DTFT

南台科大　趙春棠

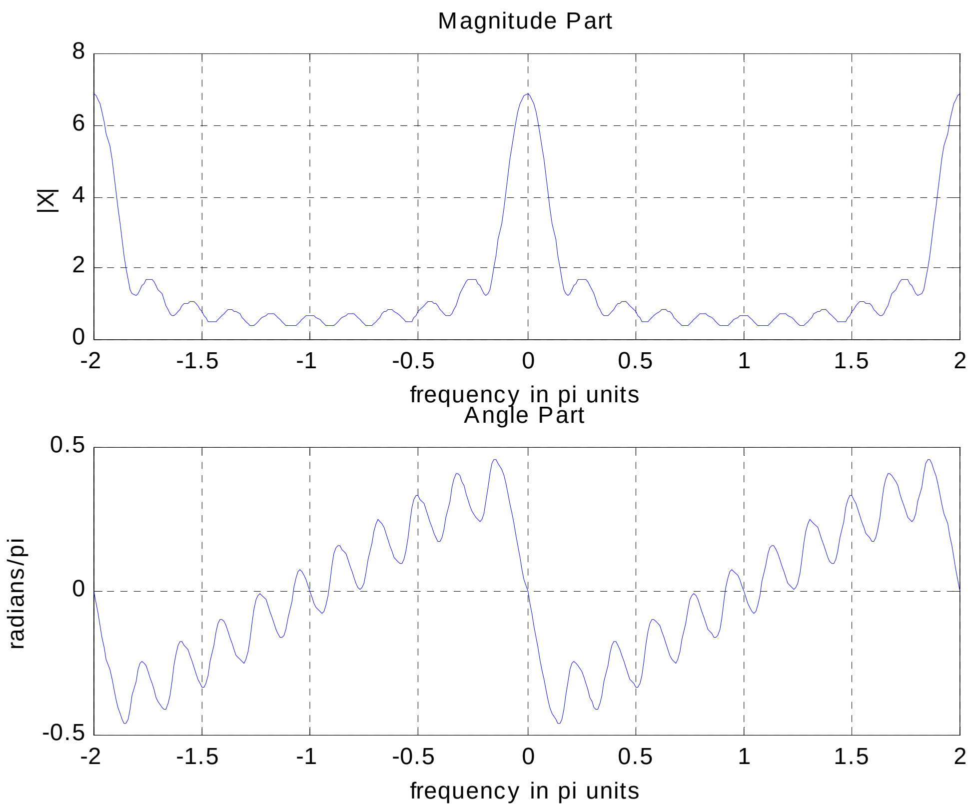
x[n]=[0.9^0　0.9^1　0.9^2　0.9^3　0.9^4 … 0.9^{10}]; 取 x[n] 之 DTFT

做法一：

```
function [X]=plot_dtft_t(x,n)
w=-2*pi: pi/100:2*pi;            % -2*pi ~ 2*pi 間取 401 點
for i=1:length(w)
   W=exp(-j*w(i)*n);
   W=W.';                         %註
   X(i)=x*W;
end
magX = abs(X); angX = angle(X);
subplot(2,1,1); plot(w/pi,magX);grid
xlabel('frequency in pi units'); ylabel('|X|'); title('Magnitude Part')
subplot(2,1,2); plot(w/pi,angX/pi);grid
xlabel('frequency in pi units'); ylabel('radians/pi');title('Angle Part')
```

註：此時必須是 **Unconjugated Transpose** 才對，否則結果有誤

Matlab: n=0:10;　x=(0.9).^n;　[X]=plot_dtft_t(x,n);



註：上圖 **w** 是取**[-2*pi ~ 2*pi]**，然而由 **DTFT** 的週期性及對稱性可知，**w** 取 **[0 ~ pi]** 即可

做法二：

```
function [X]=plot_dtft(x,n)
w=-2*pi: pi/100:2*pi;                 % -2*pi ~ 2*pi  間取 401 點
k= -200:200                           % k= (100/pi)*w
X=x*(exp(-j*pi/100)).^(n'*k);
magX = abs(X); angX = angle(X);
subplot(2,1,1); plot(w/pi,magX);grid
xlabel('frequency in pi units'); ylabel('|X|'); title('Magnitude Part')
subplot(2,1,2); plot(w/pi,angX/pi);grid
xlabel('frequency in pi units'); ylabel('radians/pi');title('Angle Part')
```

