Online Learning in Games

DRAFT

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 7: Sample complexity of Q learning: upper bounds through the lens of episodic MDPs

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-735 (Spring 2024)

License Information for Online Learning in Games Slides

- This work is released under a Creative Commons License with the following terms:
- **Attribution**
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees must give the original authors credit.
- **Non-Commercial**
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees may not use the work for commercial purposes – unless they get the licensor's permission.
- **Share Alike**
  - The licensor permits others to distribute derivative works only under a license identical to the one that governs the licensor's work.
- Full Text of the License

Acknowledgements

These slides were originally prepared by Leello Dadi and Marina Drygala.

Outline

1. A brief introduction to reinforcment learning.
2. The difficult exploration-exploitation dilemma.
3. Borrowing from bandits to solve RL.
4. Proof of sublinear regret:
   - A key lemma.
   - A sequence of summation tricks to control the regret.
5. Beyond the tabular setting: the linear MDP case.

Introduction to Reinforcement Learning

Reinforcement Learning

A control theoretic problem in which the agent tries to maximize its cumulative rewards via interaction with an unknown environment.

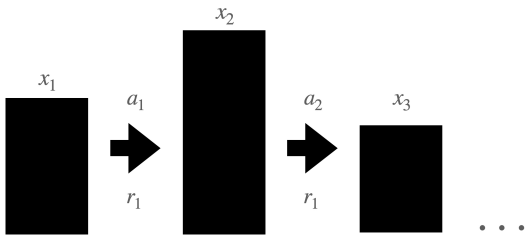
- Process begins when the environment sends the agent a state, then the agent takes an action, receives a reward and transitions to a new state. This process continues until a terminal state is reached.
- The informative rewards the environment provide may be sparse.

Example

Game of Chess: Environment sends initial board state, player can then take an action by moving his pieces, a new state is provided by the adversary, and the terminal state is reached when one player wins.

Reinforcement Learning: General Mathematical Model

- Environment defines a set of actions $\mathcal{A}$, a set of states $\mathcal{S}$, and reward function $r : \mathcal{S} \times \mathcal{A} \to [0, 1]$.
- Environment provides state initial state x_1, and at any given point h in time the agent sees state x_h, takes action a_h, and receives reward $r(x_h, a_h)$ and receives subsequent state x_{h+1}. This process terminates when a terminal state x_{H+1} reached.
- Goal: design a policy π that tells us which action to take in a given state to maximize future reward.
- The value function of a policy π, denoted $V^\pi : \mathcal{S} \to \mathbb{R}$, where $V^\pi(x)$ returns the expected future reward of following policy π beginning from a given state x.



Types of Reinforcement Learning

Model-Based

Form a model of the environment and from there form a control policy based on this learned model.

Model-Free

Directly search for optimal policy, without building an underlying model.

- Typically we assume the underlying Model is a Markov Decision Process, $MDP(\mathcal{S}, \mathcal{A}, \mathbb{P}, r)$
- $\mathbb{P}$ is called the transition matrix, where $\mathbb{P} : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, where $\mathbb{P}(x', x, a)$ is the probability that we transition to state x' given we took action a in state x.
- Assumption: There are finitely many states $\mathcal{S}$ and actions $\mathcal{A}$, where $|\mathcal{S}| = S$ and $|\mathcal{A}| = A$.

Types of Reinforcement Learning Continued

Definition (Model Free (Formal Definition))

A RL algorithm is model-free if its space complexity is sub-linear relative to the space required to store an MDP.

- Need $O(S^2A)$ space to store transition matrix
- In either case we need to collect sample of environment to either model it or estimate optimal policy.
- Algorithms that are sample efficient collect a polynomial number of samples with respect to a fixed accuracy.

Q-Learning

- Q-learning is a trial and error based model-free approach, that aims to find the best action a to take if presented with a state x.
- maintain Q-values, where $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, where $Q(x, a)$ is an estimate of the expected sum of future rewards if we take action a at state x.
- The greedy policy associated with a given set of Q-values would be to select the highest quality action for a given state.

Motivation to Study Model Free Algorithms

- Model-Free algorithms are online and can be more expressive since not restricted by model.
- However, before this work Model-Free algorithms were hypothesized to be less sample efficient than model based.

Question:

Do model-free algorithms need more samples to obtain a good policy?

Formalizing the setting: Episodic MDPs

Tabular Episodic MDP

$MDP(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, H is the number of steps in each episode, $\mathbb{P}$ is the *transition matrix* $\mathbb{P}_h : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and $r_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the reward function.

