Matlab Tutorial : Root Locus

ECE 3510

Heather Malko

THE
UNIVERSITY
OF UTAH

Table of Context

- 1.0 Introduction
- 2.0 Root Locus Design
- 3.0 SISO Root Locus
- 4.0 GUI for Controls

1.0 Introduction

- A root loci plot is simply a plot of the s zero values and the s poles on a graph with real and imaginary ordinates.

- The root locus is a curve of the location of the poles of a transfer function as some parameter (generally the gain K) is varied.

- The locus of the roots of the characteristic equation of the closed loop system as the gain varies from zero to infinity gives the name of the method.

- This method is very powerful graphical technique for investigating the effects of the variation of a system parameter on the locations of the closed loop poles.

1.0 Introduction

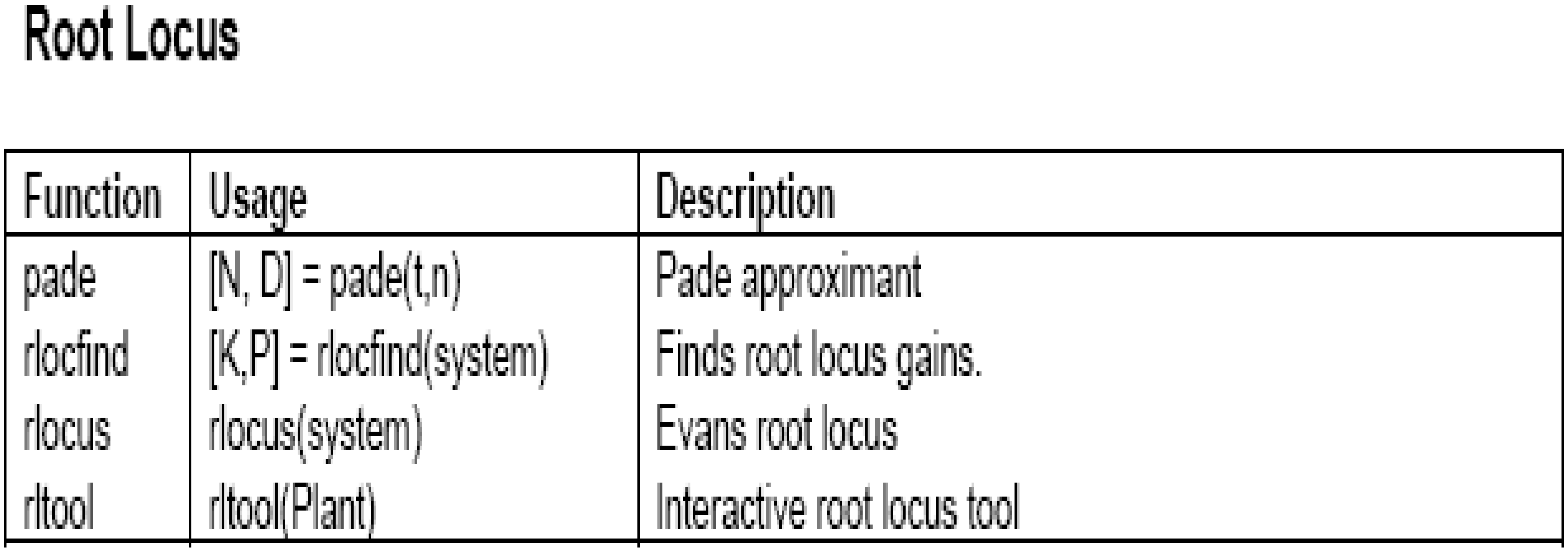
- General rules for constructing the root locus exist and if the designer follows them, sketching of the root loci becomes a simple matter.

- Root loci are completed to select the best parameter value for stability.

- A normal interpretation of improving stability is when the real part of a pole is further left of the imaginary axis.

1.0 Introduction

- Matlab and Root Locus:
- MATLAB Control System Toolbox contains two Root Locus design GUI, sisotool and rltool. These are two interactive design tools of SISO.



Root Locus

Function	Usage	Description
pade	[N, D] = pade(t,n)	Pade approximant
rlocfind	[K,P] = rlocfind(system)	Finds root locus gains.
rlocus	rlocus(system)	Evans root locus
rltool	rltool(Plant)	Interactive root locus tool

1.0 Introduction

- Matlab’s Useful Commands :
- http://courses.ece.uiuc.edu/ece486/documents/matlab_cmds.html
- http://www-ccs.ucsd.edu/matlab/toolbox/control/refintro.html
- http://users.ece.gatech.edu/~bonnie/book/TUTORIAL/tut_3.html

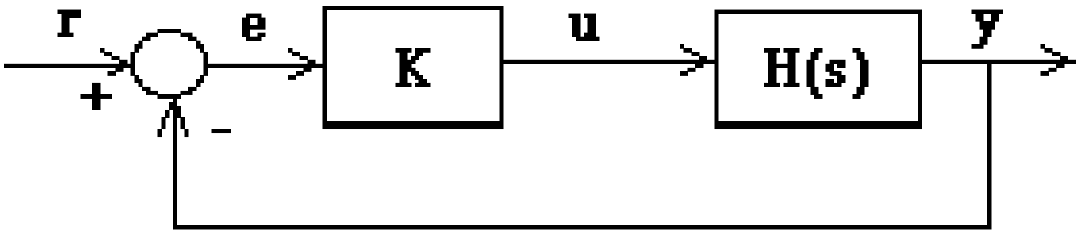
2.0 Root Locus Tutorial # 1

- Key Matlab commands used in this tutorial: **cloop, rlocfind, rlocus, sgrid, step**

- Matlab commands from the control system toolbox are highlighted in red.

2.0 Root Locus Design

- The root locus of an (open-loop) transfer function H(s) is a plot of the locations (locus) of all possible closed loop poles with proportional gain k and unity feedback:



2.0 Root Locus Design

- The closed-loop transfer function is:

$$\frac{Y(s)}{R(s)} = \frac{KH(s)}{1 + KH(s)}$$

  and thus the poles of the closed loop system are values of s such that **1 + K H(s) = 0**.

- If **H(s) = b(s)/a(s)**, then this equation has the form:

$$a(s) + K\,b(s) = 0$$

$$\frac{a(s)}{K} + b(s) = 0$$

- Let n = order of a(s) and m = order of b(s) [the order of a polynomial is the highest power of s that appears in it].

2.0 Root Locus Design

- Consider all positive values of k.
- In the limit as k -> 0, the poles of the closed-loop system are a(s) = 0 or the poles of H(s).
- In the limit as k -> infinity, the poles of the closed-loop system are b(s) = 0 or the zeros of H(s).
- No matter what we pick k to be, **the closed-loop system must always have n poles**, where n is the number of poles of H(s). **The root locus must have n branches**, each branch starts at a pole of H(s) and goes to a zero of H(s).

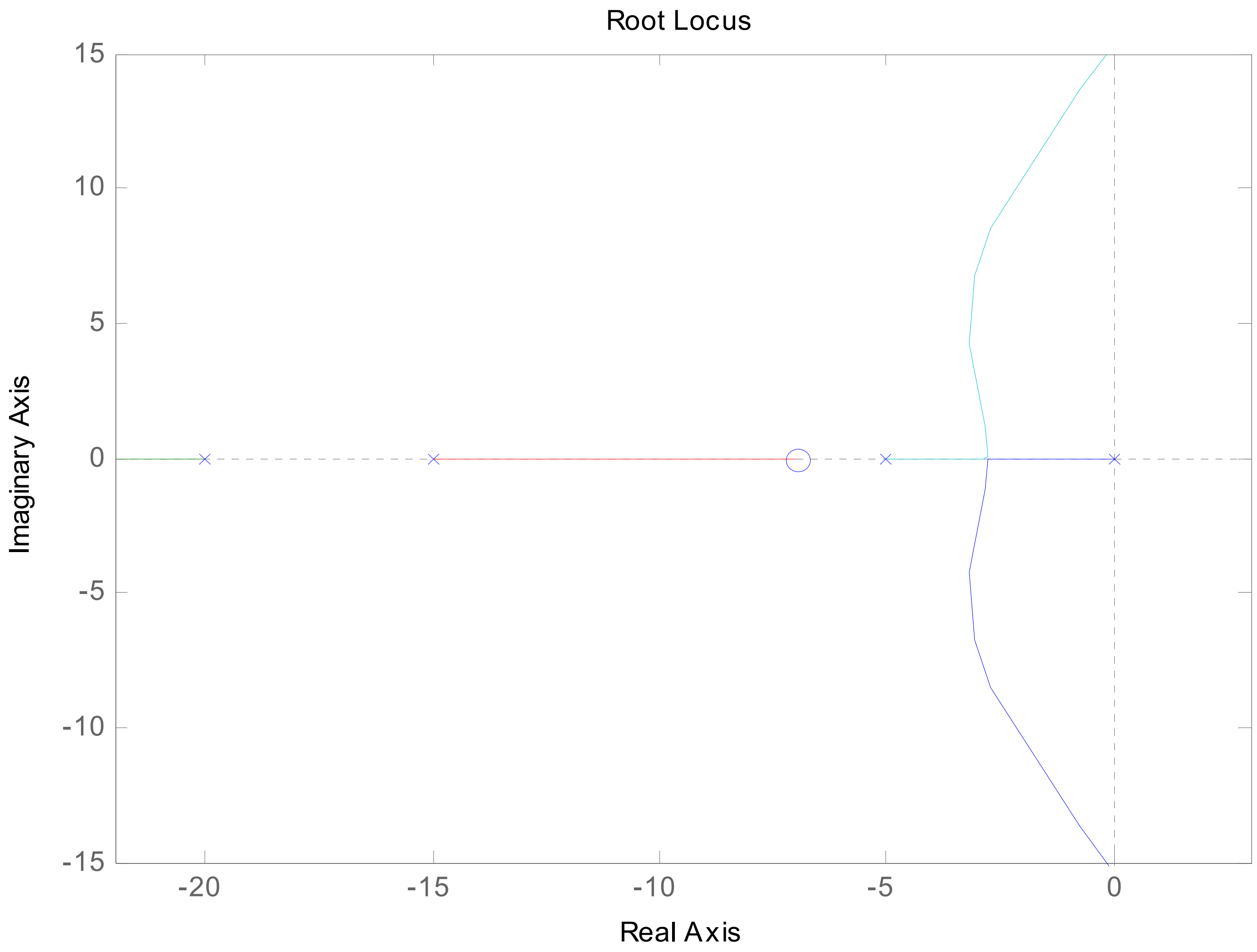
2.0 Root Locus Design

- Consider an open loop system which has a transfer function of

$$H(s) = \frac{Y(s)}{U(s)} = \frac{s-7}{s(s+5)(s+15)(s+20)}$$

```
num=[1 7];
den=conv(conv([1 0],[1 5]),conv([1 15],[1 20]));
rlocus(num,den)
axis([-22 3 -15 15])
```

2.0 Root Locus Design

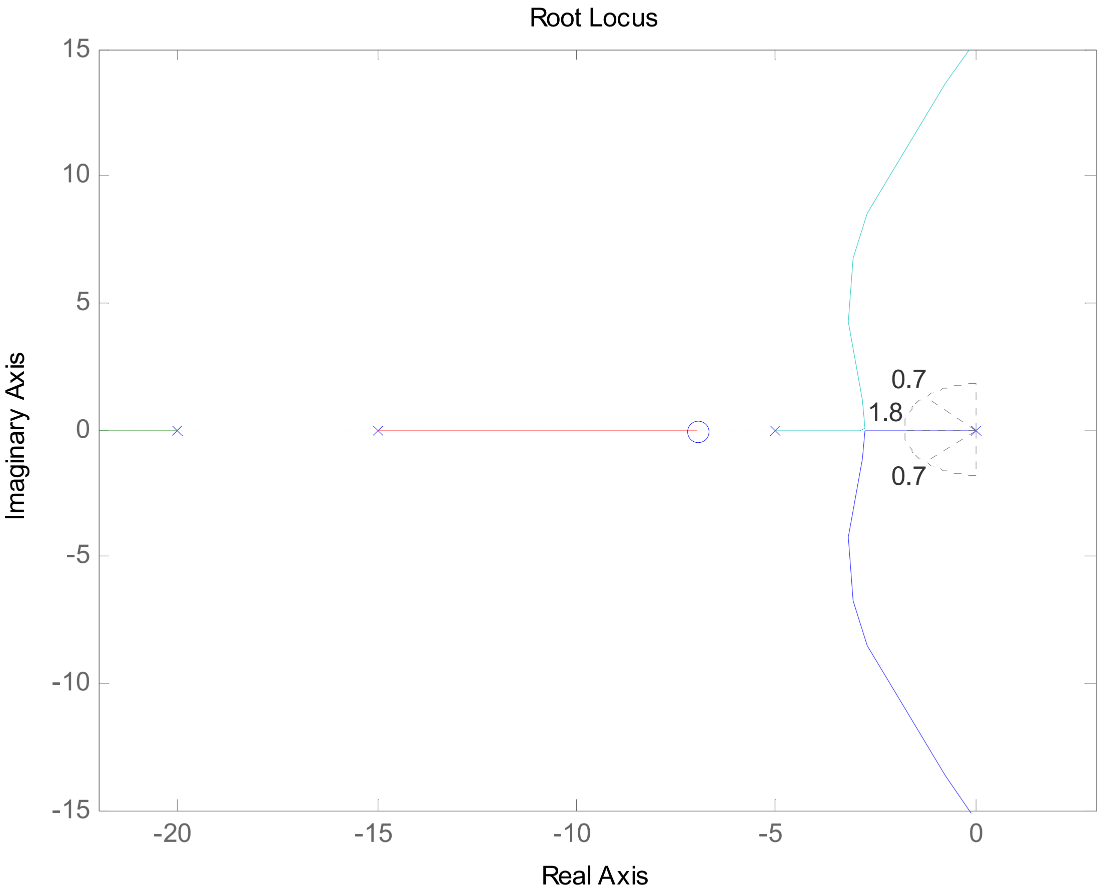


2.0 K from the root locus

- The plot above shows all possible closed-loop pole locations for a pure proportional controller.
- So use the command sgrid(Zeta,Wn) to plot lines of constant damping ratio and natural frequency.
- Its two arguments are the damping ratio (Zeta) and natural frequency (Wn) [these may be vectors if you want to look at a range of acceptable values]. In the problem, an overshoot less than 5% (which means a damping ratio Zeta of greater than 0.7) and a rise time of 1 second (which means a natural frequency Wn greater than 1.8).

