1.

% 定义分子和分母多项式系数

num = conv([1],[1]);

den = conv([3 1],[5 1]);

% 使用 tf 函数创建传递函数

sys = tf(num,den);

% 绘制Bode图

figure;

bode(sys);

title('开环传递函数的伯德图');

% 绘制Nyquist曲线

figure;

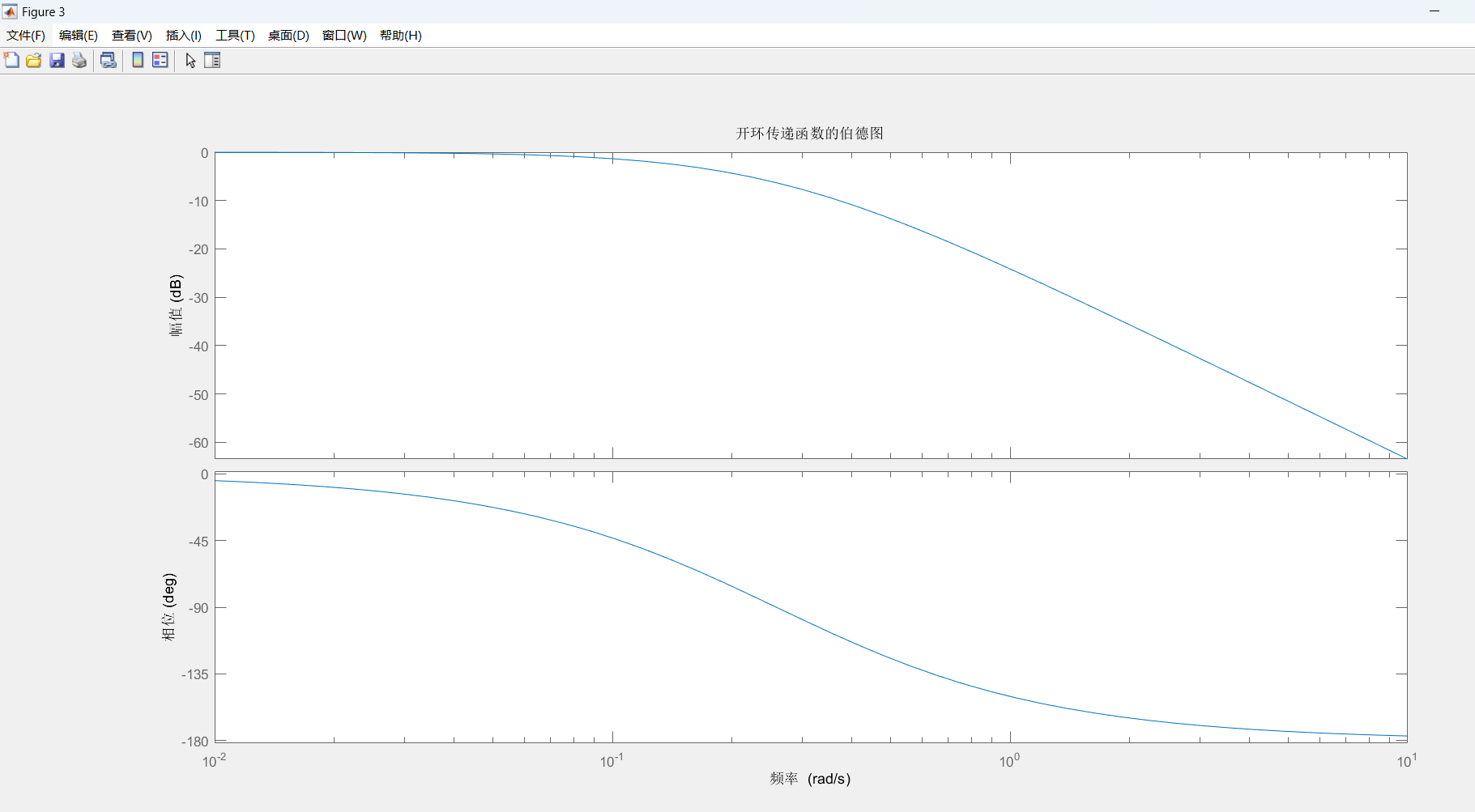
nyquist(sys);

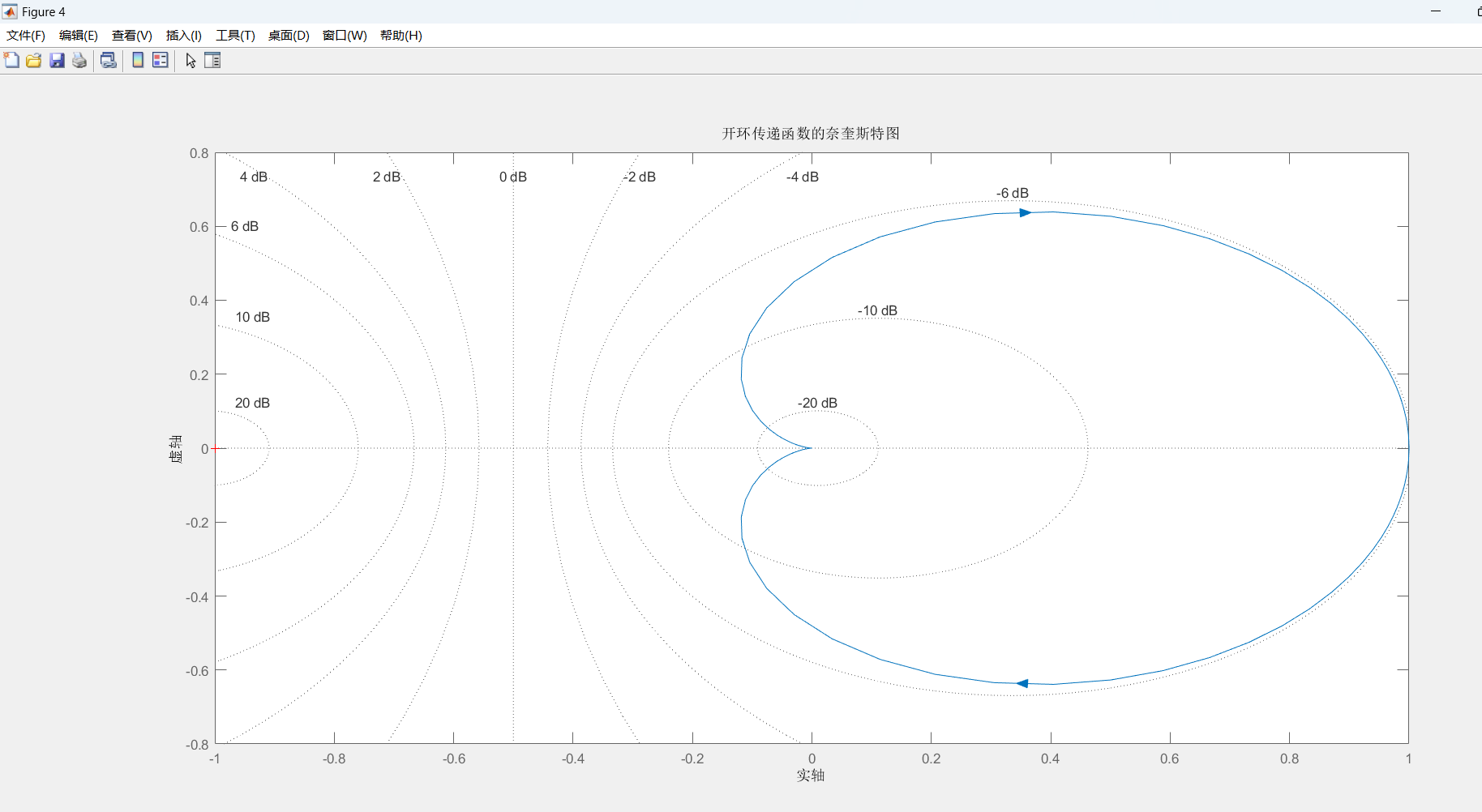
title('开环传递函数的奈奎斯特图');

