Solutions: Homework Set # 2

Problem 1

We are looking for $N \in \mathbb{N}$ such that for all $n$, $\tilde{x}[n+N] = \tilde{x}[n]$. This means that we need to find $N$ for which

$$2 + \sin \left( \frac{2\pi}{5} (n + N) \right) + \cos \left( \frac{3\pi}{2} (n + N) \right) = 2 + \sin \left( \frac{2\pi}{5} n \right) + \cos \left( \frac{3\pi}{2} n \right).$$

We have that $\sin \left( \frac{2\pi}{5} (n + N_1) \right) = \sin \left( \frac{2\pi}{5} n \right)$ for $N_1 = 5$ and $\sin \left( \frac{2\pi}{5} (n + N_2) \right) = \sin \left( \frac{3\pi}{2} n \right)$ for $N_2 = 4$. If we take $N$ equal to the least common multiple of $N_1$ and $N_2$ we satisfy (1). Hence $N = 20$.

Problem 2

(b) This is shown using the fact that $W_N^L x[n] \xrightarrow{\text{DFT}} X[x + L]$ and the linearity of the DFT.

(c) If $x[n]$ is even, i.e., $x[n] = x[N - n]$, then $X[k] = X[-k]$. Indeed,

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi}{N} kn}$$
$$= \sum_{n=0}^{N-1} x[N - n] e^{-j \frac{2\pi}{N} kn}$$
$$= \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi}{N} k(N-n)}$$
$$= \sum_{n=0}^{N-1} x[n] e^{j \frac{2\pi}{N} kn}$$
$$= X[-k].$$

Thus,

$$Y[-k - 1] = X_1[-k] + X_2[-k - 1]$$
$$= X_1[k] + X_2[k + 1].$$

(d) Let $X_1[0] = \sum_{n=0}^{N-1} x_1[n]$ and $X_2[0] = \sum_{n=0}^{N-1} x_2[n]$. The recursion is then

$$X_2[k + 1] = Y[N - (k + 1)] - X_1[k]$$
$$X_1[k + 1] = Y[k] - X_2[k].$$
Problem 3

(a) The period is $N = 20$, and we have
\[
\hat{X}[k] = \sum_{n=0}^{19} e^{-2\pi n} e^{-j \frac{2\pi}{20} kn} \\
= \frac{1 - e^{-(40+j2\pi k)}}{1 - e^{-(2+j \frac{2\pi}{20})}} \\
= \frac{1 - e^{-40}}{e^{-(1+j \frac{2\pi}{20})} (e^{1+j \frac{2\pi}{20}} - e^{-(1+j \frac{2\pi}{20})}), \quad k = 0, 1, \ldots, 19
\]

(b) Here the period is $N = 2$, so
\[
\hat{X}[k] = \sum_{n=0}^{1} x[n] e^{-j \frac{2\pi}{8} kn} \\
= 1 - e^{-j\pi k} \\
= \begin{cases} 
0, & k \text{ even} \\
2, & k \text{ odd.}
\end{cases}
\]

Problem 4 (MATLAB and FFT)

(b) Using $\sin x = (e^{jx} - e^{-jx})/2j$ and $N = 8$, we have
\[
x[n] = \frac{1}{2j} \left( e^{\frac{2\pi n}{8}} - e^{-\frac{2\pi n}{8}} \right) \\
= \frac{1}{2j} (W_8^{-n} - W_8^n),
\]
so the DFT is given by (shift property)
\[
X[k] = \frac{8}{2j} (\delta[k - 1] - \delta[k - 7]), \quad k = 0, \ldots, 7.
\]

Computing the magnitude of the DFT in MATLAB results in the plot shown on Figure 1.

(Alternatively, if you have used $N = 16$, then the DFT is $X[k] = (16/2j) (\delta[k - 2] - \delta[k - 14])$, $k = 0, \ldots, 15$.)

Problem 5

(a) cf. Figure 2

(b) cf. Figure 3

(c) The DFTs of $x_1[k]$ and $x_2[k]$ are
\[
X_1[k] = \sum_{n=0}^{N-1} x_1[n] e^{-j \frac{2\pi}{20} kn}, \quad k = 0, \ldots, N - 1
\]
Figure 1: Magnitude plot for Problem 4.

Figure 2: Magnitude and angle plot for Problem 5(a).
and

\[ X_2[k] = \sum_{n=0}^{N-1} x_2[n]e^{-j\frac{2\pi}{2N}kn}, \quad k = 0, \ldots, 2N - 1. \]

Writing

\[ X_1[k] = \sum_{n=0}^{N-1} x_1[n]e^{-j\frac{2\pi}{2N}2kn} = X_2[2k], \]

we see that \( X_1[k] \) is a decimated version of \( X_2[k] \), with a decimation factor of 2.

(d) cf. Figure 4

(e) We have

\[
X_3[k] = \sum_{n=0}^{2N-1} x_3[n]e^{-j\frac{2\pi}{2N}kn}
\]
\[
= \sum_{n=0}^{N-1} x_1[n]e^{-j\frac{2\pi}{2N}kn} + \sum_{n=N}^{2N-1} x_1[n - N]e^{-j\frac{2\pi}{2N}kn}
\]
\[
= \sum_{n=0}^{N-1} x_1[n]e^{-j\frac{2\pi}{2N}kn} + \sum_{n'=0}^{N-1} x_1[n']e^{-j\frac{2\pi}{2N}k(n'+N)}
\]
\[
= \left(1 + e^{-j\frac{2\pi}{2N}kN}\right) \sum_{n=0}^{N-1} x_1[n]e^{-j\frac{2\pi}{2N}kn}
\]
\[
= \begin{cases} 
2X_1[k/2], & k \text{ even} \\
0, & k \text{ odd},
\end{cases}
\]
which explains the figure. (The non-zero angle values for odd \(k\) are due to rounding errors).

**Problem 6** (DFT AND DTFT)

(a) The following listing shows a possible way to compute the DTFT.

```matlab
function X = dtft(x, w, n0)
% DTFT Compute discrete-time Fourier transform
% X = DTFT(x, w, n0) computes the discrete time Fourier transform of the
% sequence x, evaluated at the frequencies indicated by w. Values of w
% must be between 0 and 2 pi. n0 indicates the index of the first sample
% of x; if unspecified it is assumed to be 0.

% Default arguments
if nargin < 3
    n0 = 0;
end

% Indices of the sequence
n = n0:n0 + length(x) - 1;

% Allocate output vector
X = zeros(1, length(w));

% Loop for summation
```
Figure 5: Plots for Problem 6. You can see that the DFT is a sampled version of the DTFT.

for k = 1:length(n)
    % Get current index
    l = n(k);
    % Add term to DTFT sum
    X = X + x(k) * exp(-j*w*l);
end

(b), (c) The resulting plots are shown on Figure 5.