2.0 K from the root locus

- zeta=0.7; Wn=1.8; sgrid(zeta, Wn)

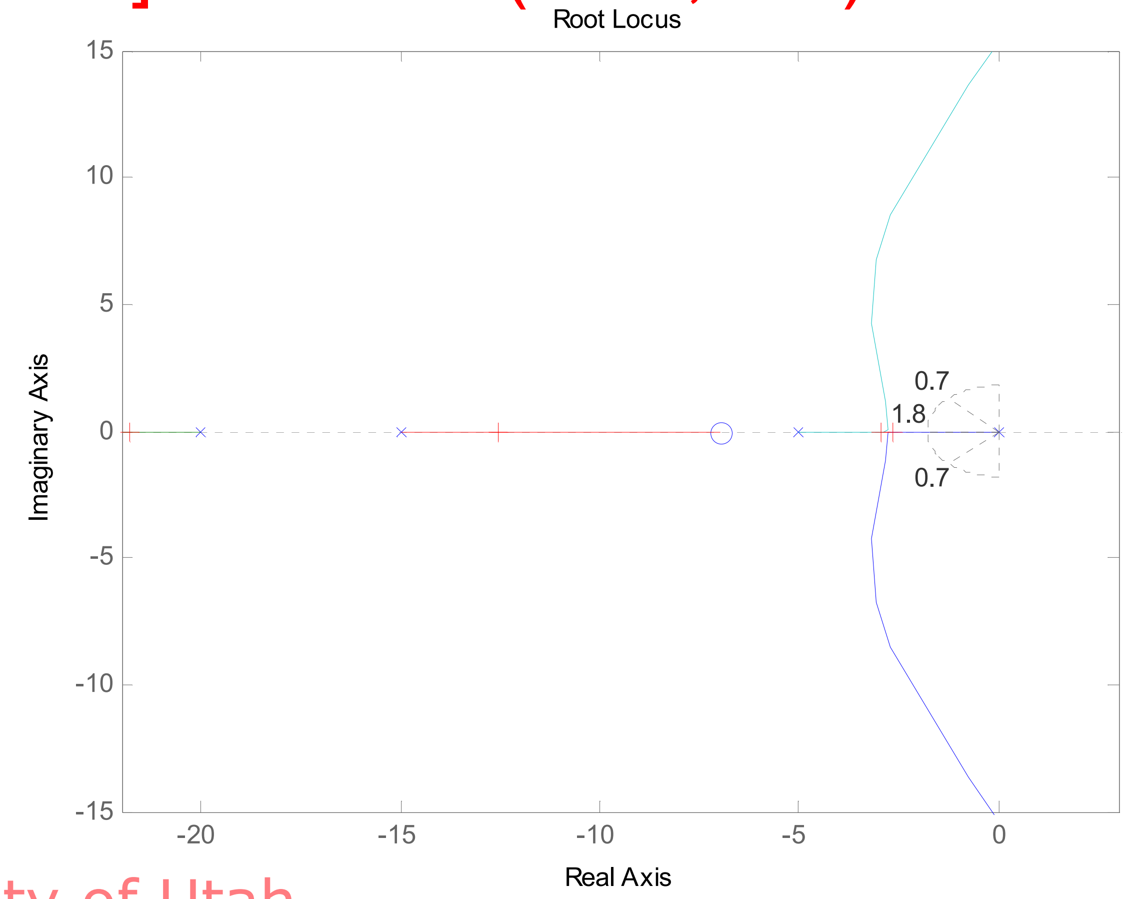


2.0 K from the root locus

- The plot above shows all possible closed-loop pole locations for a pure proportional controller.
- So use the command sgrid(Zeta,Wn) to plot lines of constant damping ratio and natural frequency.
- Its two arguments are the damping ratio (Zeta) and natural frequency (Wn) [these may be vectors if you want to look at a range of acceptable values]. In the problem, an overshoot less than 5% (which means a damping ratio Zeta of greater than 0.7) and a rise time of 1 second (which means a natural frequency Wn greater than 1.8).

2.0 K from the root locus

- [kd,poles] = rlocfind(num,den)



2.0 K from the root locus

- [kd,poles] = rlocfind(num,den)
- Select a point in the graphics window
- selected_point = -2.6576 + 0.0466i
- kd = 306.9457
- poles =

1) -21.8180

2) -12.5794

3) -2.9373

4) -2.6652

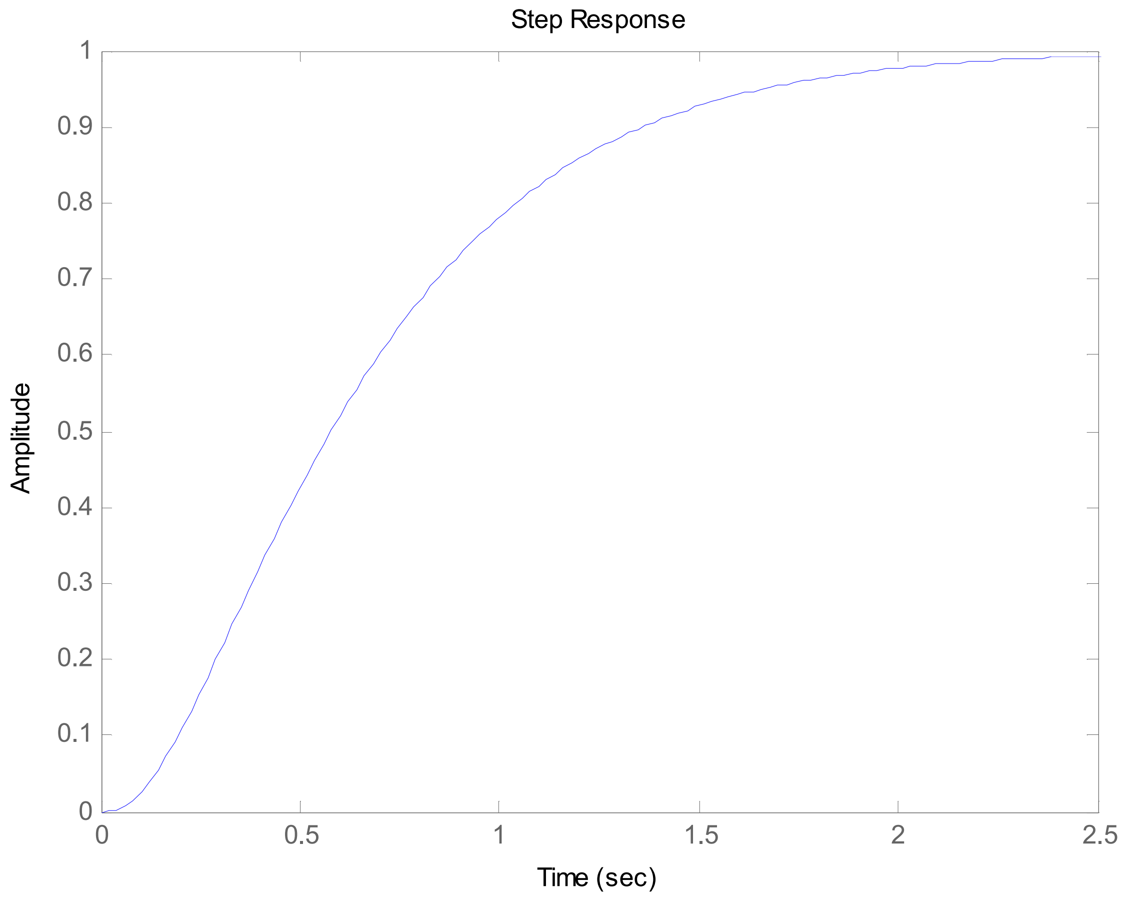
2.0 Closed-loop response

- In order to find the step response, one needs to know the closed-loop transfer function. [numCL, denCL] = cloop((kd)*num, den)
- The two arguments to the function cloop are the numerator and denominator of the open-loop system. You need to include the proportional gain that you have chosen. Unity feedback is assumed.
- If you have a non-unity feedback situation, look at the help file for the Matlab function feedback, which can find the closed-loop transfer function with a gain in the feedback loop.

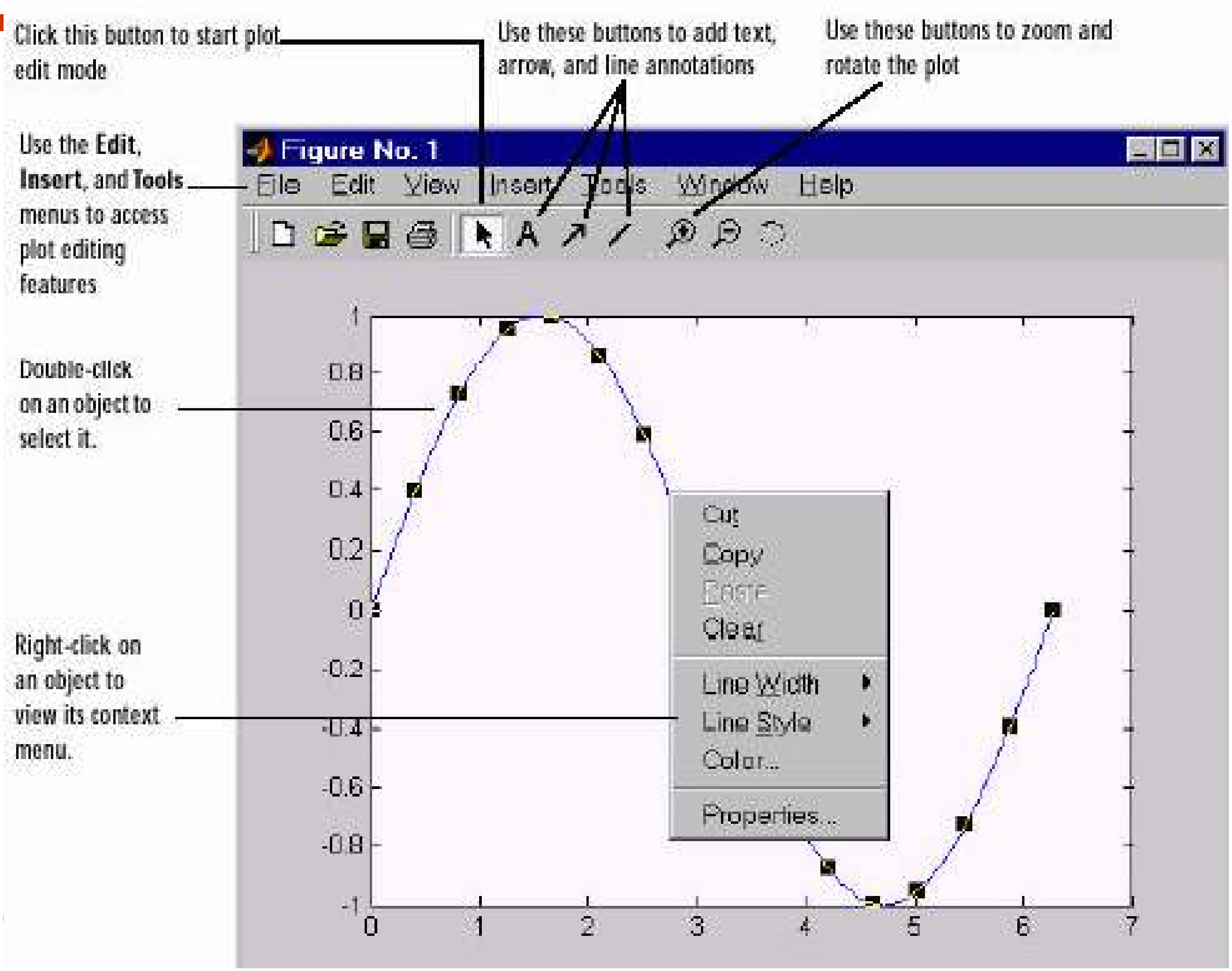
2.0 Closed-loop response

- [numCL, denCL] = cloop((kd)*num, den)
- numCL = 1.0e+003 * 0 0 0 0.3069 2.1486
- denCL = 1.0e+003 * 0.0010 0.0400 0.4750 1.8069 2.1486
- >> step(numCL,denCL)

2.0 Closed-loop response



2.0 Root Locus Design



Click this button to start plot edit mode

Use these buttons to add text, arrow, and line annotations

Use these buttons to zoom and rotate the plot

Use the **Edit**, **Insert**, and **Tools** menus to access plot editing features

Double-click on an object to select it.

Right-click on an object to view its context menu.

Th

2.0 Root Locus Design

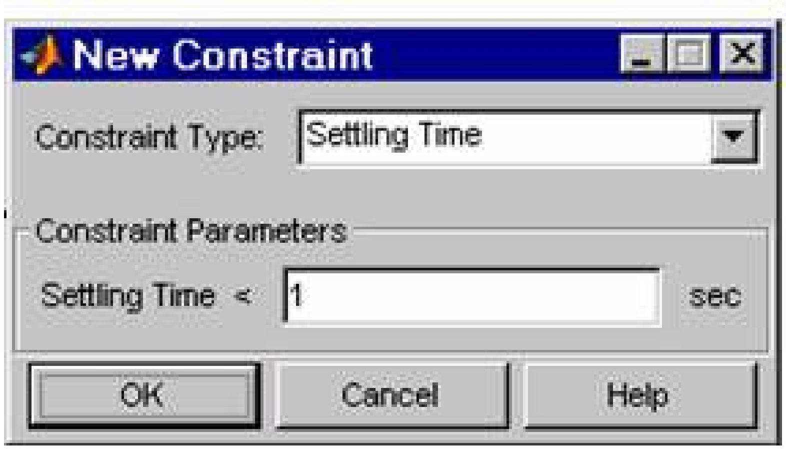
- Reference: http://www.engin.umich.edu/group/ctm/rlocus/rlocus.html
- http://www.cs.wright.edu/people/faculty/krattan/courses/414/lab1/matlabcommands.pdf

3.0 SICO Root Locus Design

- **Design Constraints**
- When designing compensators, it is common to have design specifications that call for specific settling times, damping ratios, and other characteristics.
- The SISO Design Tool provides design constraints that can help make the task of meeting design specifications easier. The **New Constraint** window, which allows you to create design constraints, automatically changes to reflect which constraints are available for the view in which you are working.

3.0 SICO Root Locus Design

- **Design Constraints**
- Select **Design Constraints** and then **New** to open the **New Constraint** window, which is shown below.



Constraint Type: Settling Time

Constraint Parameters

Settling Time < 1 sec

3.0 SICO Root Locus Design

- **Design Constraints for the Root Locus**

- For the root locus, you have the following constraint types:
- Settling Time
- Percent Overshoot
- Damping Ratio
- Natural Frequency

3.0 SICO Root Locus Design

- **Settling Time.** If you specify a settling time in the continuous-time root locus, a vertical line appears on the root locus plot at the pole locations associated with the value provided (using a first-order approximation). In the discrete-time case, the constraint is a curved line.
- **Percent Overshoot.** Specifying percent overshoot in the continuous-time root locus causes two rays, starting at the root locus origin, to appear. These rays are the locus of poles associated with the percent value (using a second-order approximation). In the discrete-time case, In the discrete-time case, the constraint appears as two curves originating at (1,0) and meeting on the real axis in the left-hand plane.

3.0 SICO Root Locus Design

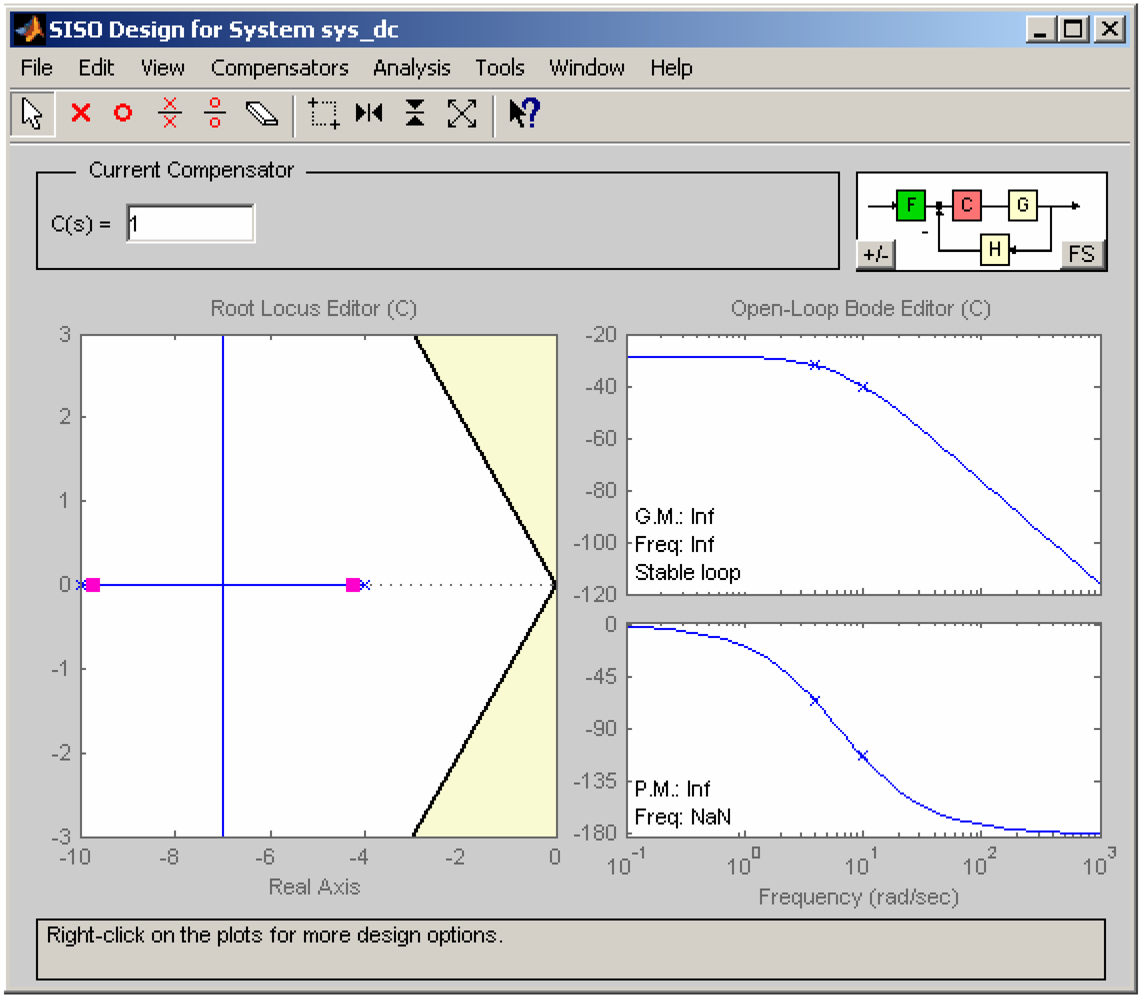
- **Damping Ratio.** Specifying a damping ratio in the continuous-time root locus causes two rays, starting at the root locus origin, to appear. These rays are the locus of poles associated with the damping ratio. In the discrete-time case, the constraint appears as curved lines originating at (1,0) and meeting on the real axis in the left-hand plane.

