Lecture 2b: Policy Iteration
CS885 Reinforcement Learning

2025-01-09

Complementary readings: [SutBar] Sec. 4.3, [Put] Sec. 6.4-6.5, [SigBuf] Sec. 1.6.2.3, [RusNor] Sec. 17.3

Pascal Poupart
David R. Cheriton School of Computer Science

UNIVERSITY OF WATERLOO

Policy Optimization

- Value iteration
  - Optimize value function
  - Extract induced policy

- Can we directly optimize the policy?
  - Yes, by policy iteration

Policy Iteration

- Alternate between two steps

1. Policy evaluation

$$V^{\pi}(s) = R\big(s,\pi(s)\big) + \gamma \sum_{s'} \Pr(s'|s,\pi(s))\, V^{\pi}(s') \;\; \forall s$$

2. Policy improvement

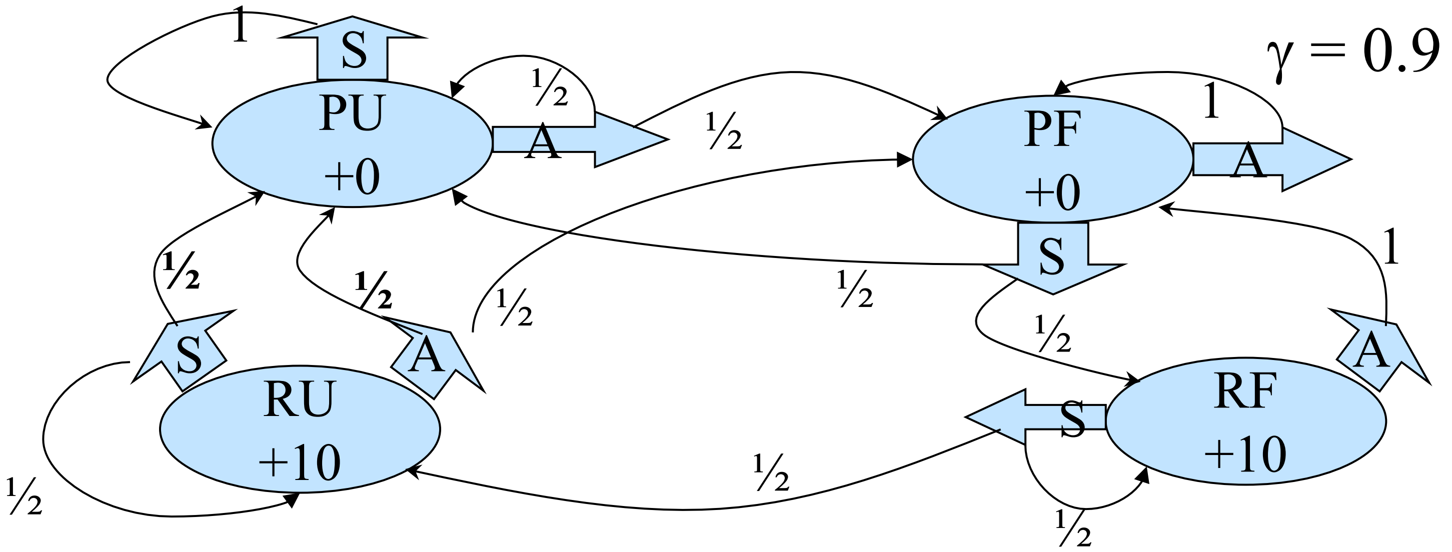
$$\pi(s) \leftarrow \operatorname*{argmax}_{a} R(s,a) + \gamma \sum_{s'} \Pr(s'|s,a)\, V^{\pi}(s') \;\; \forall s$$

Algorithm

policyIteration(MDP)

Initialize π_0 to any policy
$n \leftarrow 0$
Repeat
  Eval: $V_n = R^{\pi_n} + \gamma T^{\pi_n} V_n$
  Improve: $\pi_{n+1} \leftarrow argmax_a \ R^a + \gamma T^a V_n$
  $n \leftarrow n + 1$
Until $\pi_{n+1} = \pi_n$
Return π_n

