8 IIR Filter Design　Butterworth Filter Design

南台科大　趙春棠

Step1 ：設計一Ωc=0.5 的三階 Butterworth 類比原型 Filter

＊ **[z,p,k]=buttap(N)** %Matlab 提供之函數用以設計Ω**c=1** 的 **N** 階 Butterworth 類比（**A**nalog）原型（**P**rototype）Filter

＊ **function [b,a] = u_buttap(N,Omegac); % 根據 buttap(N)，進一步使Ωc 不限為 1**

```
% Unnormalized Butterworth Analog Lowpass Filter Prototype
% [b,a] = u_buttap(N,Omegac);
%        b = numerator polynomial coefficients of Ha(s)
%        a = denominator polynomial coefficients of Ha(s)
%        N = Order of the Butterworth Filter
% Omegac = Cutoff frequency in radians/sec
 [z,p,k] = buttap(N);
p = p*Omegac;
k = k*Omegac^N;
B = real(poly(z));
b0 = k;
b = k*B;
a = real(poly(p));
```

＊ **function [C,B,A] = sdir2cas(b,a); %將 H(s)直接式轉換成串接形式**

```
% DIRECT-form to CASCADE-form conversion in s-plane
% [C,B,A] = sdir2cas(b,a);
%   C = gain coefficient
%   B = K by 3 matrix of real coefficients containing bk's
%   A = K by 3 matrix of real coefficients containing ak's
%   b = numerator polynomial coefficients of DIRECT form
%   a = denominator polynomial coefficients of DIRECT form
Na = length(a)-1; Nb = length(b)-1;
% compute gain coefficient C
b0 = b(1); b = b/b0;
a0 = a(1); a = a/a0;
C = b0/a0;
%
% Denominator second-order sections:
p = cplxpair(roots(a)); K = floor(Na/2);
if K*2 == Na     % Computation when Na is even
    A = zeros(K,3);
    for n=1:2:Na
        Arow = p(n:1:n+1,:);
        Arow = poly(Arow);
        A(fix((n+1)/2),:) = real(Arow);
    end
elseif Na == 1     % Computation when Na = 1
    A = [0 real(poly(p))];
else                    % Computation when Na is odd and > 1
    A=zeros(K+1,3);
```

```
    for n=1:2:2*K
        Arow = p(n:1:n+1,:);
        Arow = poly(Arow);
        A(fix((n+1)/2),:) = real(Arow);
    end
    A(K+1,:) = [0 real(poly(p(Na)))];
end

% Numerator second-order section:
z = cplxpair(roots(b)); K = floor(Nb/2);
if Nb == 0      % Computation when Nb = 0
    B = [0 0 poly(z)];
elseif K*2 == Nb      % Computation when Nb is even
    B=zeros(K,3);
    for n=1:2:Nb
        Brow = z(n:1:n+1,:);
        Brow = poly(Brow);
        B(fix((n+1)/2),:) = real(Brow);
    end
elseif Nb == 1      % Computation when Nb = 1
    B = [0 real(poly(z))];
else                              % Computation when Nb is odd and > 1
    B = zeros(K+1,3);
    for n=1:2:2*K
        Brow = z(n:1:n+1,:);
        Brow = poly(Brow);
        B(fix((n+1)/2),:) = real(Brow);
    end
    B(K+1,:) = [0 real(poly(z(Nb)))];
end
```

＊ **Matlab Program:**

```
N=3; OmegaC=0.5; [b,a]=u_buttap(N,OmegaC); [C,B,A]=sdir2cas(b,a);
結果：a = 1.0000        1.0000        0.5000        0.1250
      b = 0.1250
      A = 1.0000        0.5000        0.2500
                0        1.0000        0.5000
      B =    0          0          1
      C =    0.1250
```

