Provably Efficient Exploration in Policy Optimization

Presentor: Tian Xu
xut@lamda.nju.edu.cn

Nanjing University, Nanjing, China

Mainly based on the following papers:
Cai, Qi, et al. "Provably efficient exploration in policy optimization." ICML, 2020.
Shani, Lior, et al. "Optimistic policy optimization with bandit feedback." ICML, 2020.

August 3, 2021

Outline

Background

- With exact policy gradient, [Agarwal et al., 2020, Bhandari and Russo, 2021] proved that policy gradient methods can converge to global optimal solution.
- To perform the exact policy gradient $\left(\frac{\partial V(\pi)}{\partial \pi_h(a|s)} = P_h^{\pi}(s) Q_h^{\pi}(s,a)\right)$, we require the access to the reward function and transition probability.
- In practice, we often do not have full knowledge of the MDP and need to collect dataset in an online manner.

Whether policy optimization methods can converge to the optimal policy under the online setting?

Main Contributions

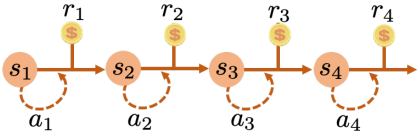
- In [Cai et al., 2020, Shani et al., 2020], the authors propose the first provably efficient policy-based method under the online setting.
- They develop a novel regret decomposition (Lemma 3.2) upon which the algorithm and regret analysis are built.
- From this regret decomposition, they construct upper confidence bound (UCB) in policy evaluation and perform mirror ascent in policy improvement, which are two main ingredients of the algorithm.

Comparison of Different Algorithms

Algorithms	Regret	Algorithm Type	Setting
POMD [Shani et al., 2020]	$\tilde{\mathcal{O}}\left(\sqrt{S^2AH^4K}\right)$	Policy-based	Tabular MDP
UCB-VI [Azar et al., 2017]	$\tilde{\mathcal{O}}\left(\sqrt{SAH^3K}\right)$	Value-based	Tabular MDP
OPPO [Cai et al., 2020]	$\tilde{\mathcal{O}}\left(\sqrt{d^2H^4K}\right)$	Policy-based	Linear Mixture MDP
UCRL-VTR [Ayoub et al., 2020]	$\tilde{\mathcal{O}}\left(\sqrt{d^2H^4K}\right)$	Value-based	Linear Mixture MDP

Table: Comparison of regret bounds for different algorithms. Under the linear mixture MDP, the transition probability is linear w.r.t the known feature and d is the feature dimension. Compared with UCB-VI, the regret of POMD is sub-optimal. The regret of POMD and UCB-VI is dominated by the size of bonus. POMD builds UCB for the policy in each iteration, rather than only the optimal policy like in UCB-VI. Hence, POMD requires a larger bonus than that in UCB-VI (see page 29 for details).

Markov Decision Process



Markov Decision Process

- Consider a finite episodic Markov Decision Process $\left(\mathcal{S}, \mathcal{A}, H, \{P_h\}_{h\in[H]}, \{r_h\}_{h\in[H]}\right)$.
  - $\mathcal{S}$ and $\mathcal{A}$ are the finite state and action space, respectively.
  - $r_h(s, a) \in [0, 1]$ is reward received after taking the action a in state s at step h.
  - $P_h(s'|s, a)$ specifies the transition probability of s' conditioned on s and a at step h.
  - H is the horizon length.
  - The initial state s_1 is fixed.

Markov Decision Process

- A policy π is a collection of functions $\pi_h : \mathcal{S} \to \Delta(\mathcal{A})$ for all $h \in [H]$ and $\pi_h(a|s)$ gives the probability of taking action a on state s at step h.
- For a policy π, its value function V^π and Q-value function Q^π are defined as

$$V_h^\pi(s) \triangleq \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'}\left(s_{h'}, a_{h'}\right) \mid s_h = s, \pi\right]$$

$$Q_h^\pi(s, a) \triangleq \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'}\left(s_{h'}, a_{h'}\right) \mid s_h = s, a_h = a, \pi\right]$$

- The value of policy π: $V(\pi) = V_1^\pi(s_1)$.
- For an algorithm, we use the regret defined as $\sum_{k=1}^{K} V(\pi^*) - V(\pi^k)$ to measure its performance, where π^k is the policy obtained by the algorithm in the iteration (or episode) k.