- **Natural Frequency.** If you specify a natural frequency, a semicircle centered around the root locus origin appears. The radius equals the natural frequency.

3.0 SICO Root Locus Design

- load ltiexamples

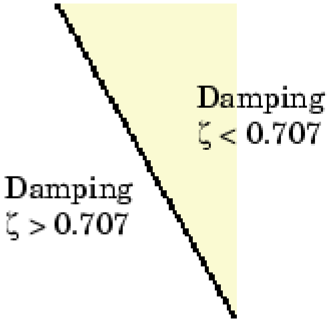
  sisotool(sys_dc)



Right-click on the plots for more design options.

3.0 SICO Root Locus Design

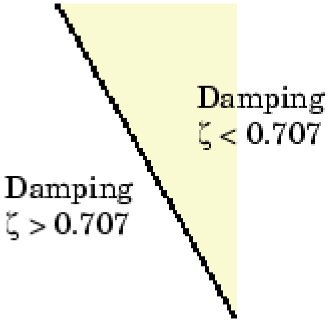
- The two rays centered at (0,0) represent the damping ratio constraint. The dark edge is the region boundary, and the shaded area outlines the exclusion region. This figure explains what this means for this constraint.



- use this design constraint to ensure that the closed-loop poles, represented by the red squares, have some minimum damping. Try adjusting the gain until the damping ratio of the closed-loop poles is 0.7.

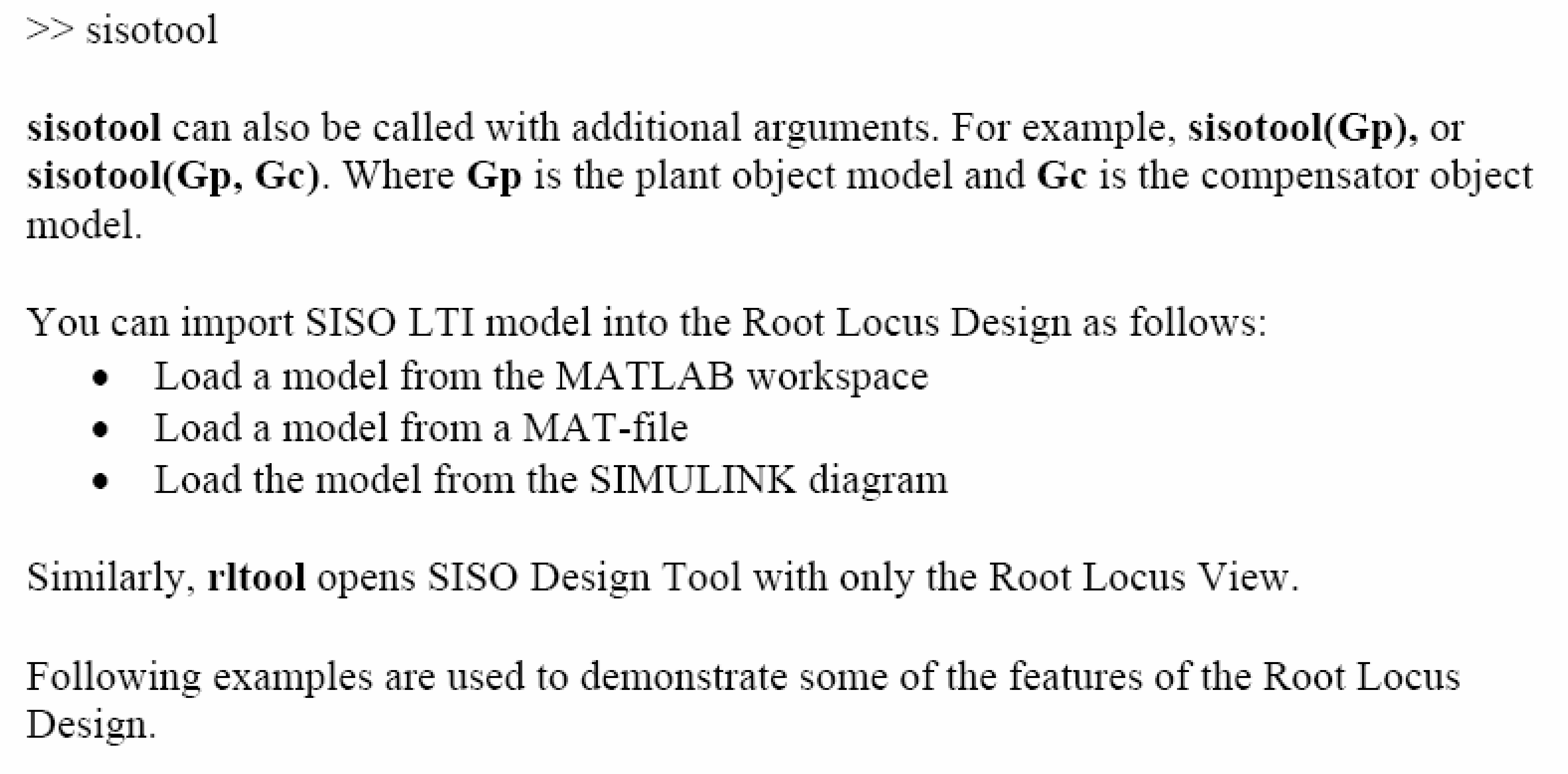
3.0 SICO Root Locus Design

- The two rays centered at (0,0) represent the damping ratio constraint. The dark edge is the region boundary, and the shaded area outlines the exclusion region. This figure explains what this means for this constraint.



- use this design constraint to ensure that the closed-loop poles, represented by the red squares, have some minimum damping. Try adjusting the gain until the damping ratio of the closed-loop poles is 0.7.

3.0 SICO Root Locus Design



```
>> sisotool
```

sisotool can also be called with additional arguments. For example, **sisotool(Gp),** or **sisotool(Gp, Gc)**. Where **Gp** is the plant object model and **Gc** is the compensator object model.

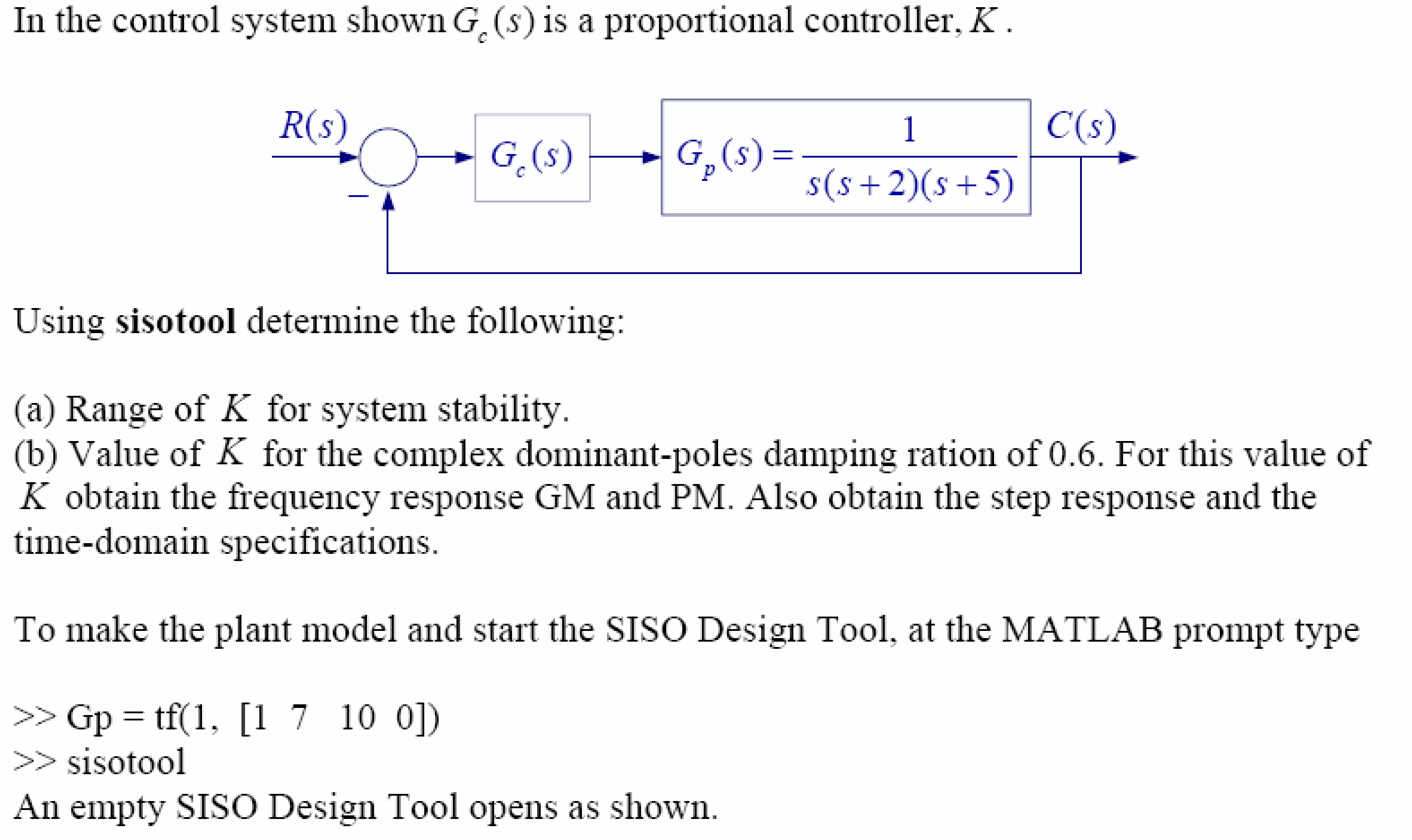
You can import SISO LTI model into the Root Locus Design as follows:

- Load a model from the MATLAB workspace
- Load a model from a MAT-file
- Load the model from the SIMULINK diagram

Similarly, **rltool** opens SISO Design Tool with only the Root Locus View.

Following examples are used to demonstrate some of the features of the Root Locus Design.

3.0 SICO Root Locus Design



In the control system shown $G_c(s)$ is a proportional controller, K.

$R(s)$
$-$
$G_c(s)$
$G_p(s) = \frac{1}{s(s+2)(s+5)}$
$C(s)$

Using **sisotool** determine the following:

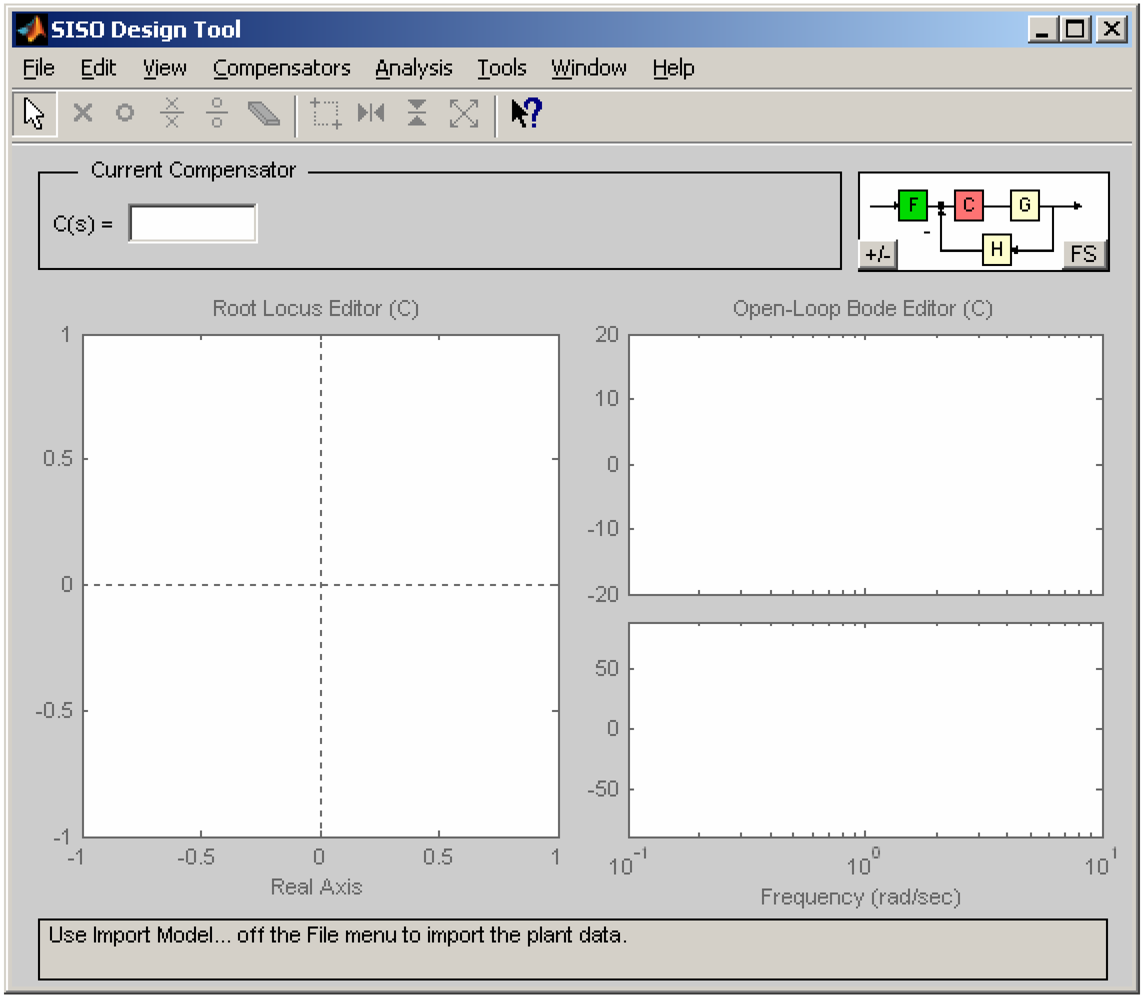
(a) Range of K for system stability.
(b) Value of K for the complex dominant-poles damping ration of 0.6. For this value of K obtain the frequency response GM and PM. Also obtain the step response and the time-domain specifications.

To make the plant model and start the SISO Design Tool, at the MATLAB prompt type

```
>> Gp = tf(1,  [1  7   10  0])
>> sisotool
```

An empty SISO Design Tool opens as shown.

