DTU Library

z-transform DFT filters and FFT's

Bruun, G.

Published in:
Institute of Electrical and Electronics Engineers. Transactions on Acoustics, Speech and Signal Processing

Publication date:
1978

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

z-Transform DFT Filters and FFT's

GEORG BRUUN, SENIOR MEMBER, IEEE

Abstract—The paper shows how discrete Fourier transformation can be implemented as a filter bank in a way which reduces the number of filter coefficients. A particular implementation of such a filter bank is directly related to the normal complex FFT algorithm. The principle developed further leads to types of DFT filter banks which utilize a minimum of complex coefficients. These implementations lead to new forms of FFT's, among which is a cos/sin FFT for a real signal which only employs real coefficients. The new FFT algorithms use only half as many real multiplications as does the classical FFT.

INTRODUCTION

IN this paper we deal with filters which perform discrete Fourier transforms (DFT). In one type of such filters, z-transform filters, a sampled signal is analyzed by means of delay sections and multipliers. Such a filter can either be analog or digital. The analog types of z-transform filters have recently become of increasing importance, as they can be implemented as transversal filters by using CCD or bucket-brigade circuits as delay sections.

In a study of DFT z-transform filters we are attacking the problem: *Can we at the cost of an increased number of delay sections reduce the number of multipliers?* It appears that this is possible. The result of the search leads to a general type of filter which in a special case handles the signal to be analyzed in a way which is analog to the classical fast Fourier transform (FFT) algorithm.

In the analysis of a signal using the z-transform filter, to obtain the FFT algorithm we just need at a fixed time to consider how the set of filter output sample values are calculated from the set of sample values of the input signal loaded into the filter at that particular time. An example of this procedure is given in Section V. The result is the classical FFT.

Using the methods mentioned above, we develop in Section VI types of FFT algorithms which to the best of the author's knowledge are new. In Section VI, the following question is raised concerning the z-transform filter: *Is it possible to reduce the number of complex multipliers?* In answer to this question, the general filter principle mentioned above leads to a filter configuration, which performs complex DFT with a minimum of complex coefficients, and to another filter with only real coefficients which performs cos- and sin-DFT on a real signal. These two filters lead directly to the new FFT algorithms.

I. THE DISCRETE FOURIER TRANSFORMS

The representations of the DFT we are dealing with are the complex DFT and the real cos/sin DFT's.

A. The Complex DFT

Given N complex values $x(0), x(1), x(2), \cdots, x(N-1)$, the DFT is defined by

$$a(n) = \sum_{k=0}^{N-1} x(k) \, W_N^{-nk} \tag{1}$$

where

$$W_N = \exp(2\pi j/N). \tag{2}$$

n can take the values $0, 1, \cdots (N-1)$.

B. The Real DFT

Given N real values $y(0), y(1), y(2), \cdots, y(N-1)$, the cos-DFT is defined by

$$A(n) = \sum_{k=0}^{N-1} y(k) \cos(2\pi nk/N) \tag{3}$$

and the sin-DFT by

$$B(n) = \sum_{k=1}^{N-1} y(k) \sin(2\pi nk/N). \tag{4}$$

We define $A(n)$ for $n = 0, 1, \cdots, (N/2)$ and $B(n)$ for $n = 1, 2, \cdots, (N/2 - 1)$; N is assumed being even.

II. THE DFT z-TRANSFORMS

We let the two series $x(k)$ and $y(k)$ represent two time series, $x(0), x(1), \cdots, x(k) \ldots$ being signal values at times $t = 0, 1, \cdots, k, \cdots$, and similarly for the $y(k)$ series. The DFT's can then be implemented by well-known transversal filters. Fig. 1 shows such a filter for the complex DFT. The sampled data delay line or shift register shown on top of the figure contains the complex time series signal. The Fourier transforms of the time series are taken from a set of adding devices after multiplication has taken place.

The result which is obtained in accordance with (1) is obtained at time $t = N - 1$. Because of the intricacy of the complex multiplications needed, it is most realistic to think of the filter shown as a digital filter. The filter of Fig. 1 provides N z-transfer functions, one at each port, of the form

$$F_n(z) = z^{-(N-1)} + W^{-1 \cdot n} z^{-(N-2)}$$
$$+ \cdots W^{-k \cdot n} z^{-(N-k-1)} + \cdots W^{-(N-1) \cdot n}. \tag{5}$$

The transfer functions corresponding to the real transforms $A(n)$ and $B(n)$ in (3) and (4) are considered under the assumption of N being even.