- There are K episodes and in each episode $k = 1, \ldots, K$ an initial state x_1^k is selected by an adversary. At each $h \in [H]$ the agent observes a state x_h^k and takes $a_h^k \in \mathcal{A}$ receiving reward $r_h(x_h^k, a_h^k)$ then transitions to x_{h+1}^k which is drawn from $\mathbb{P}_h(x_h^k, a_h^k)$. The episode ends when x_{H+1}^k is reached.
- The policy of an agent is a collection of H functions $\{\pi_h : \mathcal{S} \to \mathcal{A}\}_{h \in [H]}$. $V_h^\pi : \mathcal{S} \to \mathbb{R}$ is the value function at step h under policy π and is the expected sum of reqed under policy π beginning from x_h^k until the end of the episode.

Formal Problem Definition Continued

For a policy π

$$V_h^\pi(x) = \mathbb{E}[\sum_{h'=h}^{H} r_{h'}(x_{h'}, \pi_{h'}(x_{h'}))|x_h = x] \tag{1}$$

$$Q_h^\pi(x,a) = r_h(x,a) + [\mathbb{P}_h V_{h+1}](x,a) \tag{2}$$

where $[\mathbb{P}_h V_{h+1}](x,a) = \mathbb{E}_{x' \sim \mathbb{P}_h(x,a)}[V_{h+1}^\pi(x')]$

Formal Problem Definition Continued

Bellman Equations

Since the state and action spaces and time horizon are finite one can show that there exists an optimal policy $\pi^\star$ given by the policy satisfying $V_h^{\pi^\star}(x) = \sup_\pi V_h^\pi(x)$ for all $x \in \mathcal{S}, h \in [h]$. The Bellman Equation gives a dynamic programming formulation.

$$\begin{cases} V_h^\pi(x) = Q_h^\pi(x, \pi_h(x)) \\ Q_h^\pi(x,a) = (r_h + \mathbb{P}_h V_{h+1}^\pi(x,a)) \\ V_{H+1}^\pi \quad \forall x \in \mathcal{S} \end{cases} \qquad \begin{cases} V_h(x) = \max_{a \in \mathcal{A}} Q_h(x,a) \\ Q_h(x,a) = (r_h + \mathbb{P}_h V_{h+1}(x,a)) \\ V_{H+1} \quad \forall x \in \mathcal{S} \end{cases}$$

Goal:

Agent plays K episodes, $k = 1, \ldots, K$ and the adversary picks a starting state x_k^1 for each episode k and the agent chooses a policy π_k before starting k-th episode. We want to minimize the expected regret:

$$Regret(K) = \sum_{k=1}^{K} [V_1^\star(x_1)^k - V_1^{\pi_k}(x_1^k)]. \tag{3}$$

From Regret to sample complexity

- What does it mean to achieve sublinear regret using this notion ?

Theorem

Consider an algorithm achieving sublinear regret

$$\sum_{k=1}^{K}[V_1^\star(x_1^k) - V_1^{\pi_k}(x_1^k)] \leq C \cdot T^{1-\alpha}$$

Then for any $\epsilon > 0$, by applying Markov's inequality, a uniformly chosen policy π_k achieves, with probability at least $2/3$,

$$V_1^\star(x_1^k) - V_1^{\pi_k}(x_1^k) \leq \epsilon$$

with $T = O(1/\epsilon^{\frac{1}{\alpha}})$.

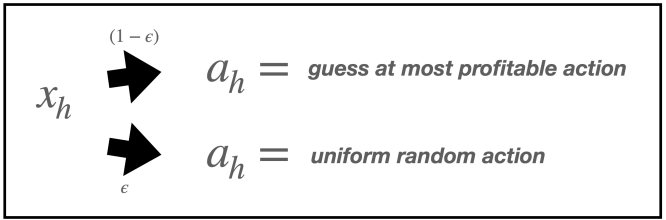
Motivation Continued [2]

	Algorithm	Regret	Time	Space
Model-based	UCRL2 [10] [1]	at least $\tilde{\mathcal{O}}(\sqrt{H^4S^2AT})$	$\Omega(TS^2A)$	$\mathcal{O}(S^2AH)$
	Agrawal and Jia [1] [1]	at least $\tilde{\mathcal{O}}(\sqrt{H^3S^2AT})$		
	UCBVI [5] [2]	$\tilde{\mathcal{O}}(\sqrt{H^2SAT})$	$\tilde{\mathcal{O}}(TS^2A)$	
	vUCQ [12] [2]	$\tilde{\mathcal{O}}(\sqrt{H^2SAT})$		
Model-free	Q-learning (ε-greedy) [14] (if 0 initialized)	$\Omega(\min\{T, A^{H/2}\})$	$\mathcal{O}(T)$	$\mathcal{O}(SAH)$
	Delayed Q-learning [25] [3]	$\tilde{\mathcal{O}}_{S,A,H}(T^{4/5})$		
	Q-learning (UCB-H)	$\tilde{\mathcal{O}}(\sqrt{H^4SAT})$		
	Q-learning (UCB-B)	$\tilde{\mathcal{O}}(\sqrt{H^3SAT})$		
	lower bound	$\Omega(\sqrt{H^2SAT})$	-	-

Table 1: Regret comparisons for RL algorithms on episodic MDP. $T = KH$ is totally number of steps, H is the number of steps per episode, S is the number of states, and A is the number of actions. For clarity, this table is presented for $T \geq \text{poly}(S, A, H)$, omitting low order terms.

