MATLAB TUTORIAL

10/10/2014 - 14/10/2014

Discrete Time Signals and Systems

Some Basic Sequences (.m file 1)

1) Unit Sample Sequence

$$\delta[n] = \begin{cases} 1, & n=0 \\ 0, & n\neq 0 \end{cases} = \{0, \ldots 0, \underset{\substack{\uparrow \\ n=0}}{1}, 0, \ldots 0\}$$

To implement $\delta[n]$ over the interval $[n_1:n_2]$ use the following command:

```
n=[n1:n2];
x=[n==0];
```

2) Unit Step Sequence

$$u[n] = \begin{cases} 1, & n\geq 0 \\ 0, & n<0 \end{cases} = \{\ldots, 0, 0, \underset{\substack{\uparrow \\ n=0}}{1}, 1, 1, \ldots 1, \ldots\}$$

To implement $u[n]$ over the interval $[n_1:n_2]$ use the following command

```
n=[n1:n2];
x=[n>=0];
```

) Real valued Exponential Sequence

$$x[n] = a^n \quad \forall n, \; a \in \mathbb{R}$$

Ex : $x[n] = (0.9)^n$

To implement $x[n]$ over the interval use the following command:

```
n = [n1:n2];
x = (0.9).^n;
```

↳ elementwise operation

4) Complex Valued Exponential Sequence

$$x[n] = e^{(\sigma + j\omega_0)n}, \; \forall n$$

σ : attenuation
ω_0 : frequency (radians per sample)

Ex 1) $x[n] = e^{j\frac{\pi}{4}n}$

```
x = exp(j*pi/4*n);
```

$real(x) = \cos\left(\frac{\pi}{4}n\right)$
$imag(x) = \sin\left(\frac{\pi}{4}n\right)$

→ Sinusoidal sequences

Ex 2) $x[n] = e^{(-0.8 + j\frac{\pi}{4})n}$

```
X = exp((-0.8+j*pi/4)*n);
```

$real(x) = e^{-0.8n} \cos\left(\frac{\pi}{4}n\right)$
$imag(x) = e^{-0.8n} \sin\left(\frac{\pi}{4}n\right)$

Ex 3) $x[n] = e^{(1.2 + j\frac{\pi}{4})n}$

```
x = exp((1.2+j*pi/4)*n);
```

$real(x) = e^{1.2n} \cos\left(\frac{\pi}{4}n\right)$
$imag(x) = e^{1.2n} \sin\left(\frac{\pi}{4}n\right)$

Operations on Sequences (.m file 2)

1) Signal addition, multiplication

It is implemented in MATLAB by "+", ".*". However, the lengths of $x_1[n]$ and $x_2[n]$ must be the same. If sequences are of unequal lengths or sample positions are different, we have to first augment $x_1[n]$ and $x_2[n]$ so that they have the same position vector.

Given (x_1, n_1)
- x_1 → sequence
- n_1 → position vector of x_1

(x_2, n_2)
- x_2 → sequence
- n_2 → position vector of x_2

n = position vector of $x_1[n]+x_2[n]$ or $x_1[n]x_2[n]$

```
n=min( min(n1), min(n2)): max(max(n1),max(n2))
x1_new=[ zeros(1, length(min(n):min(n1)-1)) x1 ...
         zeros(1, length(max(n1)+1 : max(n))) ]

x2_new=[ zeros(1, length(min(n):min(n2)-1)) x2 ...
         zeros(1, length(max(n2)+1:max(n))) ]

y1 = x1_new + x2_new
y2 = x1_new .* x2_new
```

2) Shifting $\quad y[n] = x[n-n_0]$

For "n_0" shift, change position vector.

```
n_new = n + n0;
```

3) Folding

Each sample of $x[n]$ is flipped around $n=0$ to obtain a folded sequence $y[n]$

$$y[n] = x[-n]$$

```
y = fliplr(x);
n = -fliplr(n);
```

Ex: $n = 0:3$
$= 0 \; 1 \; 2 \; 3$

$n = -\text{fliplr}(n) = -3:0$
$= -3 \; -2 \; -1 \; 0$

4) Signal Energy

The energy of a sequence $x[n]$ is given by

$$E_x = \sum_{n=-\infty}^{\infty} x[n]x^*[n] = \sum_{n=-\infty}^{\infty} |x[n]|^2$$

```
Ex = sum(x.*conj(x));
Ex = sum(abs(x).^2);
```