Step1 ：設計一滿足以下規格的 **Butterworth** 類比原型 **Filter**

通帶截止：Ωp=0.2π; 通帶漣波：Rp=7dB

阻帶截止：Ωs=0.3π; 阻帶漣波：As=16dB

註： 其實經初步計算結果，設計一Ω**c** 約 **0.5** 的三階 **Butterworth** （同上例），即可完成

＊ **function [b,a] = afd_butt(Wp,Ws,Rp,As); % 比 u_buttap 函數更廣義的設計函數**

```
% Analog Lowpass Filter Design: Butterworth
% [b,a] = afd_butt(Wp,Ws,Rp,As);
%    b = Numerator coefficients of Ha(s)
%    a = Denominator coefficients of Ha(s)
% Wp = Passband edge frequncy in rad/sec; Wp > 0
% Ws = Stopband edge frequncy in rad/sec; Ws > Wp > 0
% Rp = Passband ripple in +dB; (Rp > 0)
% As = Stopband attenuation in +dB; (As > 0)
if Wp <= 0
     error('Passband edge must be larger than 0')
end
if Ws <= Wp
     error('Stopband edge must be larger than Passband edge')
end
if (Rp <= 0) | (As < 0)
     error('PB ripple and/or SB attenuation ust be larger than 0')
end

N = ceil((log10((10^(Rp/10)-1)/(10^(As/10)-1)))/(2*log10(Wp/Ws)));
fprintf('\n*** Butterworth Filter Order = %2.0f \n',N)
OmegaC = Wp/((10^(Rp/10)-1)^(1/(2*N)));
[b,a]=u_buttap(N,OmegaC);
```

＊ **function [db,mag,pha,w] = freqs_m(b,a,wmax); % 可計算類比 H(s)之 H(jw)響應**

```
% Computation of s-domain frequnecy response: Modified version
% ----------------------------------------------------------
% [db,mag,pha,w] = freqs_m(b,a,wmax);
%      db = Relative magnitude in db over [0 to wmax]
%    nag = Absolute magnitude over [0 to wmax]
%    pha = Phase response in radinans over [0 to wmax]
%       w = array of 500 frequency samples between [0 to wmax]
%       b = Numerator polynomial coefficients of Ha(s)
%       a = Denominator polynomial coefficients of Ha(s)
% wmax = Maximum frequncy in rad/sec over which response is desired
w = [0:1:500]*wmax/500;
H = freqs(b,a,w);
mag = abs(H);
db = 20*log10((mag+eps)/max(mag));
pha = angle(H);
```

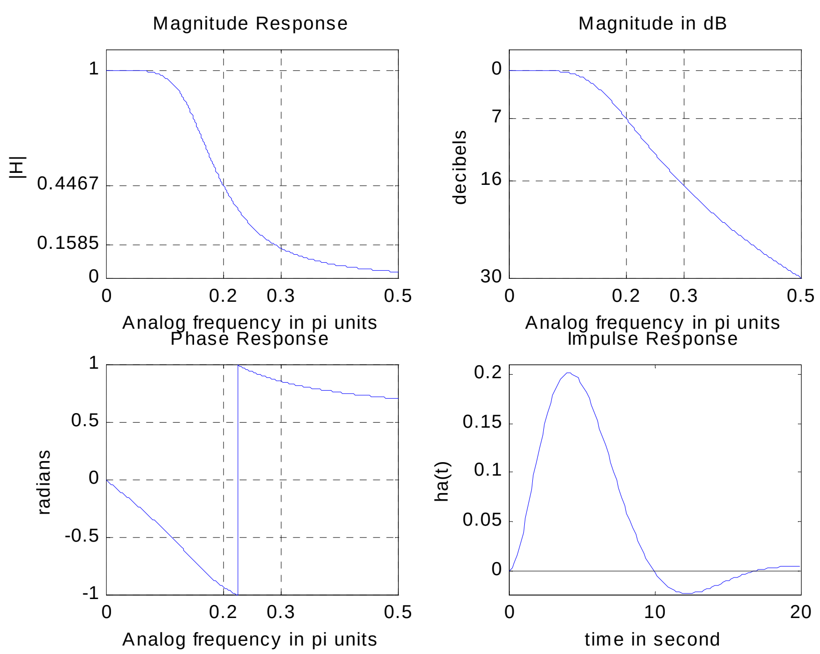
＊ **Matlab** 程式：

```
Wp=0.2*pi; Ws=0.3*pi; Rp=7; As=16;
Ripple=10^(-Rp/20); Attn=10^(-As/20);
% Analog filter design:
[b,a]=afd_butt(Wp,Ws,Rp,As);
% Calculation of second-order sections:
```

```
[C,B,A]=sdir2cas(b,a)
% Calculation of Frequncy Response:
[db,mag,pha,w]=freqs_m(b,a,0.5*pi);
% Calculation of Impulse Response:
[ha,x,t]=impulse(b,a);
% Plots
subplot(2,2,1);
plot(w/pi,mag);xlabel('Analog frequency in pi units');
ylabel('|H|'); title('Magnitude Response');axis([0,0.5,0,1.1]);
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,0.5]);
set(gca,'YTickMode','manual','YTick',[0,0.1585,0.4467,1]);grid
subplot(2,2,2);
plot(w/pi,db); xlabel('Analog frequency in pi units');
ylabel('decibels'); title('Magnitude in dB');axis([0,0.5,-30,3]);
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,0.5]);
set(gca,'YTickMode','manual','YTick',[-30,-16,-7,0]);grid
set(gca,'YTickLabelMode','manual','YTickLabels',['30';'16';' 7';' 0'])
subplot(2,2,3);
plot(w/pi,pha/pi); xlabel('Analog frequency in pi units');
ylabel('radians'); title('Phase Response');axis([0,0.5,-1,1]);
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,0.5]);
set(gca,'YTickMode','manual','YTick',[-1:0.5:1]);grid
subplot(2,2,4);
xa=0.*t;
plot(t,ha,'b',t,xa,'k'); xlabel('time in second'); ylabel('ha(t)');
title('Impulse Response');axis([0,20,-0.025,0.21]);
set(gca,'XTickMode','manual','XTick',[0:10:20]);
set(gca,'YTickMode','manual','YTick',[0:0.05:0.2]);
```

