# V-Learning—A Simple, Efficient, Decentralized Algorithm for Multiagent RL

Chi Jin
Princeton University
chij@princeton.edu

Qinghua Liu
Princeton University
qinghual@princeton.edu

Yuanhao Wang
Princeton University
yuanhao@princeton.edu

Tiancheng Yu
MIT
yutc@mit.edu

October 28, 2021

**Abstract**

A major challenge of multiagent reinforcement learning (MARL) is *the curse of multiagents*, where the size of the joint action space scales exponentially with the number of agents. This remains to be a bottleneck for designing efficient MARL algorithms even in a basic scenario with finitely many states and actions. This paper resolves this challenge for the model of episodic Markov games. We design a new class of fully decentralized algorithms—V-learning, which provably learns Nash equilibria (in the two-player zero-sum setting), correlated equilibria and coarse correlated equilibria (in the multiplayer general-sum setting) in a number of samples that only scales with $\max_{i\in[m]} A_i$, where $A_i$ is the number of actions for the $i^{\text{th}}$ player. This is in sharp contrast to the size of the joint action space which is $\prod_{i=1}^m A_i$.

V-learning (in its basic form) is a new class of single-agent RL algorithms that convert any adversarial bandit algorithm with suitable regret guarantees into a RL algorithm. Similar to the classical Q-learning algorithm, it performs incremental updates to the value functions. Different from Q-learning, it only maintains the estimates of V-values instead of Q-values. This key difference allows V-learning to achieve the claimed guarantees in the MARL setting by simply letting all agents run V-learning independently.

## 1 Introduction

A wide range of modern artificial intelligence challenges can be cast as multi-agent reinforcement learning (MARL) problems, in which agents learn to make a sequence of decisions in the presence of other agents whose decisions will influence the outcome and can adapt to the strategies of the agents. Modern MARL systems have achieved significant success recently on a rich set of traditionally challenging tasks, including the game of GO [44, 45], Poker [8], real-time strategy games [51, 34], decentralized controls or multiagent robotics systems [6], autonomous driving [41], as well as complex social scenarios such as hide-and-seek [4]. While single-agent RL has been the focus of recent intense theoretical study, MARL has been comparatively underexplored, which leaves several fundamental questions open even in the basic model of *Markov games* [42] with finitely many states and actions.

Table 1: A summary of sample complexities for V-learning under different settings. Here $H$ is the length of each episode, $S$ is the number of states, $A = \max_{i\in[m]} A_i$ is the largest number of actions for each agent, $\epsilon$ is the error tolerance. An information theoretical lower bound for all objectives is $\Omega(H^3SA/\epsilon^2)$ [22, 2].

| **Objective** | **Multiplayer General-sum** | |
|---|---|---|
| | **Two-player Zero-sum** | - |
| Nash Equilibria | $\tilde{\mathcal{O}}(H^5SA/\epsilon^2)$ | PPAD-complete |
| Coarse Correlated Equilibria | $\tilde{\mathcal{O}}(H^5SA/\epsilon^2)$ | |
| Correlated Equilibria | $\tilde{\mathcal{O}}(H^5SA^2/\epsilon^2)$ | |

One such unique challenge of MARL is *the curse of multiagents*—let $A_i$ be the number of actions for the $i^{\text{th}}$ player, then the number of possible joint actions (as well as the number of parameters to specify a Markov game) scales with $\prod_{i=1}^m A_i$, which grows exponentially with the number of agents $m$. This remains to be a bottleneck even for the best existing algorithms for learning Markov games. In fact, a majority of these algorithms adapt the classical single-agent algorithms, such as value iteration or Q-learning, into the multiagent setting [3, 30], whose sample complexity scales at least linearly with respect to $\prod_{i=1}^m A_i$. This is prohibitively large in practice even for fairly small multiagent applications, say only ten agents are involved with ten actions available for each agent.

Another challenge of the MARL is to design *decentralized* algorithms. While a centralized algorithm requires the existence of a centralized controller which gathers all information and jointly optimizes the policies of all agents, a decentralized algorithm allows each agent to only observe her own actions and rewards while optimizing her own policy independently. Decentralized algorithms are often preferred over centralized algorithms in practice since (1) decentralized algorithms are typically cleaner, easier to implement; (2) decentralized algorithms are more versatile as the individual learners are indifferent to the interaction and the number of other agents; and (3) they are also faster due to less communication required. While several provable decentralized MARL algorithms have been developed [see, e.g., 57, 40, 13], they either have only asymptotic guarantees or work only under certain reachability assumptions (see Section 1.1). The existing provably *efficient* algorithms for general Markov games (without further assumptions) are exclusively centralized algorithms [2, 55, 30].

This motivates us to ask the following open question:

**Can we design *decentralized* MARL algorithms that *break the curse of multiagents?***

This paper addresses both challenges mentioned above, and provide the first positive answer to this question in the basic setting of tabular episodic Markov games. We propose a new class of single-agent RL algorithms—V-learning, which converts any adversarial bandit algorithm with suitable regret guarantees into a RL algorithm. Similar to the classical Q-learning algorithm, V-learning also performs incremental updates to the values. Different from Q-learning, V-learning only maintains the V-value functions instead of the Q-value functions. We remark that the number of parameters of Q-value functions in MARL is $\mathcal{O}(S\prod_{i=1}^m A_i)$, where $S$ is the number of states, while the number of parameters of V-value functions is only $\mathcal{O}(S)$. This key difference allows V-learning to be readily extended to the MARL setting by simply letting all agents run V-learning independently, which gives a fully *decentralized* algorithm.

We consider the standard learning objectives in the game theory—Nash equilibrium (NE), correlated

equilibrium (CE) and coarse correlated equilibrium (CCE). Except for the task of finding a NE in multiplayer general-sum games which is PPAD-complete even for matrix games, we prove that V-learning finds a NE for two-player zero-sum Markov games within $\tilde{\mathcal{O}}(H^5SA/\epsilon^2)$ episodes, where $H$ is the length of each episode, $A = \max_{i\in[m]} A_i$ is the maximum number of actions of all agents, and $\epsilon$ is the error tolerance for the objective. We further prove that for multiplayer general-sum Markov games, V-learning coupled with suitable adversarial bandit algorithms is capable of finding a CCE within $\tilde{\mathcal{O}}(H^5SA/\epsilon^2)$ episodes, and a CE within $\tilde{\mathcal{O}}(H^5SA^2/\epsilon^2)$ episodes. All sample complexities mentioned above do not grow with the number of the agents, which thus *break the curse of multiagents* (See Table 1 for a summary).

