

# Using Matlab SISOTOOL 2016 part 1

Anthony Rossiter

<http://controleducation.group.shef.ac.uk/indexwebbook.html>

# Introduction

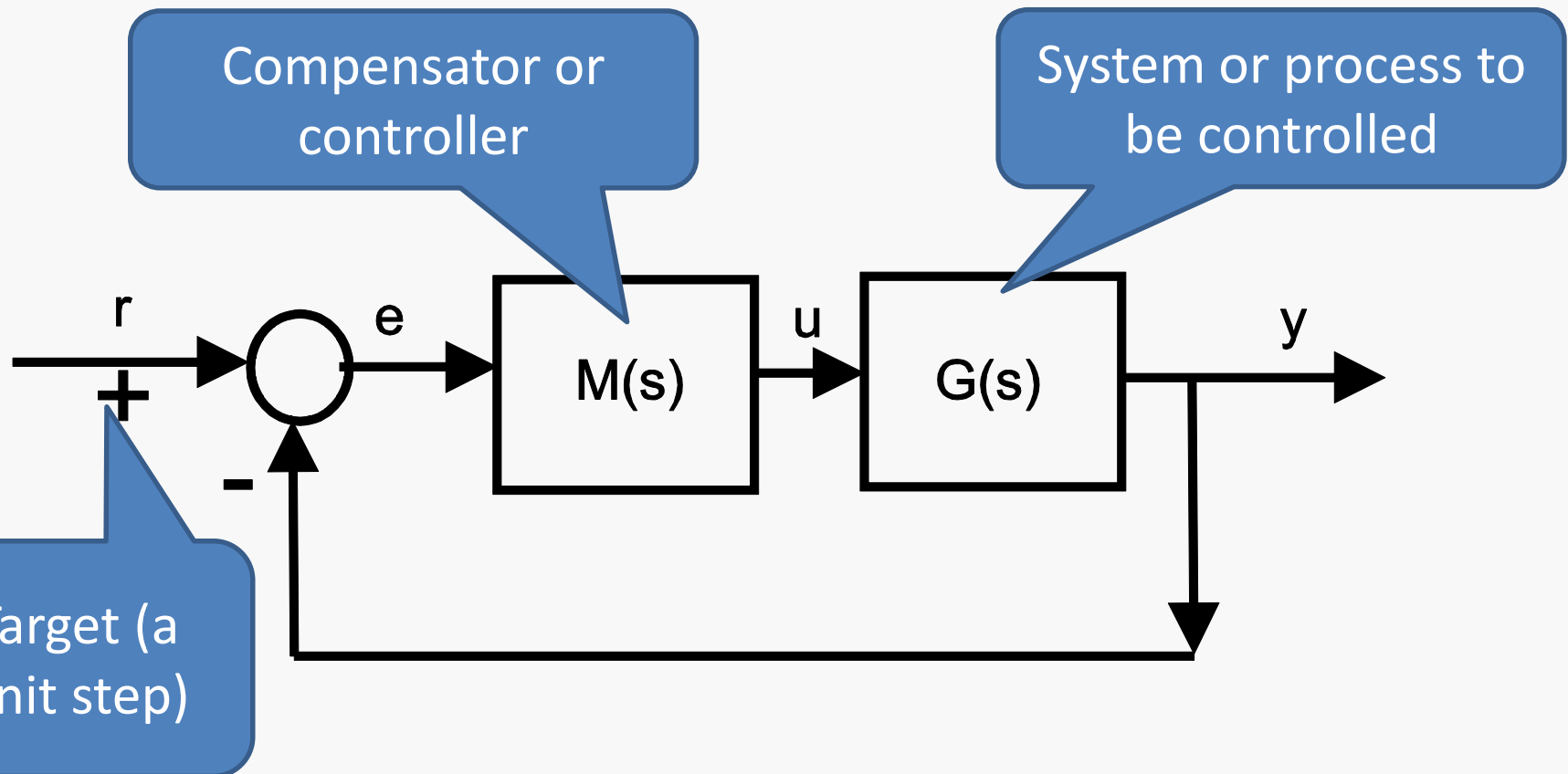
- Many earlier videos by this author use the original version of sisotool.
- However, there has been a relatively major change in the presentation and functionality of this tool in MATLAB 2016, so this resource gives a quick overview of the new tool.
- Focus is on students doing a single course in classical control methods such as Bode, Nyquist, gain and phase margins and lead and lag design.