Matlab: n=0:10;   x=(0.9).^n;   [X]=plot_dtft(x,n);

圖形： 同上

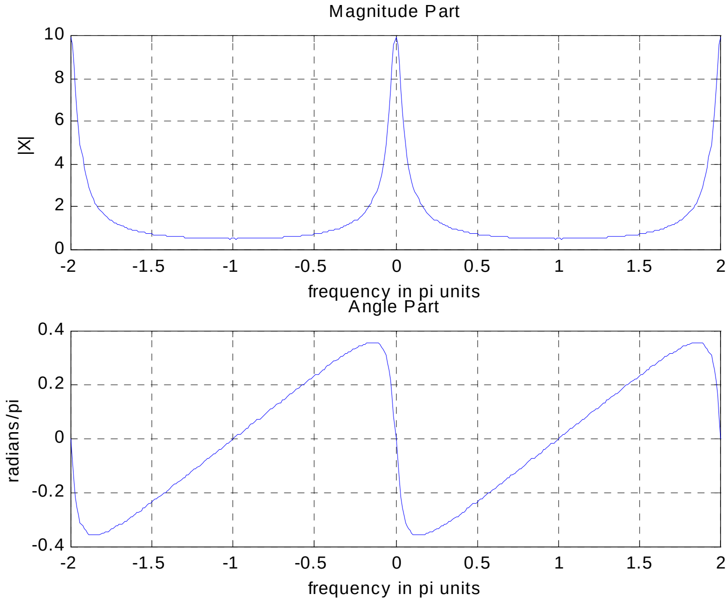
System Frequency Response

System Impulse Response: h[n]=[0.9^0 0.9^1 0.9^2 0.9^3 0.9^4 …];

做法一：由脈衝響應得系統頻率響應，取 h[n]= [0.9^0 0.9^1 0.9^2 0.9^3 0.9^4 …0.9^{50}];

之 DTFT 即為 $H(e^{jw})$

Matlab: n=0:50; x=(0.9).^n; [X]=plot_dtft(x,n);



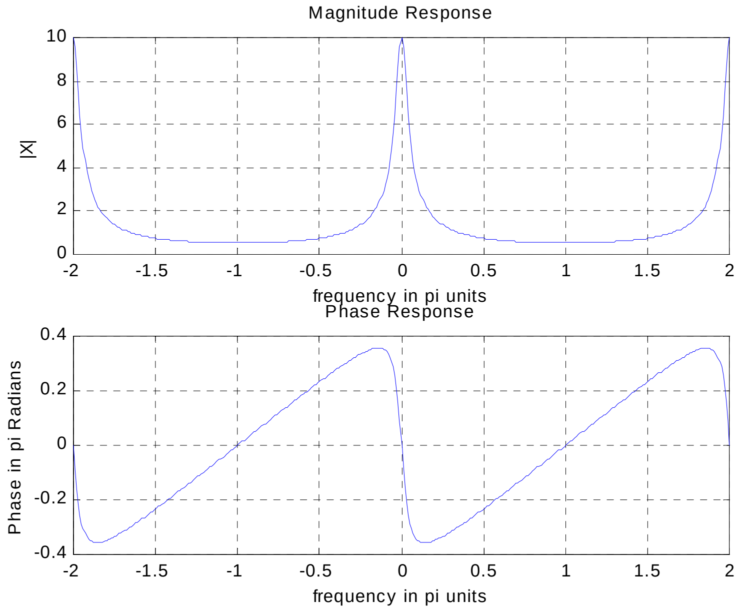
評論：以上只是近似的做法耳！

做法二：由系統差分方程式得頻率響應， **y(n)-0.9y(n-1)=x(n)**

```
function [H]=plot_freqresp(b,a)
w=-2*pi: pi/100:2*pi;          % -2*pi ~ 2*pi 間取 401 點
k= -200:200;
m=0:length(b)-1; l=0:length(a)-1;
num=b*exp(-j*(pi/100)*m'*k);
den=a*exp(-j*(pi/100)*l'*k);
H=num./den;
%for i=1:length(w)
%   W=exp(-j*w(i)*m);
%   W=W.';
%   num(i)=b*W;
%   W=exp(-j*w(i)*l);
%   W=W.';
%   den(i)=a*W;
%   H(i)=num(i)/den(i);
%end
magH = abs(H); angH = angle(H);
subplot(2,1,1); plot(w/pi,magH);grid
```

