Assignment No. 8: **Inverse Systems**

Reg. No. : 2016-EE-189

Task 1

Statement:

Consider a system $H(z)$ which has the following poles and zeros:

- 4 poles at $z = 0$
- 4 zeros at $z = 0.9e^{j\,0.6\pi},\ 0.9e^{-j\,0.6\pi},\ 1.25e^{j\,0.8\pi},\ 1.25e^{-j\,0.8\pi}$

Consider a signal $x[n]$,

$$x[n] \;=\; \mathrm{sinc}\left(\frac{\pi}{16}(n-50)\right)\cos(\omega_c\, n)$$

Question (1):

Draw the magnitude, phase and group delay response of the system using MATLAB. Do not use MATLAB's builtin functions.

Answer (1):

The system $H(z)$ described above will have the following expression,

$$H(z) \;=\; \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-1.25e^{j\,0.8\pi})(z-1.25e^{-j\,0.8\pi})}{z^4}$$

$$H(z) \;=\; (1-0.9e^{j\,0.6\pi}\,z^{-1})(1-0.9e^{-j\,0.6\pi}\,z^{-1})(1-1.25e^{j\,0.8\pi}\,z^{-1})(1-1.25e^{-j\,0.8\pi}\,z^{-1})$$

Note that it is of the form,

$$H(z) \;=\; \left(1-a_1\,z^{-1}\right)\left(1-a_1{}^*\,z^{-1}\right)\left(1-a_2\,z^{-1}\right)\left(1-a_2{}^*\,z^{-1}\right)$$

where $a_1 = 0.9e^{j\,0.6\pi}$ and $a_2 = 1.25e^{j\,0.8\pi}$. All of its 4 poles are at $z = 0$; 2 zeros (conjugate pair) are inside the unit-circle, and the other 2 (also conjugate pair) are outside the unit-circle.

The code, along with results (inline), to compute & plot various responses from $H(z)$ expression above, starts on the next page.

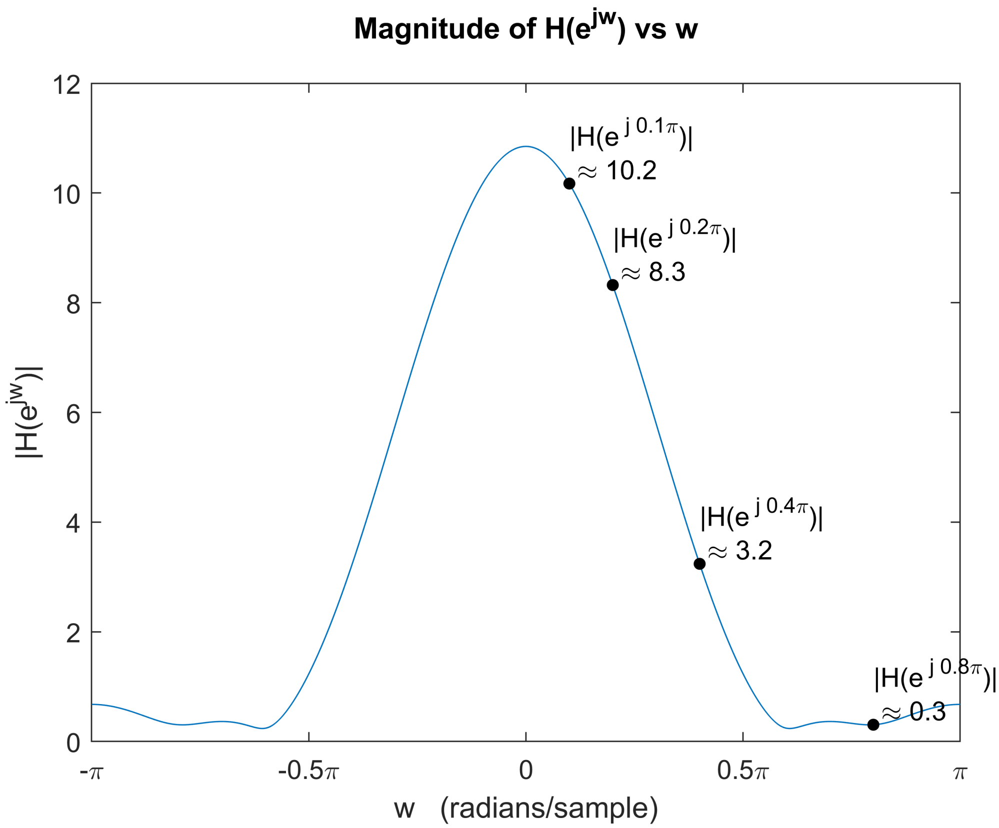
Note: The various functions used in the code ($\mathrm{markOnPlot}()$, $\mathrm{plot2SeqDTFT}()$, $\mathrm{plot3SeqDTFT}()$) have been defined at the end of report.

```
% <task> plot magnitude and phase response of given H(z)

% define z-domain expression (symbolically)
syms z;
a1 = 0.9*exp(1i*0.6*pi);
a2 = 1.25*exp(1i*0.8*pi);
Hz_N = (z-a1)*(z-conj(a1))*(z-a2)*(z-conj(a2));     % numerator of H(z)
Hz_D = z^4;                                          % denominator of H(z)
Hz = Hz_N/Hz_D;

% display H(z) expression
disp('H(z) = '); disp(vpa(Hz, 4))
```

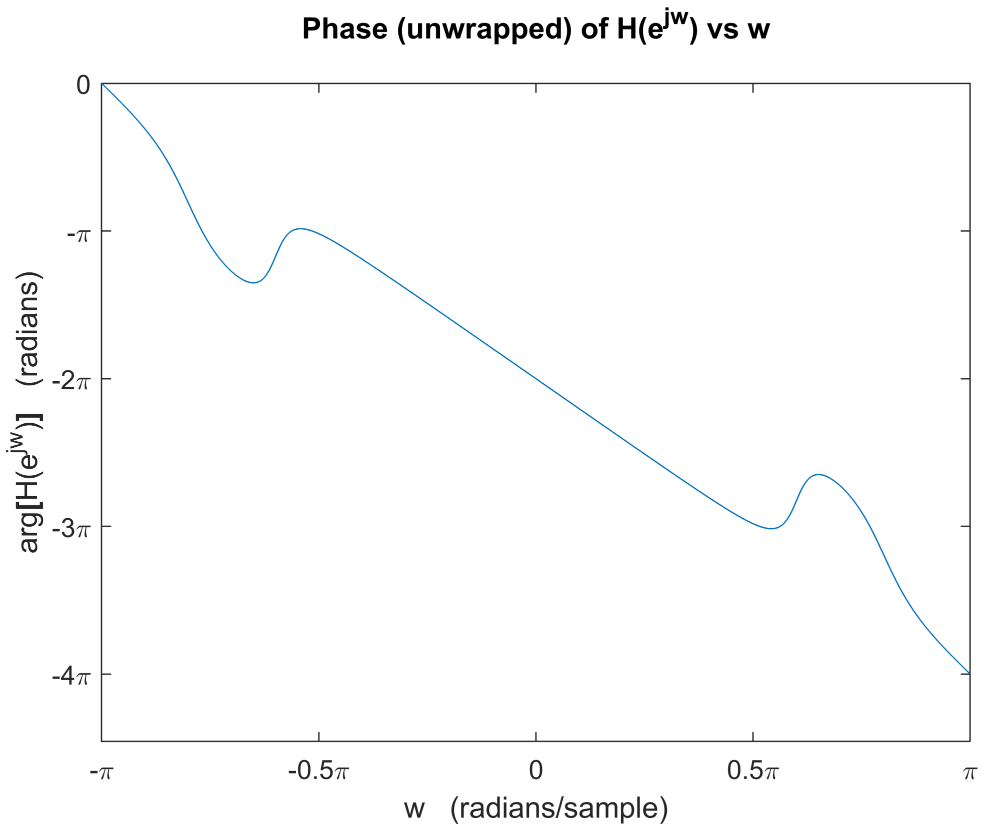
```
H(z) =
```

$$\frac{(z+1.011-0.7347\,\mathrm{i})\,(z+1.011+0.7347\,\mathrm{i})\,(z+0.2781-0.856\,\mathrm{i})\,(z+0.2781+0.856\,\mathrm{i})}{z^4}$$

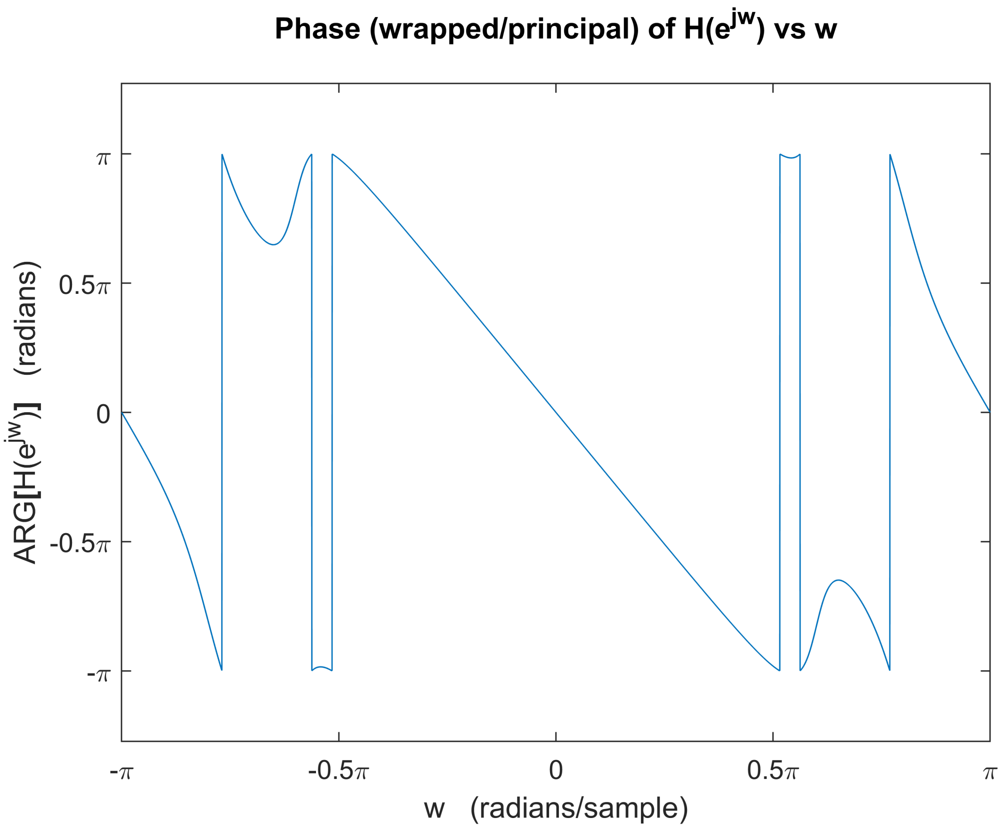
```
% obtain frequency response H(w) expression from H(z)
% by substituting z = e^(jw)
syms w_;
Hw = subs(Hz, {z}, {exp(1i*w_)});

% compute frequency response H(w) for w = [-pi, pi]
dw = 0.001;
w = -pi:dw:pi;
Hw = subs(Hw, {w_}, {w});
Hw = double(Hw);

% obtain magnitude and phase of H(w)
Hw_mag = abs(Hw);
Hw_arg = phase(Hw);                  % unwrapped phase (in radians)
Hw_ARG = angle(Hw);                  % wrapped phase (in radians)

% frequencies to be noted/marked on the response plots
wvals = [0.1, 0.2, 0.4, 0.8]*pi;

% plot magnitude of H(w) vs w
fig = figure;
plot(w, Hw_mag);
xlabel('w   (radians/sample)');
ylabel('|H(e^{jw})|');
title({'Magnitude of H(e^{jw}) vs w';''});
setDTFTradialAxis(1);
markOnPlot(wvals, w, Hw_mag, dw, [], 0.65, {'|H(e^{ j ', '\pi})|', ''}, pi);
```




```
% plot phase (unwrapped) of H(w) vs w
fig = figure;
plot(w, Hw_arg);
xlabel('w   (radians/sample)');
ylabel('arg{\bf[}H(e^{jw}){\bf]}   (radians)');
title({'Phase (unwrapped) of H(e^{jw}) vs w';''});
setDTFTradialAxis(1, 1, 1);
```




```
% plot phase (wrapped/principal) of H(w) vs w
fig = figure;
plot(w, Hw_ARG);
xlabel('w   (radians/sample)');
ylabel('ARG{\bf[}H(e^{jw}){\bf]}   (radians)');
title({'Phase (wrapped/principal) of H(e^{jw}) vs w';''});
setDTFTradialAxis(1, 0.5, 1);
```



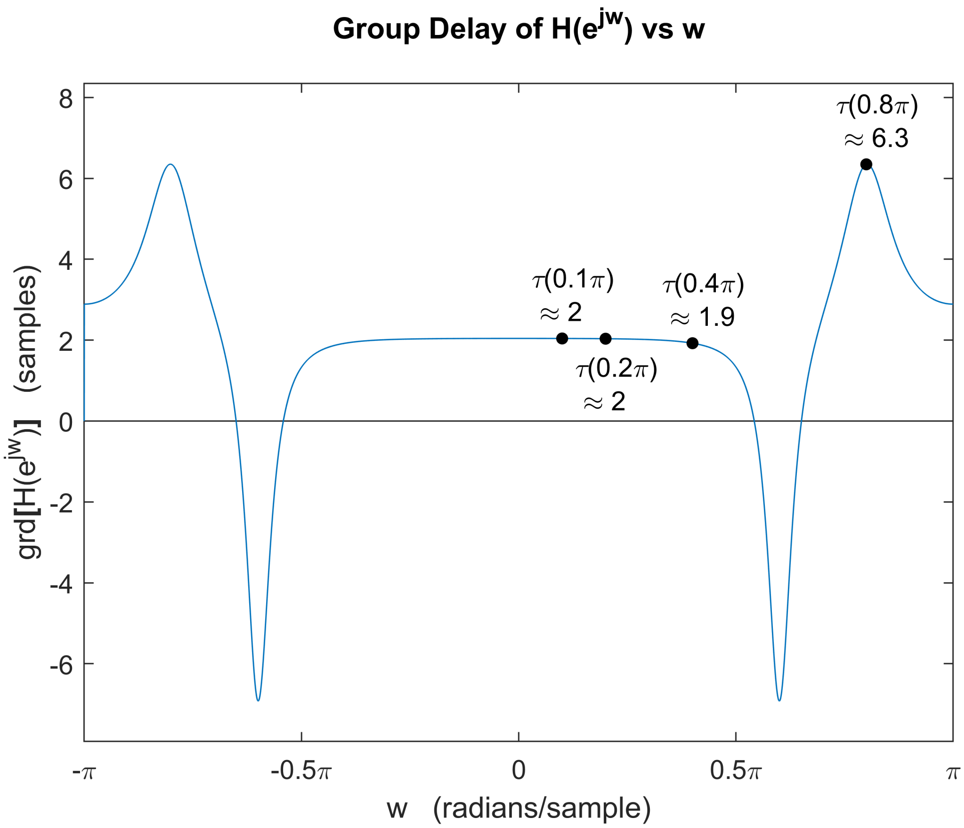
Now, the group delay for $H(e^{j\omega})$ is given by,

$$\tau(\omega) \;=\; \mathrm{grd}\big[H(e^{j\omega})\big] \;=\; -\frac{d}{d\omega}\Big\{\arg\big[H(e^{j\omega})\big]\Big\}$$

where the derivative of (unwrapped) phase may be computed using,

$$\frac{d}{d\omega}\Big\{\arg\big[H(e^{j\omega})\big]\Big\} \;=\; \lim_{\Delta\omega \to 0} \frac{\arg\big[H(e^{j(\omega+\Delta\omega)})\big] - \arg\big[H(e^{j\omega})\big]}{\Delta\omega}$$

```
% <task> plot group-delay of given H(z)

dw = (w(end)-w(1))/(length(w)-1);          % compute the w-step-size 'dw' used in w-array

% obtain group-delay of H(w)
Hw_grd = -([Hw_arg 0]-[0 Hw_arg])/dw;
Hw_grd = Hw_grd(1:(length(Hw_grd)-1));

% plot group-delay of H(w) vs w
fig = figure;
plot(w, Hw_grd);
xlabel('w   (radians/sample)'); ylabel('grd{\bf[}H(e^{jw}){\bf]}   (samples)');
title({'Group Delay of H(e^{jw}) vs w';''});
setDTFTradialAxis(1); ylim([min(Hw_grd)-1 max(Hw_grd)+2]);
markOnPlot(wvals, w, Hw_grd, dw, -0.07*pi, [1 -1 1 1]*1.1, {'\tau(', '\pi)', ''}, pi);
```



Question (2):

Is the system stable? Is the system causal?

