## 1. Fourier transform

A linear and time-invariant system can be represented using its response to the unit sample sequence. This response, called the unit impulse response `h(n)`, allows us to compute the system response to any arbitrary input `x(n)` using the linear convolution:

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = h(n) * x(n)$$

This convolution representation is based on the fact that any signal can be represented by a linear combination of scaled and delayed unit samples.

When the system is linear and time-invariant, only one representation stands out as the most useful. It is based on the complex exponential signal set $\{e^{j\omega n}\}$ and is called the *discrete-time Fourier transform*, which is given by

$$X(e^{j\omega}) \triangleq \mathcal{F}[x(n)] = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \qquad \textbf{(3.1)}$$

The inverse discrete-time Fourier transform (IDTFT) of $X(e^{j\omega})$ is given by

$$x(n) \triangleq \mathcal{F}^{-1}[X(e^{j\omega})] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega \qquad \textbf{(3.2)}$$

The operator transforms a discrete signal `x(n)` into a complex-valued continuous function $X(e^{j\omega})$ of real variable ω, called a digital frequency, which is measured in radians/sample.

### Task 1.

Determine the discrete-time Fourier transform of $x(n) = (0.5)^n u(n)$.

**Solution** The sequence $x(n)$ is absolutely summable; therefore its discrete-time Fourier transform exists.

$$X(e^{j\omega}) = \sum_{-\infty}^{\infty} x(n)e^{-j\omega n} = \sum_{0}^{\infty} (0.5)^n e^{-j\omega n}$$

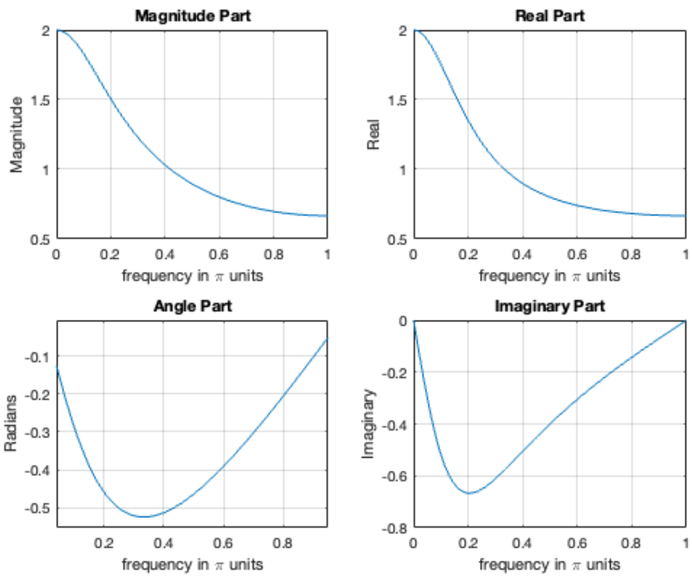
$$= \sum_{0}^{\infty} (0.5e^{-j\omega})^n = \frac{1}{1-0.5e^{-j\omega}} = \frac{e^{j\omega}}{e^{j\omega}-0.5}$$

□

Evaluate $X(e^{j\omega})$ at 501 equispaced points between $[0, \pi]$ and plot its magnitude, angle, real, and imaginary parts, use MATLAB script:

```
w = [0:1:500]*pi/500; % [0, pi] axis divided into 501 points.
X = exp(j*w) ./ (exp(j*w) - 0.5*ones(1,501));
magX = abs(X); angX = angle(X); realX = real(X); imagX = imag(X);
subplot(2,2,1); plot(w/pi,magX); grid
xlabel('frequency in \pi units'); title('Magnitude Part');
ylabel('Magnitude')
subplot(2,2,3); plot(w/pi,angX); grid
xlabel('frequency in \pi units'); title('Angle Part'); ylabel('Radians')
subplot(2,2,2); plot(w/pi,realX); grid
xlabel('frequency in \pi units'); title('Real Part'); ylabel('Real')
subplot(2,2,4); plot(w/pi,imagX); grid
xlabel('frequency in \pi units'); title('Imaginary Part');
ylabel('Imaginary')
```

The resulting plots are shown below. Note that we divided the `w` array by π before plotting so that the frequency axes are in the units of π and therefore easier to read. This practice is strongly recommended.




