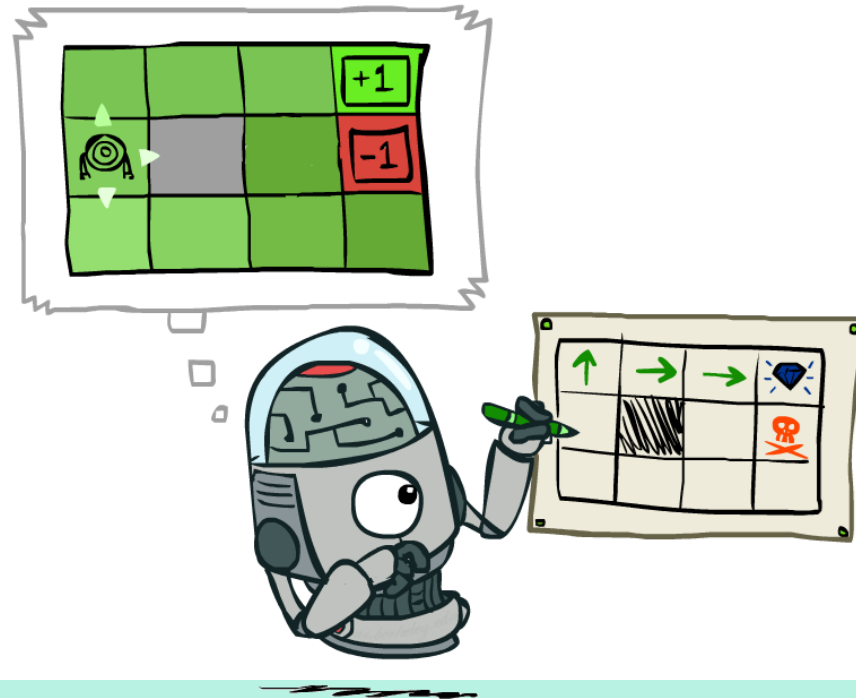


Advanced Topics in AI

Policies



Instructor: Prof. Dr. techn. Wolfgang Nejdl

Leibniz University Hannover

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All materials are available at <http://ai.berkeley.edu>.]



Computing Actions from Values

- Let's imagine we have the optimal values $V^*(s)$



- How should we act?

- It's not obvious!

- We need to do a mini-expectimax (one step)

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

- This is called **policy extraction**, since it gets the policy implied by the values