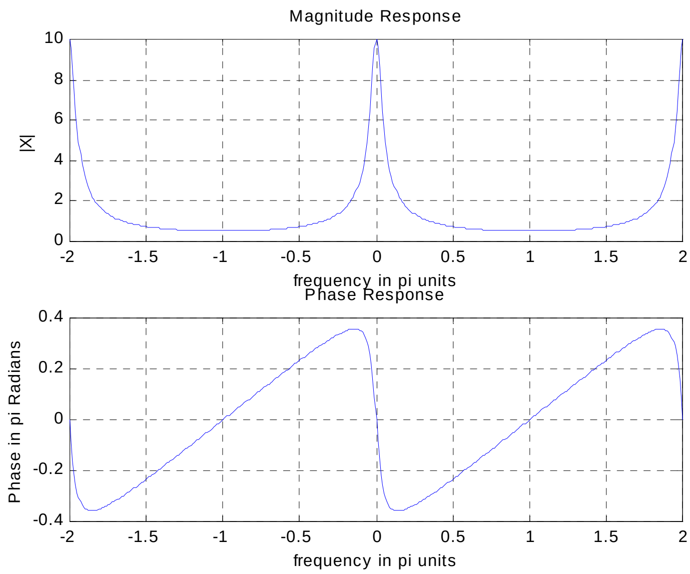
```
xlabel('frequency in pi units'); ylabel('|X|');title('Magnitude Response')
subplot(2,1,2); plot(w/pi,angH/pi);grid
xlabel('frequency in pi units'); ylabel('Phase in pi Radians');title('Phase Response');
```

Matlab: b=[1]; a=[1 -0.9]; H=plot_freqresp(b,a);



做法三：直接由計算得的 **H(e^{jw})**，畫圖之

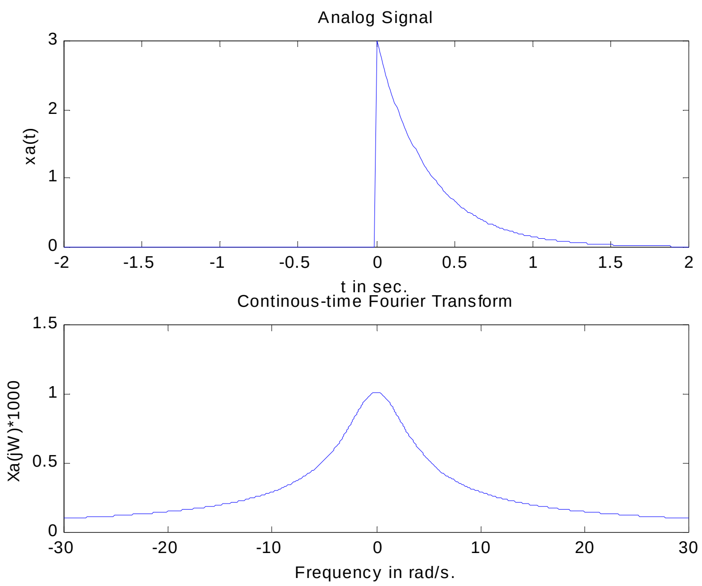
```
w = -2*pi:pi/100:2*pi;
 H = exp(j*w) ./ (exp(j*w) - 0.9*ones(1,length(w)));
 magH = abs(H); angH = angle(H);
subplot(2,1,1); plot(w/pi,magH);grid
xlabel('frequency in pi units'); ylabel('|X|');title('Magnitude Response')
subplot(2,1,2); plot(w/pi,angH/pi);grid
xlabel('frequency in pi units'); ylabel('Phase in pi Radians');title('Phase Response');
```



Sampling

∗ Continuous Signal: Xa(t)= 3e^{-3t} u(t)

