CHAPTER 8

Computation of the Discrete Fourier Transform

Tutorial Problems

1. Solution:
   The resulting trend in the computational complexity of the direct DFT computations is of power 2 of the number of points N.

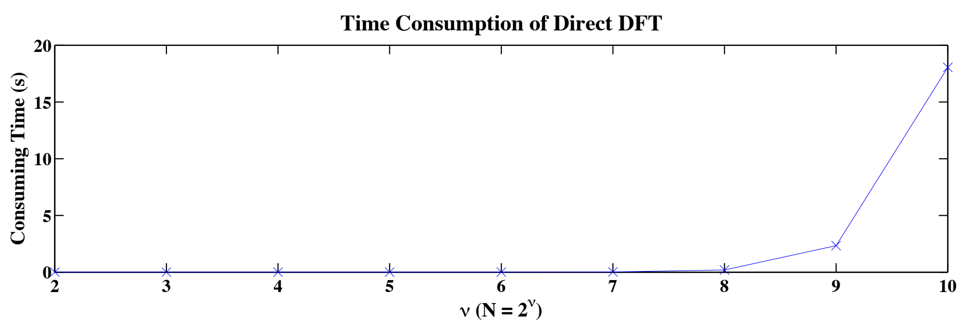


FIGURE 8.1: Plot of computation time for the `dftdirect` function for $N = 2^\nu$ where $2 \leq \nu \leq 10$.

MATLAB script:

```
% P0801: Investigate time consumption using direct DFT
close all; clc
nu = 2:10;
N = 2.^nu;
Ni = length(N);
t = zeros(1,Ni);
```

```
for ii = 1:Ni
    x = randn(1,N(ii)) + j*randn(1,N(ii));
    tic
    X = dftdirect(x);
    t(ii) = toc;
end
% Plot:
hfa = figconfg('P0801a','long');
plot(nu,t,'x-','markersize',12)
xlabel('\nu (N = 2^{\nu})','fontsize',LFS)
ylabel('Consuming Time (s)','fontsize',LFS)
title('Time Consumption of Direct DFT','fontsize',TFS)
```

2. (a) Solution:

The 4-point DIT matrix algorithm is:

$$\left[\begin{array}{c} X[0] \\ X[1] \\ \hline X[2] \\ X[3] \end{array}\right] = \left[\begin{array}{cc|cc} 1 & 1 & 1 & 1 \\ 1 & W_4^2 & W_4^1 & W_4^3 \\ \hline 1 & 1 & W_4^2 & W_4^2 \\ 1 & W_4^2 & W_4^3 & W_4^1 \end{array}\right] \left[\begin{array}{c} x[0] \\ x[2] \\ \hline x[1] \\ x[3] \end{array}\right]$$

which can be simplified as:

$$\left[\begin{array}{c} \boldsymbol{X}_T \\ \hline \boldsymbol{X}_B \end{array}\right] = \left[\begin{array}{c|c} \boldsymbol{W}_2 & \boldsymbol{D}_4\boldsymbol{W}_2 \\ \hline \boldsymbol{W}_2 & -\boldsymbol{D}_4\boldsymbol{W}_2 \end{array}\right] \left[\begin{array}{c} \boldsymbol{x}_E \\ \hline \boldsymbol{x}_O \end{array}\right]$$

where

$$\boldsymbol{W}_2 = \begin{bmatrix} 1 & 1 \\ 1 & W_4^2 \end{bmatrix}, \qquad \boldsymbol{D}_4 = \begin{bmatrix} 1 & 0 \\ 0 & W_4^1 \end{bmatrix}$$

Hence, we conclude as follows:

$$\begin{cases} \boldsymbol{X}_E = \boldsymbol{W}_2 \cdot \boldsymbol{x}_E \\ \boldsymbol{X}_O = \boldsymbol{W}_2 \cdot \boldsymbol{x}_O \end{cases}$$

and

$$\begin{cases} \boldsymbol{X}_T = \boldsymbol{X}_E + \boldsymbol{D}_4\boldsymbol{X}_O \\ \boldsymbol{X}_B = \boldsymbol{X}_E - \boldsymbol{D}_4\boldsymbol{X}_O \end{cases}$$