Bellman Equation

- For a policy π, its value function V^π and Q-value function Q^π hold the following Bellman Equations [Puterman, 2014]:

$$V_h^\pi(s) = \mathbb{E}_{a\sim\pi_h(\cdot|s)}\left[Q_h^\pi(s,a)\right] = \langle \pi_h(\cdot|s), Q_h^\pi(\cdot|s)\rangle$$

$$Q_h^\pi(s,a) = r_h(s,a) + \mathbb{E}_{s'\sim P_h(\cdot|s,a)}\left[V_{h+1}^\pi(s')\right] = r_h(s,a) + P_h V_{h+1}^\pi(s,a).$$

- Given the <u>transition probability</u> and <u>reward</u> of the MDP, for any policy π, we can calculate its value function and Q-value function via dynamic programming.

Outline

Outline

Policy Evaluation

- Policy-based methods alternate between **policy evaluation** and **policy improvement**.
- In policy evaluation, for a policy π, we aim to calculate its value function V^π and especially Q-value function Q^π.
- Policy evaluation is a key step in policy-based methods, e.g., policy iteration and policy gradient method.

Policy Evaluation via Dynamic Programming

Algorithm 1 Policy Evaluation (PE)

1: **Input:** Policy π, reward function r, transition probability P, $V_{H+1}(s) = 0, \forall s \in \mathcal{S}$.
2: **for** $h = H, H-1, \cdots, 1$ **do**
3:     **for** $(s, a) \in \mathcal{S} \times \mathcal{A}$ **do**
4:         $Q_h(s, a) = r_h(s, a) + P_h V_{h+1}(s, a)$
5:     **end for**
6:     **for** $s \in \mathcal{S}$ **do**
7:         $V_h(s) = \langle \pi_h(\cdot|s), Q_h(s, \cdot) \rangle$
8:     **end for**
9: **end for**
10: **Output:** The Q-value function of π: Q.

Outline

Mirror Ascent Policy Optimization

- Given the policy π^{old}, we consider the linear approximation of $V(\pi)$:
$$V(\pi) \approx V\left(\pi^{\text{old}}\right) + \langle \pi - \pi^{\text{old}}, \nabla V\left(\pi^{\text{old}}\right)\rangle,$$
where $\nabla V\left(\pi^{\text{old}}\right), \pi \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|H}$.
- Recall that $\frac{\partial V(\pi)}{\partial \pi_h(a|s)} = P_h^{\pi}(s) Q_h^{\pi}(s,a)$ and $P_h^{\pi}(s) = \Pr(s_h = s|\pi)$, we have
$$V(\pi) \approx V\left(\pi^{\text{old}}\right) + \sum_{h=1}^{H} \mathbb{E}_{s \sim P_h^{\pi^{\text{old}}}(\cdot)}\left[\left\langle \pi_h(\cdot|s) - \pi_h^{\text{old}}(\cdot|s), Q_h^{\pi^{\text{old}}}(s,\cdot)\right\rangle\right]$$

Mirror Ascent Policy Optimization

- To guarantee that π is close to π^{old}, mirror ascent policy optimization (MAPO) finds a policy which maximizes the linear approximation (Q-value function) with a regularizer:

$$\pi_h^{\text{new}}(\cdot|s) = \underset{\pi\in\Delta(\mathcal{A})}{\operatorname{argmax}} \left\langle \pi(\cdot|s), Q_h^{\pi^{\text{old}}}(s,\cdot) \right\rangle - \frac{1}{\eta} D_{\text{KL}}\left(\pi(\cdot|s), \pi_h^{\text{old}}(\cdot|s)\right), \ \forall (s,h) \in \mathcal{S} \times [H],$$

  where η is the stepsize.

- The closed form solution of the above problem is

$$\pi_h^{\text{new}}(a \mid s) = \frac{\pi_h^{\text{old}}(a \mid s) \exp\left(\eta Q_h^{\pi^{\text{old}}}(s,a)\right)}{\sum_{a'} \pi_h^{\text{old}}(a' \mid s) \exp\left(\eta Q_h^{\pi^{\text{old}}}(s,a')\right)}, \ \forall (s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H].$$

Mirror Ascent Policy Optimization

- Due to the constraint to the old policy, MAPO is "**conservatively**" greedy w.r.t the Q-value function, which can be regarded as "**soft**" policy iteration.
- When the stepsize $\eta \to \infty$, we exactly recover policy iteration algorithm.