### 1.1 Related work

In this section, we focus our attention on theoretical results for the tabular setting, where the numbers of states and actions are finite. We acknowledge that there has been much recent work in RL for continuous state spaces [see, e.g., 21, 23, 56, 24, 55, 25], but this setting is beyond our scope.

**Markov games.** Markov Game (MG), also known as stochastic game [42], is a popular model in multi-agent RL [28]. Early works have mainly focused on finding Nash equilibria of MGs under strong assumptions, such as known transition and reward [29, 17, 15, 53], or certain reachability conditions [52, 54] (e.g., having access to simulators [20, 43, 58]) that alleviate the challenge in exploration.

A recent line of works provide non-asymptotic guarantees for learning two-player zero-sum tabular MGs without further structural assumptions. Bai and Jin [2] and Xie et al. [55] develop the first provably-efficient learning algorithms in MGs based on optimistic value iteration. Liu et al. [30] improves upon these works and achieve best-known sample complexity for finding an $\epsilon$-Nash equilibrium—$\mathcal{O}(H^3SA_1A_2/\epsilon^2)$ episodes.

For multiplayer general-sum tabular MGs, Liu et al. [30] is the only existing work that provides non-asymptotic guarantees in the exploration setting. It proposes centralized model-based algorithms based on value-iteration, and shows that Nash equilibria (although computationally inefficient), CCE and CE can be all learned within $\mathcal{O}(H^4S^2\prod_{j=1}^m A_j/\epsilon^2)$ episodes. Note this result suffers from the curse of multiagents.

V-learning—initially coupled with the FTRL algorithm as adversarial bandit subroutine—is firstly proposed in the conference version of this paper [3], for finding Nash equilibria in the two-player zero-sum setting. During the preparation of this draft, we note two very recent independent works [47, 32], whose results partially overlap with the results of this paper in the multiplayer general-sum setting. In particular, Mao and Başar [32] use V-learning with stablized online mirror descent as adversarial bandit subroutine, and learn $\epsilon$-CCE in $\mathcal{O}(H^6SA/\epsilon^2)$ episodes, where $A = \max_{j\in[m]} A_j$. This is one $H$ factor larger than what is required in Theorem 6 of this paper. Song et al. [47] considers similar V-learning style algorithms for learning both $\epsilon$-CCE and $\epsilon$-CE. For the latter objective, they require $\mathcal{O}(H^6SA^2/\epsilon^2)$ episodes which is again one $H$ factor larger than what is required in Theorem 7 of this paper. Song et al. [47] also considers Markov potential games, which is beyond the scope of this paper. We remark that both parallel works have not presented V-learning as a generic class of algorithms which can be coupled with any adversarial bandit algorithms with suitable regret guarantees in a black-box fashion.

**Strategic games.** Strategic game is one of the most basic game forms studied in the game theory literature [37]. It can be viewed as Markov games *without* state and transition. The fully decentralized algorithm that breaks the curse of multiagents is known in the setting of strategic games. By independently running no-regret (or no-swap-regret) algorithm for all agents, one can find Nash Equilibria (in the two-player zero-sum setting), correlated equilibria and coarse correlated equilibria (in the multiplayer general-sum setting) in a

number of samples that only scales with $\max_{i\in[m]} A_i$ [10, 16, 5]. However, such successes do not directly extend to the Markov games due to the additional temporal structures involving both states and transition. In particular, there is no computationally efficient no-regret algorithm for Markov games [38, 3].

**Extensive-form games.** There is another long line of research on MARL based on the model of extensive-form games (EFG) [see, e.g., 26, 14, 59, 7, 8, 9]. EFGs can be viewed as special cases of Markov games where any state at the $h^{\text{th}}$ step can be reached from only one state at the $(h-1)^{\text{th}}$ step (due to tree structure of the game). Therefore, results on learning EFGs do not directly imply results for learning MGs.

**Decentralized MARL** There is a long line of *empirical* works on decentralized MARL [see, e.g., 31, 18, 49, 39, 46]. A majority of these works focus on the cooperative setting. They additionally attack the challenge where each agent can only observe a part of the underlying state, which is beyond the scope of this paper. For theoretical results, Zhang et al. [57] consider the cooperative setting while Sayin et al. [40] study the two-player zero-sum Markov games. Both develop decentralized MARL algorithms but provide only asymptotic guarantees. Daskalakis et al. [13] analyze the convergence rate of independent policy gradient method in episodic two-player zero-sum MGs. Their result requires the additional reachability assumptions (concentrability) which alleviates the difficulty of exploration.

**Single-agent RL** There is a rich literature on reinforcement learning in MDPs [see e.g. 19, 36, 1, 11, 48, 22, 24]. MDPs are special cases of Markov games, where only a single agent interacts with a stochastic environment. For the tabular episodic setting with nonstationary dynamics and no simulators, the best sample complexity achieved by existing model-based and model-free algorithms are $\tilde{\mathcal{O}}(H^3SA/\epsilon^2)$ (achieved by value iteration [1]) and $\tilde{\mathcal{O}}(H^4SA/\epsilon^2)$ (achieved by Q-learning [22]), respectively, where $S$ is the number of states, $A$ is the number of actions, $H$ is the length of each episode. Both of them (nearly) match the lower bound $\Omega(H^3SA/\epsilon^2)$ [19, 35, 22].

