Lecture 20
Motion Planning II

Katie DC

Modern Robotics Ch 10.2-10.5

Administrivia

- Upcoming homework due dates:
  - HW6 due 11/14 at 8pm
  - HW7 due 11/19 at 8pm
  - HW8 (bonus) due 12/10 at 8pm
    - Note that it will likely be a good review for the exam!
- Last few lectures:
  - 11/16 will be a guest lecture – attendance is required!
  - 11/18 will be a review session
  - 11/30 and 12/2 will be project presentations
- Exam 2 is on 12/7 during lecture

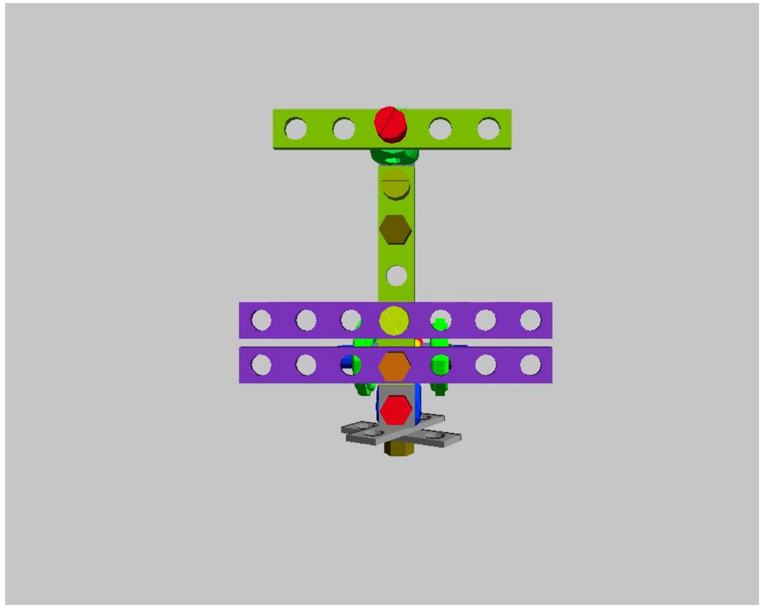
all deadlines and submission instructions are on the website

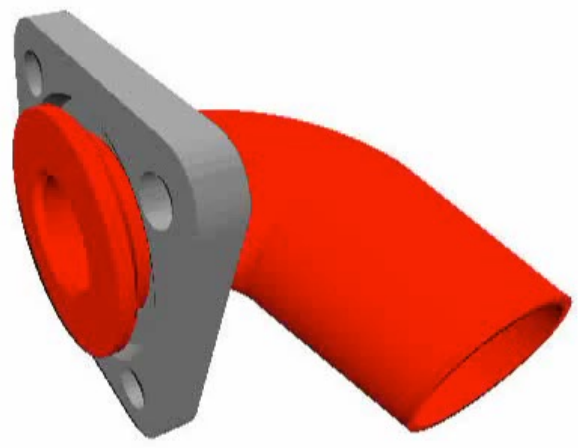
Project Video Presentations *Submit via Google Form*	11/29 at midnight
Participation Self-Assessment *Submit via Google Form*	12/8 at midnight
Participation Bonus Submissions *Submit via Google Form*	12/8 at midnight
Extra Credit Videos *Submit via Google Form*	12/15 at midnight
Team Assessment *Submit via Google Form*	12/17 at 8pm
Project Report *Submit via Gradescope*	12/17 at 8pm

Who is Nancy Amato?

- Head of the CS department and expert in motion planning
- Her paper on probabilistic planning is one of the most important papers in PRM, the first to not use uniform sampling in the configuration space
- She and her team wrote a seminal paper that shows how robot planning can be applied to protein motions (folding)