Difficulty with Model-Free

- Problem of exploitation versus exploration.
- When exploiting we take what we think is the best action.
- When exploring we try something new to gain more information about the environment.
- Naive-Approach: ϵ-greedy exploration.



ϵ-greedy Pseudocode

Algorithm 1 ϵ-greedy

Initialize: $Q(x, a)$
receive x_1
for $h = 1, \dots H$ **do**
  with probability $(1-\epsilon)$ take action $a_h \leftarrow \arg\max_{a \in \mathcal{A}} Q(x_h, a)$ and with probability ϵ take a random action a_h.
  observe x_{h+1}
  $Q(x_h, a_h) \leftarrow Q(x_h, a_h) + \alpha[r_h(x_h, a_h) + \gamma \max_{a \in \mathcal{A}} Q(x_{h+t}, a) - Q(x_h, a_h)]$
end for

Hard Example for ϵ-greedy

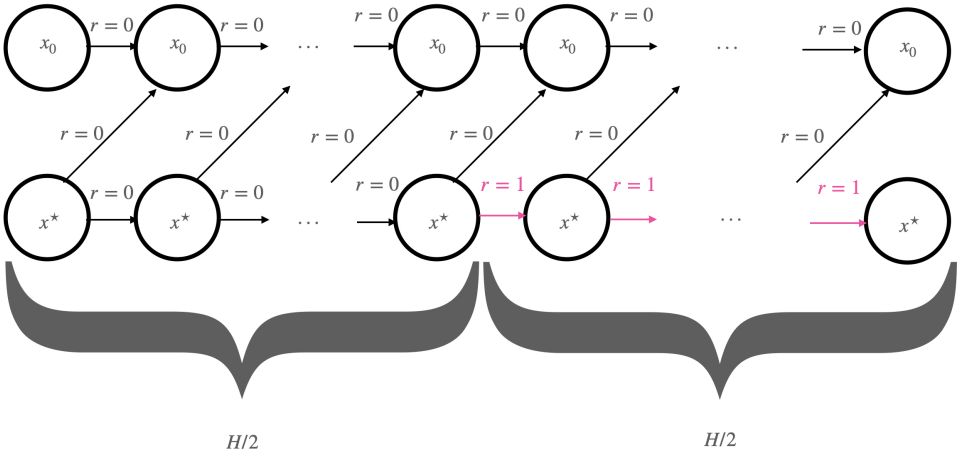


Figure: If initialization of Q-values is 0 expected number of rounds until first update of Q-values is $\Omega(A^{H/2})$, where we suffer $H/2$ regret in each of these rounds. [2]

Help from the bandits literature

- A condensed form of the exploration-exploitation trade-off is given by the Multi-Armed Bandit problem.

Figure: Imagine a row of slot machines - How do you play without knowing which one is the “best"?

Help from the bandits literature: Optimism in the face of uncertainty

- Maintain a high-probability confidence interval around the estimated mean of each arm.

$$\forall a \in A, \quad \mathbb{P}\left[|\hat{\theta}_a(t) - \theta_a| \geq b_a(t)\right] \leq \delta$$

Optimism principle

Operate under the most optimistic outlook: take the upper estimate of the confidence bound.

Remark:

- Instead of taking just the estimated mean, add the uncertainty bonus :

$$\hat{\theta}_a(t) + b_a(t)$$

Optimism in the face of uncertainty

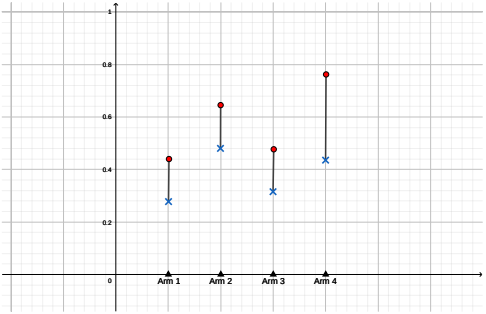


Figure: Upper Confidence Bound

UCB [1]

A possible choice, with $\delta = 1/t$ is to take: $b_a(t) = \sqrt{\frac{\ln(t)}{2N_{\pi,a}(t)}}$. The total regret is then

$$O(\sqrt{\text{numberOfArms} \times T})$$

Q-learning with UCB-Hoeffding

Theorem

There exists an algorithm that achieves a total regret of at most $O(\sqrt{H^4SAT\iota})$ *where* $\iota = \log(SAT/p)$ *with probability* $1-p$ *for any* $p \in (0,1)$.

