

Interactive tool for SISO Linear Control System Design

Erivelton Gualter dos Santos
Department of Mechanical Engineering
Cleveland State University

Abstract

GNU Octave is a high-level programming language for numerical computations which uses m-script language compatible with MATLAB. It was not created to be a free version of Matlab, however, it takes advantage of the well-known m-script languages and functions. Both Matlab and Octave presents many similar packages (also known as toolbox in matlab). One of the missing functions in this package is the [Control System Designer tool](#). This tool, also called siso tool, is used in Control System, Linear System, Modern Control class and many others. This tool has been used in courses related to control theory to teach students how to design linear controllers by interactive tools. Therefore, the students can add the controller to the plant interactively and configure it, such as, adding poles and zeros in the root locus diagram, or even add common controller types: PD, PID, Lead and Lag controller for example. The goal of this proposal is to create an Interactive Tool for SISO Linear Control System Design which is a missing feature in octave compared to Matlab and is significantly helpful for the students who are learning how to design controllers for linear plants.

Link of public application: <https://wiki.octave.org/User:Erigualter>

A An introduction

My name is Erivelton Gualter and I am a first year PhD student at [Cleveland State University](#), USA. My overall background is related to robotics applied to biomechanics. I am currently developing a robotics transfemoral prosthesis and I built a robotic wheelchair when I majored in Automation and Control Engineering at FEI University in Sao Paulo, Brazil. I am fluent in Portuguese, English and Intermediate Spanish.

It is my first time applying to the Google Summer of Code. However, I have been in touch with a couple organization: Octave and Sympy. My primary goal to participate in the gsoc is to build a long term relationship with the open source community. I want to take it serious and get involved. I have been using a lot of open source tools, and now I fell it is time to contribute for it.

I choose Octave because I have used MATLAB during my entire undergraduate program and I still use it for my PhD. However, now It is time to switch gears, so I decided to use Octave due the similarity to m scripting. Unfortunately, Octave still lacking some crucial functions which prevent some group of people to also switch gears. Therefore, I want to be part of maintainers group and I want to be able to use Octave for all my projects.

C Contact

- My time zone is UTC-4.
- I usually sleep at 2am to 8pm. So besides that I am flexible to work any time.
- Here is my nickname:
 - wiki, IRC and Savannah: **erigualter**
 - github: [EriveltonGualter](#)
- Personal Webpage: <https://eriveltongualter.github.io/>

However, I was using **dev101** in the IRC chat, which I will not longer use anymore.

E Coding Experience

Short answer of Programming Languages (in approximate order of proficiency):

- M-file scripting;
- C/C++;
- Python;
- HTML, CSS, Javascript, Pelican (Web development);
- Java.

Some illustration about what I have done:

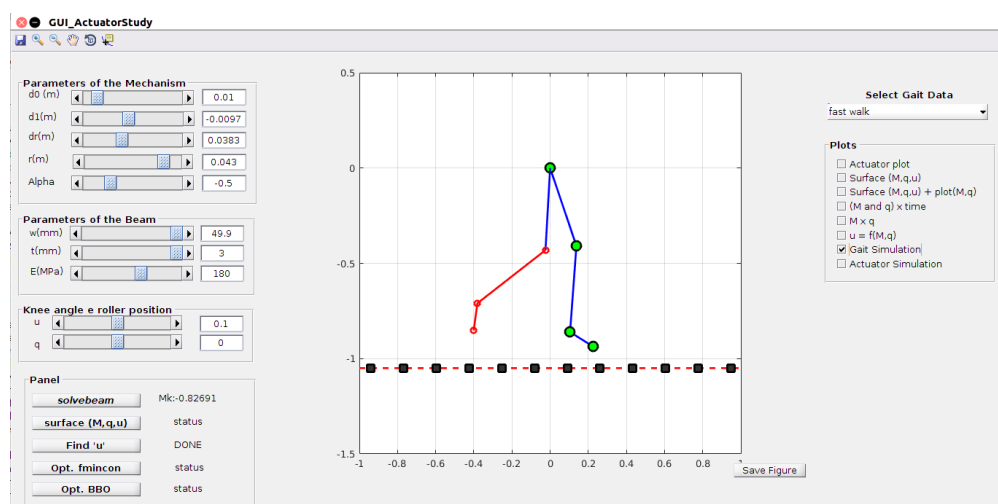


Figure 1: Simulator of gait walking with amputee subjects using prosthesis built.