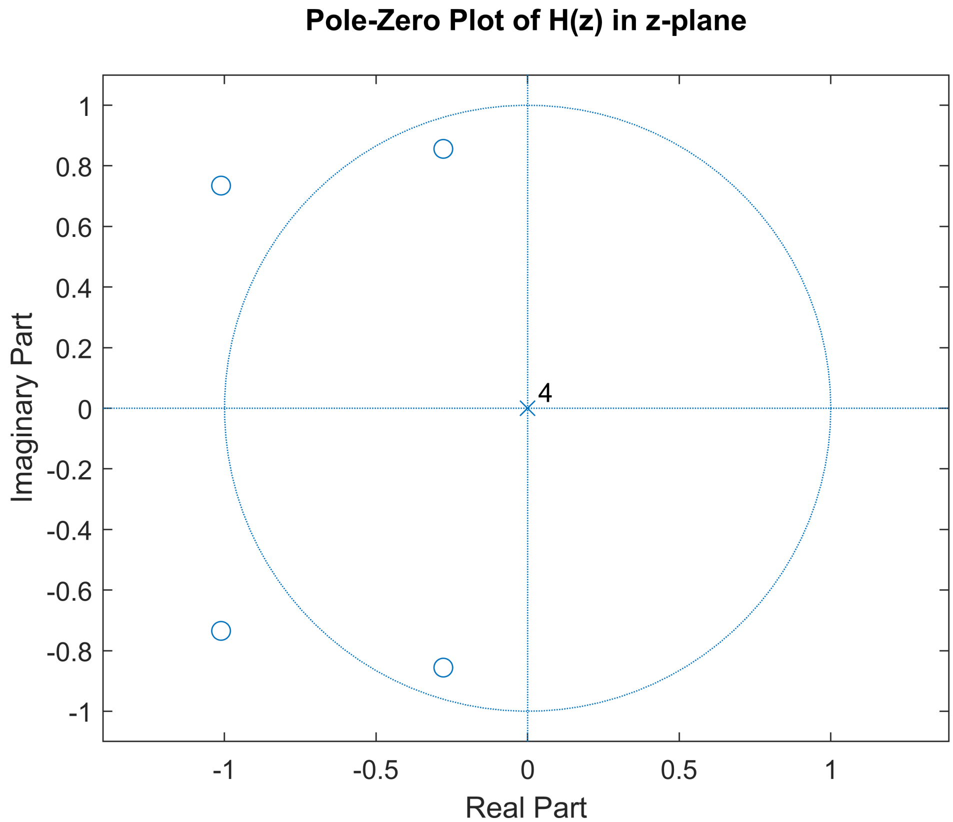
Answer (2):

Consider the pole-zero plot of this system $H(z)$, as drawn below.

```
% extract CCDE coefficients (b's and a's) from H(z) expression
[N, D] = numden(Hz);
b = coeffs(N, 'All');   a = coeffs(D, 'All');
b = double(b./a(1));    a = double(a./a(1));
disp('b ='); disp(b);   disp('a ='); disp(a);
```

```
b =
    1.0000    2.5788    3.4975    2.5074    1.2656

a =
     1     0     0     0     0
```

```
% plot the pole-zero map of H(z) using extracted b's and a's
figure;
zplane(b, a);
title({'Pole-Zero Plot of H(z) in z-plane';''});
```



As can be seen above, all poles of $H(z)$ lie within the unit-cricle. Furthermore, all 4 poles lie at origin. This means the only possible ROC is the one extending outward from $|z| > 0$. So, ROC is all of z-plane except $z = 0$.

Since the unit-circle lies in ROC, the system is stable. As here, ROC extends outward from outermost pole ($z = 0$), so $h[n]$ is a right-sided function (possibly with $h[n] = 0 \;\forall\; n < 0$). Hence the system is also causal.

So the system represented by $H(z)$ is causal and stable.

Question (3):

Pass $x[n]$ through $H(z)$ for $\omega_c = 0.1\pi,\ 0.2\pi,\ 0.4\pi,\ 0.8\pi$ and obtain the output signal $y[n]$.

How does the magnitude of the output signal change with change in ω_c? Can you explain the difference in magnitude corresponding to different ω_c using the magnitude response of $H(z)$.

Answer (3):

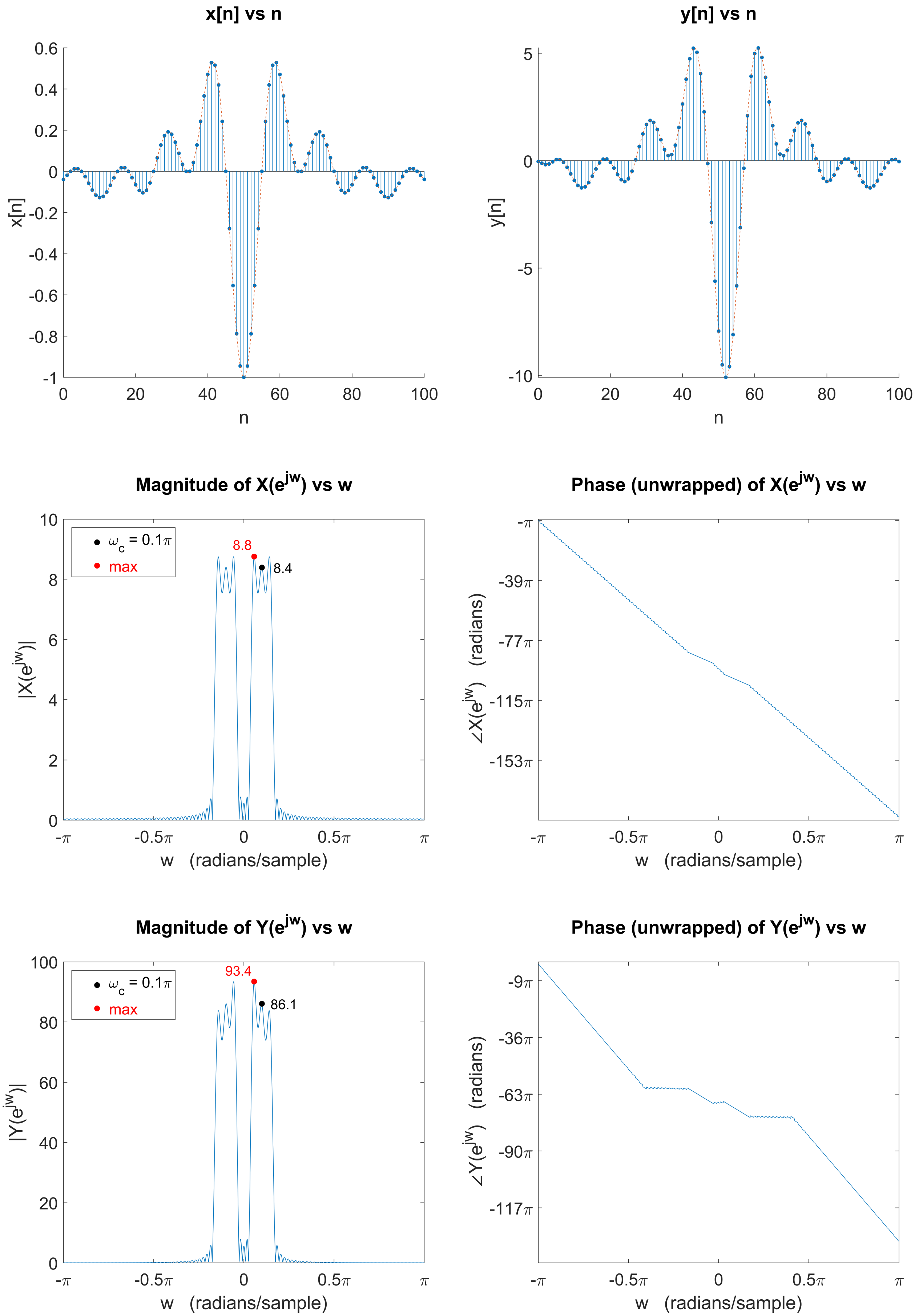
Since MATLAB defines $\mathrm{sinc}()$ function as $\mathrm{sinc}(x) = \dfrac{\sin(\pi x)}{\pi x}$, we'll define $x[n]$ as, $x[n] = \mathrm{sinc}\left(\dfrac{n-50}{16}\right)\cos(\omega_c n)$, i.e. without the extra π in MATLAB.

Note that in frequency-domain, this $\text{sinc}()$ function corresponds to a gate-pulse of certain width (W), which is shift-copied to $\omega = \pm\,\omega_c\,\text{rad/sample}$ due to the $\cos()$ term. Changing ω_c in $x[n]$ thus changes where the gate-pulse (+ve & -ve) is centered in $x[n]$'s spectrum.

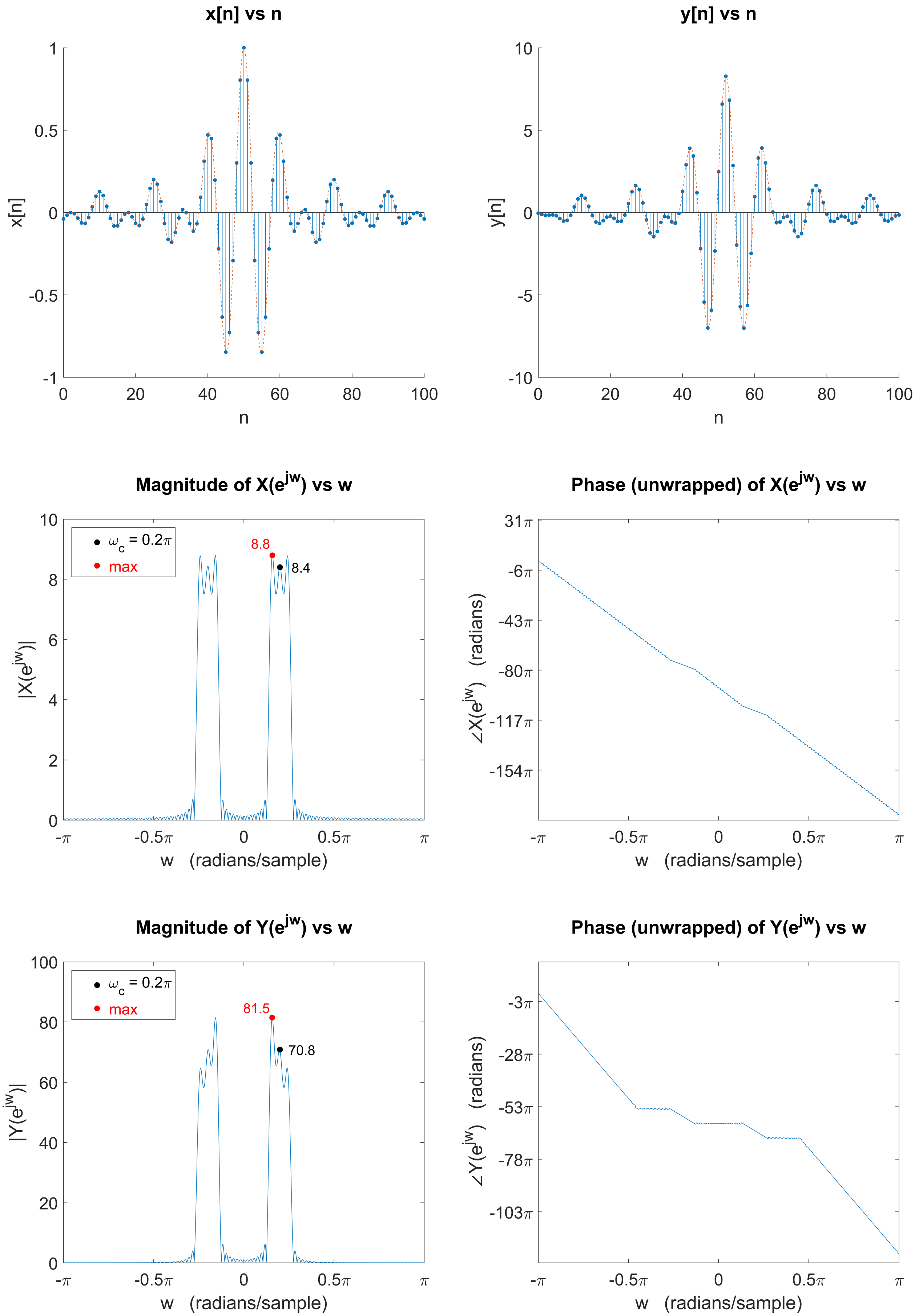
The code, along with the results, to pass $x[n]$ for each ω_c through $H(z)$, starts below.

```
% <task> pass x[n] (for each wc value) through system H(z),
% and plot the output y[n] and its spectrum

wcvals = [0.1, 0.2, 0.4, 0.8]*pi;           % wc values (x[n] center frequencies)
n = 0:100;
x = cell(1, length(wcvals));                % to store x[n] arrays for different wc
y = cell(1, length(wcvals));                % to store y[n] arrays for different wc
ny = cell(1, length(wcvals));               % to store ny arrays for different wc

for i = 1:length(wcvals)

    % define x[n] = (sin(pi*(n-50)/16)./(pi*(n-50)/16)).*cos(wc*n)
    wc = wcvals(i);
    x{i} = sinc((n-50)/16).*cos(wc*n);

    % compute DTFT X(w)
    w = -pi:0.001:pi;                       % for 1 period [-pi, pi] of DTFT
    Xw = x{i} * exp(-1i * n' * w);          % using matrix-multiplication method

    % obtain the magnitude and phase (unwrapped, in radians) of DTFT X(w)
    Xw_mag = abs(Xw);
    Xw_arg = phase(Xw);

    % assuming zero initial conditions (rest)
    N_aux = zeros(1, length(a)-1);
    M_init = zeros(1, length(b)-1);

    % compute output y[n], with input x[n] to system represented by H(z),
    % using get_system_out() and extracted b's and a's
    [y{i}, ny{i}] = get_system_out(b, a, x{i}, N_aux, M_init);

    % compute DTFT Y(w)
    w = -pi:0.001:pi;                       % for 1 period [-pi, pi] of DTFT
    Yw = y{i} * exp(-1i * ny{i}' * w);      % using matrix-multiplication method

    % obtain the magnitude and phase (unwrapped, in radians) of DTFT Y(w)
    Yw_mag = abs(Yw);
    Yw_arg = phase(Yw);

    % plot sequences x[n] & y[n], and their DTFTs X(w) & Y(w) (magnitude and phase)
    plot2SeqDTFT(n, x{i}, wc, w, Xw_mag, Xw_arg, 'x', 'X', ...
        [0.2 0 -0.04 0.4, 0.15 0 -0.04 0.04*max(Yw_mag)], ...
        ny{i}, y{i}, Yw_mag, Yw_arg, 'y', 'Y', 'left', 'right');

end
```

