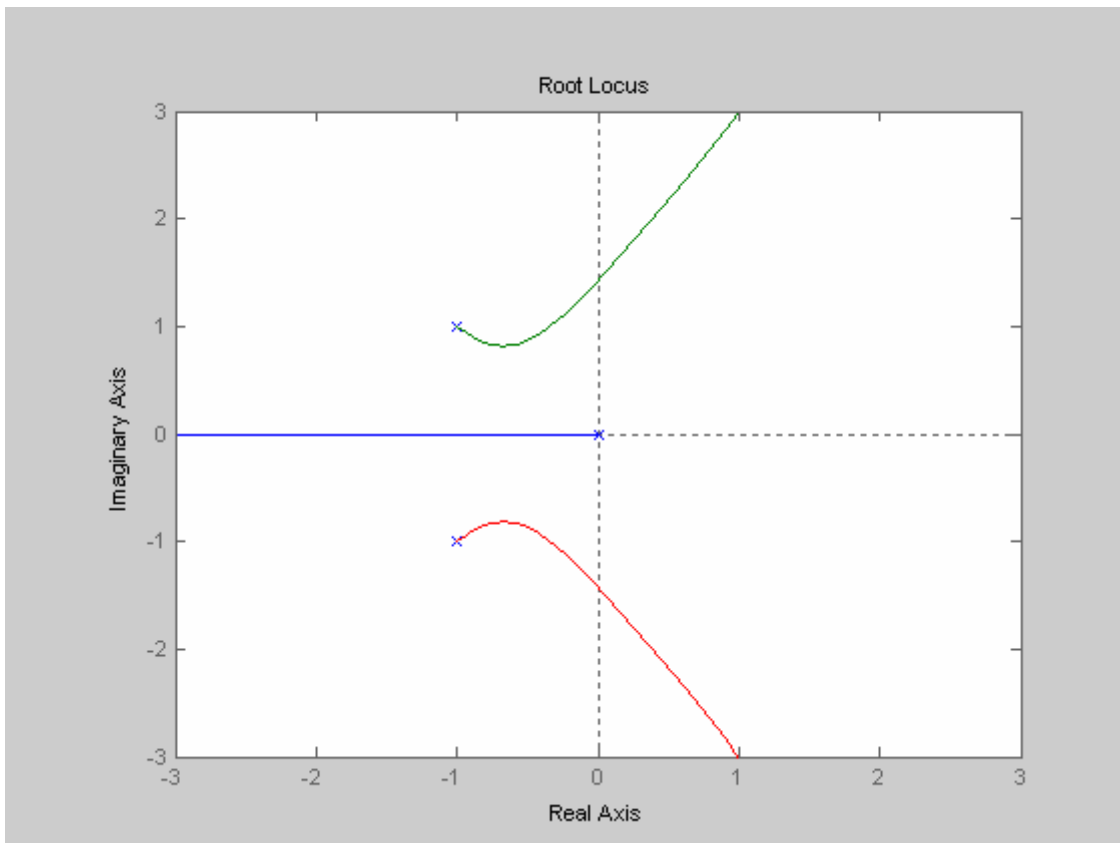


컴퓨터 예제 5-2

이 예제의 근 궤적은 MATLAB을 이용하여 다음과 같이 그릴 수 있다.