## 2 Preliminaries

We consider the model of Markov Games (MG) [42] (also known as stochastic games in the literature) in its most generic—multiplayer general-sum form. Formally, we denote an tabular episodic MG with $m$ players by a tuple $\mathrm{MG}(H, \mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, \mathbb{P}, \{r_i\}_{i=1}^m)$, where $H$, $\mathcal{S}$ denote the length of each episode and the state space with $|\mathcal{S}| = S$. $\mathcal{A}_i$ denotes the action space for the $i^{\text{th}}$ player and $|\mathcal{A}_i| = A_i$. We let $\boldsymbol{a} := (a_1, \cdots, a_m)$ denote the (tuple of) joint actions by all $m$ players, and $\mathcal{A} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_m$. $\mathbb{P} = \{\mathbb{P}_h\}_{h\in[H]}$ is a collection of transition matrices, so that $\mathbb{P}_h(\cdot|s, \boldsymbol{a})$ gives the distribution of the next state if actions $\boldsymbol{a}$ are taken at state $s$ at step $h$, and $r_i = \{r_{i,h}\}_{h\in[H]}$ is a collection of reward functions for the $i^{\text{th}}$ player, so that $r_{i,h}(s, \boldsymbol{a}) \in [0, 1]$ gives the deterministic reward received by the $i^{\text{th}}$ player if actions $\boldsymbol{a}$ are taken at state $s$ at step $h$. [1] We remark that since the relation among the rewards of different agents can be arbitrary, this model of MGs incorporates both cooperation and competition.

In each episode, we start with a *fixed initial state* $s_1$. [2] At each step $h \in [H]$, each player $i$ observes state $s_h \in \mathcal{S}$, picks action $a_{i,h} \in \mathcal{A}_i$ simultaneously, observes the actions played by other players,[3] and receives

[1] Our results directly generalize to random reward functions, since learning transitions is more difficult than learning rewards.

[2] While we assume a fixed initial state for notational simplicity, our results readily extend to the setting where the initial state is sampled from a fixed initial distribution.

[3] We assume the knowledge of other players' actions for the convenience of later definitions. We remark that the V-learning algorithm introduced in this paper does *not* require knowing the actions of other players.

her own reward $r_{i,h}(s_h, \boldsymbol{a}_h)$. Then the environment transitions to the next state $s_{h+1} \sim \mathbb{P}_h(\cdot|s_h, \boldsymbol{a}_h)$. The episode ends when $s_{H+1}$ is reached.

**Policy, value function** A (*random*) *policy* $\pi_i$ of the $i^{\text{th}}$ player is a set of $H$ maps $\pi_i := \left\{\pi_{i,h} : \Omega \times (\mathcal{S} \times \mathcal{A})^{h-1} \times \mathcal{S} \to \mathcal{A}_i\right\}_{h\in[H]}$, where $\pi_{i,h}$ maps a random sample $\omega$ from a probability space $\Omega$ and a history of length $h$—say $\tau_h := (s_1, \boldsymbol{a}_1, \cdots, s_h)$, to an action in $\mathcal{A}_i$. To execute policy $\pi_i$, we first draw a random sample $\omega$ at the beginning of the episode. Then, at each step $h$, the $i^{\text{th}}$ player simply takes action $\pi_{i,h}(\omega, \tau_h)$. We note here $\omega$ is shared among all steps $h \in [H]$. $\omega$ encodes both the correlation among steps and the individual randomness of each step. We further say a policy $\pi_i$ is *deterministic* if $\pi_{i,h}(\omega, \tau_h) = \pi_{i,h}(\tau_h)$ which is independent of the choice of $\omega$.

An important subclass of policy is *Markov policy*, which can be defined as $\pi_i := \left\{\pi_{i,h} : \Omega \times \mathcal{S} \to \mathcal{A}_i\right\}_{h\in[H]}$. Instead of depending on the entire history, a Markov policy takes actions only based on the current state. Furthermore, the randomness in each step of Markov policy is independent. Therefore, when it is clear from the context, we write Markov policy as $\pi_i := \left\{\pi_{i,h} : \mathcal{S} \to \Delta_{\mathcal{A}_i}\right\}_{h\in[H]}$, where $\Delta_{\mathcal{A}_i}$ denotes the simplex over $\mathcal{A}_i$. We also use notation $\pi_{i,h}(a|s)$ to denote the probability of the $i^{\text{th}}$ agent taking action $a$ at state $s$ at step $h$.

A joint (potentially correlated) policy is a set of policies $\{\pi_i\}_{i=1}^m$, where the same random sample $\omega$ is shared among all agents, which we denote as $\pi = \pi_1 \odot \pi_2 \odot \ldots \odot \pi_m$. We also denote $\pi_{-i} = \pi_1 \odot \ldots \pi_{i-1} \odot \pi_{i+1} \odot \ldots \odot \pi_m$ to be the joint policy excluding the $i^{\text{th}}$ player. A special case of joint policy is the *product policy* where the random sample has special form $\omega = (\omega_1, \ldots, \omega_m)$, and for any $i \in [m]$, $\pi_i$ only uses the randomness in $\omega_i$, which is independent of remaining $\{\omega_j\}_{j\neq i}$, which we denote as $\pi = \pi_1 \times \pi_2 \times \ldots \times \pi_m$.

We define the value function $V_{i,1}^{\pi}(s_1)$ as the expected cumulative reward that the $i^{\text{th}}$ player will receive if the game starts at initial state $s_1$ at the $1^{\text{st}}$ step and all players follow joint policy $\pi$:

$$V_{i,1}^{\pi}(s_1) := \mathbb{E}_{\pi}\left[\sum_{h=1}^{H} r_{i,h}(s_h, \boldsymbol{a}_h)\middle| s_1\right]. \tag{1}$$

where the expectation is taken over the randomness in transition and the random sample $\omega$ in policy $\pi$.

**Best response and strategy modification** For any strategy $\pi_{-i}$, the *best response* of the $i^{\text{th}}$ player is defined as a policy of the $i^{\text{th}}$ player which is independent of the randomness in $\pi_{-i}$, and achieves the highest value for herself conditioned on all other players deploying $\pi_{-i}$. In symbol, the best response is the maximizer of $\max_{\pi_i'} V_{i,1}^{\pi_i' \times \pi_{-i}}(s_1)$ whose value we also denote as $V_{i,1}^{\dagger,\pi_{-i}}(s_1)$ for simplicity. By its definition, we know the best response can always be achieved at *deterministic* policies.

