# Getting Started with the Control System Designer

This example shows how to tune compensators for a feedback control system using Control System Designer.

Open This Example

Using Control System Designer you can:

1) Define control design requirements on time, frequency, and pole/zero response plots.

2) Tune compensators using:

- Automated design methods, such as PID tuning, IMC, and LQG.
- Graphically tune poles and zeros on design plots, such as Bode and root locus.
- Optimization-based control design to meet time-domain and frequency-domain requirements using Simulink® Design Optimization™.

3) Visualize closed-loop and open-loop responses that dynamically update to display the control system performance.

## Compensator Design Problem

For this example, design a compensator for the system

$$G(s) = \frac{1}{s+1}$$

with the following design requirements:

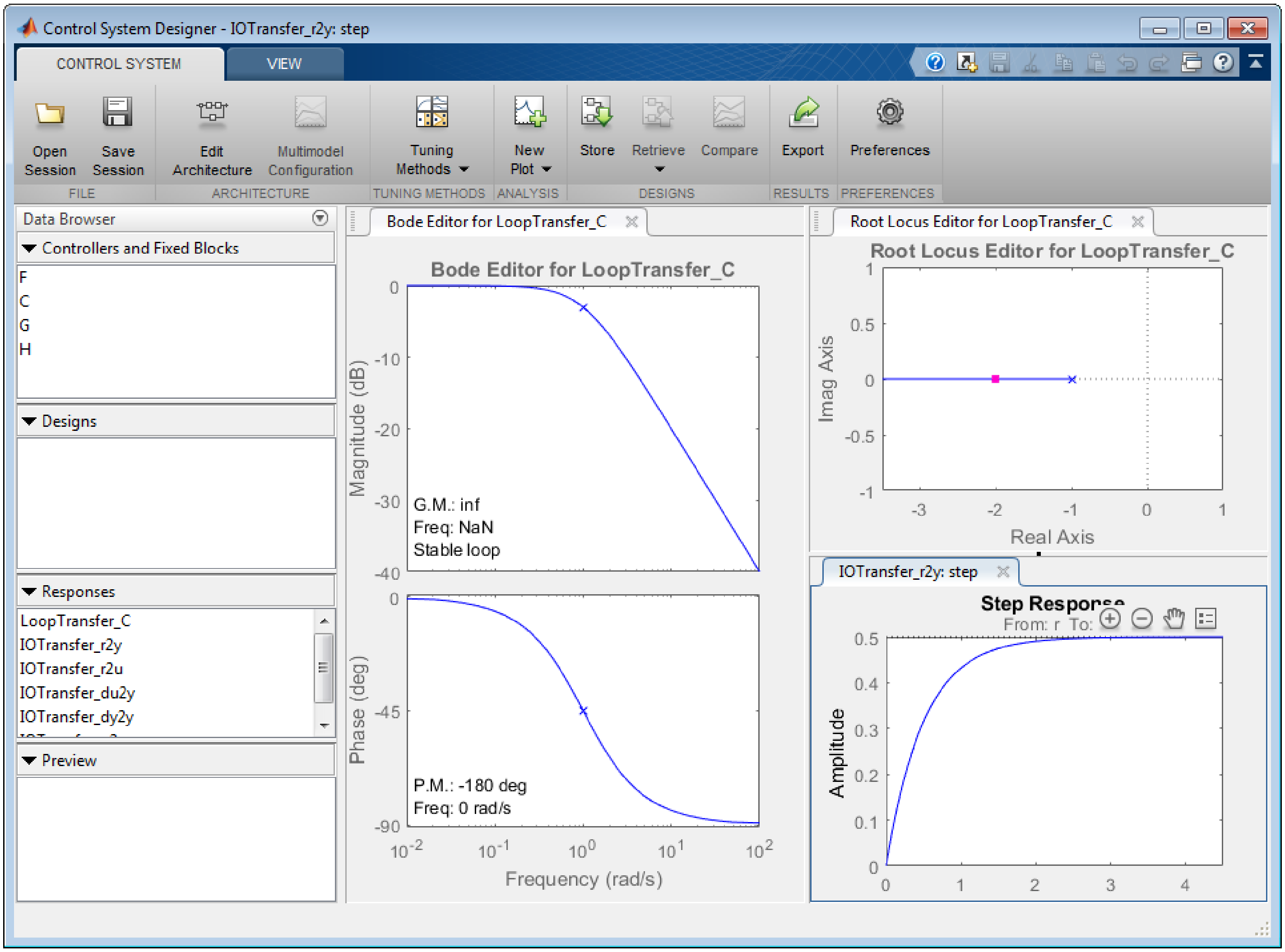
- Zero steady-state error with respect to a step input.
- 80% rise time less than 1 second.
- Settling time less than 2 seconds.
- Maximum overshoot less than 20%.
- Open-loop crossover frequency less than 5 rad/s.

