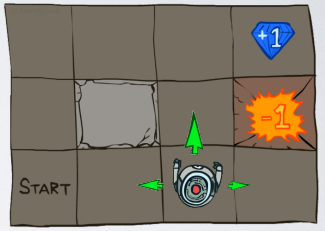
OVERVIEW OF REINFORCEMENT LEARNING

CSE 511A: Introduction to Artificial Intelligence

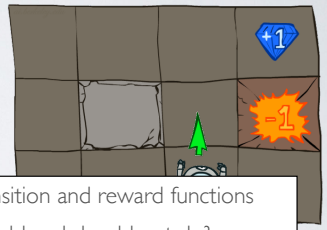
MARKOV DECISION PROCESSES

- An MDP is defined by
  - A set of states $s \in S$
  - A set of actions $a \in A$
  - A transition function $T(s, a, s')$
    - Probability that taking action a from s will lead to s'
    - i.e., $P(s' \mid s, a)$
  - A reward function $R(s, a, s')$
    - Reward of taking action a in s and ending up in s'
    - Sometimes just R(s), R(s,a), or R(s')
  - A discount factor γ
  - A start state
  - Maybe a terminal state



MARKOV DECISION PROCESSES

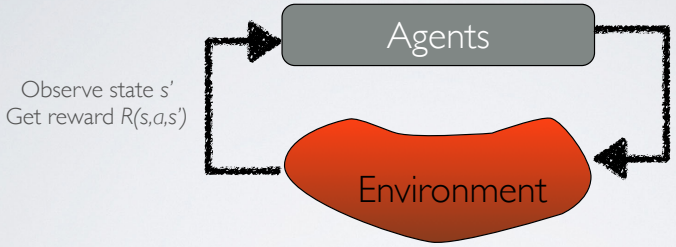
- An MDP is defined by
  - A set of states $s \in S$
  - A set of actions $a \in A$
  - A transition function $T(s, a, s')$
    - Probability that taking action a from s will lead to s'
    - Sometimes just R(s), R(s,a), or R(s')
  - A discount factor γ
  - A start state
  - Maybe a terminal state



Problem: In some applications, you don't know the transition and reward functions

Example: How does a child learn what he/she should and should not do?
They don't know transition and reward functions a priori at birth :)

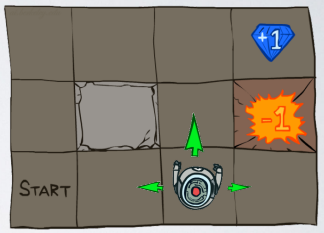
REINFORCEMENT LEARNING



Main ideas:
- Receive feedback in the form of rewards
- Agent's utility is defined by the reward function
- Must (learn to) act so as to maximize expected rewards
- All learning is based on observed samples of outcomes!

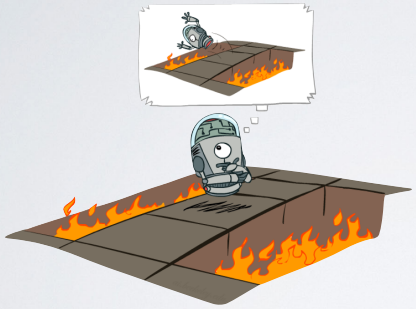
REINFORCEMENT LEARNING

- Still assume an MDP:
  - A set of states $s \in S$
  - A set of actions $a \in A$
  - A transition function $T(s, a, s')$
  - A reward function $R(s, a, s')$
- Still looking for policy $\pi(s)$



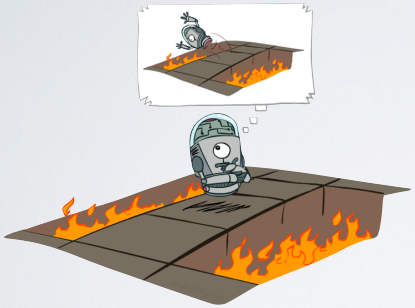
- New twist: Don't know T and R
  - i.e., we don't know which states are good or what the actions will do
  - Must actually try actions and states out to learn

OFFLINE VS ONLINE

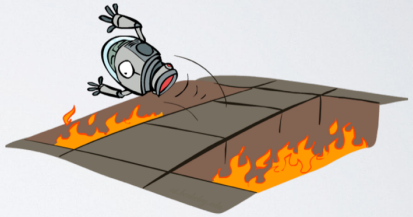


Offline solution:
Value Iteration (VI)

OFFLINE VS ONLINE



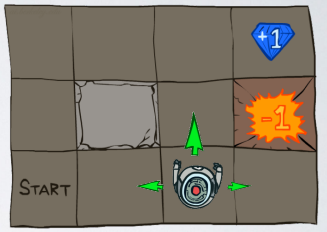
Offline solution:
Value Iteration (VI)



Offline training + Online fine-tuning:
Reinforcement Learning (RL)

MODEL-BASED VS MODEL-FREE LEARNING

- Two general types of RL:
  - Model-based learning:
    - Learn and approximate parameters (i.e., transition and reward functions) of the MDP model
    - Plan based on that approximated model just like in offline learning
  - Model-free learning:
    - Ignore MDP model completely
    - Learn and approximate the value functions $V^*(s)$ or $Q^*(s,a)$ directly
    - Extract policy based on approximated values



EXAMPLE: EXPECTED AGE

- Goal: Compute expected age of CSE 511A students
- If we know the model (i.e., probability distribution of ages), then

$$E[A] = \sum_a P(a) \cdot a = 0.35 \times 20 + \ 0.10 \ \times 21 + \ldots$$

9

EXAMPLE: EXPECTED AGE

- Goal: Compute expected age of CSE 511A students
- If we know the model (i.e., probability distribution of ages), then

$$E[A] = \sum_a P(a) \cdot a = 0.35 \times 20 + \ 0.10 \ \times 21 + \ldots$$

- If we don't know the model, then we collect samples of ages $\{a_1, a_2, \ldots, a_n\}$

10

EXAMPLE: EXPECTED AGE

- Goal: Compute expected age of CSE 511A students
- If we know the model (i.e., probability distribution of ages), then

$$E[A] = \sum_a P(a) \cdot a = 0.35 \times 20 + \ 0.10 \ \times 21 + \ldots$$

- If we don't know the model, then we collect samples of ages $\{a_1, a_2, \ldots, a_n\}$
- Model-based approach: Approximate P(A) and use it in computation

$$\hat{P}(a) = \frac{\#\text{samples with value } a}{n} \qquad E[A] \approx \sum_a \hat{P}(a) \cdot a$$

11

EXAMPLE: EXPECTED AGE

- Goal: Compute expected age of CSE 511A students
- If we know the model (i.e., probability distribution of ages), then

$$E[A] = \sum_a P(a) \cdot a = 0.35 \times 20 + \ 0.10 \ \times 21 + \ldots$$

- If we don't know the model, then we collect samples of ages $\{a_1, a_2, \ldots, a_n\}$
- Model-based approach: Approximate P(A) and use it in computation

$$\hat{P}(a) = \frac{\#\text{samples with value } a}{n} \qquad E[A] \approx \sum_a \hat{P}(a) \cdot a$$

- Model-free approach: Compute the expected age using the samples directly

$$E[A] \approx \frac{1}{n} \sum_i a_i$$

12