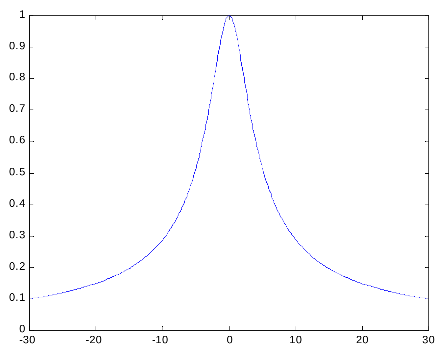
```
% sample_exp.m
   Dt = 0.01; t = -2:Dt:2; xa = 3*exp(-3*abs(t));
   t1= -2:Dt: -Dt;
   xa(1:length(t1))=zeros(1,length(t1));
% Continous-time Fourier Transform
   Wmax = 30; K = 500; k = -K:1:K; W=k*Wmax/K;
   Xa = xa* exp(-j*t'*W) * Dt; Xa = abs(Xa);
%    W = [-fliplr(W), W(2:501)];    % Omega from -Wmax to Wmax
%    Xa =[fliplr(Xa), Xa(2:501)]; % Xa over -Wmax to Wmax interval
   subplot(2,1,1); plot(t,xa);
   xlabel('t in sec.'); ylabel('xa(t)')
   title('Analog Signal')
   subplot(2,1,2); plot(W,Xa);
   xlabel('Frequency in rad/s.'); ylabel('Xa(jW)*1000')
   title('Continous-time Fourier Transform')
```



註：以上 continuous-time Xa(jw) 可直接利用 **freqs.m** 計算

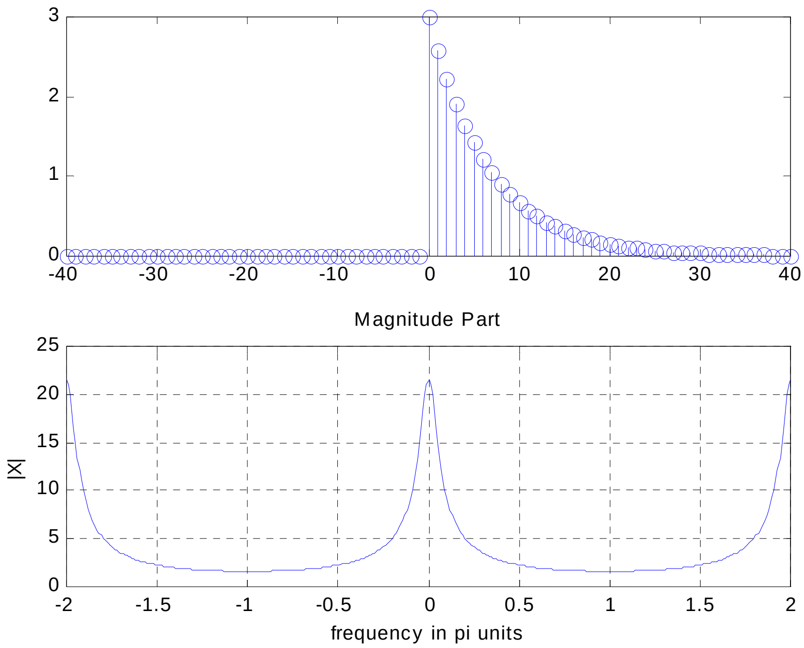
Matlab: Wmax=30; K=500; k=-K:1:K; W=k*Wmax/K; B=[0 3]; A=[1 3];

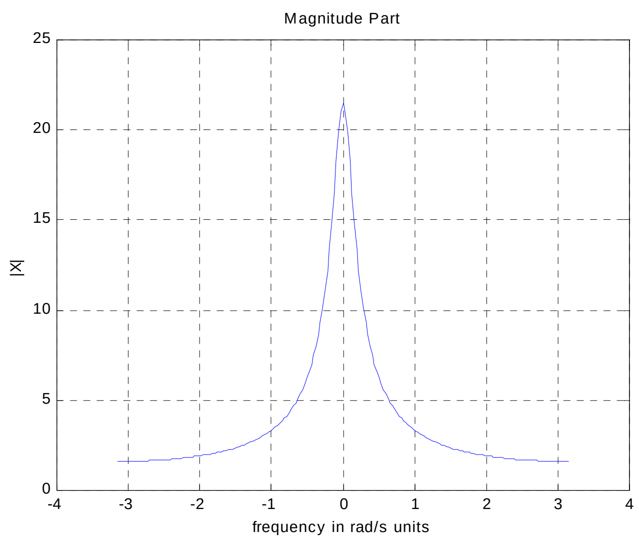
H = FREQS(B,A,W); H=abs(H); plot(W,H)



