EECS 496: Sequential Decision Making

Soumya Ray

sray@case.edu

Office: Olin 516

Office hours : T 4-5:30 or by appointment

Today

- Part 3: Sequential Decision Making under Uncertainty
- Test: 11/21, in class
  - Material: everything up to previous week (11/14)

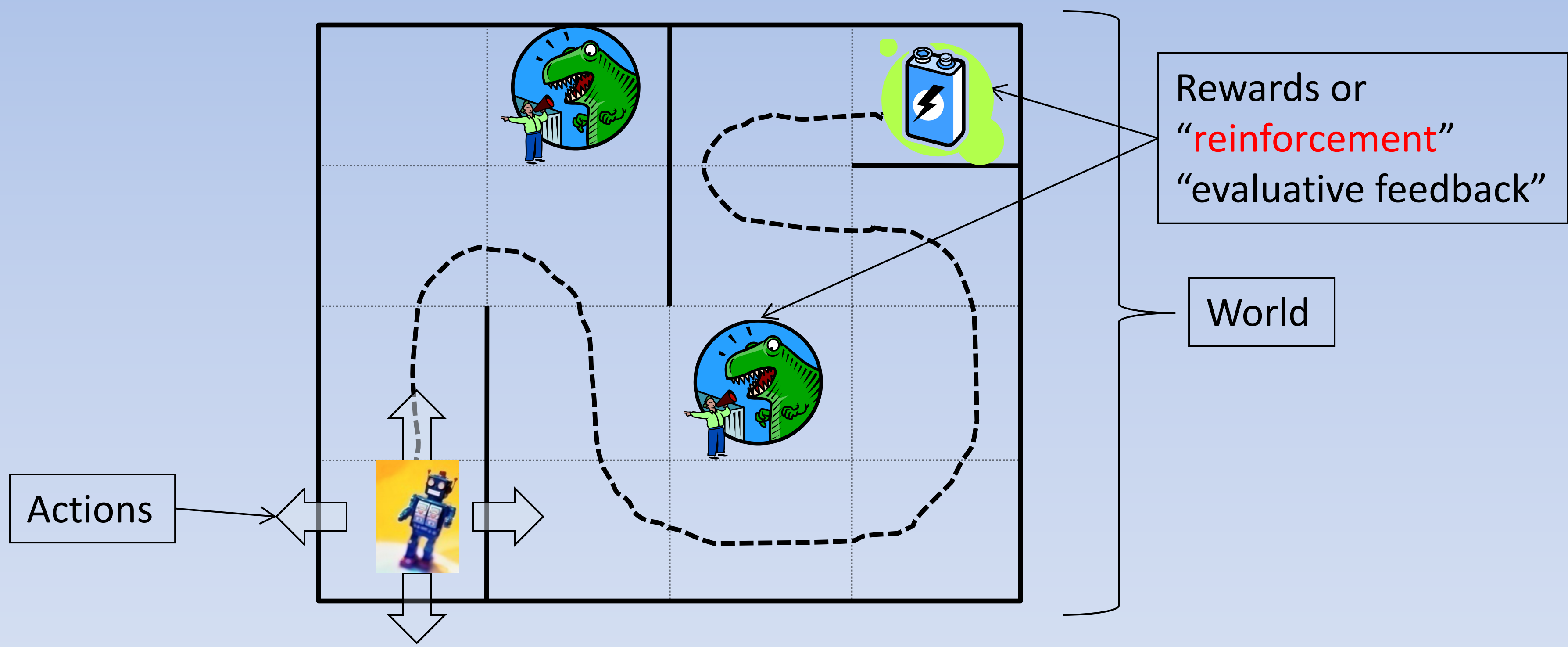
Sequential Decision Making under Uncertainty

- We've seen how to reason in uncertain environments
- Now we will put this reasoning machinery to work at *selecting actions in a stochastic environment*
- To do this we need one more piece of information, a "utility function"

Utility function

- The job of this function is to capture the *long-term value* of taking an action at some state of the world

Sequential Decision Making Example



Goal: Find a sequence of actions from *every state* that maximizes "expected future reward"

SDM and Classical Planning

Sequential Decision Making

- Agent starts with no initial knowledge
- Handles stochastic Worlds/Actions
- Produces “policy”: optimal action for each state
- Propositional only
- Optimize Utility

Classical Planning

- Agent starts with detailed structured knowledge
- Deterministic Worlds/Actions
- Produces “plan”: optimal action sequence from initial state
- Can be extended to first-order worlds
- Goal-Directed

Issues in Sequential Decision Making

- Credit Assignment
  - Suppose the agent performs a sequence of actions, and then the world gives it a reward (or penalty)
  - Which action(s) in the sequence were really responsible for this reward (or penalty)?

