CSE-571
Robotics

Planning and Control:
Markov Decision Processes

Markov Decision Process (MDP)

- **Given:**
  - States $x$
  - Actions $u$
  - Transition probabilities $p(x'|u,x)$
  - Reward / payoff function $r(x,u)$

- **Wanted:**
  - Policy $\pi(x)$ that maximizes the future expected reward

Markov Decision Process (MDP) Diagram

Rewards and Policies

- Policy (general case):
  $$\pi : \pi_{t-1},u_{t-1} \rightarrow u_t$$
- Policy (fully observable case):
  $$\pi : x_t \rightarrow u_t$$
- Expected cumulative payoff:
  $$R_T = E \left[ \sum_{t=1}^{T} \gamma^t r_{t+\tau} \right]$$
  - $T=1$: greedy policy
  - $T>1$: finite policy case, typically no discount
  - $T=\infty$: infinite-horizon case, finite reward if discount < 1
Policies contd.

- Expected cumulative payoff of policy:
  \[ R_T^\pi(x_i) = E \left[ \sum_{\tau=1}^{T} \gamma^\tau r_{t+\tau} | u_{t+\tau} = \pi(\tau_{1:t+\tau-1}u_{1:t+\tau-1}) \right] \]

- Optimal policy:
  \[ \pi^* = \arg\max_u R_T^u(x_i) \]

- 1-step optimal policy:
  \[ \pi_1(x) = \arg\max_u r(x,u) \]

- Value function of 1-step optimal policy:
  \[ V_1(x) = \gamma \max_u r(x,u) \]

2-step Policies

- Optimal policy:
  \[ \pi_2(x) = \arg\max_u \left[ r(x,u) + \int V_1(x') p(x' | u,x) dx' \right] \]

- Value function:
  \[ V_2(x) = \gamma \max_u \left[ r(x,u) + \int V_1(x') p(x' | u,x) dx' \right] \]

T-step Policies

- Optimal policy:
  \[ \pi_T(x) = \arg\max_u \left[ r(x,u) + \int V_{T-1}(x') p(x' | u,x) dx' \right] \]

- Value function:
  \[ V_T(x) = \gamma \max_u \left[ r(x,u) + \int V_{T-1}(x') p(x' | u,x) dx' \right] \]

Infinite Horizon

- Optimal policy:
  \[ V_\infty(x) = \gamma \max_u \left[ r(x,u) + \int V_\infty(x') p(x' | u,x) dx' \right] \]

- Bellman equation

- Fix point is optimal policy

- Necessary and sufficient condition
**Value Iteration**

- for all $x$ do
  $$\hat{V}(x) \leftarrow r_{\text{min}}$$
- endfor
- repeat until convergence
  - for all $x$ do
    $$\hat{V}(x) \leftarrow \gamma \max_{u} \left[ r(x,u) + \int \hat{V}(x') p(x'|u,x)dx' \right]$$
  - endfor
- endrepeat
  $$\pi(x) = \arg\max_{u} \left[ r(x,u) + \int \hat{V}(x') p(x'|u,x)dx' \right]$$
Value Function and Policy

- Each step takes $O(|A| \cdot |S| \cdot |S|)$ time.
- Number of iterations required is polynomial in $|S|$, $|A|$, $1/(1-\gamma)$
Value Iteration for Motion Planning
(assumes knowledge of robot’s location)

Frontier-based Exploration
• Every unknown location is a target point.

POMDPs
• In POMDPs we apply the very same idea as in MDPs.
• Since the state is not observable, the agent has to make its decisions based on the belief state which is a posterior distribution over states.
• For finite horizon problems, the resulting value functions are piecewise linear and convex.
• In each iteration the number of linear constraints grows exponentially.
• Full fledged POMDPs have only been applied to very small state spaces with small numbers of possible observations and actions.
• Approximate solutions are becoming more and more capable.

CSE 571
Inverse Optimal Control
(Inverse Reinforcement Learning)

Many slides by Drew Bagnell
Carnegie Mellon University
Autonomous Navigation

UPI
Somerset Woods
May, 2007

Optimal Control Solution
Cost Map

Learning
2-D Planner
Cost Map

Mode 1: Training example
Mode 1: Training example

Mode 1: Learned behavior

Mode 1: Learned behavior

Mode 1: Learned cost map
Mode 2: Training example

Mode 2: Learned behavior

Mode 2: Training example

Mode 2: Learned behavior
Mode 2: Learned cost map

$\text{Cost} = w'F$

Ratliff, Bagnell, Zinkevich 2005
Ratliff, Bradley, Bagnell, Chestnutt, NIPS 2006
Silver, Bagnell, Stentz, RSS 2008

$w = [w_1, F_1]$ -> Learn $F_1$

$w = [w_2, F_2]$ -> Learn $F_2$
Learning Manipulation Preferences

- **Input**: Human demonstrations of preferred behavior (e.g., moving a cup of water upright without spilling)
- **Output**: Learned cost function that results in trajectories satisfying user preferences
Setup

• **Binary** state-dependent features (~95)
  - Histograms of distances to objects
  - Histograms of end-effector orientation
  - Object specific features (electronic vs non-electronic)
  - Approach direction w.r.t goal

• **Task**
  - Hold cup upright while not moving above electronics

Laptop task: Demonstration
(Not part of training set)

Readings

• Max-Ent IRL (Ziebart, Bagnell): http://www.cs.cmu.edu/~bziebart/
• CIOC (Levine) http://graphics.stanford.edu/projects/cioc/cioc.pdf
• Imitation learning (Ermon): https://cs.stanford.edu/~ermon/
• Human/manipulation (Dragan): https://people.eecs.berkeley.edu/~anca/research.html