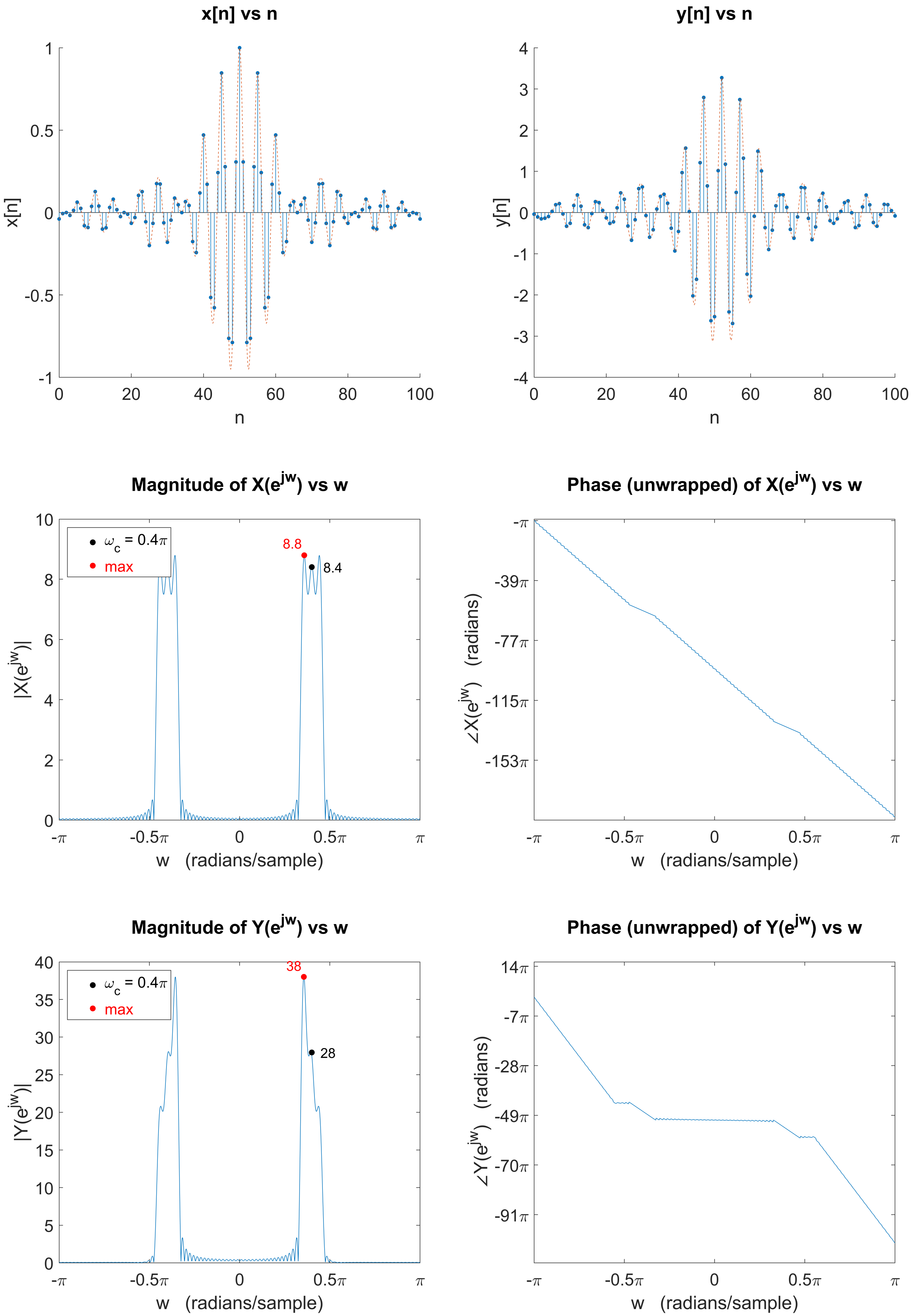
$\omega_c = 0.1\pi$ rad/sample



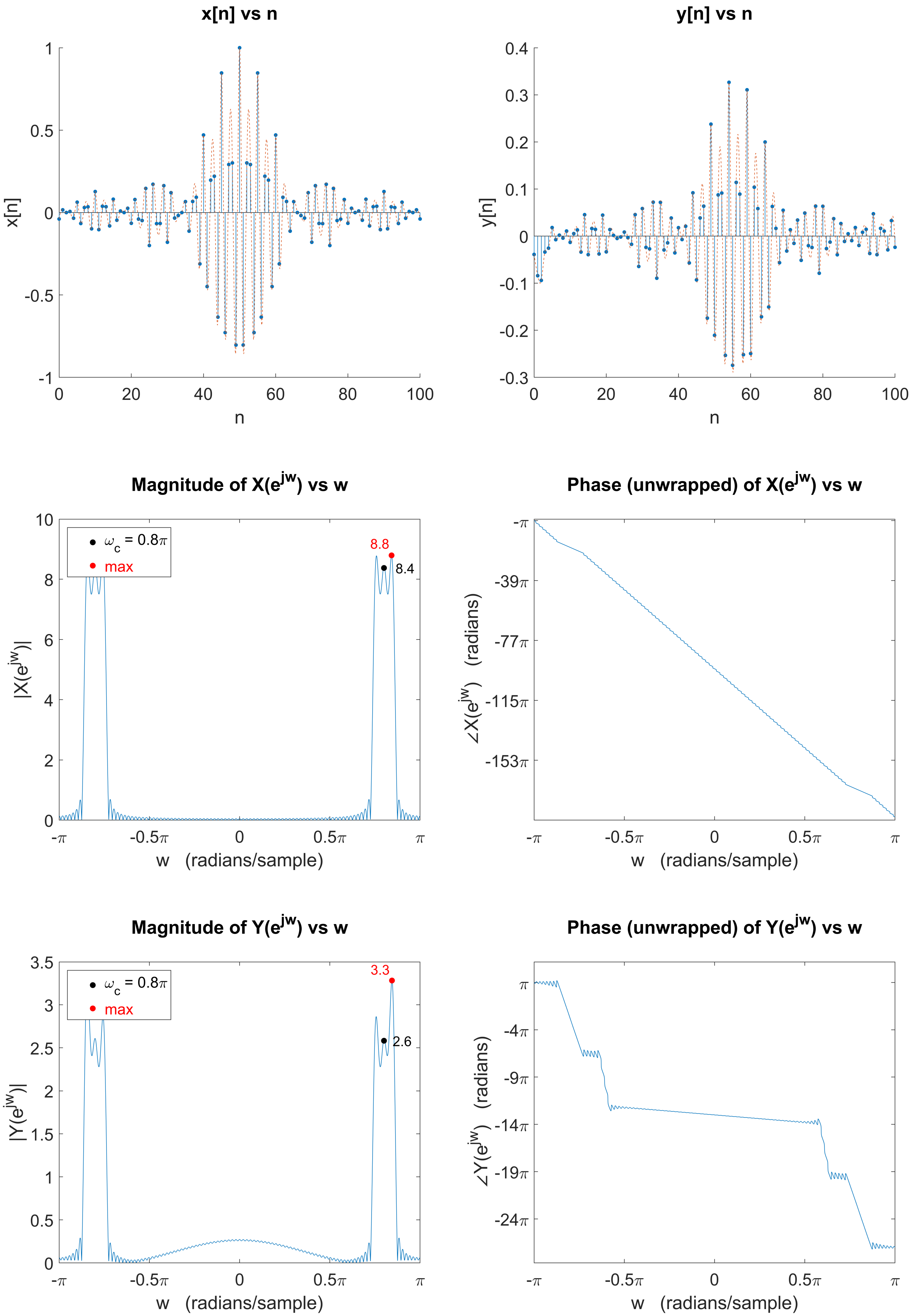
$\omega_c = 0.2\pi$ rad/sample



$$\omega_c = 0.4\pi \quad \text{rad/sample}$$



$\omega_c = 0.8\pi$ rad/sample



In the magnitude plot of $H(e^{j\omega}) = H(\omega)$ (Page 3), the magnitude values i.e. gains for each of the ω_c frequencies have been marked. These gain values are as follows:

$$\begin{aligned}
|H(\ 0.1\pi \ \text{rad/sample}\ )| &\approx 10.2 \\
|H(\ 0.2\pi \ \text{rad/sample}\ )| &\approx 8.3 \\
|H(\ 0.4\pi \ \text{rad/sample}\ )| &\approx 3.2 \\
|H(\ 0.8\pi \ \text{rad/sample}\ )| &\approx 0.3
\end{aligned}$$

Since $H(z)$ is not an all-pass system, the various frequency components (centered around ω_c) in $x[n]$ are distorted (in magnitude) by different amounts. Following table summarizes the variation in magnitude from $X(e^{j\omega})$ to $Y(e^{j\omega})$ after being passed through $H(z)$:

	$\lvert X(e^{j\omega})\rvert$		$\lvert Y(e^{j\omega})\rvert$		$\frac{\lvert Y(e^{j\omega})\rvert}{\lvert X(e^{j\omega})\rvert} = \lvert H(e^{j\omega})\rvert$	
center frequency of x[n] (ω_c)	at ω_c	max	at ω_c	max	at ω_c	max
0.1π rad/sample	8.4	8.8	86.1	93.4	10.3	10.6
0.2π rad/sample	8.4	8.8	70.8	81.5	8.4	9.26
0.4π rad/sample	8.4	8.8	28	38	3.3	4.3
0.8π rad/sample	8.4	8.8	2.6	3.3	0.3	0.38

The ratio of output to input magnitude (shown in the last 2 columns of above table) corresponds to the gain provided by the system $H(z)$ to the input. Note that the gain values at center frequencies ω_c, calculated in above table, match (approximately) with the magnitude values noted further above from the $|H(e^{j\omega})|$ plot at respective frequencies. Thus the output magnitude is given by,

$$|Y(e^{j\omega})| = |X(e^{j\omega})|\,|H(e^{j\omega})|$$

And since $H(e^{j\omega})$ has different (widely-varying) magnitudes at different frequencies ω, this explains the changing magnitude of output $Y(e^{j\omega})$ based on the frequency-content in input $X(e^{j\omega})$ (centered around $\pm\omega_c$).

Next Question/Answer starts from the next page.

Question (4.a):

Design a causal, stable magnitude inverse of the system. Draw magnitude, phase and group delay response of the inverse sytem.

Answer (4.a):

$H(z)$ (stable & causal) is a non-minimum-phase system (doesn't have all poles and zeros inside unit circle), so its rational inverse $1/H(z)$ is not stable & causal. But, any system $H(z)$ can be decomposed into minimum-phase $H_{\min}(z)$ and all-pass $H_{\text{ap}}(z)$ systems, as follows:

$$H(z) \;=\; H_{\min}(z)\, H_{\text{ap}}(z)$$

where $H_{\min}(z)$ has all its poles and zeros inside the unit-circle, that is, it and its inverse ($1/H_{\min}(z)$) are stable & causal; while $H_{\text{ap}}(z)$ has constant (unity) magnitude. In terms of magnitude, $H_{\text{ap}}(z)$ does not have any contribution, hence the magnitude distortion in $H(z)$ comes from $H_{\min}(z)$.

So, to compensate for this magnitude distortion, we may use the system given by $1/H_{\min}(z)$ — which will be a casual and stable magnitude inverse of $H(z)$. We'll call the inverse system $H_{\text{inv}}(z)$, given as,

$$H_{\text{inv}}(z) \;=\; \frac{1}{H_{\min}(z)}$$

First, we decompose $H(z)$ into $H_{\min}(z)$ and $H_{\text{ap}}(z)$.

$H(z)$ has two zeros outside the unit-circle (at $a_2 = 1.25e^{j\,0.8\pi}$ and $a_2^* = 1.25e^{-j\,0.8\pi}$). So, first place a zero and a pole at conjugate-reciprocal positions ($0.8e^{\mp j\,0.8\pi}$) of these zeros. So, we have,

$$H(z) \;=\; \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-1.25e^{j\,0.8\pi})(z-1.25e^{-j\,0.8\pi})}{z^4}\cdot\frac{(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}{(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}$$

This does not change overall $H(z)$ as the pole & zero cancel out to give original $H(z)$. From this expression, we can extract an all-pass system, of the form with $M_c = 1$ complex-conjugate pole-pair & respective reciprocal zero-pair,

$$H_{\text{ap}}(z) \;=\; \frac{\left(z^{-1}-e_k{}^*\right)\left(z^{-1}-e_k\right)}{\left(1-e_k\,z^{-1}\right)\left(1-e_k^*\,z^{-1}\right)}$$

obtained by keeipng the new zeros (placed inside unit-circle) in $H_{\min}(z)$ and putting the outside zeros along with new inside poles into the $H_{\text{ap}}(z)$ part.

The decomposition proceeds as follows:

$$H(z) \;=\; \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}{z^4}\cdot\frac{(z-1.25e^{j\,0.8\pi})(z-1.25e^{-j\,0.8\pi})}{(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}$$

$$H(z) = \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}{z^4} \cdot \frac{(1.25)^2(0.8z-e^{j\,0.8\pi})(0.8z-e^{-j\,0.8\pi})}{(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}$$
$$\times \frac{z^{-2}(e^{-j\,0.8\pi})(e^{j\,0.8\pi})}{z^{-2}}$$

$$H(z) = (1.25)^2 \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}{z^4} \cdot \frac{(0.8e^{-j\,0.8\pi}-z^{-1})(0.8e^{j\,0.8\pi}-z^{-1})}{(1-0.8e^{j\,0.8\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})}$$

$$H(z) = (1.25)^2(1-0.9e^{j\,0.6\pi}z^{-1})(1-0.9e^{-j\,0.6\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})(1-0.8e^{j\,0.8\pi}z^{-1})$$
$$\times \frac{(z^{-1}-0.8e^{-j\,0.8\pi})(z^{-1}-0.8e^{j\,0.8\pi})}{(1-0.8e^{j\,0.8\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})}$$

And so, the all-pass and minimum-phase parts of the system are respectively,

$$H_{\text{ap}}(z) = \frac{(z^{-1}-0.8e^{-j\,0.8\pi})(z^{-1}-0.8e^{j\,0.8\pi})}{(1-0.8e^{j\,0.8\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})}$$

$$H_{\text{min}}(z) = (1.25)^2(1-0.9e^{j\,0.6\pi}z^{-1})(1-0.9e^{-j\,0.6\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})(1-0.8e^{j\,0.8\pi}z^{-1})$$

Finally, the magnitude inverse system of $H(z)$ is given by,

$$H_{\text{inv}}(z) = \frac{1}{H_{\text{min}}(z)} = \frac{1}{(1.25)^2(1-0.9e^{j\,0.6\pi}z^{-1})(1-0.9e^{-j\,0.6\pi}z^{-1})(1-0.8e^{-j\,0.8\pi}z^{-1})(1-0.8e^{j\,0.8\pi}z^{-1})}$$
$$= \frac{z^4}{(1.25)^2(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}$$

which is causal & stable since all its poles and zeros lie inside unit circle.

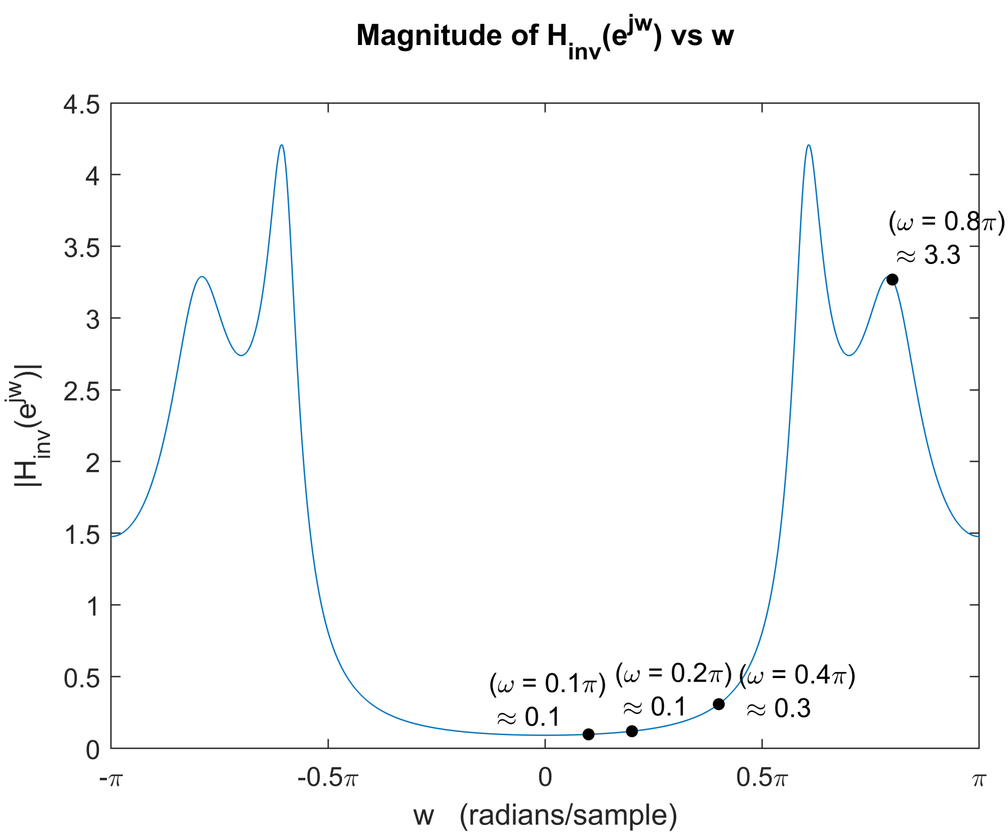
The code, along with the results (inline), to compute & plot various responses from $H_{\text{inv}}(z)$ expression above, starts below.

```
% <task> plot magnitude and phase response of inverse system Hinv(z)

% define z-domain expression (symbolically)
syms z;
a1 = 0.9*exp(1i*0.6*pi);
a2 = 0.8*exp(1i*0.8*pi);
Hinvz_N = z^4;                                              % numerator of Hinv(z)
Hinvz_D = (1.25^2)*(z-a1)*(z-conj(a1))*(z-conj(a2))*(z-a2);     % denominator of Hinv(z)
Hinvz = Hinvz_N/Hinvz_D;

% display Hinv(z) expression
disp('Hinv(z) = '); disp(vpa(Hinvz, 4))
```

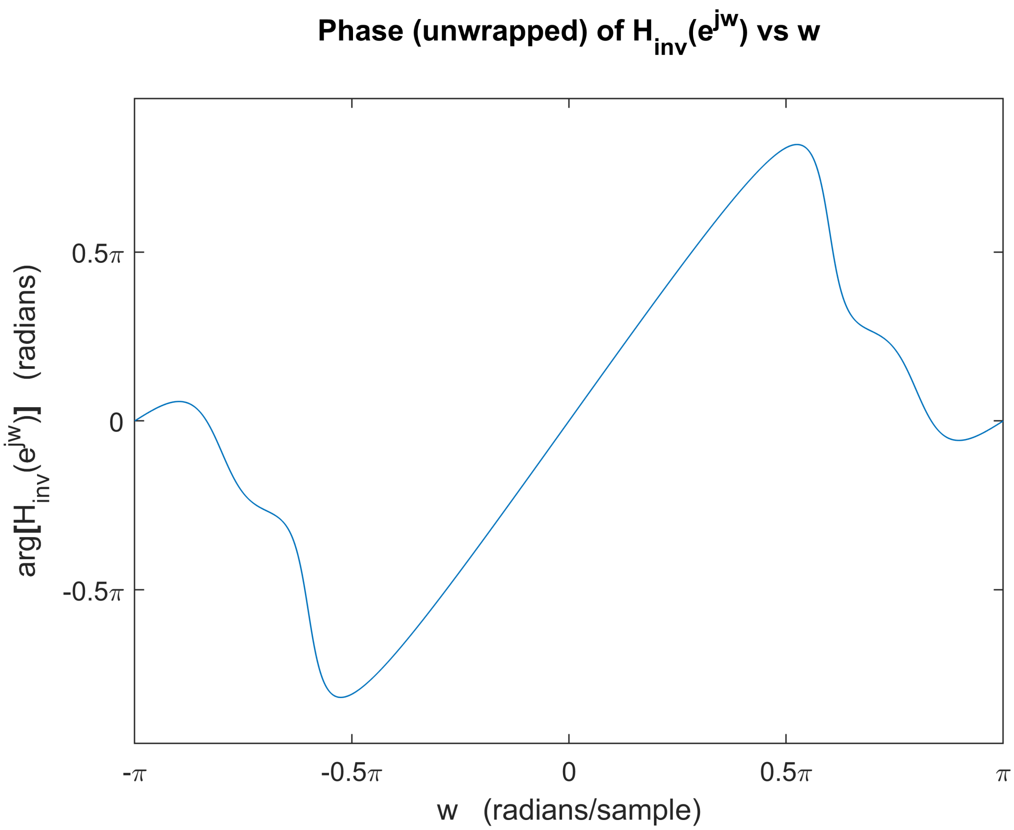
```
Hinv(z) =
```

$$\frac{z^4}{(1.562\,z+0.4346-1.337\,\mathrm{i})\;(z+0.2781+0.856\,\mathrm{i})\;(z+0.6472-0.4702\,\mathrm{i})\;(z+0.6472+0.4702\,\mathrm{i})}$$