Issues in Sequential Decision Making

- Exploration versus Exploitation
  - Generally, the agent will not start off by knowing the characteristics of the world it is in, specifically, how to get to the high utility states
    - It has to discover these by *exploring* the world
  - Suppose it has explored a bit and found some sequence of actions that looks good
    - Should it just follow (*exploit*) this sequence or explore some more and possibly find an even better sequence?

SDM Formalization

- A formal model for an SDM is defined via a Markov Decision Process (MDP)
- An MDP has six components:
  - A set of states, S, representing possible states of the world
  - A set of actions, A, representing possible actions of the agent
  - A transition function, T
  - A reward function, R
  - An initial state distribution, P_0
  - A "discount factor", $0 \leq \gamma \leq 1$

Transition Function

- The transition function maps a state and action to a probability over the next state: $T: S \times A \times S \to [0,1]$
  - $T(s,a,s')=Pr(s'|s,a)$

Markov property: The next state only depends on the current state and action.

- Actions in the real world aren't necessarily deterministic
  - For a deterministic domain, $T(s,a,s')=1$ for one next state s' and zero elsewhere

Reward Function

- The reward function maps a state and action to a real number: $R\colon S \times A \rightarrow \Re$
  - $R(s,a)$

Markov property: The reward only depends on the current state and action.

- We assume R is a bounded function
- If there is no feedback from the environment when the agent carries out an action, this will be zero

Assumptions

- **First-Order Markovian dynamics** (history independence)
  - $\Pr(S^{t+1}|A^t,S^t,A^{t-1},S^{t-1},\ldots, S^0) = \Pr(S^{t+1}|A^t,S^t)$
  - Next state only depends on current state and current action
- **First-Order Markovian reward process**
  - $\Pr(R^t|A^t,S^t,A^{t-1},S^{t-1},\ldots, S^0) = \Pr(R^t|A^t,S^t)$
  - Reward only depends on current state and action
- **Stationary dynamics and reward**
  - $\Pr(S^{t+1}|A^t,S^t) = Pr(S^{k+1}|A^k,S^k)$ for all t, k
  - The world dynamics do not depend on the absolute time
- **Full observability**
  - Though we can't predict exactly which state we will reach when we execute an action, once it is realized, we know what it is
- **Static, Single Agent**

Policy

- Given a Markov Decision Process, an agent follows a (deterministic) "policy" $\pi: S \rightarrow A$
  - $\pi(s)$ is the action the agent will execute in state s

- An optimal policy, π^*, is a policy that maximizes the expected future reward from any state
  - This is what the agent needs to learn

Optimality Criterion

- Suppose the agent, following policy π, visits a state sequence $s_0, s_1, s_2, \ldots$

- We will measure the goodness or utility of this sequence as the *discounted infinite-horizon cumulative reward:*

$$U([s_0, s_1, \ldots]) = \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)$$

Polynomial discount factor

Discounting

- Two reasons to use this criterion:
  - Behavioral
  - Mathematical
- Behavioral: People and animals appear to prefer short term rewards over long term rewards
- Mathematical: Since visit sequences can be infinitely long, if we just add up the rewards, the sum is not well defined

Aside

- Other optimality criteria exist
  - E.g., could choose to optimize *average reward*
  - Or in the *finite horizon* case, optimize *cumulative reward*
- Algorithms we describe can be extended to these cases

Visit Distribution

- Since actions are stochastic, if we start at some state s_0 and follow π, we will generate many state sequences, each with some probability (product of the transition functions)
  - Call this the visit distribution

Value of a policy

- We define the value of a policy as the *expected utility,* where expectation is with respect to the visit distribution