% 显示图像

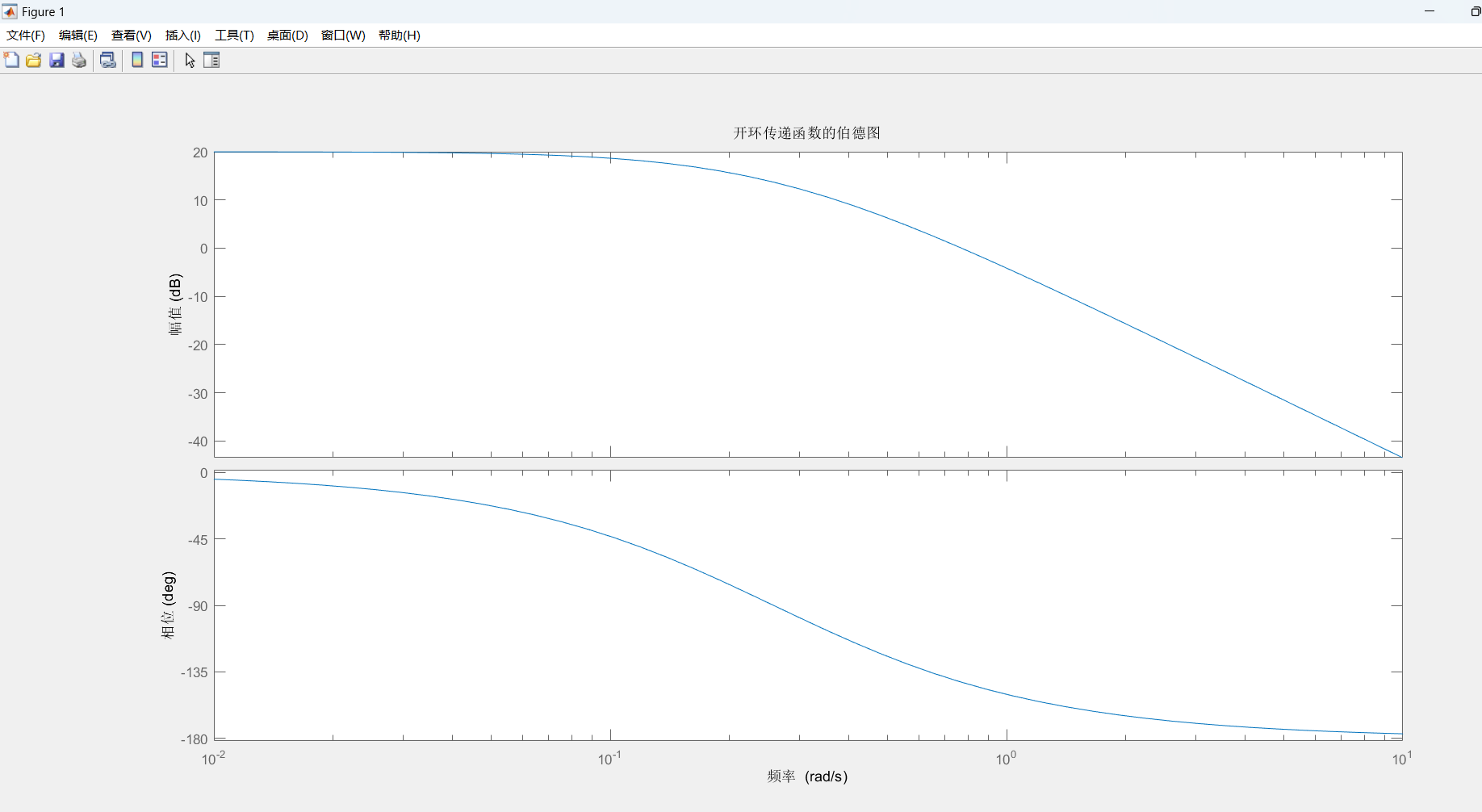
grid on;

