Reinforcement Learning Fundamentals

Lecture 15: Policy Iteration

Dr Sandeep Manjanna
Assistant Professor, Plaksha University
sandeep.manjanna@plaksha.edu.in

Announcements

- Quiz 2 will be held in class at 12:00 pm on the **28th Feb 2024.**
- Please bring pens, pencils, erasers, calculators, and a sheet for rough work**.**

In today's class…

- In-class Presentations
- Policy Evaluation
- Policy Improvement
- Policy Iteration

Policy Evaluation

- For a given policy π, compute the state value function v_π
- Recall Bellman equation for v_π:

$$v_\pi(s) = \sum_a \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a) \left[r + \gamma v_\pi(s') \right]$$

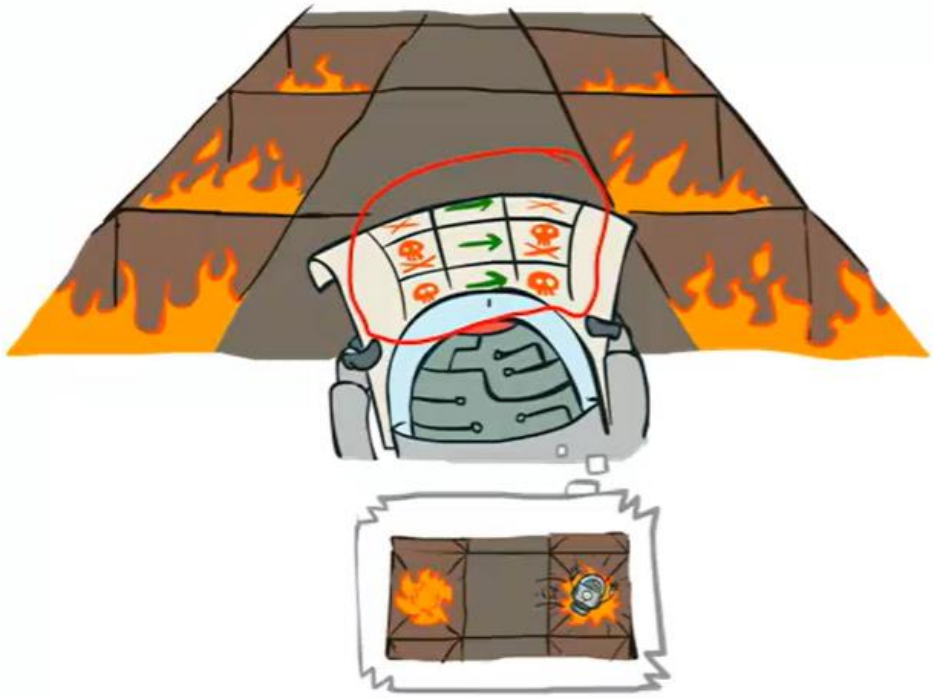
  - a system of $|S|$ simultaneous linear equations
  - solve iteratively **?**

First solve for zero step problem, use that to solve the 1-step problem, in-turn use that to solve the 2-step problem… following the policy π

