Optimal Control for Robotics - Course Syllabus

Tufts University - Mechanical Engineering

Spring 2018

Personnel and Office Hours

Instructor: Matthew Kelly
Email: `Matthew.Kelly@tufts.edu`
Course Website: `https://github.com/MatthewPeterKelly/ME149_Spring2018`
Office Hours:

- 30 minutes before each class and up to 60 minutes following each class
- students are encouraged to attend at least 20 minutes of office hours per week

Course Information

- *Description:* Students taking this course will learn the basics of optimal control for robotics applications. There will be a strong focus on trajectory optimization and trajectory-tracking controllers, and assignments will focus on applications and implementation, rather than on theory. In the first part of the course students will learn the core concepts of optimal control: programming, simulation, control, and optimization. In the second part of the course the students will put these concepts together to design and stabilize reference trajectories in simulation. The topics covered in this course are used in a variety of applications, including aircraft, satellites, robot arms, legged robots, quad-rotor helicopters.
- *Goals:* Upon completing this course, students should:
  - understand the major concepts in optimal control, especially trajectory optimization
  - develop strong programming skills in Matlab
  - be able to implement their own trajectory optimization, design a trajectory-stabilizing controller, and simulate a closed-loop dynamic system as it tracks a trajectory.
- *Recommended:* ME80 or ME180

General Policies

- *Scheduling Conflicts:* If you have a conflict with the midterm exam, final project, or a homework assignment, then you must notify the instructure at least two weeks in advance.
- *Honor Policy:* Your work should be your own. Cite any help that you receive or resources that you use. Do not directly copy code from other students or elsewhere.
- *Style:* All work must be clear and legible. Code should be well documented and follow the style guide. All plots should be fully annotated with axis labels, title, and a legend if appropriate.

Course Grading

- **Homework assignments:** 40%
- **Midterm exam:** 25%
- **Final project:** 35%

Homework Policies

Homework schedule: (usually - see schedule for details)

- Assigned weekly on Tuesday in class
- Due on the following Wednesday at midnight (eight days later)
- Solutions posted on Thursday and briefly covered in class
- Graded and returned on the following Monday (five day turn-around)

Homework grading:

- Each assignment is worth 50 points
- The assignment with the lowest score will be dropped from final grade
- You will lose points if your work is not clear, even if your answer is correct
- Late Homework:
  - Up to 12 hours late: -5 points
  - Up to two weeks late: -20 points
  - By end of the course: -40 points

Team work:

- All assignments must be submitted individually
- Students are encouraged to help each other work through assignments
- Directly copying code from other students is prohibited
- In the write-up for each assignment (including the final project), you must list the names of any other students that you worked with, as well as any external resources that you used.

Useful Resources

Matlab Style guide: (`.pdf`)
`https://www.mathworks.com/matlabcentral/fileexchange/46056-matlab-style-guidelines-2-0`

Textbook: (primary text for the course)
Practical Methods for Optimal Control and Estimation Using Nonlinear Programming
- John T. Betts, SIAM, Second Edition.

Textbook: (additional reading)
Applied Optimal Control: Optimization, Estimation, and Control
- Arthur E. Bryson Jr. and Yu-Chi Ho

Textbook: (additional reading)
Numerical Recipes in C
- William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery

Online Course: (60% overlap with this course)
MIT Open Courseware: Underactuated Robotics.
- Russ Tedrake (and others), two versions, both are good.

Lectures

(table on following page)

Date	**HW**	**Topic**
R - Jan 18	1 →	Intro to optimal control: major concepts and applications. Intro to Matlab.
T - Jan 23	2 →	Intro to simulation: Euler's method and error analysis.
R - Jan 25		Live Matlab: simulation with Euler's method and Heun's method.
T - Jan 30	3 →	Stability analysis and Runge–Kutta methods methods.
R - Feb 1		Introduction to SISO control.
T - Feb 6	4 →	Trajectory-tracking control for SISO systems
R - Feb 8		Control basics for MIMO systems: state space, linear control; 3 Dof planar arm simulation
T - Feb 13	5 →	Trajectory-tracking for MIMO systems; Working overview of LQR; underactuated systems
R - Feb 15		Scalar optimization and root-finding part one
T - Feb 20	6 →	Scalar optimization and root-finding part two
R - Feb 22		**No Class – Monday schedule on Thursday**
T - Feb 27	7 →	Single Shooting - part one
R - Mar 1		Single Shooting - part two
T - Mar 6		Multiple Shooting - part one
R - Mar 8	R →	Multiple Shooting - part two
T - Mar 13		Review for Midterm Exam
R - Mar 15	8 →	In-class Midterm Exam
T - Mar 20		**No Class – Spring Recess**
R - Mar 22		**No Class – Spring Recess**
T - Mar 27	9 →	Direct Collocation - Trapezoid Method
R - Mar 29		Direct Collocation - Hermite–Simpson
T - Apr 3	10 →	Direct Collocation - advanced topics
R - Apr 5	F →	Final Project Overview
T - Apr 10		TBD (analytic gradients?)
R - Apr 12		TBD (pseudo-spectral methods?)
T - Apr 17		TBD (gues lecture?)
R - Apr 19		TBD (adaptive meshing?)
T - Apr 24		Trajectory optimization in the real world; In-class office hours for final project
R - Apr 26		In-class presentations and discussion
T - May 1		**No Class – Reading Period**