Computation of the DFT of Real Sequences

• In most practical applications, sequences of interest are real
• In such cases, the symmetry properties of the DFT given in Table 5.2 can be exploited to make the DFT computations more efficient

N-Point DFTs of Two Length-N Real Sequences

• Let $g[n]$ and $h[n]$ be two length-$N$ real sequences with $G[k]$ and $H[k]$ denoting their respective $N$-point DFTs
• These two $N$-point DFTs can be computed efficiently using a single $N$-point DFT
• Define a complex length-$N$ sequence
• Hence, $g[n] = \text{Re}\{x[n]\}$ and $h[n] = \text{Im}\{x[n]\}$

Example - We compute the 4-point DFTs of the two real sequences $g[n]$ and $h[n]$ given below

$g[n] = \{1, 2, 0, 1\}$, $h[n] = \{2, 2, 1, 1\}$

Then $x[n] = g[n] + jh[n]$ is given by

$x[n] = \{1 + j2, 2 + j2, j, 1 + j\}$

N-Point DFTs of Two Length-N Real Sequences

• From the above
• Hence

\[
X[k] = \begin{bmatrix}
1 & 1 & 1 & \frac{4 + j6}{2} \\
1 & -j & 1 & \frac{2 + j2}{2} \\
1 & 1 & -1 & \frac{-2}{2} \\
1 & j & -1 & \frac{2}{2}
\end{bmatrix}
\]

\[
X[k] = \begin{bmatrix}
4 - j6 & 2 & -2 & -j2
\end{bmatrix}
\]

\[
X[(4-k)4] = \begin{bmatrix}
4 - j6 & -j2 & -2 & 2
\end{bmatrix}
\]

Therefore

$G[k] = \{4, 1 - j, -2, 1 + j\}$

$H[k] = \{6, 1 - j, 0, 1 + j\}$

verifying the results derived earlier
2N-Point DFT of a Real Sequence Using an N-point DFT

- Let \( v[n] \) be a length-2N real sequence with an 2N-point DFT \( V[k] \)
- Define two length-N real sequences \( g[n] \) and \( h[n] \) as follows:
  \[ g[n] = v[2n], \quad h[n] = v[2n+1], \quad 0 \leq n \leq N \]
- Let \( G[k] \) and \( H[k] \) denote their respective N-point DFTs

\[
V[k] = \sum_{n=0}^{2N-1} v[n]W_{2N}^{nk} = \sum_{n=0}^{N-1} v[2n]W_{2N}^{2nk} + \sum_{n=0}^{N-1} v[2n+1]W_{2N}^{(2n+1)k} = \sum_{n=0}^{N-1} g[n]W_{N}^{nk} + \sum_{n=0}^{N-1} h[n]W_{N}^{nk}W_{2N}^{k} = \sum_{n=0}^{N-1} g[n]W_{N}^{nk} + W_{2N}^{k} \sum_{n=0}^{N-1} h[n]W_{N}^{nk}, \quad 0 \leq k \leq 2N-1
\]

Example - Let us determine the 8-point DFT \( V[k] \) of the length-8 real sequence
\[ \{v[n]\} = \{1 \ 2 \ 2 \ 0 \ 1 \ 1 \ 1 \} \]

\[ V[k] = G[k] + W_{8}^{k}H[k], \quad 0 \leq k \leq 2N-1 \]

We form two length-4 real sequences as follows

\[
g[n] = \{v[2n]\} = \{1 \ 2 \ 0 \ 1\}
h[n] = \{v[2n+1]\} = \{2 \ 2 \ 1 \ 1\}
\]

Now \( V[k] = G[k] + W_{8}^{k}H[k], \quad 0 \leq k \leq 7 \)

- Substituting the values of the 4-point DFTs \( G[k] \) and \( H[k] \) computed earlier we get

\[
V[0] = G[0] + H[0] = 4 + 6 = 10
V[4] = G[0] + W_{8}^{4}H[0] = 4 + e^{-j\pi} \cdot 6 = -2
\]
= (1 - j) + e^{-j5\pi/4}(1 - j) = 1 + j0.4142 \]