Mirror Ascent Policy Optimization

Algorithm 2 Mirror Ascent Policy Optimization

1: **Input:** Uniformly initialized policy π^1, reward function r, transition probability P, stepsize η
2: **for** $k = 1, 2, \cdots, K$ **do**
3: Evaluate policy π^k via dp: $Q^{\pi^k} = \mathsf{PE}\left(\pi^k, r, P\right)$
4: Perform mirror ascent update:
5: **for** $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$ **do**
6: $\pi_h^{k+1}(a \mid s) = \frac{\pi_h^k(a|s)\exp\left(\eta Q_h^{\pi^k}(s,a)\right)}{\sum_{a'} \pi_h^k(a'|s)\exp\left(\eta Q_h^{\pi^k}(s,a')\right)}$
7: **end for**
8: **end for**

Regret of Mirror Ascent Policy Optimization

Theorem 2.1: Regret of MAPO

Consider the mirror ascent policy optimization with stepsize $\eta = \sqrt{\frac{2\ln(|\mathcal{A}|)}{H^2 K}}$, we have

$$\sum_{k=1}^{K} V_1^*(s_1) - V_1^{\pi^k}(s_1) \le \sqrt{2\log(|\mathcal{A}|)H^4 K}.$$

Mirror Ascent Policy Optimization V.S. Policy Iteration

- Under the stochastic MDP setting, policy iteration (PI) achieves the optimal policy when $K \geq H$ and thus, its regret is upper bounded by H^2, which is smaller than MAPO.
- Under the adversarial MDP setting where **the reward function changes across different iterations** k, the regret of MAPO is also $\mathcal{O}\left(\sqrt{\log(|\mathcal{A}|)H^4K}\right)$ [Cai et al., 2020] due to the robustness of mirror ascent. However, PI dose not work under the adversarial setting.
- This property is useful in the stochastic MDP and online setting where we can only use the estimated Q-value function, rather than the true Q-value function, to perform mirror ascent update.

Analysis: Policy Difference Lemma

Lemma 2.1: Policy Difference Lemma

For any policy π and π', we have

$$V_1^{\pi}(s_1) - V_1^{\pi'}(s_1) = \mathbb{E}\left[\sum_{h=1}^{H} \underbrace{\left\langle \pi_h(\cdot|s_h) - \pi'_h(\cdot|s_h), Q_h^{\pi'}(s_h,\cdot)\right\rangle}_{\text{Term (I)}} \middle| \pi \right].$$

- Although $V_1^{\pi}(s_1)$ is not concave w.r.t π, Term (I) is linear w.r.t $\pi_h(\cdot|s_h)$ and $\pi'_h(\cdot|s_h)$ when we regard $Q^{\pi'}(s_h,\cdot)$ as an arbitrary vector.

Analysis: Policy Difference Lemma

When $\pi = \pi^*$ and $\pi' = \pi^k$, policy difference lemma tells us

$$\sum_{k=1}^{K} V_1^{\pi^*}(s_1) - V_1^{\pi^k}(s_1) = \sum_{k=1}^{K} \mathbb{E}\left[\sum_{h=1}^{H} \left\langle \pi_h^*(\cdot|s_h) - \pi_h^k(\cdot|s_h), Q_h^{\pi^k}(s_h,\cdot) \right\rangle \middle| \pi^* \right].$$

- This motivates to view it as an online linear maximization problem. In each iteration k, the MA learner plays π^k and observes a linear objective $\langle \pi(\cdot|s), Q_h^{\pi^k}(s,\cdot)\rangle$ $\forall (s,h) \in \mathcal{S} \times [H]$ and updates its decision via

$$\pi_h^{k+1}(\cdot|s) = \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\langle \pi(\cdot|s), Q_h^{\pi^k}(s,\cdot) \right\rangle - \frac{1}{\eta} D_{\mathrm{KL}}\left(\pi(\cdot|s), \pi_h^k(\cdot|s)\right), \ \forall (s,h) \in \mathcal{S} \times [H].$$

- With this connection, we can leverage the known regret of mirror ascent on online linear maximization problem. This also explains why MAPO can handle with the adversarial MDP setting.

Analysis: Regret of MA on Online Linear Optimization

Algorithm 3 Mirror Ascent in Online Linear Maximization

Input: Uniformly initialized decision $x^1 = \left[\frac{1}{d}, \cdots, \frac{1}{d}\right]$, stepsize η
for $k = 1, 2, \cdots, K$ **do**
    Take decision x_k and observe objective function $l^k(x) = \langle g^k, x \rangle$
    Perform mirror ascent update: $x_{k+1} = \operatorname{argmax}_{x \in \Delta(d)} \langle g^k, x \rangle - \frac{1}{\eta} D_{\mathrm{KL}}(x, x_k)$
end for

Theorem 2.2: [Shalev-Shwartz, 2012]

Consider Algorithm 3, for any $u \in \Delta(d)$,

$$\sum_{k=1}^{K} \langle g^k, u - x^k \rangle \leq \frac{\log(d)}{\eta} + \frac{\eta}{2} \sum_{k=1}^{K} \sum_{i=1}^{d} x_i^k {g_i^k}^2$$

