COS402- Artificial Intelligence
Fall 2015

Lecture 16: MDP: Utility and Policy

Outline

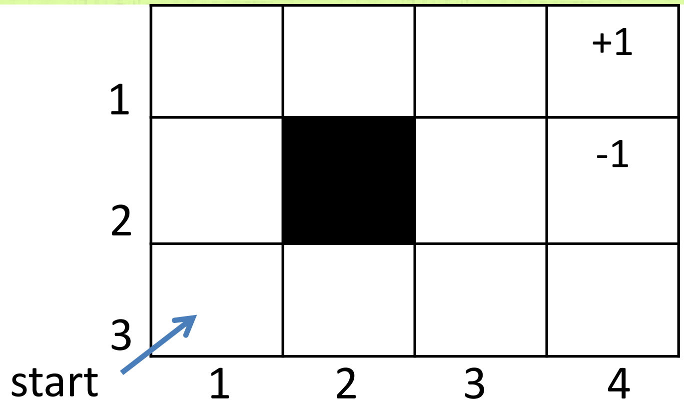
- **Markov Decision Process (MDP)**
  - **Definition and Examples**
- **Utility of a state, a policy and a state sequence**
- **Optimal policy**
- **The Bellman equation for utility**

Decision theory: MDP

- **General principle:**
  - **Assign utilities to states**
  - **Take actions that yields highest expected utility**
  - **Rational decision vs. human decision**
- **Simple decision vs complex decision**
  - **Simple decision: make a single decision, achieve short-term goal.**
  - **Complex decision: make a sequence of decisions, achieve long-term goal.**
    - **We will look at problems of making complex decisions**
    - **Markov assumption: The next state only depends on current state and action.**

MDP: Example

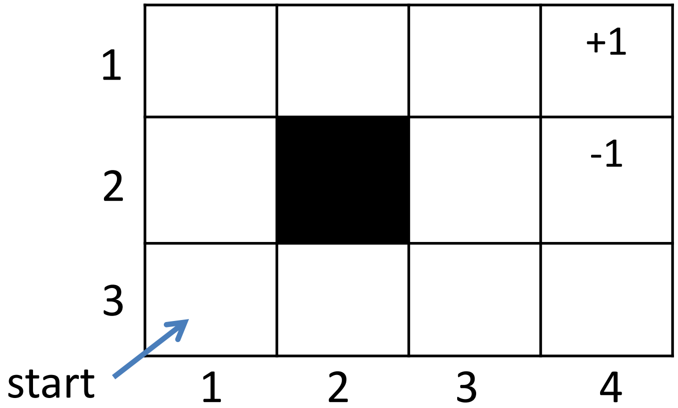
- **A robot in a grid**



- **MDP:**
  - **Initial state and states: locations/squares,**
  - **Actions: can move in 4 directions: up, down, left and right**
    - **No available actions at terminal states.**
  - **Transition model: P(s'|s, a)**
    - **80% of time moves in desired direction; 20% of time moves at right angle to the desired direction; no movement if bumps to a wall/barrier.**
  - **Rewards: +1 at [1,4], -1 at [2,4], and -0.04 elsewhere**
  - **Solution?**

MDP: Example

- **A robot in a grid**



- **MDP:**
  - **Initial state and states: fully observable**
  - **Actions:**
  - **Transition model: P(s'|s,a)**
    - **Markov assumption: The next state only depends on current state and action.**
  - **Rewards: R(s), additive**
  - **Solution: A policy maps from states to actions. An optimal policy yields the highest expected utility/sum of rewards.**

MDP: More Examples

- **Driving cars**
- **Controlling elevators**
  - **states: locations of the elevator, buttons pushed**
  - **Actions: send the elevator to particular floor**
  - **Rewards: measure of how long people wait**
- **Game playing(backgammon)**
- **Searching the web**
  - **states: urls**
  - **Actions: choose a link to expand**
  - **Rewards: find what is looking for**