3.0 SICO Root Locus Design

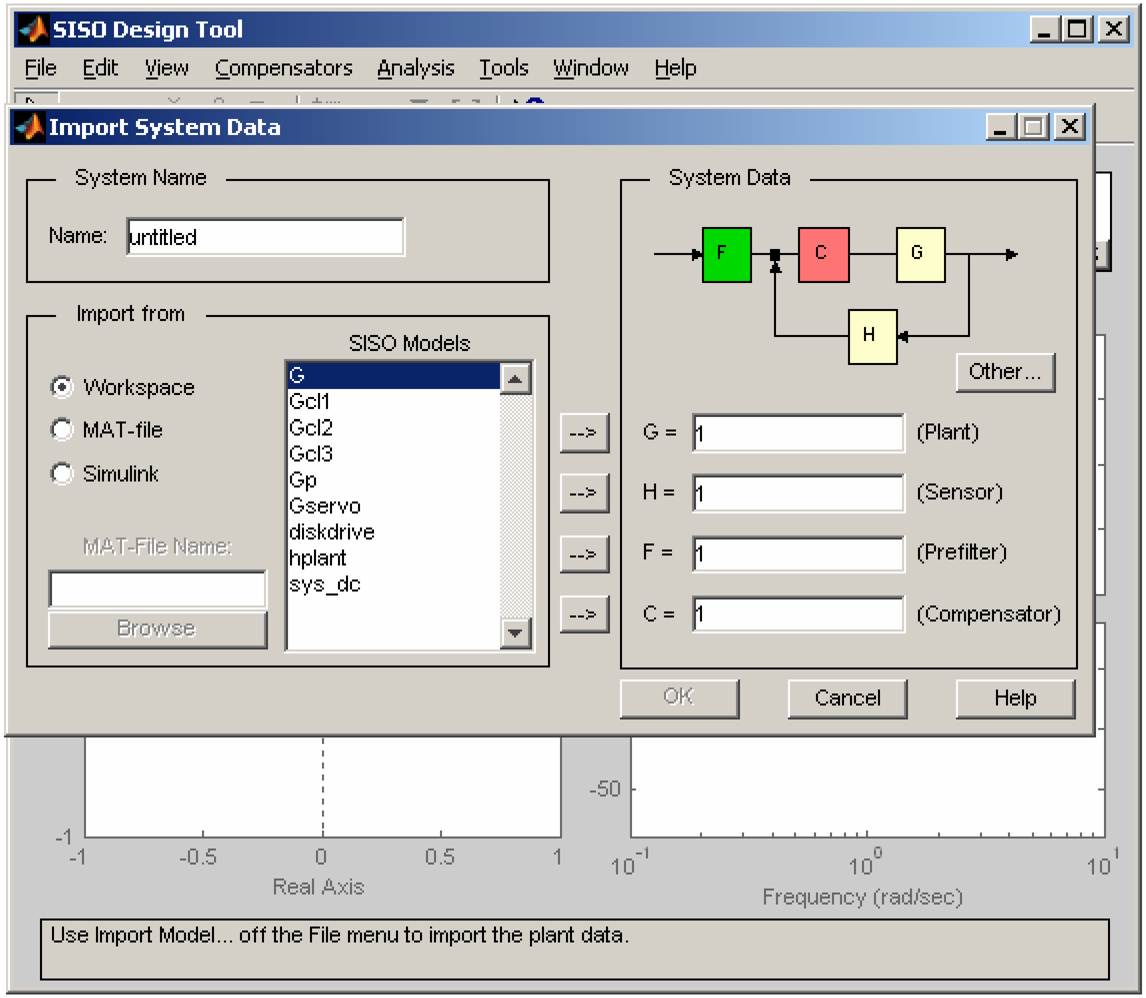


C(s) =

Use Import Model... off the File menu to import the plant data.

The University of Utah

3.0 SICO Root Locus Design



Name: untitled

● Workspace

G = 1 (Plant)

H = 1 (Sensor)

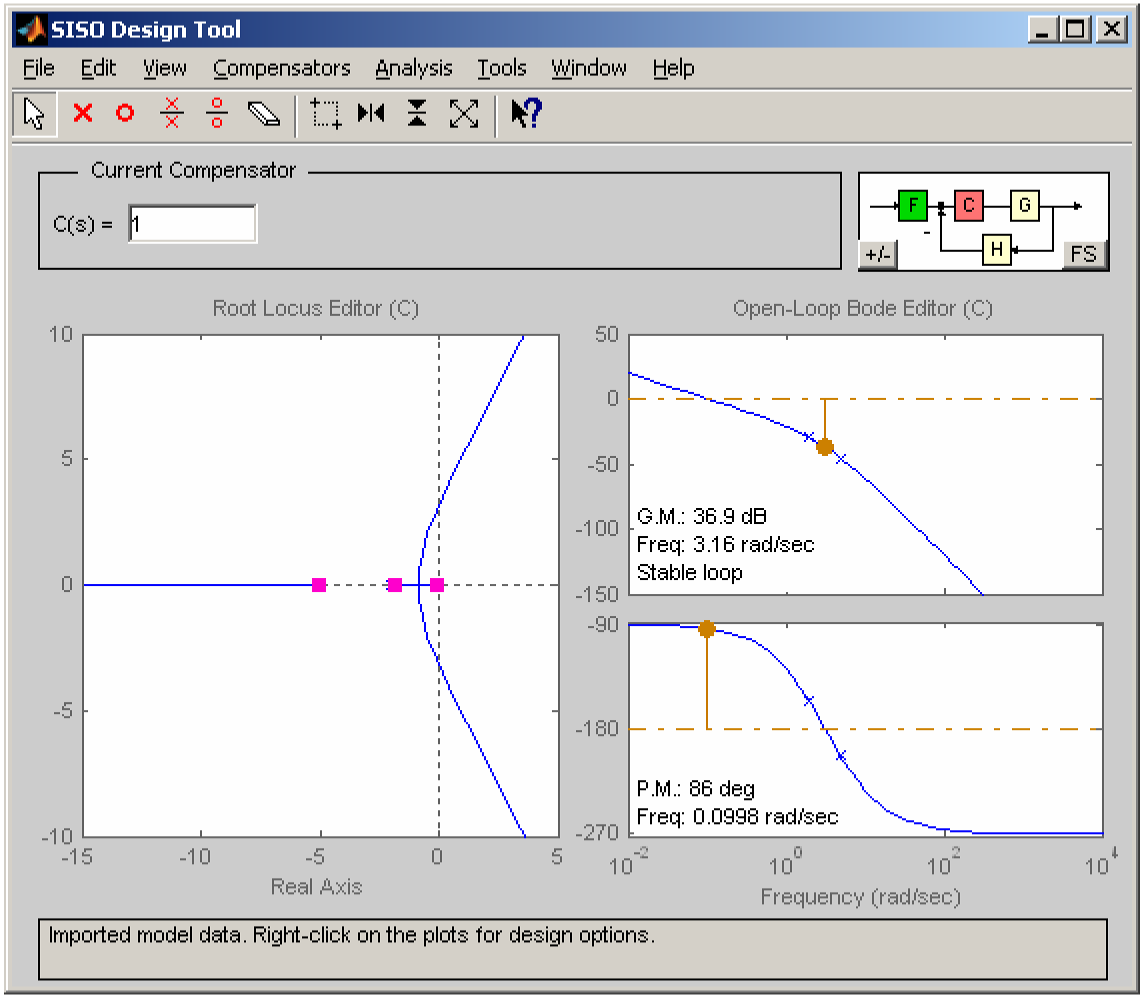
F = 1 (Prefilter)

C = 1 (Compensator)

Use Import Model... off the File menu to import the plant data.

The University of Utah

3.0 SICO Root Locus Design



Current Compensator

C(s) = 1

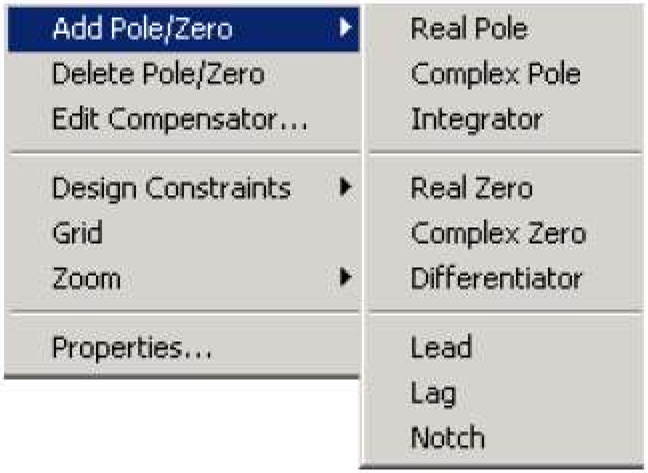
G.M.: 36.9 dB
Freq: 3.16 rad/sec
Stable loop

P.M.: 86 deg
Freq: 0.0998 rad/sec

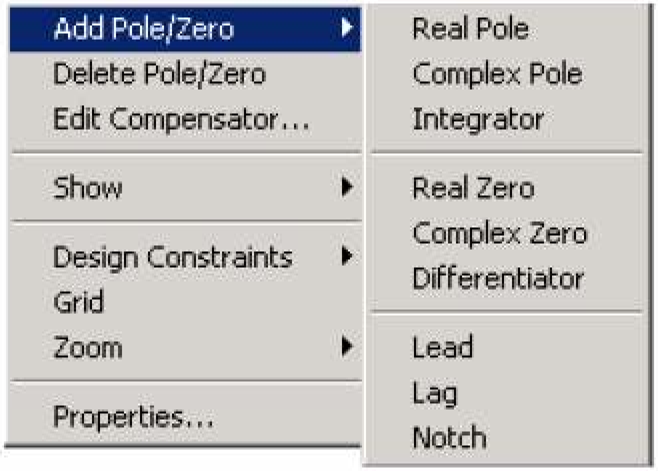
Imported model data. Right-click on the plots for design options.

3.0 SICO Root Locus Design

The SISO Design Tool has right-click menus available in the plot regions with many features.

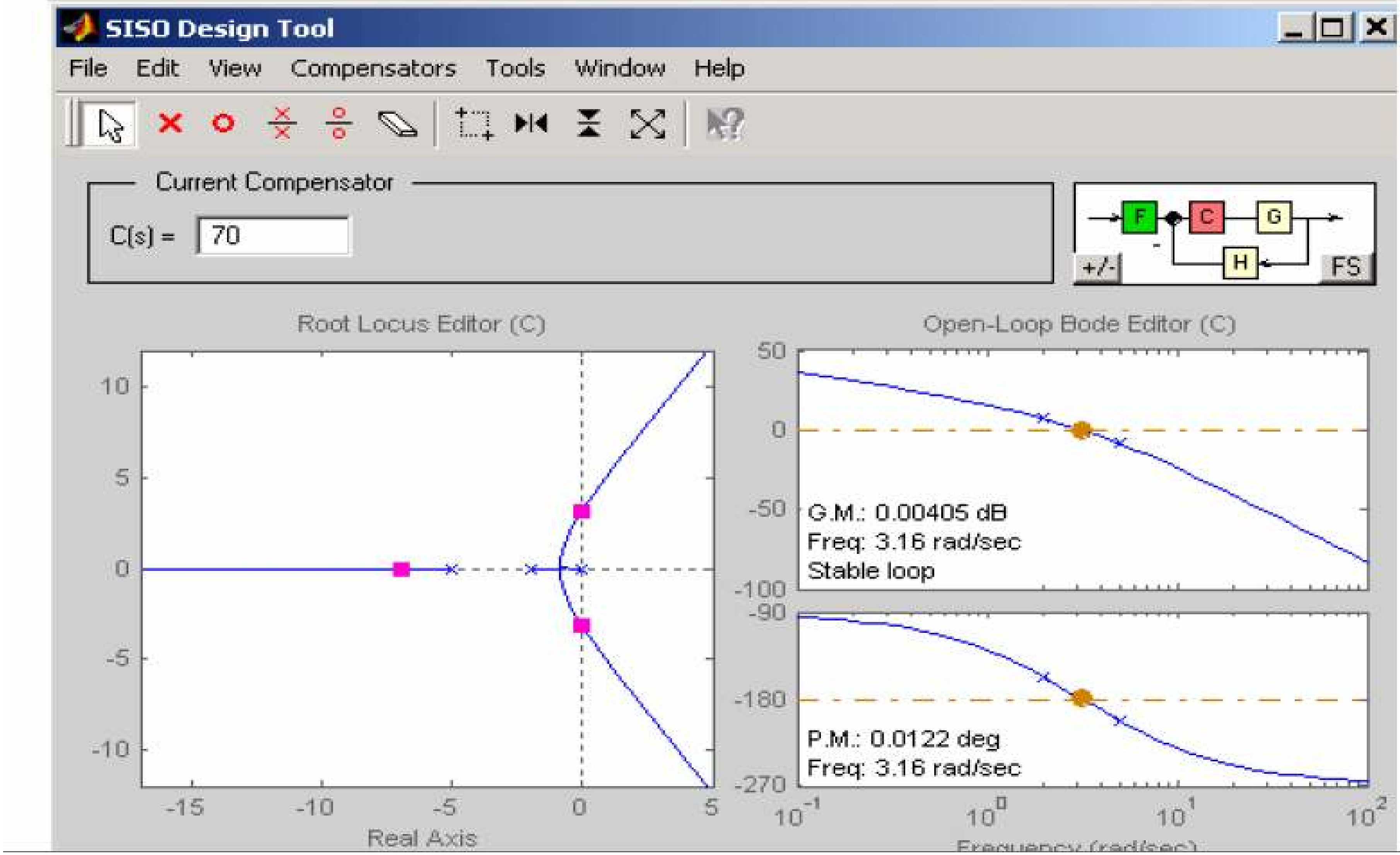


Right-Click Menu on the Root Locus plot



Right-click Menu on the Bode plot

(a) To find the value of K for marginal stability drag the closed-loop pole along the locus to the $j\omega$-axis intersect. Turn on the Mouse Zoom and drag the mouse to draw a box around the pole. When the mouse is released, root locus will zoom in on the selected region. Drag the pole to this accurate position. The gain for marginal stability is found to be $K_c = 70$. A zoom out result is as shown.

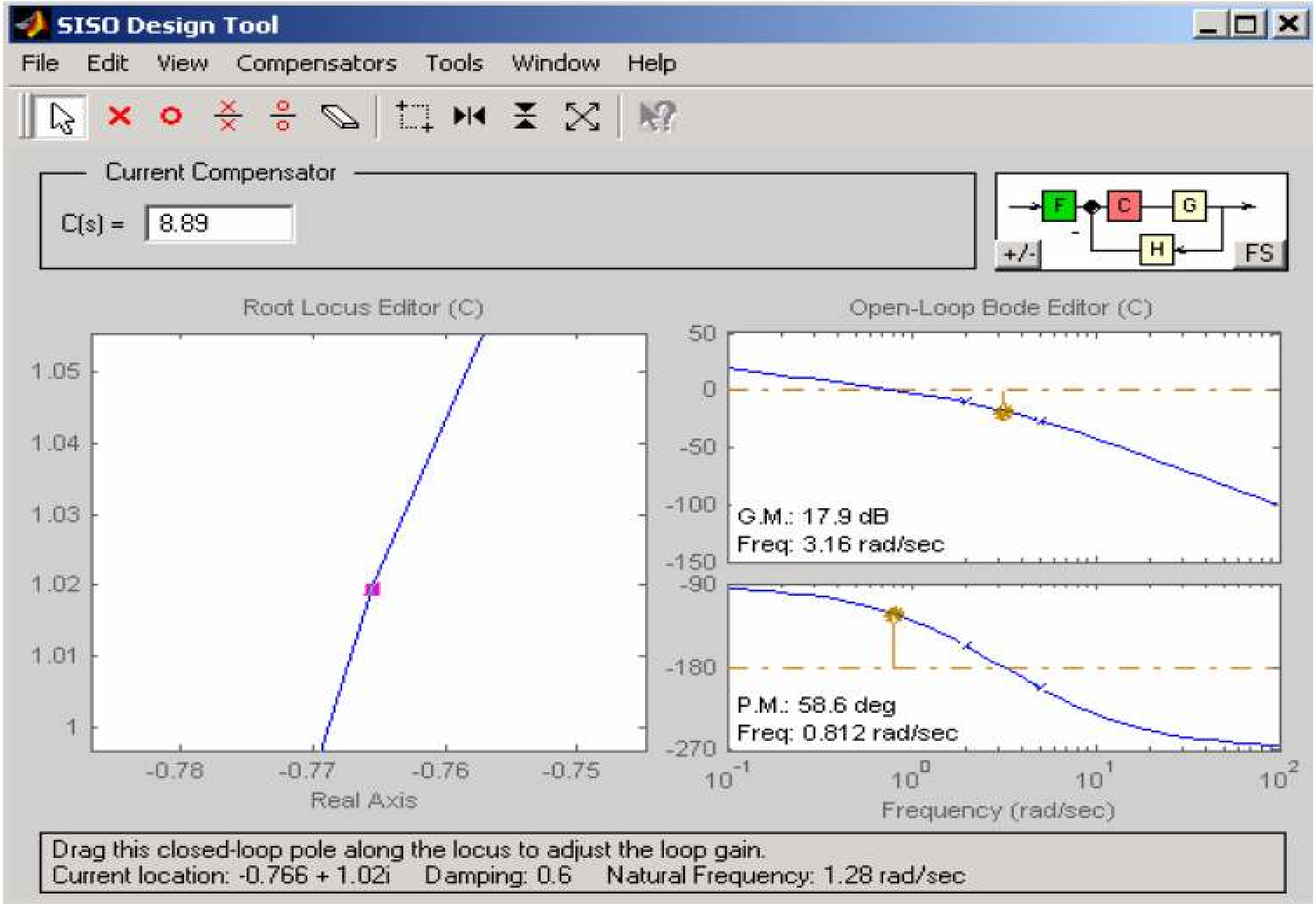


C(s) = 70

G.M.: 0.00405 dB
Freq: 3.16 rad/sec
Stable loop

P.M.: 0.0122 deg
Freq: 3.16 rad/sec

(b) To find the gain for damping ratio of 0.6, drag the complex pole until the damping ratio displayed in the bottom panel is in the neighborhood of 0.6. Then zoom in and drag the pole until damping ratio is 0.6. The result is as shown.