# Core skills

- Changing compensator gain by hand (drag) and explicitly (enter a number).
- Interactive views as all plots update instantly as compensator changes are made.
- Changing compensator poles and zeros by hand (drag) and explicitly (enter a number).
- Overlaying step responses and frequency response graphs from different compensators.
- The first video focuses on views and options; the later ones deal with compensator design.

# Assumptions

This resource assumes a simple feedback structure as follows (although sisotool will deal with more complicated structures).



# Overview

First the slides outline the core skills and notation:

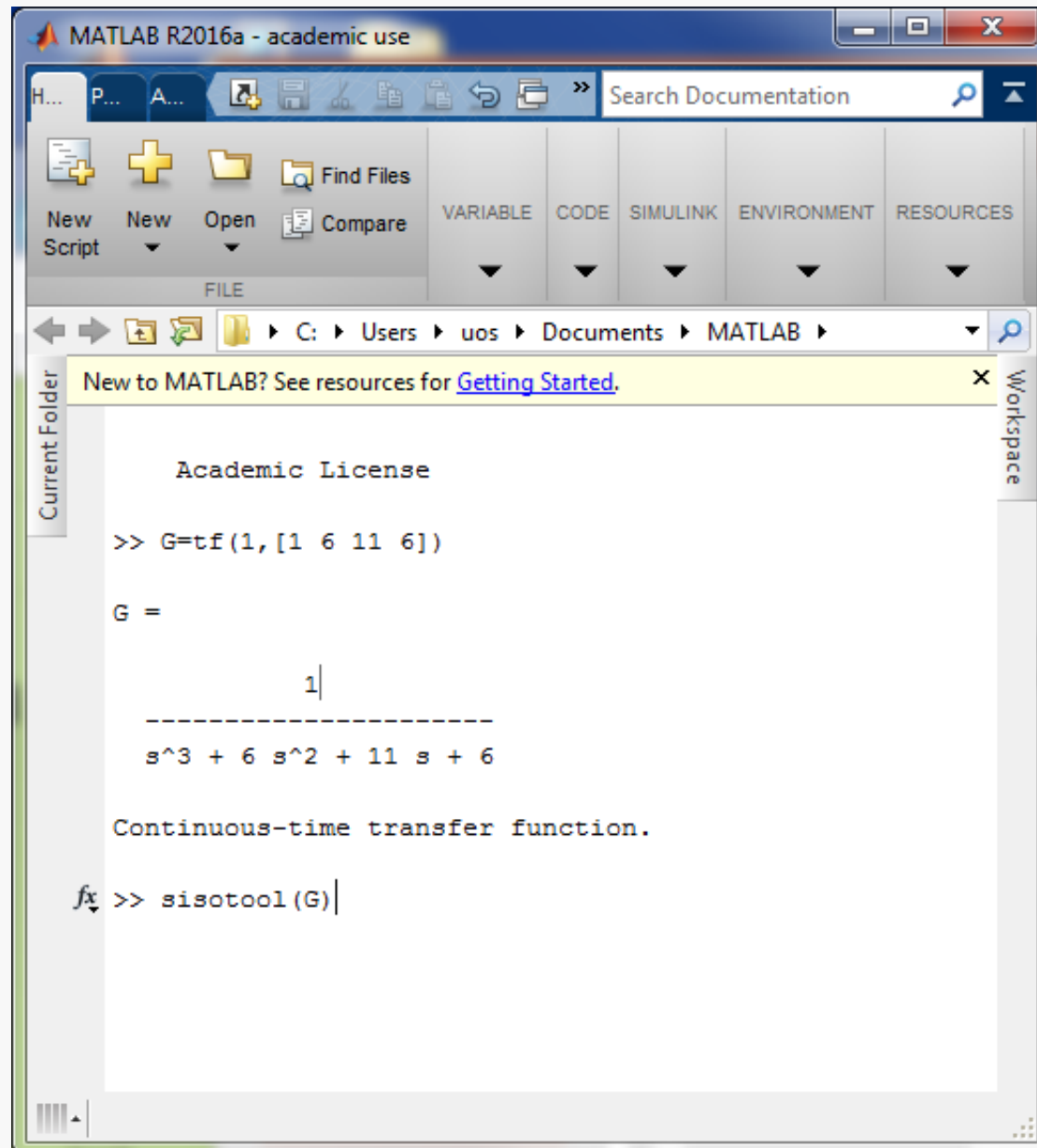
- Loop structure and naming assumptions.
- Obtaining core plots.
- Changing the compensator gain.