→ another approach

LTI Systems

Convolution (.m file 3)

Ex

$$x[n] = u[n] - u[n-10]$$

$$h[n] = (0.9)^n u[n]$$

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$$

$$= \sum_{k=0}^{9} (1)\,(0.9)^{n-k}\, u[n-k]$$

Case 1) $n < 0$ $\quad 0 \leq k \leq 9 \quad \rightarrow \quad n-k < 0 \rightarrow u[n-k] = 0$

$y[n] = 0$

Case 2) $0 \leq n < 9 \quad u[n-k] = 1 \quad 0 \leq k \leq n$

$$y[n] = (0.9)^n \sum_{k=0}^{n} (0.9)^{-k} = (0.9)^n \frac{1 - (0.9)^{-(n+1)}}{1 - (0.9)^{-1}}$$

$$= \frac{(0.9)^n - (0.9)^{-1}}{1 - \frac{10}{9}} = 10\left[1 - (0.9)^{n+1}\right] \quad 0 \leq n <$$

Case 3) $n \geq 9 \quad u[n-k] = 1 \quad 0 \leq k \leq 9$

$$y[n] = (0.9)^n \sum_{k=0}^{9} (0.9)^{-k} = (0.9)^n \frac{1 - (0.9)^{-10}}{1 - (0.9)^{-1}}$$

$$= 10\left[1 - (0.9)^{10}\right] \quad n \geq 9$$

$$\boxed{y = conv(x, h);}$$

But we don't have the information of beginning and end point of $y[n]$.

nonzero over the interval

$x[n]: \; n_{x_1} \leq n \leq n_{x_2}$

nonzero over the interval

$h[n]: \; n_{h_1} \leq n \leq n_{h_2}$

nonzero over the interval

$y[n]: \; n_{y_1} \leq n \leq n_{y_2}$

$$\boxed{\begin{array}{l} n_{y_1} = n_{x_1} + n_{h_1}; \\ n_{y_2} = n_{x_2} + n_{h_2}; \end{array}}$$

Difference Equations

An LTI system can also be described by a linear constant coefficient difference equation of the form

$$\sum_{k=0}^{N} a_k \, y[n-k] = \sum_{m=0}^{M} b_m \, x[n-m]$$

$\boxed{y = filter(b, a, x)}$ ☆ y has the same length as the input;

$b = [b_0 \; b_1 \dots \; b_M]$

$a = [a_0 \; a_1 \dots \; a_N]$

Ex 1: $y[n] - y[n-1] + 0.9y[n-2] = x[n]$

```
b=[1] ;
a=[1 -1 0.9] ;
y=filter(b,a,x);
```

Ex 2: $x[n] = u[n] - u[n-10]$

$h[n] = 0.9^n u[n]$

$y[n] = h[n] * x[n]$

$(0.9)h[n-1] = (0.9)(0.9)^{n-1}u[n-1] = (0.9)^n u[n-1]$

$\underbrace{h[n] - (0.9)h[n-1]} = (0.9)^n u[n] - (0.9)^n u[n-1]$

$= (0.9)^n \delta[n]$

$= \underbrace{\delta[n]}$

$y[n] - 0.9y[n-1] = x[n]$

```
b=[1]
a=[1 -0.9]
y=filter(b,a,x);
```

FIR Filter finite duration impulse response filter

$$y[n] = \sum_{m=0}^{M} b_m \, x[n-m]$$

$$b_m = h[m]$$

* conv(x,h) has a longer length than filter(b, 1, x).

IIR Filter: infinite duration impulse response.