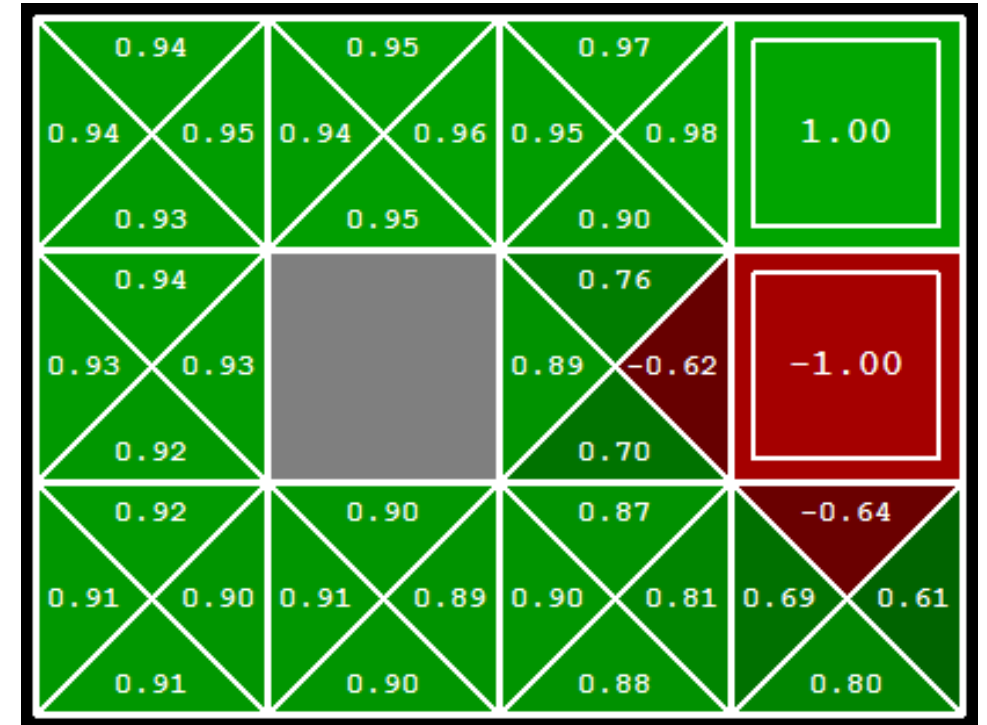
Computing Actions from Q-Values

- Let's imagine we have the optimal q-values:

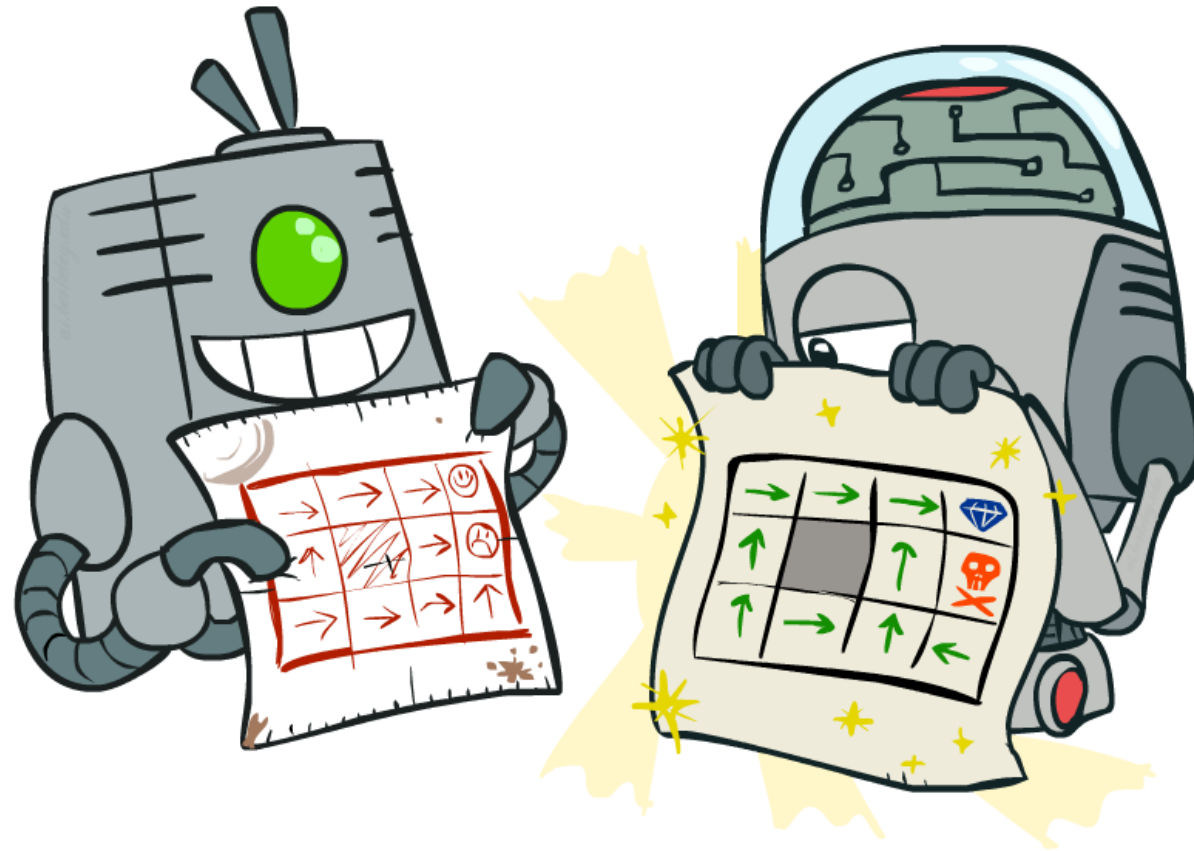
- How should we act?
 - Completely trivial to decide!