Current Compensator

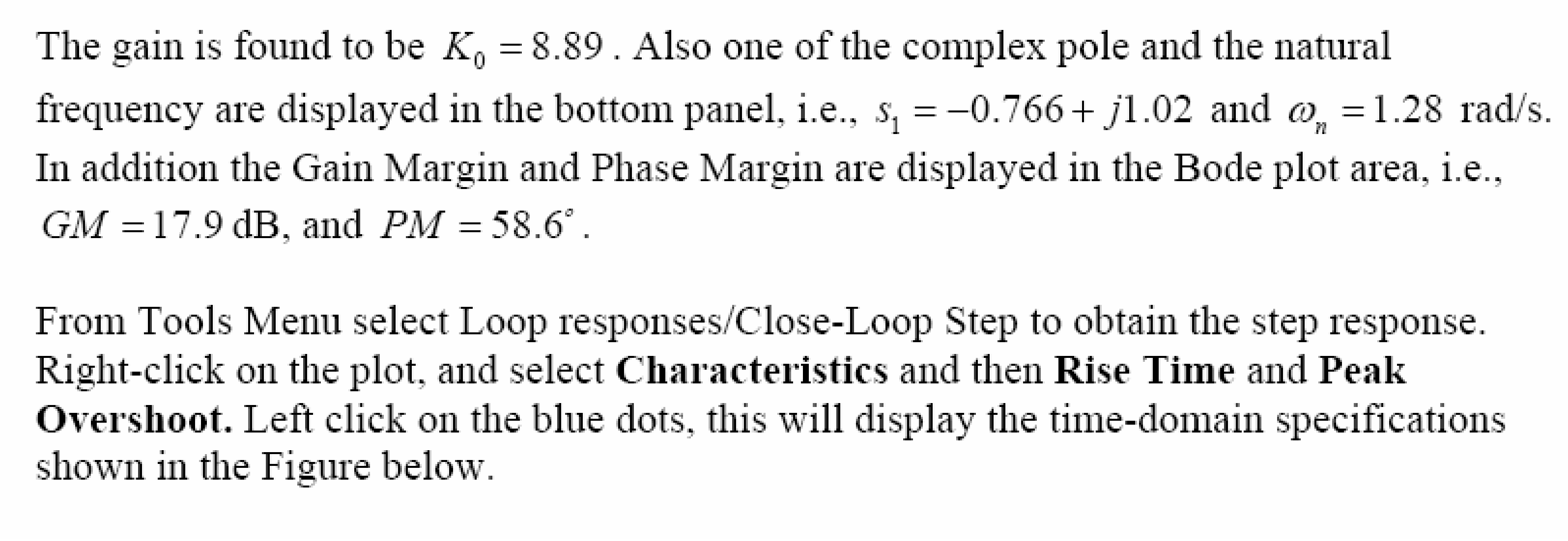
C(s) = 8.89

G.M.: 17.9 dB
Freq: 3.16 rad/sec

P.M.: 58.6 deg
Freq: 0.812 rad/sec

Drag this closed-loop pole along the locus to adjust the loop gain.
Current location: -0.766 + 1.02i   Damping: 0.6   Natural Frequency: 1.28 rad/sec

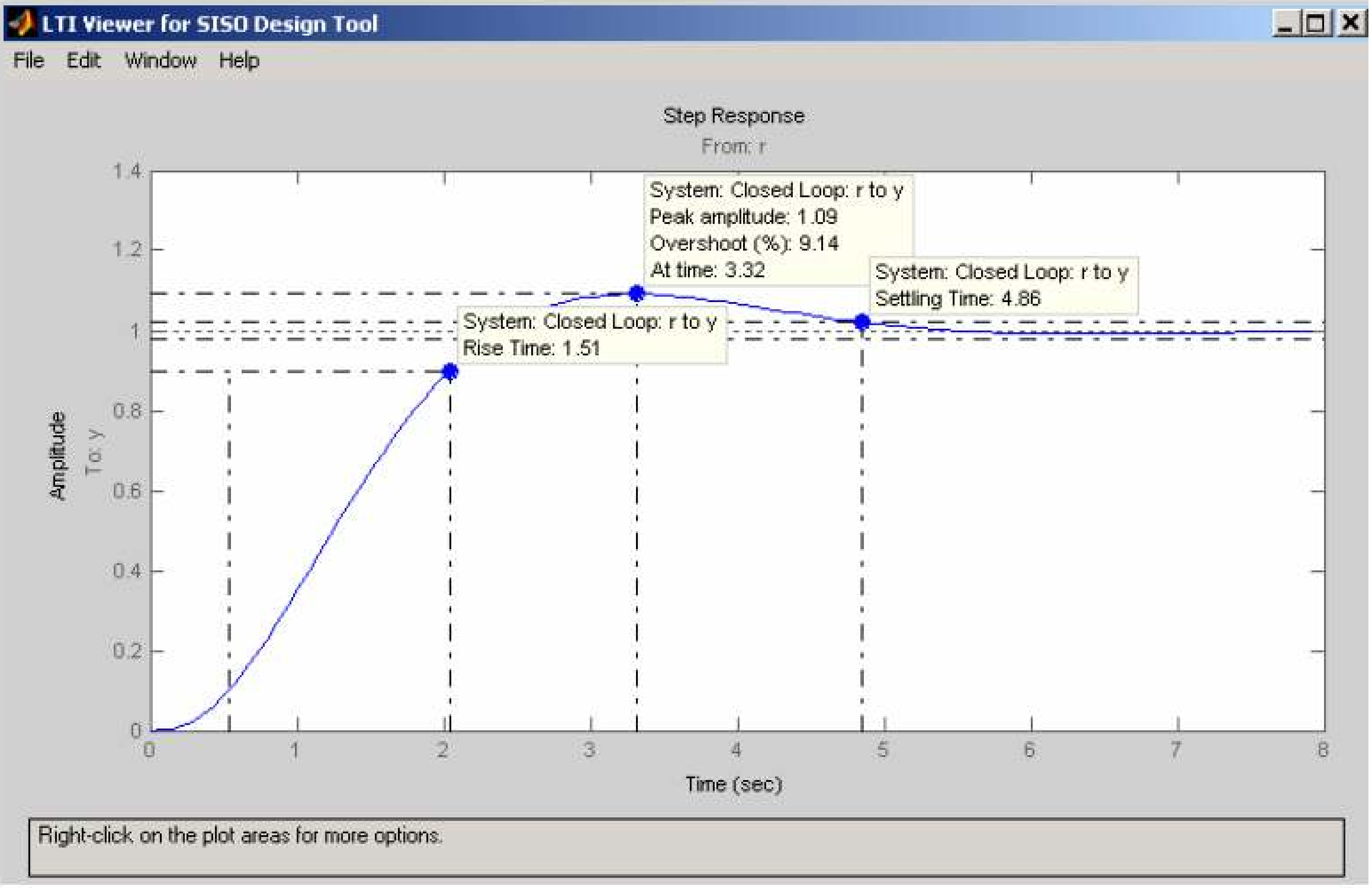
3.0 SICO Root Locus Design



The gain is found to be $K_0 = 8.89$. Also one of the complex pole and the natural frequency are displayed in the bottom panel, i.e., $s_1 = -0.766 + j1.02$ and $\omega_n = 1.28$ rad/s. In addition the Gain Margin and Phase Margin are displayed in the Bode plot area, i.e., $GM = 17.9$ dB, and $PM = 58.6^\circ$.

From Tools Menu select Loop responses/Close-Loop Step to obtain the step response. Right-click on the plot, and select **Characteristics** and then **Rise Time** and **Peak Overshoot.** Left click on the blue dots, this will display the time-domain specifications shown in the Figure below.

3.0 SICO Root Locus Design



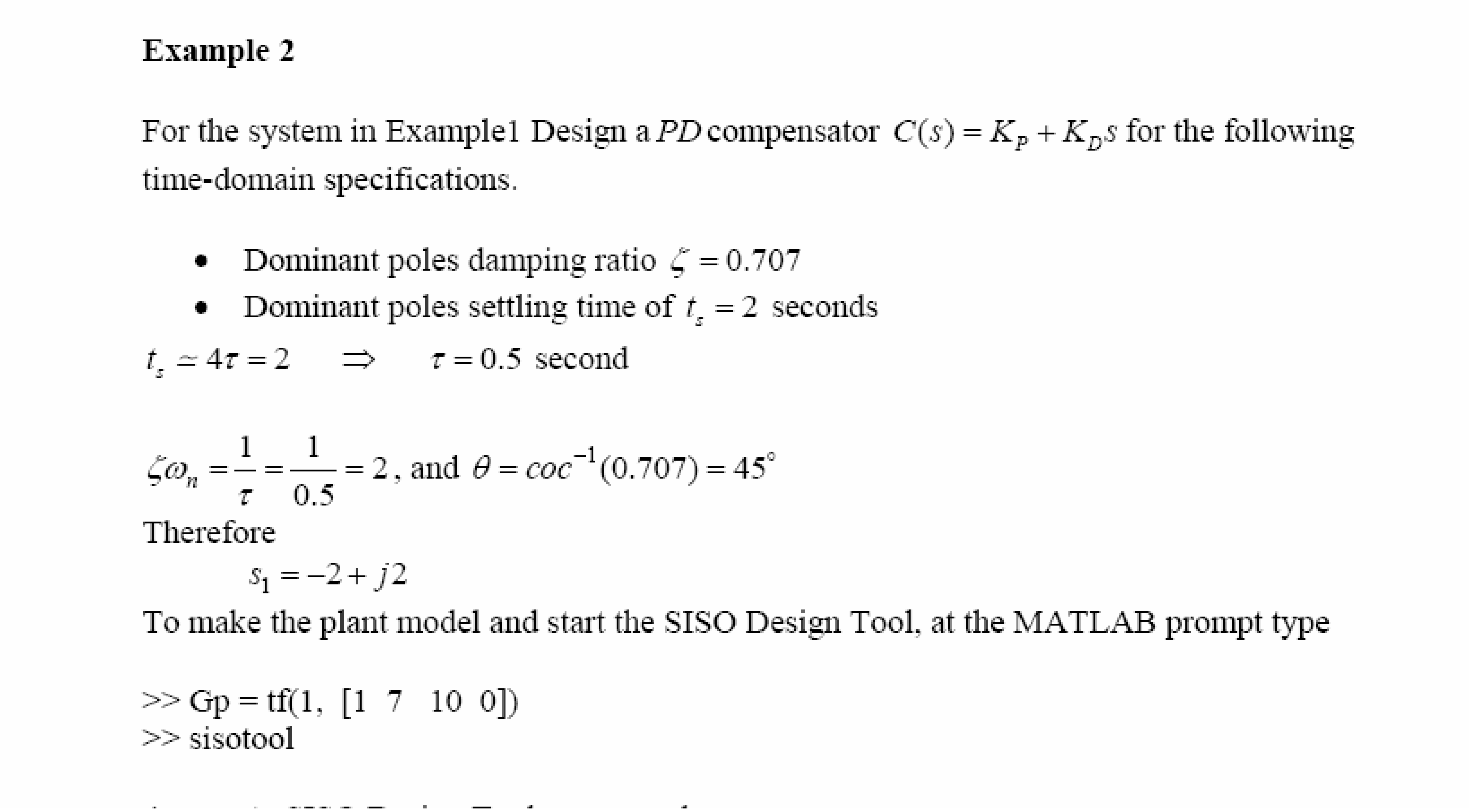
System: Closed Loop: r to y
Peak amplitude: 1.09
Overshoot (%): 9.14
At time: 3.32

System: Closed Loop: r to y
Settling Time: 4.86

System: Closed Loop: r to y
Rise Time: 1.51

Right-click on the plot areas for more options.

3.0 SICO Root Locus Design



Example 2

For the system in Example1 Design a *PD* compensator $C(s) = K_P + K_D s$ for the following time-domain specifications.

- Dominant poles damping ratio $\zeta = 0.707$
- Dominant poles settling time of $t_s = 2$ seconds

$t_s \simeq 4\tau = 2 \quad \Rightarrow \quad \tau = 0.5$ second

$\zeta\omega_n = \dfrac{1}{\tau} = \dfrac{1}{0.5} = 2$, and $\theta = coc^{-1}(0.707) = 45^\circ$

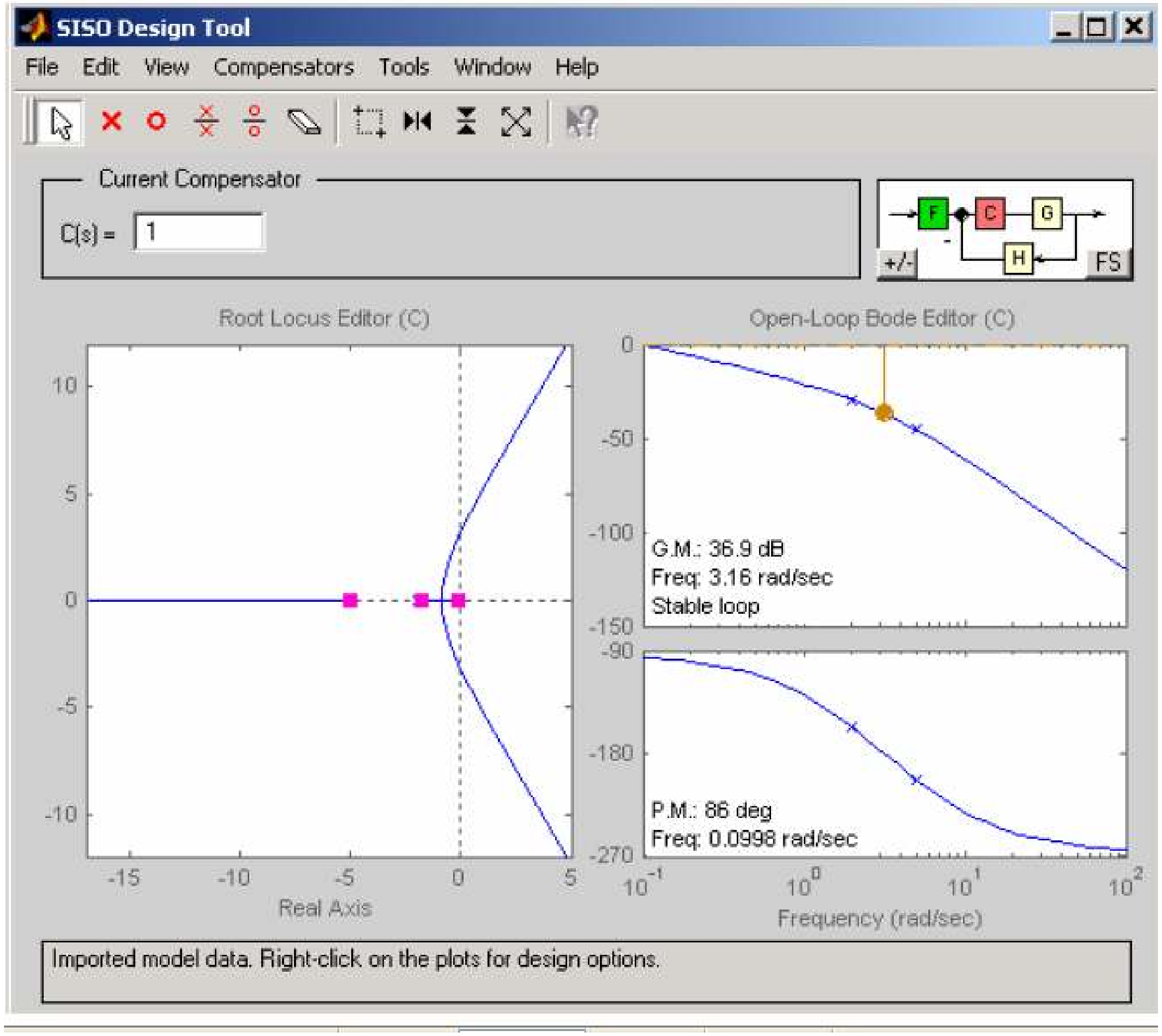
Therefore

$$s_1 = -2 + j2$$

To make the plant model and start the SISO Design Tool, at the MATLAB prompt type

```
>> Gp = tf(1,  [1  7   10  0])
>> sisotool
```