→ This line of work started a new research area in comp. biology





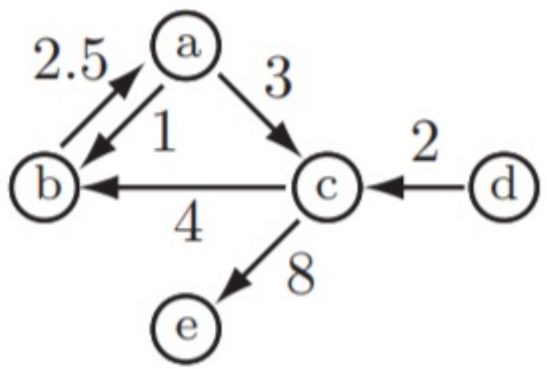
Motion Planning Overview

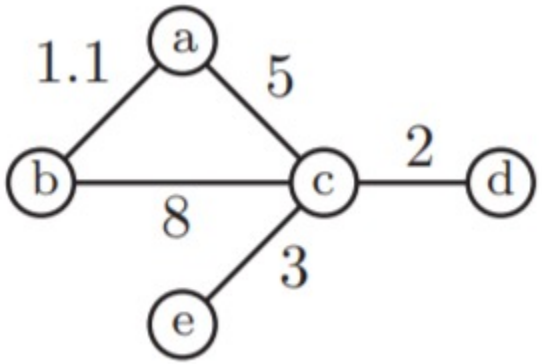
Graphs and Trees

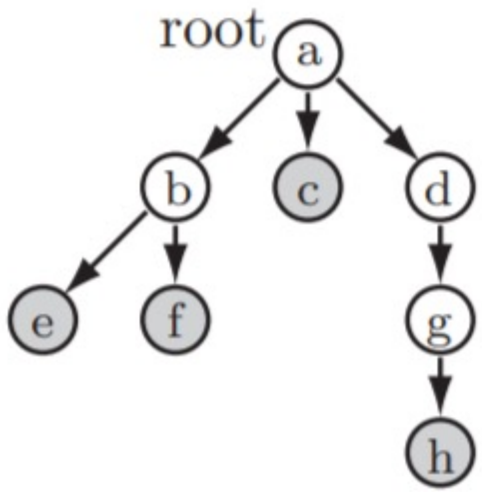
- Motion planners often represent C-space as a **graph**
- A **graph** is a collection of **nodes** $\mathcal{N}$ and **edges** $\mathcal{E}$, where edge e connects two nodes

- A **tree** is a directed graph with no cycles and each node has at least one parent

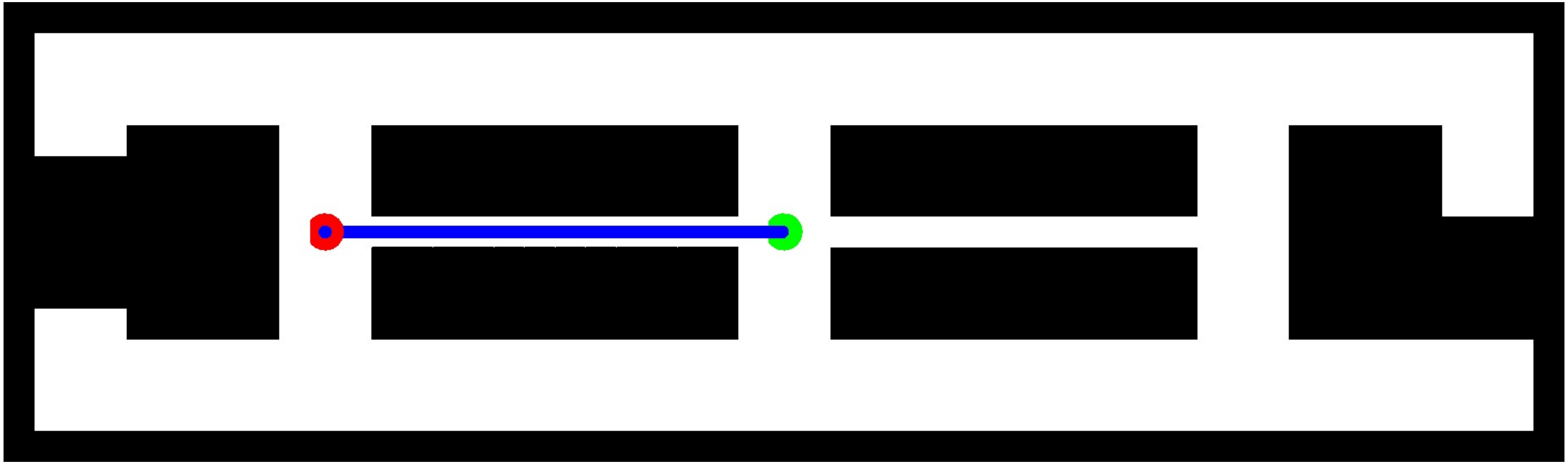
Graphs and Trees





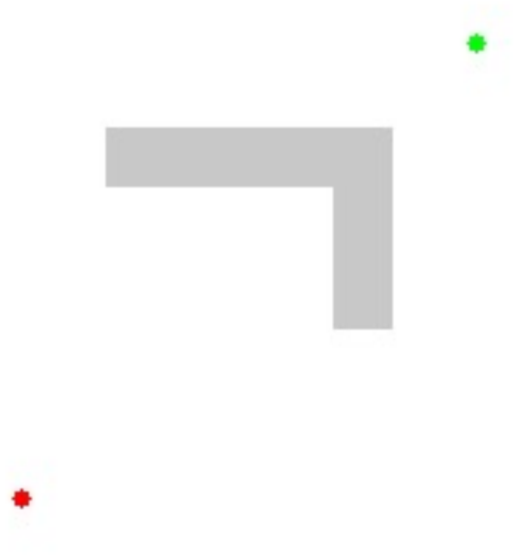


Grid-World Example

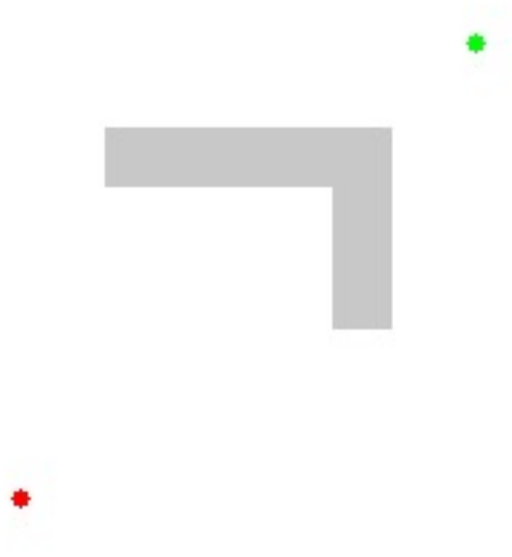


Graph Search Methods

A* search algorithm.

