CSE 574 Planning and Learning Methods in AI

Ransalu Senanayake

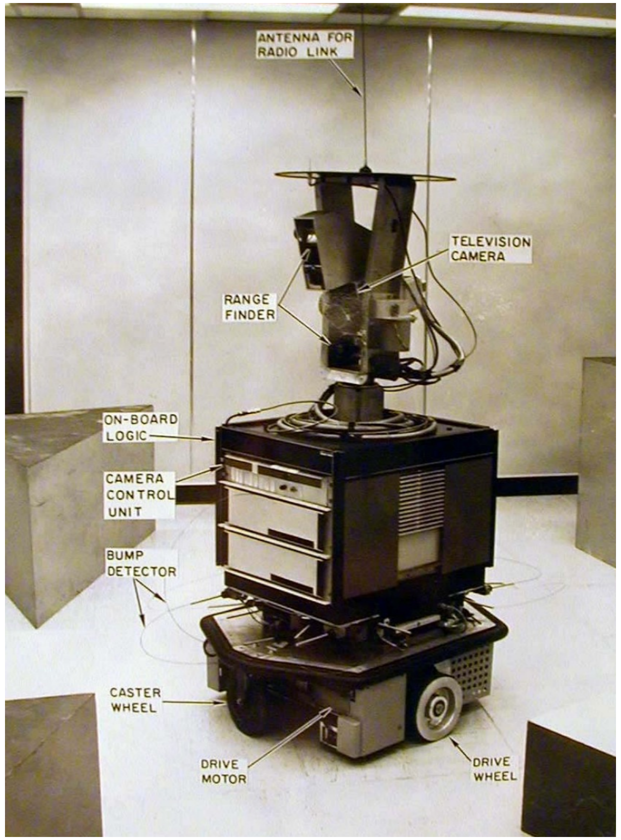
Week 10

Robot Motion Planning

Shakey the Robot (1966-1972)

- https://www.youtube.com/watch?v=7bsEN8mwUB8

- Used STRIP-based A* search



Which space should we work in?

Task space

- A set of all possible end-effector poses

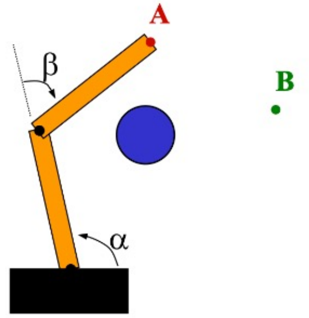
$$[x, y, z, \theta_1, \theta_2, \theta_3]$$

Configuration space

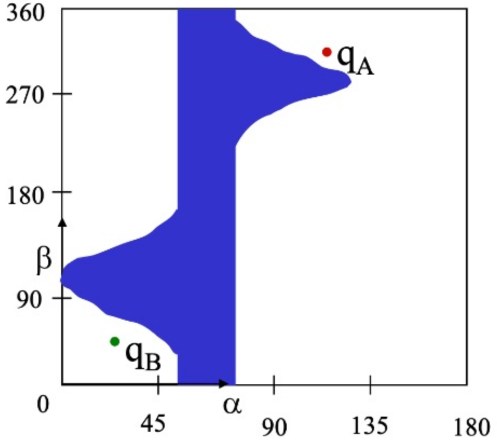
- A set of of all possible c-configuration

A vector of angles (revolute) and/or lengths (prismatic)