3.0 SICO Root Locus Design



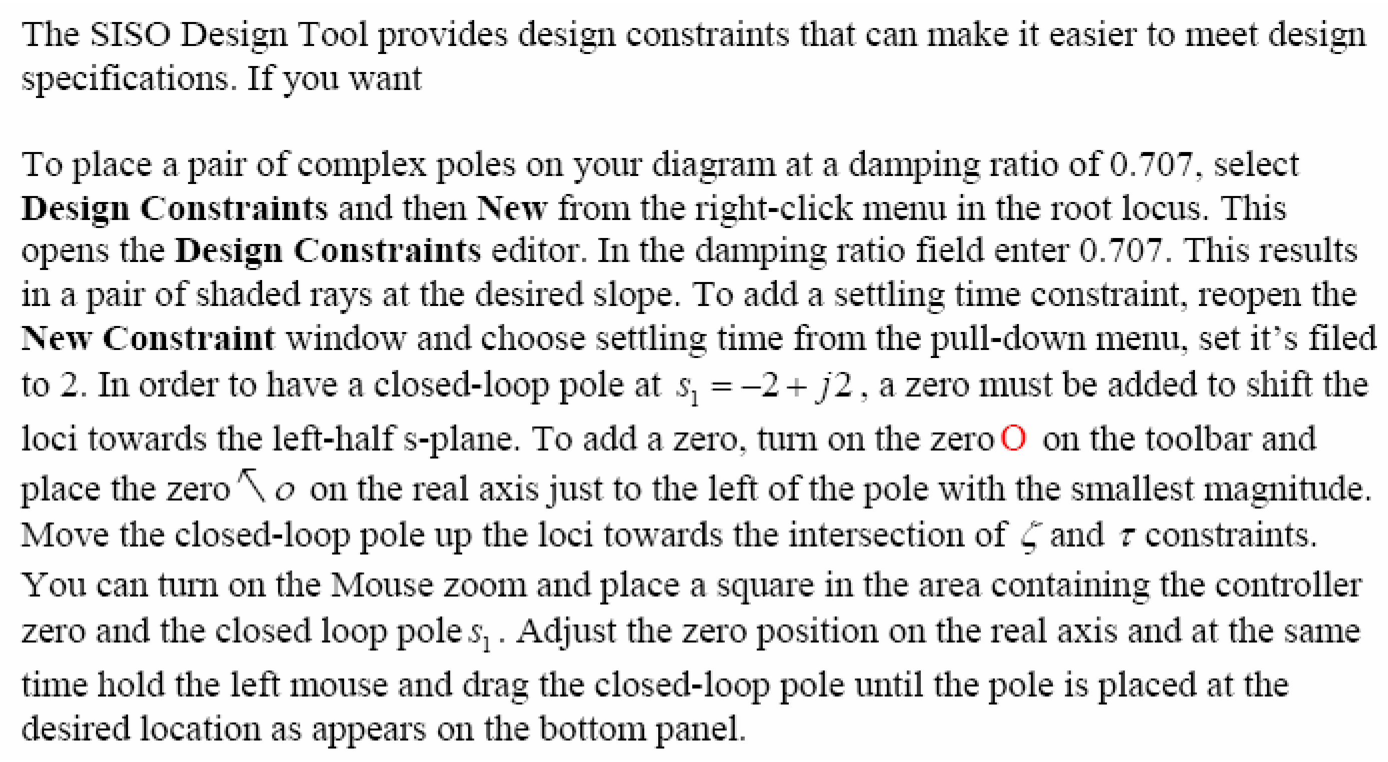
C(s) = 1

G.M.: 36.9 dB
Freq: 3.16 rad/sec
Stable loop

P.M.: 86 deg
Freq: 0.0998 rad/sec

Imported model data. Right-click on the plots for design options.

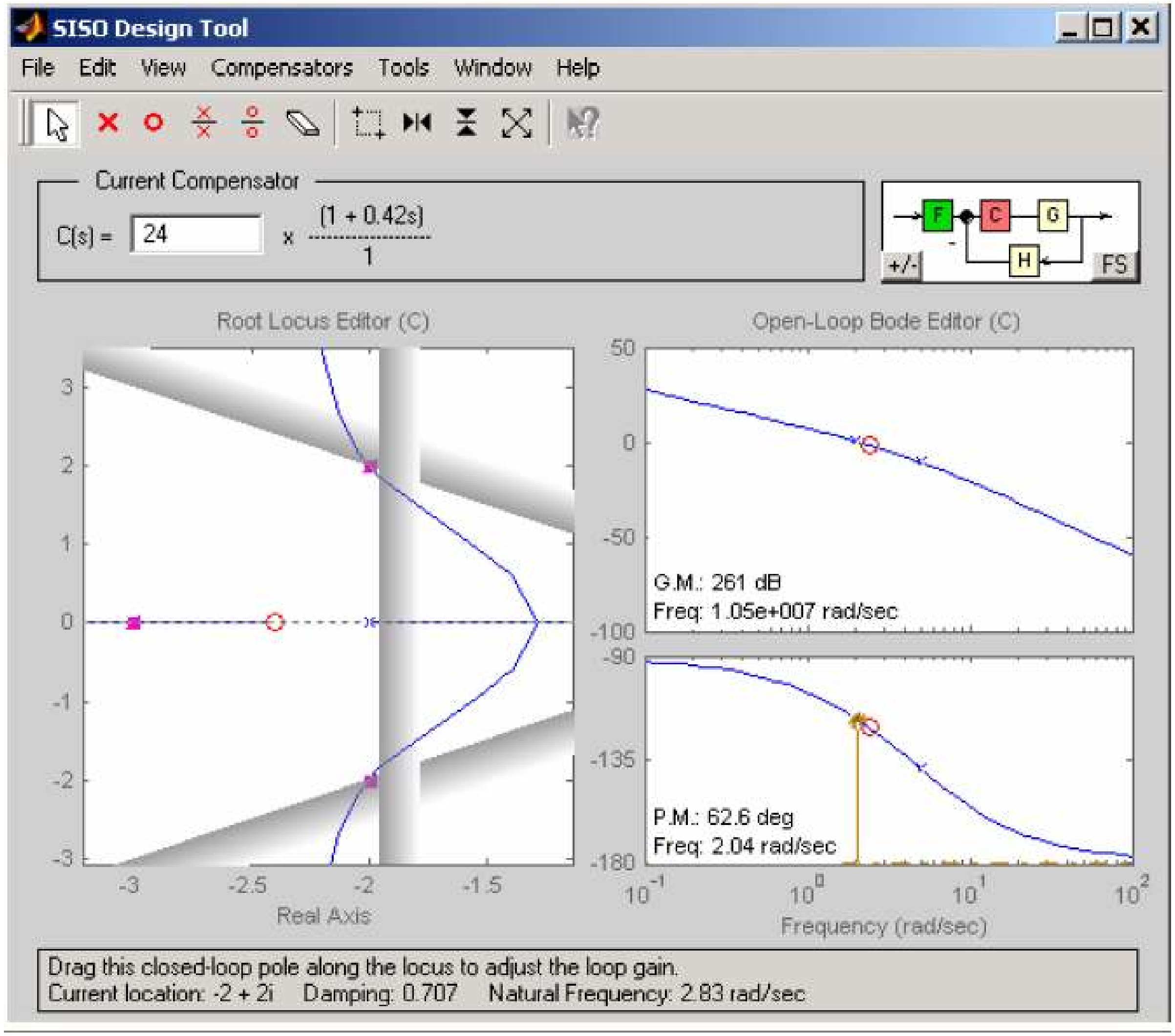
3.0 SICO Root Locus Design



The SISO Design Tool provides design constraints that can make it easier to meet design specifications. If you want

To place a pair of complex poles on your diagram at a damping ratio of 0.707, select **Design Constraints** and then **New** from the right-click menu in the root locus. This opens the **Design Constraints** editor. In the damping ratio field enter 0.707. This results in a pair of shaded rays at the desired slope. To add a settling time constraint, reopen the **New Constraint** window and choose settling time from the pull-down menu, set it's filed to 2. In order to have a closed-loop pole at $s_1 = -2 + j2$, a zero must be added to shift the loci towards the left-half s-plane. To add a zero, turn on the zero ○ on the toolbar and place the zero ↖ o on the real axis just to the left of the pole with the smallest magnitude. Move the closed-loop pole up the loci towards the intersection of ζ and τ constraints. You can turn on the Mouse zoom and place a square in the area containing the controller zero and the closed loop pole s_1. Adjust the zero position on the real axis and at the same time hold the left mouse and drag the closed-loop pole until the pole is placed at the desired location as appears on the bottom panel.

3.0 SICO Root Locus Design



Current Compensator

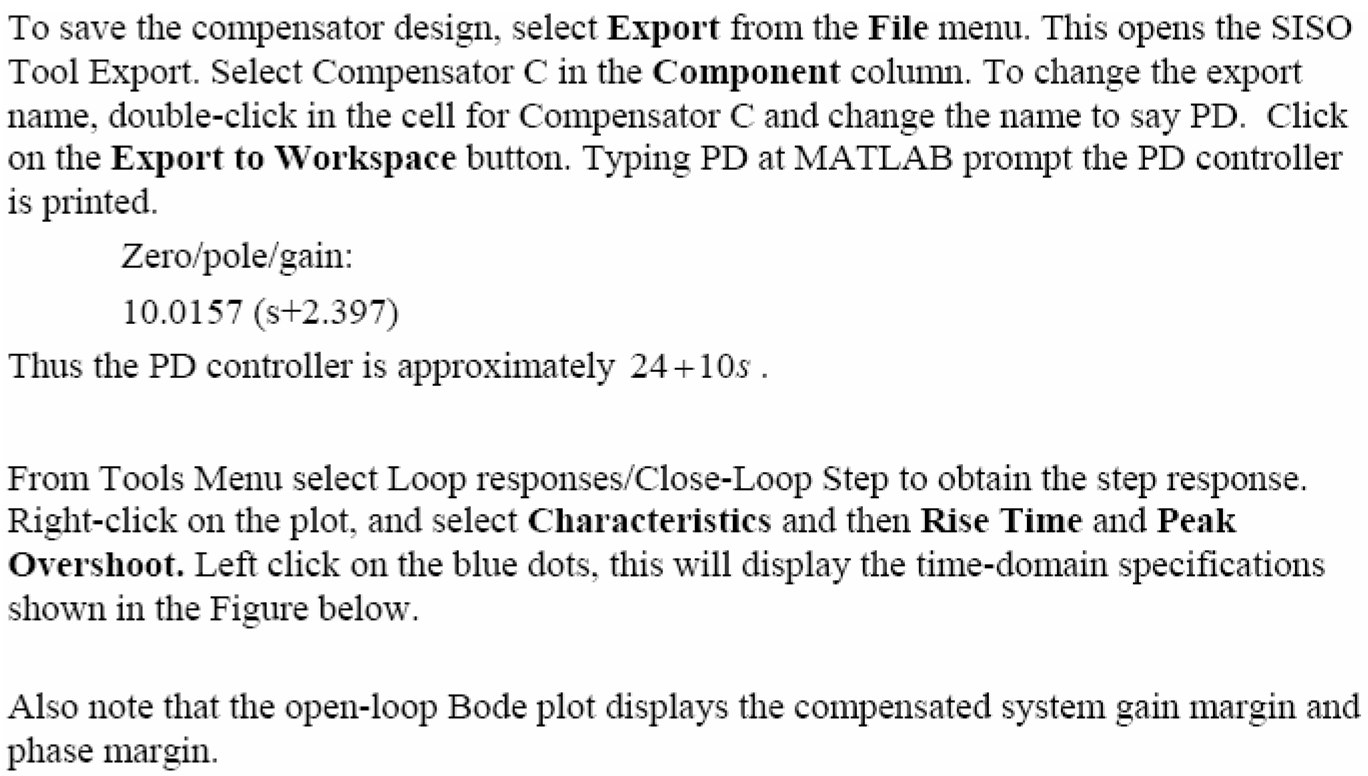
C(s) = 24 x (1 + 0.42s) / 1

G.M.: 261 dB
Freq: 1.05e+007 rad/sec

P.M.: 62.6 deg
Freq: 2.04 rad/sec

Drag this closed-loop pole along the locus to adjust the loop gain.
Current location: -2 + 2i   Damping: 0.707   Natural Frequency: 2.83 rad/sec

3.0 SICO Root Locus Design



To save the compensator design, select **Export** from the **File** menu. This opens the SISO Tool Export. Select Compensator C in the **Component** column. To change the export name, double-click in the cell for Compensator C and change the name to say PD. Click on the **Export to Workspace** button. Typing PD at MATLAB prompt the PD controller is printed.

Zero/pole/gain:

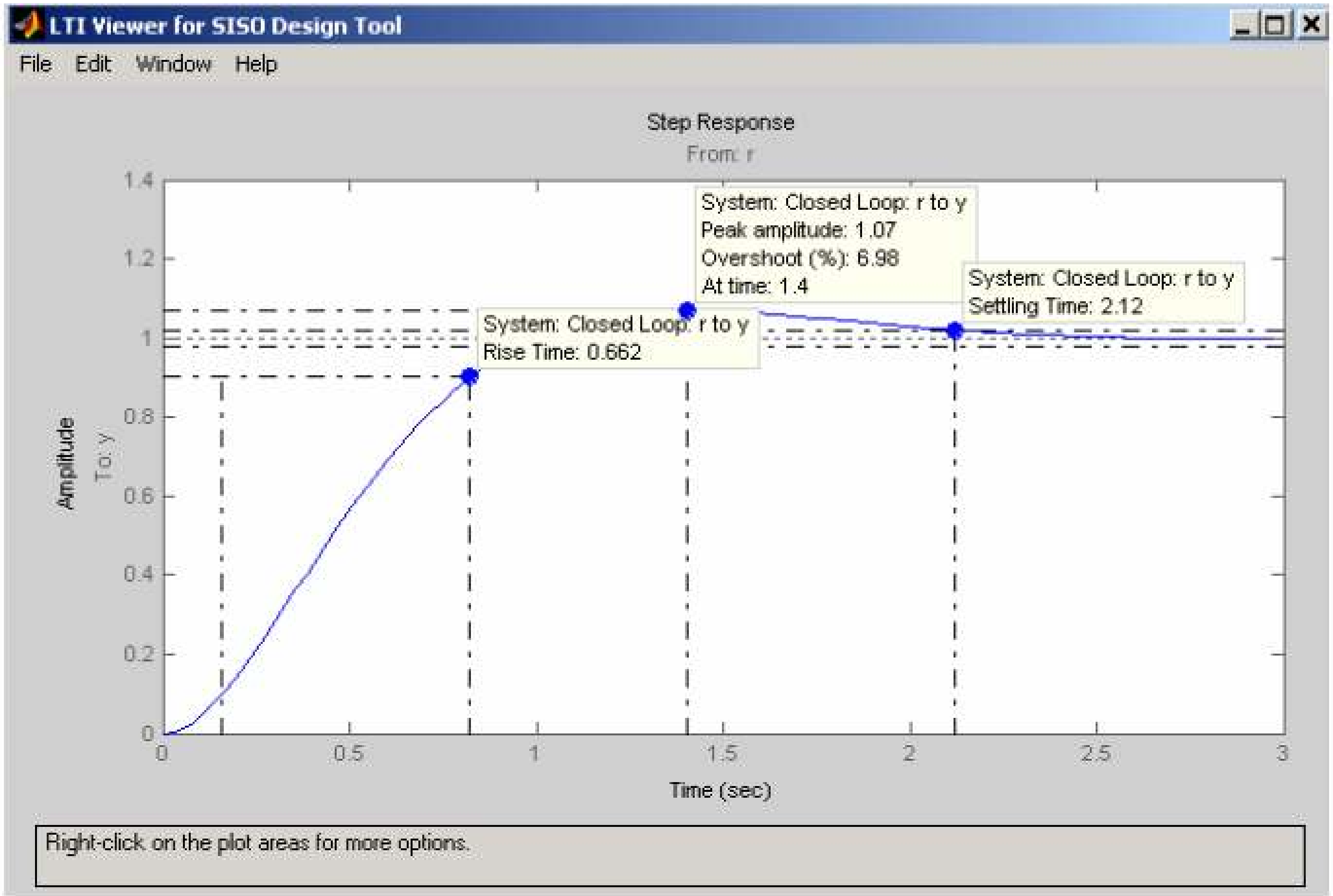
10.0157 (s+2.397)

Thus the PD controller is approximately $24+10s$.

