

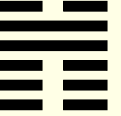
Competitive Reinforcement Learning

Peter Auer

CiT, University of Leoben



Questions we will deal with



- How can a learner choose nearly optimal actions online?
- How much does a learner lose compared to an optimal policy, when choosing actions online?
- How to deal with the exploration/exploitation trade-off?

Some answers:

- The method of upper confidence bounds does the trick.
- We lose $O(\log m)$ of the rewards in m episodes.
- We get a new reinforcement learning algorithm.

Why competitive online reinforcement learning?



- Typical RL algorithms converge to a (nearly) optimal policy during a designated learning phase.
- Exploration costs during the learning phase are not taken into account.
- We want to trade off the learning costs and the quality of the learned/chosen actions.
- A natural model to do so is the online learning model:
 - > It does not distinguish between learning and execution phase.
 - > It measures the regret relative to an optimal policy, during the lifetime of the agent.



We consider a Markov decision process (MDP) on a finite set of states S with a finite set of actions A available in each state.

- There is an initial state s_0 .
- The transition probability $p_a(s_t, s_{t+1})$ is the probability of reaching state s_{t+1} when choosing action a in state s_t .
- The reward for choosing action a in state s has mean $r_a(s)$. (We assume that rewards are bounded in $[0, 1]$.)

Goal: Maximize the sum of rewards.



Let r_t^π be the expected reward at step t when following strategy π . Then the average reward of strategy π after T steps equals

$$\bar{r}(\pi, T) = \frac{1}{T} \sum_{t=1}^T r_t^\pi .$$

Let T be fixed and let π^* be a strategy which maximizes $\bar{r}(\pi, T)$. Then the expected regret of a strategy π is given by

$$\text{regret}^\pi = \bar{r}(\pi^*, T) - \bar{r}(\pi, T) .$$

Episodic reinforcement learning



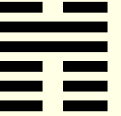
We consider episodes of length T such that the MDP is restarted in state s_0 after T steps.

We bound the regret over M episodes and get

$$\sum_{m=1}^M \text{regret}_m^{\mathcal{A}} = O(\text{poly}(T) \cdot \log M)$$

where $\text{regret}_m^{\mathcal{A}}$ denotes the regret of our algorithm \mathcal{A} in episode m .

Why episodes?



It simplifies the analysis considerably. (Right now we cannot do it without resets to the start state s_0 .)

If there are no restarts then mixing times need to be considered: How fast can an arbitrary state be reached in the MDP?

Instead of

$$O(\text{poly}(T) \cdot \log M)$$

we expect a regret bound

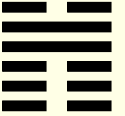
$$O(\text{poly}(T_{mix}) \cdot \log T) .$$

Policy search as a multi-armed bandit problem



- For finite MDPs we have a finite (but exponential in the parameters of the MDP) number of policies/arms.
 - Each episode can be seen as a trial in a multi-armed bandit problem.
 - The regret for the multi-armed bandit problem scales like $O(\log M)$, but with a linear dependency in the number of arms.
- ⇒ To get a polynomial dependency on the parameters of the MDP, we need to calculate information about policies from information about states.

The multi-armed (random) bandit problem

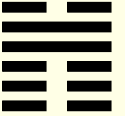


- There are K arms x_1, \dots, x_K .
- In each step t each arm would give independent payoff $x_i(t) \in [0, 1]$.
- The average payoff for each arm is stationary, $\mathbf{E}[x_i(t)] = \mu_i$.
- It is like an MDP with a single state and K actions.
- **Goal:** Minimize the regret.

Solutions go back to

[Robbins, 1952], [Lai, Robbins, 1985], [Agarwal, 1995].

Algorithm: Uses upper confidence bounds



1. Calculate an estimate $\hat{\mu}_i(t)$ for the average payoff of each arm.
2. Calculate a confidence interval such that

$$\mu_i \in [\hat{\mu}_i(t) - \sigma_i(t), \hat{\mu}_i(t) + \sigma_i(t)]$$

with probability $1/T$.

3. Choose alternative $i(t)$ which maximizes the upper confidence bound $\hat{\mu}_i(t) + \sigma_i(t)$.

Algorithm: Uses upper confidence bounds



1. Calculate an estimate $\hat{\mu}_i(t)$ for the average payoff of each arm.
2. Calculate a confidence interval such that

$$\mu_i \in [\hat{\mu}_i(t) - \sigma_i(t), \hat{\mu}_i(t) + \sigma_i(t)]$$

with probability $1/T$.