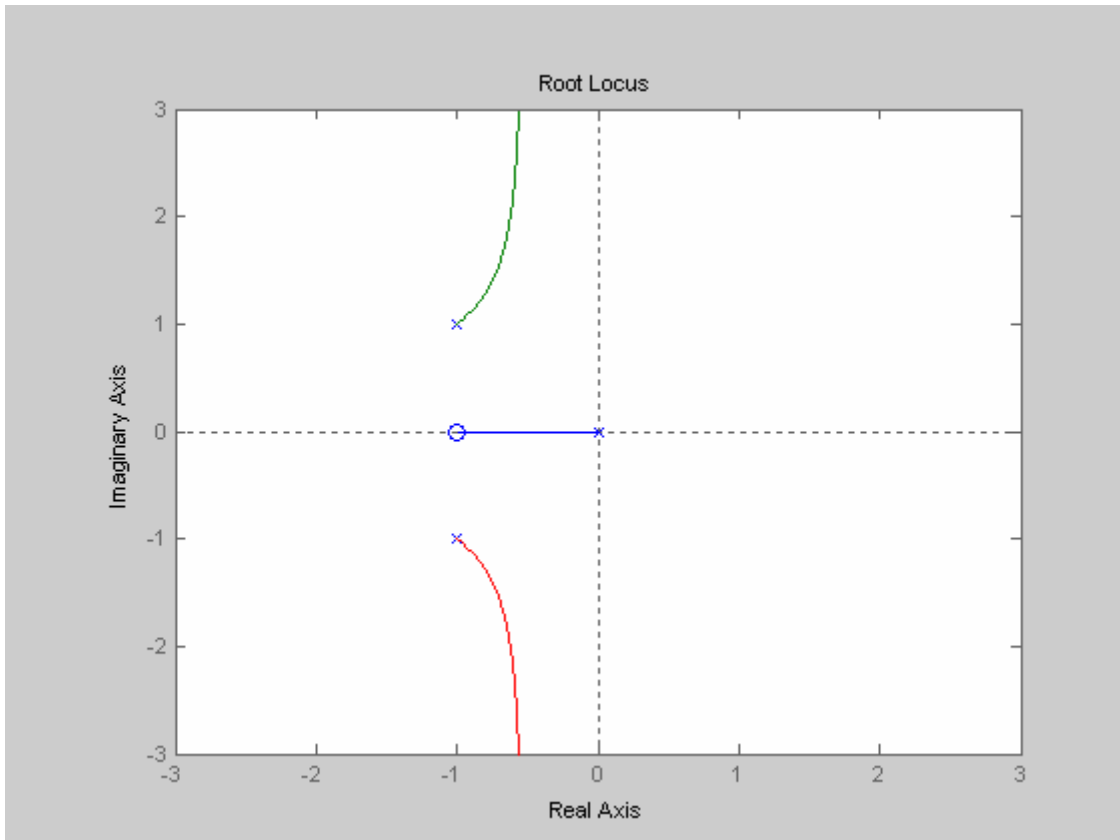
```
>> g=tf(1,[1 2 2 0]);  
>> rlocus(g)
```



컴퓨터 예제 5-3

이 예제의 근 궤적은 MATLAB을 이용하여 다음과 같이 그릴 수 있다.

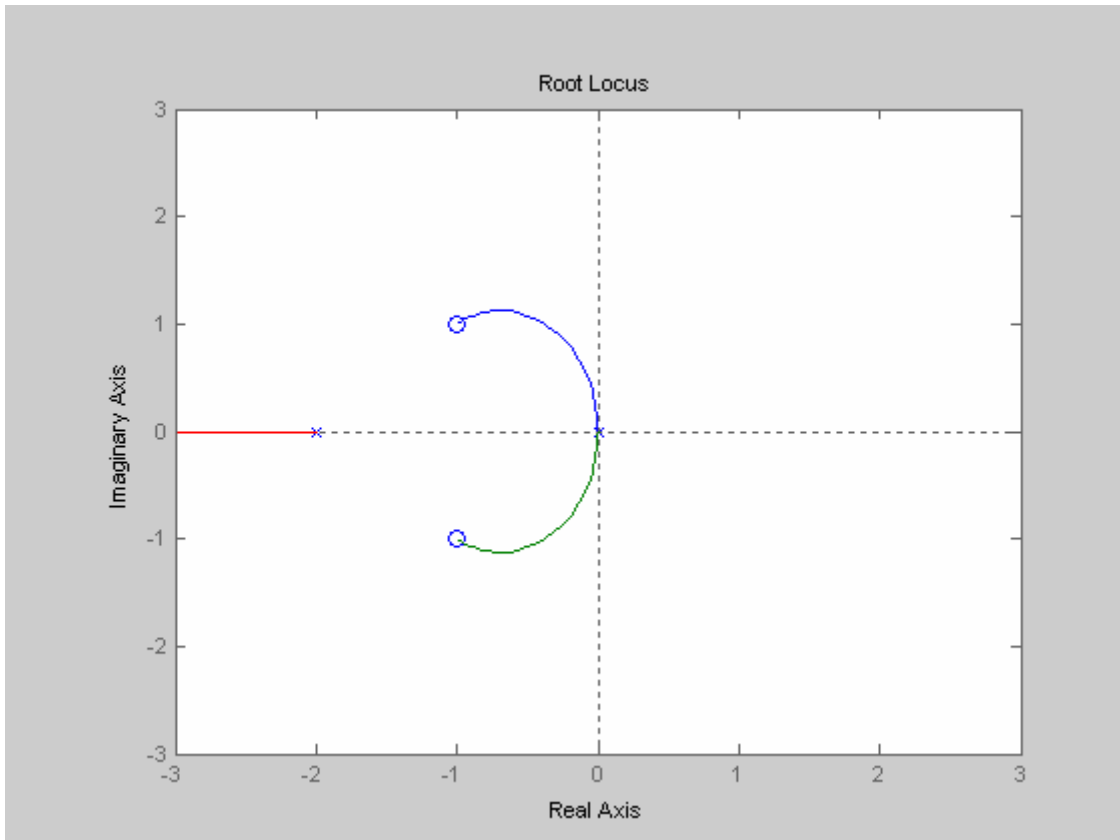
```
>> g=tf([1 1],[1 2 2 0]);  
>> rlocus(g)  
>> axis([-3 3 -3 3])
```



