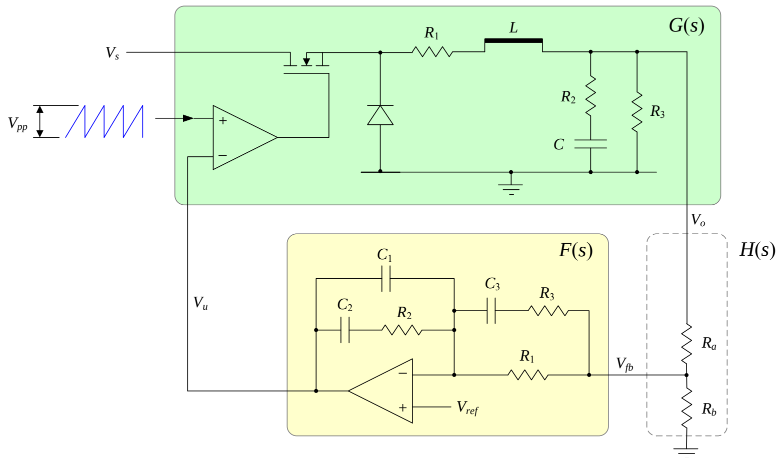
Feedback Control Tutorial

The diagram below shows a type 3 compensator applied to a switching buck regulator.



Passive component values for the buck output filter are:

$R_1 = 2.2\text{m}\Omega$ $\qquad$ $L = 0.9\mu\text{H}$

$R_2 = 0.6\text{m}\Omega$ $\qquad$ $C = 471\mu\text{F}$

$R_3 = 1\Omega$

The feedback divider has an attenuation of 0.5.

1. Model the output filter network in Matlab
2. Design a phase lead compensator to achieve a phase margin of at least 45º and a gain margin of greater than 10dB. Ensure the cross-over frequency is above 10kHz.
3. Construct S & T curves for the switching converter of tutorial 1 using the phase lead compensator designed in tutorial 4. Determine the values of M_S & M_T.
4. Determine nominal stability for the switching converter of tutorial 4. Plot the sensitivity function S and select a suitable weighting function to specify required performance. Determine whether nominal performance has been met by plotting the W_1S curve.
5. Use the Robust Control Toolbox in Matlab to model the frequency response of the passive LC filter of tutorial 1 with inductor value uncertainty of ±8% and variation of capacitor value in the range 456µF to 503µF.
6. Determine a valid shape for the W_2 weighting function and use it to determine whether the controller designed in tutorial 4 achieves robust stability for this uncertain plant model.
7. Design a controller which achieves robust performance for the buck switching converter of tutorial 4.

Solution

1. Straightforward Laplace analysis of the buck output filter yields the transfer function:

$$G(s)=\frac{sCR_2R_3+R_3}{s^2LC(R_2+R_3)+s(CR_1(R_2+R_3)+CR_2R_3+L)+(R_1+R_3)}$$

The tutorial solution file "w6_tutorial.m" defines the passive component values:

```
L1 = 0.9e-06;       % inductor = 0.9uH
C1 = 471e-06;       % capicitor = 471uF
R1 = 2.2e-03;       % DCR = 2.2mR
R2 = 0.6e-03;       % ESR = 0.6mR
R3 = 1;             % load = 1R
```

Next, the transfer function coefficients are calculated using the transfer function above:

```
b1 = C1*R2*R3;
b0 = R3;
a2 = L1*C1*(R3 + R2);
a1 = (C1*R3*R2) + (C1*R1*(R3 + R2)) + L1;
a0 = R3 + R1;
```

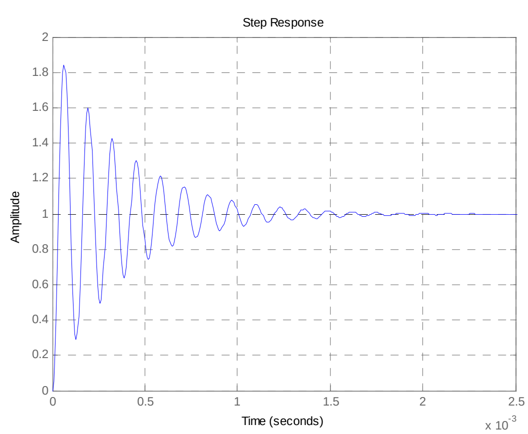
The Matlab Control Systems Toolbox contains the command `'tf'` which allows an LTI system to be defined in terms of it's transfer function coefficients as follows:

```
G = tf([b1 b0], [a2 a1 a0]);
```