The transfer functions for the $A(n)$ coefficient part of the filter are
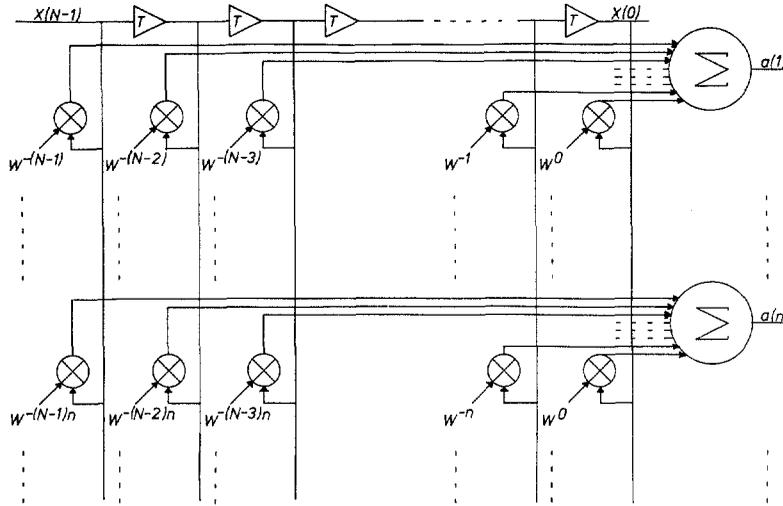
Fig. 1. Transversal filter performing complex DFT; the $a(0)$ circuit is not shown.

$$F_{An}(z) = z^{-(N-1)} + \cos{(n \cdot 2\pi \cdot 1/N)}z^{-(N-2)}$$
$$+ \cos{(n \cdot 2\pi \cdot 2/N)}z^{-(N-3)}$$
$$\cdots + \cos{(n \cdot 2\pi \cdot (N-1)/N)} \qquad (6)$$

and for the $B(n)$ coefficients

$$F_{Bn}(z) = \sin{(n \cdot 2\pi \cdot 1/N)}z^{-(N-2)}$$
$$+ \sin{(n \cdot 2\pi \cdot 2/N)}z^{-(N-3)}$$
$$\cdots + \sin{(n \cdot 2\pi(N-1)/N)}. \qquad (7)$$

III. THE CONFIGURATION OF ZEROS

The filter from Fig. 1 and the corresponding $A(n)$ and $B(n)$ filters can in principle, of course, be operating at all different frequencies. For the sake of simplicity, we will in the following let them be thought of as operating at a sample frequency $f_s = 1$ Hz, which is a shift-angular frequency $\omega_s = 2\pi$ rad/s or a delay in the shift registers of $T = 1$ s. We will now consider the zero configurations for the filters corresponding to (5), (6), and (7).

From (1) we can see directly that the particular stationary signal $x_m(k) = \exp{(mk \cdot 2\pi j/N)}$, $m = 0, 1, \cdots, (N-1)$ will only be transmitted by the discrete filter $F_n(z)$ when the frequency m/N is equal to n/N. This is a way of realizing that the zeros of $F_n(z)$ will all be located at the unit circle of the z-plane, and only one of a set of equidistant zeros is missing. The zeros are at

$$p_{nk} = \exp{\left(j \frac{2\pi}{N} \cdot k\right)}, \quad k = 0, 1, 2, \cdots, (N-1)$$
$$k \neq n.$$

Fig. 2(a) shows the zero configuration for the case of $N = 16$ and $n = 3$.

From this result we can write

$$F_n(z) = \prod_{\substack{k=0 \\ k \neq n}}^{N-1} (z^{-1} - p_{nk}^{-1}). \qquad (8)$$

For the filter characterized by $F_{An}(z)$, (6) a transmission consideration analogous to the one used above shows that a steady-state signal $y_m(k) = \cos{(mk \cdot 2\pi/N)}$ also will only be

transmitted when $m = n$. This result indicates that we have the same set of equidistant zeros on the unit circle, but as we deal with a z-transform with real coefficients, a pair of conjugate zeros at $\exp{(\pm j(2\pi/N) \cdot n)}$ is excluded. This result is valid when $n \neq 0$ and $n \neq N/2$. When $n = 0$ or when $n = N/2$, one zero at $+1$ or at -1, respectively, is missing.

When $n \neq 0$ and $n \neq N/2$, we find that the highest power z^{-1} in $F_{An}(z)$ must be $(N-1)$. The remaining missing zero is found at $1/\cos{((2\pi/N) \cdot n)}$; see the example shown in Fig. 2(b) for the same values of N and n as in Fig. 2(a). To locate this extra zero, we write the expression from (6) in the form

$$F_{An}(z) = (z^{-1} - p_{nx}^{-1}) \prod_{\substack{k=0 \\ k \neq n \\ k \neq (N-n)}}^{N-1} (z^{-1} - p_{nk}^{-1}) \qquad (9)$$
$$\substack{n \neq 0 \\ n \neq \frac{N}{2}}$$