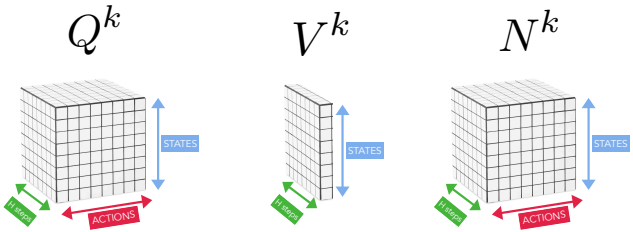
Q-learning with UCB-Hoeffding Pseudocode

Algorithm 2 Q-learning with UCB-Hoeffding

$Q_h(x,a) \leftarrow H$ $N_h(x,a) \leftarrow 0$ for all $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$
for episode $k = 1, \ldots, K$ **do**
  receive x_1
end for
for $h = 1, \ldots H$ **do**
  Take action $a_h = \arg\max_{a \in \mathcal{A}} Q_h(x_h, a)$ and with probability ϵ
  observe $x_{h+1}, t \leftarrow N_h(x_h, a_h) \leftarrow N_h(x_h, a_h) + 1, b_t \leftarrow c\sqrt{H^3\iota/t}$.
  $Q_h(x_h, a_h) \leftarrow (1-\alpha_t)Q(x_h, a_h) + \alpha_t[r_h(x_h, a_h) + V_{h+1}(x_{h+1}) + b_t]$
  $V_h(x_h) \leftarrow \min\{H, \max_{a \in \mathcal{A}} Q_h(x_h, a)\}$
end for

- Q_h^k, V_k^h, N_h^k the functions Q_h, V_h, N_h at the beginning of episode k.
- Let (x_h^k, a_h^k) be the actual state-action pair observed and chosen in step h of episode k.

The algorithm



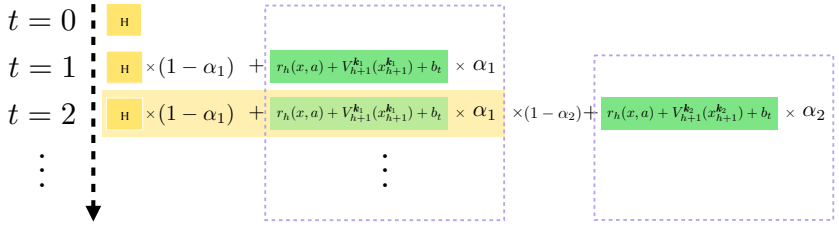
Q^k Update

- I find myself in (x, a) at step h on episode k. I observe the next state x^k_{h+1}.
- I retrieve $t = N^k_h(x, a)$ and add 1.
- I then update Q^k as follows:

$$Q^{k+1}_h(x, a) = (1 - \alpha_t)Q^k_h(x, a) + \alpha_t \left[r_h(x, a) + V^k_{h+1}(x^k_{h+1}) + b_t \right]$$

Unrolling the recursive updates

- Let us consider a fixed (x, a, h).



- We can see that an important coefficient that will appear in the summation is

$$\alpha_t^i := \{\text{First appearance } \alpha_i\} \times \{\text{Later discounts by } (1-\alpha_j) \text{ up to } t\} = \alpha_i \prod_{j=i+1}^{t} (1-\alpha_j)$$

- Rolling out these recursive updates, we have that $Q_h^k(x,a) = \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left[r_h(x,a) + V_{h+1}^{k_i}(x_{h+1}^{k_i}) + b_i \right]$

Remark

Q_h^k is a **weighted sum** of the past observed rewards and value functions. In fact, with the appropriate choice of learning rate α_t, it is a convex combination.