The workspace now contains the variable `G` representing the LC network model. We can evaluate the unit step response of the filter using the script:

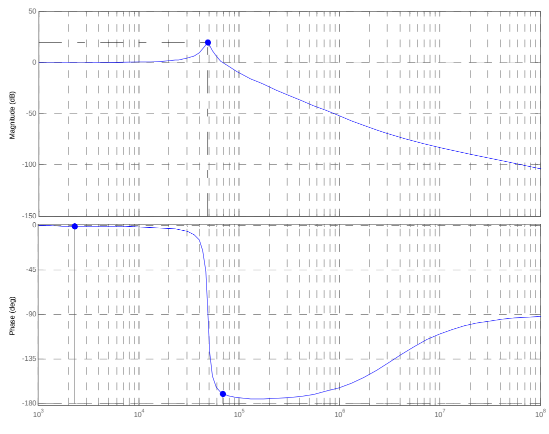
```
step(G)
```

This produces the following graph:



Clearly the step response exhibits a strongly oscillatory character. We see this in the resonant peak in the frequency response of the system which can be computed using `bode(G)`, but a better visualisation results with:

```
ltiview('bode',G)
```



The resonant peak is clearly visible, as is the rapid change of phase associated with it. The `ltiview` command in Matlab enables important characteristics of the response such as peak magnitude and stability margins to be measured and displayed on the plot.

Click on any part of the magnitude or phase plots to find the response at that frequency. In the Matlab workspace, zero and pole locations of $G(s)$ can be found as follows:

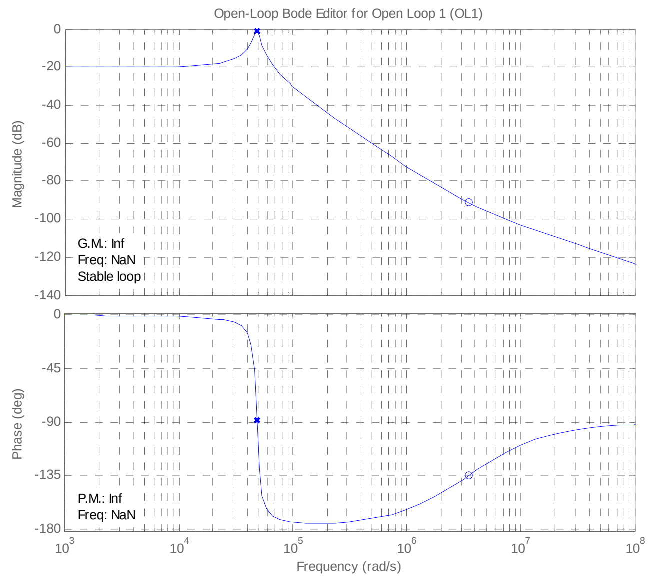
```
zero(G)

pole(G)
```

2. The transfer function of the type 3 compensator is:

$$F(s) = \frac{R_1 + R_3}{R_1 R_3 C_1} \frac{\left(s + \frac{1}{R_2 C_2}\right)\left(s + \frac{1}{(R_1 + R_3) C_3}\right)}{s\left(s + \frac{C_1 + C_2}{R_2 C_1 C_2}\right)\left(s + \frac{1}{R_3 C_3}\right)}$$

The transfer function of the output filter was found in tutorial 1. The set-up script for this tutorial is contained in the Matlab file `Tutorial_4.m` which creates transfer function objects in the Matlab workspace for the plant and feedback network, then launches the "sisotool" design tool.

The Bode plot for the open loop is shown below. Notice the presence of the resonant peak form the LC circuit modelled in tutorial 1, and of the high frequency zero arising from the ESR of the output capacitor.