컴퓨터 예제 5-6

이 예제의 근 궤적은 MATLAB을 이용하여 다음과 같이 그릴 수 있다.

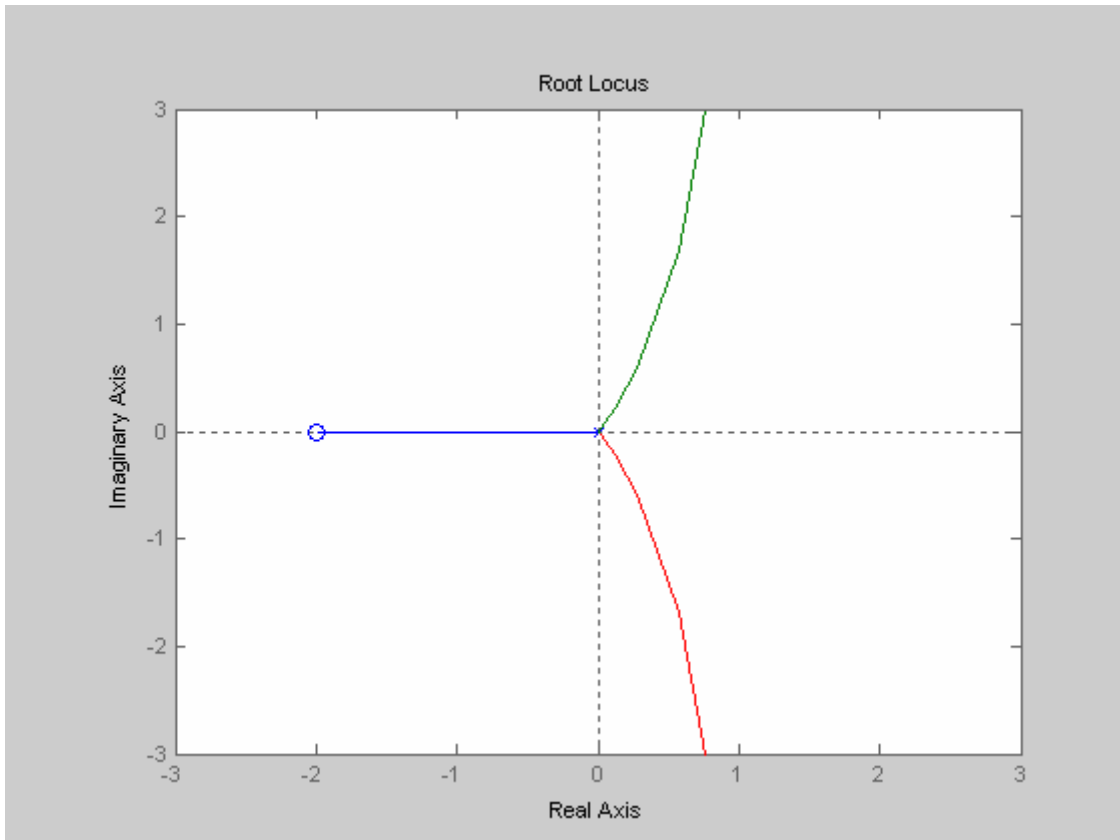
```
>> g=tf([1 2 2],[1 2 0 0]);
>> rlocus(g)
>> axis([-3 3 -3 3])
```