∗ DTFT of sampling signal (Sampling Time = 0.05)

```
%sample_exp1.m
Ts= 0.05; t = -2: Ts:2; n=20*t;
x = 3*exp(-3*t); u=stepseq(0,-40,40);
x=sigmult(x,n,u,n);
subplot(2,1,1); stem(n,x);
%以下爲描繪 DTFT “增益” 的部分，由 function   plot_dtft 中抄錄
w=-2*pi: pi/100:2*pi;              % -2*pi ~ 2*pi 間取 401 點
k= -200:200;                              % k= (100/pi)*w
X=x*exp(-j*n'*w);
magX = abs(X); angX = angle(X);
subplot(2,1,2); plot(w/pi,magX);grid;
xlabel('frequency in pi units'); ylabel('|X|'); title('Magnitude Part')
```

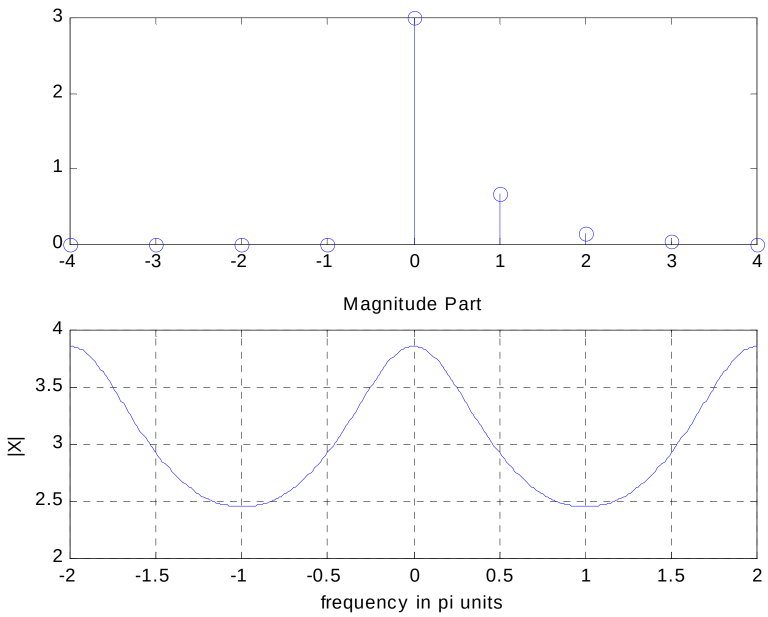




＊ **DTFT of sampling signal (Sampling Time = 0.5)**

```
%sample_exp2.m
Ts= 0.5; t = -2: Ts:2; n=2*t;
x = 3*exp(-3*t); u=stepseq(0,-4,4);
x=sigmult(x,n,u,n);
subplot(2,1,1); stem(n,x);
```

