\tau^*} = g[i](q_1^*, \cdots, q_{i-1}^*, p_{i+1}^*, \cdots, p_\tau^*)_{q_1^i,\cdots,q^i_{\lambda_i}}$$

for

$$q_i = \sum_{u=1}^{\lambda_i}\left(\prod_{v=1}^{u-1} N_{\delta_i+v}\right) q_u^i. \tag{21}$$

Since the extremes of the summations in relations (16) and (18) are the same, we can put

$$\begin{aligned} p_u^i &= n_{\delta_i+u} \\ q_u^i &= k_{\delta_i+u} \end{aligned} \qquad u = 1, 2, \cdots, \lambda_i. \tag{22}$$

Then from relation (20) and from the fact that Φ^σ and Ψ^σ are the Δ^σ-projections of type T of Φ^τ and Ψ^τ it follows that

$$\begin{aligned} a[i]_u &= a^\sigma_{\delta_i+u} \\ b[i]_u &= b^\sigma_{\delta_i+u} \end{aligned} \qquad u = 1, 2, \cdots, \lambda_i. \tag{23}$$

Moreover, from relations (17) and (19) it results

$$e[1](p_2^*, \cdots, p_\tau^*)_{p_1,\cdots,p_{\lambda_1}} = f^\tau_{p_1,p_2^*,\cdots,p_\tau^*}$$

for

$$p_1 = \sum_{u=1}^{\lambda_1}\left(\prod_{v=u+1}^{\lambda_1} N_v\right) p_u^1.$$

Likewise, from relation (15) and from the fact that F^σ is the Δ^σ-horizontal rearrangement of F^τ it results

$$c[1]_{n_1,\cdots,n_{\delta_2},n^*_{\delta_2+1},\cdots,n_\sigma^*} = f^\tau_{p_1,p_2^*,\cdots,p_\tau^*}$$

for

$$p_1 = \sum_{u=1}^{\delta_2}\left(\prod_{v=u+1}^{\delta_2} N_v\right) n_u$$

$$p_s^* = \sum_{u=\delta_s+1}^{\delta_{s+1}}\left(\prod_{v=u+1}^{\delta_{s+1}} N_v\right) n_u^*, \qquad s = 2, 3, \cdots, \tau.$$

Then we have (remember that $\lambda_1 = \delta_2$)

$$e[1](p_2^*, \cdots, p_\tau^*)_{p_1^1,\cdots,p^1_{\lambda_1}} = c[1]_{n_1,\cdots,n_{\lambda_1},n^*_{\lambda_1+1},\cdots,n^*_\sigma}$$

for

$$\begin{aligned} p_u^1 &= n_u, \qquad u = 1, 2, \cdots, \lambda_1 \\ p_s^* &= \sum_{u=\delta_s+1}^{\delta_{s+1}}\left(\prod_{v=u+1}^{\delta_{s+1}} N_v\right) n_u^*, \qquad s = 2, 3, \cdots, \tau. \end{aligned} \tag{24}$$

Since relations (16) and (18) have the same form, by taking into account relations (22), (23), and (24), it results that for $i = 1$ the same arithmetic operations on the same data are performed in relations (16) and (18) and the same result is obtained. That is,

$$g[1](p_2^*, \cdots, p_\tau^*)_{q_1^1,\cdots,q^1_{\lambda_1}} = c[2]_{k_1,\cdots,k_{\lambda_1},n^*_{\lambda_1+1},\cdots,n^*_\sigma}$$

for

$$q_u^1 = k_u, \qquad u = 1, 2, \cdots, \lambda_1 \tag{25}$$

$$p_s^* = \sum_{u=\delta_s+1}^{\delta_{s+1}} \left(\prod_{v=u+1}^{\delta_{s+1}} N_v \right) n_u^*, \qquad s = 2, 3, \cdots, \tau.$$

By taking into account relations (19) and (21) we have

$$e[2](q_1^*, p_3^*, \cdots, p_\tau^*)_{p_1^2, \cdots, p_{\lambda 2}^2}$$

$$= g[1](p_2, p_3^*, \cdots, p_\tau^*)_{q_1^{1*}, \cdots, q_{\lambda 1}^{1*}}$$

for

$$q_1^* = \sum_{u=1}^{\lambda_1} \left(\prod_{v=1}^{u-1} N_v \right) q_u^{1*} \tag{26}$$

$$p_2 = \sum_{u=1}^{\lambda_2} \left(\prod_{v=u+1}^{\lambda_2} N_{\lambda_1+v} \right) p_u^2.$$

