CS 486 — Lecture 18: Markov Decision Processes, Part 1

1 MDP — Introduction

- In some problems, we have finite stages. But in other problems, we have to solve ongoing problems (perhaps it has an infinite time horizon).
- We also define an *indefinite* time horizon — this would be when the agent knows it will *eventually* stop, but we don't know when. An infinite horizon will potentially go on forever.
- We also may calculate the utility at the end for a finite problem, but for an infinite/indefinite horizon problem, it makes more sense to do a sequence of rewards for each time step.
- We can model a MDP with S being a set of states, A being a set of actions, and R being a set of reward functions. P represents transition probabilities.

2 Rewards

- $R(s)$ is the reward of entering state s.
- We consider 3 types of rewards:
  - Total reward — adds all reward functions until we hit the current state. This wouldn't work though if we have infinite time steps, since we would have infinite reward functions.
  - Average reward — we now take the total reward but multiply it by $\frac{1}{n}$, where n approaches infinity. But if the total reward is finite, the average reward is 0!
  - Discounted reward — $R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \ldots$, where $0 \leq \gamma < 1$. That is, we want the get rewards sooner rather than later. The "discount factor" makes future rewards worse than present ones. The discount factor is also useful to help model the fact that the future may not occur — there might be a chance that our states will end at the next step!

3 Variations of MDP

- A fully observable MDP is when the agent knows what state it is in.
- A partially observable MDP (POMDP) combines a MDP and a hidden Markov model — the agent may not know what state it is in, but it can get some noisy signal of the state.

4 Policy

For the following, we use a "grid world" as such:

	1	2	3	4
1	Start			
2		X		-1
3				+1

- The robot starts at, well, start, and has a chance of going the correct direction, or bumping left, or bumping right. We could use $80\%, 10\%, 10\%$ as our split.

- A policy specifies what the agent should do as a function of the current state.
- A policy is:
  - non-stationary if it is a function of the state and the time
  - stationary if it is a function of the state
- The optimal policy of the grid world changes based on $R(s)$ for any non-goal state s. This would show a careful balancing of risk and reward.
- Based on our $R(s)$ function, we can see that:
  - If R is very negative, it tries to take the shortest route to the nearest exit, regardless of whether it is -1 or +1.
  - If R is moderately negative, if takes the shortest route to the $+1$ state, though it will be willing to risk the -1 state.
  - If R is a bit negative, it will be conservative and aim to prefer safer routes (near the bottom) to avoid falling into the -1 state by accident.
  - If R is only a tiny bit negative or 0, it takes no risk and heads directly away from the -1 state to avoid it, though it may still hit the wall a few times.
  - If R is positive, it *avoids* the goal states!