컴퓨터 예제 5-7

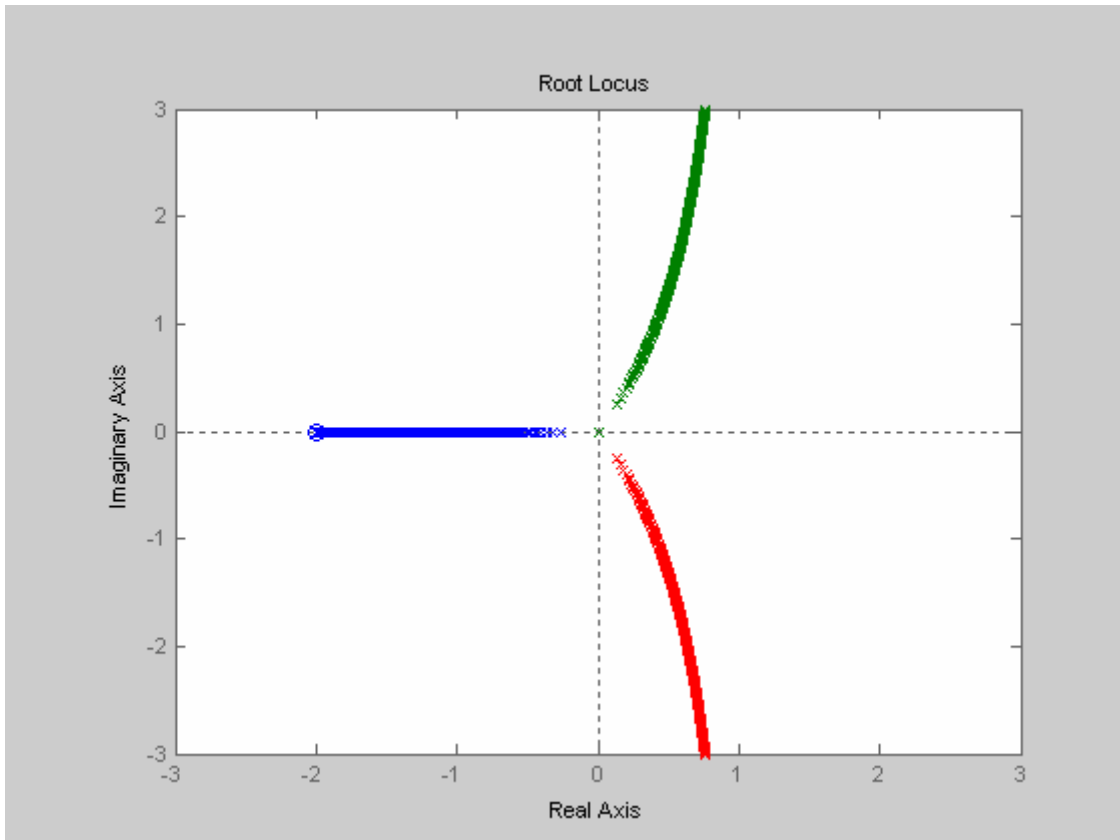
이 예제의 근 궤적은 MATLAB을 이용하여 다음과 같이 그릴 수 있다.

```
>> g=tf([1 2],[1 0 0 0]);
>> rlocus(g)
>> axis([-3 3 -3 3])
```



그런데, 이와 같이 그릴 경우 그리는 점의 개수가 적어서 근 궤적의 모양이 다소 부드럽지 못하다. 이와 같은 문제는 다음과 같이 근 궤적을 그리기 위한 계수 K의 값을 지정하면 해결될 수 있다.