Analysis: Regret of MAPO

- Translate the above regret into the MDP problem: $\forall (s,h) \in \mathcal{S} \times [H]$,

$$\sum_{k=1}^{K} \left\langle Q_h^{\pi^k}(\cdot \mid s), \pi_h^*(\cdot \mid s) - \pi_h^k(\cdot \mid s) \right\rangle \leq \frac{\log(|\mathcal{A}|)}{\eta} + \frac{\eta}{2} \sum_{k=1}^{K} \sum_{a \in \mathcal{A}} \pi_h^k(a \mid s) \left(Q_h^{\pi^k}(s,a) \right)^2$$
$$\leq \frac{\ln(|\mathcal{A}|)}{\eta} + \frac{\eta H^2 K}{2} = \sqrt{2 \ln(|\mathcal{A}|) H^2 K},$$

  where the last step follows that $\eta = \sqrt{\frac{2\ln(|\mathcal{A}|)}{H^2 K}}$.

- $\sum_{k=1}^{K} V_1^*(s_1) - V_1^{\pi^k}(s_1) = \sum_{k=1}^{K} \mathbb{E}\left[\sum_{h=1}^{H} \left\langle \pi_h^*(\cdot|s_h) - \pi_h^k(\cdot|s_h), Q_h^{\pi^k}(s_h, \cdot) \right\rangle \middle| \pi \right] \leq$
  $\sqrt{2 \ln(|\mathcal{A}|) H^4 K}$.

Conclusion for Solving MDP

We consider the setting where the transition probability and reward are known.

- Policy-based methods follow the framework of **policy evaluation** and **policy improvement**.
- Based on this framework, we introduce **mirror ascent policy optimization** (MAPO) for solving MDP.
- We prove the regret bound of MAPO from the connection to online linear optimization.

Outline

From Known MDP Setting to Online Learning Setting

- Under the known MDP setting, for a policy, we can obtain the exact Q-value function with the knowledge of transition probability and reward function.
- In practice, we often operate with the online learning setting: the agent collects samples to estimate its Q-value function.
- There exists a trade-off between exploration and exploitation, i.e., the agent should explore poorly-understood states and actions to gain information and improve future performance, or exploit well-understood states and actions to optimize short-run rewards.

Exploration V.S. Exploitation: Optimism in the Face of Uncertainty

- When we lack knowledge (uncertainty) in which action is optimal, we will construct an optimistic estimate (upper bound of the true value) and pick the action with the highest optimistic estimate.
  - If the choice is wrong, the optimistic estimate decreases and the certainty increases.
  - If the choice is right, the agent gets high reward and the certainty increases.

Exploration V.S. Exploitation: Optimism in the Face of Uncertainty

What is the knowledge of MDP?

- Reward function and transition probability.

How to estimate them from the collected data?

- Maximum likelihood estimator (i.e., counting): $\forall (s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$,

$$\widehat{r}_h(s,a) = \left(\sum_{i=1}^{k} r_h^i(s_h^i, a_h^i) \mathbb{I}\left(s_h^i = s, a_h^i = a\right) \right) / N_h^k(s,a)$$

$$\widehat{P}_h(s'|s,a) = \left(\sum_{i=1}^{k} \mathbb{I}\left(s_h^i = s, a_h^i = a, s_{h+1}^i = s'\right) \right) / N_h^k(s,a)$$

where $N_h^k(s,a) = \sum_{i=1}^{k} \mathbb{I}\left(s_h^i = s, a_h^i = a\right)$ and $(s_h^i, a_h^i, r_h^i(s_h^i, a_h^i), s_{h+1}^i)$ is the pair observed at episode i and timestep h.

Exploration V.S. Exploitation: Optimism in the Face of Uncertainty

How to measure the uncertainty?

- Thanks to the concentration inequality [Weissman et al., 2003, Wainwright, 2019],
  $|\widehat{r}_h(s,a) - r_h(s,a)| \precsim \widetilde{\mathcal{O}}\left(\sqrt{\frac{1}{N_h^k(s,a)}}\right)$ and $\left\|P_h(\cdot|s,a) - \widehat{P}_h(\cdot|s,a)\right\|_1 \precsim \widetilde{\mathcal{O}}\left(\sqrt{\frac{|\mathcal{S}|}{N_h^k(s,a)}}\right)$.

How to construct the optimistic estimate based on the uncertainty measure?

