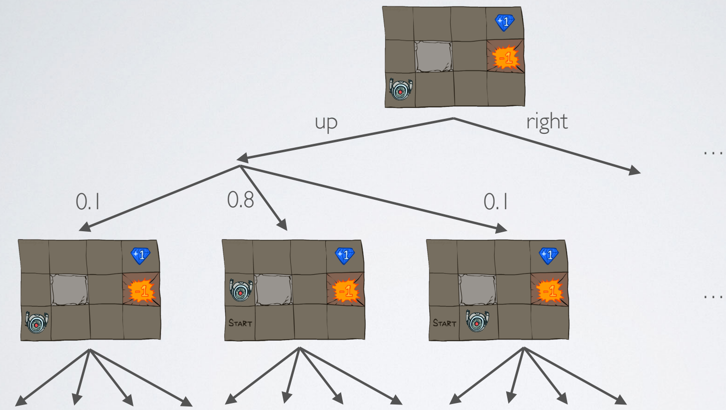
MDP SEARCH TREES

CSE 511A: Introduction to Artificial Intelligence

Some content and images are from slides created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley.
All CS188 materials are available at http://ai.berkeley.edu.

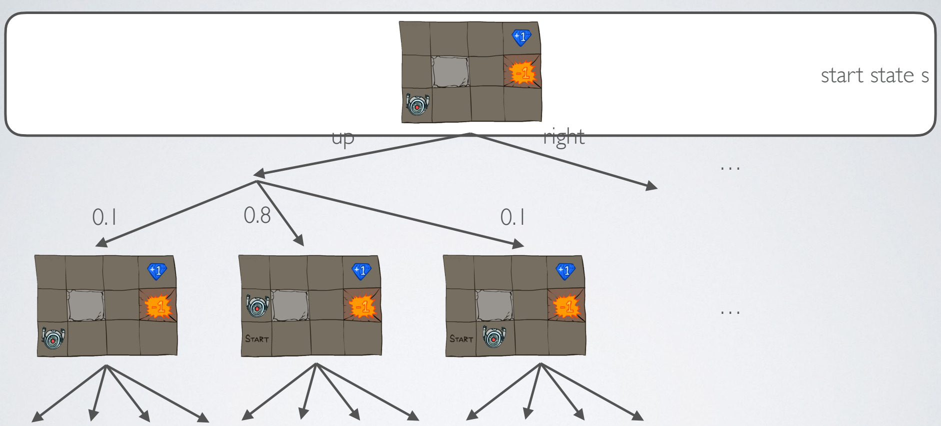
1

MDP SEARCH TREE



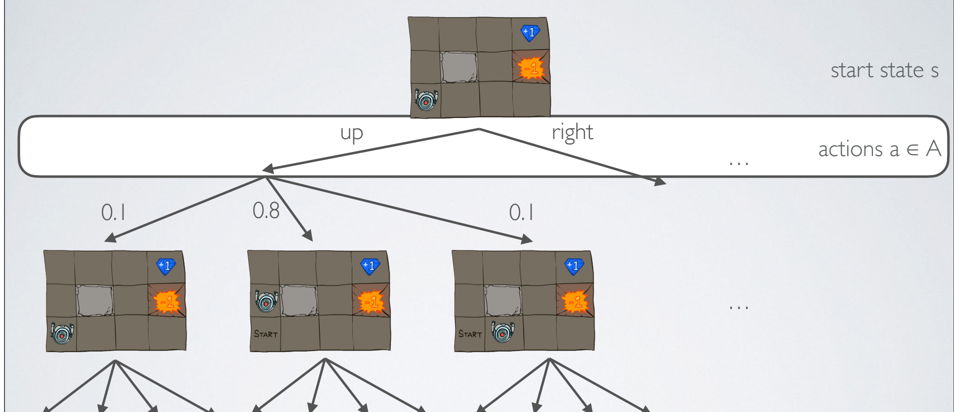
2

MDP SEARCH TREE

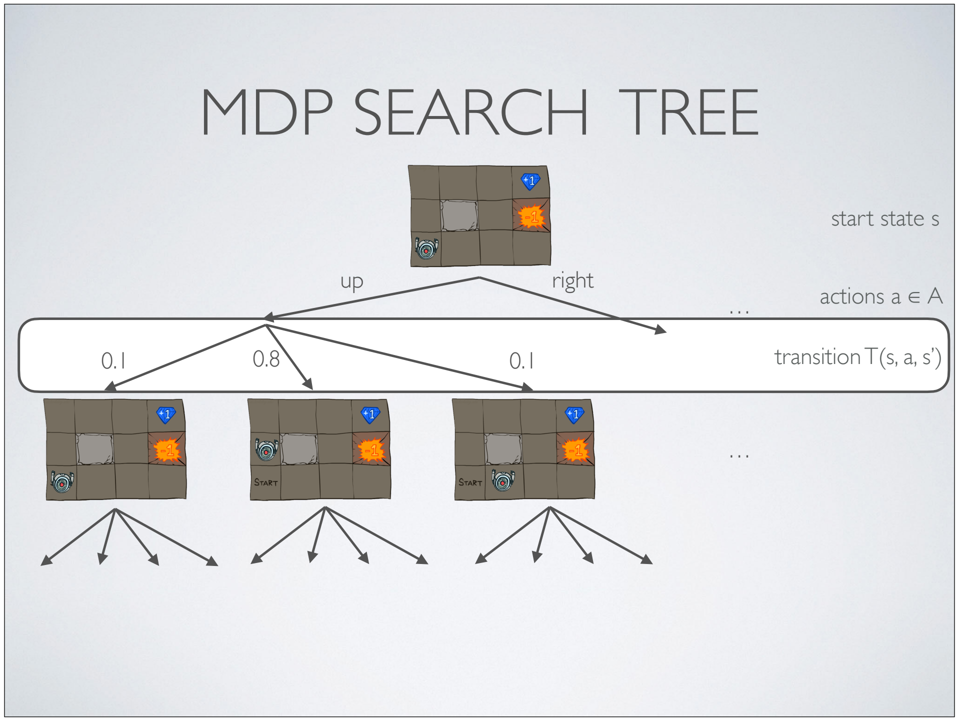


3

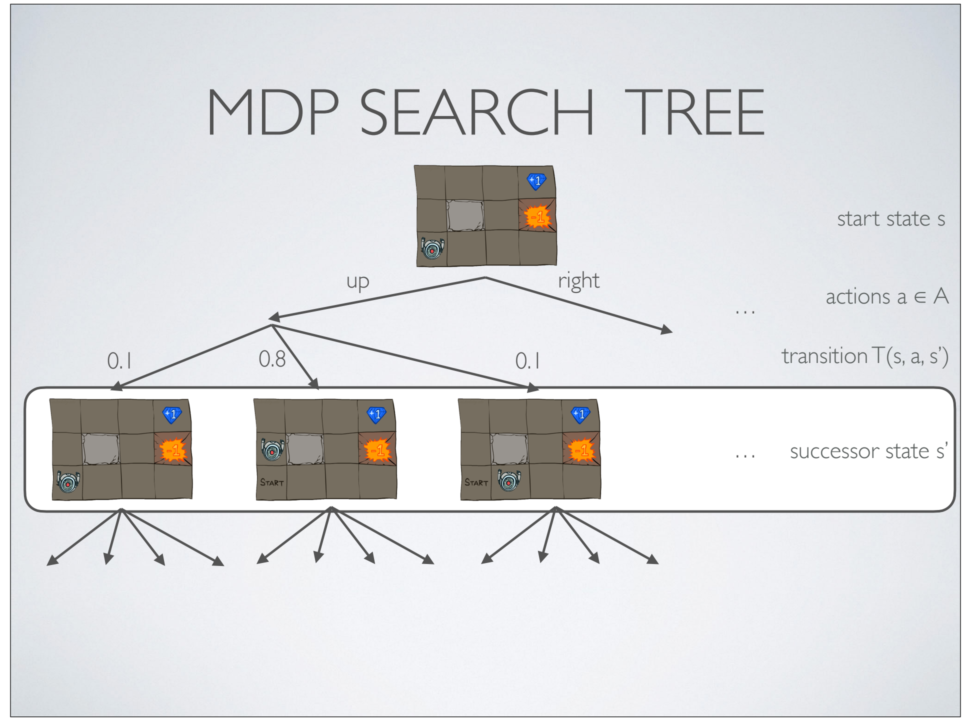
MDP SEARCH TREE



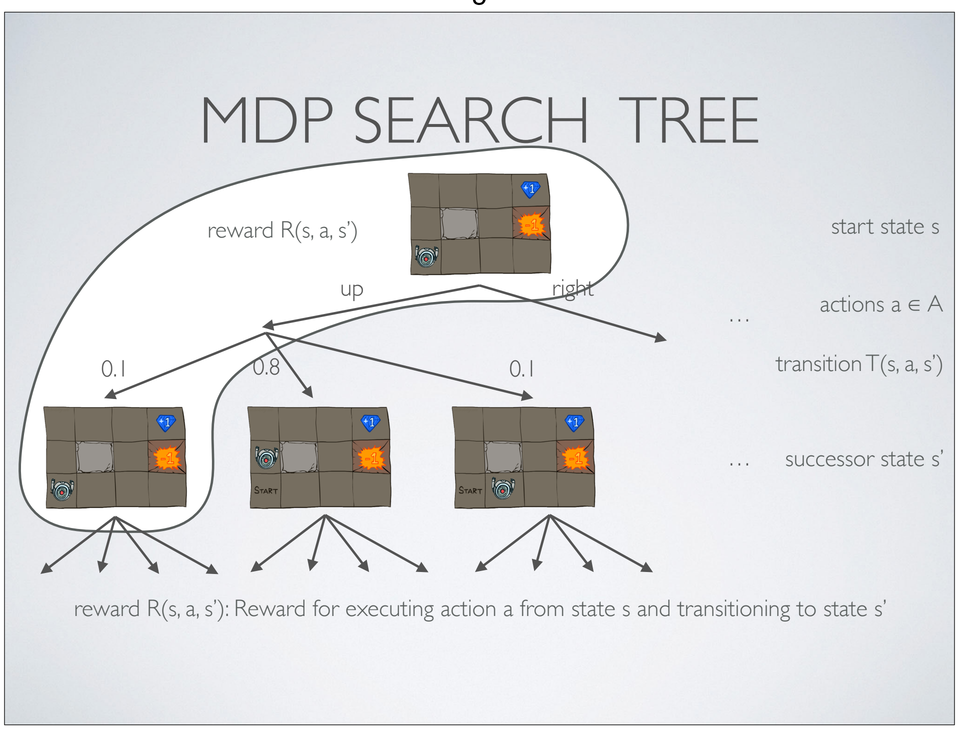