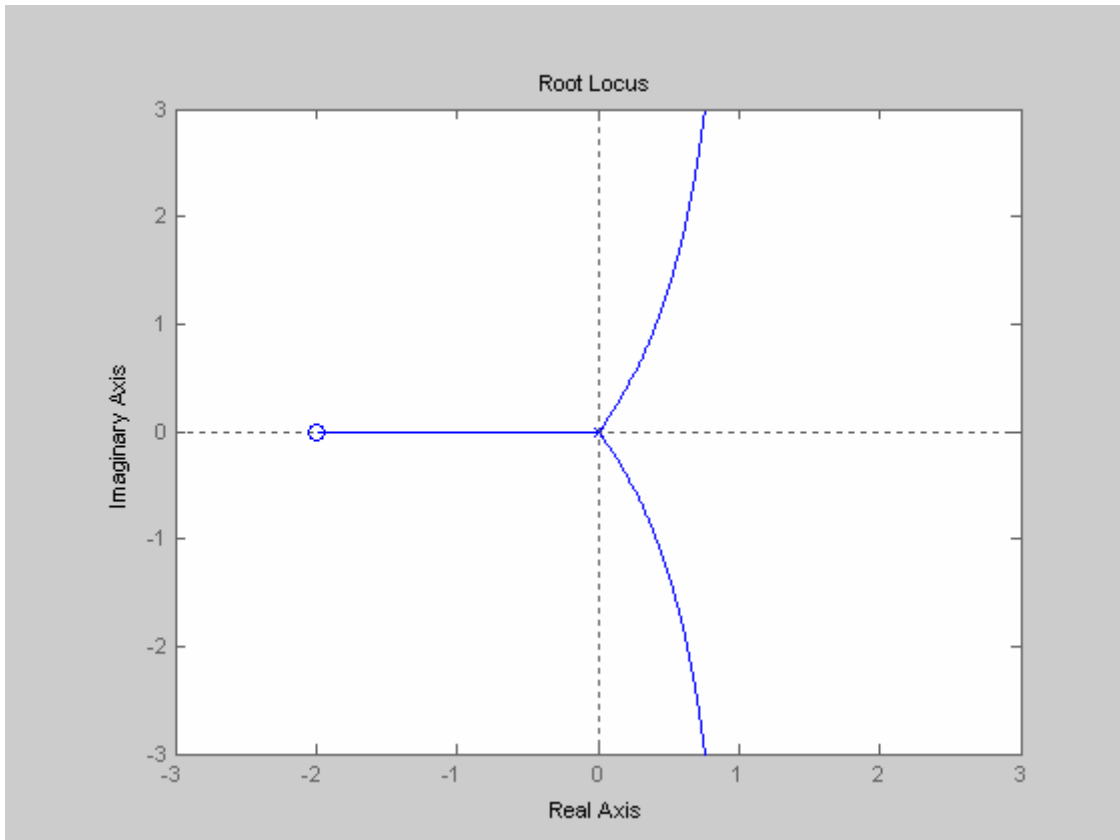
```
>> g=tf([1 2],[1 0 0 0]);
>> K=0:0.01:1000;
>> rlocus(g,K)
>> axis([-3 3 -3 3])
```



위의 근 궤적은 각 근의 위치에 x 표시를 한 것이다. 위의 근 궤적을 선으로 그리기 위해서는 다음과 같이 한다.

```
>> rlocus(g,'b-',K)
>> axis([-3 3 -3 3])
```

위의 rlocus 함수에서 'b-' 는 파란색의 직선을 의미한다.



컴퓨터 예제 5-8

이 예제의 근 궤적은 $K=4$ 에서 허수 축을 지난다. 이것을 확인하기 위해서 MATLAB 에서 $K=3$ 에서 $K=5$ 까지의 근을 계산하면 다음과 같다.

```
>> K=3:0.1:5;
>> g=tf([1 2],[1 0 0 0]);
>> [r,K]=rlocus(g,K);
>> [K' r(1,:) r(2,:) r(3,:)]
```