- Under the MDP problem, the Q-value function influences which action to take.
- Add reward bonus when calculating the Q-value function,
  $Q = \mathsf{PE}\left(\pi, \widehat{r} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{1}{N_h^k(s,a)}}\right) + \widetilde{\mathcal{O}}\left(H\sqrt{\frac{|\mathcal{S}|}{N_h^k(s,a)}}\right), \widehat{P}\right)$.
- The reward bonus in SOTA value-based method (i.e., UCB-VI) is designed as
  $\widetilde{\mathcal{O}}\left(H\sqrt{\frac{1}{N_h^k(s,a)}}\right)$, which is much smaller.

Exploration V.S. Exploitation: Optimism in the Face of Uncertainty

Algorithm 4 Optimistic Policy Evaluation

Input: Policy π, reward function $\widehat{r}$, transition probability $\widehat{P}$, $V_{H+1}(s) = 0, \forall s \in \mathcal{S}$.
for $h = H, H-1, \cdots, 1$ **do**
  for $(s,a) \in \mathcal{S} \times \mathcal{A}$ **do**
    $Q_h(s,a) = \widehat{r}_h(s,a) + \widetilde{\mathcal{O}}\left(\sqrt{\frac{1}{N_h^k(s,a)}}\right) + \widetilde{\mathcal{O}}\left(H\sqrt{\frac{|\mathcal{S}|}{N_h^k(s,a)}}\right) + \widehat{P}_h V_{h+1}(s,a)$
  end for
  for $s \in \mathcal{S}$ **do**
    $V_h(s) = \langle \pi_h(\cdot|s), Q_h(s,\cdot) \rangle$
  end for
end for
Output: The Optimistic value function V and Q-value function Q.

Upper Confidence Bound

Lemma 3.1: Upper Confidence Bound

For any policy π, let V and Q be the output of optimistic policy evaluation, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have

$$V_h(s) \geq V_h^\pi(s), \; Q_h(s, a) \geq Q_h^\pi(s, a), \; \forall (s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H].$$

Optimistic Mirror Ascent Policy Optimization

Algorithm 5 Optimistic Mirror Ascent Policy Optimization

Input: Uniformly initialized policy π^1, stepsize η

for $k = 1, 2, \cdots, K$ **do**

  Collect a trajectory via taking π^k

  Update the estimate of reward function and transition probability: $\widehat{r}^k$, $\widehat{P}^k$

  Obtain the optimistic Q-value function of π^k: $Q^k \leftarrow \mathsf{PE}\left(\pi^k, \widehat{r}^k + \text{bonus}, \widehat{P}^k\right)$

  Perform mirror ascent update:

  for $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$ **do**

    $$\pi_h^{k+1}(a \mid s) = \frac{\pi_h^k(a|s)\exp\left(\eta Q_h^k(s,a)\right)}{\sum_{a'} \pi_h^k(a'|s)\exp\left(\eta Q_h^k(s,a')\right)}$$

  end for

end for

Optimistic Mirror Ascent Policy Optimization

Theorem 3.1: Regret of OMAPO

For any $\delta \in (0,1)$, consider the optimistic mirror ascent policy optimization algorithm with $\eta = \sqrt{\frac{2\log(|\mathcal{A}|)}{H^2 K}}$, w.p. $1-\delta$, we have that

$$\sum_{k=1}^{K} V_1^*(s_1) - V_1^{\pi^k}(s_1) \leq \widetilde{\mathcal{O}}\left(\sqrt{|S|^2|\mathcal{A}|H^4 K}\right).$$

- Compared with the regret of $\widetilde{\mathcal{O}}\left(\sqrt{H^4 K}\right)$ under the known MDP setting, the regret under the online setting suffers an additional dependency on $|\mathcal{S}|$ and $|\mathcal{A}|$.

Analysis: the Regret of OMAPO

Lemma 3.2: Regret Decomposition

$$\sum_{k=1}^{K} V_1^{\pi^*} - V_1^{\pi^k} = \sum_{k=1}^{K} V_1^{\pi^*} - V_1^{k} + V_1^{k} - V_1^{\pi^k}$$

$$= \mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H} \left\langle \pi_h^*(\cdot|s_h) - \pi_h^k(\cdot|s_h), Q_h^k(s_h,\cdot)\right\rangle \middle| \pi^*, P\right] \quad \text{(I)}$$

$$+ \mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H} r_h(s_h,a_h) - \widehat{r}_h^k(s_h,a_h) + \left(P_h - \widehat{P}_h^k\right) V_{h+1}^k(s_h,a_h) - \text{bonus} \middle| \pi^*, P\right] \quad \text{(II)}$$

$$+ \sum_{k=1}^{K} V_1^{k} - V_1^{\pi^k} \quad \text{(III)}$$

Analysis: the Regret of OMAPO

Term (I): $= \mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H}\left\langle \pi_h^*(\cdot|s_h) - \pi_h^k(\cdot|s_h), Q_h^k(s_h,\cdot)\right\rangle \middle| \pi^*, P\right]$