= (1 + j) + e^{-j7\pi/4}(1 + j) = 1 + j2.4142 \]

**Linear Convolution of Two Finite-Length Sequences**

- Let \( g[n] \) and \( h[n] \) be two finite-length sequences of length \( N \) and \( M \), respectively
- Denote \( L = N + M - 1 \)
- Define two length-\( L \) sequences

\[
\begin{align*}
g_e[n] &= \begin{cases} 
g[n], & 0 \leq n \leq N - 1 \\
0, & N \leq n \leq L - 1 \end{cases} \\
h_e[n] &= \begin{cases} 
h[n], & 0 \leq n \leq M - 1 \\
0, & M \leq n \leq L - 1 \end{cases}
\end{align*}
\]

- Then

\[ y_C[n] = g[n] \circledast h[n] = y[n] = g_e[n] \circledast h_e[n] \]

- The corresponding implementation scheme is illustrated below

**The Cyclic Prefix**

- We outlined earlier a DFT-based method to perform a linear convolution of a length-\( N \) sequence \( \{g[n]\} \) with a length-\( M \) sequence \( \{h[n]\} \) with \( N > M \)
- To this end, both sequences were zero-padded to lengths \( L = N + M - 1 \)

- Next, the \( L \)-point DFTs of the extended sequences are formed and multiplied sample-wise
- An \( L \)-point inverse DFT of the product sequence leads to the convolution sum \( \{y[n]\} \) of \( \{g[n]\} \) and \( \{h[n]\} \)
The Cyclic Prefix

• In some applications, it is required to compute only a length-$N$ portion of $\{y[n]\}$
• This can be implemented using an $N$-point DFT and IDFT by appending the longer sequence with a subsequence called the cyclic prefix
• We explain the procedure next

Consider two sequences $\{x[n]\}$, and $\{h[n]\}$, with $N > M$.

- The cyclic prefix of $\{x[n]\}$ is given by the length-1 subsequence consisting of the last $M$ samples of $\{x[n]\}$.

Define a new sequence $\{\tilde{x}[n]\}$ obtained by appending $\{x[n]\}$ at the beginning with its cyclic prefix:

$$\tilde{x}[n] = \{x[N-M+1], \ldots, x[N-1], x[0], \ldots, x[N-M], \ldots, x[N-1]\}$$

- The new sequence $\{\tilde{x}[n]\}$, $-M+1 \leq n \leq N-1$ is of length $L = N + M - 1$.

Now $\tilde{x}[n] = x[(n)_N]$, $-M+1 \leq n \leq N-1$.

From the above equation it follows that $\tilde{x}[n-\ell] = x[(n-\ell)_N]$, $-M+1 \leq n-\ell \leq N-1$.

Let $\{y[n]\}$ denote the linear convolution of $\{\tilde{x}[n]\}$ and $\{h[n]\}$, i.e.

$$y[n] = \tilde{x}[n] \ast h[n] = \sum_{\ell=0}^{L-1} \tilde{x}[n-\ell] h[\ell]$$

where $-M+1 \leq n \leq N + M - 2$.

Let $\{h_k[n]\}$ denote a length-$N$ sequence obtained by zero-paddings $\{h[n]\]$ with $N-M$ zeros, i.e.

$$h_k[n] = \begin{cases} h[n], & 0 \leq n \leq M-1 \\ 0, & M \leq n \leq N-1 \end{cases}$$

Let $\{\hat{y}[n]\}$ denote the $N$-point circular convolution of $\{x[n]\}$ and $\{h_k[n]\}$.

The above circular convolution can be computed using the DFT-based method.
The Cyclic Prefix

• Taking the $N$-point DFT of both sides of
  \[
  \hat{y}[n] = \sum_{\ell=0}^{N-1} x[(n-\ell) \mod N] h_{\ell}[n],
  \]
  we arrive at
  \[
  \hat{Y}[k] = X[k] H_{e}[k]
  \]
• In the above equation, $\hat{Y}[k]$, $X[k]$, and $H_{e}[k]$ denote the $N$-point DFTs of $\hat{y}[n]$, $x[n]$, and $h_{\ell}[n]$, respectively