3. Choose alternative $i(t)$ which maximizes the upper confidence bound $\hat{\mu}_i(t) + \sigma_i(t)$.
- We can use $\sigma_i(t) \sim \sqrt{\frac{\log T}{n_i(t)}}$ where $n_i(t)$ is the number of steps in which arm i was selected.

Algorithm: Uses upper confidence bounds



1. Calculate an estimate $\hat{\mu}_i(t)$ for the average payoff of each arm.
2. Calculate a confidence interval such that

$$\mu_i \in [\hat{\mu}_i(t) - \sigma_i(t), \hat{\mu}_i(t) + \sigma_i(t)]$$

with probability $1/T$.

3. Choose alternative $i(t)$ which maximizes the upper confidence bound $\hat{\mu}_i(t) + \sigma_i(t)$.
- We can use $\sigma_i(t) \sim \sqrt{\frac{\log T}{n_i(t)}}$ where $n_i(t)$ is the number of steps in which arm i was selected.
 - Using upper confidence bounds automatically trades off exploration and exploitation!

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t)$$

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$\mu_i + 2\sigma_i(t) \geq p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t)$$

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$\mu_i + 2\sigma_i(t) \geq p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t) \geq \mu^*.$$

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$\mu_i + 2\sigma_i(t) \geq p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t) \geq \mu^*.$$

Thus $i(t) = i$ only if $\sigma_i(t) \geq \Delta_i/2$.

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$\mu_i + 2\sigma_i(t) \geq p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t) \geq \mu^*.$$

Thus $i(t) = i$ only if $\sigma_i(t) \geq \Delta_i/2$.

Since $\sigma_i(t) \sim \sqrt{\frac{\log T}{n_i(t)}}$ we have $n_i(t) \leq O\left(\frac{\log T}{\Delta_i^2}\right)$,

Analysis



Let $\mu^* = \max_i \mu_i$, $\Delta_i = \mu^* - \mu_i$. Then

$$\text{regret} \leq \sum_{i=1}^K \Delta_i \mathbf{E} [n_i(T)].$$

Furthermore, $i(t) = i$ only if

$$\mu_i + 2\sigma_i(t) \geq p_i(t) + \sigma_i(t) \geq p^*(t) + \sigma^*(t) \geq \mu^*.$$

Thus $i(t) = i$ only if $\sigma_i(t) \geq \Delta_i/2$.

Since $\sigma_i(t) \sim \sqrt{\frac{\log T}{n_i(t)}}$ we have $n_i(t) \leq O\left(\frac{\log T}{\Delta_i^2}\right)$, and hence

$$\text{regret} \leq O\left((\log T) \sum_{i=1:i \neq i^*}^K \frac{1}{\Delta_i}\right).$$



1. Let $\tilde{r}(\pi, T)$ be an upper confidence estimate for the average reward $\bar{r}(\pi, T)$.
2. In each episode m choose strategy π_m which maximizes the upper confidence bound $\tilde{r}_m(\pi, T)$.

Previous and related work:

[Kaelbling, 1993], [Fiechter, 1994],
[Burnetas & Katehakis, 1997],
[Kearns & Singh, 2002 (E^3)],
[Brafman & Tennenholtz, 2002 (R-max)],
[Even-Dar, Kakade, & Mansour, 2004],
[Strehl & Littman, 2004, 2005].

Why it works (1)



Let π^* be an optimal T -step strategy. Then

$$\tilde{r}_m(\pi_m, T) \geq \tilde{r}_m(\pi^*, T) \geq \bar{r}(\pi^*, T) \geq \bar{r}(\pi_m, T).$$

Let

$$\Delta = \min\{\bar{r}(\pi^*, T) - \bar{r}(\pi, T) : \bar{r}(\pi^*, T) \neq \bar{r}(\pi, T)\}$$

be the minimal advantage of π^* over any other non-optimal strategy. Then

$$\Delta > \tilde{r}_m(\pi_m, T) - \bar{r}(\pi_m, T) \geq \bar{r}(\pi^*, T) - \bar{r}(\pi_m, T)$$

implies that π_m is already an optimal strategy.

Why it works (2)



Thus if B is a bound on the number of episodes with

$$\tilde{r}_m(\pi_m, T) - \bar{r}(\pi_m, T) \geq \Delta,$$

we get

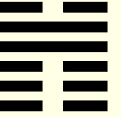
$$\text{regret} \leq B$$

(since the average reward in each episode is at most 1).

Basic principle:

If a non-optimal policy is executed, then some new information about the MDP is collected and the upper confidence estimates can be improved.