A learning rate to smoothly forget the past

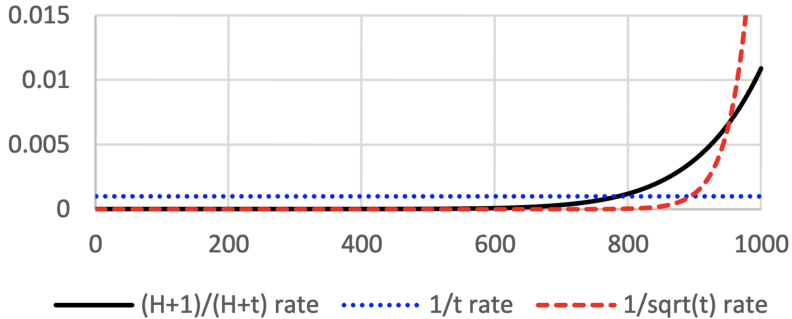


Figure: Visualization of α_t^i for different choices of learning rate

The path to showing sublinear regret

- The first step is to control the error in our estimate of the Q-function.
- Recall that

$$\begin{aligned} Q_h^\star(x,a) &= r_h(x,a) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] \\ &= r_h(x,a) + V_{h+1}^\star(x_{h+1}^k) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] - V_{h+1}^\star(x_{h+1}^k) \end{aligned}$$

$$Q_h^\star(x,a) = \alpha_t^0 Q_h^\star + \sum_{i=1}^{t} \alpha_t^i \left[r_h(x,a) + V_{h+1}^\star(x_{h+1}^{k_i}) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] - V_{h+1}^\star(x_{h+1}^{k_i}) \right]$$

- Comparing to

$$Q_h^k(x,a) = \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left[r_h(x,a) + V_{h+1}^{k_i}(x_{h+1}^{k_i}) + \quad + b_i \right]$$

Lemma (Gap between Q^k and $Q^\star$)

$$Q_h^k - Q_h^\star = \alpha_t^0 (H - Q_h^\star) + \sum_{i=1}^{t} \alpha_t^i \left((V_{h+1}^{k_i}(x_{h+1}^{k_i}) - V_{h+1}^\star(x_{h+1}^{k_i})) + (V_{h+1}^\star(x_{h+1}^{k_i}) - \mathbb{E}_{x_{next}}[V_{h+1}^\star(x_{next})]) + b_t \right)$$

The path to showing sublinear regret

- The first step is to control the error in our estimate of the Q-function.
- Recall that

$$\begin{aligned} Q_h^\star(x,a) &= r_h(x,a) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] \\ &= r_h(x,a) + V_{h+1}^\star(x_{h+1}^k) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] - V_{h+1}^\star(x_{h+1}^k) \end{aligned}$$

$$Q_h^\star(x,a) = \alpha_t^0 Q_h^\star + \sum_{i=1}^{t} \alpha_t^i \left[r_h(x,a) + V_{h+1}^\star(x_{h+1}^{k_i}) + \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})] - V_{h+1}^\star(x_{h+1}^{k_i}) \right]$$

- Comparing to

$$Q_h^k(x,a) = \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left[r_h(x,a) + V_{h+1}^{k_i}(x_{h+1}^{k_i}) + \quad + b_i \right]$$

Lemma (Gap between Q^k and $Q^\star$)

$$(Q_h^k - Q_h^\star) = \textit{Not visiting } (x,a) + \textit{Not knowing } V_{h+1}^\star + \textit{Not knowing the transition dynamics} + \textit{Bonus terms}$$

Controlling the gap

○ The second error is of a classic nature: what price do you pay when using samples to approximate an expectation ? How large can the following sum be ?

$$\sum_{i=1}^{t} \alpha_t^i (V_{h+1}^\star(x_{h+1}^{k_i}) - \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})])$$

Concentration of measure

Independent random variables cannot collaborate to deviate significantly from their expectation.

Remarks:

○ Obstacles to an immediate application of Hoeffding:

- The sequence of next states $x_{h+1}^{k_i}$ are not independent.
- The number of terms t is also random.

Azuma-Hoeffding and union bounds

- Azuma: "The sum of sub-gaussian martingale increments is sub-gaussian".
- Hoeffding: "Bounded random variables are sub-gaussian".
- How do we deal with the random number of terms ?

Union bounds

We set the failure probability to be $p/(SAHK)$, so that we have with probability $1-p$, **for all** (x, a, h, τ),

$$\left| \sum_{i=1}^{\tau} \alpha_\tau^i (V_{h+1}^\star(x_{h+1}^k) - \mathbb{E}_{x_{\text{next}}}[V_{h+1}^\star(x_{\text{next}})]) \right| \leq c \sqrt{\frac{H^3 \log(SAHK/p)}{\tau}}.$$

The key here is that this bound holds uniformly for all possible values of t.

Remark:

- From this we can easily derive the following upper bound:

$$(Q_h^k - Q_h^\star)(x,a) \leq \text{Not visiting } (x,a) + \text{Not knowing } V_{h+1}^\star + c \sqrt{\frac{H^3 \log(SAHK/p)}{t}}$$

Establishing a lower bound

- We begin with a crucial observation:

$(Q_h^k - Q_h^\star)(x,a) =$ Not visiting (x,a) + Not knowing $V_{h+1}^\star$ + Not knowing the transition dynamics + Bonus terms

The error on Q functions **at step** h is dictated by the error at **the next step** $h+1$.