From Tools Menu select Loop responses/Close-Loop Step to obtain the step response. Right-click on the plot, and select **Characteristics** and then **Rise Time** and **Peak Overshoot.** Left click on the blue dots, this will display the time-domain specifications shown in the Figure below.

Also note that the open-loop Bode plot displays the compensated system gain margin and phase margin.

3.0 SICO Root Locus Design



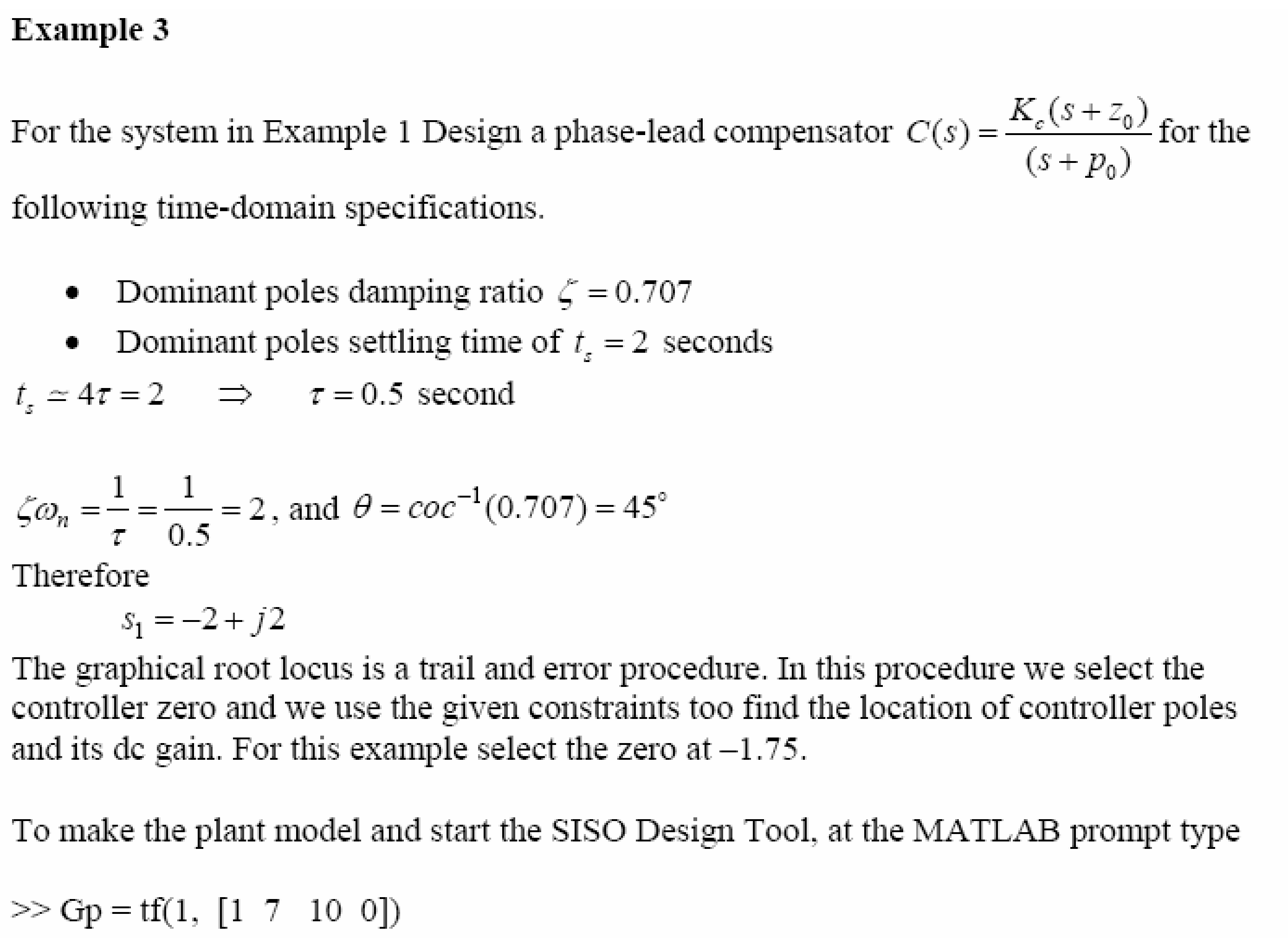
System: Closed Loop: r to y
Peak amplitude: 1.07
Overshoot (%): 6.98
At time: 1.4

System: Closed Loop: r to y
Rise Time: 0.662

System: Closed Loop: r to y
Settling Time: 2.12

Right-click on the plot areas for more options.

3.0 SICO Root Locus Design



Example 3

For the system in Example 1 Design a phase-lead compensator $C(s)=\dfrac{K_c(s+z_0)}{(s+p_0)}$ for the following time-domain specifications.

- Dominant poles damping ratio $\zeta=0.707$
- Dominant poles settling time of $t_s=2$ seconds

$t_s \simeq 4\tau = 2 \quad \Rightarrow \quad \tau = 0.5$ second

$\zeta\omega_n=\dfrac{1}{\tau}=\dfrac{1}{0.5}=2$, and $\theta = coc^{-1}(0.707)=45^{\circ}$

Therefore

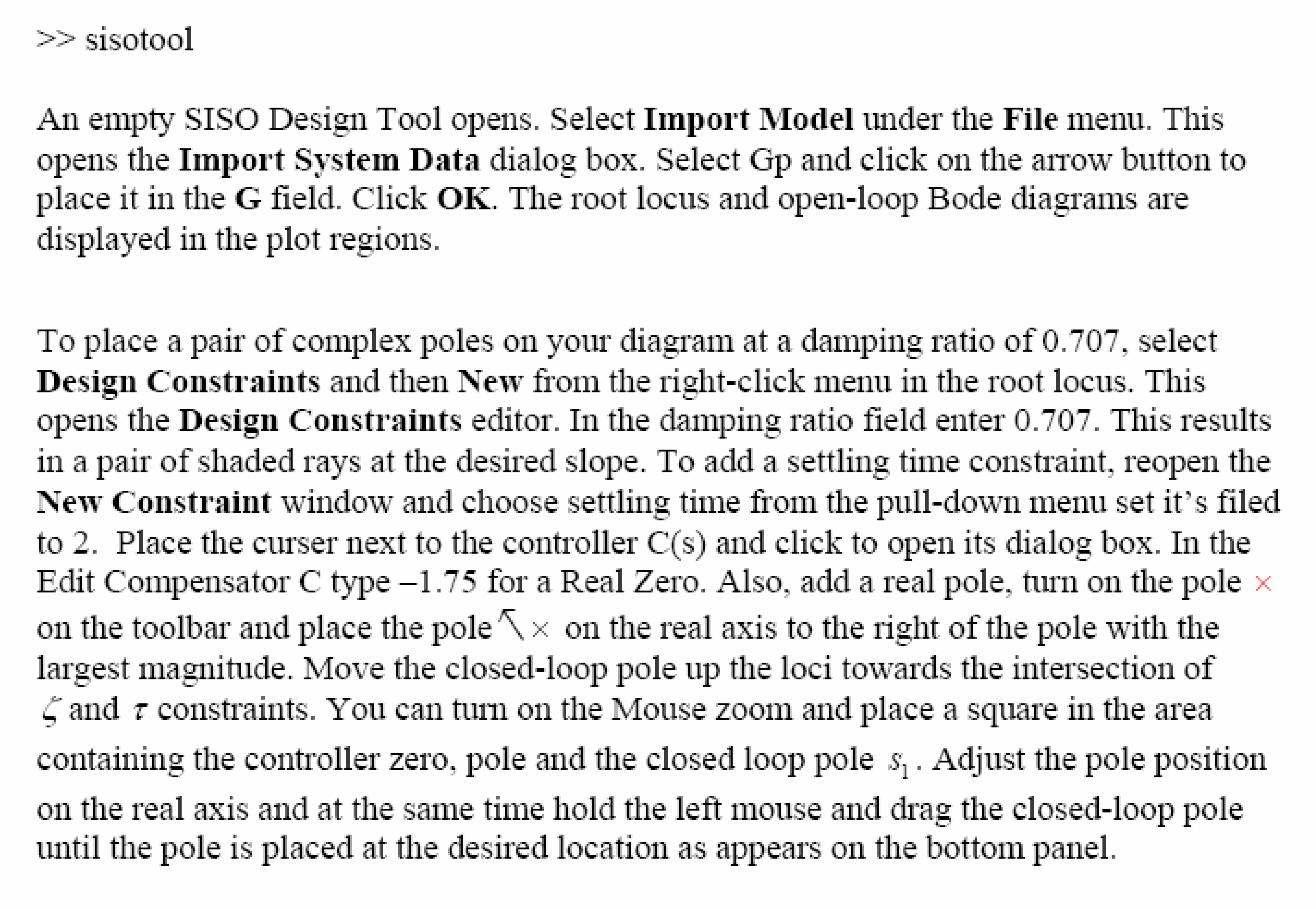
$s_1=-2+j2$

The graphical root locus is a trail and error procedure. In this procedure we select the controller zero and we use the given constraints too find the location of controller poles and its dc gain. For this example select the zero at –1.75.

To make the plant model and start the SISO Design Tool, at the MATLAB prompt type

```
>> Gp = tf(1,  [1  7   10  0])
```

3.0 SICO Root Locus Design

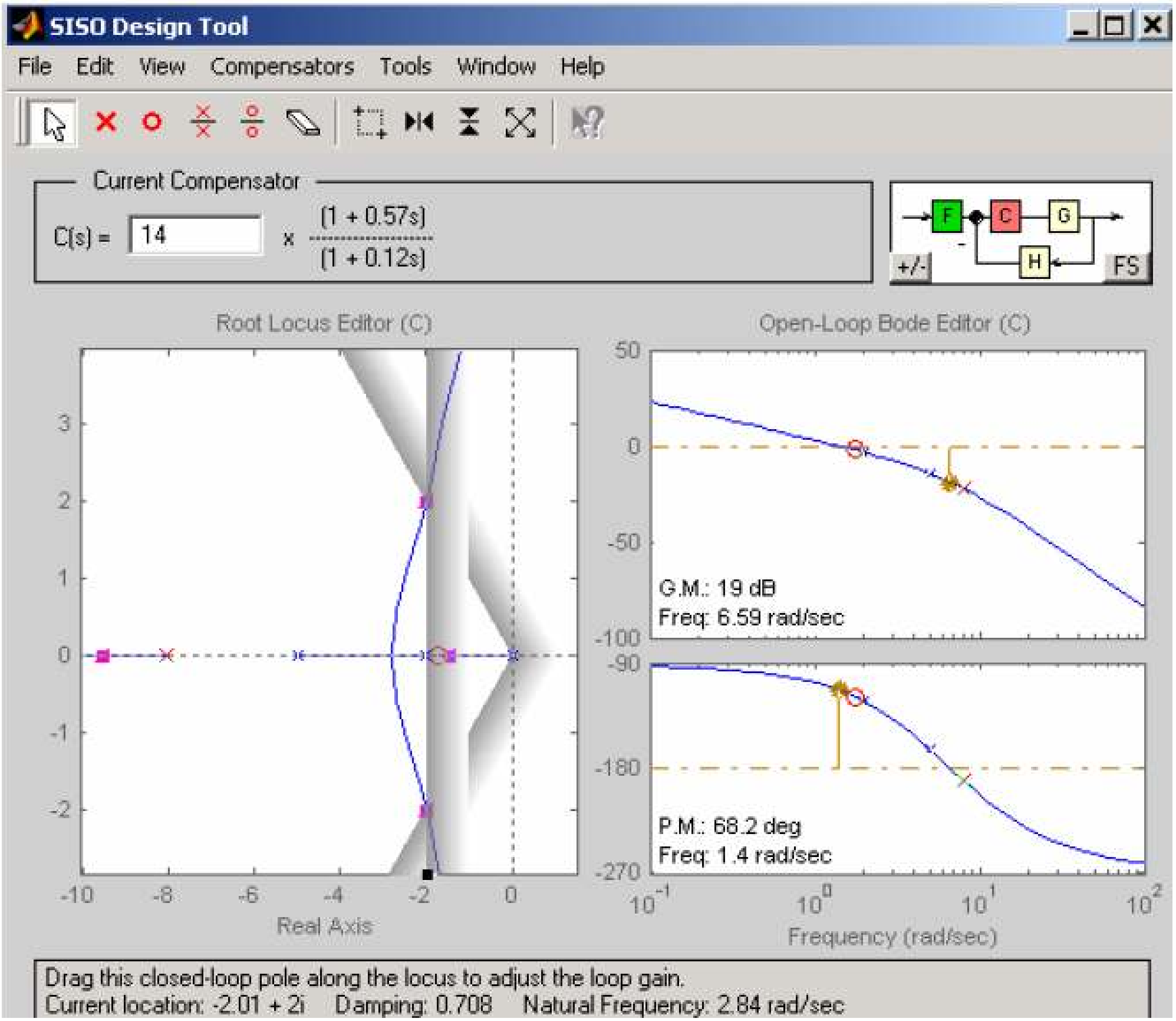


```
>> sisotool
```

An empty SISO Design Tool opens. Select **Import Model** under the **File** menu. This opens the **Import System Data** dialog box. Select Gp and click on the arrow button to place it in the **G** field. Click **OK**. The root locus and open-loop Bode diagrams are displayed in the plot regions.

To place a pair of complex poles on your diagram at a damping ratio of 0.707, select **Design Constraints** and then **New** from the right-click menu in the root locus. This opens the **Design Constraints** editor. In the damping ratio field enter 0.707. This results in a pair of shaded rays at the desired slope. To add a settling time constraint, reopen the **New Constraint** window and choose settling time from the pull-down menu set it's filed to 2. Place the curser next to the controller C(s) and click to open its dialog box. In the Edit Compensator C type –1.75 for a Real Zero. Also, add a real pole, turn on the pole × on the toolbar and place the pole ↖× on the real axis to the right of the pole with the largest magnitude. Move the closed-loop pole up the loci towards the intersection of ζ and τ constraints. You can turn on the Mouse zoom and place a square in the area containing the controller zero, pole and the closed loop pole s_1. Adjust the pole position on the real axis and at the same time hold the left mouse and drag the closed-loop pole until the pole is placed at the desired location as appears on the bottom panel.