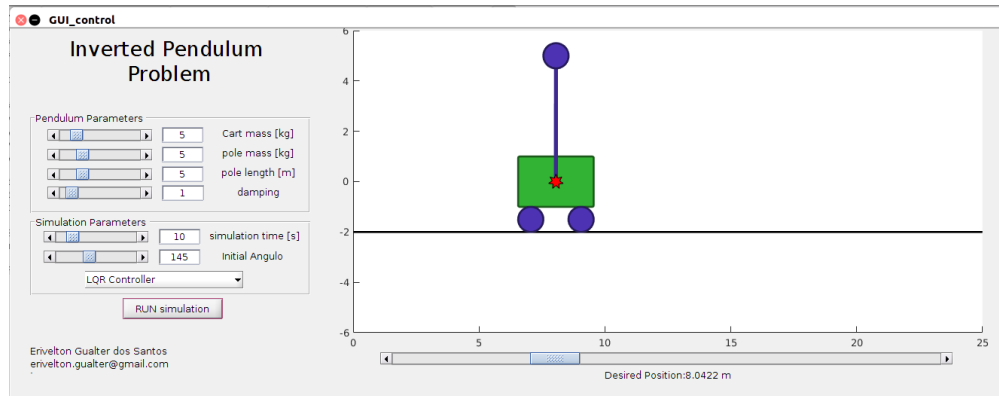


Figure 2: Inverted Pendulum Interface to simulate different parameters. Code: <https://github.com/EriveltonGualter/Inverted-Pendulum>



Figure 3: Strategy and Controller of robots to play soccer in C++: https://www.youtube.com/watch?v=L_2YBS09-YQ



Figure 4: Power-Assisted Wheelchair. Embedded Controller in C. Video: https://www.youtube.com/watch?v=rjgii2i_xP4

Deep and detailed answer about programming language experience:

- M-file scripting Octave/Matlab
 - I have been used Matlab/Octave in several classes at my undergraduate, such as: Numerical Calculus, Classical Control Systems I and II, Modern Control, Digital Signal Processing, Artificial Intelligence, and Computational Intelligence Applied to Control.
 - Also, in Online Courses at Coursera plataform: Aerial Robotics, Computational Motion Planning and Mobility <https://eriveltongualter.github.io/mooc.html>
 - Here is a link with the overall projects developed in m-scriptiong file: <https://eriveltongualter.github.io/Matlab.html>
 - I also have created some GUI (graphical user interface) in matlab:
 - * Simulation of Inverted pendulum mounted to a motorized cart, where the user can select the parameters of the system and the controller. <https://github.com/EriveltonGualter/Inverted-Pendulum>
 - * Simulation of Variable Stiffness Actuator for a Prosthesis: <https://github.com/EriveltonGualter/VSA-Simulation>
- C/C++
 - I used C and C++ in several classes: Introduction to Computer Programming, Laboratory Automation Engineering, Embedded Systems, Advanced Programming Language I and II, Data Structure.

- I joined to the Artificial Intelligence and Robotics laboratory to develop robots with the ability to play soccer. I developed part of the strategy to pass system and intercept algorithms.
- I build an Power-Assisted Wheelchair controlled by EMG signals: https://www.youtube.com/watch?v=rjgii2i_xP4 , here is the link for the full project: <https://eriveltongualter.github.io/wheelchair.html>
- Python
 - I built an Airhockey table, which I developed the scoring mechanism, upgraded airflow system, and display interface. For this task I used a Raspberry Pi as an interface. <https://eriveltongualter.github.io/airtable.html>
 - I have done some web scraping for fun.
- HTML, CSS, Javascript, Pelican (Web development):
 - I have developed some webpages, such as my personal webpage and others:
 - * <https://eriveltongualter.github.io/>
 - * <https://fei-nta.github.io/webpage/>
 - * <https://vickykras.github.io/>
- Java:
 - Just for curiosity to develop an application in Java, I created a 8 puzzle game. The initial idea was to use A* algorithm to find the fastest solution for this problem, but in the end, It just resulted in a simple 8 puzzle game <https://github.com/Eriveltongualter/Eight-Puzzle>