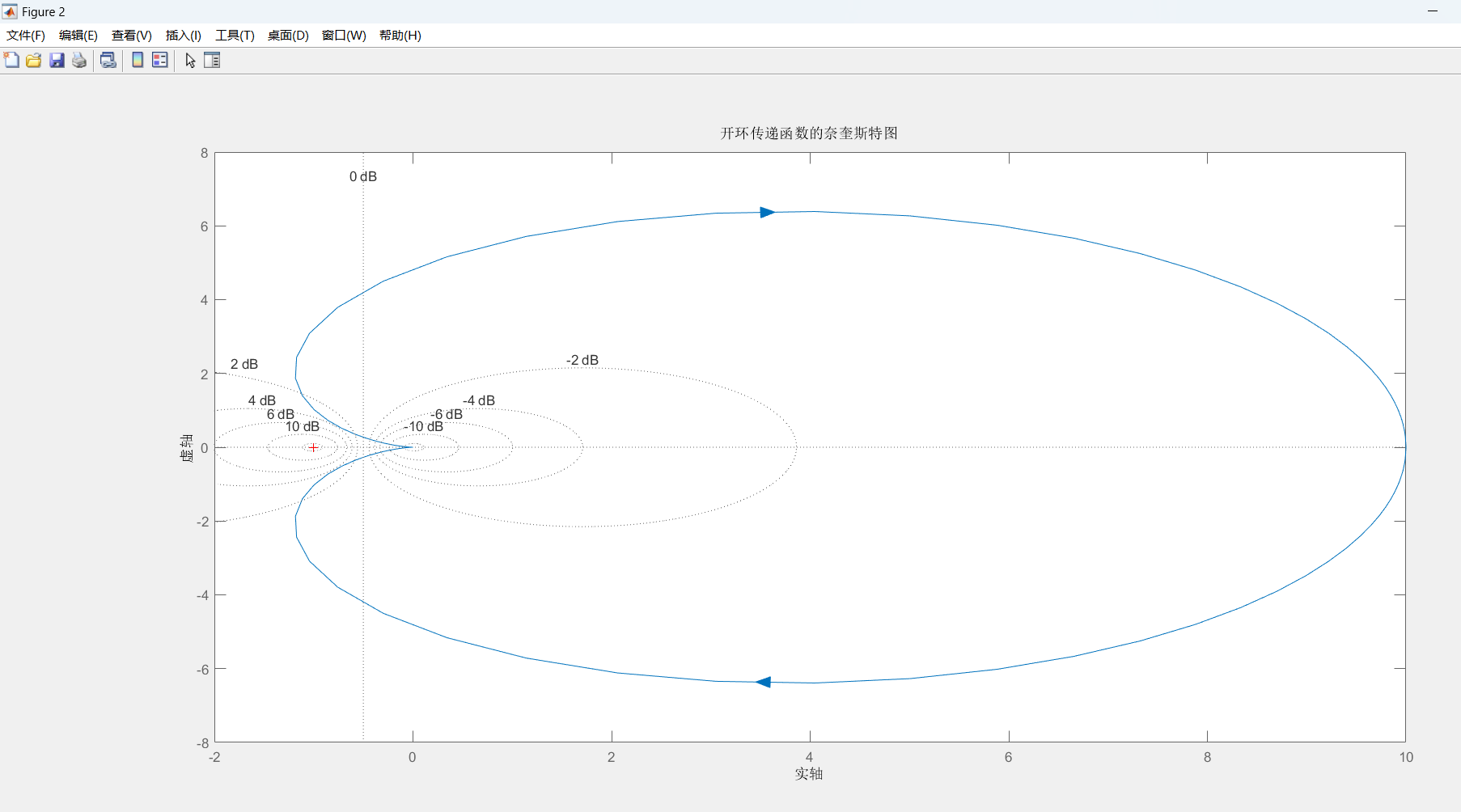
K = 1时：





K=10时：





增益变化：

由于K=10时的系统增益是K=1时的10倍，Bode图中的幅值将会提升20dB（即10倍）。

相位变化：

由于传递函数的结构相同，只是增益不同，因此相位图不会发生变化。两者的相位曲线应该是重合的。

Nyquist图分析

增益变化：Nyquist图的轨迹在增益放大后（K=10），将扩展为K=1时的10倍。曲线形状保持不变，但离原点更远。

相位变化：由于增益的变化不会影响相位，两个系统的Nyquist图形状会相同，只是大小不同。

2.

% 定义分子和分母多项式系数

num = conv([1],[1]);

den = conv([1 0],conv([3 1],[1]));

% 使用 tf 函数创建传递函数

sys = tf(num,den);

% 绘制Bode图

figure;

bode(sys);

title('开环传递函数的伯德图');

% 绘制Nyquist曲线

figure;

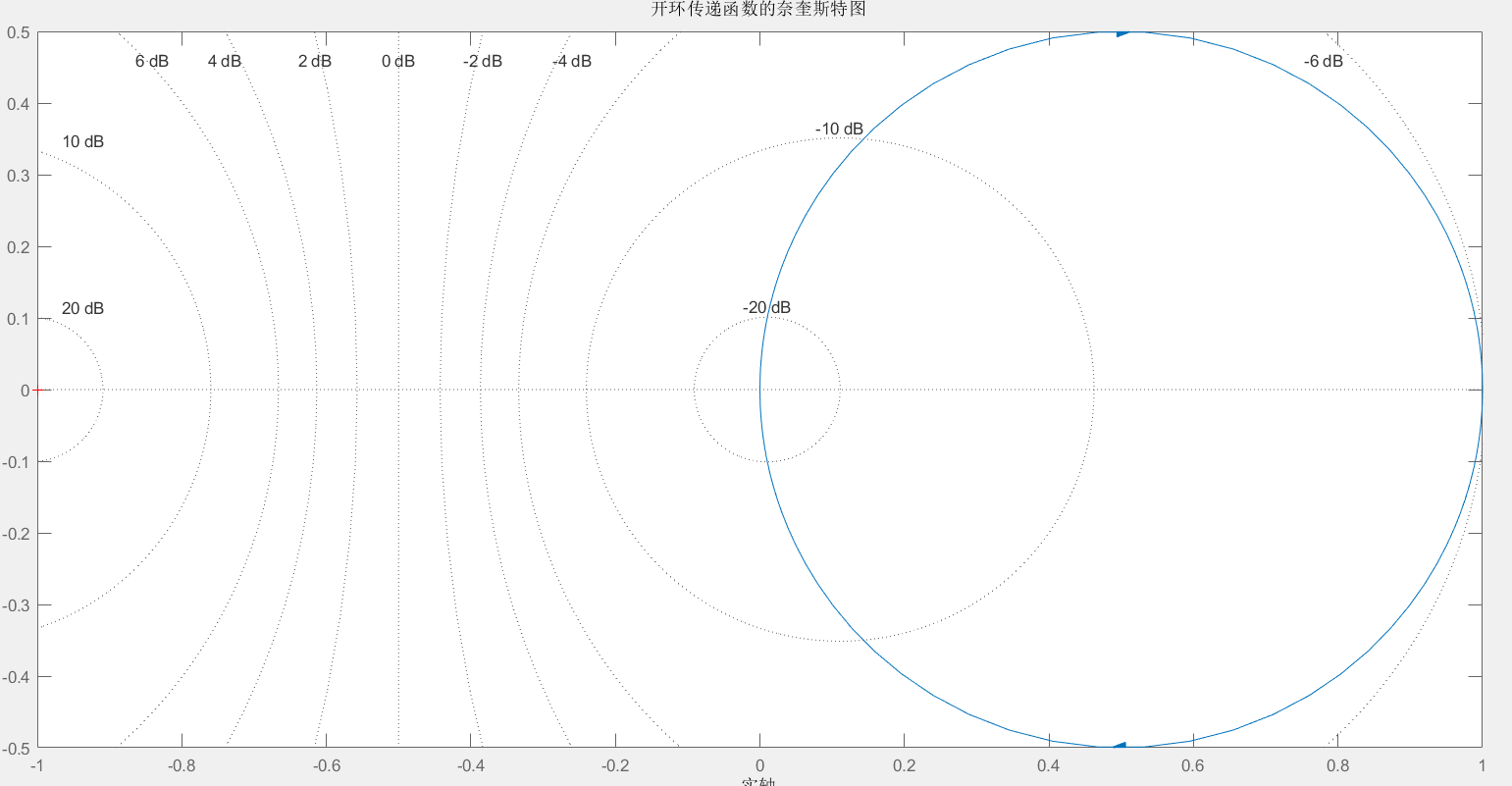
nyquist(sys);