If, in the last part of (9), we combine terms two at a time corresponding to conjugate pairs of zeros, we find that the expression can be written in the form

$$F_{An}(z) = z^{-(N-1)} + \cdots p_{nx}^{-1}. \qquad (10)$$

Comparing (10) and (6) we confirm

$$p_{nx} = (\cos{(n \cdot 2\pi(N-1)/N)})^{-1} = \cos{\left(\frac{2\pi}{N} \cdot n\right)}^{-1}. \qquad (11)$$

Using the same method as was used for $F_{An}(z)$, we find the zeros of $F_{Bn}(z)$ (7) to be at the same equidistant locations on the unit circle as found in the two previous cases. Furthermore, a constant factor is needed as seen below. In this case, the same pair of conjugate zeros used for the cos-transformation is missing. We get

$$F_{Bn}(z) = \sin{(n \cdot 2\pi/N)} \prod_{\substack{k=0 \\ k \neq n \\ k \neq (N-n)}}^{N-1} (z^{-1} - p_{nk}^{-1}) \qquad (12)$$
$$\substack{n \neq 0 \\ n \neq \frac{N}{2}}$$

Fig. 2(c) shows an example of zero locations for $F_{Bn}(z)$.

IV. A FILTER TREE FOR THE COMPLEX DFT

From the results of the previous section we see that it is possible for several of the $F_n(z)$ sections of the filter to share
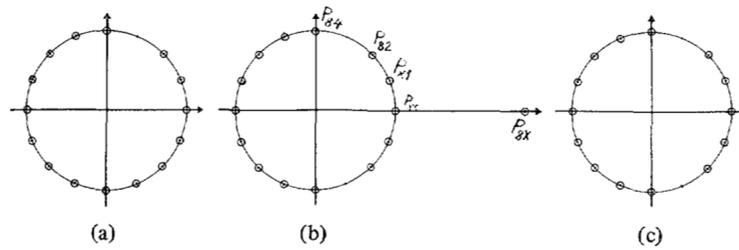
Fig. 2. Examples of zero configurations for DFT's; $N = 16$, $n = 3$ for (a) complex DFT, (b) cos-DFT, (c) sin-DFT.
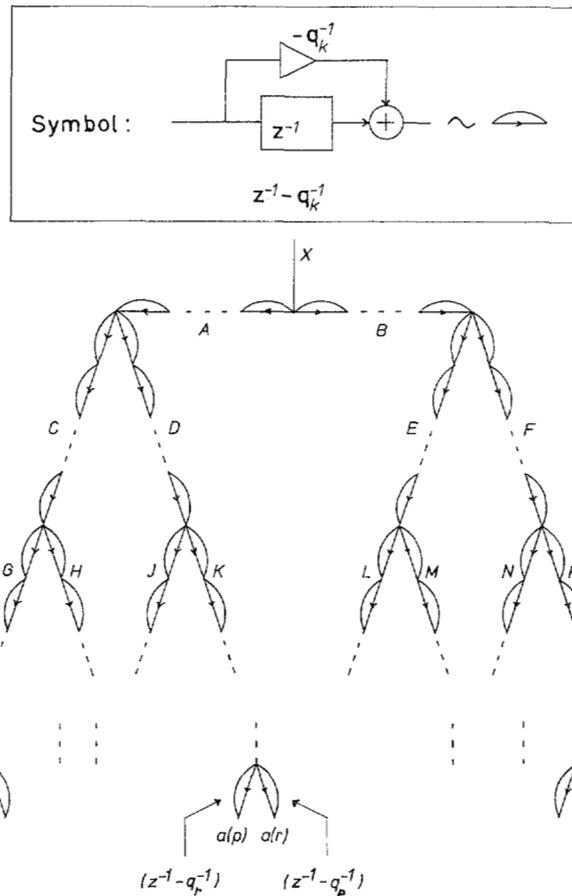

Fig. 3. Complex DFT-filter tree.

a number of zeros. This fact is utilized in the design procedure to be discussed in this section. We show a filter configuration which utilizes a much smaller number of multipliers, whereas, on the other hand, the total number of shift register units is much higher than the N units used for the filter shown in Fig. 1.

Such a filter which corresponds to the complex DFT-transfer functions (5) and (8) is shown schematically in Fig. 3. The filter is composed of filter units which are in principle of the same type as the transversal filter from Fig. 1. The filter units can, however, as we shall see in later examples, be much simpler as they can be made to contain only one or two multipliers. In Fig. 3, the filter units $A, B, C \cdots$ are thought of as consisting of a cascade of subunits of the type shown on top of Fig. 3, each representing one of the N zeros on the unit circle. A hypothetical filter unit containing the complete set of zeros is designated by U, and in the following, A, B, C, etc.,

designate the zeros contained in the particular filter unit. This filter is built according to the following principle:

