Cooperative Online Learning in Stochastic and Adversarial MDPs

Tal Lancewicki[1] Aviv Rosenberg[1] Yishay Mansour[1,2]

[1]Tel Aviv University, Israel

[2]Google Research, Israel

Background

Regret minimization in RL

- A fundamental paradigm for sequential decision making.
- In each round the agent interacts with an unknown environment.

Background

Regret minimization in RL

- A fundamental paradigm for sequential decision making.
- In each round the agent interacts with an unknown environment.
- Costs can be either stochastic or adversarial.
- At the the end of each episode the agent observes feedback.

Background

Regret minimization in RL

- A fundamental paradigm for sequential decision making.
- In each round the agent interacts with an unknown environment.
- Costs can be either stochastic or adversarial.
- At the the end of each episode the agent observes feedback.

Cooperation in RL

- Multiple agent that learn the same environment share information in order to improve performance.

(Remark: no strategic aspects)

Background

Regret minimization in RL

- A fundamental paradigm for sequential decision making.
- In each round the agent interacts with an unknown environment.
- Costs can be either stochastic or adversarial.
- At the the end of each episode the agent observes feedback.

Cooperation in RL

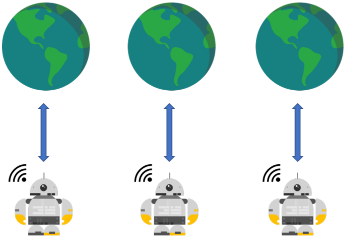
- Multiple agent that learn the same environment share information in order to improve performance.
- Applications: communication networks, traffic routing, robotics, etc.

(Remark: no strategic aspects)

Fresh vs Non-fresh randomness

Fresh randomness

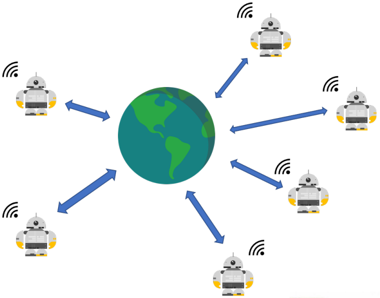
- Duplicates of the same environment - cost and transition to next state is freshly randomized.



E.g., Atari games.

Non-fresh randomness

- The same environment - agents that take the same action in the same state observe the same cost and next state.



Central questions

"How much can we gain from cooperation?"

Central questions

"How much can we gain from cooperation?"

"Is there a different limit for fresh and non-fresh randomness?"

Related work

- Optimal regret* in single-agent stochastic and adversarial MDPs. [Zimin and Neu, 2013, Rosenberg and Mansour, 2019, Jin et al., 2020]

$\sqrt{H^2K}$ (known transition full-info) $\sqrt{H^3SAK}$ (unknown transition bandit feedback)

K - #episodes A - #actions S - #states H - horizon m - #agents

*Regret bounds in this presentation ignore constants and poly-logarithmic factors

Related work

- Optimal regret* in single-agent stochastic and adversarial MDPs. [Zimin and Neu, 2013, Rosenberg and Mansour, 2019, Jin et al., 2020]

$$\sqrt{H^2K}$$
(known transition full-info)

$$\sqrt{H^3SAK}$$
(unknown transition bandit feedback)

- In multi-agent adversarial MAB, one can achieve regret that scales as [Cesa-Bianchi et al., 2019],

$$\sqrt{K} + \sqrt{AK/m}.$$

K - #episodes A - #actions S - #states H - horizon m - #agents

*Regret bounds in this presentation ignore constants and poly-logarithmic factors

Related work

- Optimal regret* in single-agent stochastic and adversarial MDPs. [Zimin and Neu, 2013, Rosenberg and Mansour, 2019, Jin et al., 2020]

$$\sqrt{H^2 K}$$
(known transition full-info)

$$\sqrt{H^3 SAK}$$
(unknown transition bandit feedback)

- In multi-agent adversarial MAB, one can achieve regret that scales as [Cesa-Bianchi et al., 2019],

$$\sqrt{K} + \sqrt{AK/m}.$$

- Cooperation in RL was considered only in the stochastic and fresh randomness case by Lidard et al. [2021],

$$\sqrt{H^4 SAK/m}.$$

K - #episodes A - #actions S - #states H - horizon m - #agents

*Regret bounds in this presentation ignore constants and poly-logarithmic factors

Our contribution

- We are the first to study non-fresh randomness, and to face new challenges in this model.
- First to consider adversarial cost in cooperative learning in MDPs.
- Thoroughly analyze all relevant settings, and prove nearly-matching regret lower and upper bounds.