At this stage the controller is a simple gain. To determine the zero and pole locations from the Matlab workspace, type:

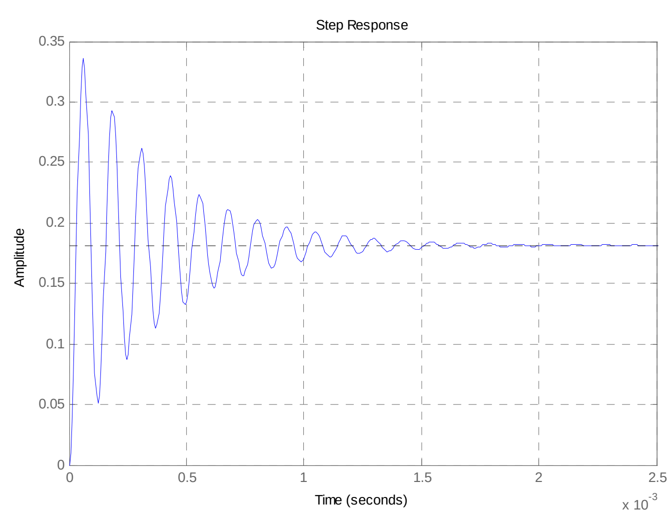
```
zpk(G*H)
```

This reveals as expected that the open-loop consists of a single real zero and two complex poles.

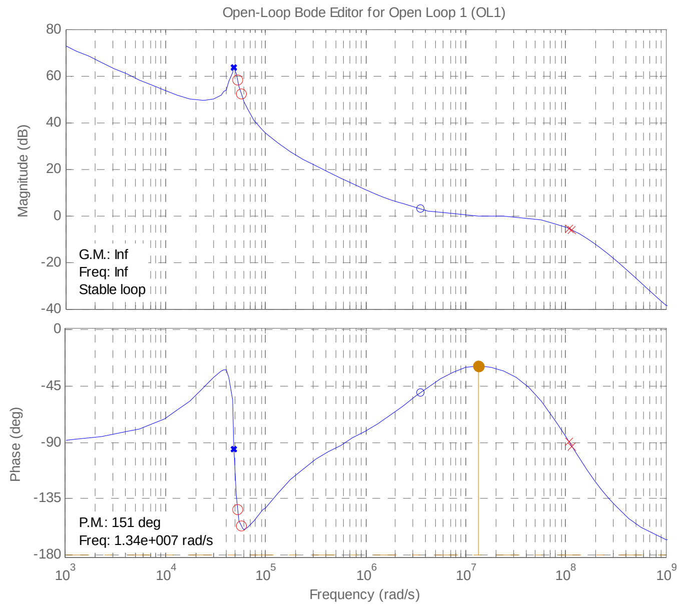
$$F(s) = 66.6267\frac{s + 3.539\times10^{6}}{s^{2} + 5233s + 2.363\times10^{9}}$$

With a unity gain controller the step response exhibits a lot of oscillation and there is very large steady-state error.

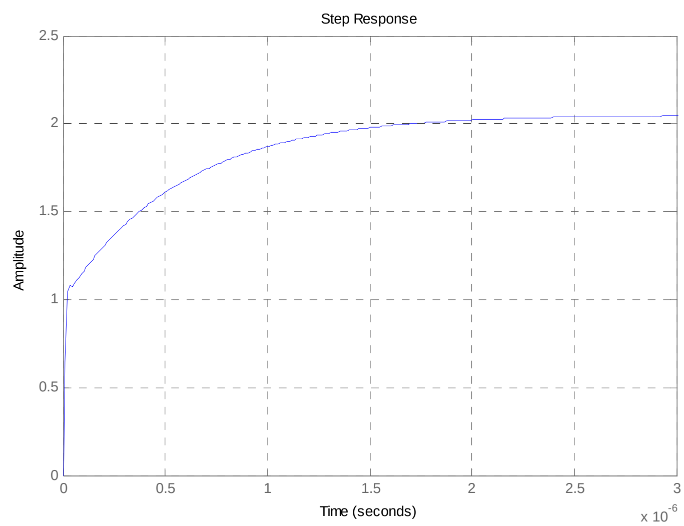
```
step(feedback(G,H))
```



The proposed type III controller contains two zeros, an integrator, and two poles. Placing the zeros near the resonant peak, and both poles well beyond the zero results in the Bode plot shown below.



