Introduction

Mengxue Hou, Assistant Professor,
Notre Dame, EE

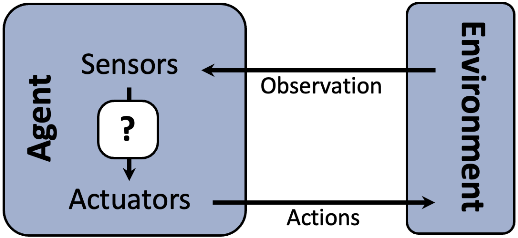
Fall 2023

What is AI Planning?

Task of making the *rational* decision

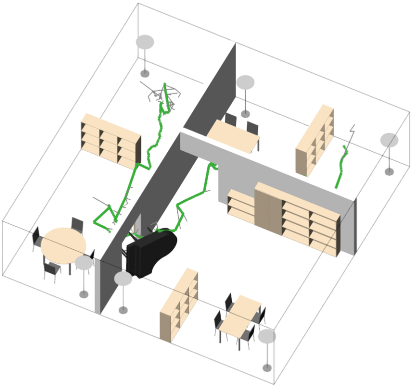
- Rational: maximally achieving pre-defined goals
- Being rational means minimizing your expected cost
- Based on *knowledge/representation* of the physical environment/agent

What is AI Planning?



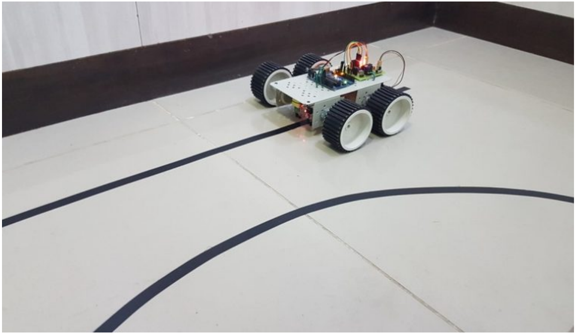
- How to acquire/represent knowledge from physical world?
- How does knowledge lead to action?

Robotic motion planning



- Continuous state space
- Large dimensional space

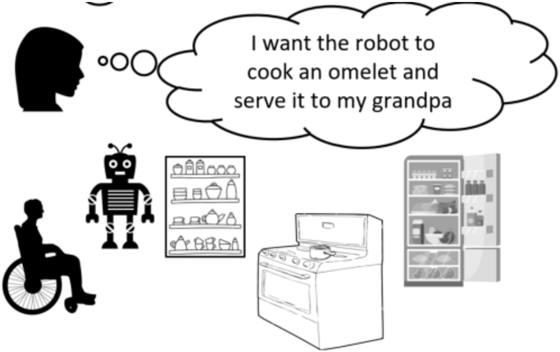
Optimal control/Model predictive control



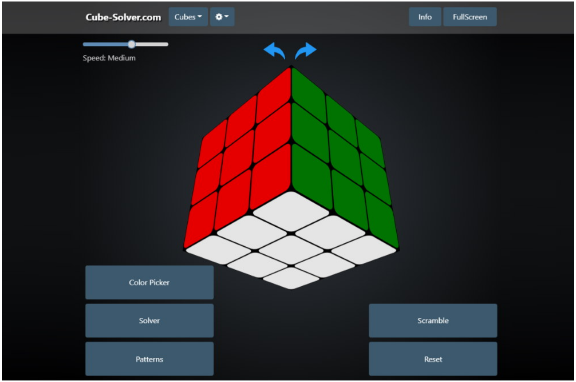
- Optimality guarantee
- Assurance

Why do we care about AI planning?

Examples



Search algorithms in AI



- Discrete state space
- Large dimensional space

Scope of this course

The “simulated world”: Cognitive systems, neural sciences, LLMs …

The real world: robotics, control, embedded systems, transportation systems …

What this course will cover?