(b) Solution:

The 4-point DIF matrix algorithm is:

$$\left[\begin{array}{c} X[0] \\ X[2] \\ \hline X[1] \\ X[3] \end{array}\right] = \left[\begin{array}{cc|cc} 1 & 1 & 1 & 1 \\ 1 & W_4^2 & 1 & W_4^2 \\ \hline 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^3 & W_4^2 & W_4^1 \end{array}\right] \left[\begin{array}{c} x[0] \\ x[1] \\ \hline x[2] \\ x[3] \end{array}\right]$$

which can be simplified as:

$$\left[\begin{array}{c} \boldsymbol{X}_E \\ \hline \boldsymbol{X}_O \end{array}\right] = \left[\begin{array}{c|c} \boldsymbol{W}_2 & \boldsymbol{W}_2 \\ \hline \boldsymbol{W}_2\boldsymbol{D}_4 & -\boldsymbol{W}_2\boldsymbol{D}_4 \end{array}\right] \left[\begin{array}{c} \boldsymbol{x}_T \\ \hline \boldsymbol{x}_B \end{array}\right]$$

where

$$\boldsymbol{W}_2 = \begin{bmatrix} 1 & 1 \\ 1 & W_4^2 \end{bmatrix}, \qquad \boldsymbol{D}_4 = \begin{bmatrix} 1 & 0 \\ 0 & W_4^1 \end{bmatrix}$$

Hence, we conclude as follows:

$$\begin{cases} \boldsymbol{\nu} = \boldsymbol{x}_T + \boldsymbol{x}_B \\ \boldsymbol{z} = \boldsymbol{D}_4(\boldsymbol{x}_T - \boldsymbol{x}_B) \end{cases}$$

and

$$\begin{cases} \boldsymbol{X}_E = \boldsymbol{W}_2 \cdot \boldsymbol{\nu} \\ \boldsymbol{X}_O = \boldsymbol{W}_2 \cdot \boldsymbol{z} \end{cases}$$

3. (a) Solution:

Stage I:

The 8-point DFT $X[k]$ can be divided according to even and odd index k, we have

$$\begin{cases} \boldsymbol{v} = \boldsymbol{x}_T + \boldsymbol{x}_B \\ \boldsymbol{w} = \boldsymbol{D}_8(\boldsymbol{x}_T - \boldsymbol{x}_B) \end{cases} \Longrightarrow \begin{cases} \boldsymbol{X}_E = \boldsymbol{W}_4\boldsymbol{v} \\ \boldsymbol{X}_O = \boldsymbol{W}_4\boldsymbol{w} \end{cases}$$

Stage II:

Each 4-point DFT $Y[k]$ can be divided according to even and odd index k, we have

$$\begin{cases} \boldsymbol{p} = \boldsymbol{y}_T + \boldsymbol{y}_B \\ \boldsymbol{q} = \boldsymbol{D}_4(\boldsymbol{y}_T - \boldsymbol{y}_B) \end{cases} \Longrightarrow \begin{cases} \boldsymbol{Y}_E = \boldsymbol{W}_2\boldsymbol{p} \\ \boldsymbol{Y}_O = \boldsymbol{W}_4\boldsymbol{q} \end{cases}$$

Stage III:

Each 2-point DFT $Z[k]$ can be divided according to even and odd index k, we have

$$\begin{cases} m = z[0] + z[1] \\ n = \boldsymbol{D}_2(z[0] - z[1]) \end{cases} \Longrightarrow \begin{cases} Z[0] = m \\ Z[1] = n \end{cases}$$

(b) MATLAB function:

```
function Xdft = difrecur(x)
% Recursive computation of the DFT using divide & conquer
% N should be a power of 2
N = length(x);
Xdft = zeros(1,N);
if N ==1
  Xdft = x;
else
     m = N/2;
     D = exp(-2*pi*sqrt(-1)/N).^(0:m-1);
     v = x(1:N/2)+x(N/2+1:end);
     z = D.*(x(1:N/2)-x(N/2+1:end));
     Xdft(1:2:N) = difrecur(v);
     Xdft(2:2:N) = difrecur(z);
end
```