- Note that π^{k+1} is obtained via mirror ascent w.r.t Q^k, i.e., $\pi_h^{k+1}(a|s) \propto \exp\left(\eta Q_h^k(s,a)\right)$.
- We can again leverage the regret of mirror ascent on online linear optimization:

$$\text{Term (I)} \leq \sqrt{2\log\left(|\mathcal{A}|\right) H^4 K}.$$

- This upper bound is the same as the regret of MAPO under the known MDP setting.

Analysis: the Regret of OMAPO

Term (II): $\mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H} r_h(s_h,a_h) - \widehat{r}_h^k(s_h,a_h) + \left(P_h - \widehat{P}_h^k\right) V_{h+1}^k(s_h,a_h) - \text{bonus}\middle|\pi^*, P\right]$

- For all $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, with high probability, we have

$$r_h(s,a) - \widehat{r}_h^k(s,a) - \widetilde{\mathcal{O}}\left(\sqrt{\frac{1}{N_h^k(s,a)}}\right) + \left(P_h - \widehat{P}_h^k\right) V_{h+1}^k(s,a) - \widetilde{\mathcal{O}}\left(H\sqrt{\frac{|\mathcal{S}|}{N_h^k(s,a)}}\right) \le 0.$$

- Due to the optimism, we get that Term (II) ≤ 0.

Analysis: the regret of OMAPO

Term (III): $\sum_{k=1}^{K} V_1^k - V_1^{\pi^k}$

- Recall that $V_1^k = \text{PE}(\pi^k, \widehat{r}^k + \text{bonus}, \widehat{P}^k)$ and $V_1^{\pi^k} = \text{PE}\left(\pi^k, r, P\right)$.
- With $|r - \widehat{r}^k| + \left|\left(\widehat{P}^k - P\right) V^k\right| \leq \text{bonus}$, $\sum_{k=1}^{K} V_1^k - V_1^{\pi^k}$ can be upper bounded by the bonus function.

$$\text{Term (III)} \leq \sum_{k=1}^{K} \mathbb{E}\left[\sum_{h=1}^{H} \widehat{r}_h^k(s_h, a_h) + \text{bonus} - r_h^k(s_h, a_h) + \left(\widehat{P}_h^k - P_h^k\right) V_h^k(s_h, a_h) \middle| \pi_k, P\right]$$
$$\leq 2\sum_{k=1}^{K} \mathbb{E}\left[\sum_{h=1}^{H} \widetilde{\mathcal{O}}\left(\sqrt{\frac{1}{N_h^k(s_h, a_h)}}\right) + \widetilde{\mathcal{O}}\left(H\sqrt{\frac{|\mathcal{S}|}{N_h^k(s_h, a_h)}}\right) \middle| \pi_k, P\right]$$

- As k increases, $N_h^k(s, a)$ increases and the size of reward bonus gets smaller.

Analysis: the Regret of OMAPO

Lemma 3.3

For any $\delta \in (0,1)$, w.p. $\geq 1-\delta$, Term (III) $= \sum_{k=1}^{K} V_1^k - V_1^{\pi^k} \leq \widetilde{\mathcal{O}}\left(\sqrt{|\mathcal{S}|^2|\mathcal{A}|H^4K}\right)$

- The total regret of OMAPO is dominated by Term (III).
- The order on $|\mathcal{S}|$ is $\mathcal{O}\left(\sqrt{|\mathcal{S}|^2}\right)$, which comes from the size of bonus function and the size of MDP.
- The order on H is of $\mathcal{O}\left(\sqrt{H^4}\right)$, which comes from the size of bonus function, the size of MDP and the total timesteps.

Outline

Conclusions

- We first introduce **mirror ascent policy optimization** (MAPO) for solving MDP. We prove its regret bound from the connection to online linear optimization.
- Under the online setting, to balance between exploration and exploitation, we incorporate the principle of optimism in the face of uncertainty into MAPO and show its regret bound.
- Compared with some SOTA value-based methods (e.g., UCB-VI [Azar et al., 2017]), the regret of Optimistic MAPO is still sub-optimal. How to design a more efficient policy-based method is an interesting future direction.