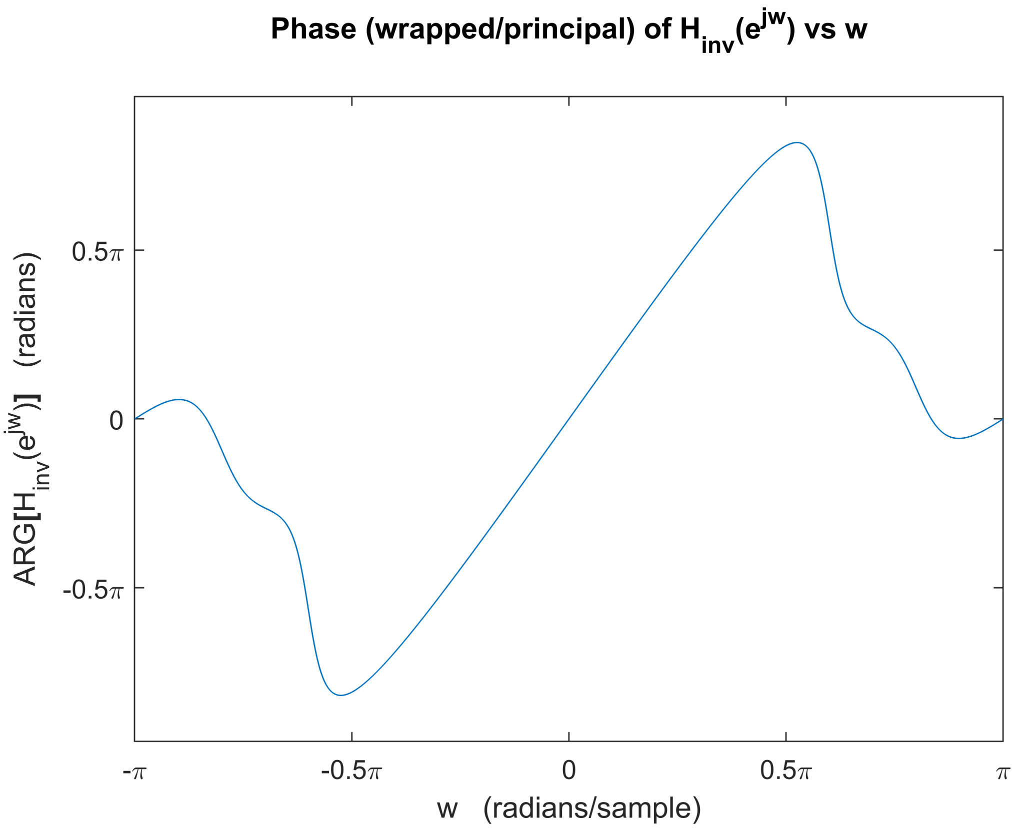
```
% obtain frequency response Hinv(w) expression from Hinv(z) by substituting z = e^(jw)
syms w_;
Hinvw = subs(Hinvz, {z}, {exp(1i*w_)});

% compute frequency response Hinv(w) for w = [-pi, pi]
dw = 0.001;
w = -pi:dw:pi;
Hinvw = subs(Hinvw, {w_}, {w});
Hinvw = double(Hinvw);

% obtain magnitude and phase of Hinv(w)
Hinvw_mag = abs(Hinvw);
Hinvw_arg = phase(Hinvw);            % unwrapped phase (in radians)
Hinvw_ARG = angle(Hinvw);            % wrapped phase (in radians)

% frequencies to be noted/marked on the response plots
wvals = [0.1, 0.2, 0.4, 0.8]*pi;

% plot magnitude of Hinv(w) vs w
fig = figure;
plot(w, Hinvw_mag);
xlabel('w   (radians/sample)');
ylabel('|H_{inv}(e^{jw})|');
title({'Magnitude of H_{inv}(e^{jw}) vs w';''});
setDTFTradialAxis(1);
markOnPlot(wvals, w, Hinvw_mag, dw, ...
    [-0.23 -0.04 0.045 -0.01]*pi, [0.25 0.3 0.1 0.3], {'(\omega = ', '\pi)', ''}, pi);
```



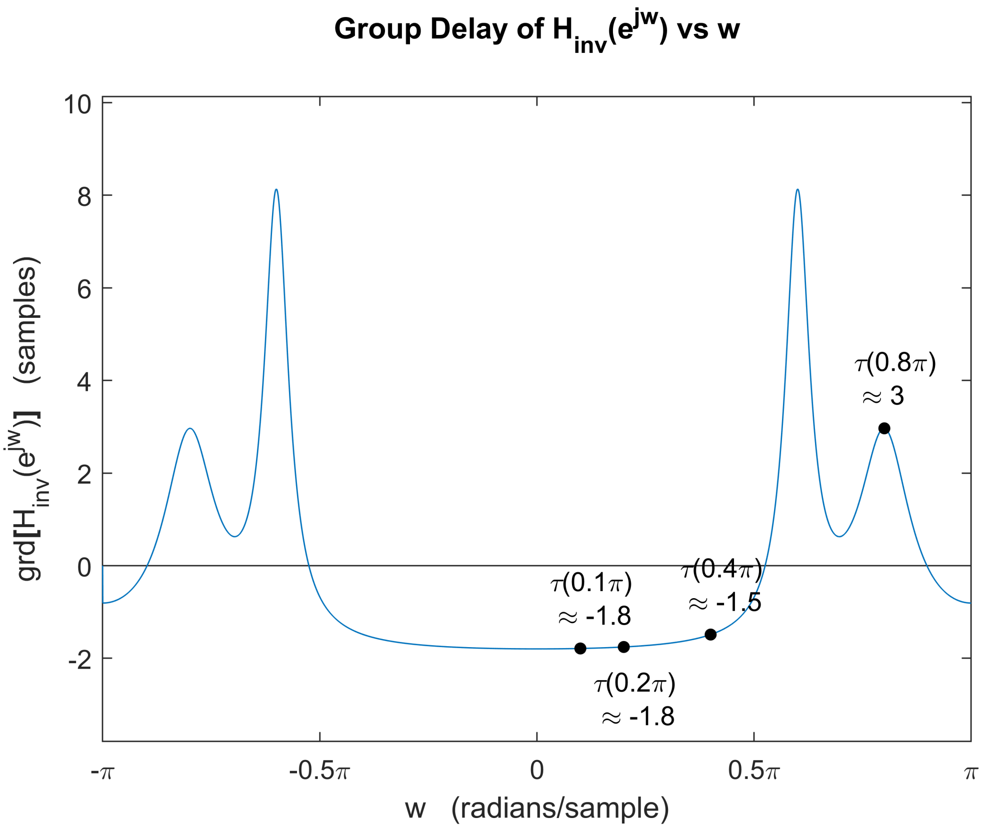
```
% plot phase (unwrapped) of Hinv(w) vs w
fig = figure;
plot(w, Hinvw_arg);
xlabel('w   (radians/sample)');
ylabel('arg{\bf[}H_{inv}(e^{jw}){\bf]}   (radians)');
title({'Phase (unwrapped) of H_{inv}(e^{jw}) vs w';''});
setDTFTradialAxis(1, 0.5, 1);
```




```
% plot phase (wrapped/principal) of Hinv(w) vs w
fig = figure;
plot(w, Hinvw_ARG);
xlabel('w   (radians/sample)');
ylabel('ARG{\bf[}H_{inv}(e^{jw}){\bf]}   (radians)');
title({'Phase (wrapped/principal) of H_{inv}(e^{jw}) vs w';''});
setDTFTradialAxis(1, 0.5, 1);
```



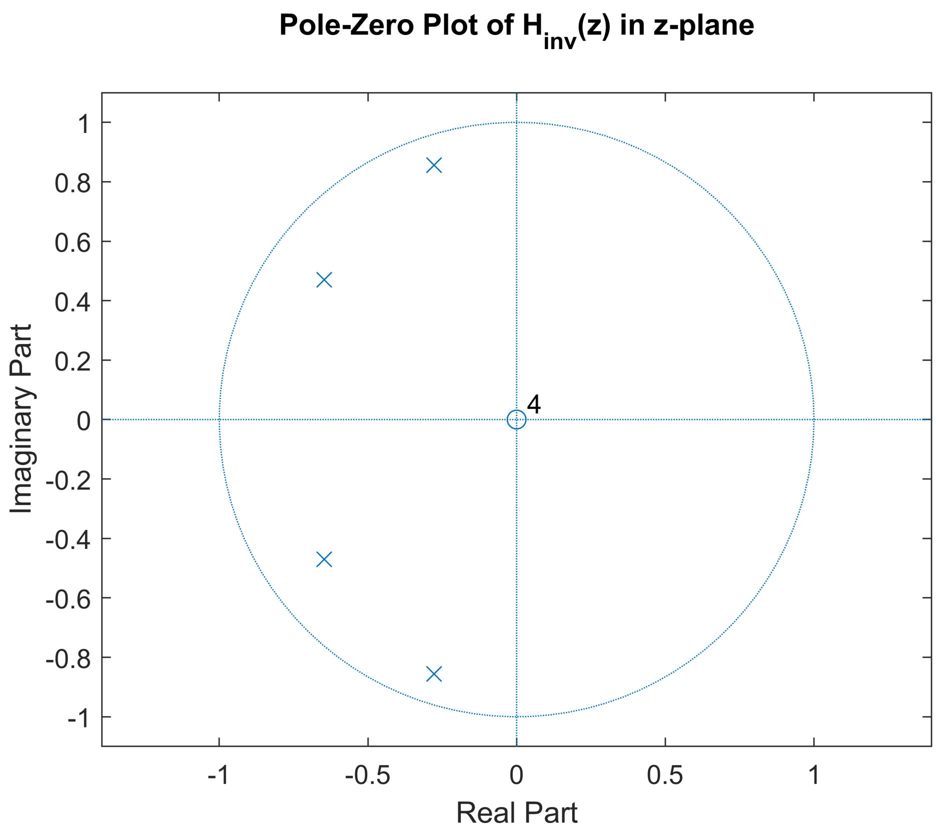

```
% <task> plot group-delay of inverse system Hinv(z)

dw = (w(end)-w(1))/(length(w)-1);          % compute the w-step-size 'dw' used in w-array

% obtain group-delay of Hinv(w)
Hinvw_grd = -([Hinvw_arg 0]-[0 Hinvw_arg])/dw;
Hinvw_grd = Hinvw_grd(1:(length(Hinvw_grd)-1));

% plot group-delay of Hinv(w) vs w
fig = figure;
plot(w, Hinvw_grd);
xlabel('w   (radians/sample)'); ylabel('grd{\bf[}H_{inv}(e^{jw}){\bf]}   (samples)');
title({'Group Delay of H_{inv}(e^{jw}) vs w';''});
setDTFTradialAxis(1); ylim([min(Hinvw_grd)-2 max(Hinvw_grd)+2]);
markOnPlot(wvals, w, Hinvw_grd, dw, -0.07*pi, [1 -1 1 1]*1.1, {'\tau(', '\pi)', ''}, pi);
```




```
% <task> plot pole-zero map of inverse system Hinv(z)

% extract CCDE coefficients (b's and a's) from Hinv(z) expression
[N, D] = numden(Hinvz);
b = coeffs(N, 'All');   a = coeffs(D, 'All');
b = double(b./a(1));    a = double(a./a(1));
disp('b ='); disp(b);   disp('a ='); disp(a);
```

```
b =
    0.6400         0         0         0         0

a =
    1.0000    1.8507    2.1700    1.4045    0.5184
```

```
% plot the pole-zero map of Hinv(z) using extracted b's and a's
figure;
zplane(b, a);
title({'Pole-Zero Plot of H_{inv}(z) in z-plane';''});
```



All poles (and zeros) of $H_{\text{inv}}(z)$ lie within the unit-cricle. A possible ROC is that extending outward from outermost pole, $|z| > 0.9$. If ROC $|z| > 0.9$ is selected, then $H_{\text{inv}}(z)$ is stable as well as causal.

Question (4.b):

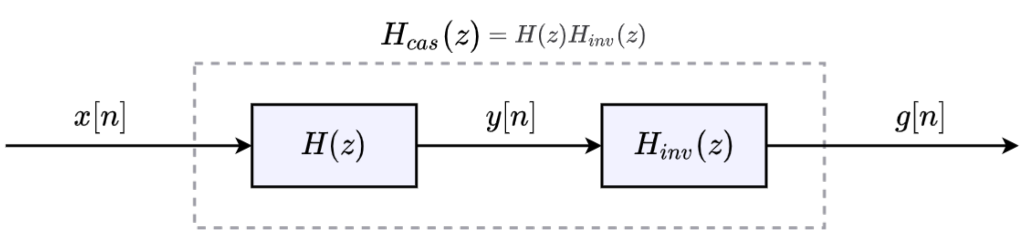
Pass all $y[n]$ obtained in the previous part through the the inverse system.

What is the maximum magnitude of the output of the inverse system? Can you explain the different maximum magnitudes of the output signals corresponding to different ω_c?

Do you think you have gotten exactly the same $x[n]$ at the output of the inverse system? Why or why not?

Answer (4.b):

Let $g[n]$ be the output of the magnitude inverse system. The overall system block diagram will be,

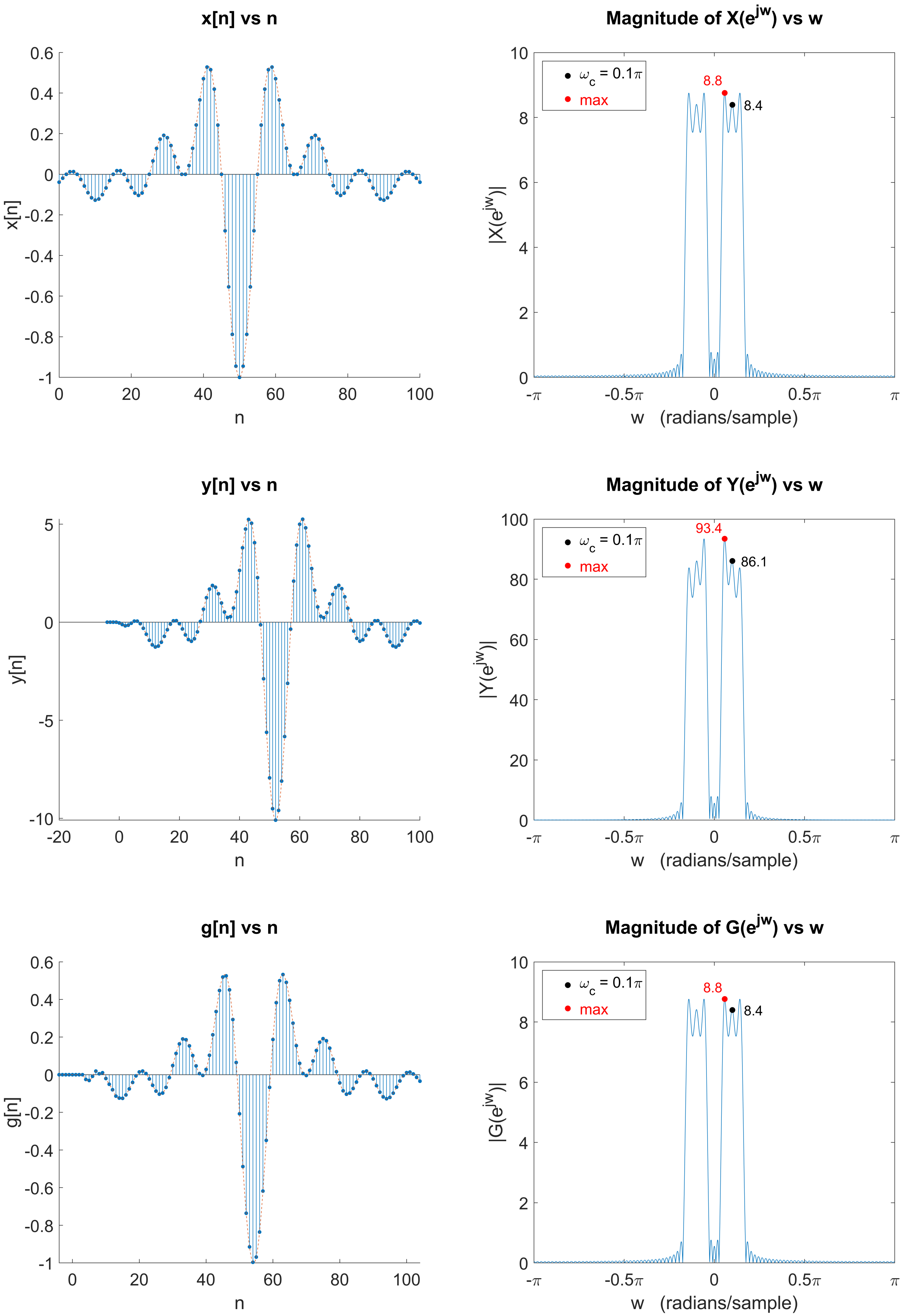


The code, along with the results, to pass $y[n]$ for each ω_c through $H_{\text{inv}}(z)$, starts below.