(c) MATLAB script:

```
% P0803: Testing DIF-FFT function 'difrecur'
close all; clc
x = [1,2,3,4,5,4,3,2];
Xdft = difrecur(x);
X_ref = fft(x);
```

4. MATLAB script:

```
% P0804: Investigate Decimation-in-time procedure
close all; clc
x = [1 2 3 4 5 4 3 2];
N = length(x);
%% Part (a):
a = x(1:2:N);
A = fft(a);
%% Part (b):
b = x(2:2:N);
B = fft(b);
%% Part (c):
W = exp(-j*2*pi/N).^(0:N/2-1);
temp = W.*B;
Xdft = zeros(1,N);
Xdft(1:N/2) = A + temp;
```

```
Xdft(N/2+1:N) = A - temp;
%% Part (d):
X_ref = fft(x);
```

5. (a) Solution:
   Since, $q = 1$, we have

$$W_N^{q\ell} = W_N^{\ell}, \qquad 0 \leq \ell \leq 8$$

   The number of complex multiplications is:

$$1 + 2 + \cdots + 7 = 28$$

   (b) Solution:
   Using the recursion formula, the number of complex multiplications is 7.

6. Proof:
   The two equations are repeated as follows:

$$X[2k] = \sum_{n=0}^{N/2-1} \left(x[n] + x[n + \frac{N}{2}] \right) W_{\frac{N}{2}}^{kn}, \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \quad (8.36)$$

$$X[2k+1] = \sum_{n=0}^{N/2-1} \left(x[n] - x[n + \frac{N}{2}] \right) W_N^n W_{\frac{N}{2}}^{kn}, \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \quad (8.37)$$

   Equation (8.37) can be derived as:

$$\begin{aligned}
X[2k+1] &= \sum_{n=0}^{N-1} x[n] W_N^{(2k+1)n} \\
&= \sum_{n=0}^{\frac{N}{2}-1} x[n] W_N^{(2k+1)n} + \sum_{n=0}^{\frac{N}{2}-1} x[n + \frac{N}{2}] W_N^{(2k+1)(n+\frac{N}{2})} \\
&= \sum_{n=0}^{\frac{N}{2}-1} x[n] W_N^n W_N^{2kn} + \sum_{n=0}^{\frac{N}{2}-1} x[n + \frac{N}{2}](-W_N^n) W_N^{2kn} \\
&= \sum_{n=0}^{N/2-1} \left(x[n] - x[n + \frac{N}{2}] \right) W_N^n W_{\frac{N}{2}}^{kn}, \quad k = 0, 1, \ldots, \frac{N}{2} - 1
\end{aligned}$$

7. (a) Solution:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \tag{8.1}$$

$$\begin{aligned} X[k] &= \sum_{n=0}^{N/2-1} x[n] W_N^{kn} + \sum_{n=0}^{N/2-1} x[n+\frac{N}{2}] W_N^{k(n+\frac{N}{2})} \\ &= \sum_{n=0}^{N/2-1} \left(x[n] + x[n+\frac{N}{2}] W_2^k \right) W_N^{kn} \end{aligned}$$

(b) Solution:
If $k = 2m,\ m = 0, 1, \ldots, \frac{N}{2} - 1$, we have

$$X[k] = X[2m] = \sum_{n=0}^{N/2-1} \left(x[n] + x[n+\frac{N}{2}] \right) W_{\frac{N}{2}}^{mn}$$

If $k = 2m+1,\ m = 0, 1, \ldots, \frac{N}{2} - 1$, we have

$$X[k] = X[2m+1] = \sum_{n=0}^{N/2-1} \left(x[n] - x[n+\frac{N}{2}] \right) W_N^n W_{\frac{N}{2}}^{mn}$$