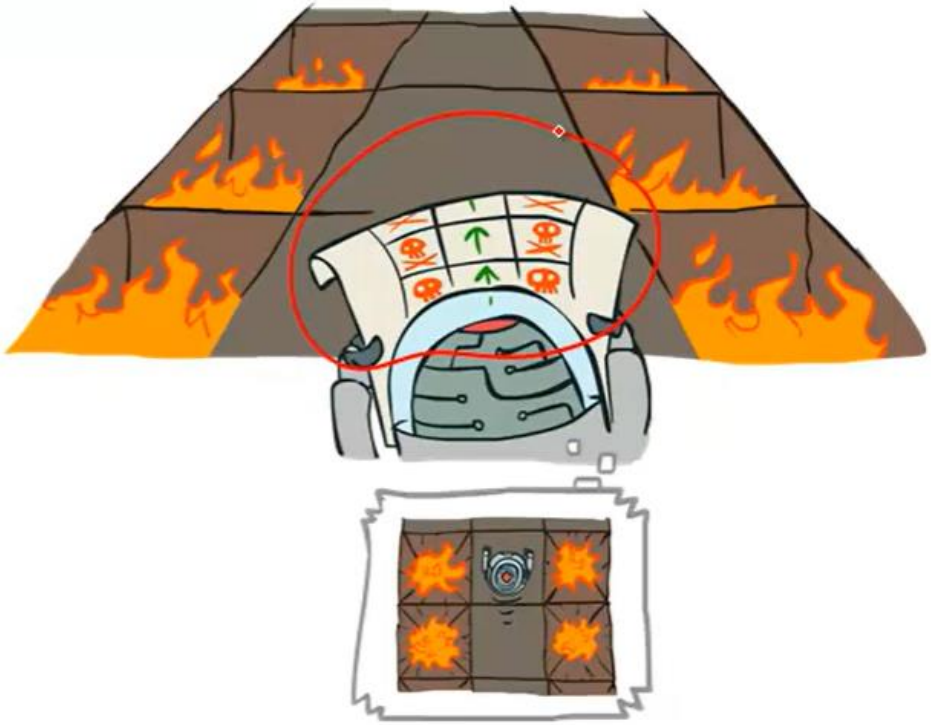
Slide from Prof. Ravindran's course: "Reinforcement Learning."