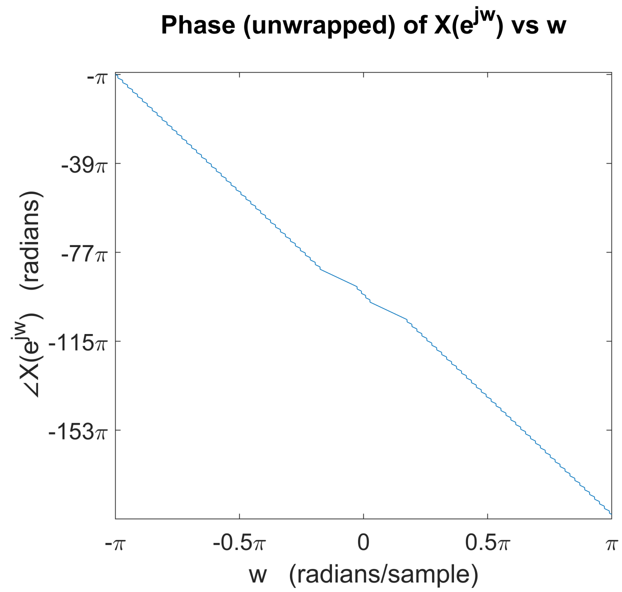
Note: The maximum magnitudes have been marked in the output spectrum Figures.

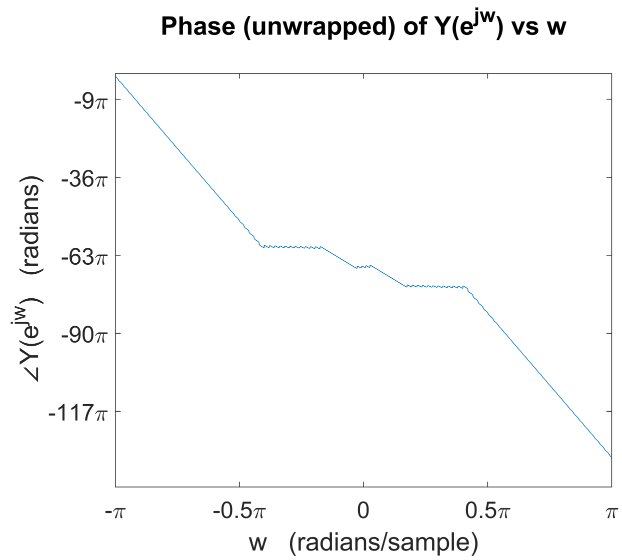
```
% <task> pass y[n] (for each wc value) through system Hinv(z),
% and plot the output g[n] and its spectrum

wcvals = [0.1, 0.2, 0.4, 0.8]*pi;           % wc values (x[n] center frequencies)
g = cell(1, length(wcvals));                % to store g[n] arrays for different wc
ng = cell(1, length(wcvals));               % to store ng arrays for different wc

for i = 1:length(wcvals)

    % assuming zero initial conditions (rest)
    N_aux = zeros(1, length(a)-1);
    M_init = zeros(1, length(b)-1);

    % compute output g[n], with input y[n] to system represented by Hinv(z),
    % using get_system_out() and extracted b's and a's
    [g{i}, ng{i}] = get_system_out(b, a, y{i}, N_aux, M_init);

    % compute DTFT G(w)
    w = -pi:0.001:pi;                       % for 1 period [-pi, pi] of DTFT
    Gw = g{i} * exp(-1i * ng{i}' * w);      % using matrix-multiplication method

    % obtain the magnitude and phase (unwrapped, in radians) of DTFT G(w)
    Gw_mag = abs(Gw);
    Gw_arg = phase(Gw);

    % compute DTFT X(w) for comparison with G(w)
    Xw = x{i} * exp(-1i * n' * w);          % using matrix-multiplication method
    % obtain the magnitude and phase (unwrapped, in radians) of DTFT X(w)
    Xw_mag = abs(Xw);
    Xw_arg = phase(Xw);

    % compute DTFT Y(w) for comparison with G(w)
    Yw = y{i} * exp(-1i * ny{i}' * w);      % using matrix-multiplication method
    % obtain the magnitude and phase (unwrapped, in radians) of DTFT Y(w)
    Yw_mag = abs(Yw);
    Yw_arg = phase(Yw);

    % plot sequences x[n], y[n] & g[n], and their DTFTs' magnitudes
    plot3SeqDTFT({n, ny{i}, ng{i}}, {x{i}, y{i}, g{i}}, wcvals(i), w, ...
        {Xw_mag, Yw_mag, Gw_mag}, {'x', 'y', 'g'}, {'X', 'Y', 'G'}, ...
        {Xw_arg, Yw_arg, Gw_arg},...
        {[0.2 0 -0.04 0.4], [0.15 0 -0.04 0.04*max(Yw_mag)], [0.2 0 -0.04 0.4]}, ...
        {'left', 'right'});

end
```

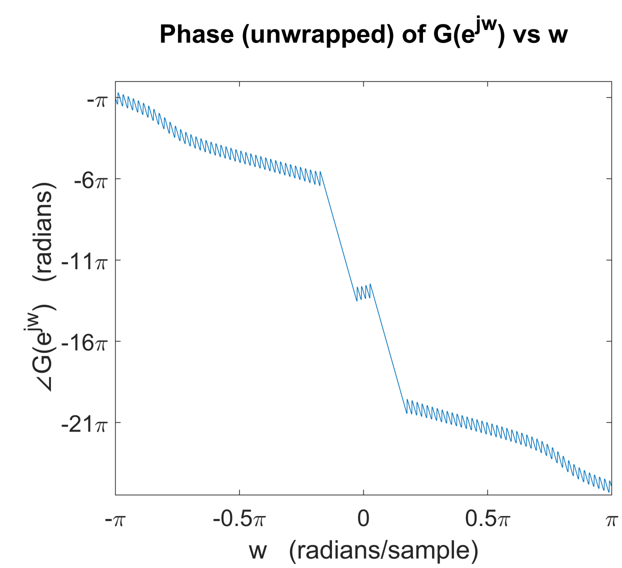
$\omega_c = 0.1\pi$ rad/sample



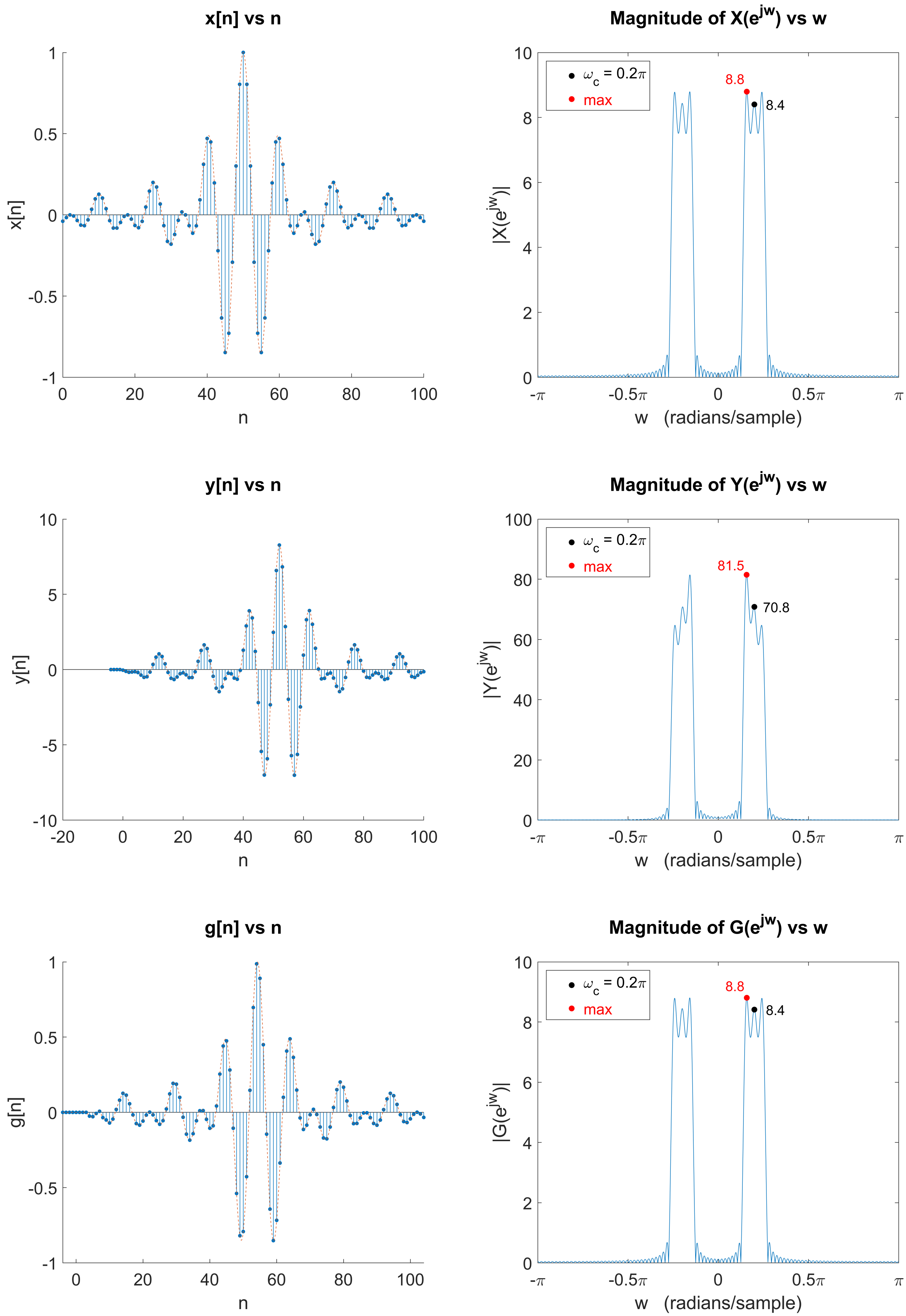
$$\omega_c = 0.1\pi \quad \text{rad/sample}$$



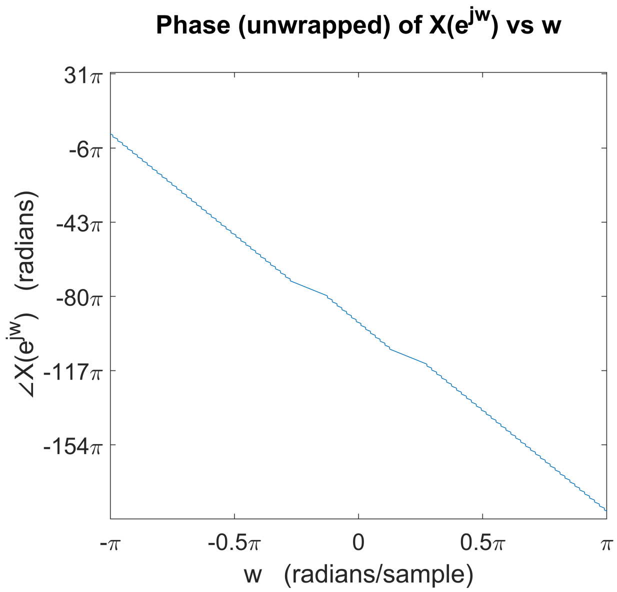


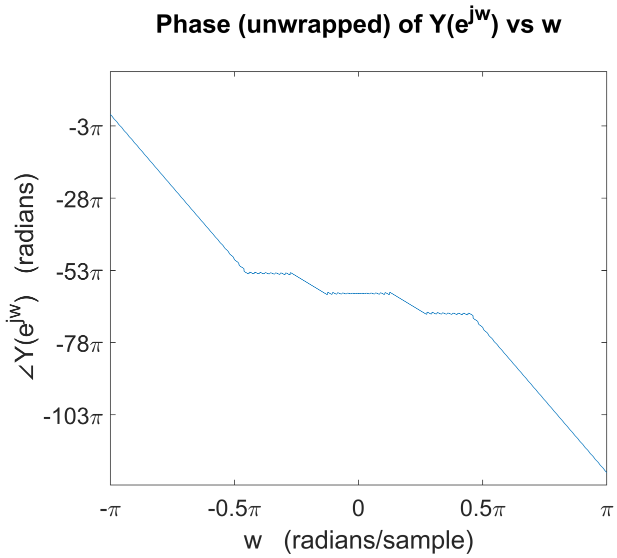


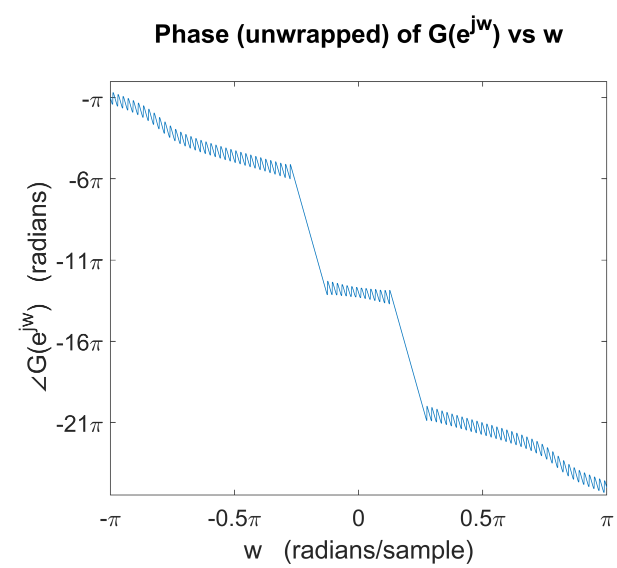
$$\omega_c = 0.2\pi \quad \text{rad/sample}$$



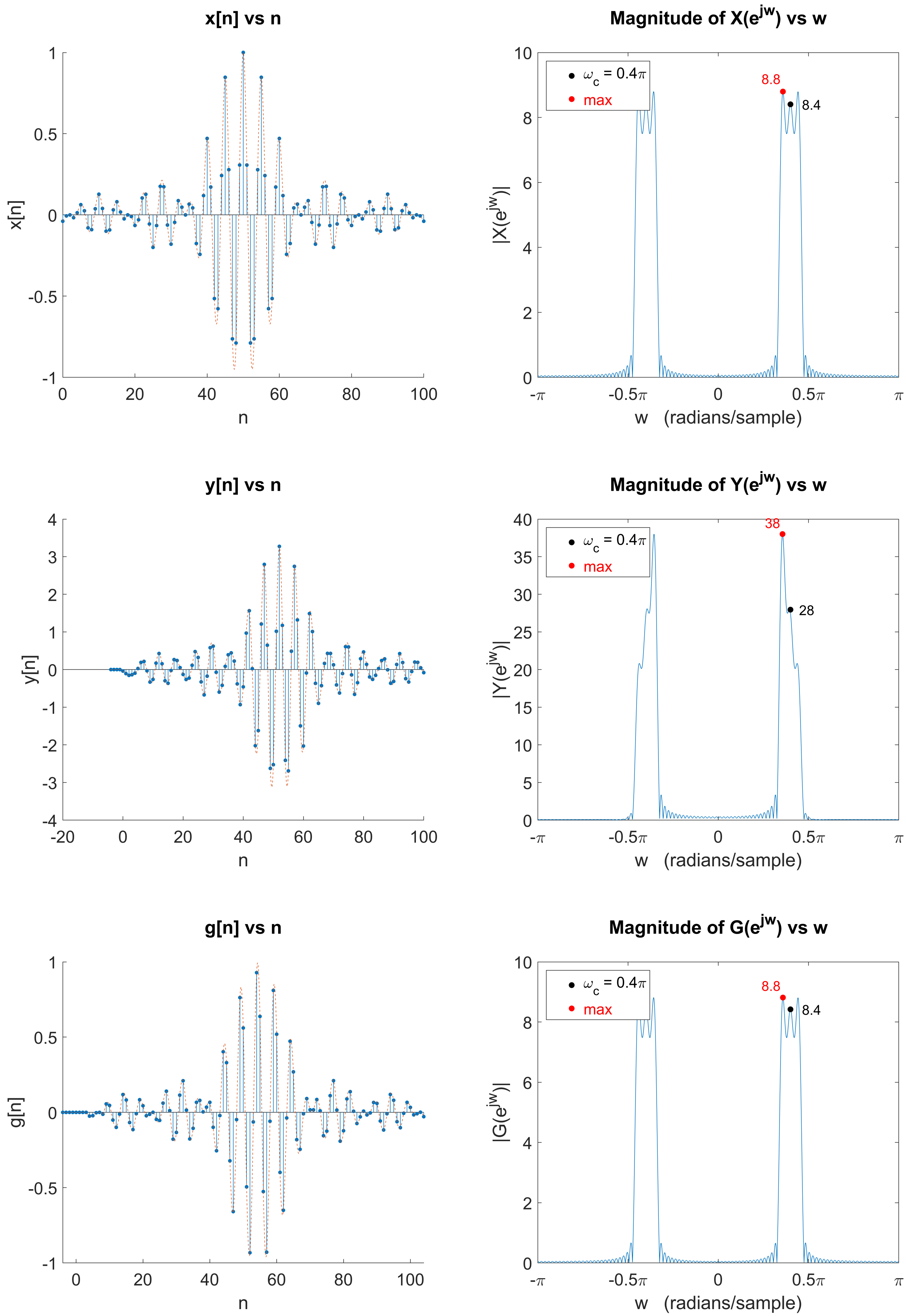
$$\omega_c = 0.2\pi \quad \text{rad/sample}$$