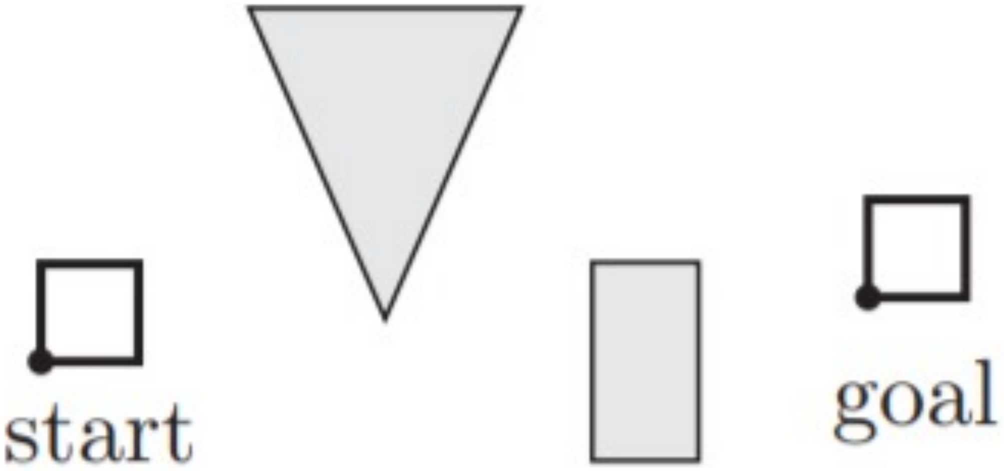


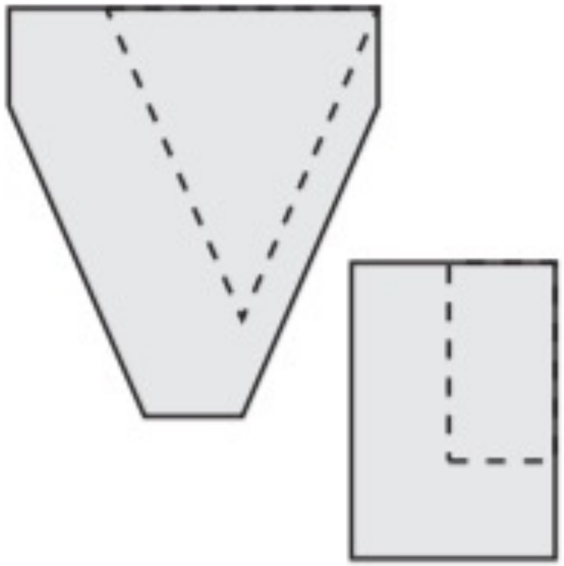
Dijkstra's algorithm.