$$A \cup B = U$$

$$C \cup D = B$$

$$E \cup F = A$$

$$J \cup K = C$$

$$L \cup M = F$$

$$N \cup P = E, \text{ etc.}$$

This way the filter may be built such that when passing from the input at the top to the last splitting node at the bottom, only two of the complete set U of unit circle zeros will be missing. For such a particular node, we will, for example, miss the zeros q_r and q_p. Now, by adding one extra set of
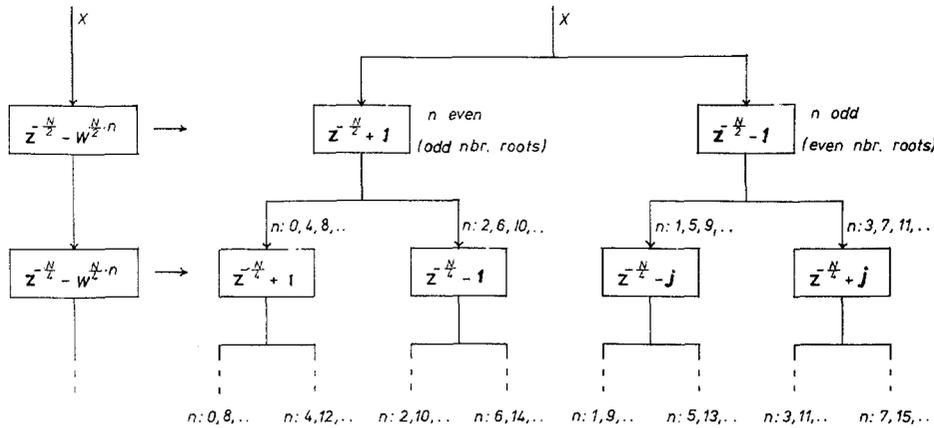
Fig. 4. Principle of complex DFT-filter tree.

one-zero branches to each of these nodes at the bottom of the filter, we obtain a complete DFT filter. For the particular node mentioned, the connection of a branch with gain $(z^{-1} - q_r^{-1})$, we obtain the output $a(p)$ and by a branch $(z^{-1} - q_p^{-1})$, we obtain $a(r)$.

According to this principle, a DFT filter can, of course, be built up in many different ways. One important class of configuration is the one made in such a way that all branches at a certain level have the same number of zeros. This procedure can, of course, only be followed when N is an integer power of 2. In the following, we will only deal with cases of this restriction.

In Fig. 4 is shown the principle of a DFT filter according to the principle last mentioned. The structure is developed as follows.

Equation (5) can be written

$$F_n(z) = (z^{-(N/2)} + W^{-(N/2) \cdot n})(z^{-((N/2)-1)}$$
$$+ W^{-n \cdot 1} z^{-((N/2)-2)} + \cdots + \cdots + W^{-((N/2)-1) \cdot n}).$$

(13)

In the second factor of (13), we find again a polynomium in z^{-1}, the difference from the original being that the highest power has been reduced by $N/2$; otherwise the form is exactly the same as of (5). From

$$W^{-(N/2) \cdot n} = \exp\left(-2\pi j \frac{n}{2}\right) = \begin{cases} 1 & (n \text{ even}) \\ -1 & (n \text{ odd}) \end{cases}$$

we find that the $N/2$ roots of the first factor in (13) are

$$P_k = \begin{cases} \exp\left(\dfrac{2\pi \cdot k}{(N/2)}\right) & \text{for } n \text{ even} \\ \exp\left(\dfrac{\pi + 2\pi \cdot k}{(N/2)}\right) & \text{for } n \text{ odd} \end{cases}$$

$$k = 0, 1, \cdots, \left(\frac{N}{2} - 1\right).$$

(14)

According to this result, we see in Fig. 4 how the blocks $(z^{-(N/2)} + 1)$ and $(z^{-(N/2)} - 1)$ are taken as the first branches

of the filter. Following the same procedure as above, we conclude with the following expression:

$$F_n(z) = \prod_{t=1}^{\log_2 N} (z^{-(N/v)} + W^{-(N/v) \cdot n})$$

(15)

where $v = 2^t$.

If we start from the top in Fig. 4, we find the blocks representing the first factor in (13) or the factor in (15) which corresponds to $t = 1$. The next factor is

$$(z^{-(N/4)} + W^{-(N/4) \cdot n}) = z^{-(N/4)} + \exp\left(-j \frac{\pi}{2} n (\text{modulo } 4)\right).$$

(16)

For the last term in (16) we find

$$\exp\left(-j \frac{\pi}{2} n (\text{modulo } 4)\right) = \begin{cases} 1 & [n = 0, 4, 8 \cdots \\ -j & [n = 1, 5, 9 \cdots \\ -1 & [n = 2, 6, 10 \cdots \\ +j & [n = 3, 7, 11 \cdots. \end{cases}$$

Blocks representing these results are shown in the second row of blocks in Fig. 4.

In Fig. 5, an example is shown how this development leads to a block diagram for determining the complex DFT's for the case of $N = 8$.