The biggest coding project I worked on with a big group was the period I was an undergraduate researcher at [Artificial Intelligence and Robotics Laboratory](#). We were in 6 programmers, 4 mechanicals and 3 electronic guys. For the software development, we used TortoiseSVN and then we switched to Bitbucket. Back then I was amused how efficiently everyone could collaborate to the software without messing up. Since then, I have been using github for my classes and research to be more organized.

I discovered GSoC in the end of January of this year, I have been exploring the Octave community and I performed some basic tasks, just as downloaded and build the system. Also, I started working in the following bug:

- # 51437 <https://savannah.gnu.org/bugs/?51437>

However, In the meanwhile I decided to focus in control system functionalities. So, I started to develop a prototype for a GUI sisotool.

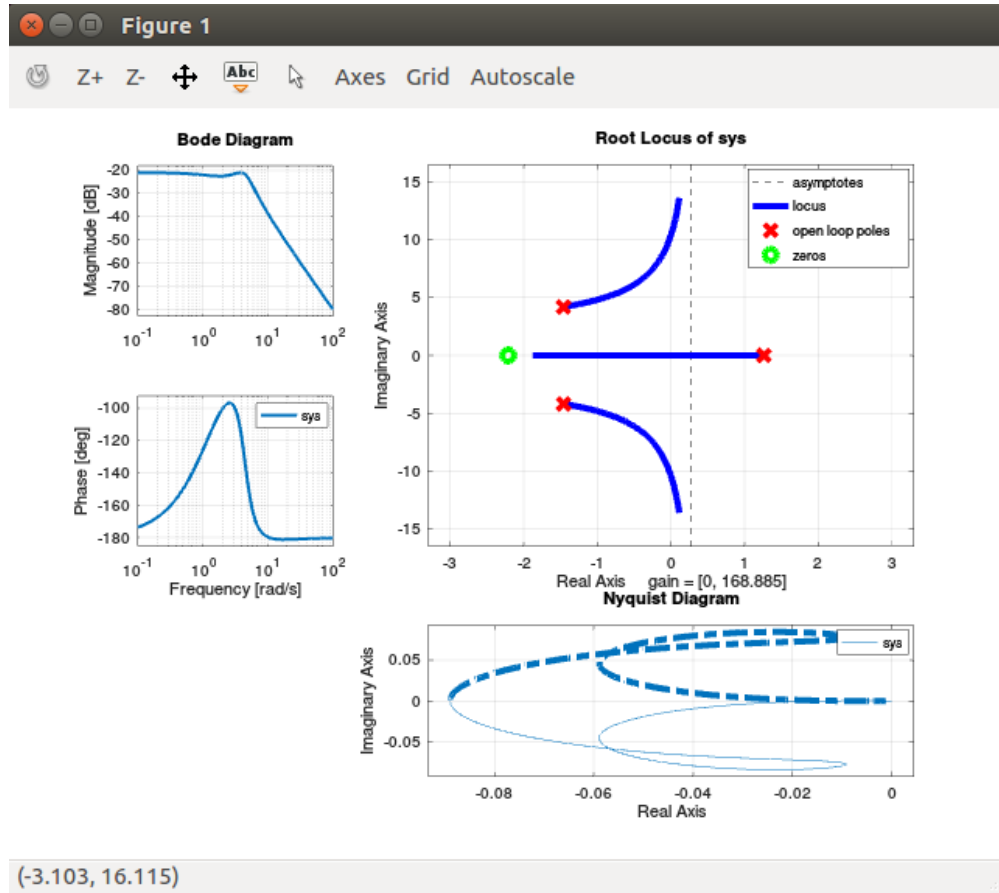


Figure 5: Concept of Interactive adjust of poles and zeros in Octave.

In this evaluation code, the user can:

- add poles and zeros by clicking with the left and right button in the mouse;
- Reallocated/Reposition by clicking and drag the pole/zero to desired position;
- To delete the pole/zero, you can click with the scroll on this.

