

# ENAS 773 MENG/EENG 443

## Fundamentals of Robot Modeling and Control

### Useful References

Fall 2023

There are a variety of skills that one might find very useful in robotics that span beyond what is taught in your engineering curriculum. This document provides a list of useful reference that outline some skills you will need to learn outside the classroom. Briefly going through this list before class starts will help to provide some basic coding and mathematical skills that are assumed known or can at least be picked up during the course.

### Command Prompts, Terminals, and Version Control

Having the ability to navigate a terminal is key to controlling robotic systems. Many robots are interfaced and programmed through the terminal interface (or command prompt if you are on Windows). In addition, many existing version control software that manages your code and different versions of your code are key to being organized when writing robotic software. MIT has a fantastic online course that teaches you all the basics. Note that for this course, none of these skills are necessarily required, but for those continuing into robotics this is definitely a skill worth having.

Link to course: <https://missing.csail.mit.edu/>

### Best Coding Practices, Colaboratory, and Jupyter Notebooks

Getting the most out of this course is in part is your ability code, debug, and use the tools you are given most effectively. We will be using [Google Colaboratory](#) which is based on [Jupyter Lab](#) that lets you debug code in a contained Python environment. This course will provide the necessary packages for which to install on Google Colab so your work is self-contained. In addition to learning these tools, it is worth taking a look at good coding practices. NASA has some coding principles that are worth using in your own writing (you do not have to write in C-Code in this course). Below are links to videos and tutorials for setting up Google Colab and Jupyter (which you do not need to install but worth looking at if you are using Colab). This course will be using Numpy and Jax to perform algebraic manipulation and basic derivatives when writing code.

Link to NASA Space-Proof Code: [NASA Space Proof Code](#) Links to Colab/Jupyter/Numpy/Jax Tutorial:

- [Jupyter Notebook Tutorial](#)
- [Intro to Colab and Python](#)

- [Numpy Tutorial](#)
- [Jax Tutorial](#)
- [Jax AutoDiff Tutorial](#)

## Linear Algebra and Matrix Calculus

Linear algebra and matrix manipulation (and differentiation) are core components when modeling and controlling robotic systems. They are often taught separately and often not even part of some engineering curriculum! However, at this stage in your academic careers, these skills are highly important to pick up and strengthen in order to quickly grasp complex concepts in robotics. Below are a few links to resources I have collected in the past that are quite useful. Using these as references will help improve your overall experience in this course.

Links to Linear Algebra:

- [Linear Algebra Done Right](#)
- [Vectors and Vector Spaces](#)

Links to matrix calculus:

- [Matrix Calculus](#)
- [Matrix Differentiation](#)
- [Matrix Derivatives Cheat Sheet](#)
- [The Matrix Cookbook](#)

## Basics in Optimization

Optimization, and the ability to pose optimization problems over continuous variables is at the core of robotics. Many of the problems and state-of-the-art solutions in robotics start by posing and solving a well-defined optimization problem. In this course you will be introduced to some basics in optimization, but having some familiarity of terminology before starting the course will provide a good preparation.

Link to optimization book: [Convex Optimization](#)