It is now necessary to show that the development which we have demonstrated is carried out in accordance with the principle discussed above in connection with Fig. 3. For the part of the filter which is shown in Fig. 4, we see directly that we follow the principle of Fig. 3. The two blocks in the upper row represent all the N zeros on the unit circle, and the product of the z-transforms for the two blocks to the right in the second row is equal to the z-transform for the block to the left in the first row. That the system, in general, follows the principle of Fig. 3 can be seen by induction.

V. THE DFT z-TRANSFORM FILTER AND THE FAST FOURIER TRANSFORM (FFT)

The filter developed in the previous paragraph can be used to show the development of the well-known algorithm of the FFT of a complex signal. Fig. 6 shows one of the possible
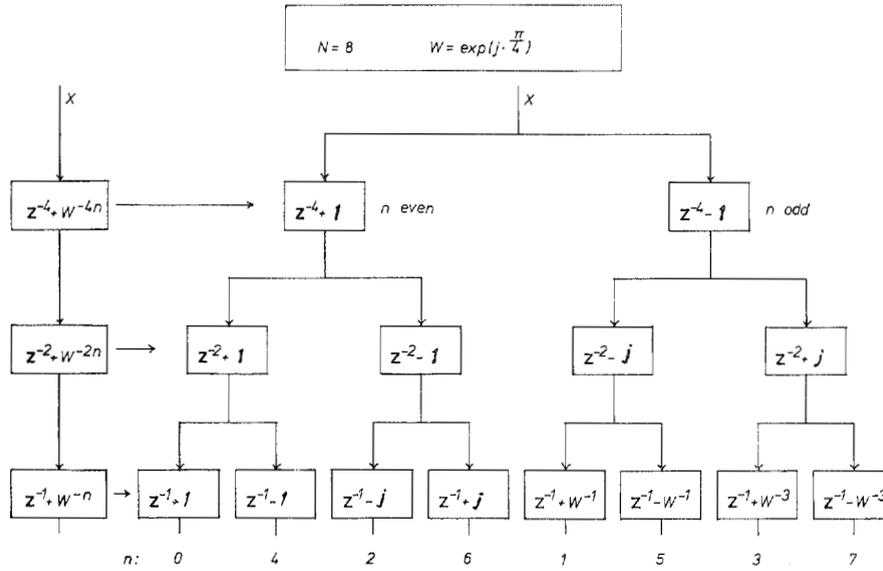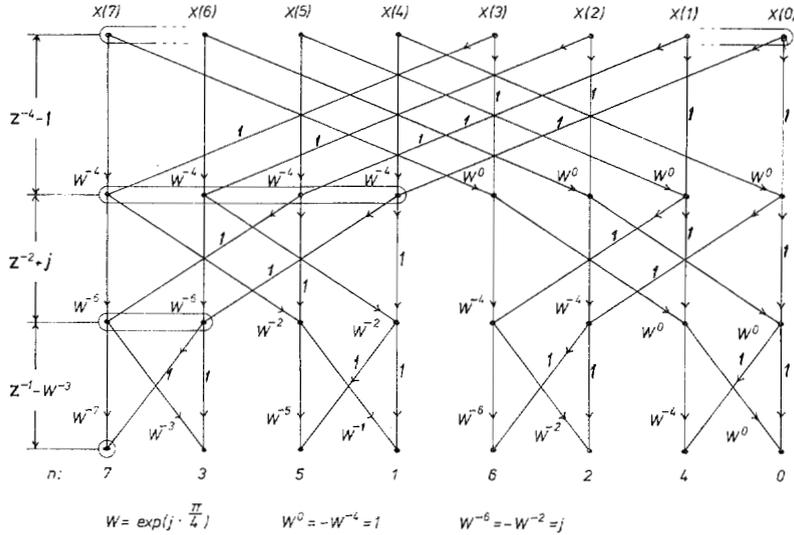
Fig. 5. Example of complex DFT-filter tree for $N = 8$.

Fig. 6. Signal-flow graph for classical FFT; $N = 8$. The graph can also be deducted from the filter tree in Fig. 5.

signal-flow diagrams for a complex 8-point FFT. On top of the diagram, nodes representing the input signal are indicated, and at the bottom we find the DFT values $a(0)$–$a(7)$. To see the relationship between the block diagram in Fig. 5 and the signal-flow graph, let us take the output from the block giving $a(7)$. This output is obtained through the $(z^{-1} - W^{-3})$ block, which means that the value $a(7)$ can be obtained from two other complex values, one of which is shifted one time unit in relation to the other. In Fig. 6 we see how this same result, $a(7)$, is obtained from two nodes which can be thought of as representing signal values with a relative shift of one time unit. From Fig. 5, we see that each of the latter signal values in turn are found from two other values with a time difference of two units. Following this kind of development, we are able to translate Fig. 5 directly into Fig. 6.