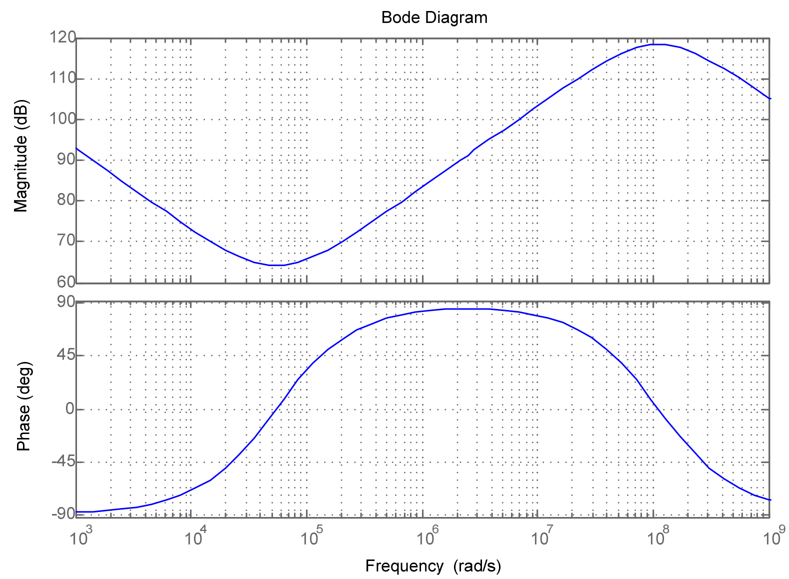
The corresponding step response shows a much faster response with little sign of oscillation.



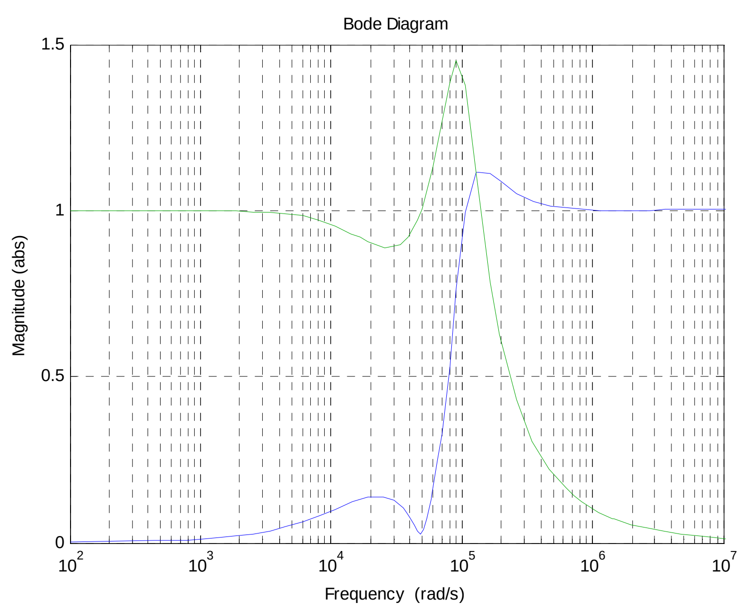
The compensator can now be exported to the Matlab workspace, where we find it's transfer function is:

$$F(s) = 185256517873073.3\,\frac{\left(s+5.253\times10^{4}\right)\left(s+5.717\times10^{4}\right)}{s\left(s+1.16\times10^{8}\right)\left(s+1.086\times10^{8}\right)}$$

The Bode plot for this compensator is shown below. The effect of the integrator and controller roots is clearly visible in the phase curve.



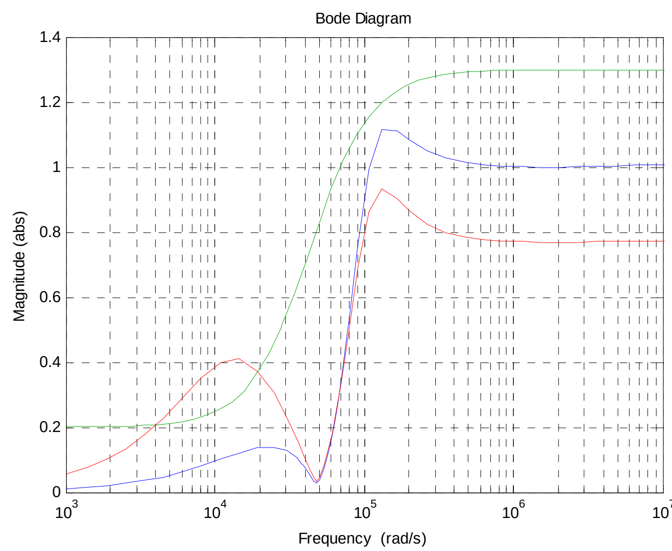
3. The S & T curves are plotted using the nominal controller, plant, and feedback transfer function for this switching converter. The combined graph is shown below.