MDP: More Examples

- **Animals deciding how to act/live**
  - **Must figure out what to do to get food, get mate, avoid predators, etc.**
  - **Cat and mouse in P5.**
    - **states:**
    - **Actions:**
    - **Rewards:**

HMM vs. MDP

HMM	MDP
States are hidden	States are visible
No control, only observation	Probabilistic control via actions
No reward	Reward at each state

Utility of a state sequence

- **Additive rewards**
  - **U([s_0, $s_{1,}$ s_2 ,,,]) = R(s_0) + R(s_1) +R(s_2) +**
- **Discounted rewards**
  - **U([s_0, $s_{1,}$ s_2 ,,,]) = R(s_0) + rR(s_1) + r^2 R(s_2) + …**
  - **Discount factor r is between 0 and 1**

Optimal policies and the utilities of states

- **$U^{\pi}(s)$: The expected utility obtained by executing π staring in s.**
  - **$U^{\pi}(s) = E[\sum_{t=0}^{\infty} r^{t} R(S_{t})]$**
- **π^{*} : an optimal policy**
  - **$\Pi^{*} = arg\max_{\pi} U^{\pi}(s)$**
- **π^{*} is independent of the starting state**
  - **When using discounted utilities with no fixed time limit.**
- **$U(s) = U^{\pi^{*}}(s)$**
  - **The true utility of a state is the expected sum of discounted rewards if an agent executes an optimal policy.**

Optimal policies and the utilities of states

- **π^* : an optimal policy**
  - **$\Pi^*(s) = \underset{a}{\boldsymbol{arg}\max} \sum_{s'} P(s'|s,a)\, \boldsymbol{U}(\boldsymbol{S'})$**
  - **Choose an action that maximizes the expected utility of the subsequent state.**
- **How to calculate U(s)?**

The Bellman equations for utilities

- **The relationship between the utility of a state and the utility of its neighbors**
  - $\mathbf{U}(\mathbf{s}) = \mathbf{R}(\mathbf{s}) + \boldsymbol{r}.\ \max_{\boldsymbol{a}\epsilon \boldsymbol{Actions}(\boldsymbol{s})} \sum_{\boldsymbol{s}'} \mathbf{P}(\mathbf{s}'|\mathbf{s},\mathbf{a})\ \boldsymbol{U}(\boldsymbol{S}')$

	1	2	3	4
1	0.812	0.868	0.918	+1
2	0.762		0.660	-1
3	0.705	0.655	0.611	0.388

- **Assuming the agent chooses the optimal action**
  - **What is the best action in state (1,1)? in state (3,4)?**
  - **How many Bellman equations do we have for this MDP?**
  - **Can we solve these equations directly and efficiently?**
    - **The value Iteration algorithm (Next class.)**

Review questions: true or false

1. **A Markov Decision Process consists of a set of states(with initial state S_0), a set of actions in each state, a transition model P(s'|s,a), and a reward function R(s).**
2. **The goal of a MDP is to find a sequence of actions that will maximize the expected utility.**
3. **A policy is a mapping from states to actions. An optimal policy is a policy which yields the highest expected utility.**
4. **The utility of a state in a MDP is the reward received at the state.**

Review questions: true or false(cnt'd)

5. **R(s) is the "short term" reward for being in state s whereas U(s) is the "long term" total rewards from s onward. The true utility of a state is the expected sum of discounted rewards if the agent executes an optimal policy start from s.**
6. **The Bellman equation for utility indicates that the utility of a state is the immediate reward for that state plus the expected discounted utility of the next state of an action chosen by the agent.**
7. **The utility of a state sequence is the sum of all the rewards over the sequence, possibly discounted over time.**

Announcement & Reminder

- **P3 is due today.**
  - **Upload files to CS dropbox by midnight.**
- **W4 is due on Tuesday Nov. 24th**
  - **Turn in hard copy in class.**
- **P4 is released and is due on Tuesday Dec. 1st**

  --- Turn in hard copies in class.