VI. DFT FILTER TREES WITH REAL COEFFICIENTS

Building up a DFT tree according to Fig. 3, we have a possibility of choosing branches with a maximum of real filter coefficients. For the filter shown in Fig. 4, only the first row

of blocks $(z^{-(N/2)} + 1)$ and $(z^{-(N/2)} - 1)$ have exclusively real coefficients. Using the following identity (17) we can, for the complex DFT, continue with real coefficients until we reach the bottom row of blocks. The identity we use is

$$1 + az^{2q} + z^{4q} = (1 + \sqrt{2 - a}\ z^q + z^{2q})(1 - \sqrt{2 - a}\ z^q + z^{2q}). \qquad (17)$$

Fig. 7 shows a filter for the case of $N = 16$ built up by utilizing an expansion which follows from (17) and is in accordance with the principle of Section IV. We see that the product of the z-transforms of two adjacent blocks a and b is equal to the z-transform of the block which is adjacent to the block a and b are connected to.

In Fig. 7 the blocks are ordered such that of two neighboring blocks, the one having plus signs before the square-root sign is placed above the minus block. In Fig. 8 is shown a table with the series of the same square-root coefficients for rows 2, 3, and 4 of a filter, i.e., coefficients from 3 of the 5 rows of blocks needed for a filter for $N = 32$. Note how a part of the same series of coefficients is used in each row.
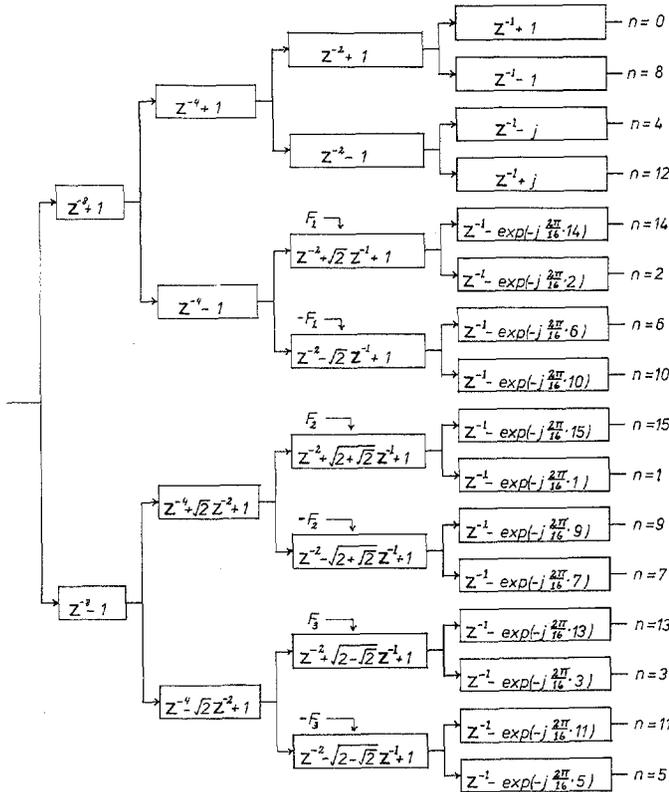
Fig. 7. Complex DFT-filter tree; $N = 16$. The filter type contains only real coefficients, except in the output blocks.

Fig. 8. Development of filter coefficients for filter of the type shown in Fig. 8 up to $N = 32$. Only the first two vertical rows of coefficients are used in a filter with $N = 16$.

In the identification of the output blocks of Fig. 7, we can use the identity

$$(z^{-1} - e^{-j\varphi})(z^{-1} - e^{+j\varphi}) = z^{-2} - 2 \cos \varphi \, z^{-1} + 1 \qquad (18)$$

where the expression to the right of the equality sign is found in the block in the next to the last row. The angle φ is a multiple of $2\pi/N$ where in the example $N = 16$. The following identity is now evident:

$$\cos \varphi = \cos \left(\frac{\pi}{2} - \frac{\pi}{4} A_1 \left(1 + A_2 \left(\frac{1}{2} + A_3 \left(\frac{1}{4} + A_4 \left(\frac{1}{8} + \cdots \right) \right) \right) \right) \right)$$

$$= \frac{1}{2} A_1 \sqrt{2 + A_2 \sqrt{2 + A_3 \sqrt{2 + A_4 \cdots}}} . \qquad (19)$$

All factors A_k are either $+1$ or -1 for $k \leqslant k_{\max}$. When $k > k_{\max}$, all A_k are equal to zero. Equation (19) can be used as a simple means of identifying the blocks in the last row.

Fig. 9 shows a signal-flow graph for the complex FFT corresponding to the block diagram in Fig. 7. In the first three levels of the graph, all gains are real. Of these gains, those which are shown as dotted lines are found from the square-root coefficients in Fig. 7, the gains of the rest of the branches at these levels are equal to unity.