Problem Setup

For each episode k:

- Choose a policy for each of the m agents.

Problem Setup

For each episode k:

- Choose a policy for each of the m agents.
- Agents start at an initial state s_1. At each time $h = 1, ..., H$:
  - Each agent sample an action $a_h \sim \pi^k(\cdot \mid s_h)$.
  - Agent suffers cost $c_h(s_h, a_h)$ and transition to a new state $s_{h+1} \sim p_h(\cdot \mid s_h, a_h)$, where p is unknown.

Problem Setup

For each episode k:

- Choose a policy for each of the m agents.
- Agents start at an initial state s_1. At each time $h = 1, ..., H$:
  - Each agent sample an action $a_h \sim \pi^k(\cdot \mid s_h)$.
  - Agent suffers cost $c_h(s_h, a_h)$ and transition to a new state $s_{h+1} \sim p_h(\cdot \mid s_h, a_h)$, where p is unknown.
    (with non-fresh randomness the next state is sampled once for each state-action pair)

Problem Setup

For each episode k:

- Choose a policy for each of the m agents.
- Agents start at an initial state s_1. At each time $h = 1, ..., H$:
  - Each agent sample an action $a_h \sim \pi^k(\cdot \mid s_h)$.
  - Agent suffers cost $c_h(s_h, a_h)$ and transition to a new state $s_{h+1} \sim p_h(\cdot \mid s_h, a_h)$, where p is unknown.
    (with non-fresh randomness the next state is sampled once for each state-action pair)
- The learner observe the trajectory and the costs over the trajectory (i.e., bandit feedback) of all agents.

Problem Setup

For each episode k:

- Choose a policy for each of the m agents.
- Agents start at an initial state s_1. At each time $h = 1, ..., H$:
  - Each agent sample an action $a_h \sim \pi^k(\cdot \mid s_h)$.
  - Agent suffers cost $c_h(s_h, a_h)$ and transition to a new state $s_{h+1} \sim p_h(\cdot \mid s_h, a_h)$, where p is unknown. (with non-fresh randomness the next state is sampled once for each state-action pair)
- The learner observe the trajectory and the costs over the trajectory (i.e., bandit feedback) of all agents.

Regret

The performance is measured by the *regret* - the difference between the total agent's cost and the cost of the best policy in hindsight.

Fresh Randomness - Stochastic MDP

- The basic approach for single-agent stochastic MDPs is *"Optimism Under Uncertainty"*.
- Compute an optimistic estimate of Q^* and act greedily with respect to it.

Fresh Randomness - Stochastic MDP

- The basic approach for single-agent stochastic MDPs is *"Optimism Under Uncertainty"*.
- Compute an optimistic estimate of Q^* and act greedily with respect to it.
- One can show that the regret scales as the sum of confidence radius on the agent's trajectory.

Fresh Randomness - Stochastic MDP

- The basic approach for single-agent stochastic MDPs is *"Optimism Under Uncertainty"*.
- Compute an optimistic estimate of Q^* and act greedily with respect to it.
- One can show that the regret scales as the sum of confidence radius on the agent's trajectory.
- With non-fresh randomness we get m times more samples and the confidence radius shrinks faster. With that we can show optimal regret for each agent:

$$R_K \lesssim \sqrt{\frac{H^3 SAK}{m}}.$$

Non-fresh - Stochastic MDP

Much more challenging setting:

- If agents play a deterministic policy (e.g., optimistic algorithm), then they all follow the same trajectory. Hence, we don't have additional feedback.

Non-fresh - Stochastic MDP

Much more challenging setting:

- If agents play a deterministic policy (e.g., optimistic algorithm), then they all follow the same trajectory. Hence, we don't have additional feedback.
- Optimism alone is no longer a good approach.

Non-fresh - Stochastic MDP

Much more challenging setting:

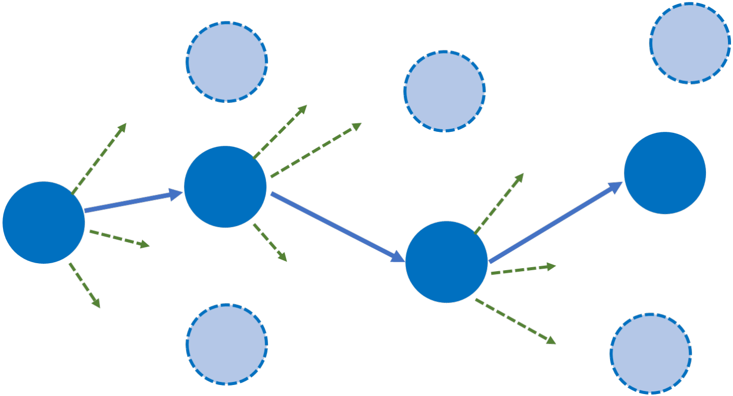
- If agents play a deterministic policy (e.g., optimistic algorithm), then they all follow the same trajectory. Hence, we don't have additional feedback.
- Optimism alone is no longer a good approach.
- In fact, we show a lower bound of $\sqrt{H^2SK}$ regardless on the number of agents!