Discrete Time Fourier Transform (.m file 4)

★ $X(e^{j\omega}) \triangleq \sum_{n=-\infty}^{\infty} x[n]\, e^{-j\omega n}$ (DTFT)

★ $x[n] \triangleq \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega})\, e^{j\omega n}\, d\omega$ (I-DTFT)

Matlab Implementation

Consider $-\pi : \pi/100 : \pi$ length of 201

```
w = -pi:pi/100:pi;
Y = zeros(1, length(w));
for rr = 1:length(w)
    for ss = 1:length(y)
        Y(rr) = Y(rr) + y(ss) * exp(-j * w(rr) * (ss-1));
    end
end
```

Ex 1: Consider $h[n] = \begin{cases} 1 & 0 \le n \le 4 \\ 0 & \text{, otherwise} \end{cases}$ moving average system

$$H(e^{j\omega}) = \sum_{n=0}^{4} e^{-j\omega n}$$

$$= \frac{1-e^{-j\omega 5}}{1-e^{-j\omega}} = \frac{e^{j\omega 5/2} - e^{-j\omega 5/2}}{e^{j\omega/2} - e^{-j\omega/2}} \cdot \frac{e^{-j\omega 5/2}}{e^{-j\omega/2}}$$

$$= \frac{\sin(\omega 5/2)}{\sin(\omega/2)}\, e^{-j\omega 2}$$

★ It has a lowpass characteristics.

Note that when $x[n] = \sum_k a_k e^{j\omega_k n}$

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m]$$

$$= \sum_{m=-\infty}^{\infty} \sum_k a_k e^{j\omega_k m} h[n-m]$$

$$= \sum_k a_k \underbrace{\sum_{m=-\infty}^{\infty} h[n-m] e^{-j\omega_k (n-m)}}_{H(e^{j\omega_k})} e^{j\omega_k n}$$

$$y[n] = \sum_k a_k H(e^{j\omega_k}) e^{j\omega_k n}$$

when $x[n] = \sum_k b_k \cos(\omega_k n + \phi_k)$

$$= \sum_k \left(\frac{b_k}{2} e^{j\phi_k} e^{j\omega_k n} + \frac{b_k}{2} e^{-j\phi_k} e^{-j\omega_k n} \right)$$

$$y[n] = \sum_k \left(\frac{b_k}{2} e^{j\phi_k} H(e^{j\omega_k}) e^{j\omega_k n} + \frac{b_k}{2} e^{-j\phi_k} H(e^{-j\omega_k}) e^{-j\omega_k n} \right)$$

If $h[n]$ is real $\Rightarrow H(e^{-j\omega_k}) = H^*(e^{j\omega_k})$

$$y[n] = \sum_k b_k \left| H(e^{j\omega_k}) \right| \cos\left(\omega_k n + \phi_k + \angle H(e^{j\omega_k}) \right)$$

Apply $x[n] = \cos\left(\frac{2\pi}{7} n\right)$

$y[n] = ?$

$Y(e^{j\omega}) = ?$

Ex 2: $x[n] = \cos\left(\frac{2\pi}{7} n\right) + \cos\left(\frac{2\pi}{5} n\right)$

$y[n] = ?$

$Y(e^{j\omega}) = ?$

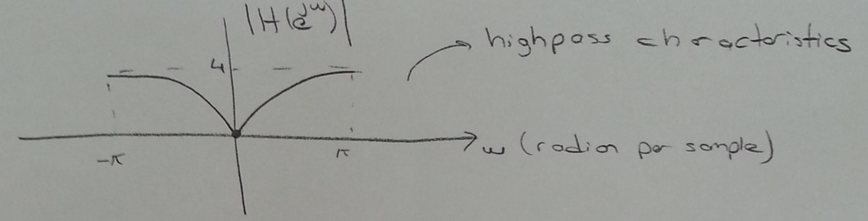
Ex 3

$$h[n] = [\underset{\uparrow\, n=0}{1} \;\; -2 \;\; 1]$$

$$H(e^{j\omega}) = 1 - 2e^{-j\omega} + e^{-j2\omega}$$

$$H(e^{j\omega}) = e^{-j\omega}\left(e^{j\omega} - 2 + e^{-j\omega}\right)$$

$$|H(e^{j\omega})| = 2\,|\cos\omega - 1|$$



Apply $x[n] = \underbrace{5}_{\text{DC part (low frequency)}} + \underbrace{\cos(\pi n)}_{\text{high frequency part}}$

$y[n] = ?$

$Y(e^{j\omega}) = ?$