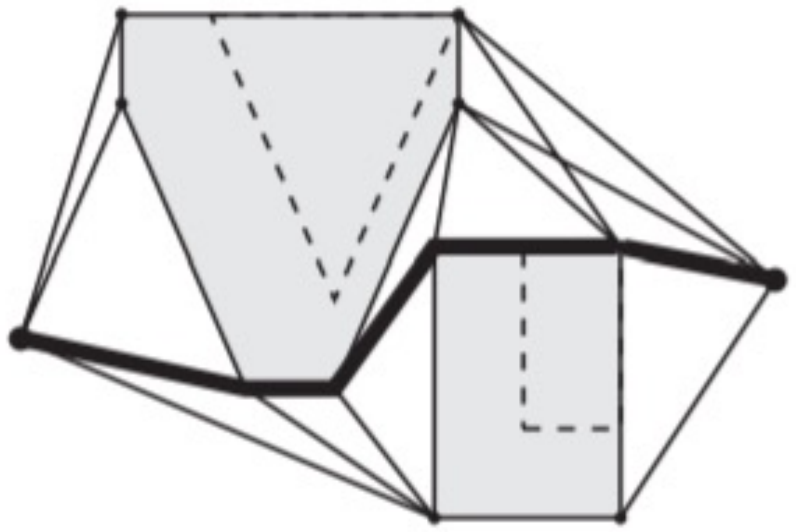
Credit: Subh83 on Wikipedia

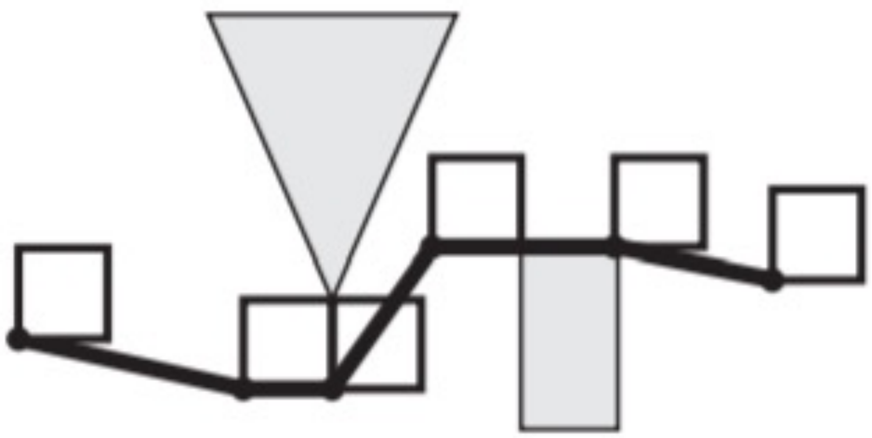
A simple roadmap: visibility graph





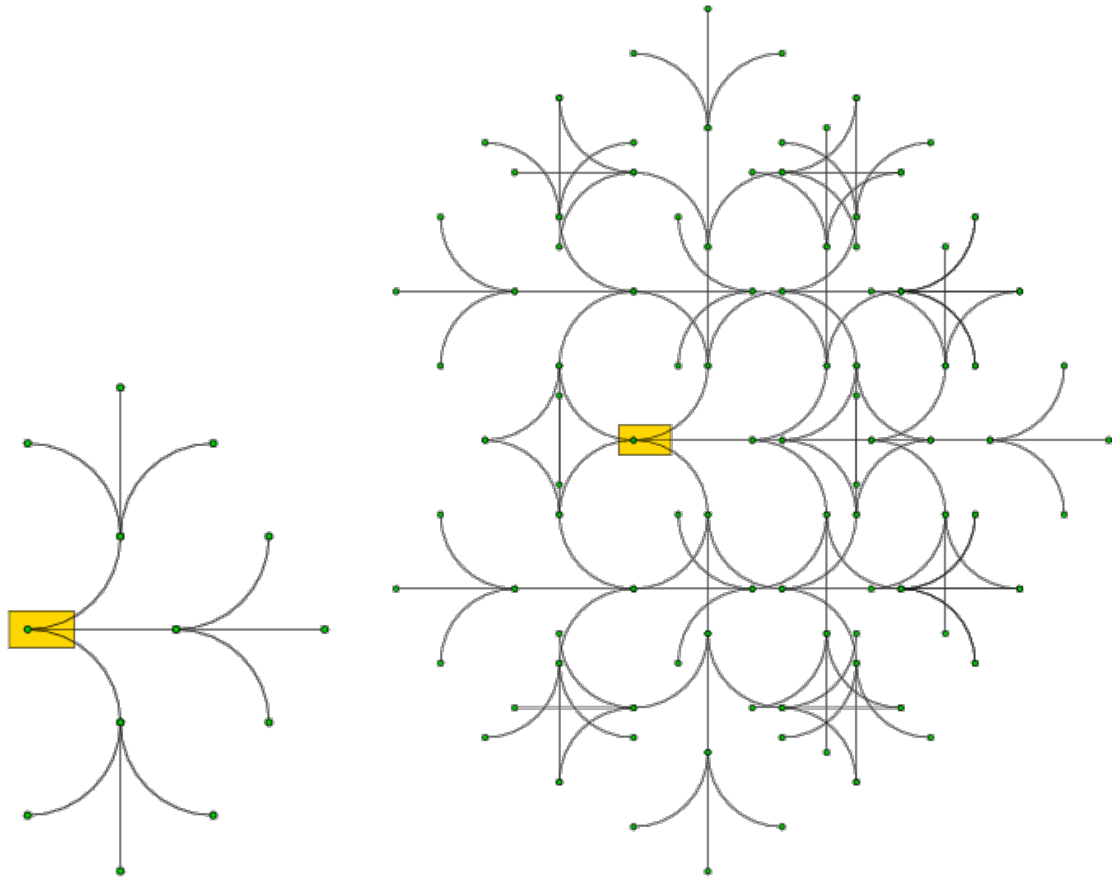
A simple roadmap: visibility graph





Sampling Based Planners: Probabilistic Roadmaps

Reachability Tree for Dubin's Car



Two stages   Four stages

Credit: Steven LaValle, Planning Algorithms

Rapidly Exploring Random Trees (RRT)

Algorithm 10.3 RRT algorithm.