Example (Policy Iteration)



n	$V(PU)$	$\pi(PU)$	$V(PF)$	$\pi(PF)$	$V(RU)$	$\pi(RU)$	$V(RF)$	$\pi(RF)$
0	0	A	0	A	10	A	10	A
1	31.6	A	38.6	S	44.0	S	54.2	S
2	31.6	A	38.6	S	44.0	S	54.2	S

Monotonic Improvement

- Lemma 1: Let V_n and V_{n+1} be successive value functions in policy iteration. Then $V_{n+1} \geq V_n$.
- Proof:
  - We know that $H^*(V_n) \geq H^{\pi_n}(V_n) = V_n$
  - Let $\pi_{n+1} = argmax_a \, R^a + \gamma T^a V_n$
  - Then $H^*(V_n) = R^{\pi_{n+1}} + \gamma T^{\pi_{n+1}} V_n \geq V_n$
  - Rearranging: $R^{\pi_{n+1}} \geq (I - \gamma T^{\pi_{n+1}}) V_n$
  - Hence $V_{n+1} = (I - \gamma T^{\pi_{n+1}})^{-1} R^{\pi_{n+1}} \geq V_n$

Convergence

- Theorem 2: Policy iteration converges to π^* & V^* in finitely many iterations when S and A are finite.

- Proof:
  - We know that $V_{n+1} \geq V_n \ \forall n$ by Lemma 1. Consider a stronger version of Lemma 1 where $\exists s$ such that $V_{n+1}(s) > V_n(s)$ unless V_n is optimal
  - Since A and S are finite, there are finitely many policies and therefore the algorithm terminates in finitely many iterations.
  - At termination, $V_n = V_{n+1}$ and therefore V_n satisfies Bellman's equation:

$$V_n = V_{n+1} = \max_a R^a + \gamma T^a V_n$$

UNIVERSITY OF WATERLOO

Complexity

- Value Iteration:
  - Each iteration: $O(|S|^2|A|)$
  - Many iterations: linear convergence

- Policy Iteration:
  - Each iteration: $O(|S|^3 + |S|^2|A|)$
  - Few iterations: linear-quadratic convergence

Modified Policy Iteration

- Alternate between two steps

1. Partial Policy evaluation

   Repeat k times:

   $$V^\pi(s) \leftarrow R\big(s,\pi(s)\big) + \gamma \textstyle\sum_{s'} \Pr(s'|s,\pi(s))\, V^\pi(s') \;\; \forall s$$

2. Policy improvement

   $$\pi(s) \leftarrow \operatorname*{argmax}_a R(s,a) + \gamma \sum_{s'} \Pr(s'|s,a)\, V^\pi(s') \;\; \forall s$$

Algorithm

modifiedPolicyIteration(MDP)

Initialize π_0 and V_0 to anything

$n \leftarrow 0$

Repeat

Eval: Repeat k times

$V_n \leftarrow R^{\pi_n} + \gamma T^{\pi_n} V_n$

Improve: $\pi_{n+1} \leftarrow argmax_a \; R^a + \gamma T^a V_n$

$V_{n+1} \leftarrow max_a \; R^a + \gamma T^a V_n$

$n \leftarrow n + 1$

Until $||V_n - V_{n-1}||_\infty \leq \epsilon$

Return π_n

Convergence

- Same convergence guarantees as value iteration:
  - Value function V_n: $||V_n - V^*||_\infty \leq \frac{\epsilon}{1-\gamma}$
  - Value function V^{π_n} of policy π_n:

$$||V^{\pi_n} - V^*||_\infty \leq \frac{2\epsilon}{1-\gamma}$$

- Proof: somewhat complicated
  (see Section 6.5 of Puterman's book)

Complexity

- Value Iteration:
  - Each iteration: $O(|S|^2|A|)$
  - Many iterations: linear convergence
- Policy Iteration:
  - Each iteration: $O(|S|^3 + |S|^2|A|)$
  - Few iterations: linear-quadratic convergence
- Modified Policy Iteration:
  - Each iteration: $O(k|S|^2 + |S|^2|A|)$
  - Few iterations: linear-quadratic convergence