A *strategy modification* $\phi_i$ for the $i^{\text{th}}$ player is a set of maps $\phi_i := \{\phi_{i,h} : (\mathcal{S} \times \mathcal{A})^{h-1} \times \mathcal{S} \times \mathcal{A}_i \to \mathcal{A}_i\}$,[4] where $\phi_{i,h}$ can depend on the history $\tau_h$ and maps actions in $\mathcal{A}_i$ to different actions in $\mathcal{A}_i$. For any policy of the $i^{\text{th}}$ player $\pi_i$, the modified policy (denoted as $\phi_i \diamond \pi_i$) changes the action $\pi_{i,h}(\omega, \tau_h)$ under random sample $\omega$ and history $\tau_h$ to $\phi_i(\tau_h, \pi_{i,h}(\omega, \tau_h))$. For any joint policy $\pi$, we define the best strategy modification of the $i^{\text{th}}$ player as the maximizer of $\max_{\phi_i} V_{i,1}^{(\phi_i \diamond \pi_i) \odot \pi_{-i}}(s_1)$.

Different from the best response, which is completely independent of the randomness in $\pi_{-i}$, the best strategy modification changes the policy of the $i^{\text{th}}$ player while still utilizing the shared randomness among $\pi_i$ and $\pi_{-i}$. Therefore, the best strategy modification is more powerful than the best response: formally one can show that $\max_{\phi_i} V_{i,1}^{(\phi_i \diamond \pi_i) \odot \pi_{-i}}(s_1) \geq \max_{\pi_i'} V_{i,1}^{\pi_i' \times \pi_{-i}}(s_1)$ for any policy $\pi$.

[4]Here, we only introduce the deterministic strategy modification for simplicity of notation, which is sufficient for discussion in the context of this paper. The random strategy modification can also be defined by introducing randomness in $\phi_i$ which is independent of randomness in $\pi_i$ and $\pi_{-i}$. It can be shown that the best strategy modification can always be deterministic.

### 2.1 Learning objectives

A special case of Markov game is Markov Decision Process (MDP). One can show there always exists an optimal policy $\pi^\star = \operatorname{argmax}_\pi V_1^\pi(s_1)$. Denote the value of the optimal policy as $V^\star$. The objective of learning MDPs is to find an $\epsilon$-optimal policy $\pi$, which satisfies $V_1^\star(s_1) - V_1^\pi(s_1) \le \epsilon$.

For Markov games, there are three common learning objectives in the game theory literature—Nash Equilibrium, Correlated Equilibrium (CE) and Coarse Correlated Equilibrium (CCE).

First, a Nash equilibrium is defined as a product policy where no player can increase her value by changing only her own policy. Formally,

**Definition 1** (Nash Equilibrium)**.** A *product* policy $\pi$ is a **Nash equilibrium** if $\max_{i\in[m]} (V_{i,1}^{\dagger,\pi_{-i}} - V_{i,1}^{\pi})(s_1) = 0$. A *product* policy $\pi$ is an $\epsilon$-approximate Nash equilibrium if $\max_{i\in[m]} (V_{i,1}^{\dagger,\pi_{-i}} - V_{i,1}^{\pi})(s_1) \le \epsilon$.

We remark that, except for the special case of two-player zero-sum Markov games where reward $r_{2,h} = -r_{1,h}$[5] for any $h \in [H]$, the Nash equilibrium in general has been proved PPAD-hard to compute [12]. Therefore, we only present results for finding Nash equilibria in two-player zero-sum MGs in this paper.

Second, a coarse correlated equilibrium is defined as a joint (potentially correlated) policy where no player can increase her value by playing a different independent strategy. In symbol,

**Definition 2** (Coarse Correlated Equilibrium)**.** A joint policy $\pi$ is a **CCE** if $\max_{i\in[m]} (V_{i,1}^{\dagger,\pi_{-i}} - V_{i,1}^{\pi})(s_1) = 0$. A joint policy $\pi$ is a $\epsilon$-approximate CCE if $\max_{i\in[m]} (V_{i,1}^{\dagger,\pi_{-i}} - V_{i,1}^{\pi})(s_1) \le \epsilon$.

The only difference between Definition 1 and Definition 2 is that Nash equilibrium requires the policy $\pi$ to be a product policy while CCE does not. Thus, it is clear that CCE is a relaxed notion of Nash equilibrium, and a Nash equilibrium is always a CCE.

Finally, a correlated equilibrium is defined as a joint (potentially correlated) policy where no player can increase her value by using a strategy modification. In symbol,

**Definition 3** (Correlated Equilibrium)**.** A joint policy $\pi$ is a **CE** if $\max_{i\in[m]} \max_{\phi_i} (V_{i,1}^{(\phi_i\diamond\pi_i)\odot\pi_{-i}} - V_{i,1}^{\pi})(s_1) = 0$. A joint policy $\pi$ is a $\epsilon$-approximate CE if $\max_{i\in[m]} \max_{\phi_i} (V_{i,1}^{(\phi_i\diamond\pi_i)\odot\pi_{-i}} - V_{i,1}^{\pi})(s_1) \le \epsilon$.

In Markov games, we also have that a Nash equilibrium is a CE, and a CE is a CCE (see Proposition 9 in Appendix A for more details).

## 3 V-Learning Algorithm

In this section, we introduce V-learning algorithm as a new class of single-agent RL algorithms, which converts any adversarial bandit algorithm with suitable regret guarantees into a RL algorithm. We also present its theoretical guarantees for finding a nearly optimal policy in the single-agent setting.

### 3.1 Training algorithm

To begin with, we describe the V-learning algorithm (Algorithm 1). It maintains a value $V_h(s)$, a counter $N_h(s)$, and a policy $\pi_h(\cdot|s)$ for each state $s$ and step $h$, and initializes them to be the max value, 0, and uniform distribution respectively. V-learning also instantiates $S \times H$ different adversarial bandit algorithms—one for each $(s, h)$ pair. At each step $h$ in each episode $k$, the algorithm performs three major steps:

[5]Technically, to ensure $r_{2,h} \in [0, 1]$, we choose $r_{2,h} = 1 - r_{1,h}$. We note that adding a constant to the reward function has no effect on the equilibria, which is our learning objective.