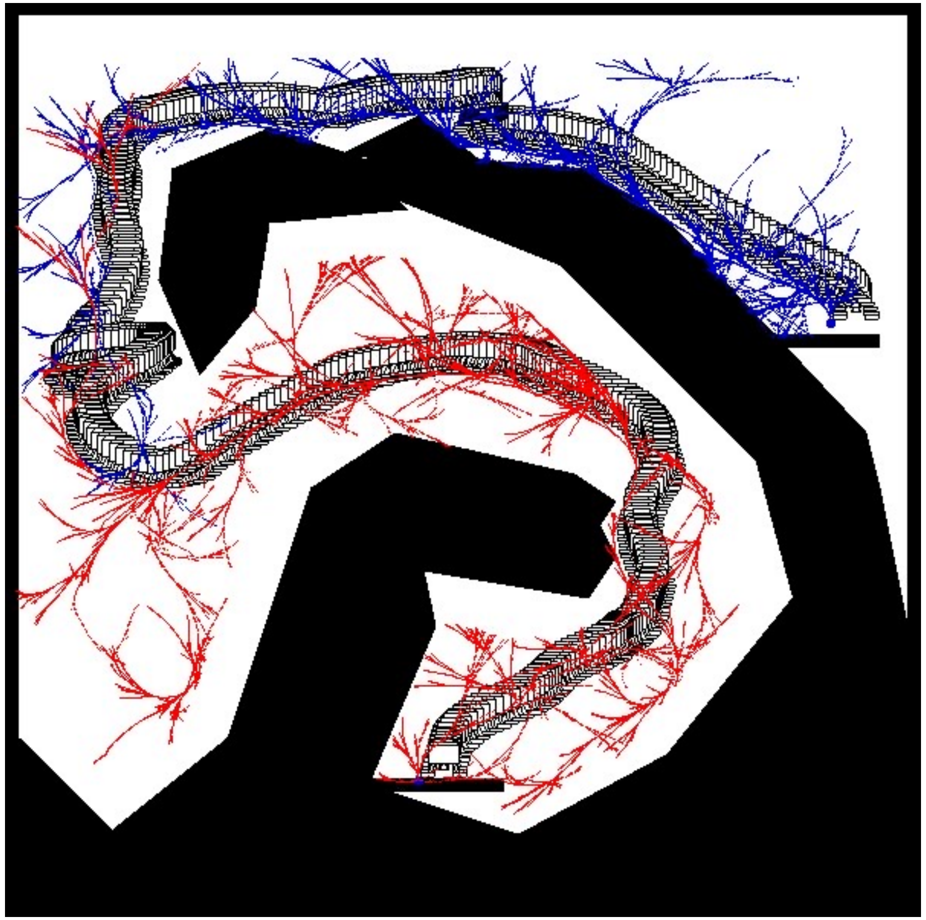
initialize search tree T with x_{start}
while T is less than the maximum tree size **do**
    $x_{\text{samp}} \leftarrow$ sample from $\mathcal{X}$
    $x_{\text{nearest}} \leftarrow$ nearest node in T to x_{samp}
    employ a local planner to find a motion from x_{nearest} to x_{new} in the direction of x_{samp}
    if the motion is collision-free **then**
        add x_{new} to T with an edge from x_{nearest} to x_{new}
        if x_{new} is in $\mathcal{X}_{\text{goal}}$ **then**
            return SUCCESS and the motion to x_{new}
        end if
    end if
end while
return FAILURE

Rapidly Exploring Random Trees (RRT)

Algorithm 10.3 RRT algorithm.

initialize search tree T with x_{start}
while T is less than the maximum tree size **do**
    $x_{\text{samp}} \leftarrow$ sample from $\mathcal{X}$
    $x_{\text{nearest}} \leftarrow$ nearest node in T to x_{samp}
    employ a local planner to find a motion from x_{nearest} to x_{new} in the direction of x_{samp}
    if the motion is collision-free **then**
        add x_{new} to T with an edge from x_{nearest} to x_{new}
        if x_{new} is in $\mathcal{X}_{\text{goal}}$ **then**
            return SUCCESS and the motion to x_{new}
        end if
    end if
end while
return FAILURE