4


MDP SEARCH TREE
start state s
up
right
actions a ∈ A
0.1
0.8
0.1
transition T(s, a, s')
…

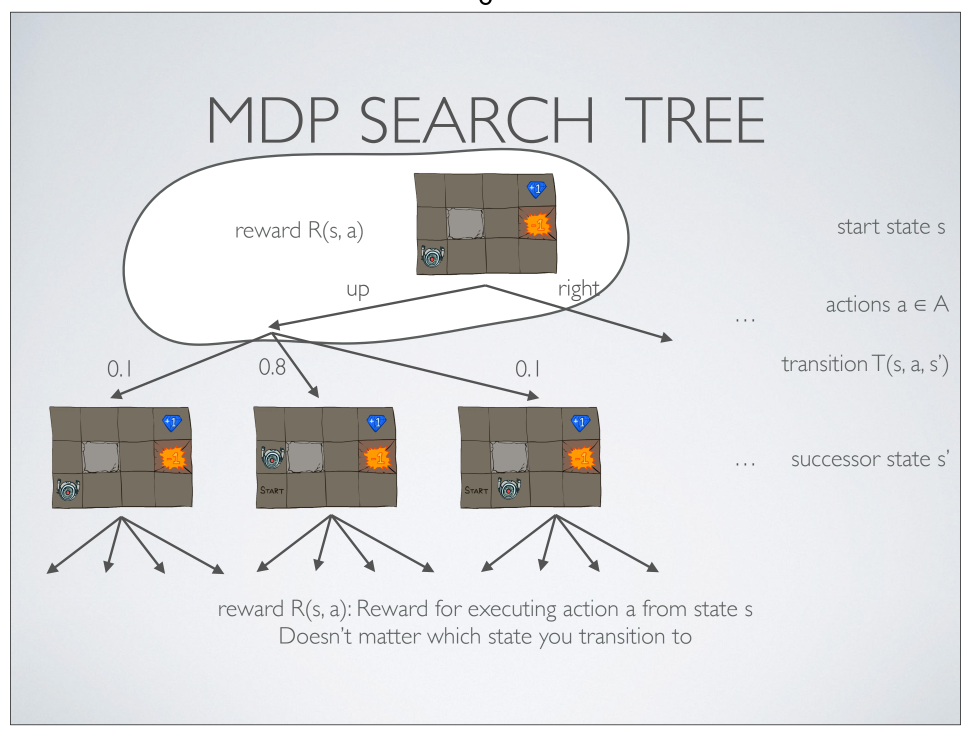
5


MDP SEARCH TREE
start state s
up
right
actions a ∈ A
0.1
0.8
0.1
transition T(s, a, s')
…
successor state s'