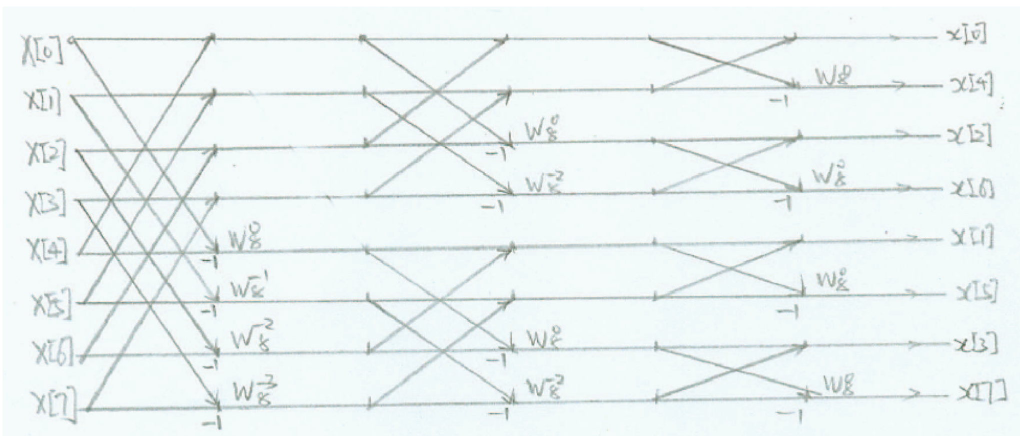
(c) Solution:
The above equations are exactly the same to the DIF FFT algorithm described in the context if we replace the variable m by k.

8. MATLAB function:

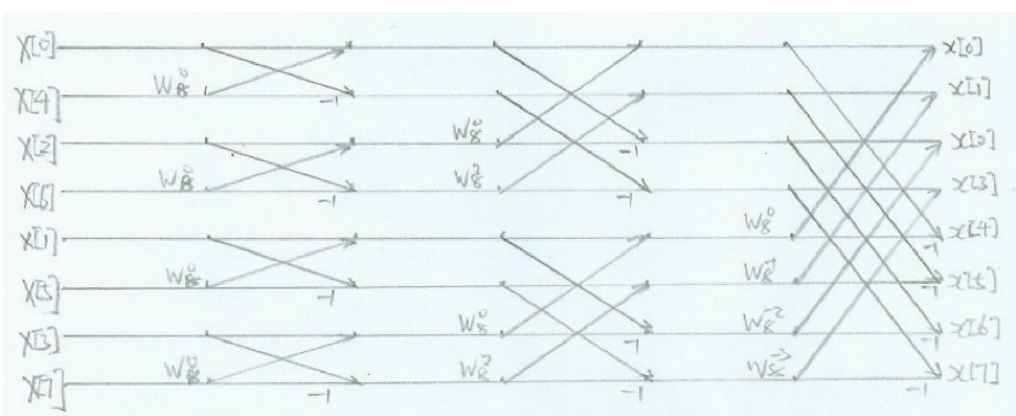
```
function x=fftdifr2(x)
% DIF Radix-2 FFT Algorithm
N=length(x); nu=log2(N);
for m=nu:-1:1;
    L=2^m;
    L2=L/2;
    for ir=1:L2;
        W=exp(-i*2*pi*(ir-1)/L);
        for it=ir:L:N;
            ib=it+L2;
            temp=x(it)+x(ib);
```

```
            x(ib)=x(it)-x(ib);
            x(ib)=x(ib)*W;
            x(it)=temp;
        end
    end
end
x = bitrevorder(x);
```

9. (a) See graph below.



(b) See graph below.



10. (a) Solution:

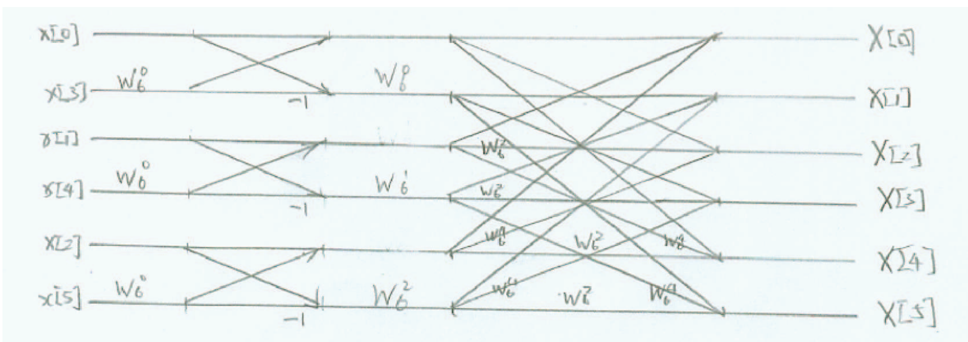
The number of complex multiplications is:

$$3 + 2 \times 6 = 15$$

The number of complex addition is:

$$6 + 2 \times 6 = 18$$

Hence, the number of real multiplication is 60 and the number of real addition is 54.