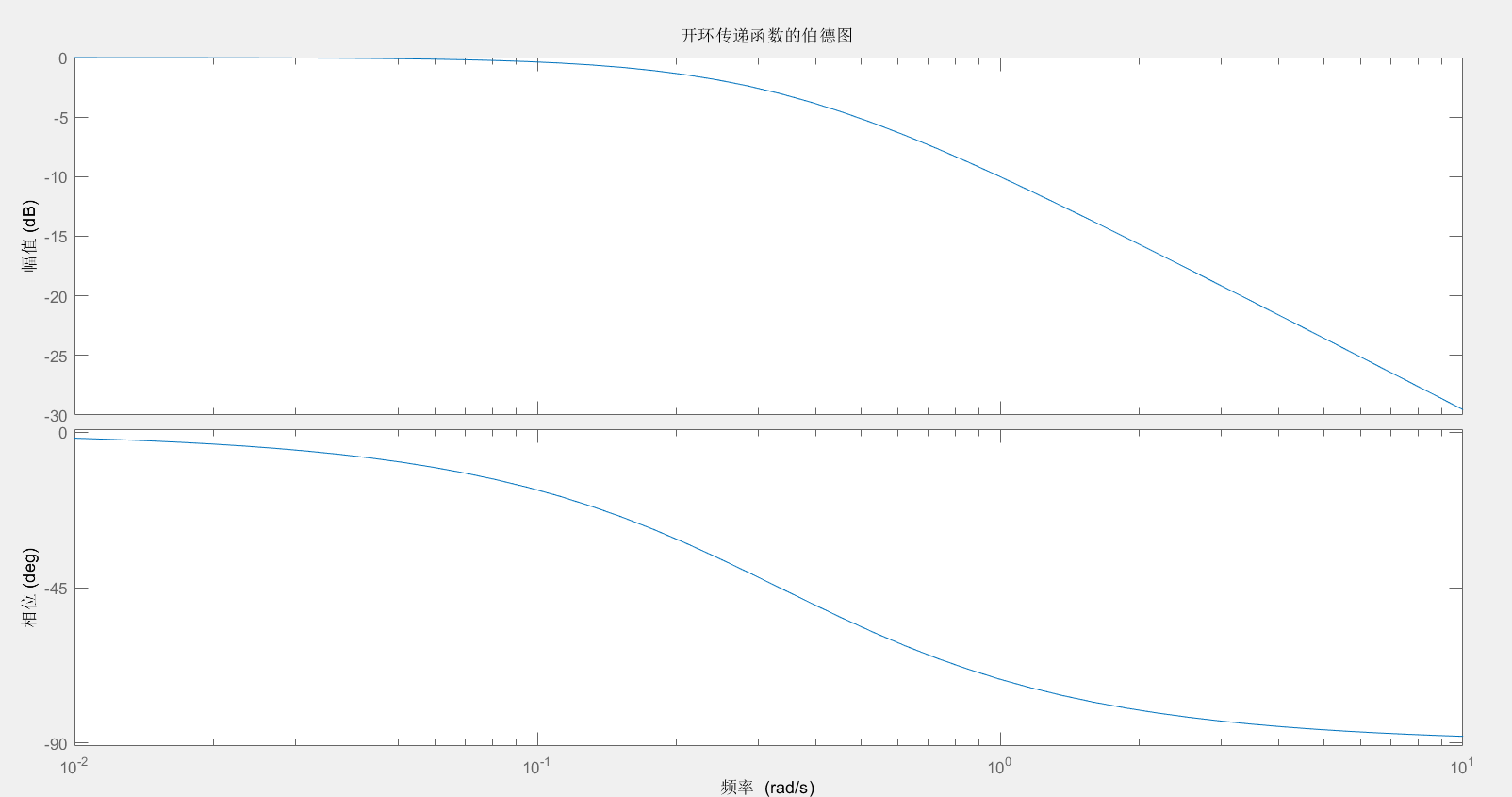
title('开环传递函数的奈奎斯特图');

% 显示图像

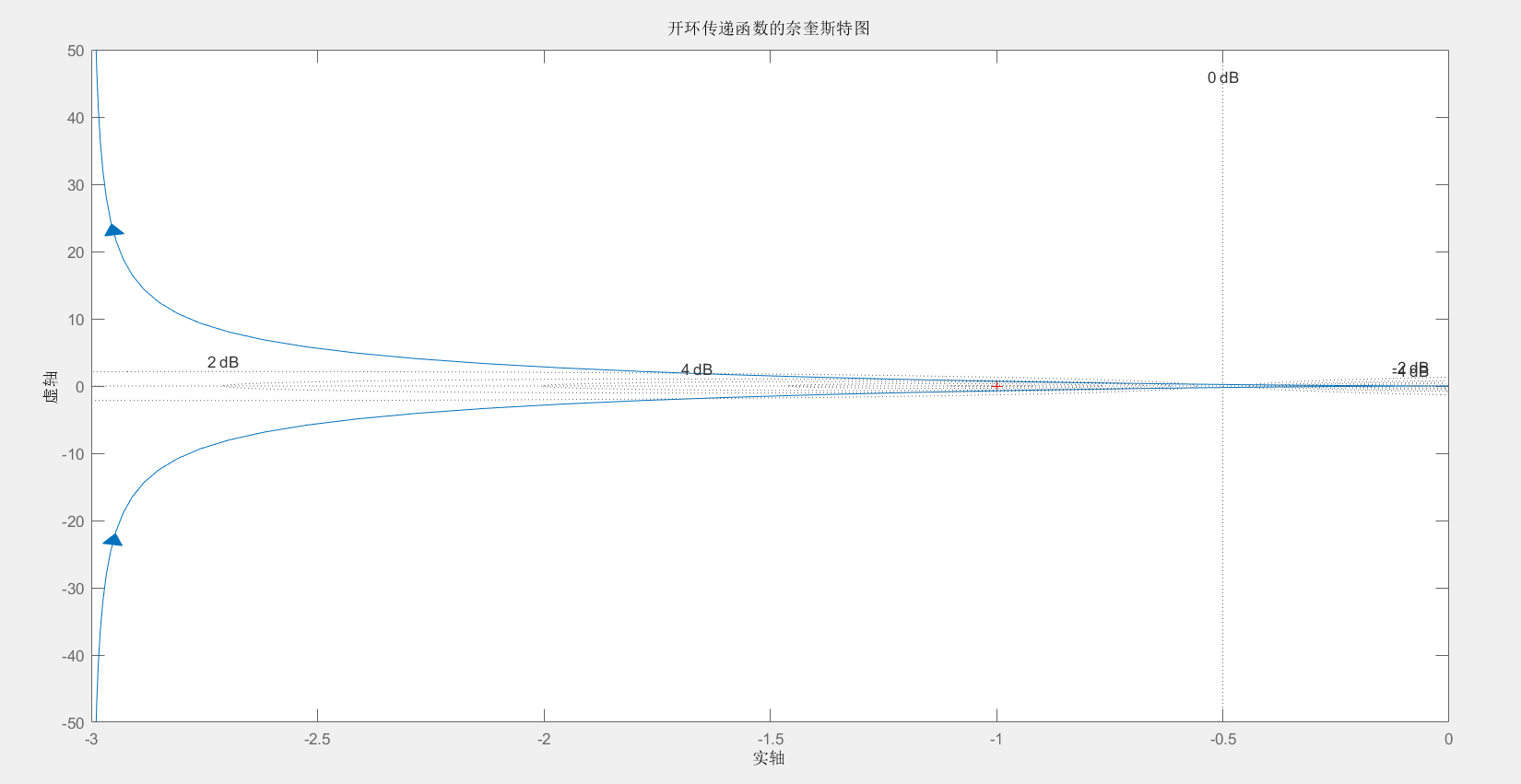
grid on;