**Algorithm 1** V-LEARNING

1: **Initialize:** for any $(s, a, h)$, $V_h(s) \leftarrow H + 1 - h$, $N_h(s) \leftarrow 0$, $\pi_h(a|s) \leftarrow 1/A$.
2: **for** episode $k = 1, \ldots, K$ **do**
3: receive $s_1$.
4: **for** step $h = 1, \ldots, H$ **do**
5: take action $a_h \sim \pi_h(\cdot|s_h)$, observe reward $r_h$ and next state $s_{h+1}$.
6: $t = N_h(s_h) \leftarrow N_h(s_h) + 1$.
7: $\tilde{V}_h(s_h) \leftarrow (1 - \alpha_t)\tilde{V}_h(s_h) + \alpha_t(r_h + V_{h+1}(s_{h+1}) + \beta_t)$.
8: $V_h(s_h) \leftarrow \min\{H + 1 - h, \tilde{V}_h(s_h)\}$.
9: $\pi_h(\cdot|s_h) \leftarrow$ ADV_BANDIT_UPDATE$(a_h, \frac{H - r_h - V_{h+1}(s_{h+1})}{H})$ on $(s_h, h)^{\text{th}}$ adversarial bandit.

**Protocol 2** ADVERSARIAL BANDIT ALGORITHM

**Initialize:** for any $b$, $\theta_1(b) \leftarrow 1/B$.
**for** step $t = 1, \ldots, T$ **do**
adversary chooses loss $\ell_t$.
take action $b_t \sim \theta_t$, observe noisy bandit-feedback $\tilde{\ell}_t(b_t)$.
$\theta_{t+1} \leftarrow$ ADV_BANDIT_UPDATE$(b_t, \tilde{\ell}_t(b_t))$.

- Policy execution (Line 5-6): the algorithm takes action $a_h$ according to the maintained $\pi_h$, then observes the reward $r_h$ and the next state $s_{h+1}$, and increases the counter $N_h(s_h)$ by 1.
- $V$-value update (Line 7-8): the algorithm performs incremental update to the value function:

$$\tilde{V}_h(s_h) \leftarrow (1 - \alpha_t)\tilde{V}_h(s_h) + \alpha_t(r_h + V_{h+1}(s_{h+1}) + \beta_t) \tag{2}$$

  here $\alpha_t$ is the learning rate, and $\beta_t$ is the bonus to promote optimism (and exploration). The choices of both quantities will be specified later. Next, we simply update $V_h$ as a truncated version of $\tilde{V}_h$.
- Policy update (Line 9): the algorithm feeds the action $a_h$ and its "loss" $\frac{H - r_h + V_{h+1}(s_{h+1})}{H}$ to the $(s_h, h)^{\text{th}}$ adversarial bandit algorithm, and receives the updated policy $\pi_h(\cdot|s_h)$.

Throughout this paper, we will always use the following learning rate $\alpha_t$. We also define an auxiliary sequence $\{\alpha_t^i\}_{i=1}^t$ based on the learning rate, which will be frequently used across the paper.

$$\alpha_t = \frac{H + 1}{H + t}, \quad \alpha_t^0 = \prod_{j=1}^{t}(1 - \alpha_j), \quad \alpha_t^i = \alpha_i \prod_{j=i+1}^{t}(1 - \alpha_j). \tag{3}$$

We remark that our incremental update (2) bears significant similarity to Q-learning, and our choice of learning rate is precisely the same as the choice in Q-learning [22]. However, a key difference is that the V-learning algorithm maintains V-value functions instead of Q-value functions. This is crucial when extending V-learning to the multiplayer setting where the number of parameters of Q-value functions becomes $\mathcal{O}(HS\prod_{i=1}^m A_i)$ while the number of parameters of V-value functions is only $\mathcal{O}(HS)$. Since V-learning does not use action-value functions, it resorts to adversarial bandit algorithms to update its policy.

**Algorithm 3** EXECUTING OUTPUT POLICY $\hat{\pi}$ OF V-LEARNING

sample $k \leftarrow \text{Uniform}([K])$.
**for** step $h = 1, \ldots, H$ **do**
  observe $s_h$, and set $t \leftarrow N_h^k(s_h)$.
  set $k \leftarrow k_h^i(s_h)$, where $i \in [t]$ is sampled with probability $\alpha_t^i$.
  take action $a_h \sim \pi_h^k(\cdot|s_h)$.

**ADV_BANDIT_UPDATE subroutine:** Consider a multi-arm bandit problem with adversarial loss, where we denote the action set by $\mathcal{B}$ with $|\mathcal{B}| = B$. At round $t$, the learner picks a strategy (distribution over actions) $\theta_t \in \Delta_{\mathcal{B}}$, and the adversary chooses a loss vector $\ell_t \in [0,1]^B$. Then the learner takes an action $b_t$ that is sampled from distribution $\theta_t$, and receives a noisy bandit-feedback $\tilde{\ell}_t(b_t) \in [0,1]$ where $\mathbb{E}[\tilde{\ell}_t(b_t)|\ell_t, b_t] = \ell_t(b_t)$. Then, the adversarial bandit algorithm performs updates based on $b_t$ and $\tilde{\ell}_t(b_t)$, and outputs the strategy for next round $\theta_{t+1}$, which we abstract as $\theta_{t+1} \leftarrow \text{ADV\_BANDIT\_UPDATE}(b_t, \tilde{\ell}_t(b_t))$ (see Protocol 2).

### 3.2 Output policy

We define the final output policy $\hat{\pi}$ of V-learning by how to execute this policy (see Algorithm 3). Let $V^k, N^k, \pi^k$ be the value, counter and policy maintained by V-learning algorithm at *the beginning* of episode $k$. The output policy maintains a scalar $k$, which is initially uniformly sampled from $[K]$. At each step $h$, after observing $s_h$, $\hat{\pi}$ plays a mixture of policy $\{\pi_h^{k^i}(\cdot|s_h)\}_{i=1}^t$ with corresponding probability $\{\alpha_t^i\}_{i=1}^t$ defined in (3). Here $t = N_h^k(s_h)$ is the number of times $s_h$ is visited at step $h$ at the beginning of episode $k$, and $k^i$ is short for $k_h^i(s_h)$ which is the index of the episode when $s_h$ is visited at step $h$ for the $i^{\text{th}}$ time. After that, $\hat{\pi}$ sets $k$ to be the index $k_h^i(s_h)$ whose policy is just played within the mixture, and continue the same process for the next step. This mixture form of output policy $\hat{\pi}$ is mainly due to the incremental updates of V-learning. One can show that, if omitting the optimistic bonus, $V_1^K(s_1)$ computed in the V-learning algorithm is a stochastic estimate of the value of policy $\hat{\pi}$.