Here is the code I have been working:

<https://github.com/EriveltonGualter/Concept-SISO-tool-for-Octave>

and videos about the performance of the interface which it was a concern. However, it has been proved promising:

- <https://youtu.be/VxosC6S3XK8>
- <https://youtu.be/YtqTC5X42jY>

Even matlab already have a sisotool interface, I created a GUI sisotool in matlab in order to practice. Here is the [repository](#). For now it contains a few elements about the compensator.

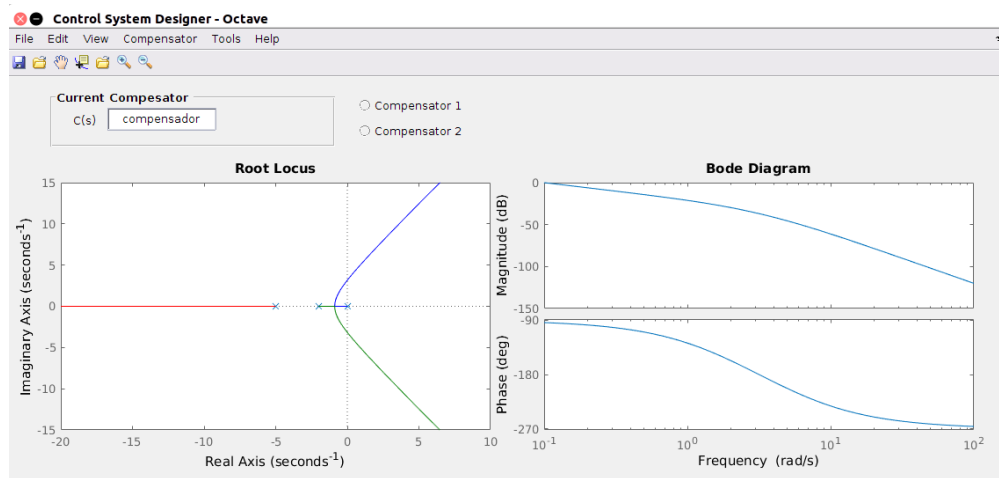


Figure 6: Concept of Interactive adjust of poles and zeros in Octave.

F Feeling fine

- “IRC and mailing list”: I have been hanging out in the IRC and following the mailing list without any problems;
- “Mercurial”: I have been using git a quite time and I used hg just in order to clone and update the Octave project. I might ask about hg later on or I am sure I can find the majority answers in hg documentation.
- “mediawiki”: I fell fine editing the wiki.
- “development tools”: Basic understand to build a GNU system, using configure-make-make install. Although I have built several programs when I was a windows user in Microsoft Visual Studio.

After the GsoC, I feel confident in stay in the community. For sure, I will be using Octave in my phd research at least for more three years, since I just started. However, my main goal is to became a maintainer of the Octave as hobby independent of my research and job.

O Only out of interest

I first heard about Octave in third year of my undergraduate program. I had a problem with my Matlab license and I needed to conclude a homework. Therefore, I found Octave by myself surfing on Google. Actually, in that time I did not feel the Octave appearance really welcoming (just noting I was a window’s user in that time). Then in the last year of college I did a dual boot Windows and Ubuntu 14.04, and I installed Octave in the Ubuntu side. However, I have been using both Octave and Matlab since I share code with my adviser, I still use Matlab a lot. (I am still trying to convince him to switch with me).

In back the time I used Octave for the first time in window's, I faced some installation issues with took a while to figure out. I am not using windows anymore, so I assume it is fine nowadays.

P Prerequisites

- I have a personal laptop with Ubuntu 16.04 and Windows 10 (but I only use windows for specific software of my research). Also, I have full access for both system.
- I basically spend my whole day in my laptop (24h).

S Self-Assessment

I belief that construct critics is essential for anyone looking to improve their abilities in certain tasks.

I am autonomous in my tasks and I consider myself very pro-activity individual. I prefer to discuss the goals desire prior to coding. After established the goals, even if we do not discussed what is the right path is to reach the goal I will be proactive and find a solution to reach this goal. Obviously with a feedback and construct criticism, the work will be concluded much faster.