RRT: Lunar Lander



Check out Steven Lavalle's RRT Gallery: http://msl.cs.uiuc.edu/rrt/gallery.html

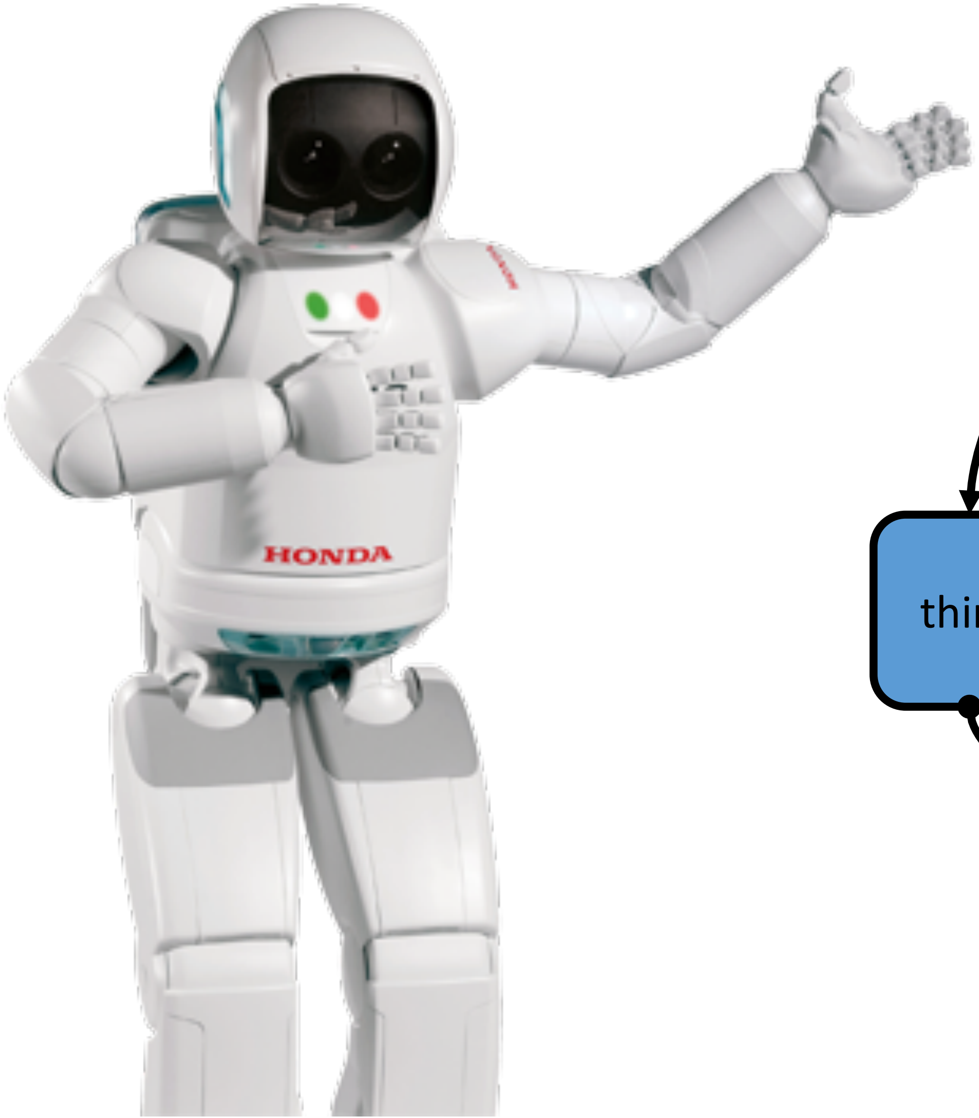
Summary

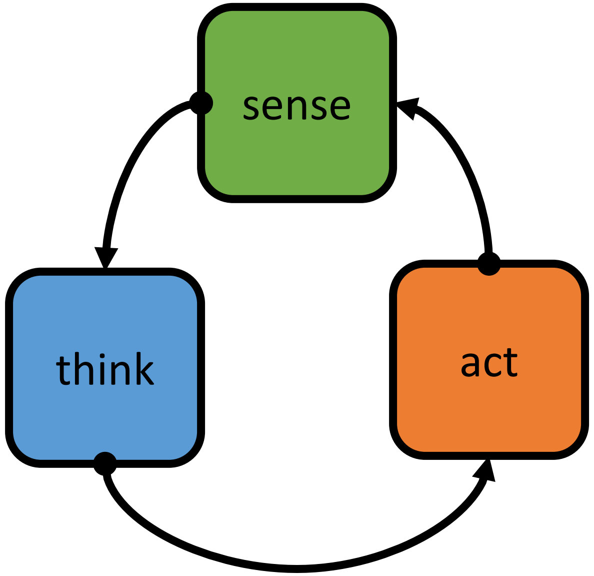
- Given an initial state and a desired final state, **motion planning** provides us with tools to find a time horizon and a sequence of actions to find a trajectory that reaches the goal without collisions
- A **roadmap** path planner uses a graph representation of free space, which can then provide a trajectory using search algorithms
- The basic **RRT** algorithm is a sampling-based method that grows a single search tree from start to find a motion to goal
  - Uses a local planner to find a motion from the nearest node to the sampled node