VII. THE REAL-COEFFICIENT cos/sin TRANSFORM

In the complex real-coefficient transform developed above, we were able to deal with only real coefficients in all rows of a filter as the one shown in Fig. 7, except for the last row. We shall now see how a cos/sin transform of a real input signal can be developed having real coefficients in all of its blocks. As an example, an $N = 16$ filter is shown in Fig. 10 and its corre-

sponding FFT signal-flow graph is shown in Fig. 11. Except for the rightmost output blocks in the block diagram and the lowest set of branches in the graph, these figures are identical with Figs. 7 and 9, respectively. We also know from Section III that a sin- or cos-transform filter contains the same zeros as the complex-transform filter except for one set of zeros. In the case of $n \neq \{^0_{N/2}\}$, these missing zeros form a conjugate pair. In the complex filters described in Section V, we end up with blocks representing conjugate zeros connected together, or the $n = 0$ and $n = N/2$ blocks connected together. A sin/cos-transform filter can thus be obtained if we modify the last set of blocks. The procedure follows from the results found in Section V. Here we only consider the output blocks for the interesting cases when $n \neq \{^0_{N/2}\}$. To obtain the output blocks connected to a $(z^{-2} + 2 \cos \varphi \, z^{-1} + 1)$-block with $\varphi = (2\pi/N) \cdot k$ we take the z-transform of the adjacent block to the former. For this block we have the following expression giving the missing conjugate pairs of zeros in the signal path:

$$z^{-2} - 2 \cos \varphi \, z^{-1} + 1 = (z^{-1} - e^{j\varphi})(z^{-1} - e^{-j\varphi}). \qquad (20)$$

Thus the missing zeros are $\exp(\pm j\varphi)$. We now know that for $n = k$ we can get the cos-transform by connecting an output block $(z^{-1} - \cos \varphi)$ and the sin-transformation is obtained by just letting the output block represent the factor $\sin \varphi$.
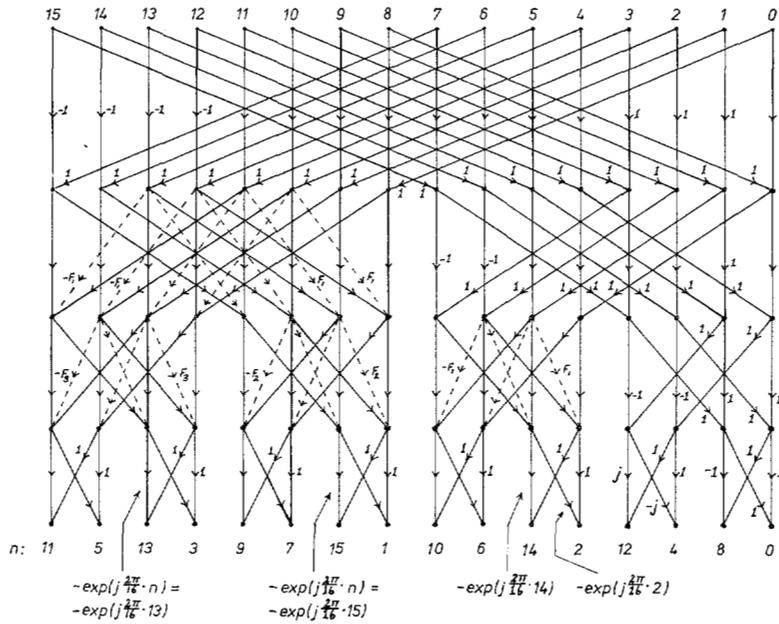
Fig. 9. Complex FFT signal-flow graph corresponding to the filter in Fig. 7; $N = 16$. Branches with gains which are not $+1$ or -1 are shown as dotted lines. For the coefficients $F_1, F_2, \cdots$ see the table Fig. 8.
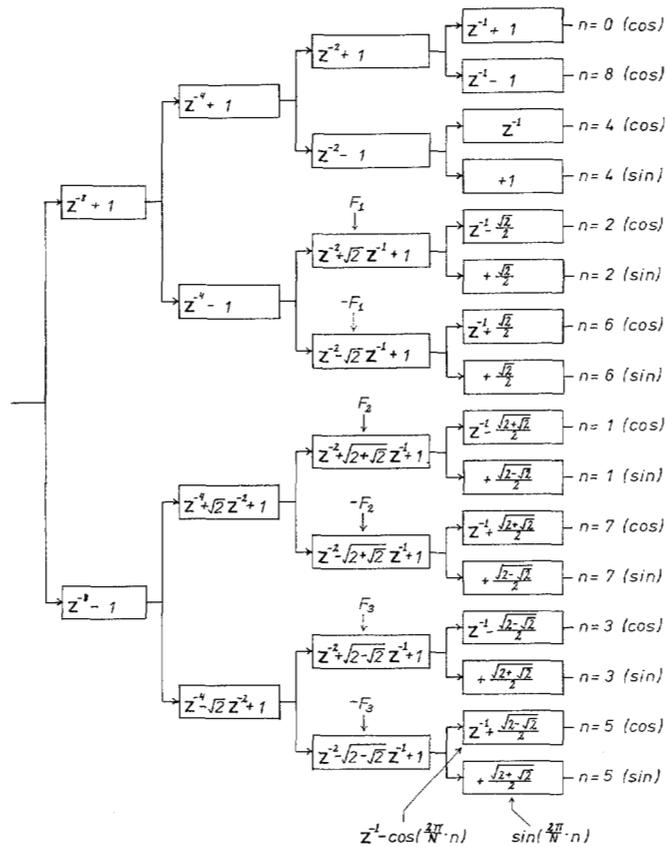

Fig. 10. Filter tree for real signal cos/sin DFT; $N = 16$. All filter coefficients are real.