Why it works (3)



Let $\tilde{\mathcal{M}}_m$ be the modified MDP which gives the upper confidence bound $\tilde{r}_m(\pi_m, T)$.

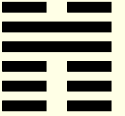
Let $v_m(s)$ and $\tilde{v}_m(s)$ be the expected number of visits to state s when following π_m , according to the original MDP and $\tilde{\mathcal{M}}_m$, respectively. Then

$$\bar{r}(\pi_m, T) = \frac{1}{T} \sum_s v_m(s) r_m(s), \quad \tilde{r}_m(\pi_m, T) = \frac{1}{T} \sum_s \tilde{v}_m(s) \tilde{r}_m(s),$$

and we get

$$\begin{aligned} & \tilde{r}_m(\pi_m, T) - \bar{r}(\pi_m, T) \\ &= \frac{1}{T} \sum_s [\tilde{v}_m(s) - v_m(s)] \tilde{r}_m(s) + \frac{1}{T} \sum_s v_m(s) [\tilde{r}_m(s) - r_m(s)] \end{aligned}$$

Why it works (4): Bounding the regret estimates



The second sum shows that an inaccurate estimate of $r_m(s)$ only matters if state s is actually visited, which implies that the estimate is improved:

Since $\tilde{r}_m(s) - r_m(s) \leq O\left(\sqrt{\frac{\log(M)}{n_m(s, \pi_m)}}\right)$ we get

$$\frac{1}{T} \sum_s v_m(s) [\tilde{r}_m(s) - r_m(s)] \geq \Delta/2$$

for at most $O\left(\frac{|S||A|}{\Delta^2} \log(M)\right)$ episodes.

Why it works (5): Considering transitions



Let $v_m^t(s)$ be the number of visits until step t . Then (for $s \neq s_0$)

$$v_m^{t+1}(s) = \sum_{s'} v_m^t(s') p_m(s', s).$$

We show that $\frac{1}{T} \sum_s [\tilde{v}_m(s) - v_m(s)] \tilde{r}_m(s) \geq \Delta/2$ only if there are states s and s' with

$$v_m(s) \geq \Delta_v \quad \text{and} \quad \tilde{p}_m(s, s') - p_m(s, s') \geq \Delta_p$$

for appropriate Δ_v, Δ_p . Then the number of such episodes is bounded by $O\left(\frac{|S||A|}{\Delta_v \Delta_p^2} \log(M)\right)$.

Why it works (6): Induction



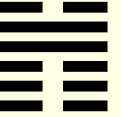
Assume to the contrary, that $v_m^t(s) \geq \Delta_v$ implies $\tilde{p}_m(s, s') - p_m(s, s') < \Delta_p$ for all s' . Then we show that for any $\rho(s) \in [0, 1]$

$$\sum_s [\tilde{v}_m^t(s) - v_m^t(s)] \rho(s) \leq t|S|^2 [\Delta_v + T\Delta_p].$$

Thus $\Delta_v = \frac{\Delta}{4|S|^2}$ and $\Delta_p = \frac{\Delta}{4T|S|^2}$ yields

$$\frac{1}{T} \sum_s [\tilde{v}_m(s) - v_m(s)] \tilde{r}_m(s) \leq \Delta/2 .$$

Why it works (7): Induction step



We have

$$\begin{aligned} & \sum_{s'} [\tilde{v}_m^{t+1}(s') - v_m^{t+1}(s')] \rho(s') \\ &= \sum_{s'} \sum_s [\tilde{v}_m^t(s) \tilde{p}_m(s, s') - v_m^t(s) p_m(s, s')] \rho(s') \\ &\leq \sum_{s'} \sum_{s: v_m^t(s) < \Delta_v} [\tilde{v}_m^t(s) \tilde{p}_m(s, s') - v_m^t(s) \tilde{p}_m(s, s')] \rho(s') + |S|^2 \Delta_v \\ &\quad + \sum_{s'} \sum_{s: v_m^t(s) \geq \Delta_v} [\tilde{v}_m^t(s) p_m(s, s') - v_m^t(s) p_m(s, s')] \rho(s') + |S|^2 T \Delta_p \\ &\leq (t + 1) |S|^2 [\Delta_v + T \Delta_p] . \end{aligned}$$

Conclusion



- I presented an online reinforcement learning algorithm.
- It does not distinguish a separate learning phase.
- It concentrates exploration on promising states and policies while avoiding to explore mediocre states and policies.
- It comes with a logarithmic regret bound.
- Upper confidence bounds are a general technique to deal with exploration/exploitation trade-offs.
- The decrease of the confidence interval governs the regret bound.