Outline

Policy Difference Lemma

Lemma 5.1: Policy Difference Lemma

For any policy π and π', we have

$$V_1^{\pi}(s_1) - V_1^{\pi'}(s_1) = \mathbb{E}\left[\sum_{h=1}^{H} \left\langle \pi_h(\cdot|s_h) - \pi'_h(\cdot|s_h), Q_h^{\pi'}(s_h, \cdot) \right\rangle \middle| \pi \right].$$

Proof of Policy Difference Lemma

- This proof is based on a simple recursion.

$$
\begin{aligned}
& V_1^{\pi}(s_1) - V_1^{\pi'}(s_1) \\
&= \langle \pi_1(\cdot|s_1), Q_1^{\pi}(s_1,\cdot)\rangle - \left\langle \pi_1'(\cdot|s_1), Q_1^{\pi'}(s_1,\cdot)\right\rangle \\
&= \left\langle \pi_1(\cdot|s_1), Q_1^{\pi}(s_1,\cdot) - Q_1^{\pi'}(s_1,\cdot)\right\rangle + \left\langle \pi_1(\cdot|s_1) - \pi_1'(\cdot|s_1), Q_1^{\pi'}(s_1,\cdot)\right\rangle \\
&= \left\langle \pi_1(\cdot|s_1) - \pi_1'(\cdot|s_1), Q_1^{\pi'}(s_1,\cdot)\right\rangle + \mathbb{E}\left[Q_1^{\pi}(s_1,a_1) - Q_1^{\pi'}(s_1,a_1)\middle| a_1 \sim \pi_1(\cdot|s_1)\right].
\end{aligned}
$$

- For $Q_1^{\pi}(s_1,a_1) - Q_1^{\pi'}(s_1,a_1)$, we have

$$
\begin{aligned}
Q_1^{\pi}(s_1,a_1) - Q_1^{\pi'}(s_1,a_1) &= r_1(s_1,a_1) + P_1 V_2^{\pi}(s_1,a_1) - r_1(s_1,a_1) - P_1 V_2^{\pi'}(s_1,a_1) \\
&= P_1\left(V_2^{\pi} - V_2^{\pi'}\right)(s_1,a_1) = \mathbb{E}\left[V_2^{\pi}(s_2) - V_2^{\pi'}(s_2)\middle| s_2 \sim P_1(\cdot|s_1,a_1)\right].
\end{aligned}
$$

Proof of Policy Difference Lemma

$$
\begin{aligned}
& V_1^{\pi}(s_1) - V_1^{\pi'}(s_1) \\
&= \left\langle \pi_1(\cdot|s_1) - \pi_1'(\cdot|s_1), Q_1^{\pi'}(s_1, \cdot) \right\rangle + \mathbb{E}\left[V_2^{\pi}(s_2) - V_2^{\pi'}(s_2) \middle| a_1 \sim \pi_1(\cdot|s_1), s_2 \sim P_1(\cdot|s_1, a_1) \right].
\end{aligned}
$$

- Expanding this equation for H steps with $V_{H+1}^{\pi}(s) - V_{H+1}^{\pi'}(s) = 0,\ \forall s \in \mathcal{S}$ yields the desired result.

Regret Decomposition

Lemma 5.2: Regret Decomposition

$$\sum_{k=1}^{K} V_1^{\pi^*} - V_1^{\pi^k} = \sum_{k=1}^{K} V_1^{\pi^*} - V_1^k + V_1^k - V_1^{\pi^k}$$

$$= \mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H} \left\langle \pi_h^*(\cdot|s_h) - \pi_h^k(\cdot|s_h), Q_h^k(s_h,\cdot)\right\rangle \middle| \pi^*, P\right]$$

$$+ \mathbb{E}\left[\sum_{k=1}^{K}\sum_{h=1}^{H} r_h(s_h,a_h) - \widehat{r}_h^k(s_h,a_h) + \left(P_h - \widehat{P}_h^k\right) V_{h+1}^k(s_h,a_h) - \text{bonus} \middle| \pi^*, P\right]$$