Secondly a live demonstration will be given on MATLAB.

# Getting started

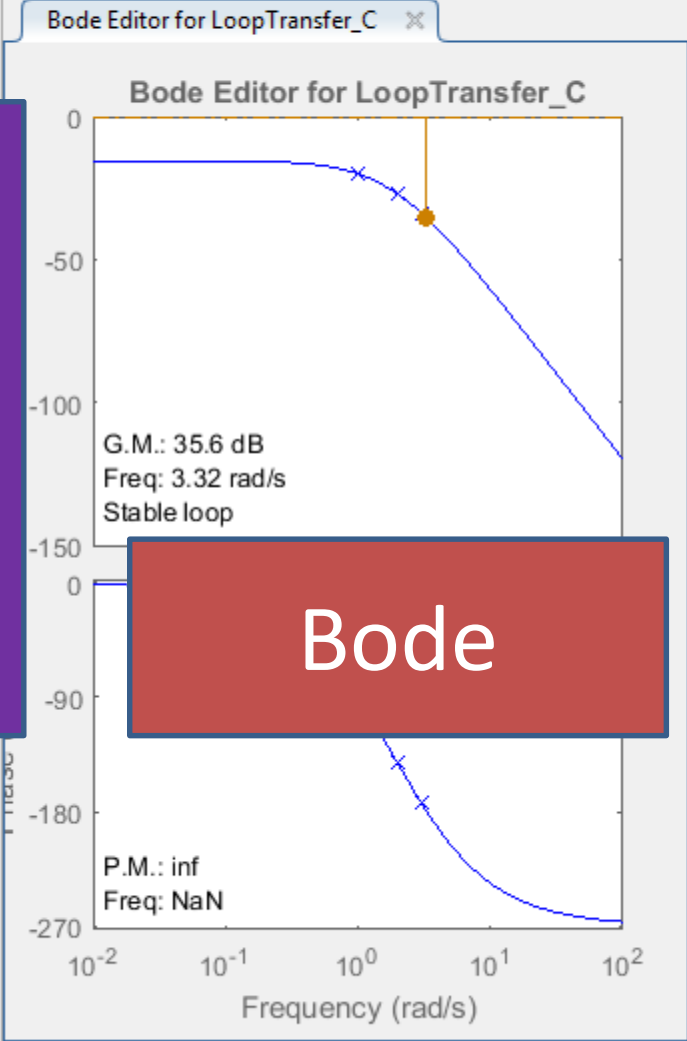
Define the system transfer function in the command window and then start sisotool as follows.

I tend to use  $G(s)$  as this is conventional and matches sisotool notation.