VIII. THE NUMBER OF MULTIPLIERS FOR THE NEW FFT ALGORITHMS

The FFT algorithm developed in Section VI uses $N \log_2 N$ *arithmetic operations*, which is the same as for the classical FFT.

The classical FFT uses asymptotically $(N/2) \log_2 N$ *complex multiplications*, each multiplication being the product of two complex numbers. The new algorithm uses the same number of multiplications, but each multiplication is now the product of one real and one complex number. If we decompose the multiplications into real multiplications, we thus find a reduction by a factor of 2 for the new algorithm as compared to the conventional; the figures are $N \log_2 N$ versus $2 N \log_2 N$.

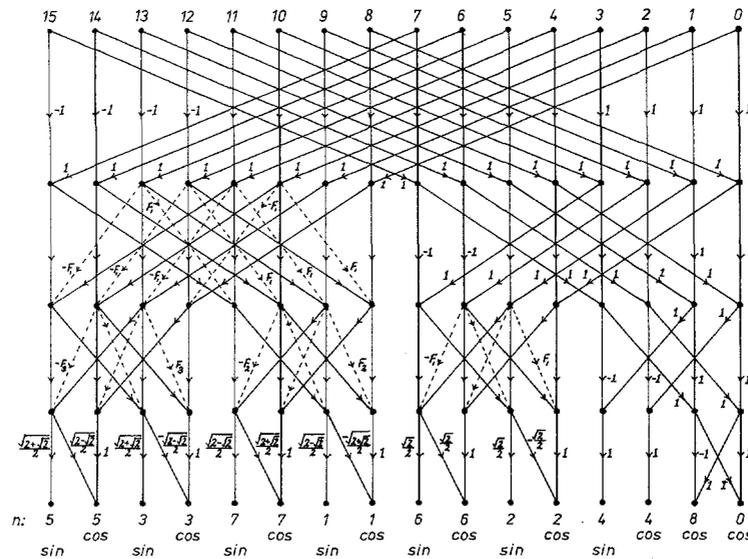For the real FFT discussed in Section VII, we have $(N/2)$

Fig. 11. Real signal cos/sin FFT signal-flow graph corresponding to the filter in Fig. 8; $N = 16$. All filter coefficients are real.

$\log_2 N$ real multiplications, or a reduction by a factor of 2 relative to the algorithm of Section VI. Using the conventional complex FFT for real numbers, it is also possible to obtain a reduction in the number of multiplications by a factor 2.[1] The number of real multiplications will then be $N \log_2 N$ which is still twice as many as for the algorithm of Section VII.

Sometimes one wishes to decompose an FFT for complex numbers into two parts so that the transform of the real part of the N values is calculated before the imaginary part is dealt with. This procedure may be performed in a very simple way by using the algorithm of Section VI twice. Note that this calculation can be made without any increase in the total number of real multiplications.

CONCLUSION

In this paper we have shown how the DFT can be represented by a coefficient saving digital or analog type of delay filter. Furthermore, the direct relationship between such a filter representing the complex DFT and the classical FFT is shown. A type of radix-2 DFT filter is developed which utilizes a minimum of complex coefficients. This filter leads to a new type of complex FFT and to a type of real-signal

---

[1] This implies using a method proposed by Cooley et al. (see Procedure 4 in [7]).

FFT with only real coefficients. The new FFT algorithms only use half as many real multiplications as the classical type of FFT.

REFERENCES

[1] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.*, vol. 19, pp. 297–301, 1965.
[2] W. T. Cochran *et al.*, "What is the fast Fourier transform?" *IEEE Trans. Audio Electroacoust.*, vol. AU-15, pp. 45–55, June 1967.
[3] B. Gold and C. M. Rader, *Digital Processing of Signals.* New York: McGraw-Hill, 1969.
[4] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.
[5] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.
[6] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms.* Reading, MA: Addison-Wesley, 1974.
[7] J. W. Cooley, P. A. W. Lewis, and P. D. Welch, "The fast Fourier transform algorithm: Programming considerations in the calculation of sine, cosine and Laplace transforms," *J. Sound Vib.*, vol. 12, no. 3, pp. 315–337, 1970.