## Open Control System Designer

Use the standard feedback structure with the controller in the forward path. This structure is the default Control System Designer architecture.

Open Control System Designer with the specified plant.

```
controlSystemDesigner(tf(1,[1,1]))
```

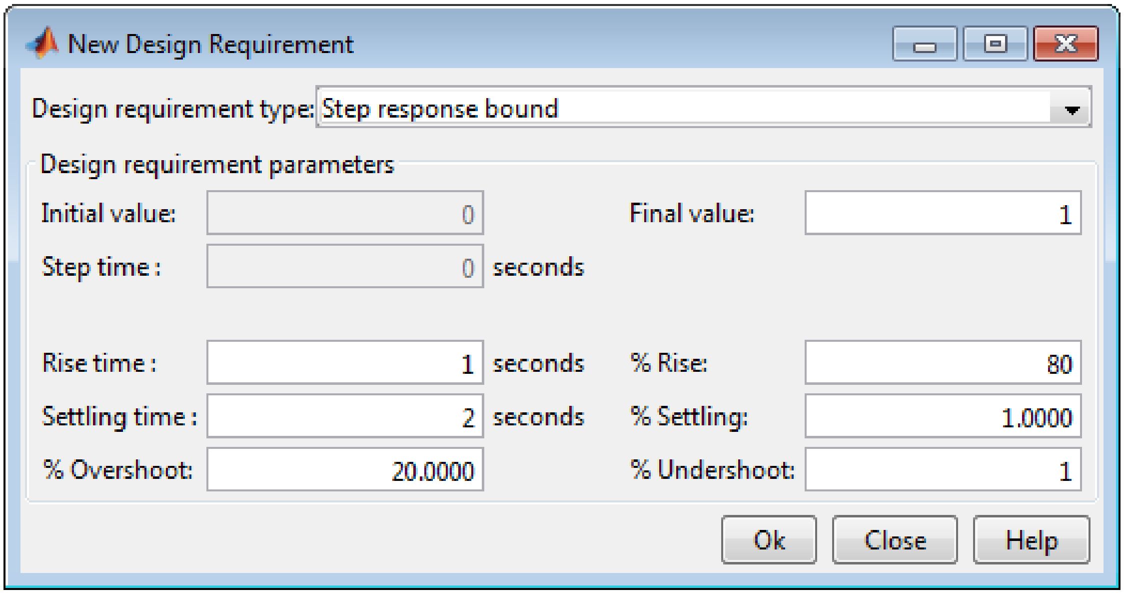



On the **Control System** tab, you can select a compensator tuning method, and create response plots for analyzing your controller performance. You can also store, compare, and export different control system designs.