We remark that $\hat{\pi}$ is not a Markov policy, but a general random policy (see Definition in Section 2), which can be written as a set of maps $\{\pi_h : \Omega \times \mathcal{S}^h \to \mathcal{A}_i\}$. The choice of action at each step $h$ depends on a joint randomness $\omega \in \Omega$ which is shared among all steps, and the history of past states $(s_1, \ldots, s_h)$. In Section 6, we will further introduce a simple monotone technique that allows V-learning to output a Markov policy in both the single-agent and the two-player zero-sum setting.

### 3.3 Single-agent guarantees

We first state our requirement for the adversarial bandit algorithm used in V-learning, which is to have a high probability *weighted* external regret guarantee as follows. The weights $\{\alpha_t^i\}_{i=1}^t$ are defined in (3).

**Assumption 1.** *For any $t \in \mathbb{N}$ and any $\delta \in (0,1)$, with probability at least $1-\delta$, we have*

$$\max_{\theta \in \Delta_{\mathcal{B}}} \sum_{i=1}^{t} \alpha_t^i [\langle \theta_i, \ell_i \rangle - \langle \theta, \ell_i \rangle] \le \xi(B, t, \log(1/\delta)). \tag{4}$$

*We further assume the existence of an upper bound $\Xi(B, t, \log(1/\delta)) \ge \sum_{t'=1}^{t} \xi(B, t', \log(1/\delta))$ where (i) $\xi(B, t, \log(1/\delta))$ is non-decreasing in $B$ for any $t, \delta$; (ii) $\Xi(B, t, \log(1/\delta))$ is concave in $t$ for any $B$, $\delta$.*

Assumption 1 can be satisfied by modifying many existing algorithms with unweighted external regret to the weighted setting. In particular, we prove that the Follow-the-Regularized-Leader (FTRL) algorithm (Algorithm 5) satisfies the Assumption 1 with bounds $\xi(B,t,\log(1/\delta)) \le \mathcal{O}(\sqrt{HB\log(B/\delta)/t})$ and $\Xi(B,t,\log(1/\delta)) \le \mathcal{O}(\sqrt{HBt\log(B/\delta)})$. The $H$ factor comes into the bounds because our choice of weights $\{\alpha_t^i\}$ in (3) involves $H$. We refer readers to Appendix F for more details.

We are now ready to introduce the theoretical guarantees of V-learning for finding near-optimal policies in the single-agent setting.

**Theorem 4.** *Suppose subroutine* ADV_BANDIT_UPDATE *satisfies Assumption 1. For any $\delta \in (0,1)$ and $K \in \mathbb{N}$, let $\iota = \log(HSAK/\delta)$. Choose learning rate $\alpha_t$ according to* (3) *and bonus $\{\beta_t\}_{t=1}^K$ so that $\sum_{i=1}^t \alpha_t^i \beta_i = \Theta(H\xi(A,t,\iota) + \sqrt{H^3\iota/t})$ for any $t \in [K]$. Then, with probability at least $1-\delta$, after running Algorithm 1 for $K$ episodes, we have the output policy $\hat{\pi}$ by Algorithm 3 satisfies*

$$V_1^\star(s_1) - V_1^{\hat{\pi}}(s_1) \le \mathcal{O}((H^2S/K)\cdot\Xi(A,K/S,\iota) + \sqrt{H^5S\iota/K}).$$

*In particular, when instantiating subroutine* ADV_BANDIT_UPDATE *by FTRL (Algorithm 5), we can choose $\beta_t = c\cdot\sqrt{H^3A\iota/t}$ for some absolute constant c, where $V_1^\star(s_1) - V_1^{\hat{\pi}}(s_1) \le \mathcal{O}(\sqrt{H^5SA\iota/K})$.*

Theorem 4 characterizes how fast the suboptimality of $\hat{\pi}$ decreases with respect to the total number of episode $K$. In particular, to obtain an $\epsilon$-optimal output policy $\hat{\pi}$, we only need to use a number of episodes $K = \tilde{\mathcal{O}}(H^5SA/\epsilon^2)$. This is $H^2$ factor larger than the information-theoretic lower bound $\Omega(H^3SA/\epsilon^2)$ in this setting [22]. We remark that one extra $H$ factor is due to the incremental update and the use of learning rate in (3) which is exactly the same for Q-learning algorithm [22]. The other $H$ factor can be potentially improved by using refined first-order regret bound (counterparts of Bernstein concentration) in V-learning.

While V-learning seems to be no better than classical value iteration or Q-learning in the single-agent setting, its true power starts to show up in the multiagent setting: Value iteration and Q-learning require highly nontrivial efforts to adapt them to the multiagent setting, and by design they suffer from the curse of multiagents [3, 30]. In the following sections, we will show that V-learning can be directly extended to the multiagent setting by simply letting all agents run V-learning independently. Furthermore, V-learning breaks the curse of multiagents.

# 4 Two-player Zero-sum Markov Games

In this section, we provide the sample efficiency guarantee for V-learning to find Nash equilibria in two-player zero-sum Markov games.

## 4.1 Finding Nash equilibria

In the two-player zero-sum setting, we have two agents whose rewards satisfy $r_{1,h} = -r_{2,h}$ for any $h \in [H]$. Our algorithm is simply that both agents run V-learning (Algorithm 1) independently with learning rate $\alpha_t$ as specified in (3). Each player $j$ will uses her own set of bonus $\{\beta_{j,t}\}$ that depends on the number of her actions and will be specified later. To execute the output policy, both agents simply execute Algorithm 3 independently using their own intermediate policies computed by V-learning.

We have the following theorem for V-learning. For clean presentation, we denote $A = \max_{j\in[2]} A_j$.