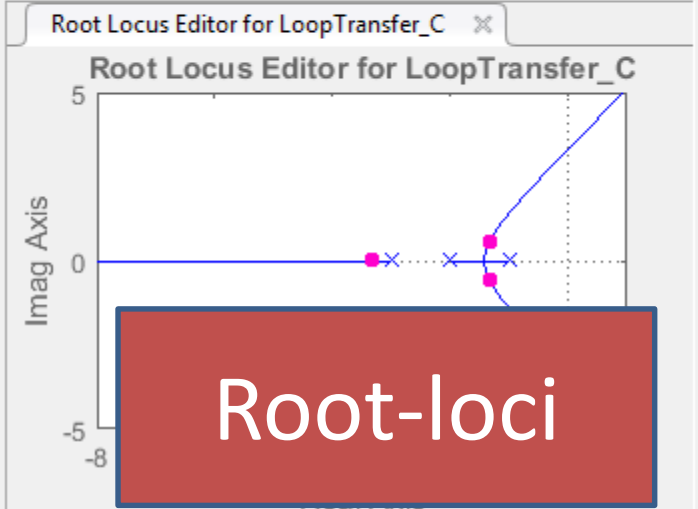


Resulting window

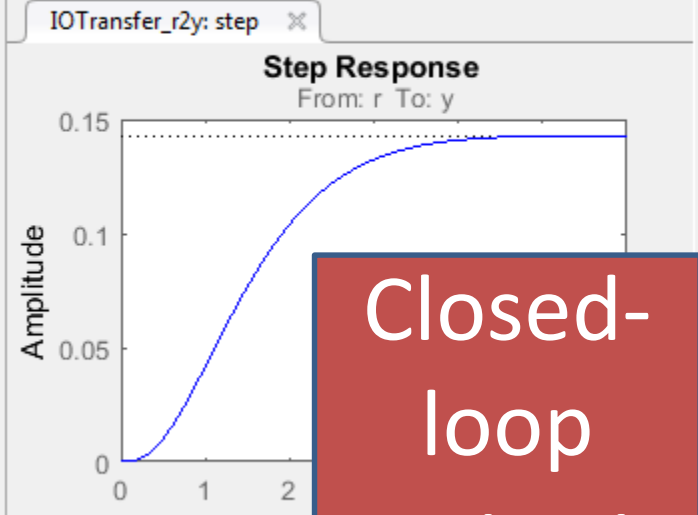
Next we will go through options and functionality



Bode



Root-loci



Closed-loop output

# Loop structure assumed in sisotool

Options are available under the architecture tab.

The screenshot shows the 'Edit Architecture' window in the Control System Designer. The main diagram illustrates a control loop with blocks F, C, G, and H. Callouts explain the components and assumptions. A table at the bottom lists the blocks and their values.

**Compensator or controller** (points to block C)

**System or process to be controlled** (points to block G)

**Negative feedback** (points to the feedback path)

**We assume  $F=H=1$**  (points to the table)

**Other possibilities** (points to the architecture selection area)

**Change G or C from workspace** (points to the table)

Identifier	Block Name	Value
C	C	<1x1 zpk>
F	F	<1x1 zpk>
G	G	<1x1 tf>



# Viewing closed-loop input responses

These are available under the tab 'new plot'

The screenshot shows the 'New Step to plot' dialog box in the MATLAB/Simulink environment. The 'Select Response to Plot' dropdown is set to 'IOTransfer\_du2y'. The 'Input-Output Transfer Function' section shows 'Name: IOTransfer\_du2y', 'Inputs: du', and 'Outputs: y'. A blue callout bubble points to this section with the text: 'Options are selected here with obvious notation.' Below the dialog is a block diagram of a closed-loop system with blocks F, C, G, and H, and signals r, e, uC, u, y, n, and ym.

Select New step

We want:  
IOTransfer\_r2u  
IOTransfer\_r2y

CONTROL SYSTEM VIEW

Open Session Save Session Edit Architecture Multamodel Configuration Tuning Methods

FILE ARCHITECTURE TUNING METHODS

Data Browser

- Controllers and Fixed Blocks
- Designs
- Responses
  - LoopTransfer\_C
  - IOTransfer\_r2y**
- Preview