ans =

3.0000	-1.8105	-0.0947 - 1.2837i	-0.0947 + 1.2837i
3.1000	-1.8320	-0.0840 - 1.2981i	-0.0840 + 1.2981i
3.2000	-1.8527	-0.0736 - 1.3122i	-0.0736 + 1.3122i
3.3000	-1.8729	-0.0635 - 1.3259i	-0.0635 + 1.3259i
3.4000	-1.8925	-0.0537 - 1.3393i	-0.0537 + 1.3393i
3.5000	-1.9116	-0.0442 - 1.3524i	-0.0442 + 1.3524i
3.6000	-1.9301	-0.0349 - 1.3653i	-0.0349 + 1.3653i
3.7000	-1.9482	-0.0259 - 1.3779i	-0.0259 + 1.3779i
3.8000	-1.9659	-0.0171 - 1.3902i	-0.0171 + 1.3902i
3.9000	-1.9831	-0.0084 - 1.4023i	-0.0084 + 1.4023i
4.0000	-2.0000	0.0000 - 1.4142i	0.0000 + 1.4142i

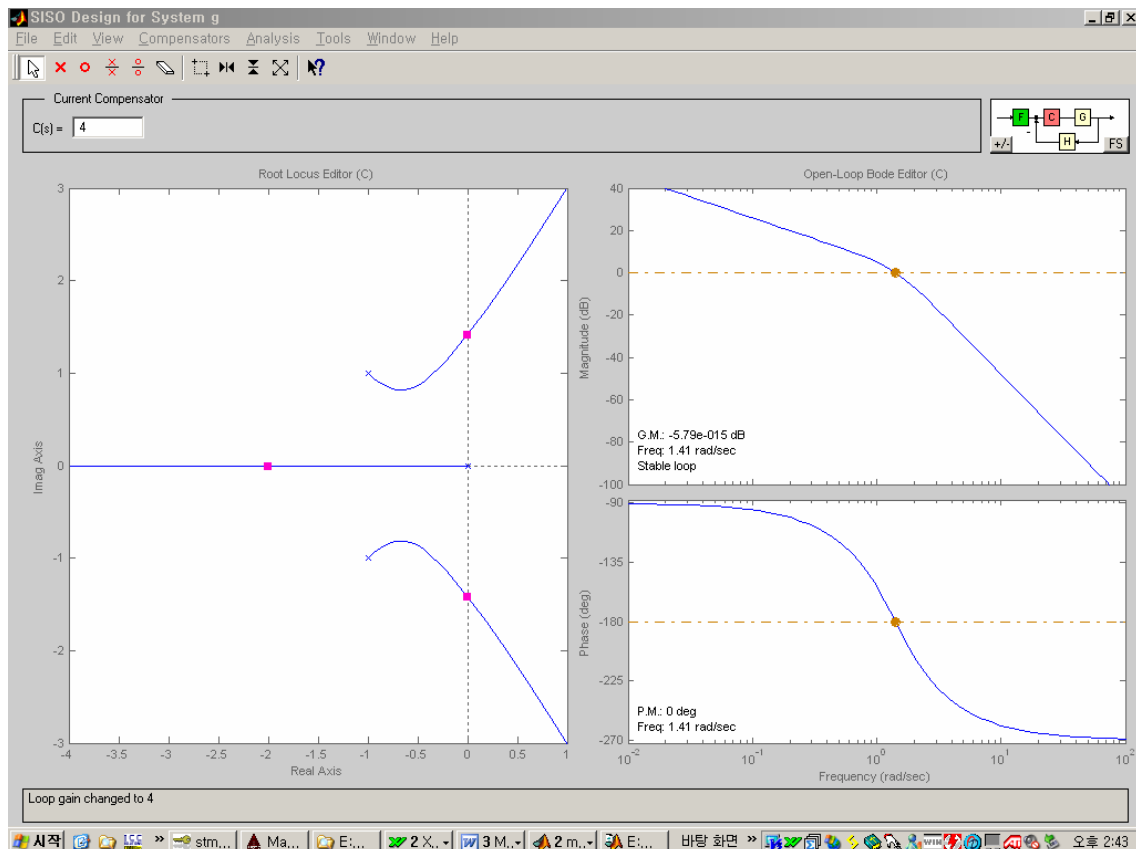
4.1000	-2.0165	0.0082 - 1.4259i	0.0082 + 1.4259i
4.2000	-2.0326	0.0163 - 1.4374i	0.0163 + 1.4374i
4.3000	-2.0484	0.0242 - 1.4487i	0.0242 + 1.4487i
4.4000	-2.0639	0.0320 - 1.4597i	0.0320 + 1.4597i
4.5000	-2.0791	0.0395 - 1.4707i	0.0395 + 1.4707i
4.6000	-2.0940	0.0470 - 1.4814i	0.0470 + 1.4814i
4.7000	-2.1086	0.0543 - 1.4920i	0.0543 + 1.4920i
4.8000	-2.1229	0.0615 - 1.5024i	0.0615 + 1.5024i
4.9000	-2.1370	0.0685 - 1.5127i	0.0685 + 1.5127i
5.0000	-2.1509	0.0755 - 1.5228i	0.0755 + 1.5228i