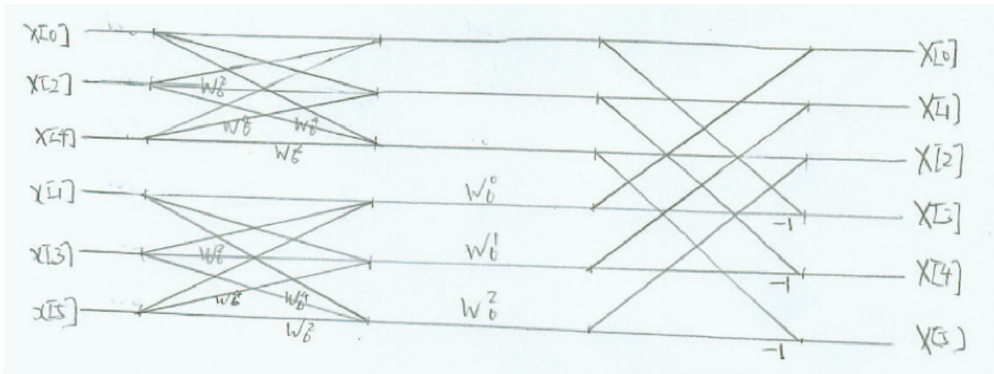
(b) Solution:
The number of complex multiplications is:

$$2 \times 2 \times 3 + 3 = 15$$

The number of complex addition is:

$$2 \times 6 + 6 = 18$$

Hence, the number of real multiplication is 60 and the number of real addition is 54.



11. (a) Proof:

$$X[2k] = \sum_{n=0}^{N/2-1} \left(x[n] + x[n + \frac{N}{2}] \right) W_{\frac{N}{2}}^{kn}, \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \tag{8.36}$$

(b) Proof:

$$X[2k+1] = \sum_{n=0}^{N/2-1} \left(x[n] - x[n + \frac{N}{2}] \right) W_N^n W_{\frac{N}{2}}^{kn}, \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \tag{8.37}$$

12. (a) Solution:

$$
\begin{aligned}
X[k] &= \sum_{n=0}^{14} x[n]W_{15}^{kn} = \sum_{m=0}^{2} x[5m]W_{15}^{5mk} + \sum_{m=0}^{2} x[5m+1]W_{15}^{(5m+1)k} \\
&= \sum_{m=0}^{2} x[5m+2]W_{15}^{(5m+2)k} + \sum_{m=0}^{2} x[5m+3]W_{15}^{(5m+3)k} \\
&\quad + \sum_{m=0}^{2} x[5m+4]W_{15}^{(5m+4)k} \\
&= \left(\sum_{m=0}^{2} x[5m]W_{3}^{km}\right) + \left(\sum_{m=0}^{2} x[5m+1]W_{3}^{km}\right) W_{15}^{k} \\
&\quad + \left(\sum_{m=0}^{2} x[5m+2]W_{3}^{km}\right) W_{15}^{2k} + \left(\sum_{m=0}^{2} x[5m+3]W_{3}^{km}\right) W_{15}^{3k} \\
&\quad + \left(\sum_{m=0}^{2} x[5m+4]W_{3}^{km}\right) W_{15}^{4k}
\end{aligned}
$$

If we define that

$$
\begin{cases}
A[k] = \sum_{m=0}^{2} x[5m]W_3^{km}, \\
B[k] = \sum_{m=0}^{2} x[5m+1]W_3^{km}, \\
C[k] = \sum_{m=0}^{2} x[5m+2]W_3^{km}, \quad k = 0, 1, 2 \\
D[k] = \sum_{m=0}^{2} x[5m+3]W_3^{km}, \\
E[k] = \sum_{m=0}^{2} x[5m+4]W_3^{km}.
\end{cases}
$$