Input-Output Transfer Function  
Name: IOTransfer\_r2y  
Inputs: r  
Outputs: y

**IOTransfer\_r2u**

**IOTransfer\_r2y**

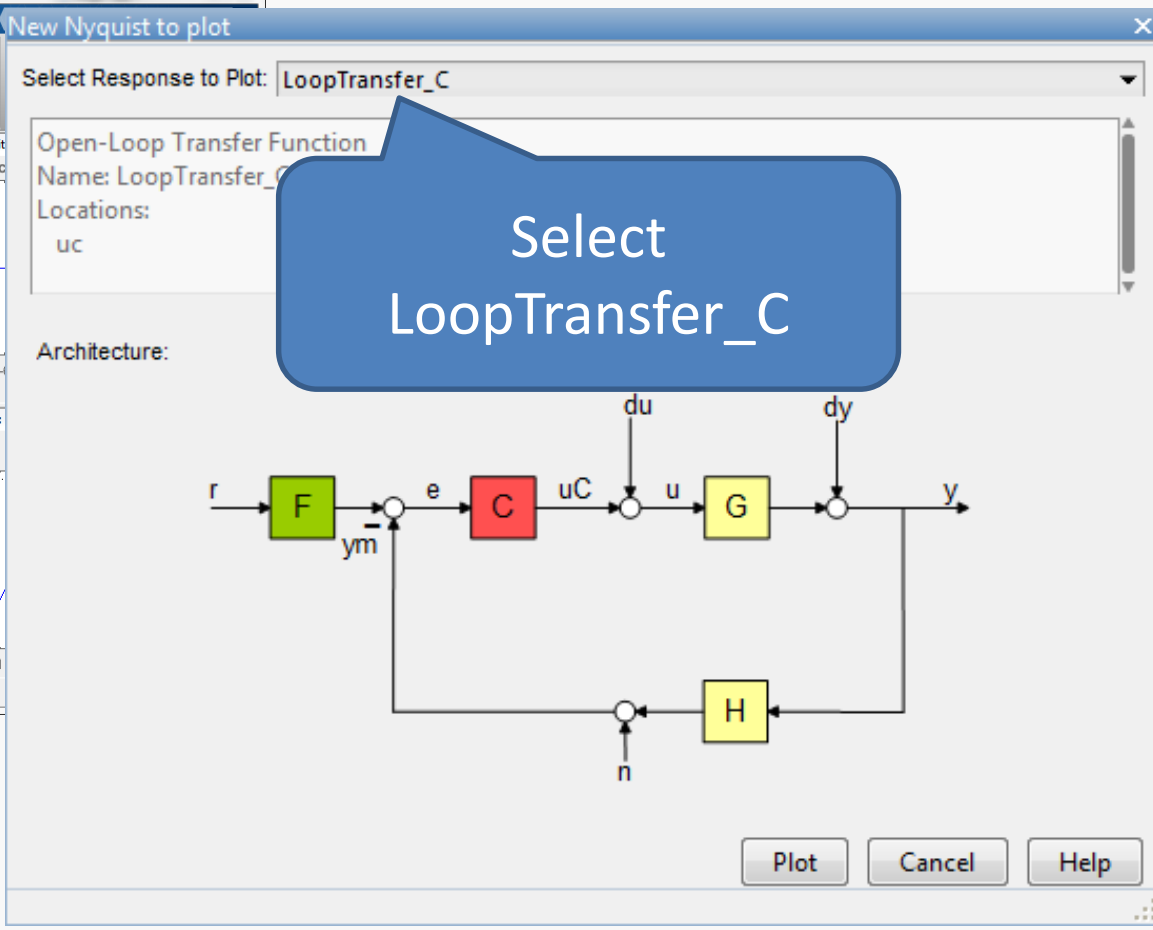
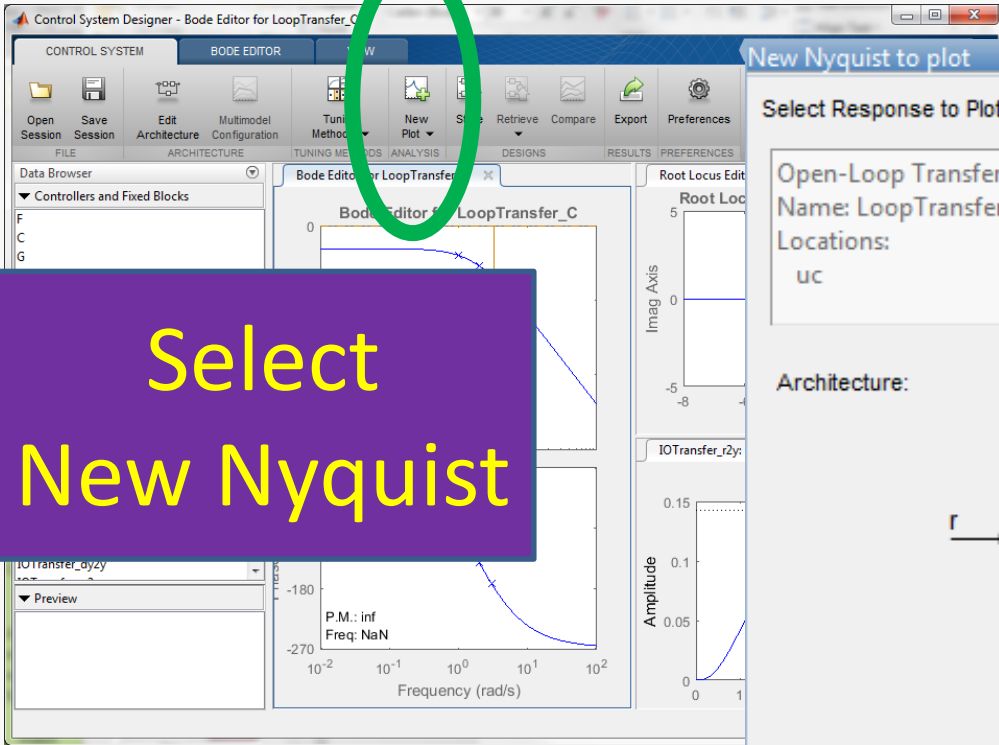
Bode Editor for LoopTransfer\_C