Then from relations (25) and (26)

$$e[2](q_1^*, p_3^*, \cdots, p_\tau^*)_{p_1^2, \cdots, p_{\lambda 2}^2} =$$

$$= c[2]_{k_1^*, \cdots, k_{\delta 2}^*, n_{\delta 2+1}, \cdots, n_{\delta 2+\lambda 2}, n_{\delta 2+\lambda 2+1}^*, \cdots, n_\sigma^*}$$

for

$$q_1^* = \sum_{u=1}^{\lambda_1} \left(\prod_{v=1}^{u-1} N_v \right) k_u^* \tag{27}$$

$$p_u^2 = n_{\delta_2+u}, \qquad u = 1, 2, \cdots, \lambda_2$$

$$p_s^* = \sum_{u=\delta_s+1}^{\delta_{s+1}} \left(\prod_{v=u+1}^{\delta_{s+1}} N_v \right) n_u^*, \qquad s = 3, 4, \cdots, \tau.$$

By reasonings similar to those relative to $i = 1$, by starting from (27) it can be shown that for $i = 2$ the same arithmetic operations on the same data are involved in relations (16) and (18), and so on. Q.E.D.

References

[1] H. L. Buijs, "Fast Fourier transformation of large arrays of data," *Appl. Opt.*, vol. 8, pp. 211–212, Jan. 1963.

[2] N. M. Brenner, "Fast Fourier transform of externally stored data," *IEEE Trans. Audio Electroacoust.*, vol. AU-17, pp. 128–132, June 1963.

[3] R. C. Singleton, "A method for computing the fast Fourier transform with auxiliary memory and limited high-speed storage," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 91–98, June 1967.

[4] B. Arazi, "Two-dimensional digital processing of one-dimensional signal," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 31–36, Apr. 1974.

[5] B. Gold and T. Bially, "Parallelism in fast Fourier transform hardware," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 5–16, Feb. 1973.

[6] G. Bongiovanni, P. Corsini, and G. Frosini, "Procedures for computing the discrete Fourier transform on staggered blocks," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 132–137, Apr. 1976.

[7] P. Corsini and G. Frosini, "Structures for evaluating the discrete Fourier transform on staggered blocks," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 128–131, Apr. 1976.

[8] G. Bongiovanni, P. Corsini, and G. Frosini, "One-dimensional and two-dimensional generalized discrete Fourier transforms," *IEEE Trans. Acoust., Speech Signal Processing*, vol. ASSP-24, pp. 97–99, Feb. 1974.

[9] ——, "Two-dimensional properties of discrete Fourier transform," *Calcolo*, vol. XIII, pp. 131–143, Apr.–June 1976.

[10] J. W. Cooley and J. W. Tukey, "An algorithm for the machine computation of complex Fourier series," *Math. Comput.*, vol. 19, pp. 297–301, Apr–June 1965.

[11] W. M. Gentleman and G. Sande, "Fast Fourier transforms for fun and profit," in *1966 Fall Joint Comput. Conf., AFIPS Conf. Proc.*, vol. 29, 1966, pp. 563–578.

[12] B. Gold and C. M. Rader, *Digital Processing of Signals*. New York: McGraw-Hill, 1969.

[13] G. D. Bergland, "The fast Fourier transform recursive equations for arbitrary length records," *Math. Comput.*, vol. 21, pp. 236–238, Apr. 1967.

Paolo Corsini was born in Arcidosso, Italy, on April 27, 1944. He received the Dott. Ing. degree in electronic engineering, cum laude, from the University of Pisa, Pisa, Italy, in 1969.

Presently he is an Associate Professor of Digital Computers with the Engineering Faculty, University of Pisa, Pisa, Italy. His current research interests lie in computer architecture, multimicroprocessor systems, and signal processing.

Graziano Frosini was born in Lari, Italy, on April 17, 1942. He received the Dott. Ing. degree in electronic engineering, cum laude, from the University of Pisa, Pisa, Italy, in 1967.

Since 1967 he has been with the Istituto di Elaborazione della Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy, where he was engaged in research concerning computer system architectures. Since 1971 he has also been an Associate Professor of Digital Computers with the Engineering Faculty, University of Pisa, Pisa, Italy. His present interests are in the areas of multimicroprocessor systems and architectures for signal processing.