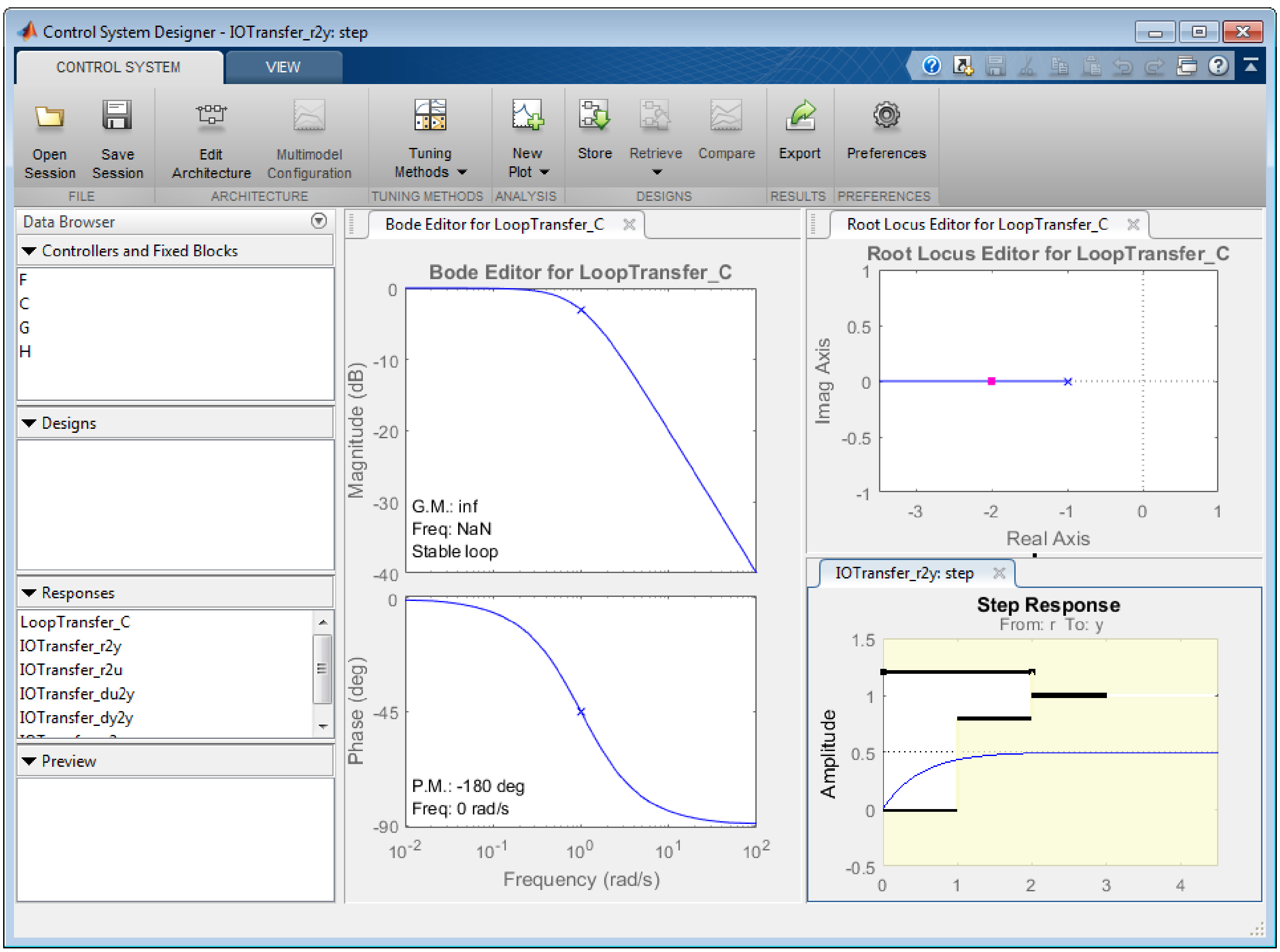
For this example, graphically tune your compensator using the **Root Locus Editor** and open-loop **Bode Editor**, and validate the design using the closed-loop **Step Response**. By default, Control System Designer displays these responses when it opens. To add additional response plots, click **New Plot**.