As expected for a system with integrating controller, the DC gain of the sensitivity curve (blue) is approximately zero and rises to 1 at high frequency. The transition happens near the resonant frequency of the LC network, and is accompanied by a peak in the magnitude curve. At frequencies where the S curve is greater than 1, the Nyquist curve has entered the unit circle centred on the critical point.

The complementary sensitivity curve (green) exhibits a very strong peak near resonance, indicating peaking in the closed loop gain around this frequency. Peaking in the T curve indicates the Nyquist curve lies to the left of a vertical line at -0.5.

4. The tutorial solution script computes the sensitivity function and plots its magnitude (blue). The script uses the function rps_makeweight to specify a weight (green) whose inverse lies above the *S* curve at all frequencies. The sensitivity and weighting product is then calculated and the result plotted on the same graph (red). The resulting graph is shown below.



The W_1S curve clearly lies below 1 at all frequencies, indicating that nominal performance has indeed been achieved. The last instruction in the script calculates the infinity norm of the W_1S function and prints it to the workspace. The indicated value of 0.933 confirms NP has been achieved with this controller.

5. Both passive component variations are specified in terms of parametric uncertainty. The Robust Control Toolbox in Matlab contains the function `ureal`, which permits uncertainty to be expressed as either a percentage of the nominal value, or as a specified range.

The inductor and capacitor tolerances specified above would be applied as follows:

```
L1 = ureal('L1',0.9e-06,'Percentage',[-8 8]);
```

```
C1 = ureal('C1',471e-06,'Range',[456e-06 503e-06]);
```

The transfer function coefficients can now be computed in the same way as before…

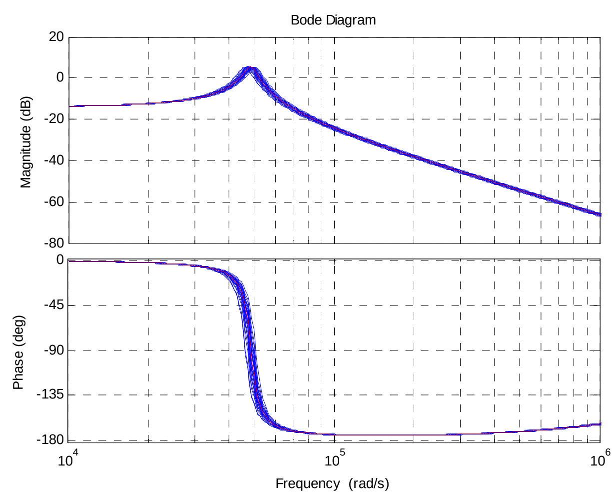
```
b1 = C1*R2*R3;
b0 = R3;
a2 = L1*C1*(R3 + R2);
a1 = (C1*R3*R2) + (C1*R1*(R3 + R2)) + L1;
a0 = R3 + R1
```

We construct a frequency description of the uncertain plant model using the script:

```
Gfrd = frd(Gunc,logspace(3,8,500));
```

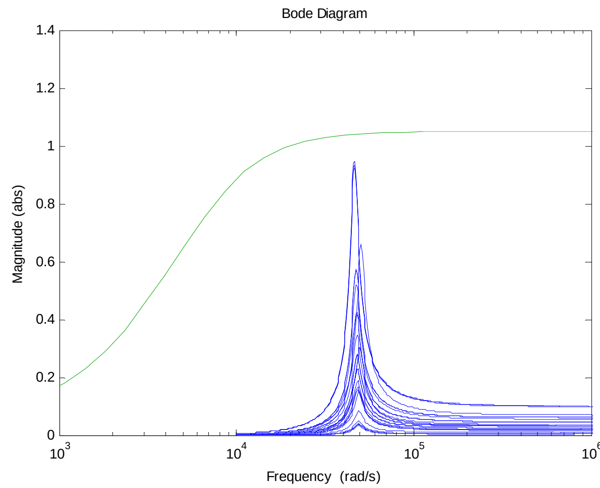
To plot 25 sample frequency responses for random parameter variation over the specified range:

```
bodeplot(usample(Gfrd,25))
```