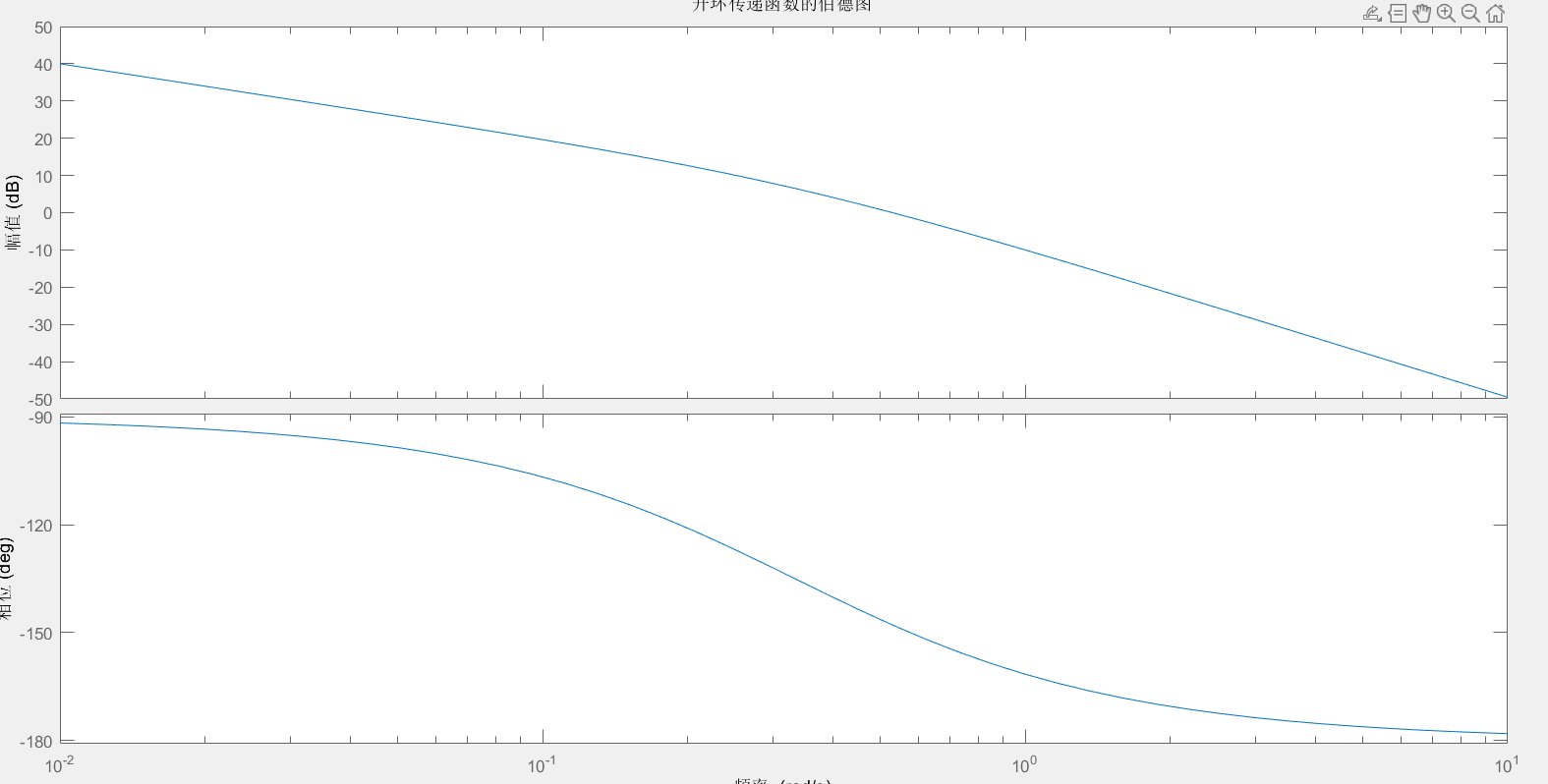
串联之前：





串联之后：





总结：

**Bode图分析**

**幅值响应：**

初始系统：幅值曲线在低频段呈现常数值，在高频段随着频率增加逐渐下降。串联1/s后的新系统幅值曲线在低频段下降20dB/dec，在高频段下降速度变为40dB/dec，因为系统的极点增加了一个。

**相位响应：**

初始系统：相位曲线大约从0度到-90度。串联1/s后的新系统相位曲线从-90度（低频）到-180度，因为增加了一个额外的极点

**Nyquist图分析**

**轨迹形状：**

初始系统：在复平面上描绘出的轨迹相对较小。

串联后的新系统轨迹扩展更大，同时由于增加了一个极点，轨迹会更偏向原点的负实轴方向。

**相位变化：**

初始系统：在绘制的复平面图中，轨迹的相位变化范围为0度到-90度。串联后的新系统 轨迹的相位变化范围为-90度到-180度，由于额外的极点导致相位整体下降了90度。

3.

% 定义开环传递函数的分子和分母多项式系数

num = 24000\*conv([1 3], [1 3]); % 分子

den = conv([1 0], conv([1 200], conv(conv([1 1], [1 2]), [1 100]))); % 分母

roots\_den = roots(den);

if all(real(roots\_den) < 0)

disp('系统稳定');

else

disp('系统不稳定');

end

% 创建传递函数

sys = tf(num, den);

% 绘制Bode图，并获取增益裕量和相位裕量

figure;

[mag, phase, w] = bode(sys);

[GM, PM, Wcg, Wcp] = margin(sys);

bode(sys);

title('伯德图');

grid on;

% 显示增益裕量和相位裕量

GM\_dB = 20\*log10(GM); % 将增益裕量转换为dB

fprintf('幅值裕量(GM): %.1f dB\n', GM\_dB);

fprintf('相位裕量 (PM): %.1f degrees\n', PM);

fprintf('幅值裕量频率值 (Wcg): %.1f rad/s\n', Wcg);