Non-fresh - Stochastic MDP

Algorithm (COOP-ULCAE):

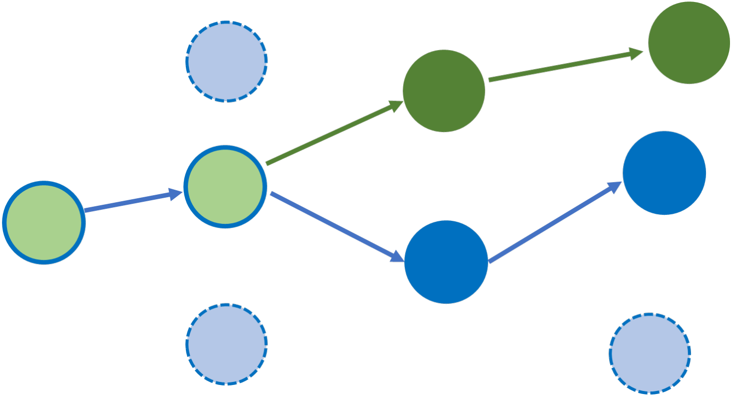
- Maintain upper and lower confidence bounds on Q^*.
- Eliminate arms a such that $\underline{Q}_h^k(s,a) > \overline{Q}_h^k(s,a')$.
- With probability $1-\epsilon$ play the optimistic policy.
- With probability ϵ:
  - Sample random h.
  - At time h take a random active action.
  - At the rest of the time play the optimistic policy.

Analysis



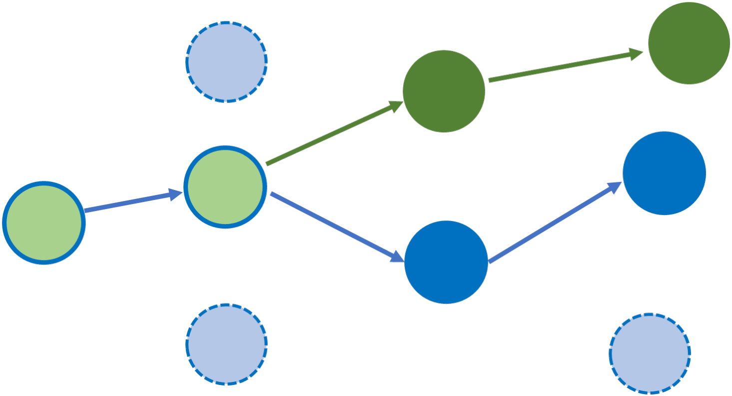
- On the optimistic policy path we obtain ϵm times more feedback.
- Hence, the regret in these rounds is at most $\sqrt{\frac{SAK}{m\epsilon}}$.

Analysis



- We take an active action with the exploration policy. Using that, we can show that the regret is similar to a single-agent regret.

Analysis



- We take an active action with the exploration policy. Using that, we can show that the regret is similar to a single-agent regret.
- Each agent explores only $O(\epsilon K)$ episodes.
- Hence, the total regret from these rounds is $\sqrt{SAK\epsilon}$

Non-fresh - Stochastic MDP

Setting ϵ properly allows us to prove the following regret bound.

Theorem

Under non-fresh randomness and stochastic costs COOP-ULCAE guarantees individual regret of,

$$R_K \lesssim \sqrt{H^5 SK} + \sqrt{\frac{H^7 SAK}{\sqrt{m}}}.$$

Adversarial cost

- The adversarial setting is a very general model which generalizes stochastic costs.
- It is more challenging to estimate the cost.

Adversarial cost

- The adversarial setting is a very general model which generalizes stochastic costs.
- It is more challenging to estimate the cost.
- We use an importance-sampling estimator

$$\hat{c}(s,a) = \frac{\mathbb{I}\{\text{“}\textit{some}\text{ agent visited } s \text{ and took } a\text{”}\}}{\Pr(\text{“}\textit{some}\text{ agent visited } s \text{ and took } a\text{”})} \cdot c(s,a)$$

Adversarial cost

- The adversarial setting is a very general model which generalizes stochastic costs.
- It is more challenging to estimate the cost.
- We use an importance-sampling estimator

$$\hat{c}(s,a) = \frac{\mathbb{I}\{\text{``}\textit{some} \text{ agent visited } s \text{ and took } a\text{''}\}}{\Pr(\text{``}\textit{some} \text{ agent visited } s \text{ and took } a\text{''})} \cdot c(s,a)$$

- This is an unbiased estimator.

