

ECE 492/592 Algorithmic Foundations of Robotics

Spring 2027 (Unofficial) Syllabus

Course Information

Course Title: ECE 492/592 Algorithmic Foundations of Robotics

Course Instructor: Dr. Max Cohen — mhcohen2@ncsu.edu — 1226 Research Building IV

Meeting Time and Location: Tuesdays/Thursday from 3:00pm – 4:15pm in 2236 EB-III.

Course Description: This course provides an introduction to the algorithmic foundations of robotics and autonomous systems. Topics addressed include robot dynamics, feedback control, motion planning, trajectory optimization, state estimation, localization, perception, and robot learning. The course will emphasize theoretical and algorithmic foundations of autonomy and strives for breadth over depth, providing students with a “system-level” understanding of the autonomy stack found in modern autonomous systems ranging from self-driving cars to humanoid robots. Prerequisites include a solid mathematical background in linear algebra, calculus, differential equations, and probability, as well as comfort using a programming language of choice. Previous exposure to control systems would be useful, but is not strictly necessary. Learning outcomes of the course material through project-based assignments, which require implementation of algorithms on various robots in simulation.

Important Note: If you took ECE 492/592 *Introduction to Robotics and Autonomous Systems* with me in Fall 2025, please note that this is the same course with a different name to better reflect the course content and to distinguish it from other emerging courses in robotics.

Prerequisites

This is an upper-level undergraduate/lower-level graduate course that requires mathematical maturity and programming skills to be successful in. We will use the language of calculus, linear algebra, differential equations, and probability to describe how robots operate in the real world. It is important that you feel comfortable with the fundamentals from these areas. To be concrete, you should probably have taken:

- MA 242 Calculus III;
- ECE 220 Analytical Foundations of ECE;
- ECE 209 Computer Systems Programming (or equivalent programming course).

In addition, while not strictly necessary, you may find it useful to have taken a controls course and a statistics/probability course, such as:

- ECE 308 Elements of Control Systems;
- ST 371 Introduction to Probability and Distribution Theory.

Learning Outcomes

Upon completion of this course, students will be able to:

- **Analyze** the behavior of complex robotic systems via their associated dynamic models to evaluate fundamental properties such as stability.
- **Design** layered autonomy stacks by integrating feedback control, trajectory optimization, and motion planning to generate dynamically feasible trajectories for nonlinear and underactuated robotic systems.
- **Implement** probabilistic algorithms to enable a robot to estimate its state and build a map of its environment.
- **Evaluate** the trade-offs between different algorithmic approaches for modules within the autonomy stack, and **justify** the selection of a proposed architecture for a given robotic application.

Course Materials

Required Textbook: There is no required textbook for this class. Lecture notes will be provided throughout the semester that cover course content and provide references for further reading. There are various supplemental texts that may be helpful to consult from time-to-time, outlined below.

Optional Materials: This course will provide a high-level overview of various topics in robotics and autonomous systems. As such, you may find it helpful to have other references that contain more in-depth information on various topics. A few books that may help supplement the material in this course are:

- Kevin M. Lynch and Frank C. Park, “Modern Robotics: Mechanics, Planning, and Control,” Cambridge University Press, 2017 ([freely available online](#));
- Russ Tedrake, “Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying, and Manipulation,” Course Notes for MIT 6.832 ([freely available online](#));
- Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia Kavraki, and Sebastian Thrun, “Principles of Robot Motion: Theory, Algorithms, Implementations,” MIT Press, 2005;
- Steven M. LaValle, “Planning Algorithms,” Cambridge University Press, 2006;
- Sebastian Thrun, Wolfram Burgard, and Dieter Fox, “Probabilistic Robotics,” MIT Press, 2005.

Course Assessments, Grading, and Feedback

Homework Assignments (30%): Assessments that go along with weekly lecture content. Grading criteria are included within each assignment. Consists of both conceptual questions and short programming tasks. Allowed to work in small groups (2-3 students), but must write up your own answers and code.

Midterm Exam (25%): There will be a midterm exam approximately halfway through the course. Content will be similar to the conceptual questions asked on homework assignments. Midterm will take place in class and will be closed notes.

Project Report (30%): There will be a course project that requires you to implement the algorithms covered in this course to an application of choice. Projects can be done in groups and the exact requirements will depend on whether you are enrolled in the undergraduate or graduate section of the course.

Project Presentation (15%): In place of a final exam, we will hold project presentations that require you to present the motivation, methods, and results behind your project to the entire course.

List of Topics and Tentative Schedule

This course will cover the following topics in roughly the order outlined below. Two lectures during the semester will be taken up by the midterm review and the midterm exam, so this exact schedule is subject to change depending on if any lectures are cancelled due to unforeseen events.

Background and Motivation

Lecture 1: Course Introduction and Motivation

Lecture 2: Mathematical Preliminaries and Programming Review

Dynamics and Control

Lecture 3: The Configuration Space, Topology, and Differentiable Manifolds

Lecture 4: The State Space, Nonlinear Dynamics, and Numerically Solving ODEs

Lecture 5: Feedback Control, Robotic Dynamics, and Underactuated Systems

Lecture 6: Stability and Lyapunov Analysis

Lecture 7: Linear Systems, Linearization, and Domain of Attraction Analysis

Lecture 8: The Linear Quadratic Regulator

Lecture 9: Trajectory Tracking and Feedback Linearization

Lecture 10: Rigid Bodies and Rotational Kinematics

Lecture 11: Differential Flatness and Quadrotor Control

Trajectory Optimization and Motion Planning

Lecture 12: A Crash Course on Optimization

Lecture 13: Optimal Control, Dynamic Programming, and Hamilton-Jacobi-Bellman

Lecture 14: Trajectory Optimization, Discrete-time Dynamics, and Direct Transcription

Lecture 15: Model Predictive Control

Lecture 16: Control Barrier Functions

Lecture 17: Discrete Search and Planning

Lecture 18: Sampling-based Motion Planning

State Estimation

Lecture 19: State Estimation and the Nondeterministic Filter

Lecture 20: Probability Review and Bayesian Filtering

Lecture 21: The Kalman Filter, Extended Kalman Filter, and Particle Filter

Lecture 22: Localization

Lecture 23: Mapping

Lecture 24: Simultaneous Localization and Mapping

Advanced Topics

Lecture 25: Reinforcement Learning

Lecture 26: Sampling-based Optimal Control