**Theorem 5.** *Suppose subroutine* ADV_BANDIT_UPDATE *satisfies Assumption 1. For any $\delta \in (0,1)$ and $K \in \mathbb{N}$, let $\iota = \log(HSAK/\delta)$. Choose learning rate $\alpha_t$ according to* (3) *and bonus $\{\beta_{j,t}\}_{t=1}^K$ of the $j^{th}$*

*player so that* $\sum_{i=1}^{t}\alpha_t^i\beta_{j,i} = \Theta(H\xi(A_j,t,\iota)+\sqrt{H^3\iota/t})$ *for any* $t\in[K]$. *Then, with probability at least* $1-\delta$, *after running Algorithm 1 for* $K$ *episodes, let* $\hat{\pi}_1,\hat{\pi}_2$ *be the output policies by Algorithm 3 for each player, then we have the product policy* $\hat{\pi}=\hat{\pi}_1\times\hat{\pi}_2$ *satisfies*

$$\max_{j\in[2]}[V_{j,1}^{\dagger,\hat{\pi}_{-j}}(s_1)-V_{j,1}^{\hat{\pi}}(s_1)]\le\mathcal{O}((H^2S/K)\cdot\Xi(A,K/S,\iota)+\sqrt{H^5S\iota/K}).$$

*When instantiating* ADV_BANDIT_UPDATE *by FTRL (Algorithm 5), we can choose* $\beta_{j,t}=c\cdot\sqrt{H^3A_j\iota/t}$ *for some absolute constant c, which leads to* $\max_{j\in[2]}[V_{j,1}^{\dagger,\hat{\pi}_{-j}}(s_1)-V_{j,1}^{\hat{\pi}}(s_1)]\le\mathcal{O}(\sqrt{H^5SA\iota/K})$.

Theorem 5 claims that, to find an $\epsilon-$approximate Nash equilibrium, we only need to use a number of episodes $K=\tilde{\mathcal{O}}(H^5SA/\epsilon^2)$, where $A=\max_{j\in[2]}A_j$. In contrast, value iteration or Q-learning based algorithms require at least $\Omega(H^3SA_1A_2/\epsilon^2)$ episodes to find Nash equilibria [3, 30]. Furthermore, V-learning is a fully decentralized algorithm. To our best knowledge, V-learning is the only algorithm up to today that achieves sample complexity linear in $A$ for finding Nash equilibrium in two-player zero-sum Markov games.

We remark that V-learning only performs $\mathcal{O}(1)$ operations and calls subroutine ADV_BANDIT_UPDATE once every time a new sample is observed. As long as the adversarial bandit algorithm used in V-learning is computationally efficient (which is the case for FTRL), V-learning itself is also computationally efficient.

# 5 Multiplayer General-sum Markov Games

In multiplayer general-sum games, finding Nash equilibria is computationally hard in general (which is technically PPAD-complete [12]). In this section, we focus on finding two commonly-used alternative notions of equilibria in the game theory—coarse correlated equilibria (CCE), and correlated equilibria (CE). Both are relaxed notions of Nash equilibira.

## 5.1 Finding coarse correlated equilibria

The algorithm for finding CCE is again running V-learning (Algorithm 1) independently for each agent $j$ with learning rate $\alpha_t$ (as specified in (3)) and bonus $\{\beta_{j,t}\}$ (to be specified later). The major difference from the case of finding Nash equilibria is that CCE and CE require the output policy to be joint correlated policy. We achieve this correlation by feeding the same random seed to all agents at the very beginning when they execute the output policy according to Algorithm 3. That is, while training can be done in the fully decentralized fashion, we require one round of communication at the beginning of the execution to broadcast the shared random seed. After that, each agent can simply execute her own output policy independently. During the execution, since the states visited are shared among all agents, shared random seed allows the same index $i$ to be sampled across all agents in the Step 4 of Algorithm 3 at every step. We denote this correlated joint output policy as $\hat{\pi}=\hat{\pi}_1\odot\ldots\odot\hat{\pi}_m$.

We remark that to specify a correlated policy in general, we need to specify the probability for taking all action combinations $(a_1,\ldots,a_m)$ for each $(s,h)$. This requires at least $\Omega(HS\prod_{j=1}^m A_j)$ space, which grows exponentially with the number of agents $m$. The way V-learning specifies the joint policy only requires agents to store their own intermediate counters and policies computed during training. This only takes a total of $\mathcal{O}(HSK(\sum_{j=1}^m A_j))$ space, which scales only linearly with the number of agents. Our approach dramatically improve over the former approach in space complexity when the number of agents is large.

We now present the guarantees for V-learning to learn a CCE as follows. Let $A=\max_{j\in[m]}A_j$.

**Theorem 6.** *Suppose subroutine* ADV_BANDIT_UPDATE *satisfies Assumption 1. For any $\delta \in (0,1)$ and $K \in \mathbb{N}$, let $\iota = \log(mHSAK/\delta)$. Choose learning rate $\alpha_t$ according to* (3) *and bonus $\{\beta_{j,t}\}_{t=1}^K$ of the $j^{th}$ player so that $\sum_{i=1}^t \alpha_t^i \beta_{j,i} = \Theta(H\xi(A_j, t, \iota) + \sqrt{H^3\iota/t})$ for any $t \in [K]$. Then, with probability at least $1-\delta$, after all the players running Algorithm 1 for $K$ episodes, let $\hat{\pi}_j$ be the output policy by Algorithm 3 for the $j^{th}$ player, then we have the joint policy $\hat{\pi} = \hat{\pi}_1 \odot \ldots \odot \hat{\pi}_m$ satisfies*

$$\max_{j\in[m]}[V_{j,1}^{\dagger,\hat{\pi}_{-j}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \le \mathcal{O}((H^2S/K)\cdot \Xi(A, K/S, \iota) + \sqrt{H^5S\iota/K}).$$

*When instantiating* ADV_BANDIT_UPDATE *by FTRL (Algorithm 5), we can choose $\beta_{j,t} = c\cdot\sqrt{H^3A_j\iota/t}$ for some absolute constant c, which leads to $\max_{j\in[m]}[V_{j,1}^{\dagger,\hat{\pi}_{-j}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \le \mathcal{O}(\sqrt{H^5SA\iota/K})$.*

Theorem 6 claims that, to find an $\epsilon-$approximate CCE, V-learning only needs to use a number of episodes $K = \tilde{\mathcal{O}}(H^5SA/\epsilon^2)$, where $A = \max_{j\in[m]} A_j$. This is in sharp contrast to the prior results for multiplayer general-sum Markov games, which use value iteration based algorithms, and require at least $\Omega(H^4S^2(\prod_{i=1}^m A_i)/\epsilon^2)$ episodes [30]. As a result, V-learning is the first algorithm that breaks the curse of multiagents for finding CCE in Markov games.