위의 숫자들에서 가장 왼쪽은 K의 값이며 오른쪽의 숫자들은 3개의 근의 값들이다. K=4에서 두 개의 근이 허수 축 위에 있음을 확인할 수 있다.

또 다른 방법으로는 아래와 같이 sisotool을 사용하는 방법이 있다.

```
>> g=tf(1,[1 2 2 0]);
>> sisotool(g)
```

아래 그림의 화면에서 마우스를 이용하여 근의 위치를 옮기면 그에 따라서 계수의 값이 변하는 것은 볼 수 있으며, 반대로 근의 값을 변화하면 그에 따라서 근의 위치가 바뀌는 것도 볼 수 있다. 그림에서 계수의 값이 4일 때 근이 허수 축에 있음을 알 수 있다.



컴퓨터 예제 5-10

근 궤적이 만나서 2중근이 되는 점을 확인하기 위하여, K의 값이 3 에서 7 까지 변화하는 동안의 근의 위치를 MATLAB을 이용하여 계산해 본다. 또는 위의 예제와 같이 sisotool을 이용하여도 근 궤적의 움직임을 확인할 수 있다.

```
>> g=tf([1,conv([1 2 0],[1 2 5]))
```

Transfer function:

$$\frac{1}{s^4 + 4s^3 + 9s^2 + 10s}$$

```
>> K=3:0.1:7;  
>> [r,K]=rlocus(g,K);  
>> [K' r(1,:) r(2,:) r(3,:) r(4,:)]
```