Induction from $H, H-1, \cdots, 1$ as a central tool to prove results

The problem is at *its easiest* at step H. Indeed there is only a single action to take, there is no planning involved as it is the last round. So the error at step H is easy to control.

$$(Q_H^k - Q_H^\star)(x,a) = \text{Not visiting } (x,a) + \text{Bonus terms}$$

Consequently $Q_H^k - Q_H^\star \geq 0$. Which implies that $V_H^k - V_H^\star \geq 0$.
By taking the bonus terms large enough to compensate for the possibly negative sample-expectation errors, we can have that

$$Q_{H-1}^k - Q_{H-1}^\star \geq 0.$$

We proceed like so, by induction, to show that $Q_h^k - Q_h^\star \geq 0$ for all h.

The central lemma

Lemma (Lemma 4.3)

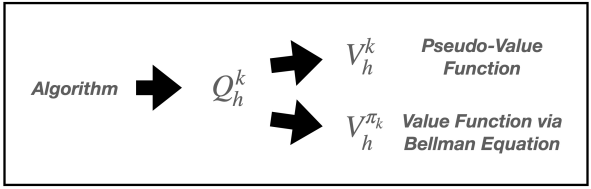
There exists an absolute constant $c > 0$ such that for any $p \in (0,1)$, and $b_t = c\sqrt{H^3 \log(SAT/p)/t}$, we have with probability at least $1-p$, simultanuously for all (x,a,h),

$$0 \leq Q_h^k - Q_h^\star \leq \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i (V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i}) + \beta_t \tag{4}$$

where $t = N_h^k(x,a)$ and $\beta_t := 2\sum_{i=1}^t \alpha_t^i b_t$.

Notation

- π_k is the policy specified by the Bellman Equation for Q-value Q_h^k.
- Let $k_i(x_h^k, a_h^k)$ be the episode in which (x_h^k, a_h^k) was taken at step h for the i^{th} time.



Proof of Theorem 4

- Let $\delta_h^k = (V_h^k - V_h^{\pi_k})(x_h^k)$ and $\phi_h^k = (V_h^k - V_h^\star)(x_h^k)$.
- We know by Lemma (4) that with probability $1-p$ that $Q_h^k \geq Q_h^\star$ and since $V_h^k = \min\{H, \max_{a\in\mathcal{A}} Q_h^k(x_h, a)\}$ and by the Bellman Equation $V_h^\star = \max_{a\in\mathcal{A}} Q_h^\star(x, a)$ we obtain that $V_h^k \geq V_h^\star$.

Regret Bound

We obtain an upper bound for our regret,

$$\text{Regret}(K) = \sum_{k=1}^{K} (V_1^\star - V_1^{\pi_k})(x_1^k) \leq \sum_{k=1}^{K} (V_1^k - V_1^{\pi_k})(x_1^k) = \sum_{k=1}^{K} \delta_1^k \tag{5}$$

Proof of Theorem 4 Continued

- Main idea is to bound $\sum_{k=1}^{K} \delta_h^k$ by the values for the next step, namely $\sum_{k=1}^{K} \delta_{h+1}^k$ to get a recursive formula and use Equation 5 to bound the regret. Let $n_h^k = N_h^k(x_h^k, a_h^k)$
-

$$
\begin{aligned}
\delta_h^k &= (V_h^k - V_h^{\pi_k})(x_h^k) \\
&\leq (Q_h^k - Q_h^{\pi_k})(x_h^k, a_h^k) \\
&= (Q_h^k - Q_h^\star)(x_h^k, a_h^k) + (Q_h^\star - Q_h^{\pi_k})(x_h^k, a_h^k) \\
&\leq \alpha_{n_h^k}^0 H + \sum_{i=1}^{t} \alpha_{n_h^k}^i (V_{h+1}^{k_i} - V_{h+1}^\star(x_{h+1}^{k_i})) + \beta_t + [\mathbb{P}_h(V_{h+1}^\star - V_{h+1}^{\pi_k})](x_h^k, a_h^k) \\
&= \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \phi_{h+1}^{k_i} + \beta_{n_h^k} - \phi_{h+1}^k + \delta_{h+1}^k + \xi_{h+1}^k
\end{aligned}
\tag{6}
$$

- Where the first inequality follows because by the Bellman Equation and algorithm's choice of V_h, Q_h. The first part of the second inequality comes from applying Equation (4) and the second by the Bellman Equation. The last inequality follows if we set $\xi_{h+1}^k = [(\mathbb{P}_h - \hat{\mathbb{P}}_h)(V_{h+1}^\star - V_{h+1}^{\pi_k})](x_h^k, a_h^k)$.