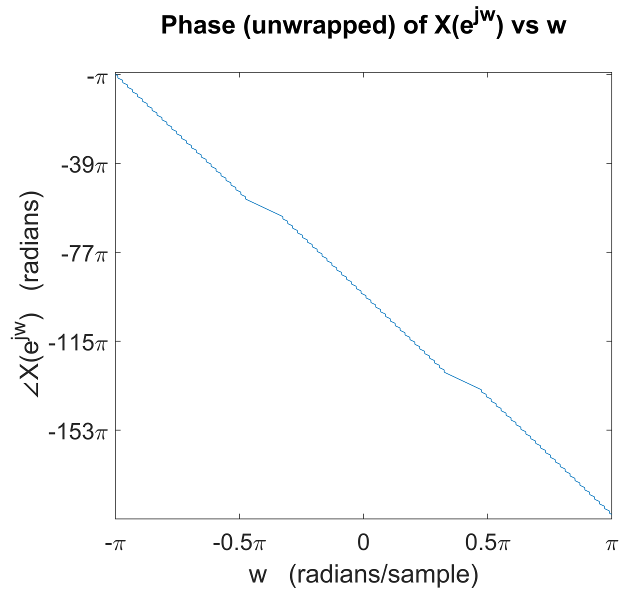


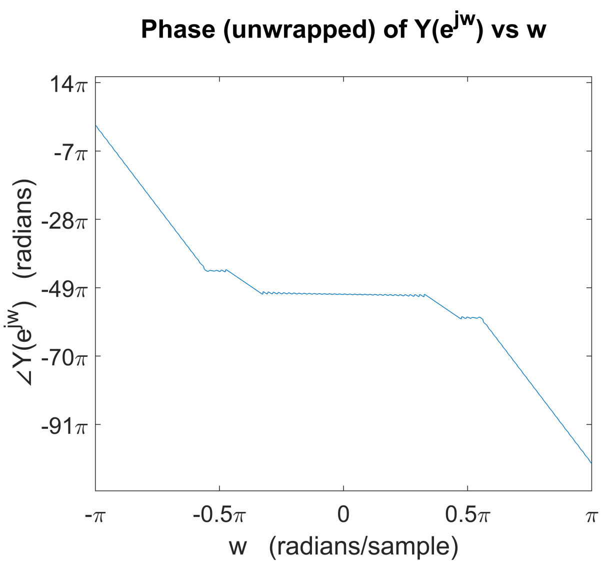


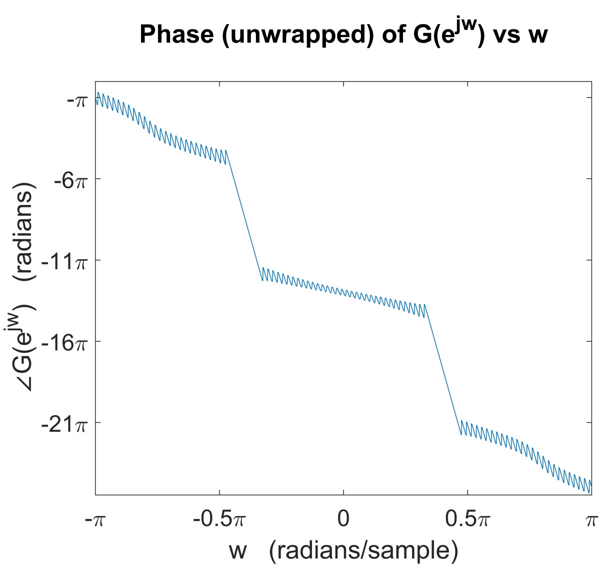
$\omega_c = 0.4\pi$ rad/sample



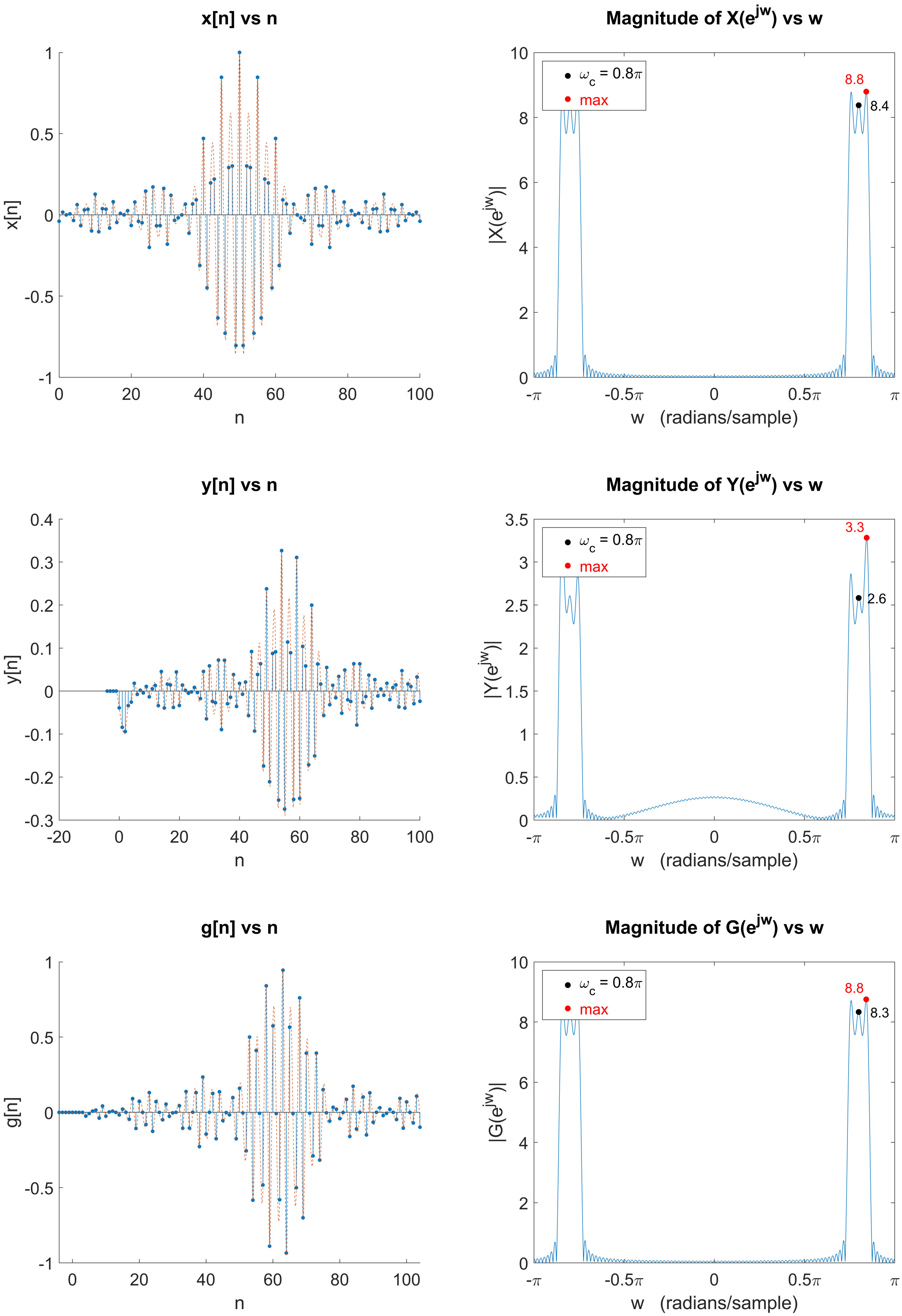
$$\omega_c = 0.4\pi \quad \text{rad/sample}$$



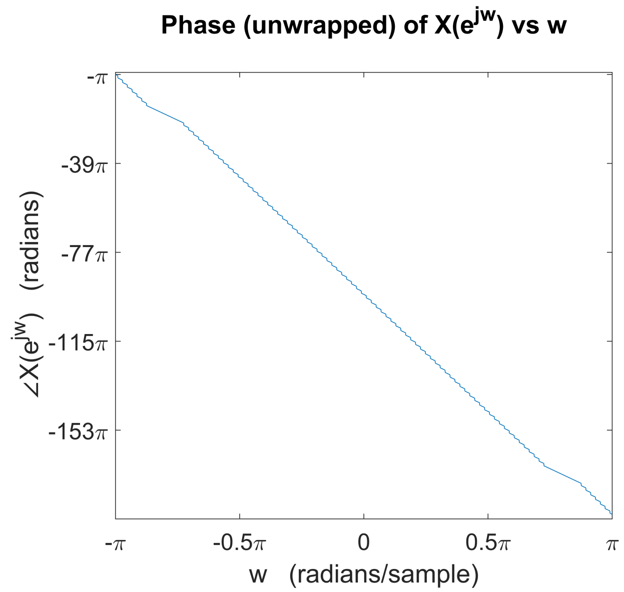


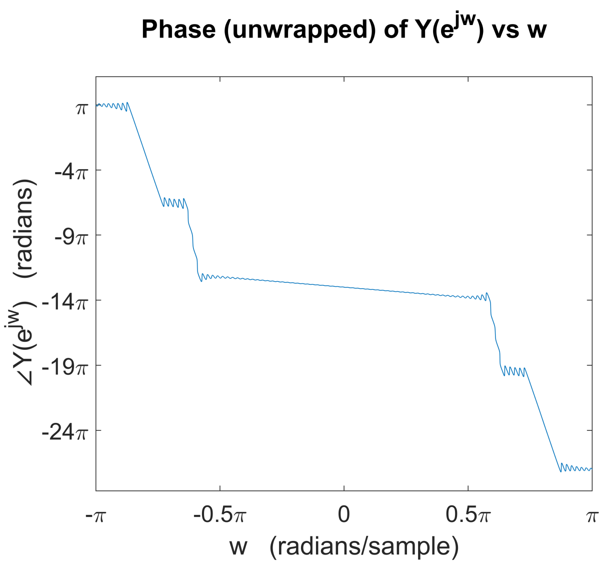


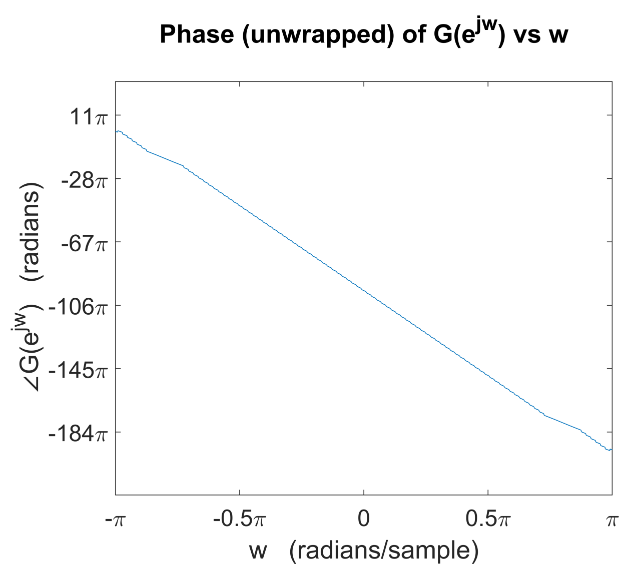
$$\omega_c = 0.8\pi \quad \text{rad/sample}$$



$\omega_c = 0.8\pi$ rad/sample







Comparing the magnitude plots of $X(e^{j\omega})$ and $G(e^{j\omega})$, it can be seen that they approximately match. Hence, any magnitude distortion experienced in $Y(e^{j\omega})$ from $H(z)$ is compensated by $H_{\text{inv}}(z)$. A summary of the magnitudes (maximum and at ω_c) for the input and intermediate/final output of the overall system is shown below.

	$\lvert X(e^{j\omega})\rvert$		$\lvert Y(e^{j\omega})\rvert$		$\lvert G(e^{j\omega})\rvert$	
ω_c (rad/sample)	at ω_c	max	at ω_c	max	at ω_c	max
0.1π	8.4	8.8	86.1	93.4	8.4	8.8
0.2π	8.4	8.8	70.8	81.5	8.4	8.8
0.4π	8.4	8.8	28	38	8.4	8.8
0.8π	8.4	8.8	2.6	3.3	8.3	8.8

	$\frac{\lvert Y(e^{j\omega})\rvert}{\lvert X(e^{j\omega})\rvert} = \lvert H(e^{j\omega})\rvert$		$\frac{\lvert G(e^{j\omega})\rvert}{\lvert Y(e^{j\omega})\rvert} = \lvert H_{inv}(e^{j\omega})\rvert$		$\frac{\lvert G(e^{j\omega})\rvert}{\lvert X(e^{j\omega})\rvert} = \lvert H(e^{j\omega})\rvert \lvert H_{inv}(e^{j\omega})\rvert$	
ω_c (rad/sample)	at ω_c	max	at ω_c	max	at ω_c	max
0.1π	10.3	10.6	0.1	0.1	1	1
0.2π	8.4	9.26	0.1	0.1	1	1
0.4π	3.3	4.3	0.3	0.2	1	1
0.8π	0.3	0.38	3.2	2.7	0.99	1

The magnitude ratios for input and output of $H_{\text{inv}}(z)$ can be verified by noting its gains from the magnitude plot at respective center frequencies (ω_c):

$$
\begin{aligned}
\lvert H_{\text{inv}}(\ 0.1\pi\ \text{rad/sample}\ )\rvert &\approx 0.1 \approx \frac{1}{\lvert H(\ 0.1\pi\ \text{rad/sample}\ )\rvert} \approx \frac{1}{10.2} \\
\lvert H_{\text{inv}}(\ 0.2\pi\ \text{rad/sample}\ )\rvert &\approx 0.1 \approx \frac{1}{\lvert H(\ 0.2\pi\ \text{rad/sample}\ )\rvert} \approx \frac{1}{8.3} \\
\lvert H_{\text{inv}}(\ 0.4\pi\ \text{rad/sample}\ )\rvert &\approx 0.3 \approx \frac{1}{\lvert H(\ 0.4\pi\ \text{rad/sample}\ )\rvert} \approx \frac{1}{3.2} \\
\lvert H_{\text{inv}}(\ 0.8\pi\ \text{rad/sample}\ )\rvert &\approx 3.3 \approx \frac{1}{\lvert H(\ 0.8\pi\ \text{rad/sample}\ )\rvert} \approx \frac{1}{0.3}
\end{aligned}
$$

These values from the magnitude plot match with the respective values of $\lvert H_{\text{inv}}(e^{j\omega})\rvert$ computed in the table above. Hence, the following equation summarizes the magnitude variation from input to output:

$$
\begin{aligned}
\lvert G(e^{j\omega})\rvert &= \lvert Y(e^{j\omega})\rvert\,\lvert H_{\text{inv}}(e^{j\omega})\rvert = \lvert X(e^{j\omega})\rvert\,\lvert H(e^{j\omega})\rvert\,\lvert H_{\text{inv}}(e^{j\omega})\rvert = \lvert X(e^{j\omega})\rvert\,\lvert H(e^{j\omega})\rvert\frac{1}{\lvert H(e^{j\omega})\rvert} \\
&= \lvert X(e^{j\omega})\rvert
\end{aligned}
$$

But the above relation is not true for the phase responses, $\angle G(e^{j\omega})$ and $\angle X(e^{j\omega})$, because $H_{\text{inv}}(z)$ does not compensate for the phase distortion (i.e. delay) introduced by $H(z)$, rather it introduces further delays in the output. This delay varies with the frequency-content of the input, and hence the output $g[n]$ does not have the same phase or exact shape as $x[n]$. This can be observed from the plots of the input and output sequences above. So, the phase plots show that,

$$\begin{aligned}\angle G(e^{j\omega}) &= \angle Y(e^{j\omega}) + \angle H_{\text{inv}}(e^{j\omega}) = \angle X(e^{j\omega}) + \angle H(e^{j\omega}) + \angle H_{\text{inv}}(e^{j\omega}) \\ &\neq \angle X(e^{j\omega})\end{aligned}$$

This is due to the fact that $\angle H_{\text{inv}}(e^{j\omega}) \neq -\angle H(e^{j\omega})$. As a result,

$$g[n] \neq x[n]$$

Rather, $g[n] = x[n] * h[n] * h_{\text{inv}}[n]$.

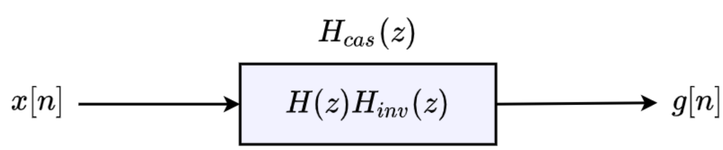
Hence, only the magnitude response of the input is recovered by using the compensator system, and not the entire input itself.

Question (5):

What is the magnitude, phase, group delay of the overall system, i.e., $H(z)$ cascaded with its magnitude inverse.