- Then the optimal policy is the policy that maximizes this expected utility:

$$\pi^* = \arg\max_{\pi} E\left(\sum_t \gamma^t R(s_t, \pi(s_t)) \right)$$

Value of a state under a policy

- We define the value of a state s under a policy π as the value of the policy given that we start at s:

$$V^{\pi}(s) = E\left(\sum_t \gamma^t R(s_t, \pi(s_t)) \mid s_0 = s \right)$$

- This is called the “value function”

Rewriting the value function

- For a Markov Decision Process, we have:

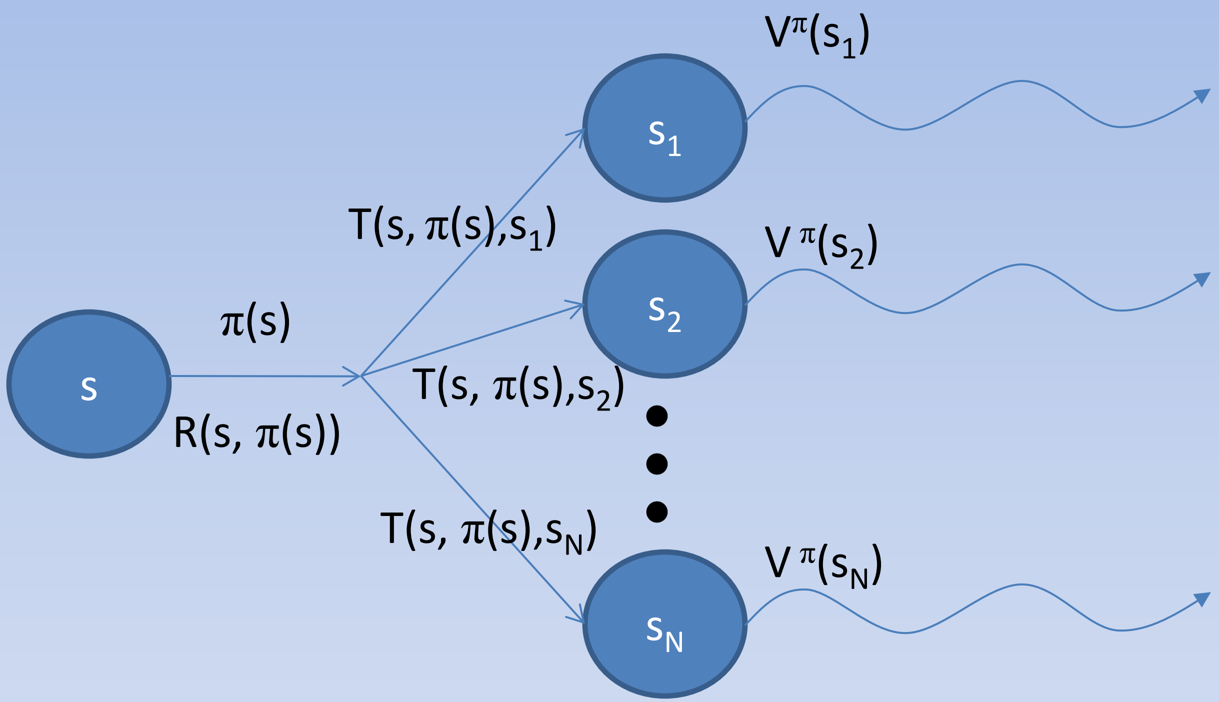
$$V^{\pi}(s) = E\left(\sum_{t=0}^{\infty} \gamma^t R(s_t, \pi(s_t)) \mid s_0 = s\right)$$

$$= R(s, \pi(s)) + \gamma E\left(\sum_{t=1}^{\infty} \gamma^{t-1} R(s_t, \pi(s_t))\right)$$

$$= R(s, \pi(s)) + \gamma \sum_{s'} \Pr(s' \mid s, \pi(s)) \left[E\left(\sum_{t=1}^{\infty} \gamma^{t-1} R(s_t, \pi(s_t)) \mid s_1 = s'\right)\right]$$

$$= R(s, \pi(s)) + \gamma \sum_{s'} T(s, \pi(s), s') V^{\pi}(s')$$

Picture



$$V^\pi(s) = R(s, \pi(s)) + \gamma \sum_{s'} T(s, \pi(s), s') V^\pi(s')$$

Bellman equation