Step Response (From: r To: u)

Step Response (From: r To: y)

# Viewing Nyquist plots

These are available under the tab 'new plot'



Control System Designer - LoopTransfer\_C: nyquist

CONTROL SYSTEM VIEW

FILE ARCHITECTURE TUNING METHODS ANALYSIS DESIGNS RESULTS PREFERENCES

Open Session Save Session Edit Architecture Multamodel Configuration Tuning Methods New Plot Store Retrieve Compare Export Preferences

Data Browser

Controllers and Fixed Blocks

Designs

Responses

- LoopTransfer\_C
- IOTransfer\_r2y
- IOTransfer\_r2u
- IOTransfer\_du2y
- IOTransfer\_dy2y**
- IOTransfer\_n2y

Preview

Input-Output Transfer Function  
Name: IOTransfer\_dy2y  
Inputs: dy  
Outputs: y

Bode Editor for LoopTransfer\_C

Step Response

Nyquist Diagram

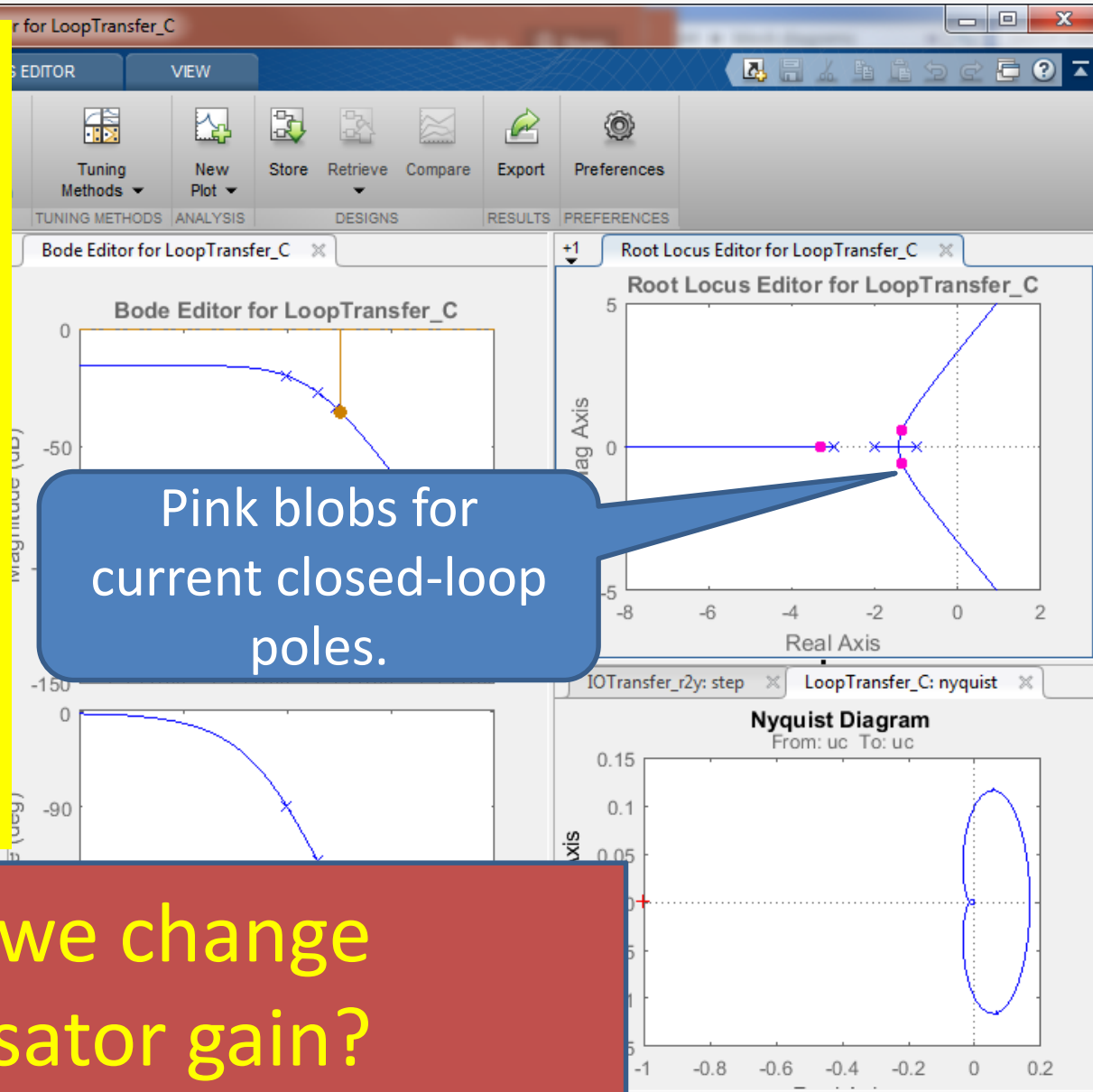
Notice tabs with all optional plots available