The Cyclic Prefix

• The cyclic prefix plays an important role in multicarrier-based digital communication
• Here, the objective is to recover the length-$N$ input sequence $x[n]$ knowing the output sequence $\hat{y}[n]$ and the length-$M$ impulse response $h[n]$ of the channel

The Cyclic Prefix

• To this end, $x[n]$ is enlarged to a length-$(N+M-1)$ sequence $\hat{x}[n]$ by appending it at the beginning by its last $M-1$ samples as indicated below
  \[\hat{x}[n] = [x[N-M+1], \ldots, x[N-1], x[0], \ldots, x[M-1], x[N-1]]\]
  cyclic prefix original sequence $\{x[n]\}$

The Cyclic Prefix

• In the absence of noise, original input sequence $x[n]$ can be recovered from $\hat{x}[n]$ knowing the channel impulse response $h[n]$ and the output sequence $y[n]$ as follows:
  1) Develop $\hat{y}[n]$ by extracting the middle $N$ samples from $y[n]$
  2) Zero-pad $h[n]$ with $(N-M)$ zeros to generate a length-$N$ sequence $h_{e}[n]$

The Cyclic Prefix

• 3) Form the $N$-point DFT $\hat{Y}[k]$ of $\hat{y}[n]$, and the $N$-point DFT $H_{e}[k]$ of $h_{e}[n]$
• The desired input sequence $x[n]$ is then recovered as indicated below
  \[x[n] = \text{IDFT}\left\{ \frac{\hat{Y}[k]}{H_{e}[k]} \right\}
  \]
  provided none of the samples of $H_{e}[k]$ is zero

• Even though the output sequence $y[n]$ is of length $N+2M-2$, the first and last $M-1$ samples of $y[n]$ do not have to be computed as they are not needed to recover the input sequence $x[n]$
The Cyclic Prefix

- Example – Consider the length-6 sequence
  \( \{x[n]\} = \{-2, 4, 1, -1, 3, 5\}, 0 \leq n \leq 5 \)
  and the length-4 sequence
  \( \{h[n]\} = \{1, -2, 4, -1\}, 0 \leq n \leq 3 \)
- The cyclic prefix of \( \{x[n]\} \) is thus the length-3 sequence consisting of the last 3 samples of \( \{x[n]\} \)

The Cyclic Prefix

- The new sequence \( \{x[n]\} \) is hence given by
  \( \{\hat{x}[n]\} = \{-1, 3, 5, -2, 4, 1, -1, 3, 5\}, \)
  cyclic prefix \( -3 \leq n \leq 5 \)
- The convolution sum \( \{y[n]\} \) of \( \{\hat{x}[n]\} \) and \( \{h[n]\} \) is given by
  \( \{y[n]\} = \{-1, -5, 1, 25, -20, 15, 5, -6, 3, 17, -5\}, \)
  \( -3 \leq n \leq 8 \)

The Cyclic Prefix

- Note: The samples of \( \{\hat{y}[n]\} \) given in the previous slide are precisely the middle 6 samples of \( \{y[n]\} \) given earlier:
  \( \{y[n]\} = \{-1, 5, -5, 1, 25, -20, 15, 5, -6, 3, 17, -5\}, \)
  \( -3 \leq n \leq 8 \)
- Using MATLAB we compute the 6-point DFT \( \{Y[k]\} \) of \( \{\hat{y}[n]\} \) and the 6-point DFT \( H_e[k] \) of \( h[n] \)

The Cyclic Prefix

- Dividing \( \{\hat{y}[k]\} \) by \( H_e[k] \) sample-wise, and then taking the 6-point IDFT of the result we arrive at
  \( \{-20.0, 4.0, 1.0, -1.0, 3.0, 5.0\}, 0 \leq n \leq 5 \)
  which is precisely the desired input sequence

Linear Convolution of a Finite-Length Sequence with an Infinite-Length Sequence

- We next consider the DFT-based implementation of
  \[
  y[n] = \sum_{\ell=0}^{M-1} h[\ell]x[n-\ell] = h[n] \oplus x[n]
  \]
  where \( h[n] \) is a finite-length sequence of length \( M \) and \( x[n] \) is an infinite length (or a finite length sequence of length much greater than \( M \))
Overlap-Add Method