ans =

3.0000	-1.0000 - 1.8174i	-1.0000 + 1.8174i	-1.5503	-0.4497
3.1000	-1.0000 - 1.8096i	-1.0000 + 1.8096i	-1.5242	-0.4758
3.2000	-1.0000 - 1.8018i	-1.0000 + 1.8018i	-1.4964	-0.5036
3.3000	-1.0000 - 1.7938i	-1.0000 + 1.7938i	-1.4664	-0.5336
3.4000	-1.0000 - 1.7856i	-1.0000 + 1.7856i	-1.4338	-0.5662
3.5000	-1.0000 - 1.7772i	-1.0000 + 1.7772i	-1.3979	-0.6021
3.6000	-1.0000 - 1.7686i	-1.0000 + 1.7686i	-1.3576	-0.6424
3.7000	-1.0000 - 1.7598i	-1.0000 + 1.7598i	-1.3112	-0.6888
3.8000	-1.0000 - 1.7508i	-1.0000 + 1.7508i	-1.2554	-0.7446
3.9000	-1.0000 - 1.7415i	-1.0000 + 1.7415i	-1.1816	-0.8184
4.0000	-1.0000 - 1.7321i	-1.0000 + 1.7321i	-1.0000 + 0.0000i	-1.0000 - 0.0000i
4.1000	-1.0000 - 1.7223i	-1.0000 + 1.7223i	-1.0000 + 0.1836i	-1.0000 - 0.1836i
4.2000	-1.0000 - 1.7122i	-1.0000 + 1.7122i	-1.0000 + 0.2612i	-1.0000 - 0.2612i
4.3000	-1.0000 - 1.7019i	-1.0000 + 1.7019i	-1.0000 + 0.3218i	-1.0000 - 0.3218i
4.4000	-1.0000 - 1.6912i	-1.0000 + 1.6912i	-1.0000 + 0.3740i	-1.0000 - 0.3740i
4.5000	-1.0000 - 1.6801i	-1.0000 + 1.6801i	-1.0000 + 0.4209i	-1.0000 - 0.4209i
4.6000	-1.0000 - 1.6687i	-1.0000 + 1.6687i	-1.0000 + 0.4642i	-1.0000 - 0.4642i
4.7000	-1.0000 - 1.6568i	-1.0000 + 1.6568i	-1.0000 + 0.5050i	-1.0000 - 0.5050i
4.8000	-1.0000 - 1.6444i	-1.0000 + 1.6444i	-1.0000 + 0.5439i	-1.0000 - 0.5439i
4.9000	-1.0000 - 1.6315i	-1.0000 + 1.6315i	-1.0000 + 0.5815i	-1.0000 - 0.5815i
5.0000	-1.0000 - 1.6180i	-1.0000 + 1.6180i	-1.0000 + 0.6180i	-1.0000 - 0.6180i
5.1000	-1.0000 - 1.6039i	-1.0000 + 1.6039i	-1.0000 + 0.6539i	-1.0000 - 0.6539i
5.2000	-1.0000 - 1.5889i	-1.0000 + 1.5889i	-1.0000 + 0.6894i	-1.0000 - 0.6894i
5.3000	-1.0000 - 1.5731i	-1.0000 + 1.5731i	-1.0000 + 0.7248i	-1.0000 - 0.7248i
5.4000	-1.0000 - 1.5563i	-1.0000 + 1.5563i	-1.0000 + 0.7603i	-1.0000 - 0.7603i
5.5000	-1.0000 - 1.5382i	-1.0000 + 1.5382i	-1.0000 + 0.7962i	-1.0000 - 0.7962i
5.6000	-1.0000 - 1.5186i	-1.0000 + 1.5186i	-1.0000 + 0.8329i	-1.0000 - 0.8329i
5.7000	-1.0000 - 1.4972i	-1.0000 + 1.4972i	-1.0000 + 0.8709i	-1.0000 - 0.8709i
5.8000	-1.0000 - 1.4734i	-1.0000 + 1.4734i	-1.0000 + 0.9106i	-1.0000 - 0.9106i
5.9000	-1.0000 - 1.4462i	-1.0000 + 1.4462i	-1.0000 + 0.9531i	-1.0000 - 0.9531i
6.0000	-1.0000 - 1.4142i	-1.0000 + 1.4142i	-1.0000 + 1.0000i	-1.0000 - 1.0000i

6.1000	-1.0000 - 1.3738i	-1.0000 + 1.3738i	-1.0000 + 1.0548i	-1.0000 - 1.0548i
6.2000	-1.0000 - 1.3129i	-1.0000 + 1.3129i	-1.0000 + 1.1298i	-1.0000 - 1.1298i
6.3000	-1.0910 - 1.2281i	-1.0910 + 1.2281i	-0.9090 + 1.2281i	-0.9090 - 1.2281i
6.4000	-1.1568 - 1.2347i	-1.1568 + 1.2347i	-0.8432 + 1.2347i	-0.8432 - 1.2347i
6.5000	-1.2014 - 1.2412i	-1.2014 + 1.2412i	-0.7986 + 1.2412i	-0.7986 - 1.2412i
6.6000	-1.2371 - 1.2475i	-1.2371 + 1.2475i	-0.7629 + 1.2475i	-0.7629 - 1.2475i
6.7000	-1.2676 - 1.2536i	-1.2676 + 1.2536i	-0.7324 + 1.2536i	-0.7324 - 1.2536i
6.8000	-1.2944 - 1.2596i	-1.2944 + 1.2596i	-0.7056 + 1.2596i	-0.7056 - 1.2596i
6.9000	-1.3185 - 1.2655i	-1.3185 + 1.2655i	-0.6815 + 1.2655i	-0.6815 - 1.2655i
7.0000	-1.3406 - 1.2712i	-1.3406 + 1.2712i	-0.6594 + 1.2712i	-0.6594 - 1.2712i

컴퓨터 예제 5-12

음의 계수에 대한 근 궤적을 MATLAB으로 그리기 위해서는 다음과 같이 분모의 계수를 나타내는 row vector의 앞에 - 부호를 붙인다.

```
>> g=tf(-[1],[1 2 2 0])
```

Transfer function:

-1

s^3 + 2 s^2 + 2 s

```
>> rlocus(g)
```