Answer (5):

We'll represent the overall cascaded system as $H_{\text{cas}}(z)$, given as,



and it will have the following expression,

$$\begin{aligned}H_{\text{cas}}(z) &= H(z)H_{\text{inv}}(z) = H(z)\frac{1}{H_{\text{min}}(z)} \\ &= \frac{(z-0.9e^{j\,0.6\pi})(z-0.9e^{-j\,0.6\pi})(z-1.25e^{j\,0.8\pi})(z-1.25e^{-j\,0.8\pi})}{z^4} \\ &\quad \times \frac{(z-0.9e^{j\,0.6\pi})^{-1}(z-0.9e^{-j\,0.6\pi})^{-1}z^4}{(1.25)^2(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}\end{aligned}$$

$$H_{\text{cas}}(z) = \frac{(z-1.25e^{j\,0.8\pi})(z-1.25e^{-j\,0.8\pi})}{(1.25)^2(z-0.8e^{-j\,0.8\pi})(z-0.8e^{j\,0.8\pi})}$$

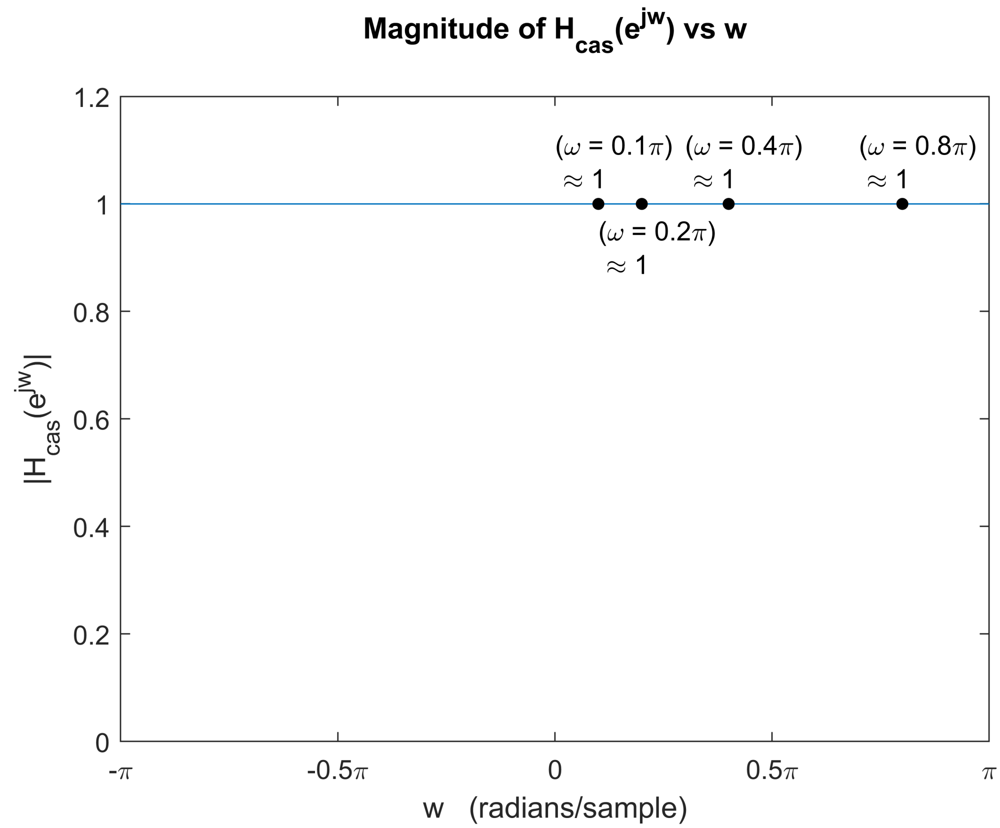
The code, along with the results (inline), to compute & plot various responses from $H_{\text{cas}}(z)$ expression above, starts below.

```
% <task> plot magnitude and phase response of cascaded system Hcas(z)

% define z-domain expression for Hcas(z) = H(z).Hinv(z)
Hcasz = Hz*Hinvz;

% display Hcas(z) expression
disp('Hcas(z) = '); disp(vpa(Hcasz, 4))
```

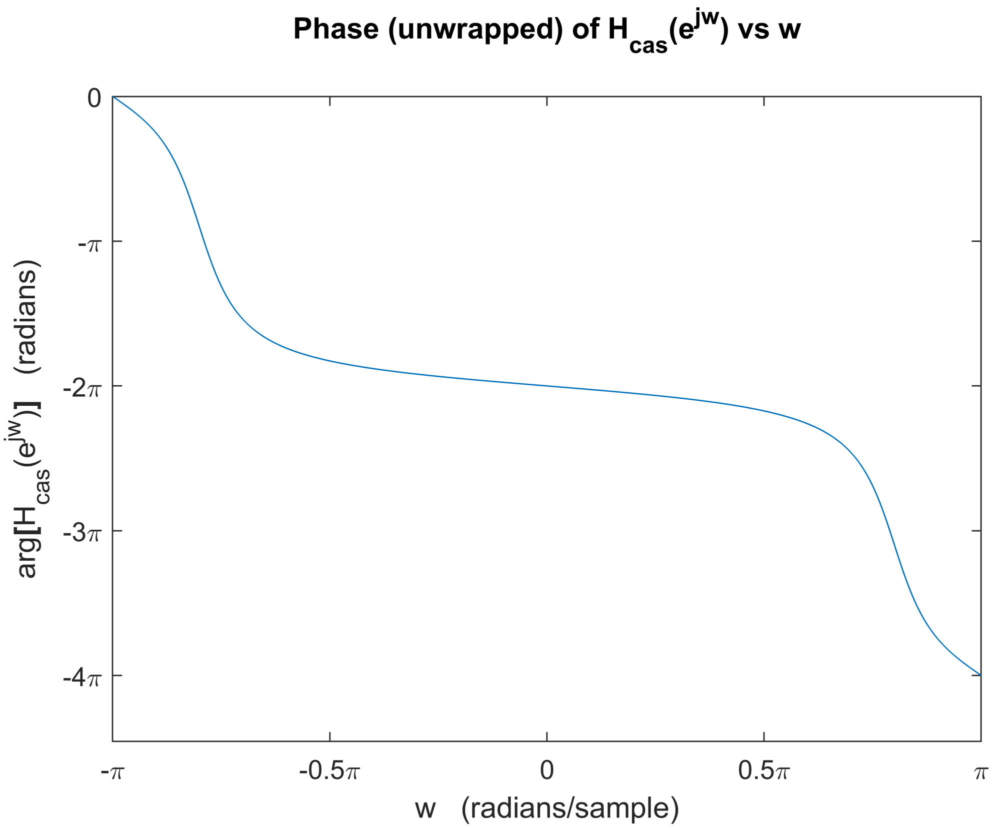
```
Hcas(z) =
```

$$\frac{(z + 1.011 - 0.7347\,\mathrm{i})\ (z + 1.011 + 0.7347\,\mathrm{i})\ (z + 0.2781 - 0.856\,\mathrm{i})}{(1.562\,z + 0.4346 - 1.337\,\mathrm{i})\ (z + 0.6472 - 0.4702\,\mathrm{i})\ (z + 0.6472 + 0.4702\,\mathrm{i})}$$

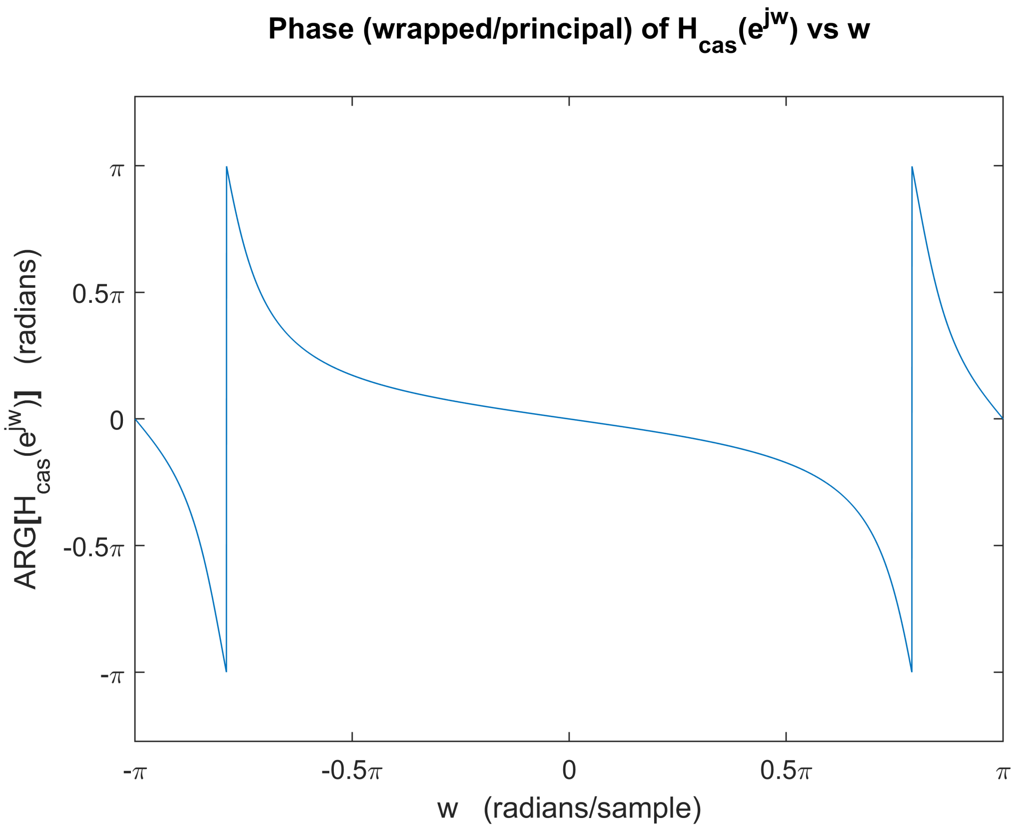
```
% obtain frequency response Hcas(w) expression from Hcas(z)
% by substituting z = e^(jw)
syms w_;
Hcasw = subs(Hcasz, {z}, {exp(1i*w_)});

% compute frequency response Hcas(w) for w = [-pi, pi]
dw = 0.001;
w = -pi:dw:pi;
Hcasw = subs(Hcasw, {w_}, {w});
Hcasw = double(Hcasw);

% obtain magnitude and phase of Hcas(w)
Hcasw_mag = abs(Hcasw);
Hcasw_arg = phase(Hcasw);            % unwrapped phase (in radians)
Hcasw_ARG = angle(Hcasw);            % wrapped phase (in radians)

% frequencies to be noted/marked on the response plots
wvals = [0.1, 0.2, 0.4, 0.8]*pi;

% plot magnitude of Hcas(w) vs w
fig = figure;
plot(w, Hcasw_mag);
xlabel('w   (radians/sample)');
ylabel('|H_{cas}(e^{jw})|');
title({'Magnitude of H_{cas}(e^{jw}) vs w';''});
setDTFTradialAxis(1);
markOnPlot(wvals, w, Hcasw_mag, dw, -0.1*pi, [1 -1 1 1]*0.08, ...
    {'(\omega = ', '\pi)', ''}, pi);
```



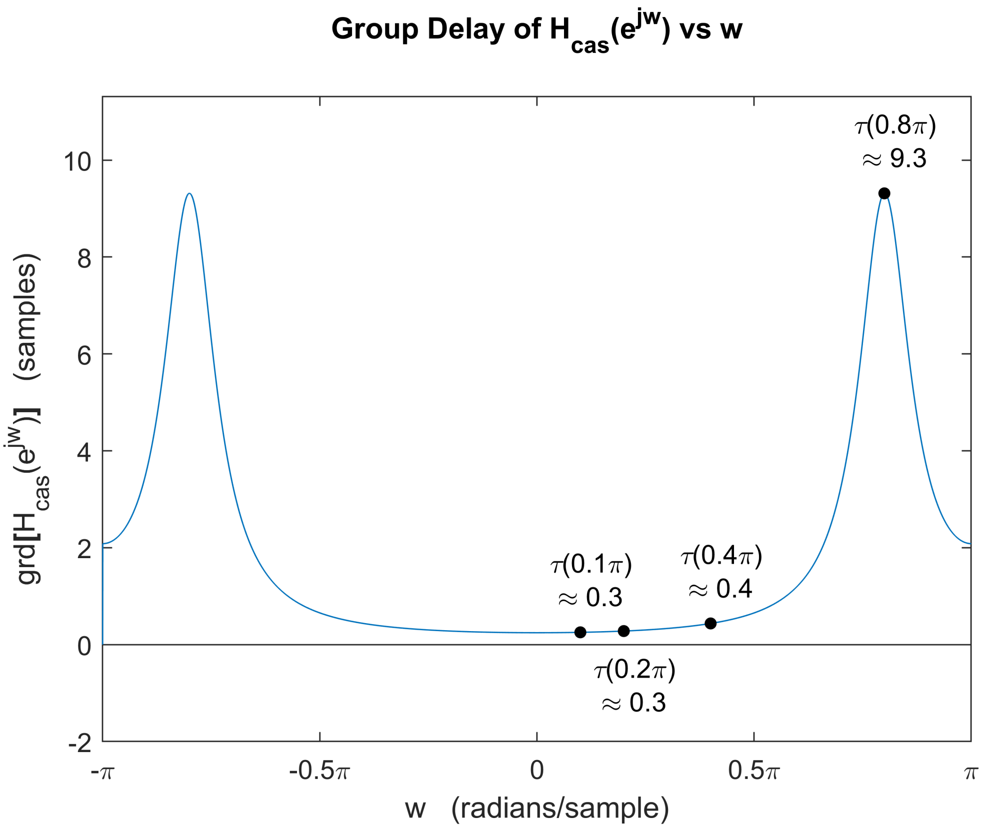

```
% plot phase (unwrapped) of Hcas(w) vs w
fig = figure;
plot(w, Hcasw_arg);
xlabel('w   (radians/sample)');
ylabel('arg{\bf[}H_{cas}(e^{jw}){\bf]}   (radians)');
title({'Phase (unwrapped) of H_{cas}(e^{jw}) vs w';''});
setDTFTradialAxis(1, 1, 1);
```




```
% plot phase (wrapped/principal) of Hcas(w) vs w
fig = figure;
plot(w, Hcasw_ARG);
xlabel('w   (radians/sample)');
ylabel('ARG{\bf[}H_{cas}(e^{jw}){\bf]}   (radians)');
title({'Phase (wrapped/principal) of H_{cas}(e^{jw}) vs w';''});
setDTFTradialAxis(1, 0.5, 1);
```




```
% <task> plot group-delay of cascaded system Hcas(z)

dw = (w(end)-w(1))/(length(w)-1);           % compute the w-step-size 'dw' used in w-array

% obtain group-delay of Hcas(w)
Hcasw_grd = -([Hcasw_arg 0]-[0 Hcasw_arg])/dw;
Hcasw_grd = Hcasw_grd(1:(length(Hcasw_grd)-1));

% plot group-delay of Hcas(w) vs w
fig = figure;
plot(w, Hcasw_grd);
xlabel('w   (radians/sample)'); ylabel('grd{\bf[}H_{cas}(e^{jw}){\bf]}   (samples)');
title({'Group Delay of H_{cas}(e^{jw}) vs w';''});
setDTFTradialAxis(1); ylim([min(Hcasw_grd)-2 max(Hcasw_grd)+2]);
markOnPlot(wvals, w, Hcasw_grd, dw, -0.07*pi, [1 -1 1 1]*1.1, {'\tau(', '\pi)', ''}, pi);
```

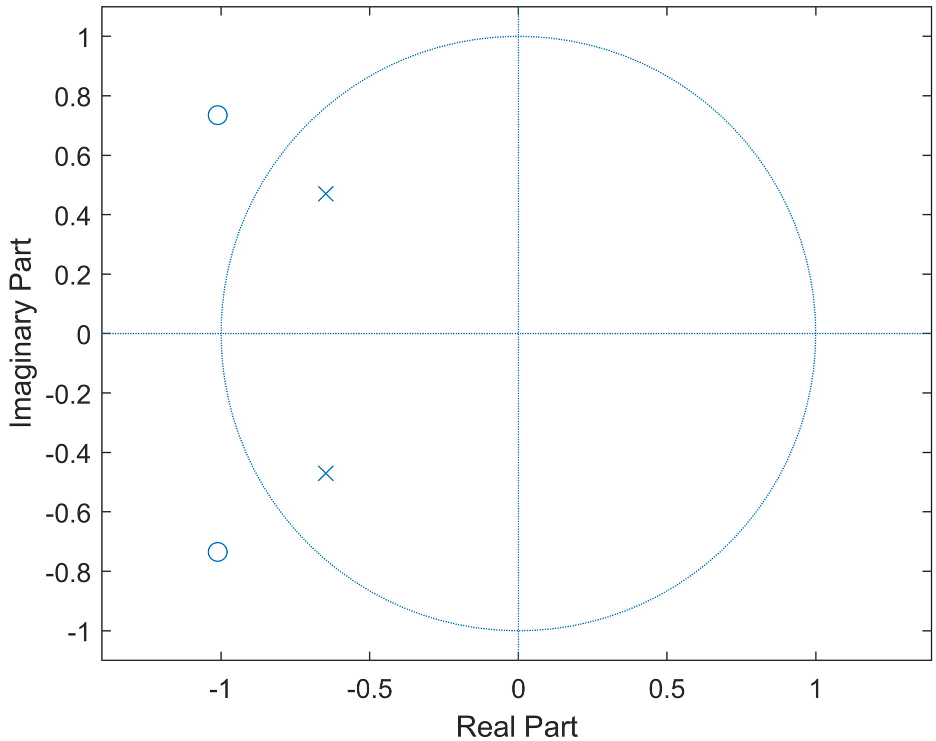



```
% <task> plot pole-zero map of cascaded system Hcas(z)

% extract CCDE coefficients (b's and a's) from Hcas(z) expression
[N, D] = numden(Hcasz);
b = coeffs(N, 'All');   a = coeffs(D, 'All');
b = double(b./a(1));    a = double(a./a(1));
disp('b ='); disp(b);   disp('a ='); disp(a);
```

```
b =
    0.6400    1.2944    1.0000

a =
    1.0000    1.2944    0.6400
```

```
% plot the pole-zero map of Hcas(z) using extracted b's and a's
figure;
zplane(b, a);
title({'Pole-Zero Plot of H_{cas}(z) in z-plane';''});
```

Pole-Zero Plot of $H_{cas}(z)$ in z-plane



The pole-zero map above matches with the z-domain expression $H_{\text{cas}}(z)$ obtained at start of this section (Q5).