**To do:**

- Tape code to Matlab, explain operation in the second line, run the code.
- Plot figures presenting X(e$^{j\omega}$)

If $x(n)$ is of finite duration, then MATLAB can be used to compute $X(e^{j\omega})$ numerically at any frequency $\omega$. The approach is to implement (3.1) directly. If, in addition, we evaluate $X(e^{j\omega})$ at equispaced frequencies between $[0, \pi]$, then (3.1) can be implemented as a *matrix-vector multiplication* operation. To understand this, let us assume that the sequence $x(n)$ has $N$ samples between $n_1 \le n \le n_N$ (i.e., not necessarily between $[0, N-1]$) and that we want to evaluate $X(e^{j\omega})$ at

$$\omega_k \triangleq \frac{\pi}{M}k, \quad k = 0, 1, \ldots, M$$

which are $(M+1)$ equispaced frequencies between $[0, \pi]$. Then (3.1) can be written as

$$X(e^{j\omega_k}) = \sum_{\ell=1}^{N} e^{-j(\pi/M)kn_\ell} x(n_\ell), \quad k = 0, 1, \ldots, M$$

When $\{x(n_\ell)\}$ and $\{X(e^{j\omega_k})\}$ are arranged as *column* vectors $\mathbf{x}$ and $\mathbf{X}$, respectively, we have

$$\mathbf{X} = \mathbf{W}\mathbf{x} \tag{3.3}$$

where $\mathbf{W}$ is an $(M+1) \times N$ matrix given by

$$\mathbf{W} \triangleq \left\{ e^{-j(\pi/M)kn_\ell};\ n_1 \le n \le n_N, \quad k = 0, 1, \ldots, M \right\}$$

In addition, if we arrange $\{k\}$ and $\{n_\ell\}$ as *row* vectors $\mathbf{k}$ and $\mathbf{n}$ respectively, then

$$\mathbf{W} = \left[ \exp\left( -j\frac{\pi}{M}\mathbf{k}^T\mathbf{n} \right) \right]$$

In MATLAB we represent sequences and indices as row vectors; therefore taking the transpose of (3.3), we obtain

$$\mathbf{X}^T = \mathbf{x}^T \left[ \exp\left( -j\frac{\pi}{M}\mathbf{n}^T\mathbf{k} \right) \right] \tag{3.4}$$

Note that $\mathbf{n}^T\mathbf{k}$ is an $N \times (M+1)$ matrix. Now (3.4) can be implemented in MATLAB as follows.

```
>> k = [0:M]; n = [n1:n2];
>> X = x * (exp(-j*pi/M)) .^ (n'*k);
```

**Task 2.**

Numerically compute the discrete-time Fourier transform of the following finite-duration sequence $x(n)$

$x(n) = \{1\ 2\ 3\ 4\ 5\}$

at 501 equispaced frequencies between $[0, \pi]$.
According to the definition 3.1 we get

$$X(e^{j\omega}) = \sum_{-\infty}^{\infty} x(n)\ e^{-j\omega n} = 1 + 2e^{-j\omega} + 3e^{-j2\omega} + 4e^{-j3\omega} + 5e^{-j4\omega}$$

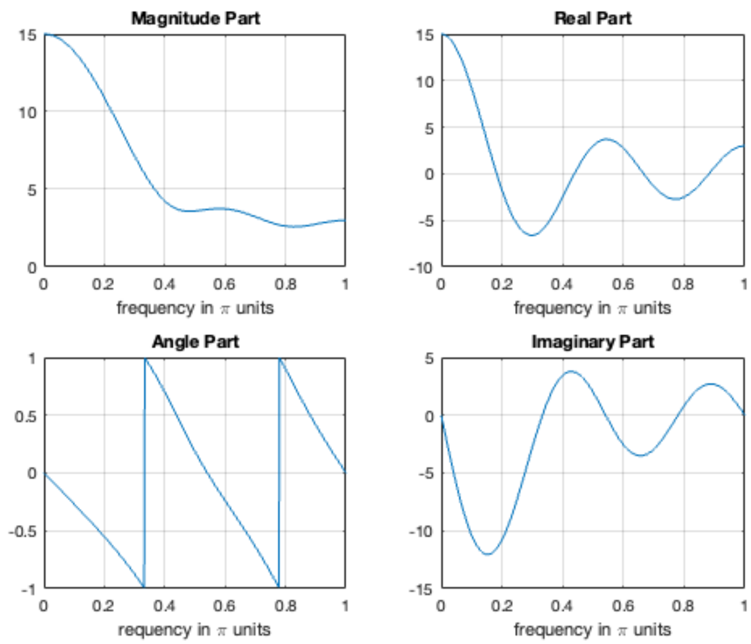
n = 0, 1, 2, 3, 4

Run the following script

```
clear
n = 0:4; x = 1:5; k = 0:500;
w = (pi/500)*k; % Frequency range 0 to pi
X = x * (exp(-j*pi/500)) .^ (n'*k);
magX = abs(X); angX = angle(X);
realX = real(X); imagX = imag(X);
subplot(2,2,1); plot(k/500,magX);grid
xlabel('frequency in \pi units'); title('Magnitude Part')
subplot(2,2,3); plot(k/500,angX/pi);grid
xlabel('requency in \pi units'); title('Angle Part')
subplot(2,2,2); plot(k/500,realX);grid
xlabel('frequency in \pi units'); title('Real Part')
subplot(2,2,4); plot(k/500,imagX);grid
xlabel('frequency in \pi units'); title('Imaginary Part')
```

- Tape code to Matlab, lain operation in the third line, run the code.