Proof of Theorem 4 Continued

- Now we use Equation (6) to compute $\sum_{k=1}^{K} \delta_h^k$. Clearly an upper bound for

$$\sum_{k=1}^{K} \alpha^0_{n_h^k} H$$

is H times the number of state action pairs. This follows because $\alpha_t^0 = \begin{cases} 1 & t = 0 \\ 0 & \text{otherwise} \end{cases}$.

Proof of Theorem 4 Continued

Then the second term in Equation (6)

$$\sum_{k=1}^{K}\sum_{i=1}^{n_h^k}\alpha^i_{n_h^k}\phi_{h+1}^{k_i(x_h^k,a_h^k)} \le \sum_{k'=1}^{K}\phi_{h+1}^{k'}\sum_{t=n_h^{k'}+1}^{\infty}\alpha_t^{n_h^{k'}} \le (1+1/H)\sum_{k=1}^{K}\phi_{h+1}^{k} \tag{7}$$

Where the first inequality follows because the term $\phi_{h+1}^{k'}$ only appears in terms for $k > k'$ and when $(x_h^k, a_h^k) = (x_h^{k'}, a_h^{k'})$, and the final inequality because $\sum_{t=i}^{\infty}\alpha_t^i = (1+1/H)$ for all $i \ge 1$.

	$(x_h^{k'}, a_h^{k'})$	$(x_h^{k'}, a_h^{k'})$	$\cdots$	$(x_h^{k'}, a_h^{k'})$
Appearance	1	2	$\cdots$	i
Index $(k'+t)$	k'	$k'+1$		$k'+t$
Contribution to Sum		$\phi_{h+1}^{k'}\alpha_1^{n_h^{k'}}$	$\cdots$	$\phi_{h+1}^{k'}\alpha_t^{n_h^{k'}}$

Proof of Theorem 4 Continued

Then,

$$\begin{aligned}\sum_{k=1}^{K}\delta_h^k &\leq SAH + (1+1/H)\sum_{k=1}^{K}\phi_{h+1}^k - \sum_{k=1}^{K}\phi_{h+1}^k + \sum_{k=1}^{K}\delta_{h+1}^k + \sum_{k=1}^{K}(\beta_{n_h^k} + \xi_{n_h^k}) \\ &\leq SAH + (1+1/H)\sum_{k=1}^{K}\delta_{h+1}^k + \sum_{k=1}^{K}(\beta_{n_h^k} + \xi_{n_h^k})\end{aligned} \tag{8}$$

where here we use the fact that since $V^\star \geq V^{\pi_k}$ we have $\delta_{h+1}^k \geq \phi_{h+1}^k$.

Proof of Theorem 4 Continued

$$\sum_{k=1}^{K} \delta_H^k \leq SAH + (1+1/H) \sum_{k=1}^{K} \delta_{H+1}^k + \sum_{k=1}^{K} (\beta_{n_H^k} + \xi_{n_H^k})$$

$$\dots$$

$$\sum_{k=1}^{K} \delta_2^k \leq SAH + (1+1/H) \sum_{k=1}^{K} \delta_3^k + \sum_{k=1}^{K} (\beta_{n_2^k} + \xi_{n_2^k})$$

$$\sum_{k=1}^{K} \delta_1^k \leq SAH + (1+1/H) \sum_{k=1}^{K} \delta_2^k + \sum_{k=1}^{K} (\beta_{n_1^k} + \xi_{n_1^k})$$

we obtain that

$$\sum_{k=1}^{K} \delta_1^k \leq O\left(SAH^2 + \sum_{h=1}^{H} \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k) \right)$$

since $\delta_{H+1}^K = 0$ and $\sum_{h=1}^{H} (1+1/h)^h = O(H)$.

Proof of Theorem 4 Continued

- Then by Lemma (4) $\beta_t \leq 4c\sqrt{H^3\iota/t}$ and thus

$$\sum_{k=1}^{K} \beta_{n_h^k} \leq O\left(\sum_{k=1}^{K} \sqrt{H^3\iota/n_h^k}\right) = O\left(\sum_{(x,a)} \sum_{n=1}^{N_h^K(x,a)} \sqrt{H^3\iota/n}\right) \leq O\left(\sum_{(x,a)} \sum_{n=1}^{K/SA} \sqrt{H^3\iota/n}\right) \tag{9}$$

  Where the first equality follows because we need to sum over all pairs (x, a) that appeared as (x_h^k, a_h^k) for some k. Then since $\sum_{(x,a)} N_h^K(x, a) = K$ this quantity is maximized when each state action pair appears about K/SA times.
- Continuing we obtain,

$$\sum_{k=1}^{K} \beta_{n_h^k} \leq O(\sqrt{H^3SAK\iota}) \leq O(\sqrt{H^2SAT\iota}) \tag{10}$$

  Since $\sum_{i=1}^{n} \frac{1}{\sqrt{i}} = O(\sqrt{n})$ we obtain the the first inequality. Then last inequality comes from the fact that $T = KH$.