Adversarial cost

- The adversarial setting is a very general model which generalizes stochastic costs.
- It is more challenging to estimate the cost.
- We use an importance-sampling estimator

$$\hat{c}(s,a) = \frac{\mathbb{I}\{\text{``}\textit{some} \text{ agent visited } s \text{ and took } a\text{''}\}}{\Pr(\text{``}\textit{some} \text{ agent visited } s \text{ and took } a\text{''})} \cdot c(s,a)$$

- This is an unbiased estimator.
- We can show small variance with multi agent, which allows us to show lower regret.

Adversarial cost

- The adversarial setting is a very general model which generalizes stochastic costs.
- It is more challenging to estimate the cost.
- We use an importance-sampling estimator

$$\hat{c}(s,a) = \frac{\mathbb{I}\{\text{“}\textit{some} \text{ agent visited } s \text{ and took } a\text{”}\}}{\Pr(\text{“}\textit{some} \text{ agent visited } s \text{ and took } a\text{”})} \cdot c(s,a)$$

- This is an unbiased estimator.
- We can show small variance with multi agent, which allows us to show lower regret.
- More challenging analysis under non-fresh randomness.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K}+\sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK}+\sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m=\sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K} + \sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK} + \sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m = \sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K} + \sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK} + \sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m = \sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\boxed{\sqrt{H^2K}} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\boxed{\sqrt{H^2K}} + \sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK} + \sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m = \sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3 SAK}{m}}$	$\sqrt{\frac{H^3 SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2 K} + \sqrt{\frac{H^2 SAK}{m}}$	$\sqrt{H^2 K} + \sqrt{\frac{H^2 SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2 K} + \sqrt{\frac{H^4 S^2 AK}{m}}$	$\sqrt{H^2 K} + \sqrt{\frac{H^3 SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5 SK} + \sqrt{\frac{H^7 SAK}{\sqrt{m}}}$	$\sqrt{H^2 SK} + \sqrt{\frac{H^3 SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2 SK} + \sqrt{\frac{H^2 SAK}{m}}$	$\sqrt{H^2 SK} + \sqrt{\frac{H^2 SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4 S^2 K}$ (*)	$\sqrt{H^2 SK} + \sqrt{\frac{H^3 SAK}{m}}$

(*) The algorithm requires $m = \sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K} + \sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK} + \sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK} + \sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK} + \sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m = \sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K}+\sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK}+\sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m=\sqrt{K}$ agents.

Summary of our results

Setting	Regret	Lower Bound
Fresh, stochastic, unknown p	$\sqrt{\frac{H^3SAK}{m}}$	$\sqrt{\frac{H^3SAK}{m}}$
Fresh, adversarial, known p	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^2SAK}{m}}$
Fresh, adversarial, unknown p	$\sqrt{H^2K}+\sqrt{\frac{H^4S^2AK}{m}}$	$\sqrt{H^2K}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, stochastic, unknown p	$\sqrt{H^5SK}+\sqrt{\frac{H^7SAK}{\sqrt{m}}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$
Non-fresh, adversarial, known p	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$	$\sqrt{H^2SK}+\sqrt{\frac{H^2SAK}{m}}$
Non-fresh, adversarial, unknown p	$\sqrt{H^4S^2K}$ (*)	$\sqrt{H^2SK}+\sqrt{\frac{H^3SAK}{m}}$

(*) The algorithm requires $m=\sqrt{K}$ agents.

Thank you

N. Cesa-Bianchi, C. Gentile, and Y. Mansour. Delay and cooperation in nonstochastic bandits. *Journal of Machine Learning Research*, 20 (17):1–38, 2019.

C. Jin, T. Jin, H. Luo, S. Sra, and T. Yu. Learning adversarial markov decision processes with bandit feedback and unknown transition. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2020.

J. Lidard, U. Madhushani, and N. E. Leonard. Provably efficient multi-agent reinforcement learning with fully decentralized communication. *arXiv preprint arXiv:2110.07392*, 2021.

A. Rosenberg and Y. Mansour. Online convex optimization in adversarial markov decision processes. In *International Conference on Machine Learning*, pages 5478–5486. PMLR, 2019.

A. Zimin and G. Neu. Online learning in episodic markovian decision processes by relative entropy policy search. In *Neural Information Processing Systems 26*, 2013.