The frequency-domain plots are shown below. Note that the **angle plot** is depicted as a **discontinuous function** between $-\pi$ and $\pi$. This is because the angle function in MATLAB computes the principal angle.




**Note** that you can calculate the discrete-time Fourier transform using the `fft` command in Matlab. Note that you have to indicate number of frequencies in the `fft(x,N)` in the **whole frequency range** 0 to $2\pi$.

Run the following script:

```
x = [1:5]; k = 0:500; w = (pi/500)*k;
X = fft(x,1000); % DFT calculated with zero-padding from 0 to 2pi
X=X(1:501); % Plot half spectrum 0 to pi
magX = abs(X); angX = angle(X);
realX = real(X); imagX = imag(X);
subplot(2,2,1); plot(k/500,magX);grid
xlabel('frequency in \pi units'); title('Magnitude Part')
subplot(2,2,3); plot(k/500,angX/pi);grid
xlabel('requency in \pi units'); title('Angle Part')
subplot(2,2,2); plot(k/500,realX);grid
xlabel('frequency in \pi units'); title('Real Part')
subplot(2,2,4); plot(k/500,imagX);grid
xlabel('frequency in \pi units'); title('Imaginary Part')
```

- Tape code to Matlab, explain operation in the second line, run the code.

**To do:**

- Plot figures presenting magnitude and angle of $X(e^{j\omega})$
- Plot figure presenting Real and Imaginary part of $X(e^{j\omega})$ in complex plane (*Re vs Im*)
- Compare the results obtained using both scripts.
- Explain the zero-padding operation.

## 2. Frequency domain representation of LTI systems

Fourier transform representation is the most useful signal representation for LTI systems. An LTI system can be represented in the frequency domain by

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

**Task 3.**

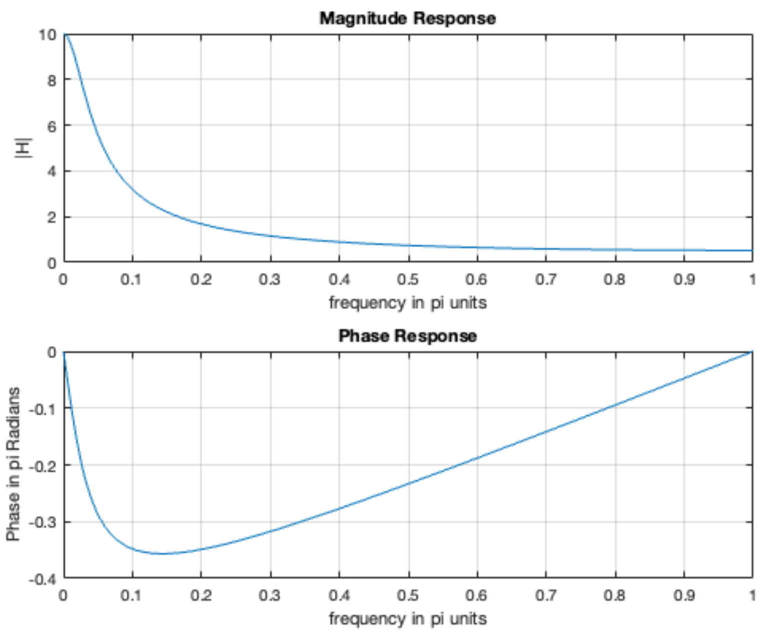
Determine the frequency response $H(e^{j\omega})$ of a system characterized by $h(n) = (0.9)^n u(n)$. Plot the magnitude and the phase responses.

$$H(e^{j\omega}) = \sum_{-\infty}^{\infty} h(n) e^{-j\omega n} = \sum_{0}^{\infty} (0.9)^n e^{-j\omega n}$$

$$= \sum_{0}^{\infty} (0.9 e^{-j\omega})^n = \frac{1}{1 - 0.9e^{-j\omega}}$$

```
w = [0:1:500]*pi/500; % [0, pi] axis divided into 501 points.
H = exp(j*w) ./ (exp(j*w) - 0.9*ones(1,501));
magH = abs(H); angH = angle(H);
subplot(2,1,1); plot(w/pi,magH); grid;
xlabel('frequency in pi units'); ylabel('|H|');
title('Magnitude Response');
subplot(2,1,2); plot(w/pi,angH/pi); grid
xlabel('frequency in pi units'); ylabel('Phase in pi Radians');
title('Phase Response');
```

The resulting plots are shown below. Note that we divided the `w` array by `pi` before plotting so that the frequency axes are in the units of $\pi$ and therefore easier to read.




**To do:**

- Tape code to Matlab, explain operation in the second line, run the code.
- Plot figures presenting magnitude and phase responses of H(jω)