執行：*** Butterworth Filter Order =   3

```
C =      0.1238
B =        0         0         1
A =      1.0000      0.4985      0.2485
              0      1.0000      0.4985
```



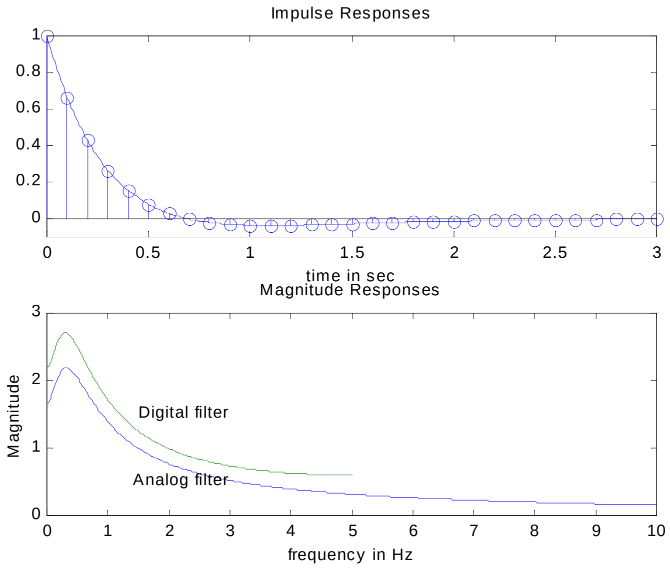
Impulse Invariance Transformation H(s) -> H(z)

```
function [b,a] = imp_invr(c,d,T)
% Impulse Invariance Transformation from Analog to Digital Filter
% ------------------------------------------------------------
% [b,a] = imp_invr(c,d,T)
%    b = Numerator polynomial in z^(-1) of the digital filter
%    a = Denominator polynomial in z^(-1) of the digital filter
%    c = Numerator polynomial in s of the analog filter
%    d = Denominator polynomial in s of the analog filter
%    T = Sampling (transformation) parameter
%
[R,p,k] = residue(c,d);
p = exp(p*T);
[b,a] = residuez(R,p,k);
b = real(b'); a = real(a');
```