A few things that might be useful to know

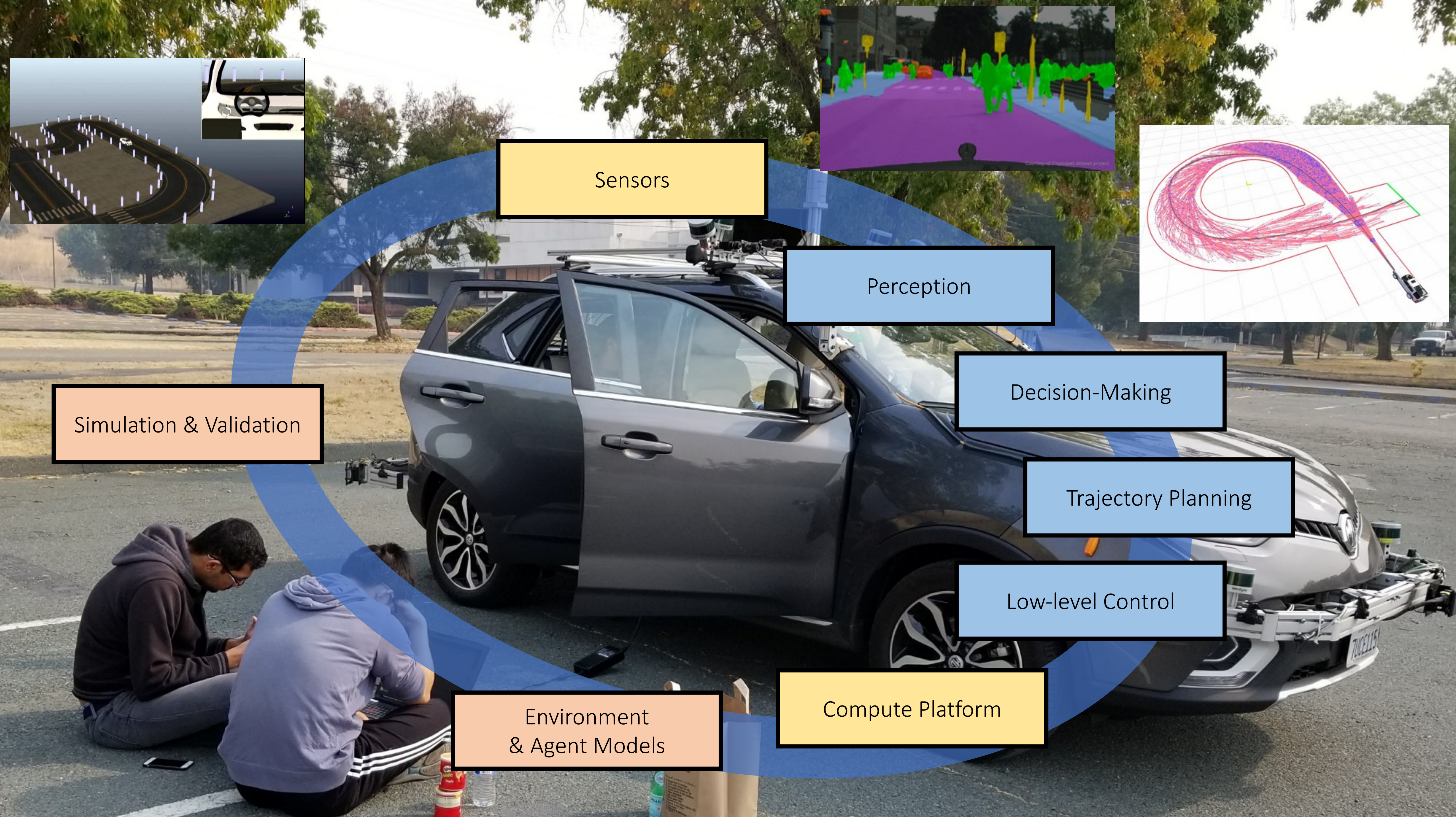
- What are some key properties of planners?
- Think about what applications some properties or types of planners might be needed.
- If given a very simple graph, can you find the shortest path?
- Be somewhat familiar with the pros and cons of the planners we just discussed.