- Fwd vs. inverse kinematics



An obstacle in the robot's workspace



The C-space representation of this obstacle…

Howie Choset, 16-311, Spring 2018

Motion Planning Algorithms

1. Naive methods such as Bug
   - Move towards the goal in a straight line. Follow the walls of the obstacles

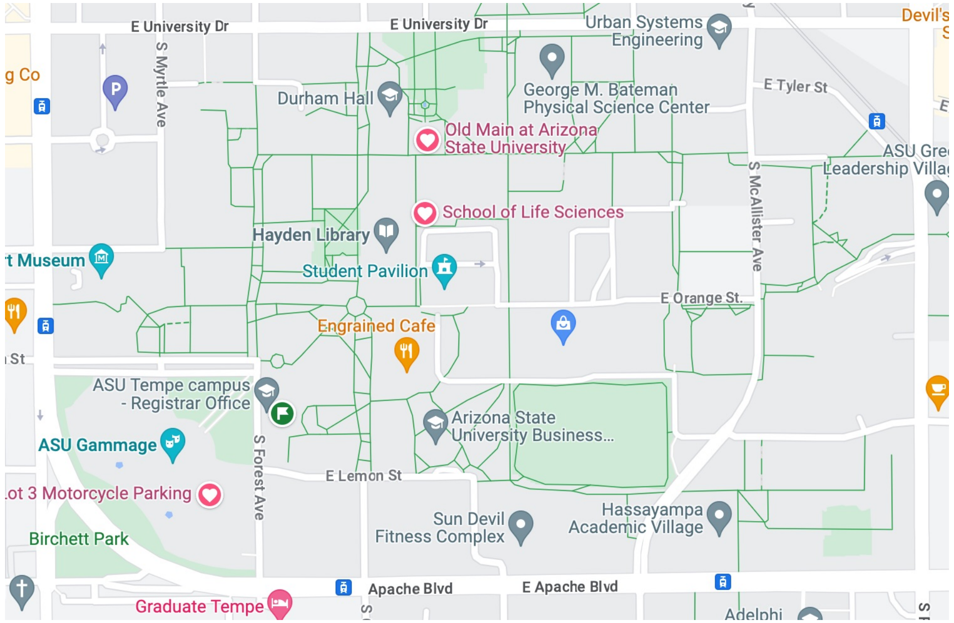
   to avoid it, if meet any.

2. Naive Discretizations
   - Discretize the configuration space
     - e.g. grid/interval
   - Search using A*, Dijkstra’s, etc.

3. Sampling based techniques
   - **PRM**
   - Sampling-Based Roadmap of Trees (SRT)
   - Expansive-Spaces Tree planner (EST)
   - **Rapidly Exploring Random Trees (RRT)**
     - Growing a tree by randomly generating configurations and connecting a feasible closer line-of-sight edge to the nearest vertex
     - Good for generating a single plan (PRMs are for any place to any place path planning). Save the <u>one-time path</u> as you grow.
     - Bi-directional RRT

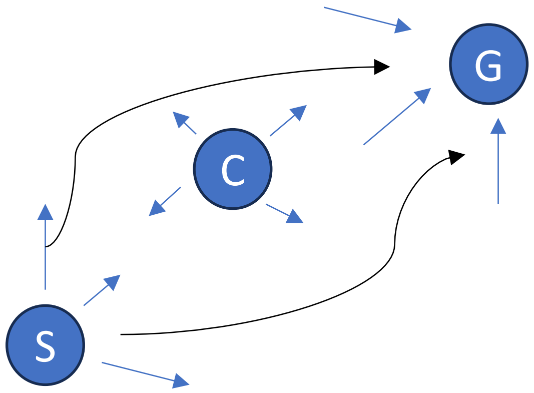
Motion Planning Algorithms (Cont.)

- **Probabilistic road maps (PRMs)**
  - Incrementally build the graph
    - Take random samples from the configuration space
    - Check if they are feasible or not
    - Connect nearby configurations
  - Find a good global set of routes that can be used for going from anywhere to anywhere (whereas RRT is a onetime plan to go from A to B)
  - Find the shortest path using A*/Dijkstra's graph search



Motion Planning Algorithms (Cont.)

