ENAS 773 MENG/EENG 443
Fundamentals of Robot Modeling and Control

Fall 2023

Description: This course introduces fundamental concepts for modeling and controlling robotic systems. The course is divided into two components: Part 1 introduces mathematical tools for modeling and simulating complex robot dynamics. Part 2 formulates various ways to control robots through comprehensive analysis of dynamics and a deep dive into control theory. Specific lecture topics will cover an introduction to variational calculus, state representation, kinematics and dynamics, manipulator equations, contact dynamics and collision detection, observability and controllability, control of fully actuated and underactuated robots, model-based methods for control, and control for manipulation and locomotion.

The course will focus on connecting mathematical topics with concrete algorithmic implementation where the mid-term project assignment has students model the dynamics of a robot of their choosing. Coding assignments throughout the semester will provide experience setting up and interfacing with URDFs, automatic differentiation math libraries in python, and algorithmic implementation of state-of-the-art robot control methods. Students taking this class will finish with a codebase and foundational knowledge for simulating and controlling general robotic systems. Special topic lectures will focus on recent developments in the field of robotics and highlight core research areas. A final class project will take place instead of a final exam where students will leverage the mid-term robot simulation to control a robot to perform a task of their choosing.

Meeting Time: TTh 4-5:25pm at HLH17 room 115

Office Hours: TBD

Prerequisites: The course is designed for incoming graduate students (and advanced undergraduates). Experience with differential equations, extensive linear algebra and numerical methods for solving ordinary differential equations skills is required. Functional and object-oriented coding experience in e.g., python, C/C++, is also required.

Access to a computer is required, but no special software is needed. Online Colab notebooks will be used to ensure consistency in code-base and that software packages are maintained across machines.

Instructors

Prof. Ian Abraham	**Email:** `ian.abraham@yale.edu`
TF: Yilang Liu	**Email:** `yilang.liu@yale.edu`
ULA: Chris Ward	**Email:** `chris.ward@yale.edu`

Course Structure

Learning Objectives:

1. Deep understanding of variational calculus and optimal control
2. Model and simulate robotic systems with constraints and contact dynamics
3. Pose optimization problems and solve for control solutions
4. Optimize trajectories and controllers that leverage robot models to perform specified tasks
5. Overview of state-of-the-art control methods (indirect and direct methods, model-predictive control, policy optimization, LQR/iLQR)

Lectures and Resources: Occur two times a week. Lecture notes will be provided prior to the class. If possible, please bring the notes in and use them to follow along. The lecture will go over the notes and provide deeper insights on motivation and derivations as well as algorithmic implementation.

Course material will include select open-access books on robotics, dynamics, and control. Additionally, several academic papers will be recommended along with special topic lectures.

Coding Assignments: There will be a total of 10 coding assignments that build up tools necessary towards your midterm and final projects. The assignments will use Google Colab Jupyter notebooks with a dedicated package software for the course. Submission of assignments will be done through canvas as a Jupyter notebook for which TF/ULA will run your code and grade on successful runs and completion of questions.

Midterm Project: Students will leverage the concepts from the first part of the course to simulate a robot by developing their own multi-body contact dynamics simulator. The choice of robotic system is arbitrary, but must consistent of several degrees of freedom and require resolving contact or constraint dynamics. The simulator will serve as foundational code for the final project for which control will be introduced into the simulation.

Final Project: A short proposal outlining the project problem statement, goals, and measurable outcomes will be required before starting the final project. Feedback will be provided based on the project goals to ensure its viability and success by the course conclusion. Projects can range from implementation and analysis of recent research papers on a simulated robot of your choosing, to novel robot tasks and control of manipulation, locomotion, or your own research question. A final project report in the form of a short academic robotics conference paper will be required upon completion as well as a presentation (with robot videos, or live robot demos). Grading will consist of how well the problem is articulated, the challenges and assumptions are highlighted, and the thoroughness of the subsequent analysis and benchmarking.

Assessment & Grading

1. Participation 5%
2. Coding assignments 25%
3. Midterm project 25%
4. Final project 35 %

Assessment of the course serves to 1) have engaged discourse on ideas, mathematical concepts, and rigorously work through assumptions; and 2) provide hands-on coding experience that connects learned mathematical concepts to practical implementation on robots. The final project (and requisite proposal, presentation, and report) serves to prepare students to formulate proper problem statements in robotics, conduct appropriate benchmarks and evaluations, and clearly present and articulate challenges, research questions, and results.

Schedule

Week	Date	Topic	Assignments
1	Sept 5	Course Intro, Robotics and Control Systems	HW1 Out
	Sept 7	Variational Calc Intro: Norms, vector spaces, directional derivatives	
2	Sept 12	Geometry I: Rotations, twists, wrenches	HW1 Due HW2 Out
	Sept 14	Geometry II: Jacobians, forward kinematics, inverse kinematics	
3	Sept 19	Dynamics I: Euler-Lagrange Equation	HW2 Due HW3 Out
	Sept 21	Dynamics II: EL Generalization to transforms, URDFs, integration, and simulation	
4	Sept 26	Dynamics III: Constraints and Contact	HW3 Due HW4 Out
	Sept 28	Practical considerations in simulation	
5	Oct 3	Refresher on control systems: Feedback and feedforward control	HW4 Due HW5 Out
	Oct 5	Operational space control and feedback linearization	
6	Oct 10	Underactuated systems and intro to model-based control	HW5 Due Midterm Out
	Oct 12	Anatomy of an Optimal Control Problem: Direct and Indirect Approaches	
7	Oct 17	Indirect Methods: Maximum Principle, Hamilton-Jacobi-Bellman Equations	
	Oct 19	No Class	
8	Oct 24	The LQR Controller	
	Oct 26	iterative LQR and Diff. Dynamic Programing	Midterm Due HW6 Out
9	Oct 31	Direct Methods: Transcription and Nonlinear Programs	
	Nov 2	Receding Horizon Model-Predictive Control	HW6 Due HW7 Out
10	Nov 7	Examples of control for manipulation	
	Nov 9	Examples of control for locomotion	HW7 Due HW8 Out Project Out
11	Nov 14	Systems Identification & Uncertainty: Partial Observability	
	Nov 16	Estimation and Filtering	HW8 Due
12	Nov 21	No class	
	Nov 23	No class	
13	Nov 28	Advanced Topics I: Exploration and Uncertainty Reducing Control	
	Nov 30	Advanced Topics II: Objective design, practical considerations, Course Recap	
14	Dec 5	Project presentations I	Project due
	Dec 7	Project presentation II	

Course Policies

Late Assignments: Notice of late assignments must be given to the instructor and TF/ULA before the due date. Students will be provided with a budget of 3 late days which they can spend however they want before turning in the assignment. The instructor must be given notice for when and how much of the budget is to be used. Any late assignments beyond the budget will be penalized 10% of the final grade per day late.

Grading Disputes: Grading will primarily be done by TF/ULA (aside from midterm and final projects). Any student wishing to dispute the grade can do so formally in writing and sent to the instructor. The instructor will re-grade the whole assignment to ensure fairness in the grading process, but the final resulting grade (whether higher or lower) will be fixed and final.

Academic Integrity: Academic integrity is a core value of the university and will be upheld within this course by its students. While collaboration is encouraged, this does not include sharing or copying of code in any way. Each student is required to submit coding solutions independently. Project proposal, presentations, and report should be done independently. Any student wishing to collaborate in a team for a project should meet with me early on with the idea, how each member of the group will play a role, and how each project report will differ within the project.