We have

$$
\begin{aligned}
X[k] &= A[k] + B[k]W_{15}^{k} + C[k]W_{15}^{2k} + D[k]W_{15}^{3k} + E[k]W_{15}^{4k} \\
X[k+3] &= A[k] + B[k]W_{15}^{k}W_{15}^{3} + C[k]W_{15}^{2k}W_{15}^{6} + D[k]W_{15}^{3k}W_{15}^{9} + E[k]W_{15}^{4k}W_{15}^{12} \\
X[k+6] &= A[k] + B[k]W_{15}^{k}W_{15}^{6} + C[k]W_{15}^{2k}W_{15}^{12} + D[k]W_{15}^{3k}W_{15}^{3} + E[k]W_{15}^{4k}W_{15}^{9} \\
X[k+9] &= A[k] + B[k]W_{15}^{k}W_{15}^{9} + C[k]W_{15}^{2k}W_{15}^{3} + D[k]W_{15}^{3k}W_{15}^{12} + E[k]W_{15}^{4k}W_{15}^{6} \\
X[k+12] &= A[k] + B[k]W_{15}^{k}W_{15}^{12} + C[k]W_{15}^{2k}W_{15}^{9} + D[k]W_{15}^{3k}W_{15}^{6} + E[k]W_{15}^{4k}W_{15}^{3}
\end{aligned}
$$

(b) Solution:

$$
\begin{aligned}
X[k] &= \sum_{n=0}^{14} x[n]W_{15}^{nk} \\
&= \sum_{m=0}^{4} x[3m]W_{15}^{(3m)k} + \sum_{m=0}^{4} x[3m+1]W_{15}^{(3m+1)k} + \sum_{m=0}^{4} x[3m+2]W_{15}^{(3m+2)k} \\
&= \left(\sum_{m=0}^{4} x[3m]W_{5}^{km}\right) + \left(\sum_{m=0}^{4} x[3m+1]W_{5}^{km}\right) W_{15}^{k} \\
&\quad + \left(\sum_{m=0}^{4} x[3m+2]W_{5}^{km}\right) W_{15}^{2k}
\end{aligned}
$$

If we define that

$$
\begin{cases}
A[k] = \sum_{m=0}^{4} x[3m]W_5^{km}, \\
B[k] = \sum_{m=0}^{4} x[3m+1]W_5^{km}, \quad k = 0, 1, 2, 3, 4 \\
C[k] = \sum_{m=0}^{4} x[3m+2]W_5^{km}.
\end{cases}
$$

We conclude

$$
\begin{aligned}
X[k] &= A[k] + B[k]W_{15}^{k} + C[k]W_{15}^{2k} \\
X[k+5] &= A[k] + B[k]W_{15}^{k}W_{15}^{5} + C[k]W_{15}^{2k}W_{15}^{10} \\
X[k+10] &= A[k] + B[k]W_{15}^{k}W_{15}^{10} + C[k]W_{15}^{2k}W_{15}^{5}
\end{aligned}
$$

(c) Solution:
For part (a), the number of complex multiplication is:

$$5 \times 2 \times 3 + 4 \times 15 = 90$$

The number of complex addition is:

$$2 \times 15 + 4 \times 15 = 90$$

For part (b), the number of complex multiplication is:

$$3 \times 4 \times 5 + 2 \times 15 = 90$$

The number of complex addition is:

$$4 \times 15 + 2 \times 15 = 90$$

13. (a) Solution:

$$
\begin{aligned}
X[k] &= \sum_{n=0}^{15} x[n]W_{16}^{kn} = \sum_{m=0}^{3} x[4m]W_{16}^{k(4m)} + \sum_{m=0}^{3} x[4m+1]W_{16}^{k(4m+1)} \\
&\quad + \sum_{m=0}^{3} x[4m+2]W_{16}^{k(4m+2)} + \sum_{m=0}^{3} x[4m+3]W_{16}^{k(4m+3)} \\
&= \left(\sum_{m=0}^{4} x[4m]W_{4}^{km}\right) + \left(\sum_{m=0}^{4} x[4m+1]W_{4}^{km}\right) W_{16}^{k} \\
&\quad + \left(\sum_{m=0}^{4} x[4m+2]W_{4}^{km}\right) W_{16}^{2k} + \left(\sum_{m=0}^{4} x[4m+3]W_{4}^{km}\right) W_{16}^{3k}
\end{aligned}
$$

