

ボード線図を描くプログラム `bode2.sci`, `gainplot2.sci` (横軸が rad/s)

佐伯正美 2014年5月8日

2018年4月2日

ボード線図のゲイン線図と位相線図を描くプログラムとして、`scilab` の `CACSD` にある関数 `bode` は横軸が Hz である。教科書では横軸が rad/s であるので、横軸を rad/s としたボード線図を描くプログラムとして既に `bode2.sci` を作成していたが、それは汎用性がないことが判明した。`bode2.sci` を改訂したので、それを以下に示す。ただし、位相は一意でないので、位相線図は $2n\pi$ の任意性があることを念頭においてほしい。このため、`bode2` で描かれたボード線図が `MATLAB` の `control toolbox` にある `bode` による図と異なる場合もあるが誤りではない。

さらに、ゲイン線図だけを描くプログラムとして、`bode2.sci` からゲイン線図を描く部分を切り出して、`gainplot2.sci` を作成した。

(a) テスト用の main プログラム : `test2.sce`

```
// test2.sce
clear
exec('bode2.sci',-1); // 関数を使うのに必要
exec('gainplot2.sci',-1); // 関数を使うのに必要
//
s=poly(0,'s');
sysG=syslin('c',(s+5)/((s+3)*(s*s+0.01*s+1)));
N=300;
w=logspace(-2,2,N);
//
scf(1);
bode2(sysG,w)
scf(2);
gainplot2(sysG,w)
```

(b) ボード線図を描く関数 : bode2.sci

```
function bode2(G, w)  
// bode2.sci by M. Saeki, May 8, 2014  
// ボード線図を横軸を rad/s で描く関数  
// 注意 : Scilab の用意している'bode'では横軸が Hz である.  
//準備  
denG=G.den;//分母多項式  
numG=G.num;//分子多項式  
n=degree(denG);//分母多項式の次数  
m=degree(numG);//分子多項式の次数  
denG_vec=coeff(denG);//分母多項式の係数  
numG_vec=coeff(numG);//分子多項式の係数  
  
//1.ゲイン[dB]の計算  
//準備 zpK=極ゼロゲイン形式の分解  
// G=K((s-z1)(s-z2)s)/((s-p1)(s-p2)(s-p3))  
K=numG_vec(m+1)/denG_vec(n+1);//ゲイン K  
p_vec=roots(denG);//極  
z_vec=roots(numG);//ゼロ点  
for i=1:length(w)  
    s=%i*w(i);  
    // 分母多項式のゲインの和[dB]  
    den_gain=0;  
    if n<>0 then  
        for kp=1:n  
            p=p_vec(kp);  
            den_gain=den_gain+20*log10(abs(s-p));  
        end  
    end  
    // 分子多項式のゲインの和[dB]  
    num_gain=0;  
    if m<>0 then  
        for kz=1:m  
            z=z_vec(kz);  
            num_gain=num_gain+20*log10(abs(s-z));  
        end  
    end
```

```

end
//伝達関数のゲイン
gain(i)=num_gain-den_gain+20*log10(abs(K));
end

//2.位相の計算
//準備 G=b0((1-s/z1)(1-s/z2)s)/((1-s/p1)(1-s/p2)(1-s/p3))
//のための b0=K(-z1)(-z2)/((-p1)(-p2)(-p3))の計算
b0=K;
if n<>0 then
    for kp=1:n
        p=p_vec(kp);
        if p<>0 then
            b0=b0*(-p);
        end
    end
end
if m<>0 then
    for kz=1:m
        z=z_vec(kz);
        if z<>0 then
            b0=b0*(-z);
        end
    end
end
end
for i=1:length(w)
    s=%i*w(i);
    // 分母多項式の位相の和
    den_phase=0;
    if n<>0 then
        for kp=1:n
            p=p_vec(kp);
            if p==0 then
                den_phase=den_phase+%pi/2;
            else
                v=1-s/p;
            end
        end
    end
end

```

```

        den_phase=den_phase+atan(imag(v),real(v));
    end
end
end
//分子多項式の位相の和
num_phase=0;
if m<>0 then
    for kz=1:m
        z=z_vec(kz);
        if z==0 then
            num_phase=num_phase+%pi/2;
        else
            v=1-s/z;
            num_phase=num_phase+atan(imag(v),real(v));
        end
    end
end
end
//伝達関数の位相
phase(i)=num_phase-den_phase;
if real(b0)<0 then
    phase(i)=num_phase-den_phase-%pi;
    //b0 が負のとき-%pi とする
    // +%pi もあり得るが用途に依存する
end
end

//3.ボード線図の描画
subplot(2,1,1);
plot2d(w,gain,logflag='ln');
xgrid(4)
xtitle('Bode diagram','omega[rad/s]','gain [dB]')
subplot(2,1,2);
plot2d(w,phase*180/%pi,logflag='ln');
xgrid(4)
xtitle('','omega[rad/s]','phase [deg]')
endfunction

```

(c) ゲイン線図を描く関数 : gainplot2.sci

```
function gainplot2(G, w)
```

```
// gainplot2.sci by M. Saeki, May 8, 2014
```

```
// ゲイン線図を横軸を rad/s で描く関数 (自作)
```

```
// 注意 : Scilab の用意している 'gainplot' では横軸が Hz である.
```

```
//準備
```

```
denG=G.den;//分母多項式
```

```
numG=G.num;//分子多項式
```

```
n=degree(denG);//分母多項式の次数
```

```
m=degree(numG);//分子多項式の次数
```

```
denG_vec=coeff(denG);//分母多項式の係数
```

```
numG_vec=coeff(numG);//分子多項式の係数
```

```
//ゲイン[dB]の計算
```

```
//準備 zpK=極ゼロゲイン形式の分解
```

```
// G=K((s-z1)(s-z2)s)/((s-p1)(s-p2)(s-p3))
```

```
K=numG_vec(m+1)/denG_vec(n+1);//ゲイン K
```

```
p_vec=roots(denG);//極
```

```
z_vec=roots(numG);//ゼロ点
```

```
for i=1:length(w)
```

```
    s=%i*w(i);
```

```
    // 分母多項式のゲインの和[dB]
```

```
    den_gain=0;
```

```
    if n<>0 then
```

```
        for kp=1:n
```

```
            p=p_vec(kp);
```

```
            den_gain=den_gain+20*log10(abs(s-p));
```

```
        end
```

```
    end
```

```
    // 分子多項式のゲインの和[dB]
```

```
    num_gain=0;
```

```
    if m<>0 then
```

```
        for kz=1:m
```

```
            z=z_vec(kz);
```

```
            num_gain=num_gain+20*log10(abs(s-z));
```

```
        end
```

```
end
//伝達関数のゲイン
gain(i)=num_gain-den_gain+20*log10(abs(K));
end
//ゲイン線図の描画
plot2d(w,gain,logflag='ln');
xgrid(4)
xtitle('gain plot','omega[rad/s]','gain [dB]')
endfunction
```