以下描繪 DTFT 的程式部分與上面同，故省略



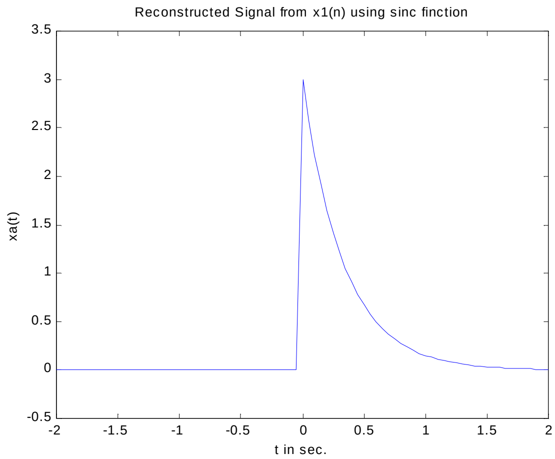
Reconstruction

以上例說明之

＊ 以 **sinc(x)** 重建

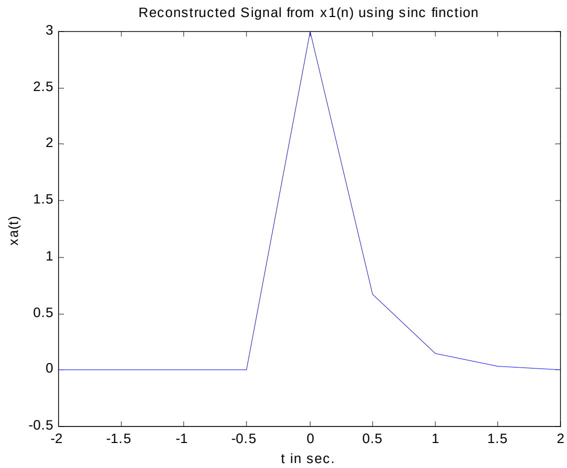
Case 1: Sampling Time = 0.05

```
Ts= 0.05; t = -2: Ts:2; n=20*t; nTs = n*Ts; Fs=1/Ts;
x = 3*exp(-3*t); u=stepseq(0,-40,40);
x=sigmult(x,n,u,n);
xa = x * sinc(Fs*(ones(length(n),1)*t - nTs'*ones(1,length(t))));
plot(t,xa);
xlabel('t in sec.'); ylabel('xa(t)')
title('Reconstructed Signal from x1(n) using sinc finction');
```



Case 2: Sampling Time = 0.5

```
Ts= 0.5; t = -2: Ts:2; n=2*t;
x = 3*exp(-3*t); u=stepseq(0,-4,4);
x=sigmult(x,n,u,n);
nTs = n*Ts; Fs=1/Ts;
xa = x * sinc(Fs*(ones(length(n),1)*t - nTs'*ones(1,length(t))));
plot(t,xa);    xlabel('t in sec.'); ylabel('xa(t)')
title('Reconstructed Signal from x1(n) using sinc finction');
```



＊ 以 **ZOH (stairs 指令)** 重建

Ts= 0.05; t = -2: Ts:2; n=20*t; nTs = n*Ts; Fs=1/Ts;

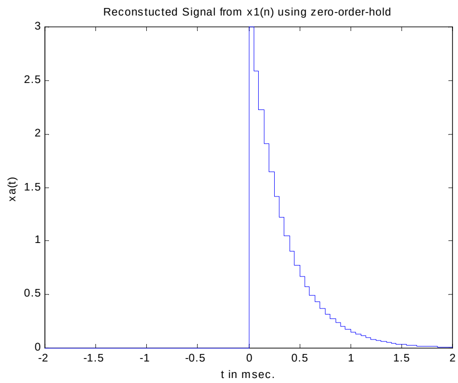
x = 3*exp(-3*t); u=stepseq(0,-40,40);

x=sigmult(x,n,u,n);

stairs(t,x);

xlabel('t in sec.'); ylabel('xa(t)')

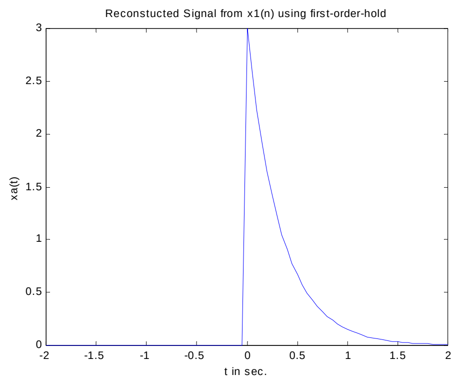
title('Reconstucted Signal from x1(n) using zero-order-hold');



＊ 以 **FOH** 內插法 **(plot 指令)** 重建

上述中 stairs(t,x); 指令改爲 plot(t,x); 另 title 指令更改如下，其餘皆同

title('Reconstucted Signal from x1(n) using first-order-hold');



＊ 以 弧線內插法 **(spline 指令)** 重建

Ts= 0.05; nTs = -2: Ts:2; n=20*nTs; Fs=1/Ts;

x = 3*exp(-3*t); u=stepseq(0,-40,40); x=sigmult(x,n,u,n);

Dt=0.00005; t= -2: Dt: 2;

spline(nTs,x,t);   xlabel('t in sec.'); ylabel('xa(t)')

title('Reconstucted Signal from x1(n) using cubic spline function');