Y Your Task

Interactive tool for SISO Linear Control System Design

I did not select a direct task from the list of proposal. However, I checked in the wiki and I found a project in the name of Matheus Vieira Portela called "*GUI for manipulating closed-loop systems for the control package*" from GSoC 2013 (https://wiki.octave.org/Summer_of_Code). Unfortunately, he did not finish the project. As I do not have access to his proposal, I tracked past conversations in the mailing list and talked to the IRC fellows, such as dougstew, andy1978 and batterylow, then I decided to propose "**Interactive tool for SISO Linear Control System Design**".

Just as the Matlab has a function called sisotool to performs the the design of Single-Input, Single-Output (SISO) controllers for feedback system, my idea is to create a GUI with the same characteristics.

For this proposal, the GUI will be able of design controllers from interactive plots. For example, adding, modifying and removing poles and zeros to the root locus diagram, where both the Bode and Nichols diagram can also be visualized.

Community Bonding Period (23 April - 13 May)

I will explore the control package created by Lukas Reichlin and maintainers Alexander Wilms and Doug Stewart. My goal is to become comfortable with the structure of the following codes:

- rlocus.m;
- nyquist.m;
- bode.m.

I will discuss the project with the mentor aiming to refine this proposal goals and timeline.

The tasks, notes and decisions made during this week will be posted in the following blog which is already published: <https://eriveltongualter.github.io/gsoc2018>

Phase 1 Initial phase of coding

Week 1 (14 May - 20 May)

- Setting up.

Week 2 (21 May - 27 May)

- Implement the function `sisotool(Gp)`. This function will plot the Root Locus, Nyquist and Bode diagram in the same plot;
- The goal at this point is to create the layout of the plots.
- The distribution in the plots will be fixed at this point. However, in Phase 2 the function will have attributes to choose the desired diagrams in the interface. Example: `sisotool(Gb, 'rlocus', 'nyquist')`;

Week 3 (28 May - 3 June)

- Add polos and zeros, being real or complex, to the plant through the UI elements.
- Position polos and zeros according to the mouse input.
- Polos and zeros can be reallocated by dragging with the left button of the mouse.

Week 4 & 5 (4 June - 17 June)

- Any remaining work will be completed;
- Submission of Phase 1 evaluation;
- Complete post about the phase with detailed documentation.

Phase 2

Week 6 (18 June - 24 June)

- Create the layout of the complete GUI interface with buttons, sliders, and others, examples:
 - *edit box* for the compensator gain;
 - poles and zeros *buttons*;
 - *listboxes* for the desired plots to be shown.

Week 7 & 8(25 June - 8 July)

- Add functionalities for the graphical elements from week 6.

Week 9 (9 July - 15 July)

- Submission of Phase 2 evaluation;
- Complete post about this phase with detailed documentation.

Phase 3

Week 10 & 11 (16 July - 29 July)

- Add Step Response Time functionality. The user can select the step response to visualize the characteristics of the output, such as: Rise time and Peak Overshoot.
- Export compensator design to the workspace.

Week 12 (30 July - 5 August)

- Any remaining work will be completed.

Week 13 (6 August - 14 August)

- Complete post about the phase with detailed documentation.
- Final Submission

Extra Comments

- Root Locus Controller Design using Matlab 'sisotool' Toolbox [Notes](#).
- Mentor: Doug Stewart.
- Resume: https://eriveltongualter.github.io/pdfs/Erivelton_CV_EN.pdf
- Linkedin: <https://www.linkedin.com/in/eriveltongualter/>

References

- Control System Designer Matlab: <https://www.mathworks.com/help/control/ref/controlsystemdesigner-app.html>
- Interactive plots: https://wiki.octave.org/Interactive_plots
- GUI Development - Octave <https://octave.org/doc/v4.2.0/UI-Elements.html>
- Discussion about sisotool in mailing list:
 - <http://octave.1599824.n4.nabble.com/closed-loop-control-GSoC-Design-Question-tt4653528.html#a4653528>
 - <http://octave.1599824.n4.nabble.com/GSoc-sisotool-project-tt4663204.html>