3.0 SICO Root Locus Design



Current Compensator

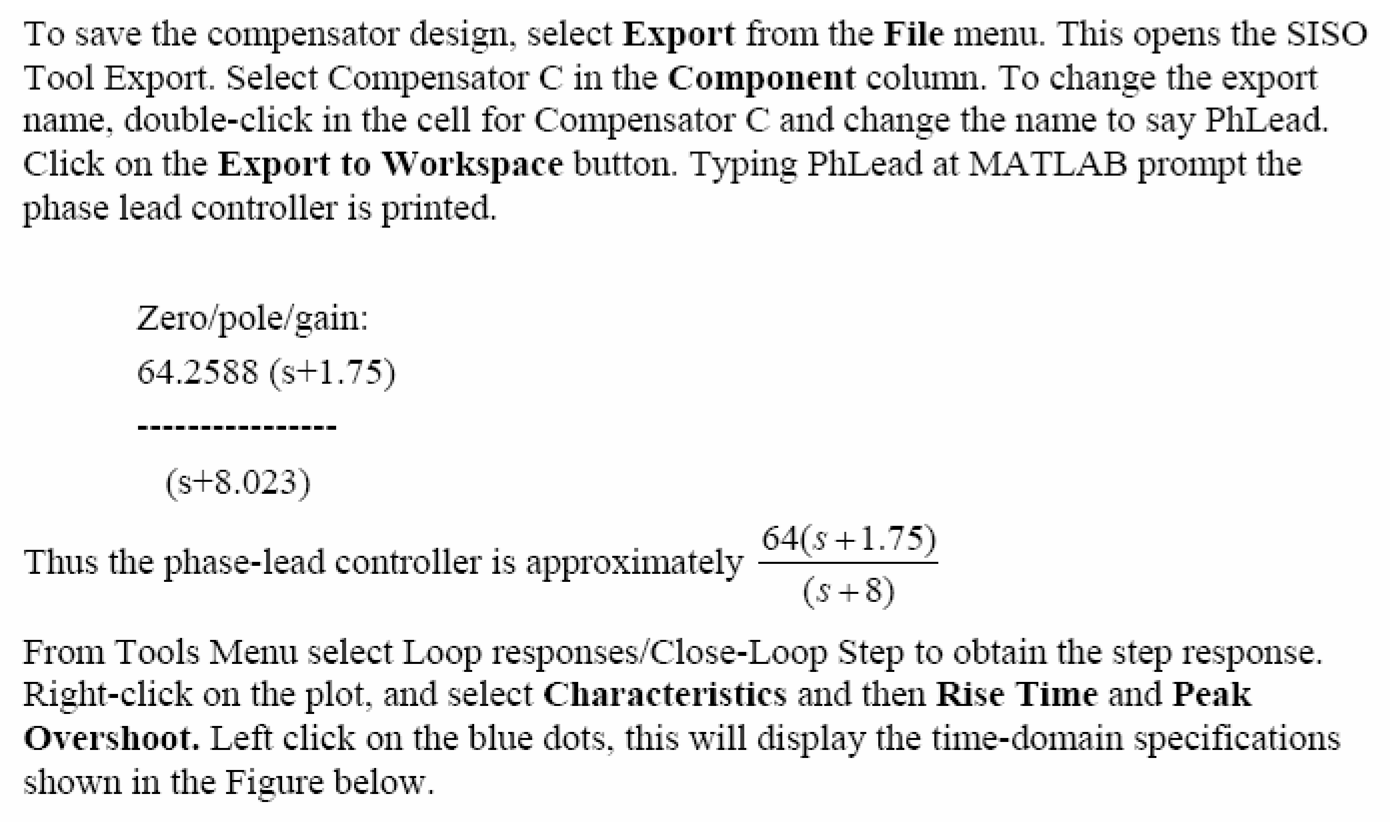
$C(s) = 14 \times \frac{(1 + 0.57s)}{(1 + 0.12s)}$

G.M.: 19 dB
Freq: 6.59 rad/sec

P.M.: 68.2 deg
Freq: 1.4 rad/sec

Drag this closed-loop pole along the locus to adjust the loop gain.
Current location: -2.01 + 2i   Damping: 0.708   Natural Frequency: 2.84 rad/sec

3.0 SICO Root Locus Design



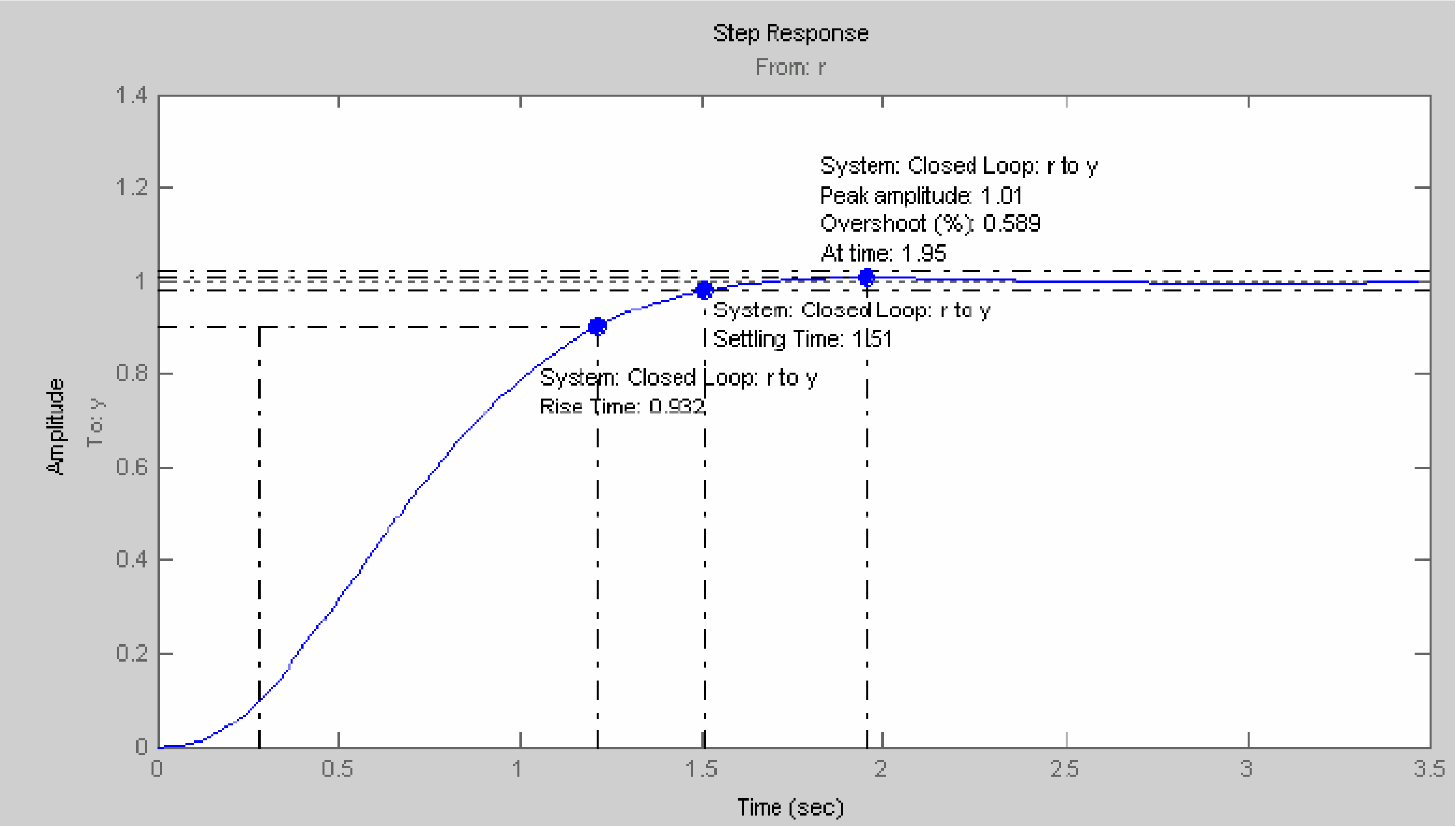
To save the compensator design, select **Export** from the **File** menu. This opens the SISO Tool Export. Select Compensator C in the **Component** column. To change the export name, double-click in the cell for Compensator C and change the name to say PhLead. Click on the **Export to Workspace** button. Typing PhLead at MATLAB prompt the phase lead controller is printed.

```
Zero/pole/gain:
64.2588 (s+1.75)
----------------
  (s+8.023)
```

Thus the phase-lead controller is approximately $\dfrac{64(s+1.75)}{(s+8)}$

From Tools Menu select Loop responses/Close-Loop Step to obtain the step response. Right-click on the plot, and select **Characteristics** and then **Rise Time** and **Peak Overshoot.** Left click on the blue dots, this will display the time-domain specifications shown in the Figure below.

3.0 SICO Root Locus Design

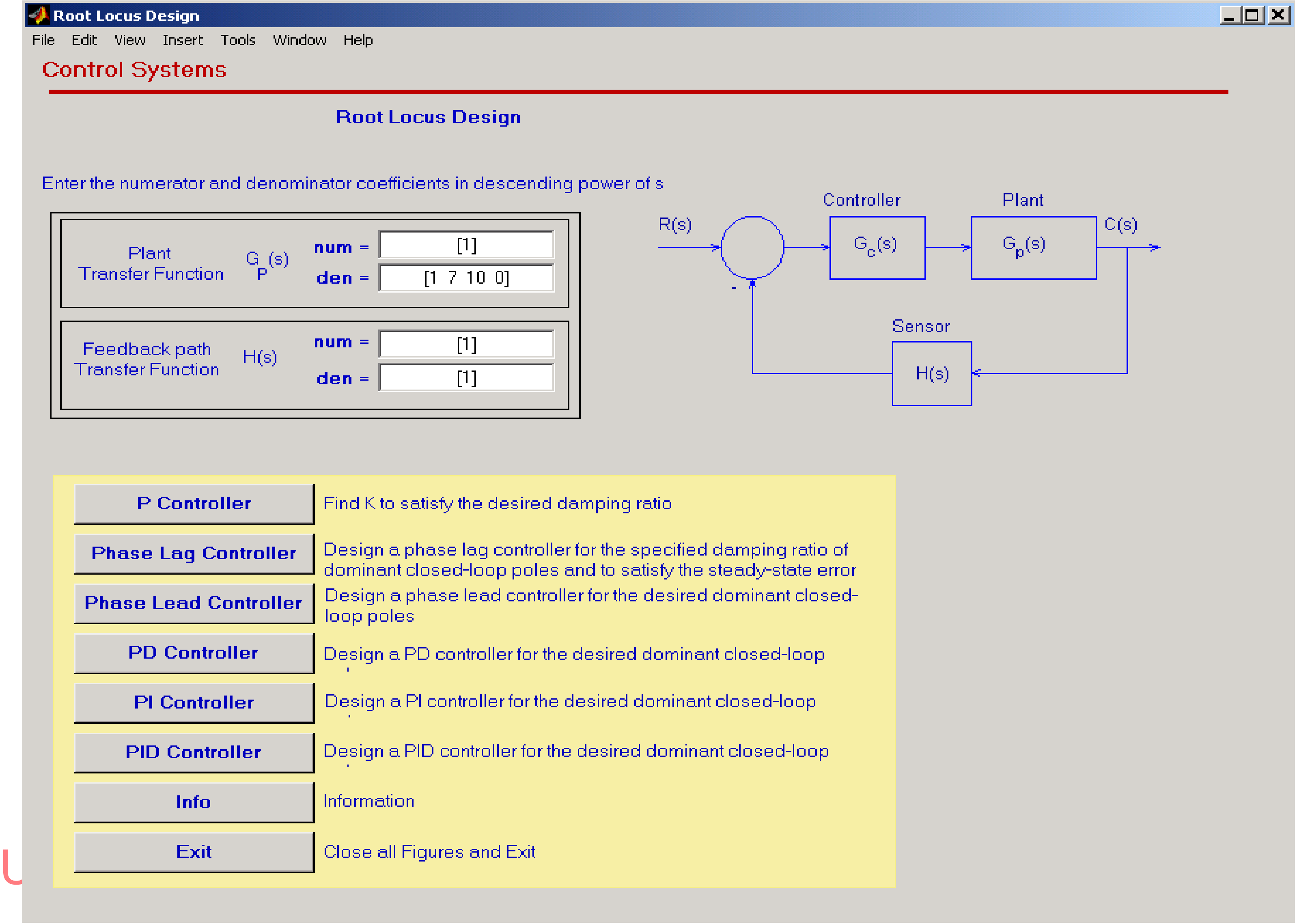


3.0 Root Locus Design

- Reference: http://www.mathworks.com

- Matlab Help

4.0 GUI for Controls



Control Systems

Root Locus Design

Enter the numerator and denominator coefficients in descending power of s

Plant Transfer Function $G_P(s)$ num = [1] den = [1 7 10 0]

Feedback path Transfer Function H(s) num = [1] den = [1]

P Controller: Find K to satisfy the desired damping ratio

Phase Lag Controller: Design a phase lag controller for the specified damping ratio of dominant closed-loop poles and to satisfy the steady-state error

Phase Lead Controller: Design a phase lead controller for the desired dominant closed-loop poles

PD Controller: Design a PD controller for the desired dominant closed-loop

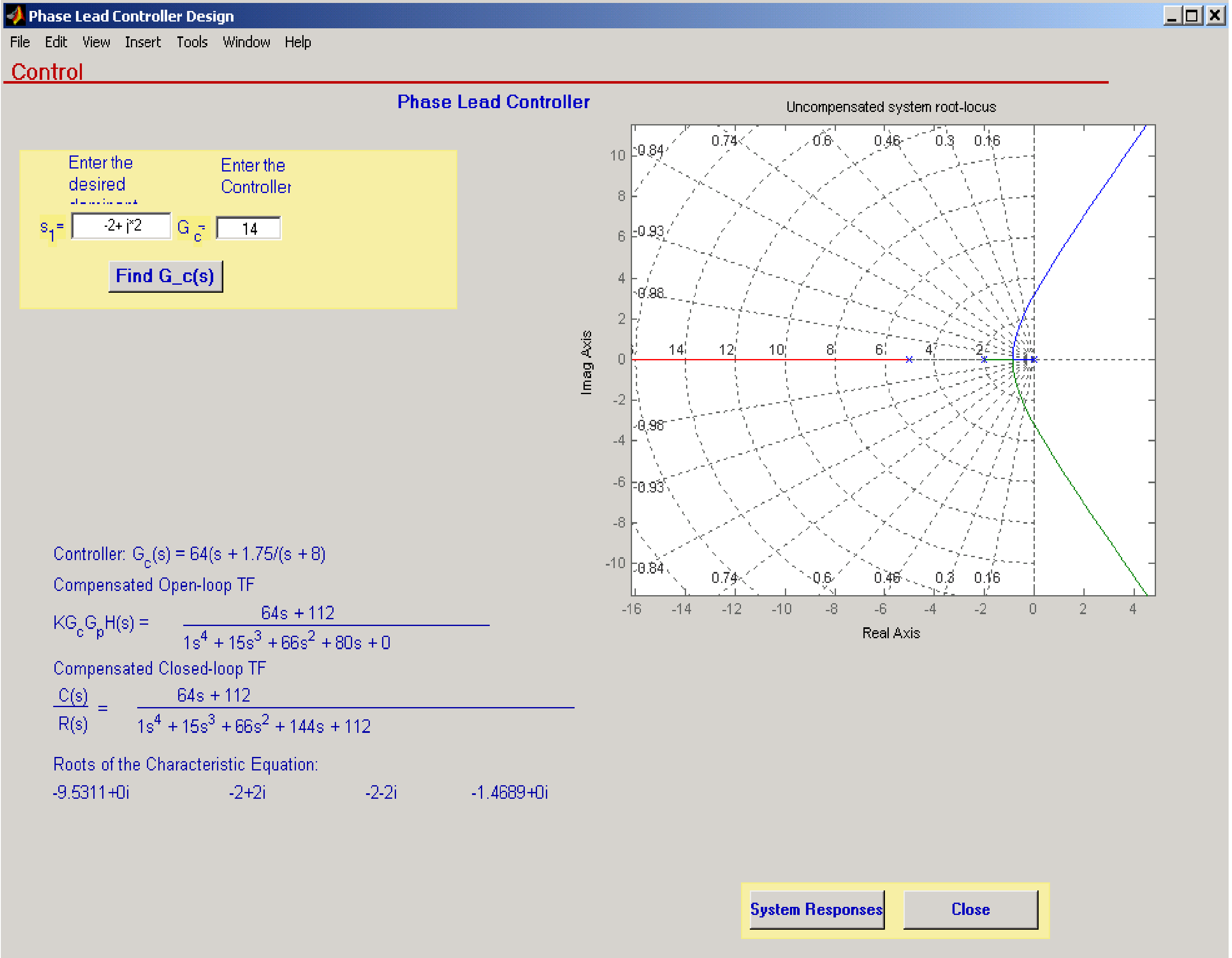
PI Controller: Design a PI controller for the desired dominant closed-loop

PID Controller: Design a PID controller for the desired dominant closed-loop

Info: Information

Exit: Close all Figures and Exit

4.0 GUI for Controls



s_1 = -2+ j*2 G_c = 14

Controller: $G_c(s) = 64(s + 1.75/(s + 8)$

Compensated Open-loop TF

$$KG_cG_pH(s) = \frac{64s + 112}{1s^4 + 15s^3 + 66s^2 + 80s + 0}$$

Compensated Closed-loop TF

$$\frac{C(s)}{R(s)} = \frac{64s + 112}{1s^4 + 15s^3 + 66s^2 + 144s + 112}$$

Roots of the Characteristic Equation:

-9.5311+0i -2+2i -2-2i -1.4689+0i

Concluding Remarks

- This tutorial and the GUI for Controls can be downloaded from
- http://www.cs.utah.edu/~malko/controls/