## Add Design Requirements

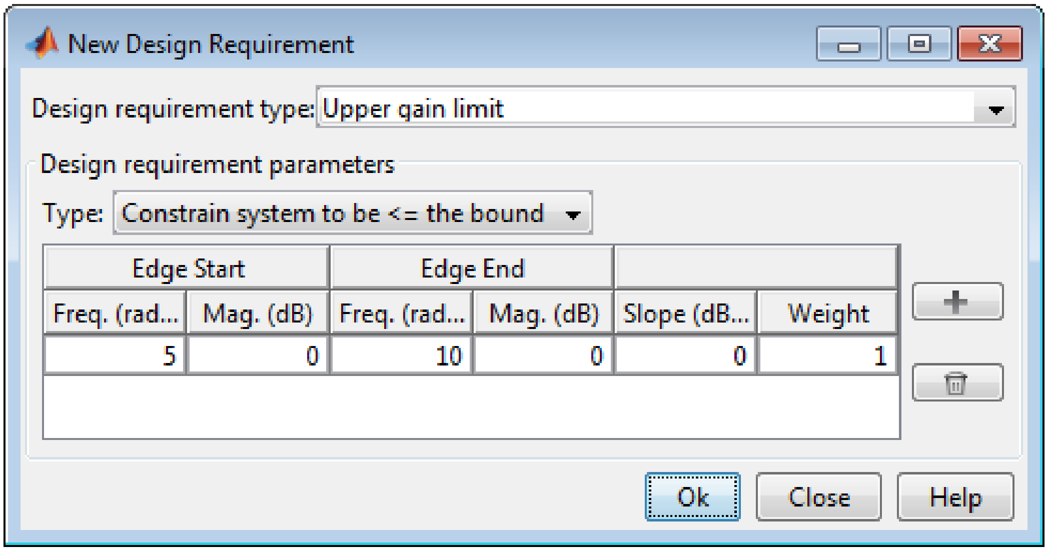
Add the time-domain design requirements to the **Step Response** plot. Right-click the plot area, and select **Design Requirements > New**. In the **Design requirmeent type** drop-down list, select `Step response bound`. Enter the time-domain design requirements.




Click **OK**. The app adds the design requirement to the step response plot as a shaded exclusion region. To meet the requirement, the step response must remain outside of this region.




To specify the frequency-domain crossover requirement, right-click the **Bode Editor** plot area, and select **Design Requirements > New**. In the **Design requirement type** drop-down list, select `Upper gain limit`, and specify the design requirement.