Proof of Theorem 4 Continued

Azuma Hoeffding again

We can then apply Azuma Hoeffding to get that with probability $1-p$

$$|\sum_{h=1}^{H}\sum_{k=1}^{K}\xi_{h+1}^{k}| = |\sum_{h=1}^{H}\sum_{k=1}^{K}(\mathbb{P}_h - \hat{\mathbb{P}}_h^k)| \leq cH\sqrt{T\iota}$$

via the same argument from Lemma (4).

- We conclude that $\mathrm{Regret}(K) \leq O(H^2SA + \sqrt{H^4SAT\iota})$, observe that when T is large ($\geq \sqrt{H^4SAT\iota}$) then the second term dominates, and if T is small ($\leq \sqrt{H^4SAT\iota}$) then $\sum_{k=1}^{K}\delta_1^k \leq HK$ and therefore we are bounded by the second term as well.
  In summary the equation holds with probability $1-2p$ and thus if we scale the choice of p appropriately we obtain the desired result.

Discussion of the result

Hoeffding bonus:

$$\text{Regret} \leq O(\sqrt{H^4 SAT\iota})$$

Discussion of the result

Bernstein bonus:

$$\text{Regret} \leq O(\sqrt{H^2SAT\iota})$$

MAB vs trajectory planning

The cost of trajectory planning is a $\sqrt{H}$ factor.

A meaningless bound for large state spaces

Unsatisfactory dependence on the number of states

How can we remove the dependence on S ?

Remark:

- Often the number of states is exponentially large.
- Is it possible to have compressed representations of the Q-action-value function ?

Linear function approximation

- Let us introduce a restricted problem class.

Linear MDPs

$\mathrm{MDP}(\mathrm{S}, \mathrm{A}, \mathrm{H}, \mathbb{P}, \mathrm{r})$ is a *linear MDP* with a feature map $\phi : S \times A \to \mathbb{R}^d$, if for any $h \in [H]$, there exist d *unknown* (signed) measures $\mu_h = (\mu_h^{(1)}, \ldots, \mu_h^{(d)})$ over S and an *unknown* vector $\theta_h \in \mathbb{R}^d$, such that for any $(x, a) \in S \times A$, we have

$$\mathbb{P}_h(\cdot|x, a) = \langle \phi(x, a), \mu_h(\cdot) \rangle, \qquad r_h(x, a) = \langle \phi(x, a), \theta_h \rangle. \tag{11}$$

Without loss of generality, we assume $\|\phi(x, a)\| \leq 1$ for all $(x, a) \in S \times A$, and $\max\{\|\mu_h(S)\|, \|\theta_h\|\} \leq \sqrt{d}$ for all $h \in [H]$.

Remark:

- In this setting the action value function is also linear:

$$Q_h^\pi(x, a) = \langle w_h^\pi, \phi(x, a) \rangle,$$

for all $h \in [H]$ and for all policies π.

Least-Squares Value Iteration - with UCB

Compile error

A sketch of the proof

Regret Bound [3]

The total regret can be upper bounded by $O(\sqrt{dH^3T})$

- First step: Establish concentration result for

$$\phi(x,a)^\top \Lambda_h^{-1} \sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau)[V_{h+1}(x_{h+1}^\tau) - \mathbb{P}_h V_{h+1}(x_h^\tau, a_h^\tau)]$$

  requires the introduction of a restriction on the possible V functions.

- Second step: Use recursion from H to 1 to propagate error back down.

Conclusion

- Sublinear regret is achievable in “model-free" reinforcement learning.
- Can we devise algorithms that do not know a priori the number of episodes that will be played ?

References I

[1] Peter Auer.
Using confidence bounds for exploitation-exploration trade-offs.
Journal of Machine Learning Research, 3(Nov):397–422, 2002.
(Cited on page 21.)

[2] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan.
Is q-learning provably efficient?
Advances in neural information processing systems, 31, 2018.
(Cited on pages 15 and 18.)

[3] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan.
Provably efficient reinforcement learning with linear function approximation.
In *Conference on Learning Theory*, pages 2137–2143. PMLR, 2020.
(Cited on page 47.)