If we define that

$$
\begin{cases}
A[k] = \sum_{m=0}^{3} x[4m]W_4^{km}, & k = 0,1,2,3 \\
B[k] = \sum_{m=0}^{3} x[4m+1]W_4^{km}, & k = 0,1,2,3 \\
C[k] = \sum_{m=0}^{3} x[4m+2]W_4^{km}, & k = 0,1,2,3 \\
D[k] = \sum_{m=0}^{3} x[4m+3]W_4^{km}, & k = 0,1,2,3
\end{cases}
$$

We conclude that

$$
\begin{aligned}
X[k] &= A[k] + B[k]W_{16}^{k} + C[k]W_{16}^{2k} + D[k]W_{16}^{3k}, \quad k = 0,1,2,3 \\
X[k+4] &= A[k] + B[k]W_{16}^{k}W_{16}^{4} + C[k]W_{16}^{2k}W_{16}^{8} + D[k]W_{16}^{3k}W_{16}^{12}, \quad k = 0,1,2,3 \\
X[k+8] &= A[k] + B[k]W_{16}^{k}W_{16}^{8} + C[k]W_{16}^{2k}W_{16}^{0} + D[k]W_{16}^{3k}W_{16}^{8}, \quad k = 0,1,2,3 \\
X[k+12] &= A[k] + B[k]W_{16}^{k}W_{16}^{12} + C[k]W_{16}^{2k}W_{16}^{8} + D[k]W_{16}^{3k}W_{16}^{4}, \quad k = 0,1,2,3
\end{aligned}
$$

(b) Solution:
The total number of complex multiplications to implement the radix-4 FFT is:

$$2 \times 16 + 2 \times 16 = 64$$

The total number of complex additions to implement the radix-4 FFT is:

$$3 \times 16 + 3 \times 16 = 96$$

(c) Solution:
The number of complex multiplications to implement the radix-2 FFT is:

$$4 \times 8 = 32$$

which is two times the number of complex multiplications in radix-4 FFT.
Since, in the radix-4 algorithm, each complex multiplication only requires two real multiplication while in general the complex multiplication in radix-2 requires four real multiplications, the number of multiplications are reduced by half.

14. (a) Proof:

$$X[n] = X(\mathrm{e}^{\mathrm{j}\omega_n}) \triangleq \sum_{k=0}^{N-1} g[k]W^{nk} \tag{8.67}$$

$$g[n] \triangleq x[n]\mathrm{e}^{-\mathrm{j}\omega_L n}, \quad \text{and} \quad W = \mathrm{e}^{-\mathrm{j}\delta\omega} \tag{8.68}$$

$$\begin{cases} \mathrm{e}^{\mathrm{j}\omega_L} \to R\mathrm{e}^{\mathrm{j}\omega_L} & \Longrightarrow \quad g[n] = x[n]\left(\frac{1}{R}\mathrm{e}^{-\mathrm{j}\omega_L}\right)^n \\ \mathrm{e}^{\mathrm{j}\delta\omega} \to r\mathrm{e}^{\mathrm{j}\delta\omega} & \Longrightarrow \quad W = \frac{1}{r}\mathrm{e}^{-\mathrm{j}\delta\omega} \end{cases} \tag{8.70}$$

$$z_n = \left(R\mathrm{e}^{\mathrm{j}\omega_L}\right)\left(r\mathrm{e}^{\mathrm{j}\delta\omega}\right)^n, \quad 0 \le n \le M \tag{8.71}$$

$$X(z_n) = \left\{\left(g[n]W^{n^2/2}\right) * W^{-n^2/2}\right\} W^{n^2/2} \tag{8.72}$$