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

- Important lesson:** actions are easier to select from q-values than values!



Policy Methods

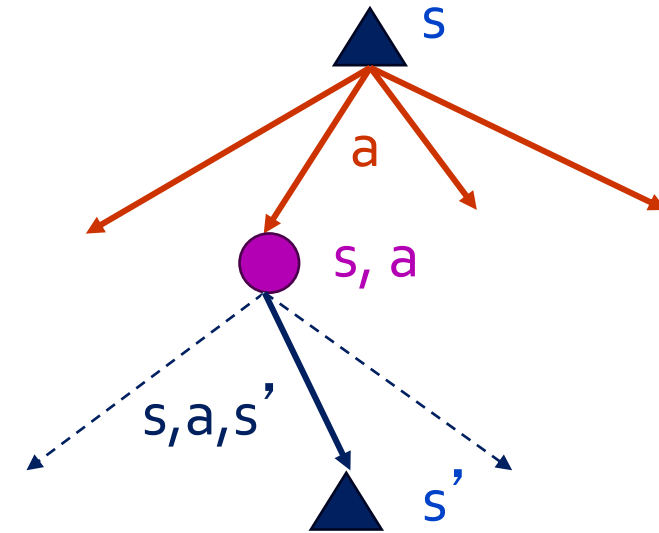


Problems with Value Iteration

- Value iteration repeats the Bellman updates:

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s, a, s') [R(s, a, s') + \gamma V_k(s')]$$

- Problem 1: It's slow – $O(S^2A)$ per iteration
- Problem 2: The “max” at each state rarely changes
- Problem 3: The policy often converges long before the values



k=12



Noise = 0.2
Discount = 0.9
Living reward = 0



k=100

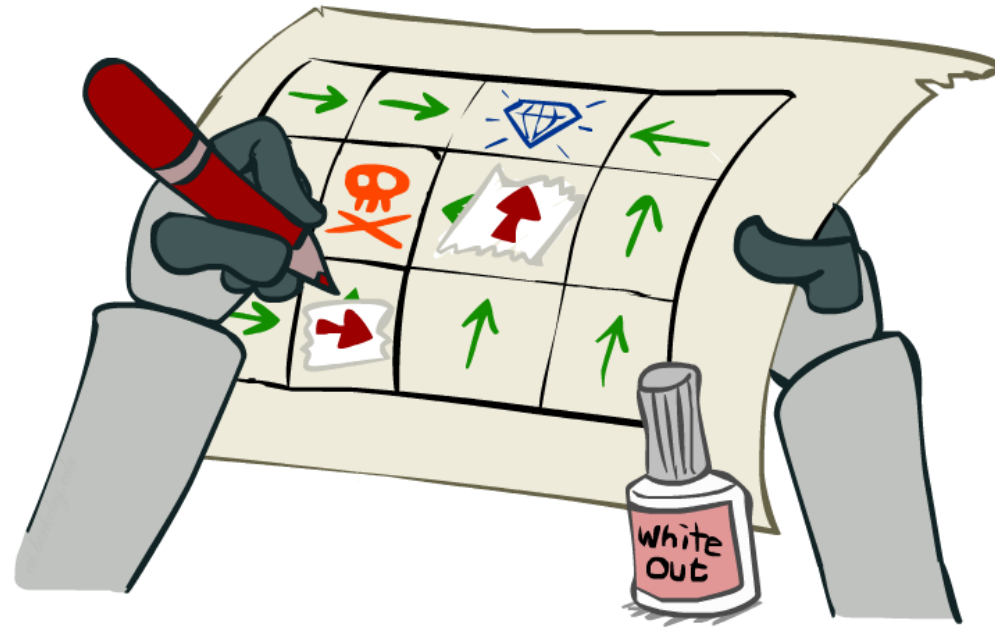


Noise = 0.2
Discount = 0.9
Living reward = 0



Advanced Topics in AI

Next: Policy Iteration



Instructor: Prof. Dr. techn. Wolfgang Nejdl

Leibniz University Hannover

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All materials are available at <http://ai.berkeley.edu>.]



Co-financed by the Connecting Europe Facility of the European Union