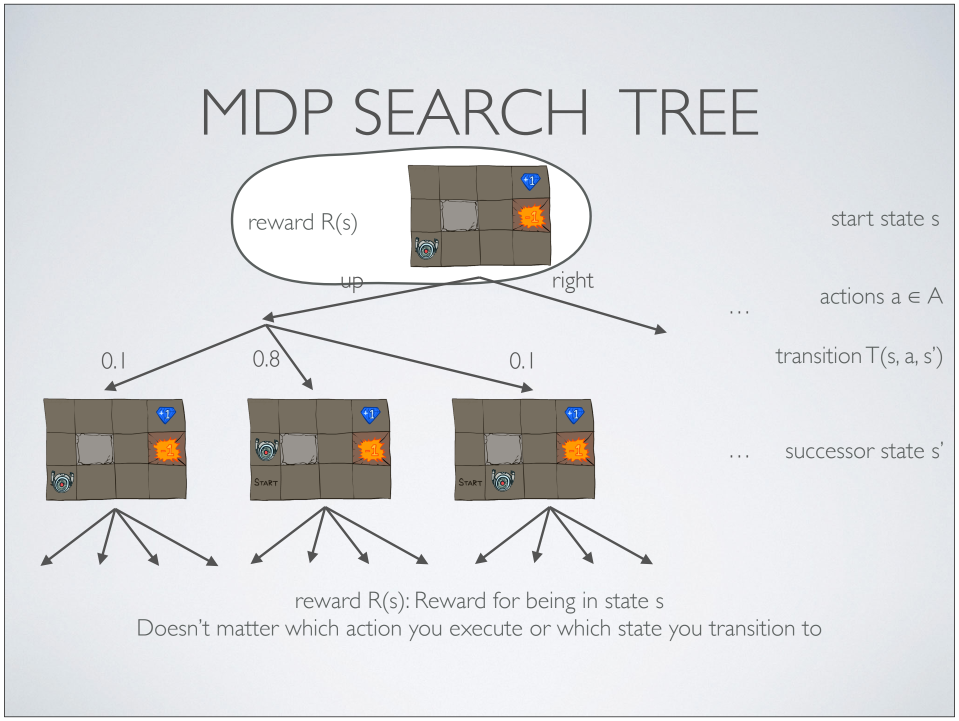
6


MDP SEARCH TREE
reward R(s, a, s')
start state s
up
right
actions a ∈ A
0.1
0.8
0.1
transition T(s, a, s')
…
successor state s'
reward R(s, a, s'): Reward for executing action a from state s and transitioning to state s'