$$+ \sum_{k=1}^{K} V_1^k - V_1^{\pi^k}$$

Proof of the Regret Decomposition

- We first consider the term $V_1^{\pi^*} - V_1^k$. For any $h \in [H]$, we have

$$
\begin{aligned}
& V_h^{\pi^*}(s_h) - V_h^k(s_h) \\
&= \left\langle Q_h^{\pi^*}(s_h, \cdot), \pi_h^*(\cdot|s_h) \right\rangle - \left\langle Q_h^k(s_h, \cdot), \pi_h^k(\cdot|s_h) \right\rangle \\
&= \left\langle Q_h^{\pi^*}(s_h, \cdot) - {\color{blue}Q_h^k(s_h, \cdot)}, \pi_h^*(\cdot|s_h) \right\rangle + \left\langle {\color{blue}Q_h^k(s_h, \cdot)}, \left(\pi_h^* - \pi_h^k\right)(\cdot|s_h) \right\rangle \\
&= \left\langle Q_h^k(s_h, \cdot), \left(\pi_h^* - \pi_h^k\right)(\cdot|s_h) \right\rangle + \mathbb{E}\left[Q_h^{\pi^*}(s_h, a_h) - Q_h^k(s_h, a_h) \middle| a_h \sim \pi_h^*(\cdot|s_h) \right].
\end{aligned}
$$

Proof of the Regret Decomposition

- For $a_h \sim \pi_h^*(\cdot|s_h)$,

$$
\begin{aligned}
& Q_h^{\pi^*}(s_h, a_h) - Q_h^k(s_h, a_h) \\
&= r_h(s_h, a_h) + P_h V_h^{\pi^*}(s_h, a_h) - Q_h^k(s_h, a_h) \\
&= r_h(s_h, a_h) + P_h V_{h+1}^k(s_h, a_h) - Q_h^k(s_h, a_h) + P_h \left(V_{h+1}^{\pi^*} - V_{h+1}^k \right)(s_h, a_h) \\
&= r_h(s_h, a_h) + P_h V_{h+1}^k(s_h, a_h) - \widehat{r}_h^k(s_h, a_h) - \widehat{P}_h^k V_{h+1}^k(s_h, a_h) - \text{bonus} \\
&+ \mathbb{E}\left[V_{h+1}^{\pi^*}(s_{h+1}) - V_{h+1}^k(s_{h+1}) \middle| s_{h+1} \sim P_h(\cdot|s_h, a_h) \right].
\end{aligned}
$$

Proof of the Regret Decomposition

- Combining the above two equations yields

$$
\begin{aligned}
& V_h^{\pi^*}(s_h) - V_h^k(s_h) \\
&= \left\langle Q_h^k(s_h, \cdot), \left(\pi_h^* - \pi_h^k\right)(\cdot|s_h)\right\rangle \\
&+ \mathbb{E}\left[r_h(s_h, a_h) + P_h V_{h+1}^k(s_h, a_h) - \widehat{r}_h^k(s_h, a_h) - \widehat{P}_h^k V_{h+1}^k(s_h, a_h) - \text{bonus} \middle| a_h \sim \pi_h^*(\cdot|s_h)\right] \\
&+ \mathbb{E}\left[V_{h+1}^{\pi^*}(s_{h+1}) - V_{h+1}^k(s_{h+1}) \middle| a_h \sim \pi_h(\cdot|s_h), s_{h+1} \sim P_h(\cdot|s_h, a_h)\right].
\end{aligned}
$$

- Expanding this equation for $H - h$ times finishes the proof.

References I

A. Agarwal, S. M. Kakade, J. D. Lee, and G. Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. In Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria], volume 125 of Proceedings of Machine Learning Research, pages 64–66. PMLR, 2020.

A. Ayoub, Z. Jia, C. Szepesvári, M. Wang, and L. Yang. Model-based reinforcement learning with value-targeted regression. In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of Proceedings of Machine Learning Research, pages 463–474. PMLR, 2020.

M. G. Azar, I. Osband, and R. Munos. Minimax regret bounds for reinforcement learning. In Proceedings of the 34th International Conference on Machine Learning, pages 263–272, 2017.

References II

J. Bhandari and D. Russo. On the linear convergence of policy gradient methods for finite mdps. In The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event, volume 130 of Proceedings of Machine Learning Research, pages 2386–2394. PMLR, 2021.

Q. Cai, Z. Yang, C. Jin, and Z. Wang. Provably efficient exploration in policy optimization. In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of Proceedings of Machine Learning Research, pages 1283–1294. PMLR, 2020.

M. L. Puterman. Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons, 2014.

S. Shalev-Shwartz. Online learning and online convex optimization. Foundations and Trends in Machine Learning, 4(2):107–194, 2012.

References III

L. Shani, Y. Efroni, A. Rosenberg, and S. Mannor. Optimistic policy optimization with bandit feedback. In Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of Proceedings of Machine Learning Research, pages 8604–8613. PMLR, 2020.

M. J. Wainwright. High-dimensional statistics: A non-asymptotic viewpoint. Cambridge University Press, 2019.

T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M. J. Weinberger. Inequalities for the l1 deviation of the empirical distribution. Hewlett-Packard Labs, Techical Report, 2003.