$$\begin{aligned}
X[z_n] &= \sum_{k=0}^{N-1} x[k](z_n)^{-k} = \sum_{k=0}^{N-1} x[k]\left[(R\mathrm{e}^{\mathrm{j}\omega_L})(r\mathrm{e}^{\mathrm{j}\delta_\omega})^n\right]^{-k} \\
&= \sum_{k=0}^{N-1}\left[x[k](R\mathrm{e}^{\mathrm{j}\omega_L})^{-k}\right]\left[(r\mathrm{e}^{\mathrm{j}\delta\omega})^{-1}\right]^{nk} \\
&= \sum_{k=0}^{N-1}\left[x[k](R\mathrm{e}^{\mathrm{j}\omega_L})^{-k}\right]\left[(r\mathrm{e}^{\mathrm{j}\delta\omega})^{-1}\right]^{\frac{k^2}{2}}\left[(r\mathrm{e}^{\mathrm{j}\delta\omega})^{-1}\right]^{-\frac{(n-k)^2}{2}}\left[(r\mathrm{e}^{\mathrm{j}\delta\omega})^{-1}\right]^{\frac{n^2}{2}} \\
&= \left[\sum_{k=0}^{N-1}\left(g[k]W^{\frac{k^2}{2}}\right)W^{-\frac{(n-k)^2}{2}}\right]W^{\frac{n^2}{2}} \\
&= \left\{\left(g[n]W^{n^2/2}\right) * W^{-n^2/2}\right\} W^{n^2/2}
\end{aligned}$$

(b) MATLAB function:

```
function [X,w] = czta(x,M,wL,wH,R,r)
% Chirp z-Transform Algorithm (CTA)
%   Given x[n] CZTA computes M z-transform values
%   on the spiral line over wL <= w <= wH
% [X,w] = czta(x,M,wL,wH,R,r)
```

```
Dw = wH-wL; dw = Dw/(M-1); W = exp(-1j*dw)/r;
N = length(x); nx = 0:N-1;
K = max(M,N); n = 0:K; Wn2 = W.^(n.*n/2);
g = x.*R.^(-nx).*exp(-1j*wL*nx).*Wn2(1:N);
nh = -(N-1):M-1; h = W.^(-nh.*nh/2);
y = conv(g,h);
X = y(N:N+M-1).*Wn2(1:M); w = wL:dw:wH;
```

15. (a) Proof:

$$X_n[k] = \sum_{m=0}^{N-1} x_n[m] W_N^{mk} = \sum_{m=0}^{N-1} x_n[n-N+1+m] W_N^{mk}, \quad \begin{cases} n \geq N-1, \\ 0 \leq k \leq N-1 \end{cases} \tag{8.85}$$

$$X_n[k] = \{X_{n-1}[k] + x[n] - x[n-N]\} W_N^{-k} W_N^{-k}, \quad \begin{cases} n \geq N-1, \\ 0 \leq k \leq N-1 \end{cases} \tag{8.86}$$

$$\begin{aligned} X_{n-1}[k] &= \sum_{m=0}^{N-1} x_{n-1}[m] W_N^{mk} = \sum_{m=0}^{N-1} x[n-1-N+1+m] W_N^{mk} \\ &= \sum_{m=0}^{N-1} x[n-N+m] W_N^{mk} = x[n-N] + \sum_{m=1}^{N} x[n-N+m] W_N^{mk} - x[n] \end{aligned}$$

Hence, we can conclude that

$$\begin{aligned} &\sum_{m=1}^{N} x[n-N+m] W_N^{mk} W_N^{-k} = \sum_{m=1}^{N} x[n-N+m] W_N^{(m-1)k} \\ &= \sum_{m=0}^{N-1} x[n-N+m+1] W_N^{mk} = X_n[k] \end{aligned}$$

(b) Solution:

$$x_n[k] = \sum_{m=0}^{N-1} w_{\mathrm{e}}[m] x[n-N+1+m] W_N^{mk} = \sum_{m=0}^{N-1} \lambda^{N-1-m} x[n-N+1+m] W_N^{mk}$$

$$x_{n-1}[k] = \sum_{m=0}^{N-1} \lambda^{N-1-m} x[n-N+m] W_N^{mk}$$

We can summarize a recursive SDFT algorithm, that is

$$X_n[k] = \left\{\lambda^{-1} X_{n-1}[k] + x[n] - \lambda^{N-2} x[n-N]\right\} W_N^{-k}$$