Course Recap



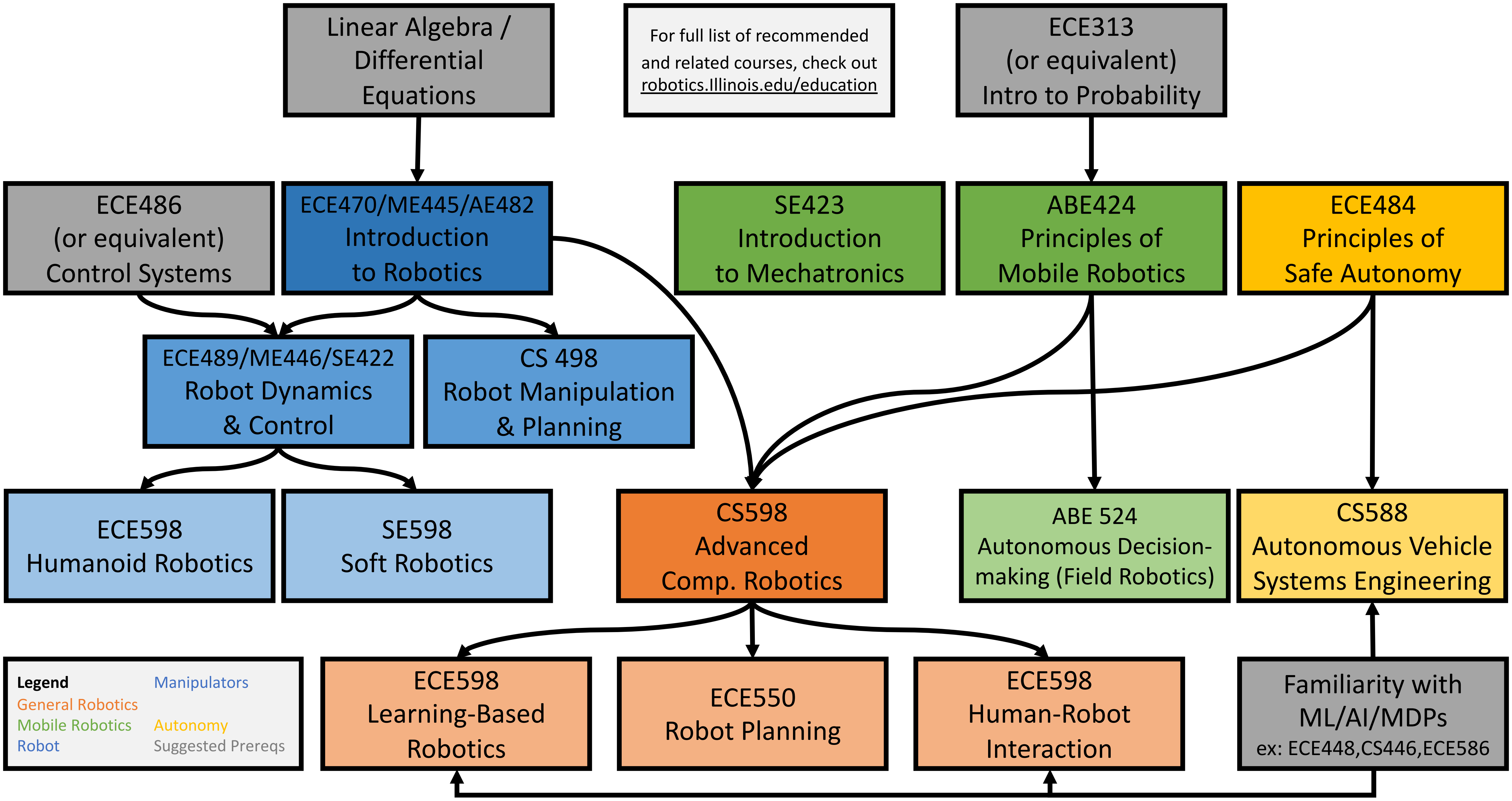


1. Linear algebra and diff. eq. review
2. DoF, configuration space
3. Rigid body motion & transformations
4. Screw theory
5. Forward Kinematics
6. Velocity Kinematics
7. Inverse Kinematics
8. Dynamics
9. Motion Planning


Sensors
Perception
Decision-Making
Trajectory Planning
Low-level Control
Compute Platform
Environment
& Agent Models
Simulation & Validation

If you liked…	Try this!
Everything!	ABE 424 Principles of Mobile Robotics ECE 484 Principles of Safe Autonomy

If you liked…	Try this!
Everything!	ABE 424 Principles of Mobile Robotics ECE 484 Principles of Safe Autonomy
Linear Algebra	MATH 415 Applied Linear Algebra ECE 515 / ME 540 Control System Theory and Design
Sensing and State Estimation	ECE 310 / 417 Signal Processing ECE 437 Sensors and Instrumentation ABE 424 Principles of Mobile Robotics
Robot Kinematics	ECE 489 / ME 446 / SE 422 Robot Dynamics and Control CS 498 Robot Manipulation and Planning
Rigid Body Motion	SE 598 Soft Robotics ECE 549 Computer Vision
Control	ECE 486 Control Systems (or equivalent in your department) ECE 515 / ME 540 Control System Theory and Design
Decision-Making	ECE 448 Introduction to AI CS 446 Machine Learning
Planning	CS 498 Robot Manipulation and Planning
Labs	SE 423 Introduction to Mechatronics
Graduate-Level Topics Courses	ECE 598SG Learning-Based Robotics ECE 598HCR Human-Robot Interaction ECE 598JK Humanoid Robotics CS 598 Advanced Computational Robotics



Linear Algebra / Differential Equations
For full list of recommended and related courses, check out robotics.Illinois.edu/education
ECE313 (or equivalent) Intro to Probability
ECE486 (or equivalent) Control Systems
ECE470/ME445/AE482 Introduction to Robotics
SE423 Introduction to Mechatronics
ABE424 Principles of Mobile Robotics
ECE484 Principles of Safe Autonomy
ECE489/ME446/SE422 Robot Dynamics & Control
CS 498 Robot Manipulation & Planning
ECE598 Humanoid Robotics
SE598 Soft Robotics
CS598 Advanced Comp. Robotics
ABE 524 Autonomous Decision-making (Field Robotics)
CS588 Autonomous Vehicle Systems Engineering
Legend
Manipulators
General Robotics
Mobile Robotics
Autonomy
Robot
Suggested Prereqs
ECE598 Learning-Based Robotics
ECE550 Robot Planning
ECE598 Human-Robot Interaction
Familiarity with ML/AI/MDPs
ex: ECE448,CS446,ECE586