• We first segment \( x[n] \), assumed to be a causal sequence here without any loss of generality, into a set of contiguous finite-length subsequences \( x_m[n] \) of length \( N \) each:

\[
x[n] = \sum_{m=0}^{\infty} x_m[n - mN]
\]

where

\[
x_m[n] = \begin{cases} x[n + mN], & 0 \leq n \leq N - 1 \\ 0, & \text{otherwise} \end{cases}
\]

Thus we can write

\[
y[n] = h[n] \circledast x[n] = \sum_{m=0}^{\infty} y_m[n - mN]
\]

where

\[
y_m[n] = h[n] \circledast x_m[n]
\]

Since \( h[n] \) is of length \( M \) and \( x_m[n] \) is of length \( N \), the linear convolution \( h[n] \circledast x_m[n] \) is of length \( N + M - 1 \).

As a result, the desired linear convolution \( y[n] = h[n] \circledast x[n] \) has been broken up into a sum of infinite number of short-length linear convolutions of length \( N + M - 1 \) each:

\[
y_m[n] = x_m[n] \circledast h[n]
\]

Each of these short convolutions can be implemented using the DFT-based method discussed earlier, where now the DFTs (and the IDFT) are computed on the basis of \( (N + M - 1) \) points.

Thus there is one more subtlety to take care of before we can implement

\[
y[n] = \sum_{m=0}^{\infty} y_m[n - mN]
\]

using the DFT-based approach.

The second short convolution \( y_1[n] = h[n] \circledast x_1[n] \), is also of length \( N + M - 1 \) but is defined for \( N \leq n \leq 2N + M - 2 \).

There is an overlap of \( M - 1 \) samples between these two short linear convolutions.

Likewise, the third short convolution \( y_2[n] = h[n] \circledast x_2[n] \), is also of length \( N + M - 1 \) but is defined for \( 2N \leq n \leq 3N + M - 2 \).

Thus there is an overlap of \( M - 1 \) samples between \( h[n] \circledast x_1[n] \) and \( h[n] \circledast x_2[n] \).

In general, there will be an overlap of \( M - 1 \) samples between the samples of the short convolutions \( h[n] \circledast x_{r-1}[n] \) and \( h[n] \circledast x_r[n] \) for \((r - 1)N \leq n \leq rN + M - 2\).

This process is illustrated in the figure on the next slide for \( M = 5 \) and \( N = 7 \).
Overlap-Add Method

• Therefore, \( y[n] \) obtained by a linear convolution of \( x[n] \) and \( h[n] \) is given by

\[
\begin{align*}
y[n] &= y_0[n], & 0 \leq n \leq 6 \\
y[n] &= y_0[n] + y_1[n - 7], & 7 \leq n \leq 10 \\
y[n] &= y_1[n - 7], & 11 \leq n \leq 13 \\
y[n] &= y_1[n - 7] + y_2[n - 14], & 14 \leq n \leq 17 \\
y[n] &= y_2[n - 14], & 18 \leq n \leq 20 \\
& \vdots \\
\end{align*}
\]

Overlap-Add Method

• The above procedure is called the overlap-add method since the results of the short linear convolutions overlap and the overlapped portions are added to get the correct final result.

• The function `fftfilt` can be used to implement the above method.

Overlap-Add Method

• Program 5_5 illustrates the use of `fftfilt` in the filtering of a noise-corrupted signal using a length-3 moving average filter.

• The plots generated by running this program is shown below.

Overlap-Save Method

• In implementing the overlap-add method using the DFT, we need to compute two \((N + M - 1)\)-point DFTs and one \((N + M - 1)\)-point IDFT since the overall linear convolution was expressed as a sum of short-length linear convolutions of length \((N + M - 1)\) each.

• It is possible to implement the overall linear convolution by performing instead circular convolution of length shorter than \((N + M - 1)\).
Overlap-Save Method

• To this end, it is necessary to segment $x[n]$ into overlapping blocks $x_m[n]$, keep the terms of the circular convolution of $h[n]$ with $x_m[n]$ that corresponds to the terms obtained by a linear convolution of $h[n]$ and $x_m[n]$, and throw away the other parts of the circular convolution.