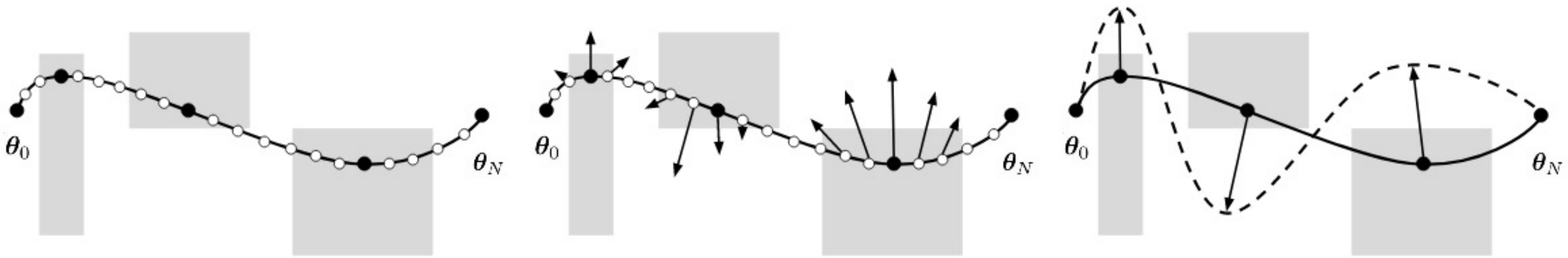
4. Artificial potential fields



Motion Planning Algorithms (Cont.)

5. Trajectory optimization

- Starts with a possibly infeasible trajectory (parameterized by way points)
- Optimize it to guarantee feasibility (no collisions) and smoothness/shortest



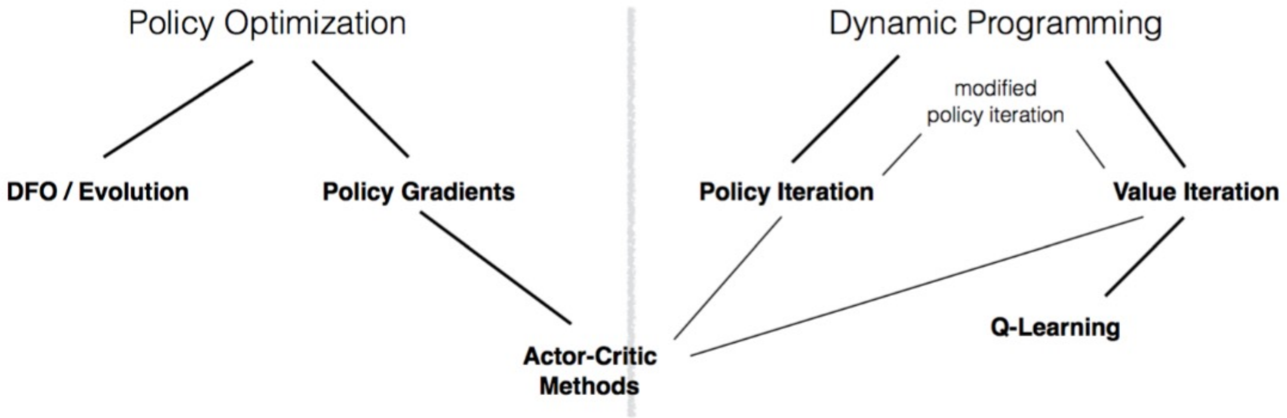
- E.g.
  - Covariant Hamiltonian Optimization for Motion Planning (CHOMP)
  - Stochastic Trajectory Optimization for Motion Planning (STOMP)
  - TrajOpt
  - Gaussian Process Motion Planning (GPMP)

Mukadam et al, Continuous-time Gaussian process motion planning via probabilistic inference

Motion Planning Algorithms (Cont.)

6. Reward-based algorithms

- Dynamic programming
  - Dividing into small sub-problems
  - Using Bellman-Ford algorithm (finding the shortest path)
- Approximate dynamic programming (a.k.a. Reinforcement Learning) - unknown exact model



Source: Peeter Abeel, ICML’17 tutorial

Robot Motion Planning

- https://moveit.ros.org/

- Howie Choset’s course at CMU