Policy Evaluation Example

Always Go Right

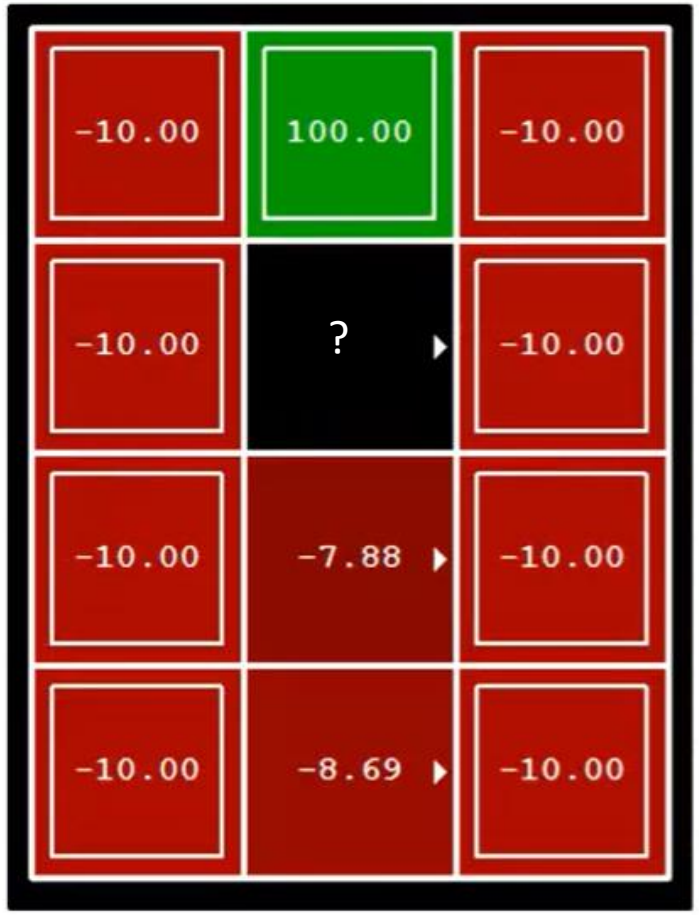


Always Go Forward

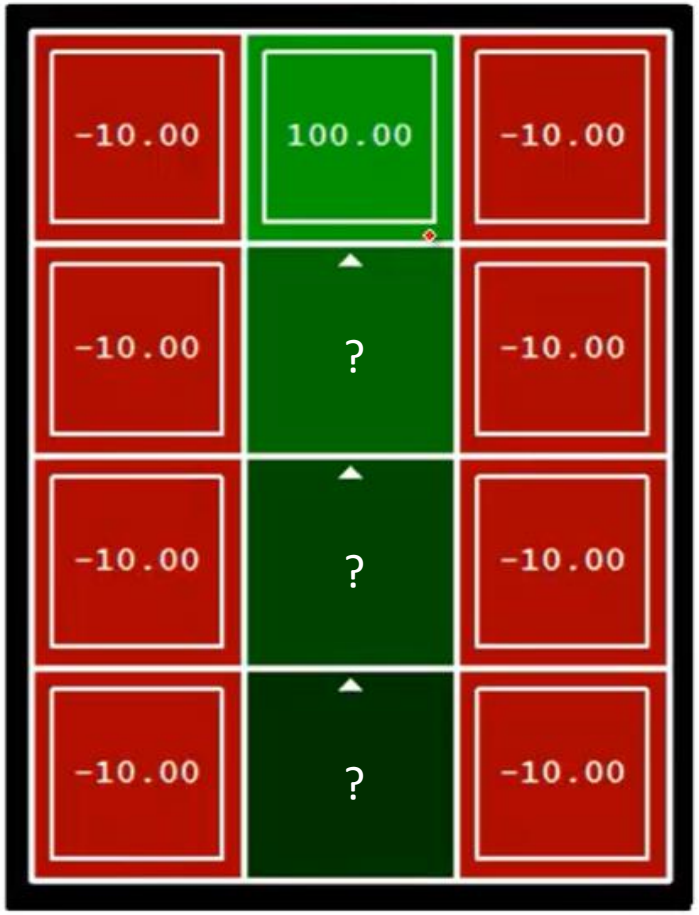


Policy Evaluation Example

Always Go Right



Always Go Forward



Policy Evaluation Example

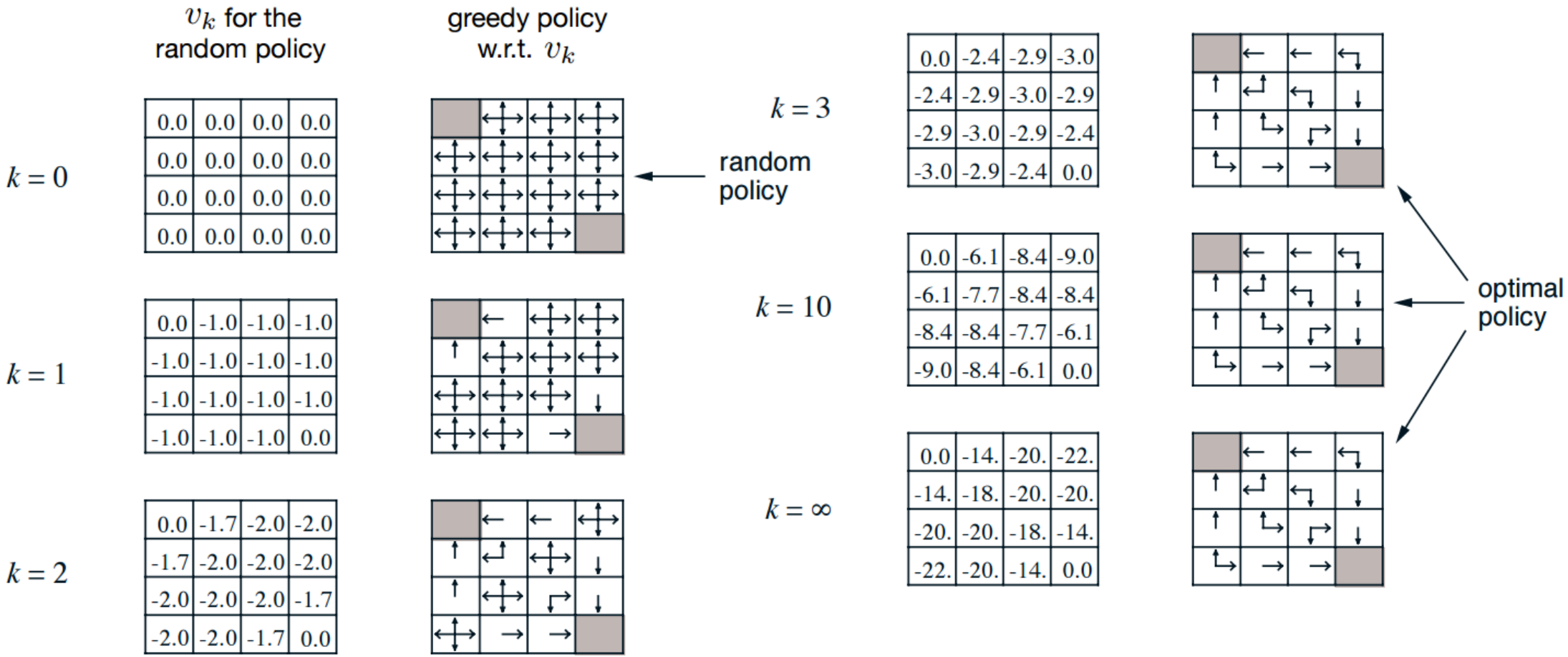


Figure 4.1: Convergence of iterative policy evaluation on a small gridworld. The left column is the sequence of approximations of the state-value function for the random policy (all actions equally likely). The right column is the sequence of greedy policies corresponding to the value function estimates (arrows are shown for all actions achieving the maximum, and the numbers shown are rounded to two significant digits). The last policy is guaranteed only to be an improvement over the random policy, but in this case it, and all policies after the third iteration, are optimal.

Policy Improvement

Finite MDPs have at least one optimal deterministic policy.

- Suppose we have computed v_π for an arbitrary deterministic policy π
- For a given state s, would it be better to choose an action $a \neq \pi(s)$?
- The value of doing a in state s is:

$$\begin{aligned} q_\pi(s,a) &\doteq \mathbb{E}[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s, A_t = a] \\ &= \sum_{s',r} p(s', r \mid s, a)\Big[r + \gamma v_\pi(s')\Big]. \end{aligned}$$

- It is better to switch to action a for state s if and only if

$$q_\pi(s,a) > v_\pi(s)$$

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Policy Improvement

Do this for all states to get a new policy π' that is **greedy** with respect to v_π:

$$\pi'(s) = \arg\max_a q_\pi(s,a)$$
$$= \arg\max_a \sum_{s',r} p(s',r \mid s,a)\,[r + \gamma v_\pi(s')]$$

Then, $v_{\pi'} \geq v_\pi$

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Policy Iteration

What if $v_{\pi'} = v_\pi$? Then, for all $s, \in \mathcal{S}$, we have