Overlap-Save Method

• To understand the correspondence between the linear and circular convolutions, consider a length-4 sequence $x[n]$ and a length-3 sequence $h[n]$. Let $y_L[n]$ denote the result of a linear convolution of $x[n]$ with $h[n]$. The six samples of $y_L[n]$ are given by

\[
\begin{align*}
y_L[0] &= h[0]x[0] \\
\end{align*}
\]

Overlap-Save Method

• If we append $h[n]$ with a single zero-valued sample and convert it into a length-4 sequence $h_C[n]$, the 4-point circular convolution $y_C[n]$ of $h_C[n]$ and $x[n]$ is given by

\[
\begin{align*}
\end{align*}
\]

Overlap-Save Method

• General case: $N$-point circular convolution of a length-$M$ sequence $h[n]$ with a length-$N$ sequence $x[n]$ with $N > M$.

• First $M-1$ samples of the circular convolution are incorrect and are rejected.

• Remaining $N-M+1$ samples correspond to the correct samples of the linear convolution of $h[n]$ with $x[n]$. 

Overlap-Save Method
Overlap-Save Method

• Now, consider an infinitely long or very long sequence $x[n]$
• Break it up as a collection of smaller length (length-4) overlapping sequences $x_m[n]$ as $x_m[n] = x[n+2m]$,

$$0 \leq n \leq 3,
0 \leq m \leq \infty$$

• Next, form

$$w_m[n] = h[n] \oplus x_m[n]$$

Overlap-Save Method

• Or, equivalently,

$$w_m[0] = h[0]x_m[0] + h[1]x_m[3] + h[2]x_m[2] \quad \leftarrow \text{Reject}$$
$$w_m[1] = h[0]x_m[1] + h[1]x_m[0] + h[2]x_m[3] \quad \leftarrow \text{Reject}$$
$$w_m[2] = h[0]x_m[2] + h[1]x_m[1] + h[2]x_m[0] \quad \leftarrow \text{Save}$$

• Computing the above for $m = 0, 1, 2, 3, \ldots$, and substituting the values of $x_m[n]$ we arrive at

Overlap-Save Method

• It should be noted that to determine $y[0]$ and $y[1]$, we need to form $x_{-1}[n]$:

$$x_{-1}[0] = 0,
 x_{-1}[1] = 0,$$

$$x_{-1}[2] = x[0],
 x_{-1}[3] = x[1]$$

and compute $w_{-1}[n] = h[n] \oplus x_{-1}[n]$ for $0 \leq n \leq 3$

reject $w_{-1}[0]$ and $w_{-1}[1]$, and save $w_{-1}[2] = y[0]$ and $w_{-1}[3] = y[1]$

Overlap-Save Method

• General Case: Let $h[n]$ be a length-$N$ sequence

• Let $x_m[n]$ denote the $m$-th section of an infinitely long sequence $x[n]$ of length $N$ and defined by

$$x_m[n] = x[n + m(N - m + 1)], \quad 0 \leq n \leq N - 1$$

with $M < N$
Overlap-Save Method

• Let \( w_m[n] = h[n] \otimes x_m[n] \)
• Then, we reject the first \( M - 1 \) samples of \( w_m[n] \) and “abut” the remaining \( N - M + 1 \) samples of \( w_m[n] \) to form \( y_L[n] \), the linear convolution of \( h[n] \) and \( x[n] \)
• If \( y_m[n] \) denotes the saved portion of \( w_m[n] \), i.e.
\[
y_m[n] = \begin{cases} 
0, & 0 \leq n \leq M - 2 \\
 w_m[n], & M - 1 \leq n \leq N - 2
\end{cases}
\]

Overlap-Save Method

• Then
\[
y_L[n + m(N - M + 1)] = y_m[n], \quad M - 1 \leq n \leq N - 1
\]
• The approach is called overlap-save method since the input is segmented into overlapping sections and parts of the results of the circular convolutions are saved and abutted to determine the linear convolution result

Overlap-Save Method

• Process is illustrated next

Overlap-Save Method

• Process is illustrated next