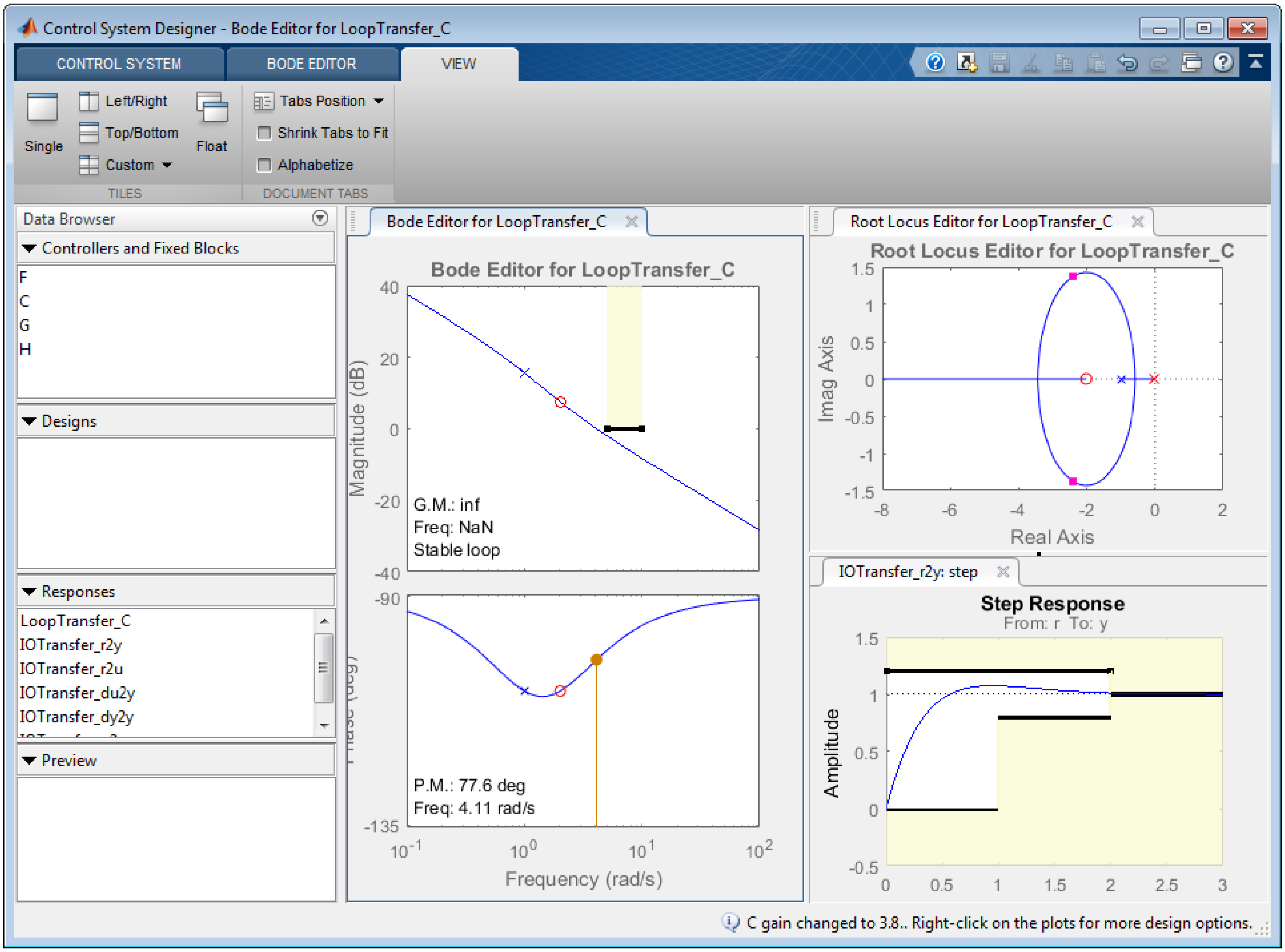

Click **OK**.