$$v_{\pi'}(s) = \max_a \sum_{s', r} p(s', r \mid s, a) \left[r + \gamma v_\pi(s') \right]$$

But this is the Bellman Optimality equation.
So $v_{\pi'} = v_*$ and both π and π' are optimal policies.

$$\pi_0 \xrightarrow{\text{E}} v_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} v_{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} v_*$$

policy evaluation

policy improvement
"greedification"

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Policy Iteration

- Alternative approach for optimal values:
  - Step 1: Policy evaluation: calculate utilities for some fixed policy (not optimal utilities!) until convergence
  - Step 2: Policy improvement: update policy using one-step look-ahead with resulting converged (but not optimal!) utilities as future values
  - Repeat steps until policy converges

Policy Iteration

1. Initialization
   $V(s) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$

2. Policy Evaluation
   Loop:
   $\Delta \leftarrow 0$
   Loop for each $s \in \mathcal{S}$:
   $v \leftarrow V(s)$
   $V(s) \leftarrow \sum_{s',r} p(s', r \mid s, \pi(s))\left[r + \gamma V(s')\right]$
   $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
   until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)

Implement Policy Iteration on a simple grid-world example and compare the effect of discount factor, size of the world, convergence threshold, reward function, and transition probability on the values achieved.

3. Policy Improvement
   policy-stable $\leftarrow$ *true*
   For each $s \in \mathcal{S}$:
   old-action $\leftarrow \pi(s)$
   $\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma V(s')\right]$
   If *old-action* $\neq \pi(s)$, then *policy-stable* $\leftarrow$ *false*
   If *policy-stable*, then stop and return $V \approx v_*$ and $\pi \approx \pi_*$; else go to 2

Break ties consistently