- Deterministic planning
  - Graph search
  - Automated planners, AI planning tools
  - Dynamic programming
  - Model predictive control
- Stochastic planning
  - Markov chain, Hidden Markov chain
  - Markov decision processes, Partially observable Markov decision processes
- Reinforcement learning
  - Approximation in value space/policy space
  - Model-based v.s. model-free methods

Pre-requisite

- Linear algebra, undergrad level probability
  - Will have a brief review later in class
- Coding: Python
  - Tutorial: `https://cs231n.github.io/python-numpy-tutorial/`
  - Will give coding examples in class (as we move to RL part)

Course Materials

- Slides
- Notes
- Reference books (recommended but not required)

Course Materials

How I will teach the class:

- Slides
- Hand writing annotation on the slides

Link to all resources: canvas

Course materials will be posted approx. 1 day before class

Grading

- Homework (40%)
  - 1 per 2 week, assigned every Tuesday, about 4 hours to finish.
  - 6 HW in total (1 optional, 8% each one)
  - Every HW will be 2 parts: writing + coding

Grading

- Midterm exam (30%)
  - Midterm will be in class written exam
- Final project (30%)
  - Coding + group report (20%)
  - presentation (10%)

Office hour

Thursday after class at my office (Cushing Hall 226B)
Send me an email before coming to office hour

Something about honor code

- If you have trouble with HWs, you are encouraged to work together
- You are encouraged to use LLM (e.g. ChatGPT) **for this course**, but use it wisely
- Adhere to ND guidelines

Chapter 1: Maps and Deterministic Planning

Mengxue Hou, Assistant Professor,
Notre Dame, EE

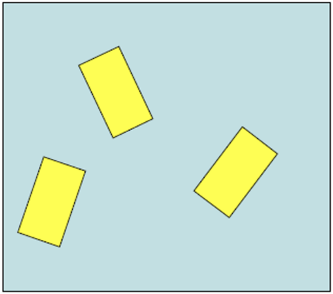
Fall 2023

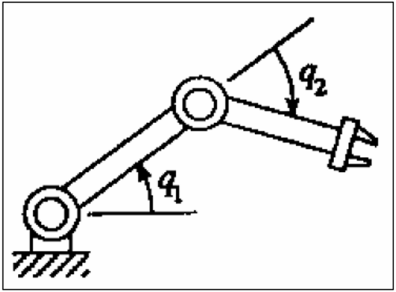
Degree of Freedom

- The geometric configuration of a robot is defined by p degrees of freedom (DOF)
- Assue p DOFs, the geometric configuration of a robot can be defined as

$$\mathrm{q} = (q_1, \ldots, q_p)$$

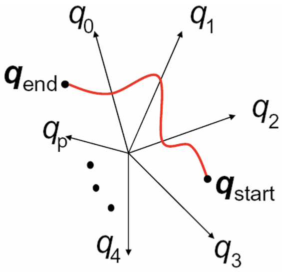
Configuration Space (C-Space)





- Configuration space $\mathcal{C}$ = set of values of q corresponding to legal configurations of the robot
- Defines the set of possible parameters (the search space) and the set of allowed paths

The motion planning problem



- A configuration q is legal if it does not cause the robot to intersect the obstacles
- Given start and goal (q_{start} and q_{goal}), find a continuous sequence of legal (and optimal) configurations from q_{start} to q_{goal}
- Report failure if no path found

Obstacle Maps

An obstacle map represents a 2D plane where all the points that are occupied by obstacles are marked as unaccessible for the robot.

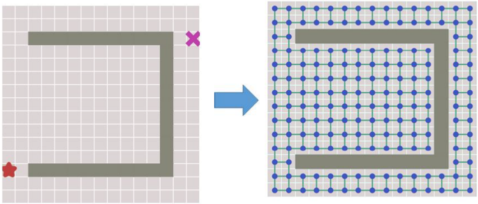


Figure: Occupancy Grid Map

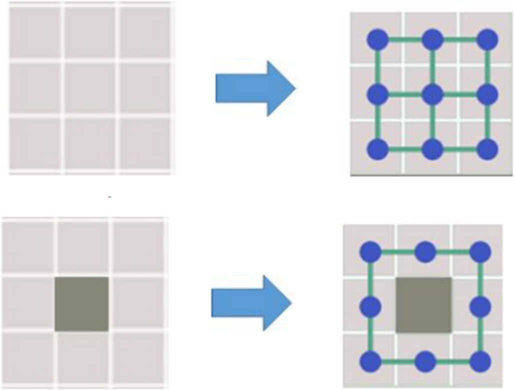
Occupancy Grid Map

The first step for path planning is to discretize the map. How to discretize the map?

Connectivity Graph

For each cell, we consider cells that can be accessible from it as its neighbors.

Obstacle Maps



The Decision Tree

- States
- Initial/goal state
- Actions, action cost
- Transition model

Graph Search Problem

In order to find a path between the starting point and goal point. We now want to find a path from the starting node to the target node.
We have converted the path planning problem into a graph search problem.
From the starting node, we iteratively search the neighbors, until we reached the target node.
Which cell should we search next? The sequence of this search is very important!