Nyquist diagram here

The screenshot displays the MATLAB Control System Designer interface. The main workspace is divided into several panes. On the left, the 'Data Browser' pane shows a tree view of the system components, including 'Controllers and Fixed Blocks', 'Designs', and 'Responses'. The 'Responses' pane is expanded, showing a list of transfer functions, with 'IOTransfer\_dy2y' selected. Below this, the 'Preview' pane shows the input-output transfer function details. The central workspace contains three plots: a Bode Editor for 'LoopTransfer\_C' showing magnitude and phase plots, a Step Response plot for 'IOTransfer\_r2u: step' showing amplitude vs. time, and a Nyquist Diagram for 'LoopTransfer\_C: nyquist' showing the imaginary axis vs. real axis. A purple callout box points to the tabs at the top of the plots, stating 'Notice tabs with all optional plots available'. A blue callout box points to the Nyquist Diagram, stating 'Nyquist diagram here'. The status bar at the bottom indicates 'Stable loop' and 'Freq: 3.32 rad/s'.

# Root-loci plot

The root-loci plot marks the position of the closed-loop poles with the current value of compensation.

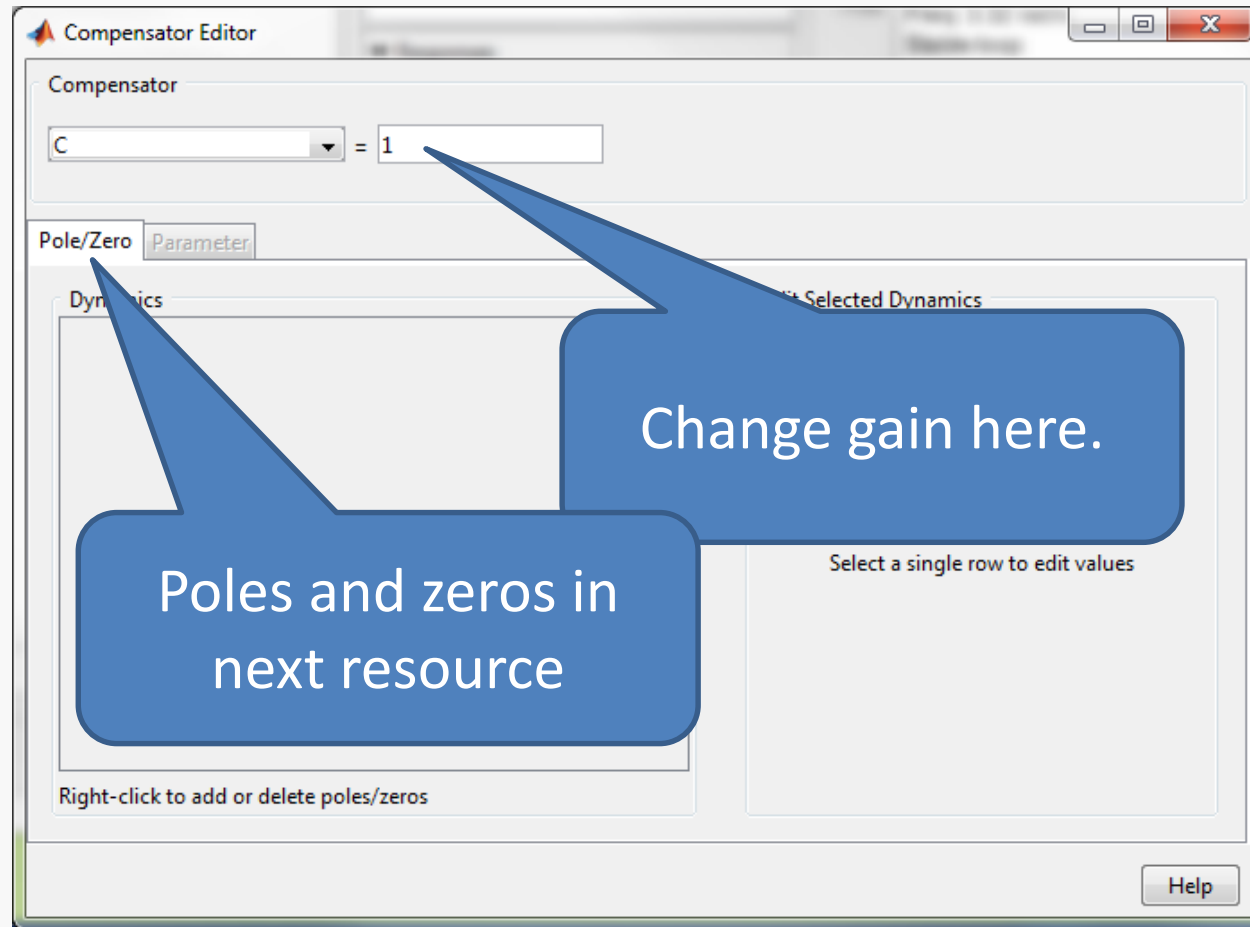


Pink blobs for current closed-loop poles.

How do we change compensator gain?

# Compensator gain

Right click on the Bode diagram and select 'edit compensator'. The following appears.



# Live demonstration

- Opening sisotool
- Adding closed-loop inputs
- Adding a Nyquist plot
- Changing the compensator gain.
- Importing a different  $G(s)$ .

# Conclusion

This short video has demonstrated the basic use of sisotool in MATLAB 2016.

We have not covered every option, but rather focussed on core functionality to support students doing a 1<sup>st</sup> course in classical control.

The following videos show how to use the tool for comparing compensator designs and then doing design.



<http://controleducation.group.shef.ac.uk/indexwebbook.html>



Anthony Rossiter  
Department of Automatic Control and  
Systems Engineering  
University of Sheffield  
[www.shef.ac.uk/acse](http://www.shef.ac.uk/acse)

© 2016 University of Sheffield

This work is licensed under the Creative Commons Attribution 2.0 UK: England & Wales Licence. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/2.0/uk/> or send a letter to: Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.



It should be noted that some of the materials contained within this resource are subject to third party rights and any copyright notices must remain with these materials in the event of reuse or repurposing.

If there are third party images within the resource please do not remove or alter any of the copyright notices or website details shown below the image.

*(Please list details of the third party rights contained within this work.)*

*If you include your institutions logo on the cover please include reference to the fact that it is a trade mark and all copyright in that image is reserved.)*