## Tune Compensator

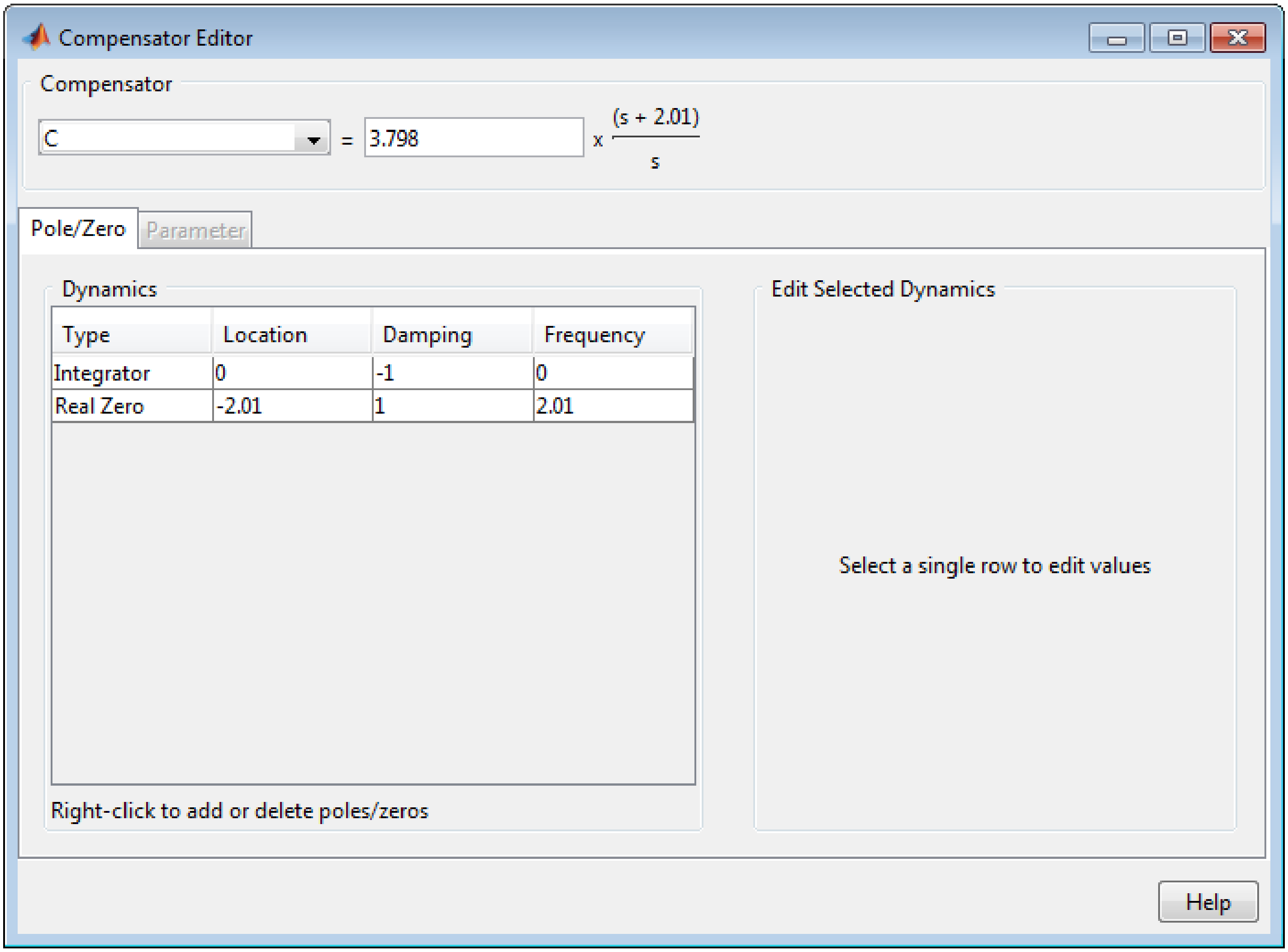
To meet the zero steady-state error design requirement, add an integrator to the compensator. Right-click the **Root Locus Editor** plot area, and select **Add Pole/Zero > Integrator**.

To create a desirable shape for the root locus plot,add a real zero near -2. Right-click the root locus plot aream and select **Add Pole/Zero > Real Zero**. In the root locus plot, left-click the real axis near -2.

To create a faster response by increasing the compensator gain, in the **Bode Editor**, drag the magnitude response upward. To satisfy the crossover frequency requirement, keep the response below the exclusion region in the Bode editor.




To view the compensator, right-click in the **Bode Editor** or **Root Locus Editor** plot area, and select **Edit Compensator**.




You can also tune the compensator parameters using the Compensator Editor dialog box.

## Automated Compensator Tuning

In addition to graphical tuning, you can also use automated tuning methods. To select an automated tuning method, click **Tuning Methods**.

- **PID Tuning**, **IMC Tuning**, and **LQG Synthesis** - Compute initial compensator parameters based on tuning specifications such as closed-loop time constants. See the example Design LQG Tracker Using Control System Designer.
- **Optimization-Based Tuning** - Optimize compensators using both time-domain and frequency-domain design requirements (requires Simulink Design Optimization). See the example DC Motor Controller Tuning.
- **Loop Shaping** - Specify a desired target loop shape (requires Robust Control Toolbox™).