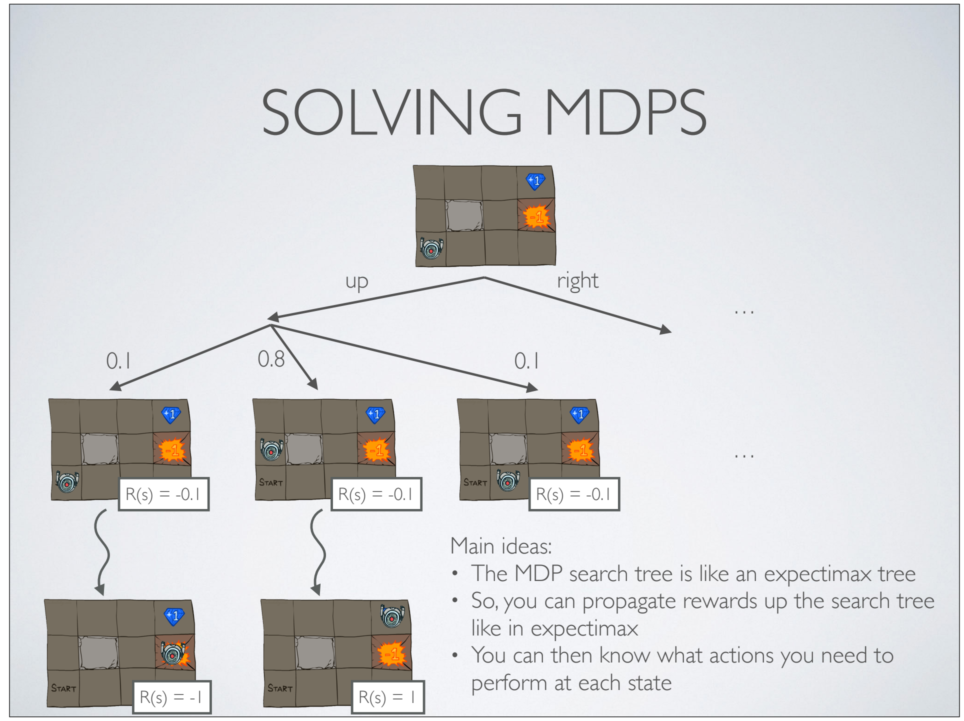
7


MDP SEARCH TREE
reward R(s, a)
start state s
up
right
actions a ∈ A
0.1
0.8
0.1
transition T(s, a, s')
…
successor state s'
reward R(s, a): Reward for executing action a from state s
Doesn't matter which state you transition to

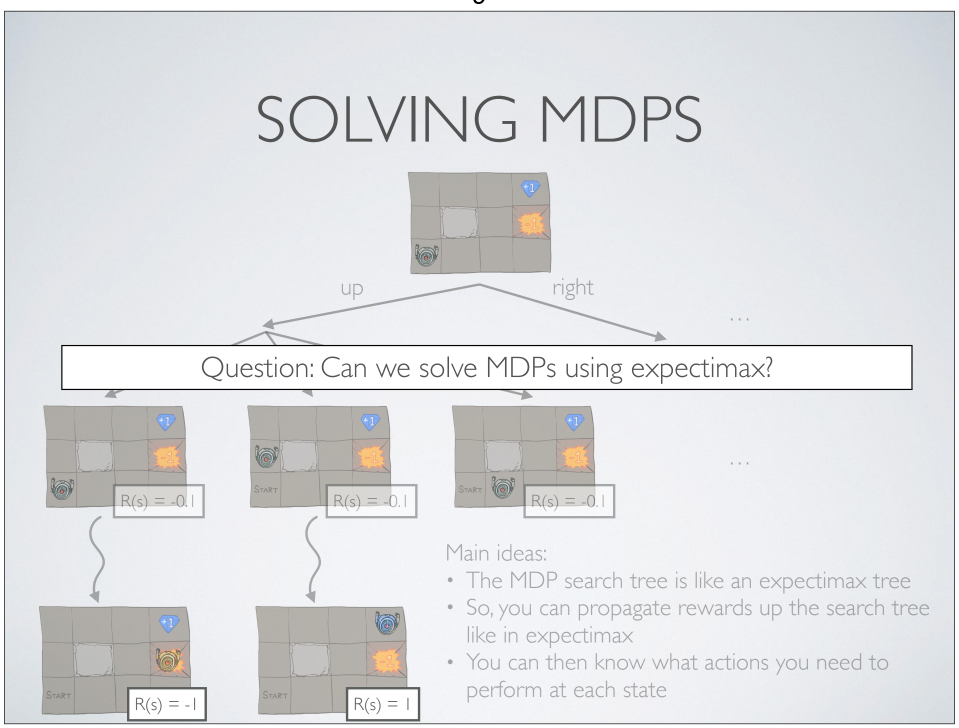
8


MDP SEARCH TREE
reward R(s)
start state s
up
right
actions a ∈ A
0.1
0.8
0.1
transition T(s, a, s')
successor state s'
reward R(s): Reward for being in state s
Doesn't matter which action you execute or which state you transition to