The magnitude plot $|H_{\text{cas}}(e^{j\omega})|$ (Page 33) shows that the gain is a constant 1 at all frequencies, that is, $H_{\text{cas}}(z)$ is an all-pass system. So, any input passed through the overall system experiences no magnitude distortion. Although, input will experience phase-distortion (delay in samples) affecting different frequencies differently thereby distorting the signal shape at output.

Question (6):

How much delay do you observe corresponding to different ω_c? Can you explain the delay in the output to the group delay of the overall system $H_{\text{cas}}(z)$ ($H(z)$ cascaded with its inverse).

Answer (6):

From the group-delay plot $\tau(\omega) = \text{grd}[H_{\text{cas}}(z)]$, the delays introduced by the overall system, for specified ω_c values, can be noted as follows:

$$\begin{aligned}
\tau(\ 0.1\pi \ \text{rad/sample}\ ) &\approx 0.3 \ \text{samples delay} \\
\tau(\ 0.2\pi \ \text{rad/sample}\ ) &\approx 0.3 \ \text{samples delay} \\
\tau(\ 0.4\pi \ \text{rad/sample}\ ) &\approx 0.4 \ \text{samples delay} \\
\tau(\ 0.8\pi \ \text{rad/sample}\ ) &\approx 9.3 \ \text{samples delay}
\end{aligned}$$

Different delays occur in input for its different frequency components (due to it being a collection/pulse of frequencies). Since the frequency content in $x[n]$ (for any ω_c) is banded together around ω_c, we can get a rough idea of the overall delay in output $g[n]$ by looking at the delay values for its center frequency ω_c (delays noted above using the group-delay plot).

Comparing the plots of $x[n]$ and $g[n]$, for various ω_c, following overall delays can be noted:

$$\begin{aligned}
&\text{for } \ \omega_c = 0.1\pi \ \text{ rad/sample} &&\Longrightarrow\ g[n] \text{ is delayed by about } \ \approx\ 2-3 \ \text{ samples}\\
&\text{for } \ \omega_c = 0.2\pi \ \text{ rad/sample} &&\Longrightarrow\ g[n] \text{ is delayed by about } \ \approx\ 2-3 \ \text{ samples}\\
&\text{for } \ \omega_c = 0.4\pi \ \text{ rad/sample} &&\Longrightarrow\ g[n] \text{ is delayed by about } \ \approx\ 3-4 \ \text{ samples}\\
&\text{for } \ \omega_c = 0.8\pi \ \text{ rad/sample} &&\Longrightarrow\ g[n] \text{ is delayed by about } \ \approx\ 10 \ \text{ samples}
\end{aligned}$$

compared to $x[n]$.

This agrees with the group-delays noted for the center-frequencies $\omega_c = 0.1\pi,\ 0.2\pi,\ 0.4\pi,\ 0.8\pi \ \text{ rad/sample}$ above, that is, input with frequencies around $0.1\pi,\ 0.2\pi,\ 04.\pi \text{ rad/sample}$ incurs a small delay (2 or 3 samples), while input with frequencies around $0.8\pi \text{ rad/sample}$ incurs a larger delay ($\approx 10 \text{ samples}$).

Functions used in the tasks are provided starting on the next page.

Functions used in the tasks

markOnPlot()

Used in Task 1 (Q1, Q4.a, Q5) — to mark given x-axis datapoints on given plot curve.

```
% <function>
% marks given dat(x) values on plot.
%
% <syntax>
% markOnPlot(xvals, x, dat, tol, xoff, yoff, <txt>, <xdiv>, <rnd>)
%
% <I/O>
% xvals, x, dat = values on x-axis to mark, x-axis array, y-axis dat(x) array
% tol = tolerance in finding x-value in x-axis array
% xoff, yoff = horizontal, vertical offset of text (value/array)
% txt = cell structure specifying text to show around values

function markOnPlot(xvals, x, dat, tol, xoff, yoff, txt, xdiv, rnd)

    if length(xoff)==1; xoff = repelem(xoff, length(xvals));
    elseif isempty(xoff); xoff = repelem(0, length(xvals)); end
    if length(yoff)==1; yoff = repelem(yoff, length(xvals));
    elseif isempty(yoff); yoff = repelem(0, length(xvals)); end
    txt1 = 'y('; txt2 = ')'; txt_eq = '\approx'; newline = 0;
    if nargin >= 7
        txt1 = txt{1}; txt2 = txt{2};
        if length(txt) >= 3
            if isequal(txt{3}, ''); newline = 1; else; txt_eq = txt{2}; end
        end
    end
    if ~(nargin >= 8); xdiv = 1; end; if ~(nargin == 9); rnd = 1; end

    hold on;
    for i = 1:length(xvals)
        yval = dat(find(x>=xvals(i)-tol & x<=xvals(i)+tol, 1));
        stem(xvals(i), yval, 'k', 'filled', 'LineStyle', 'none', 'MarkerSize', 4);
        if newline
          text(xvals(i)+xoff(i), yval+yoff(i), {[txt1, num2str(xvals(i)/xdiv), txt2]; ...
            [' ', txt_eq,' ', num2str(round(yval, rnd))]});
        else
            text(xvals(i)+xoff(i), yval+yoff(i), [txt1, num2str(xvals(i)/xdiv), txt2, ...
                ' ', txt_eq,' ', num2str(round(yval, rnd))]);
        end
    end
    hold off;

end
```

plot2SeqDTFT()

Used in Task 1 (Q3) — to plot 2 sequences and their DTFTs (magnitude and unwrapped-phase response).

```
% <function>
% plots given sequence(s) against given n values and their given DTFT
% (magnitude and phase response) against given w values.
%
% <syntax>
% plot2SeqDTFT(n, x, wc, w, Xw_mag, Xw_arg, Nn, Nw, off, ...
%     <ny>, <y>, <Yw_mag>, <Yw_arg>, <Nny>, <Nwy>, <align1>, <align2>);
%
% <I/O>
% x = sequence x[n] array
% wc = center frequency of x[n]
% w = DT frequencies array [-pi, pi]
% Xw_mag, Xw_arg = DTFT magnitude, phase (unwrapped)
% Nn = name of sequence x[n] for label/title in plot
% Nw = name of DTFT X(w) for label/title in plot
% off = offsets for texts in form of array
% y = optional sequence y[n] array
% Yw_mag, Yw_arg = optional y[n] DTFT magnitude, phase (unwrapped)
% Nny = name of optional sequence y[n] for label/title in plot
% Nwy = name of optional y[n] DTFT for label/title in plot

function plot2SeqDTFT(n, x, wc, w, Xw_mag, Xw_arg, Nn, Nw, off, ...
    ny, y, Yw_mag, Yw_arg, Nny, Nwy, align1, align2);

    if length(off) < 4; off = [off repelem(0, 4-length(off))]; end
    plt2 = 0;
    if nargin >= 10
        plt2 = 1;
        if length(off) < 8; off = [off repelem(0, 8-length(off))]; end
    end
    if ~(nargin >= 16); align1 = 'left'; end; if ~(nargin == 17); align2 = 'left'; end
    lb2 = round(length(w)/2); w2 = w(lb2:end);

    % plot sequence(s)
    fig = figure;
    if plt2
        set(fig,'Units','normalized','Position',[0 0 1.4 1]);
        subplot(1, 2, 1);
    end
        % plot x[n] vs n
        hold on;
            stem(n, x, 'filled', 'MarkerSize', 3);
            n2 = min(n):1/1000:max(n);
            plot(n2, interp1(n,x,n2,'spline'), '--');
        hold off;
        xlabel('n'); ylabel([Nn, '[n]']);
```

```
    if plt2
        title({['{\bf', Nn, '[n] vs n}'];''}, 'FontSize', 16);
        ax = gca; ax.FontSize = 16;
    else
        title({'';['{\bf', Nn, '[n] vs n}'];''});
    end
if plt2
    subplot(1, 2, 2);
    % plot y[n] vs n
    hold on;
        stem(ny, y, 'filled', 'MarkerSize', 3);
        n2 = min(ny):1/1000:max(ny);
        plot(n2, interp1(ny,y,n2,'spline'), '--');
    hold off;
    xlabel('n'); ylabel([Nny, '[n]']);
    xlim([min(n) max(n)]);
    title({['{\bf', Nny, '[n] vs n}'];''}, 'FontSize', 16);
    ax = gca; ax.FontSize = 16;
end

% plot magnitude and phase of DTFT X(w) vs w
fig = figure; set(fig,'Units','normalized','Position',[0 0 1.4 1]);
subplot(1, 2, 1);
plot(w, Xw_mag);
xlabel('w   (radians/sample)'); ylabel(['|', Nw, '(e^{jw})|']);
title({'';['{\bfMagnitude of ', Nw, '(e^{jw}) vs w}'];''}, 'FontSize', 16);
setDTFTradialAxis(1);
ax = gca; ax.FontSize = 16;
hold on;
    yval = Xw_mag(find(w>=wc-0.01 & w<=wc+0.01, 1));
    l1 = stem(wc, yval, 'k', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
    text(wc+off(1), yval+off(2), num2str(round(yval, 1)), ...
        'FontSize', 13, 'HorizontalAlignment', align1);
    [yval, yinx] = max(Xw_mag(lb2:end)); wval = w2(yinx);
    l2 = stem(wval, yval, 'r', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
    text(wval+off(3), yval+off(4), num2str(round(yval, 1)), ...
        'Color', 'r', 'FontSize', 13, 'HorizontalAlignment', align2);
    legend([l1 l2], ['\omega_c = ', num2str(wc/pi), '\pi'], ...
        '{\color{red}max}', 'Location', 'northwest');
hold off;
subplot(1, 2, 2);
plot(w, Xw_arg);
xlabel('w   (radians/sample)'); ylabel(['∠', Nw, '(e^{jw})   (radians)']);
title({'';['{\bfPhase (unwrapped) of ', Nw, '(e^{jw}) vs w}'];''}, ...
    'FontSize', 16);
step = round(max(abs([max(Xw_arg) min(Xw_arg)]))/pi*0.2);
setDTFTradialAxis(1, step, 1);
ax = gca; ax.FontSize = 16;

if plt2
    % plot magnitude and phase of DTFT Y(w) vs w
    fig = figure; set(fig,'Units','normalized','Position',[0 0 1.4 1]);
    subplot(1, 2, 1);
    plot(w, Yw_mag);
```

```
        xlabel('w   (radians/sample)'); ylabel(['|', Nwy, '(e^{jw})|']);
        title({'';['{\bfMagnitude of ', Nwy, '(e^{jw}) vs w}'];''}, 'FontSize', 16);
        setDTFTradialAxis(1);
        ax = gca; ax.FontSize = 16;
        hold on;
            yval = Yw_mag(find(w>=wc-0.01 & w<=wc+0.01, 1));
            l1 = stem(wc, yval, 'k', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
            text(wc+off(5), yval+off(6), num2str(round(yval, 1)), ...
                'FontSize', 13, 'HorizontalAlignment', align1);
            [yval, yinx] = max(Yw_mag(lb2:end)); wval = w2(yinx);
            l2 = stem(wval, yval, 'r', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
            text(wval+off(7), yval+off(8), num2str(round(yval, 1)), ...
                'Color', 'r', 'FontSize', 13, 'HorizontalAlignment', align2);
            legend([l1 l2], ['\omega_c = ', num2str(wc/pi), '\pi'], ...
                '{\color{red}max}', 'Location', 'northwest');
        hold off;
        subplot(1, 2, 2);
        plot(w, Yw_arg);
        xlabel('w   (radians/sample)'); ylabel(['∠', Nwy, '(e^{jw})   (radians)']);
        title({'';['{\bfPhase (unwrapped) of ', Nwy, '(e^{jw}) vs w}'];''}, ...
            'FontSize', 16);
        step = round(max(abs([max(Yw_arg) min(Yw_arg)]))/pi*0.2);
        setDTFTradialAxis(1, step, 1);
        ax = gca; ax.FontSize = 16;
    end

end
```

plot3SeqDTFT()

Used in Task 1 (Q4.b) — to plot 3 sequences and their DTFTs (magnitude and unwrapped-phase response).

```
% <function>
% plots given sequence(s) against given n values and their given DTFT
% (magnitude and phase response) against given w values.
%
% <syntax>
% plot3SeqDTFT(n, a, wc, w, Aw_mag, Nn, Nw, Aw_arg, <off>, <align>)
%
% <I/O>
% a = {x[n], y[n], g[n]}
% wc = center frequency of a[n]
% w = DT frequencies array [-pi, pi]
% Aw_mag = {Xw_mag, Yw_mag, Gw_mag}
% Nn = {Nnx, Nny, Nng}
% Nw = {Nwx, Nwy, Nwg}
% Aw_arg = optional {Xw_arg, Yw_arg, Gw_arg}
% off = optional {offX, offY, offG}
% align = optional {alignWC, alignMX}
```

```
function plot3SeqDTFT(n, a, wc, w, Aw_mag, Nn, Nw, Aw_arg, off, align)

    if ~(nargin >= 8); Aw_arg = {}; end
    if ~(nargin >= 9); off = {zeros(1, 4), zeros(1, 4), zeros(1, 4)}; end
    if ~(nargin == 10); align = {'left', 'left'}; end
    lb2 = round(length(w)/2); w2 = w(lb2:end);

    for i = 1:length(n)
        % plot sequence a[n] and DTFT A(w) magnitude
        fig = figure;
        set(fig,'Units','normalized','Position',[0 0 1.4 1]);
        subplot(1, 2, 1);
        % plot a[n] vs n
        hold on;
            stem(n{i}, a{i}, 'filled', 'MarkerSize', 3);
            n2 = min(n{i}):1/1000:max(n{i});
            plot(n2, interp1(n{i},a{i},n2,'spline'), '--');
        hold off;
        xlabel('n');
        ylabel([Nn{i}, '[n]']);
        if (i == 1)
            title({['{\bf', Nn{i}, '[n] vs n}'];''}, 'FontSize', 16);
        else
            title({'';['{\bf', Nn{i}, '[n] vs n}'];''}, 'FontSize', 16);
        end
        ax = gca; ax.FontSize = 16;
        subplot(1, 2, 2);
        % plot magnitude of A(w) vs w
        plot(w, Aw_mag{i});
        xlabel('w   (radians/sample)');
        ylabel(['|', Nw{i}, '(e^{jw})|']);
        if (i == 1)
            title({['{\bfMagnitude of ', Nw{i}, '(e^{jw}) vs w}'];''}, 'FontSize', 16);
        else
            title({'';['{\bfMagnitude of ', Nw{i}, '(e^{jw}) vs w}'];''}, ...
                'FontSize', 16);
        end
        setDTFTradialAxis(1);
        ax = gca; ax.FontSize = 16;
        % mark wc and max points on plot
        hold on;
            yval = Aw_mag{i}(find(w>=wc-0.01 & w<=wc+0.01, 1));
            l1 = stem(wc, yval, 'k', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
            text(wc+off{i}(1), yval+off{i}(2), num2str(round(yval, 1)), ...
                'FontSize', 13, 'HorizontalAlignment', align{1});
            [yval, yinx] = max(Aw_mag{i}(lb2:end)); wval = w2(yinx);
            l2 = stem(wval, yval, 'r', 'filled', 'LineStyle', 'none', 'MarkerSize', 5);
            text(wval+off{i}(3), yval+off{i}(4), num2str(round(yval, 1)), ...
                'Color', 'r', 'FontSize', 13, 'HorizontalAlignment', align{2});
            legend([l1 l2], ['\omega_c = ', num2str(wc/pi), '\pi'], ...
                '{\color{red}max}', 'Location', 'northwest');
        hold off;
    end
```

```
    for i = 1:length(Aw_arg)
        % plot DTFT A(w) phase (unwrapped)
        fig = figure;
        set(fig,'Units','normalized','Position',[0 0 1.4 1]);
        subplot(1, 2, 1)
        plot(w, Aw_arg{i});
        xlabel('w   (radians/sample)');
        ylabel(['∠', Nw{i}, '(e^{jw})   (radians)']);
        if (i == 1)
            title({['{\bfPhase (unwrapped) of ', Nw{i}, '(e^{jw}) vs w}'];''}, ...
                'FontSize', 16);
        else
            title({'';['{\bfPhase (unwrapped) of ', Nw{i}, '(e^{jw}) vs w}'];''}, ...
                'FontSize', 16);
        end
        step = round(max(abs([max(Aw_arg{i}) min(Aw_arg{i})]))/pi*0.2);
        setDTFTradialAxis(1, step, 1);
        ax = gca; ax.FontSize = 16;
    end

end
```

```
'End of Report';
```