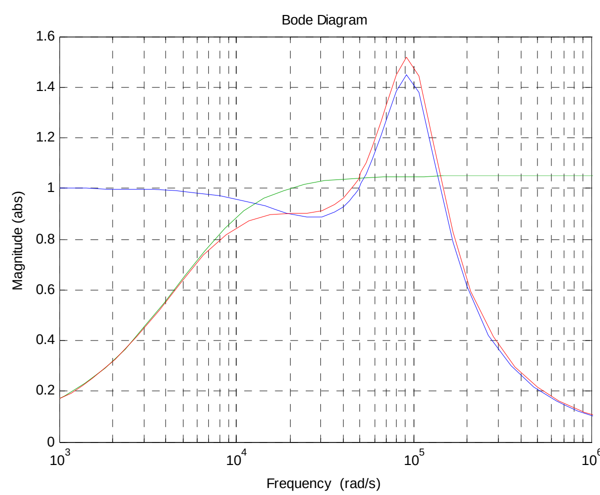
Observe that the major contribution of uncertainty in L & C is to change the frequency of the resonance rather than the size of the peak.

To determine robust stability we need to find the weighting function W_2 which bounds the worst case plant perturbation. To do this, construct the graph of relative plant magnitude perturbation over frequency and select a weighting function which lies above the curve at all frequencies.



From the previous two graphs, it is clear that the major effect of uncertainty in L & C is in the frequency range near resonance. The peaks near $5x10^4$ rad/sec indicate large potential errors in relative plant model uncertainty there.

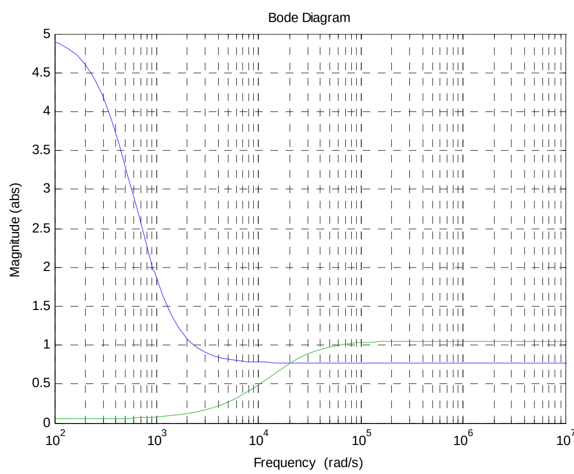
We now have all the information we need to determine whether stability will hold over all operating conditions. Plot the graphs of nominal complementary sensitivity function T, and W_2. then multiply these and plot the W_2T curve on the same axes. These three curves are shown below.



To achieve robust stability, the W_2T curve must lie below 1 at all frequencies. Clearly there is a range of frequencies over which this is not true, indicating loss of stability for at least some possible values of L & C.

The cause is the plant uncertainty near the resonant peak in the nominal T curve, which is being amplified by a value of W_2 greater than 1 at that frequency. In this case the designer would have to think seriously about re-designing the controller to diminish the value of M_T. It is likely this will result in a reduction in control loop bandwidth, but this is the cost of maintaining stability over the full range of component values.

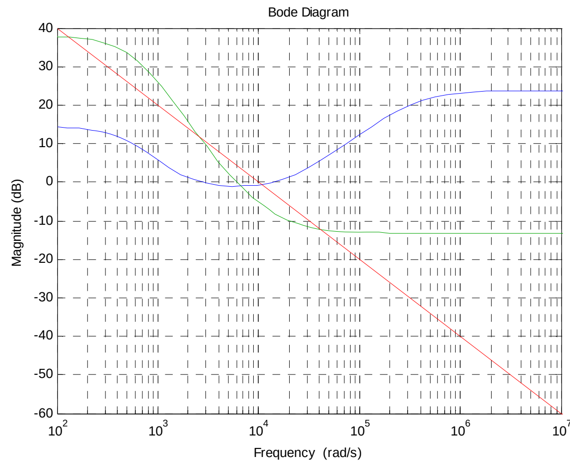
7. Performance & uncertainty weights for the buck converter were established above. We begin by plotting these on the same graph, and observe that at every frequency the minimum of the two weights lies below 1 at all frequencies, which is the necessary condition for robust performance.