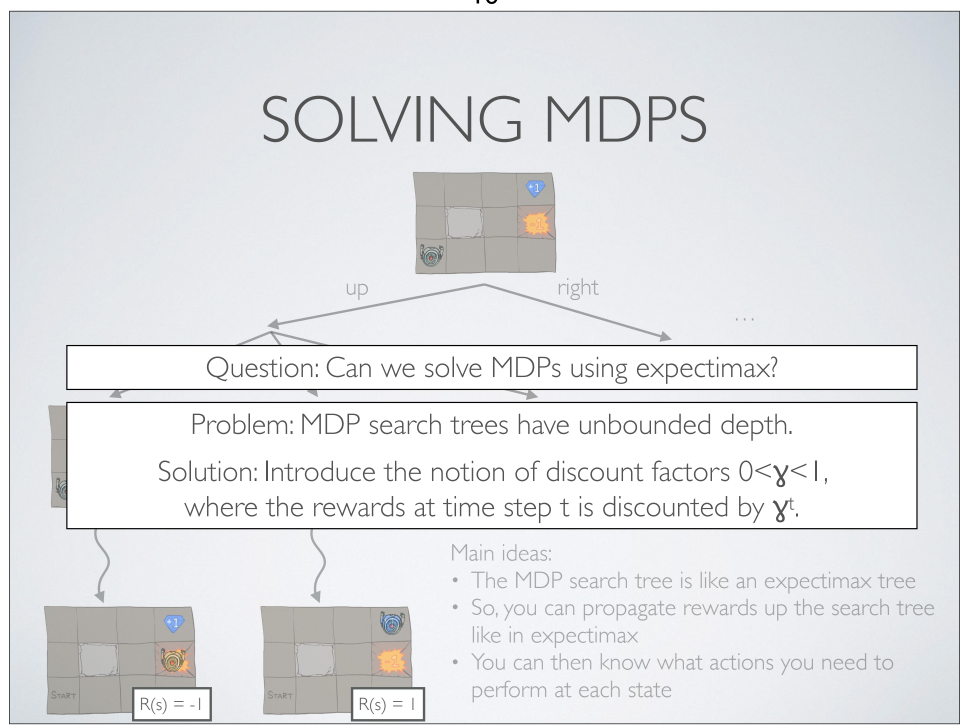
9


SOLVING MDPS
up
right
0.1
0.8
0.1
R(s) = -0.1
R(s) = -0.1
R(s) = -0.1
R(s) = -1
R(s) = 1
Main ideas:
• The MDP search tree is like an expectimax tree
• So, you can propagate rewards up the search tree like in expectimax
• You can then know what actions you need to perform at each state

10

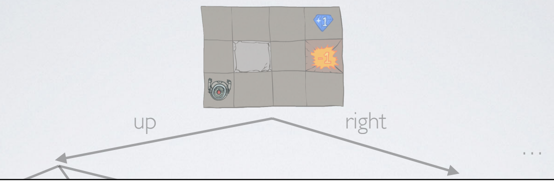

SOLVING MDPS
up
right
Question: Can we solve MDPs using expectimax?
R(s) = -0.1
R(s) = -0.1
R(s) = -0.1
R(s) = -1
R(s) = 1
Main ideas:
• The MDP search tree is like an expectimax tree
• So, you can propagate rewards up the search tree like in expectimax
• You can then know what actions you need to perform at each state

11


SOLVING MDPS
up
right
Question: Can we solve MDPs using expectimax?
Problem: MDP search trees have unbounded depth.
Solution: Introduce the notion of discount factors 0<γ<1, where the rewards at time step t is discounted by γ^t.
R(s) = -1
R(s) = 1
Main ideas:
• The MDP search tree is like an expectimax tree
• So, you can propagate rewards up the search tree like in expectimax
• You can then know what actions you need to perform at each state

12

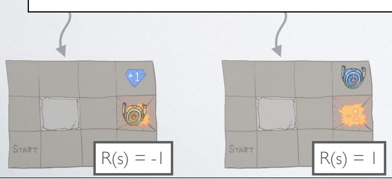
SOLVING MDPS



Question: Can we solve MDPs using expectimax?

Problem: Solution to expectimax with discount factors may suggest different actions for the same state for different branches of the tree (i.e., not an MDP policy)

Solution: Constrain the solution to be an MDP policy

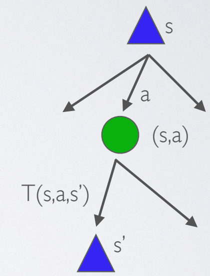


- The MDP search tree is like an expectimax tree
- So, you can propagate rewards up the search tree like in expectimax
- You can then know what actions you need to perform at each state

SOLVING MDPS

- The value (utility) of a q-state *(s,a)*:
  - *Q*(s,a)* = expected utility starting in s, taking action *a*, and thereafter acting optimally.
  - Important note: The action *a* may not be the optimal action to take at state s.
- The value (utility) of a state s:
  - *V*(s)* = expected utility starting in s and acting optimally
  - *V*(s)* = max$_a$ *Q*(s,a)*
- Optimal policy **π***
  - **π***(s) = optimal action to take at state s.
  - **π***(s) = argmax$_a$ *Q*(s,a)*

    i.e., it is the action *a* that has the largest *Q*(s,a)* over all possible actions



SOLVING MDPS

- Fundamental operation: compute the (expectimax) value of a state
  - Expected utility under optimal action
  - Average sum of discounted rewards
  - Note that this is just like what expectimax computed
- Recursive definition of value:

$$V^*(s) = \max_a Q^*(s,a)$$

$$Q^*(s,a) = \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]$$

$$V^*(s) = \max_a \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]$$