例： $Ha(s) = \frac{s+1}{s^2+5s+6}$ T=0.1

Matlab:

```
subplot(1,1,1);
c = [1,1]; d = [1,5,6]; T = 0.1; Fs = 1/T;
[b,a] = imp_invr(c,d,T)
%%b =    1.0000      -0.8966
%%a =    1.0000      -1.5595        0.6065
% Impulse response of the analog filter
t = [0:0.01:3]; [ha,ta] = impulse(c,d,t);
subplot(2,1,1); plot(t,ha,t,0.*t,'k'); axis([0,3,-0.1,1]);hold on
% Impulse response of the digital filter
n = [0:1:3/T]; hn = filter(b,a,impseq(0,0,3/T));
stem(n*T,hn); xlabel('time in sec'); title ('Impulse Responses');
hold off
% Magnitude Response of the digital filter
[db,magd,pha,grd,wd] = freqz_m(b,a);
% magnitude response of the analog filter
[db,mags,pha,ws] = freqs_m(c,d,2*pi*Fs);
subplot(2,1,2); plot(ws/(2*pi),mags*Fs,wd/(2*pi)*Fs,magd)
xlabel('frequency in Hz'); title('Magnitude Responses');
ylabel('Magnitude');
text(1.4,.5,'Analog filter'); text(1.5,1.5,'Digital filter')
```



例： $Ha(s) = \dfrac{3}{s+3}$ T=0.1

Matlab:

c = [3]; d = [1 3]; T = 0.1; Fs = 1/T;

[b,a] = imp_invr(c,d,T)

結果：

b = 3

a = 1.0000

-0.7408

Step2 : 使用 s -> z 轉換

例： 使用 Butterworth 原型設計滿足以下規格之 LP Digital Filter

wp=0.2π; Rp=1dB

ws=0.3π; As=15dB

註： 以上給的是 H(z)頻譜的規格，需先轉回 H(s)上的規格，再利用 Step1 中的做法，設計出 Butterworth Analog Filter H(s)，之後再將 H(s)轉成 H(z)

＊ **Matlab Program:**

```
% Digital Filter Specifications:
wp = 0.2*pi;                          % digital Passband freq in Hz
ws = 0.3*pi;                          % digital Stopband freq in Hz
Rp = 1;                               % Passband ripple in dB
As = 15;                              % Stopband attenuation in dB

% Analog Prototype Specifications: Inverse mapping for frequencies
T = 1;                                % Set T=1
OmegaP = wp * T;                      % Prototype Passband freq
OmegaS = ws * T;                      % Prototype Stopband freq
ep = sqrt(10^(Rp/10)-1);              % Passband Ripple parameter
Ripple = sqrt(1/(1+ep*ep));           % Passband Ripple
Attn = 1/(10^(As/20));                % Stopband Attenuation

% Analog Butterworth Prototype Filter Calculation:
[cs,ds] = afd_butt(OmegaP,OmegaS,Rp,As);

% Impulse Invariance transformation:
[b,a] = imp_invr(cs,ds,T);
[C,B,A] = dir2par(b,a)

% Plotting
figure(1); subplot(1,1,1)
[db,mag,pha,grd,w] = freqz_m(b,a);
subplot(2,2,1); plot(w/pi,mag); title('Magnitude Response')
xlabel('frequency in pi units'); ylabel('|H|'); axis([0,1,0,1.1])
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,1]);
set(gca,'YTickmode','manual','YTick',[0,Attn,Ripple,1]); grid
subplot(2,2,3); plot(w/pi,db); title('Magnitude in dB');
xlabel('frequency in pi units'); ylabel('decibels'); axis([0,1,-40,5]);
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,1]);
set(gca,'YTickmode','manual','YTick',[-15,0]); grid
subplot(2,2,2); plot(w/pi,pha/pi); title('Phase Response')
xlabel('frequency in pi units'); ylabel('pi units'); axis([0,1,-1,1]);
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,1]);
```

```
set(gca,'YTickmode','manual','YTick',[-1,0,1]); grid
subplot(2,2,4); plot(w/pi,grd); title('Group Delay')
xlabel('frequency in pi units'); ylabel('Samples'); axis([0,1,0,10])
set(gca,'XTickMode','manual','XTick',[0,0.2,0.3,1]);
set(gca,'YTickmode','manual','YTick',[0:2:10]); grid
```

執行結果：

```
*** Butterworth Filter Order =    6
C =        []
B =    1.8557     -0.6304
     -2.1428      1.1454
       0.2871    -0.4466
A =    1.0000     -0.9973      0.2570
       1.0000    -1.0691      0.3699
       1.0000    -1.2972      0.6949
```