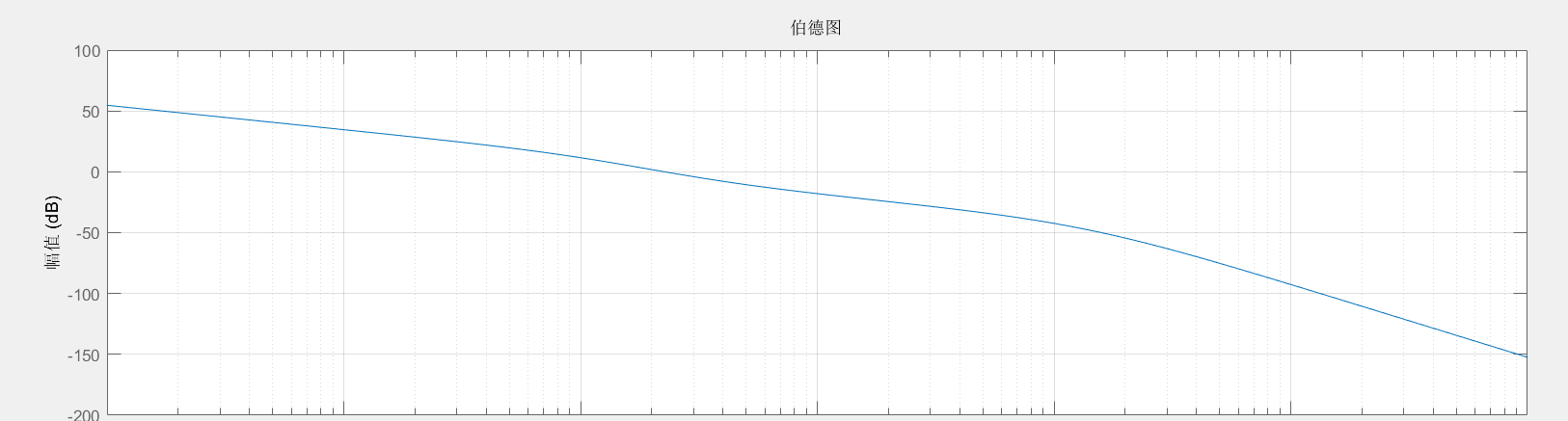
fprintf('相位裕量频率值(Wcp): %.1f rad/s\n', Wcp);

% 绘制增益裕量和相位裕量在Bode图上

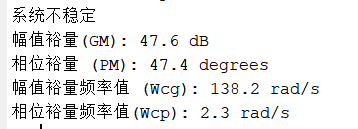
figure;

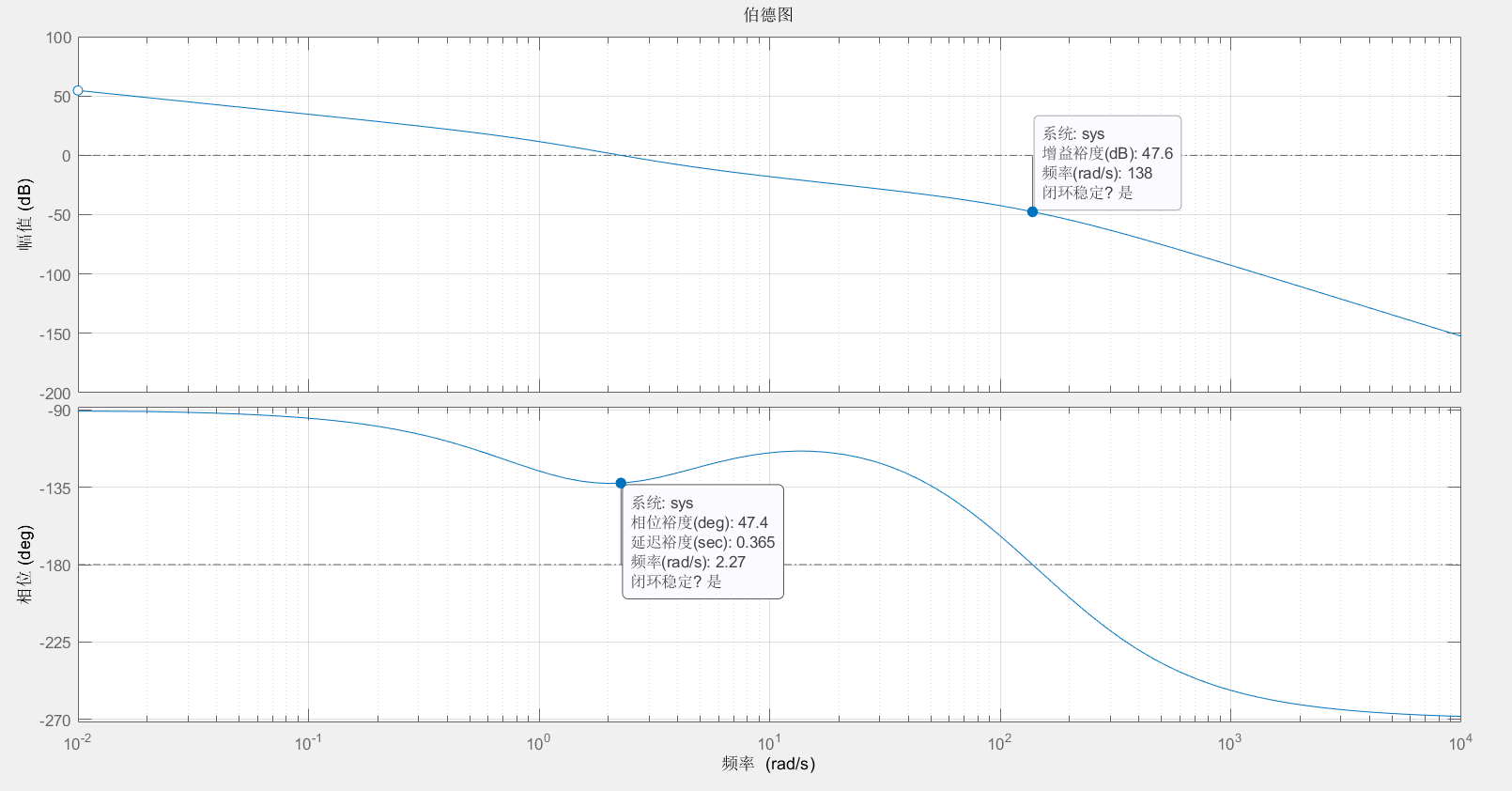
margin(sys);

伯德图中幅频特性曲线：



计算得到的值：





（2）

% 定义传递函数的分子和分母多项式系数

num = conv([1 3],[1 3]); % 分子

den = conv([1 0],conv([1 200],conv(conv([1 1], [1 2]), [1 100])));

% 创建传递函数

sys = tf(num, den);

% 获取相位裕量和增益交叉频率

[GM, PM, Wcg, Wcp] = margin(sys);

% 相位裕量临界值

PM\_critical = 0;

% 计算允许的最大延迟时间 tau

tau\_max = (PM - PM\_critical) / Wcg;

% 打印结果

fprintf('tao的最大值为: %.4f seconds\n', tau\_max);

利用公式计算得到tao能取的最大值



（3）

num = 24000 \* conv([1 3], [1 3]);

den = conv(conv(conv(conv([1 0],[1 1]),[1 2]),[1 100]),[1 200]);

sys = tf(num, den);

Kp = dcgain(sys);

syms s;

G\_s = poly2sym(num, s) / poly2sym(den, s);

Kv = limit(s \* G\_s, s, 0);

Ka = limit(s^2 \* G\_s, s, 0);

ess\_step = 1 / (1 + Kp);

ess\_ramp = 1 / Kv;

ess\_parabolic = 1 / Ka;

fprintf('位置误差系数 Kp: %.4f\n', Kp);