We will use a loop-shaping approach to controller design. First, we construct curves to constrain the high and low frequency shape of the open loop plot based on the selected weights. A target open loop shape can then be specified which crosses unity gain with slope of no less than -2, which is the condition for closed-loop stability for minimum phase systems. In this example, we will select an integrating open-loop shape which gives constant slope of -1. In Matlab, the target loop shape is:

```
Lt = 1e+04 * tf(1, [1 0])
```

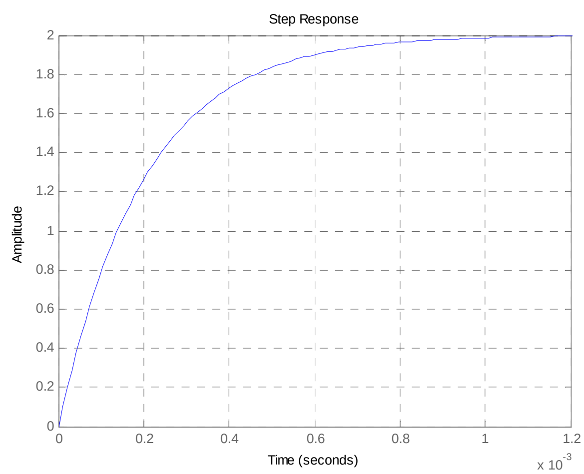
The open loop magnitude curve (red) is shown below, together with the low (blue) and high (green) frequency limits construced as described.



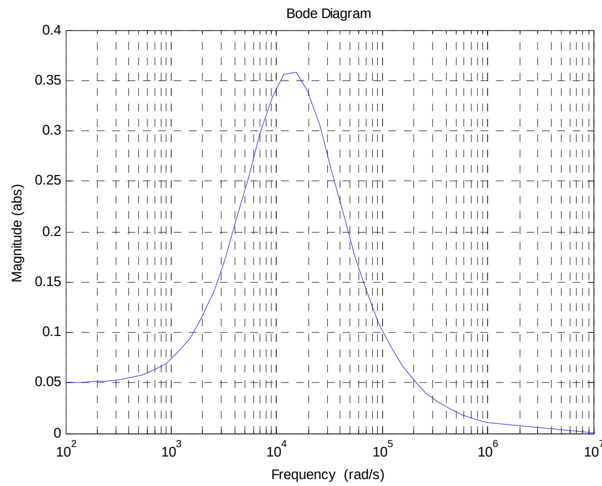
A controller which achieves this shape can be found by dividing Lt by the nominal plant model. This results in the semi-proper integrating controller:

$$F(s) = \frac{4.242 \times 10^{-6} s^2 + 0.02219 s + 10022}{5.625 \times 10^{-8} s^2 + 0.2 s}$$

A closed loop step response plot based on the nominal plant is shown below. The response is first order with time constant of approximately 200μs.



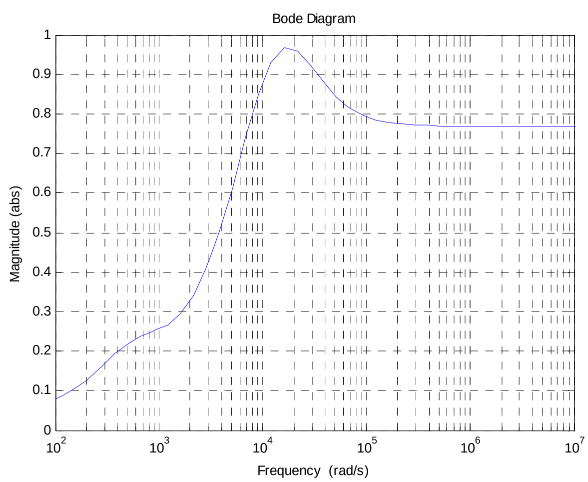
We now establish robust stability by computing the complementary sensitivity function and multiplying by the weight W_2. The resulting graph (shown below) lies below 1 at all frequencies, indicating that the design is indeed stable over all plant perturbations bounded by this weight. The infinity norm of S, computed in Matlab, is about 0.36.



Finally, we need to determine whether the performance specification defined by the weight W_1 is met for all plants in the uncertainty set defined by W_2. To do so, we plot the curve of:

$$|W_1 S| + |W_2 T|$$

…and observe the maximum peak over all frequency. The result is shown below.



The peak of this curve is about 0.965, indicating that the design has indeed achieved robust performance.

It should be noted that the high & low frequency bounds placed on the target loop shape offer no more than a guide, and that a candidate loop shape selected on this basis might not necessarily have resulted in a design achieving RS or RP. The critical frequency is around cross-over, and here the design of the loop shape is often very difficult to define. In most practical cases an iterative trial-and-error approach to design is taken. The real value of the method outlined above is that it allows candidate controller designs to be tested for robustness without the need for large scale statistical testing. ■