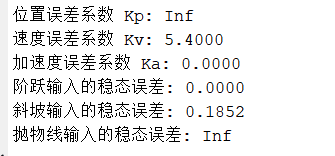
fprintf('速度误差系数 Kv: %.4f\n', double(Kv));

fprintf('加速度误差系数 Ka: %.4f\n', double(Ka));

fprintf('阶跃输入的稳态误差: %.4f\n', ess\_step);

fprintf('斜坡输入的稳态误差: %.4f\n', double(ess\_ramp));

fprintf('抛物线输入的稳态误差: %.4f\n', double(ess\_parabolic));



4.

wn = 6;

zeta = [0.5 , 0.75 , 1];

figure;

for i = 1:length(zeta)

num = [wn^2];

den = [1 2\*zeta(i)\*wn wn^2];

sys = tf(num, den);

subplot(length(zeta), 1, i);

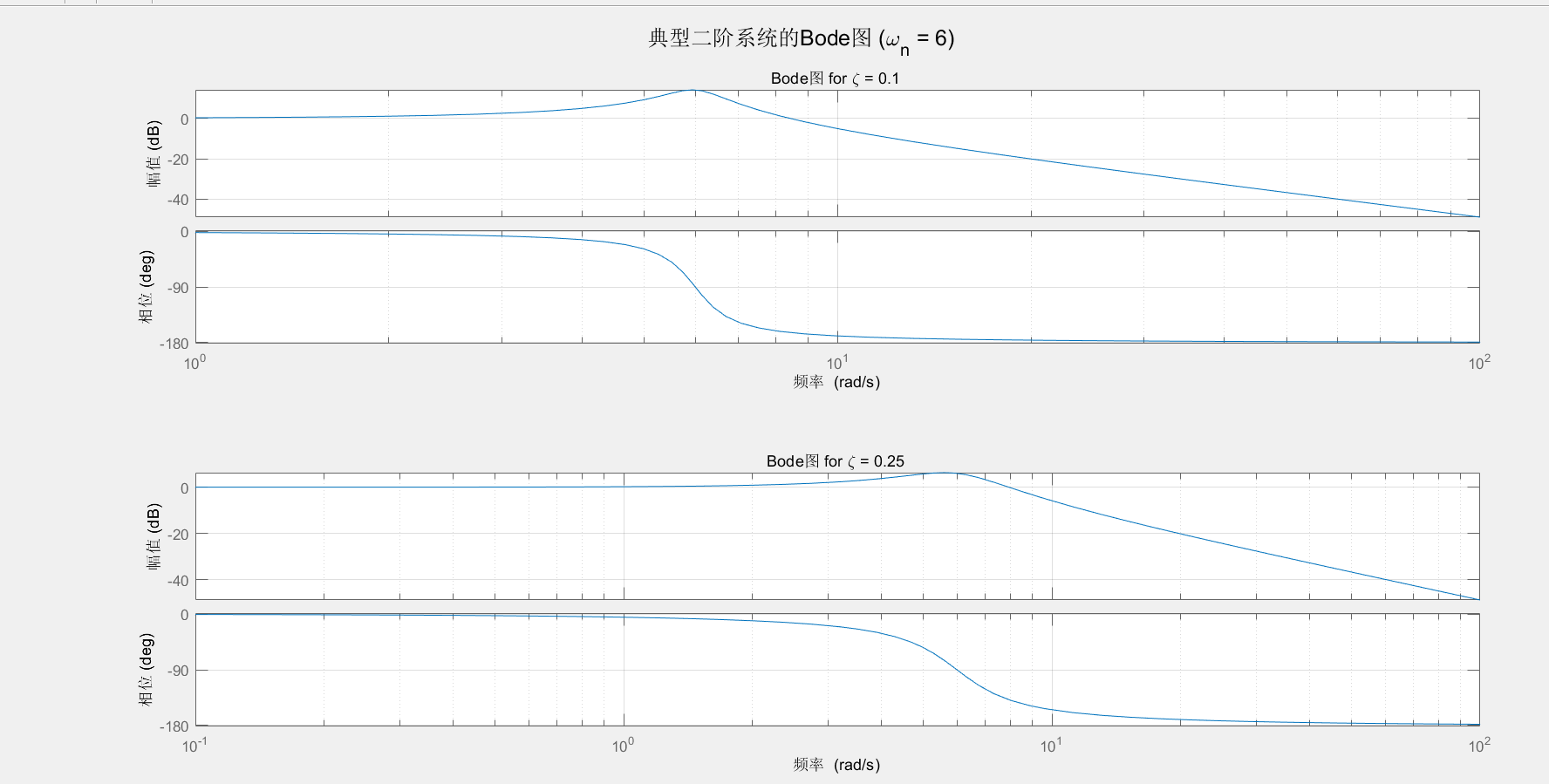
bode(sys);

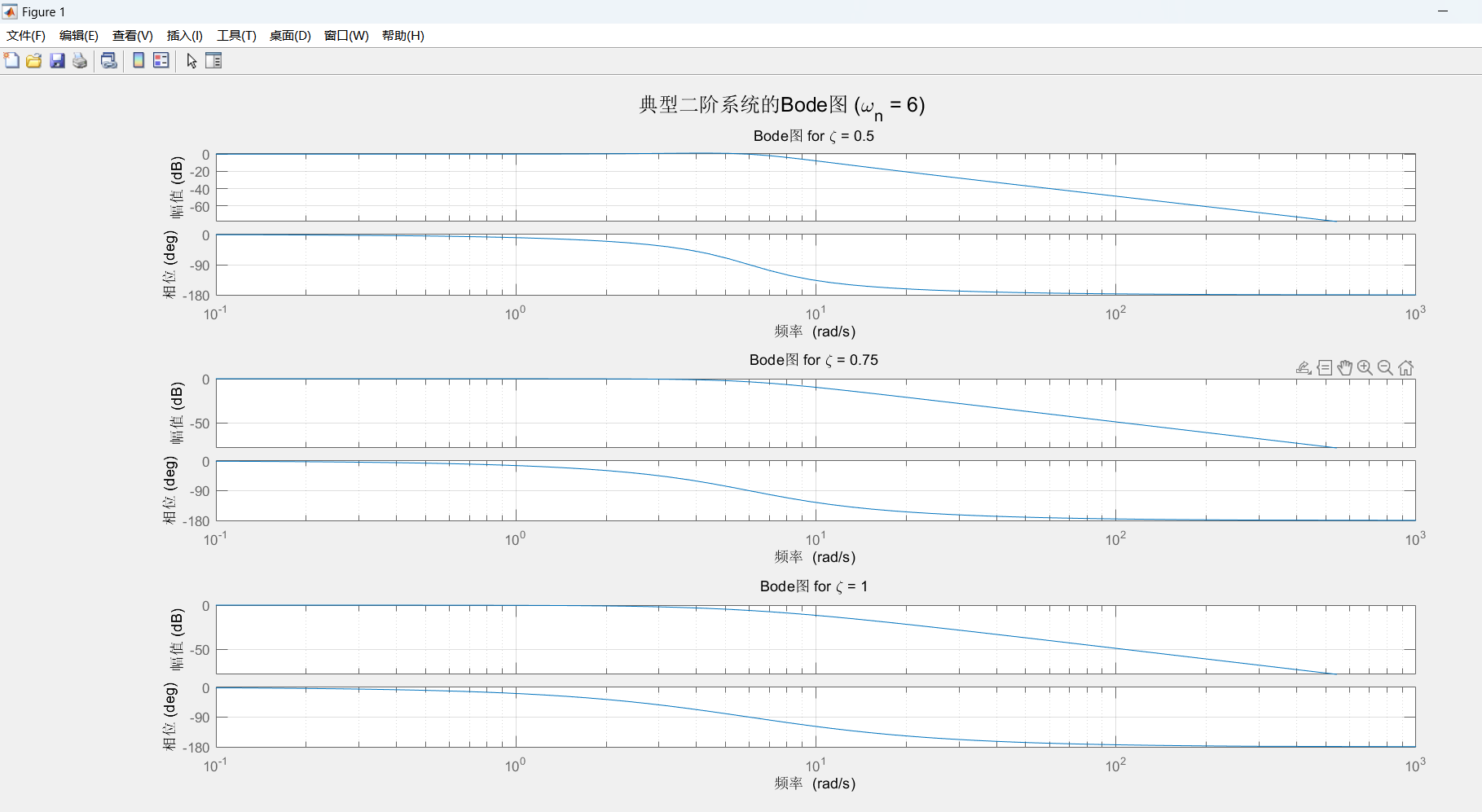
title(['Bode图 for \zeta = ', num2str(zeta(i))]);

grid on;

end

sgtitle('典型二阶系统的Bode图 (\omega\_n = 6)');





观察到相位图几乎没有很大区别；但是幅值相应图中中间突起逐渐减弱。

5.

num= [1];

den = conv([3 1], [5 1]); % (3s + 1)(5s + 1)

k\_values = [1, 3, 5, 10, 15];

figure;

for i = 1:length(k\_values)

k = k\_values(i);

num = k \* num;

sys = tf(num, den);

subplot(3, 2, i);

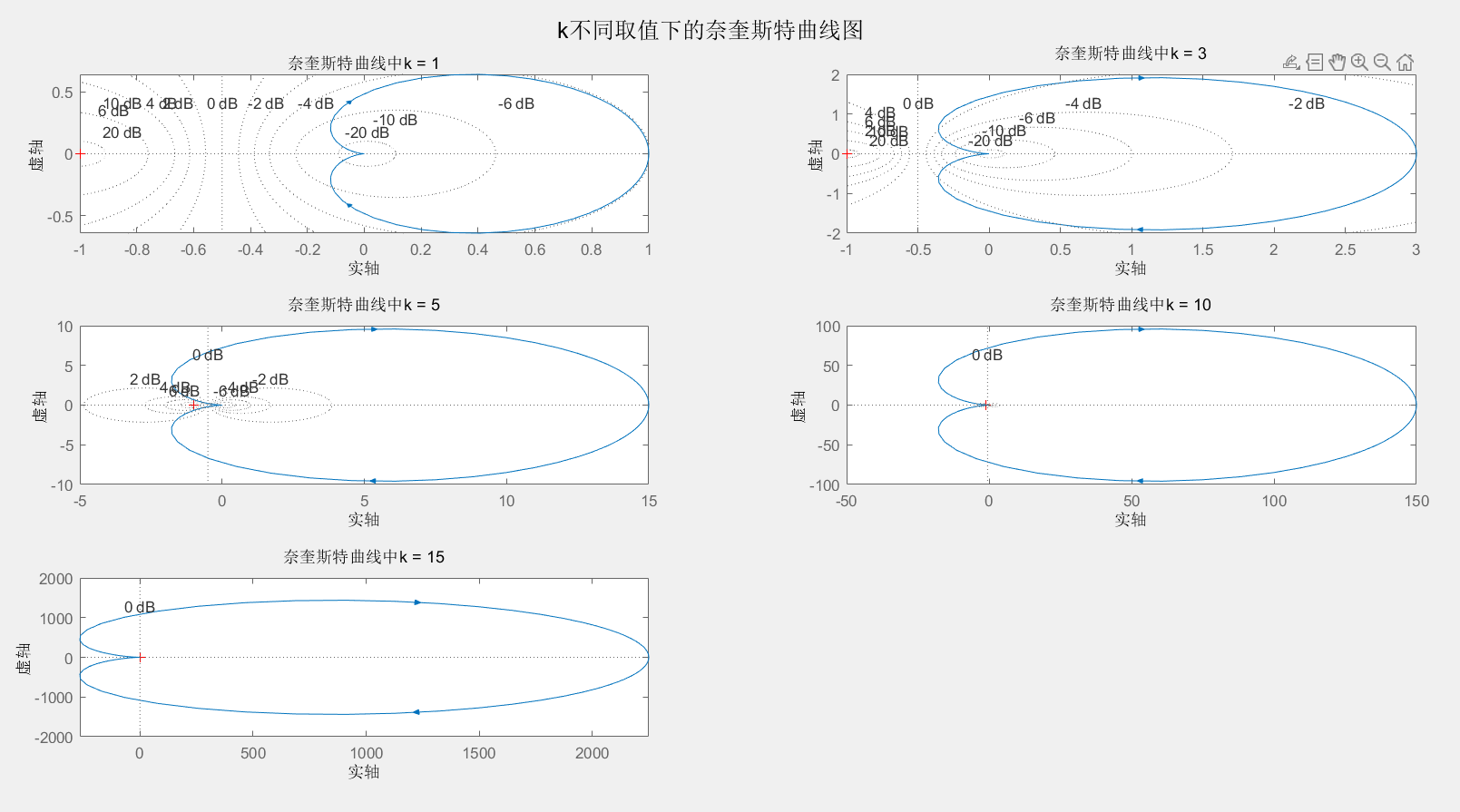
nyquist(sys);

title(['奈奎斯特曲线中k = ', num2str(k)]);

grid on;

end

sgtitle('k不同取值下的奈奎斯特曲线图');



从五个图中可以看到，增益k的增加会使系统的增益裕度和相位裕度减少，导致系统可能变得不稳定。

6.

num = [64 128];

den = conv(conv([1 0.5],[1 0]),[1 6.4 256.1]);

G = tf(num, den);

[mag, phase, wout] = bode(G);

mag\_db = 20\*log10(mag);

[GM, GM\_freq] = margin(G);

[PM, PM\_freq] = margin(G);

fprintf('幅值裕度: %.2f dB \n', GM);

fprintf('交界频率: %.2f rad/s\n',GM\_freq);

fprintf('相位裕度: %.2f degrees\n', PM, PM\_freq);

fprintf('交界频率: %.2f rad/s\n',PM\_freq);