## 5.2 Finding correlated equilibria

The algorithm for finding CE is almost the same as the algorithm for finding CCE except that we now require a different ADV_BANDIT_UPDATE subroutine, which has the following high probability weighted *swap regret* guarantee.

**Assumption 2.** *For any $t \in \mathbb{N}$ and any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\max_{\psi\in\Psi}\sum_{i=1}^t \alpha_t^i[\langle\theta_i, l_i\rangle - \langle\psi\diamond\theta_i, l_i\rangle] \le \xi_{sw}(B, t, \log(1/\delta)). \tag{5}$$

*We assume the existence of an upper bound $\Xi_{sw}(B,t,\log(1/\delta)) \ge \sum_{t'=1}^t \xi_{sw}(B,t',\log(1/\delta))$where (i) $\xi_{sw}(B,t,\log(1/\delta))$ is non-decreasing in $B$ for any $t,\delta$; (ii) $\Xi_{sw}(B,t,\log(1/\delta))$ is concave in $t$ for any $B$, $\delta$.*

Here $\Psi$ denotes the set $\{\psi : \mathcal{B}\to\mathcal{B}\}$ which consists of all maps from actions in $\mathcal{B}$ to actions in $\mathcal{B}$. Meanwhile, for any $\theta \in \Delta_{\mathcal{B}}$, the term $\psi\diamond\theta \in \Delta_{\mathcal{B}}$ denotes the distribution over actions where $\psi\diamond\theta(b) = \sum_{b':\psi(b')=b}\theta(b')$. We note that bounded swap regret is a stronger requirement compared to bounded external regret as in (4), since by maximizing over a subset of functions in $\Psi$ which map all actions in $\mathcal{B}$ to one single action, we recover the external regret by (5).

Assumption 2 can be satisfied by modifying many existing algorithms with external regret to the swap regret setting. In particular, we prove that the Follow-the-Regularized-Leader for swap regret (FTRL_swap) algorithm (Algorithm 6) satisfies Assumption 2 with bounds $\xi_{\text{sw}}(B,t,\log(1/\delta)) \le \mathcal{O}(B\sqrt{H\log(B/\delta)/t})$ and $\Xi_{\text{sw}}(B,t,\log(1/\delta)) \le \mathcal{O}(B\sqrt{Ht\log(B/\delta)})$. Both bounds have one extra $\sqrt{B}$ factor comparing to the counterparts in external regret. We refer readers to Appendix G for more details.

We now present the guarantees for V-learning to learn a CCE as follows. Let $A = \max_{j\in[m]} A_j$.

**Theorem 7.** *Suppose subroutine* ADV_BANDIT_UPDATE *satisfies Assumption 2. For any $\delta \in (0,1)$ and $K \in \mathbb{N}$, let $\iota = \log(mHSAK/\delta)$. Choose learning rate $\alpha_t$ according to* (3) *and bonus $\{\beta_{j,t}\}_{t=1}^K$ of the $j^{th}$ player so that $\sum_{i=1}^t \alpha_t^i\beta_{j,i} = \Theta(H\xi_{sw}(A_j,t,\iota) + \sqrt{H^3\iota/t})$ for any $t\in[K]$. Then, with probability at least*

$1-\delta$, *after all the players running Algorithm 1 for* $K$ *episodes, let* $\hat{\pi}_j$ *be the output policy by algorithm 3 for the* $j^{th}$ *player, then we have the joint policy* $\hat{\pi} = \hat{\pi}_1 \odot \ldots \odot \hat{\pi}_m$ *satisfies*

$$\max_{j\in[m]} \max_{\phi_j} [V_{j,1}^{\phi_j \diamond \hat{\pi}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \le \mathcal{O}((H^2 S/K) \cdot \Xi_{sw}(A, K/S, \iota) + \sqrt{SH^5\iota/K}).$$

*When instantiating* ADV_BANDIT_UPDATE *by FTRL_swap (Algorithm 6), we can choose* $\beta_{j,t} = c \cdot A_j \sqrt{H^3\iota/t}$ *for some absolute constant c, which leads to* $\max_{j\in[m]} \max_{\phi_j} [V_{j,1}^{\phi_j \diamond \hat{\pi}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \le \mathcal{O}(A\sqrt{H^5 S\iota/K})$.

Theorem 7 claims that, to find an $\epsilon-$approximate CE, V-learning only needs to use a number of episodes $K = \tilde{\mathcal{O}}(H^5 S A^2/\epsilon^2)$, where $A = \max_{j\in[m]} A_j$. It has an extra $A$ multiplicative factor comparing to the sample complexity of finding CCE, since CE is a subset of CCE thus finding CE is expected to be more difficult. Nevertheless, the sample complexity presented here is far better than value iteration based algorithm, which requires at least $\Omega(H^4 S^2(\prod_{i=1}^m A_i)/\epsilon^2)$ episodes for finding CE [30]. V-learning is also the first algorithm that breaks the curse of multiagents for finding CE in Markov games.

# 6 Monotonic V-Learning

In the previous sections, we present the V-learning algorithm whose output policy (Algorithm 3) is a nested mixture of Markov polices. Storing such a output policy requires $\mathcal{O}(HSA_jK)$ space for the $j^{\text{th}}$ player. In Section 5, we argue this approach has a significant advantage over directly storing a general correlated policy when the number of agents is large. Nevertheless, this space complexity can be undesirable when the number of agents is small.

In this section, we introduce a simple monotonic techique to V-learning, which allows each agent to output a Markov policy when finding Nash equilibria in the two-player zero-sum setting. Storing a Markov policy only takes $\mathcal{O}(HSA_j)$ space for the $j^{\text{th}}$ player. A similar result for the single-agent setting can be immediately obtained by setting the second player in the Markov game to be a dummy player with only a single action to choose from.

**Monotonic update** Monotonic V-learning is almost the same as V-learning with only the Line 8 in Algorithm 1 changed to

$$V_h(s_h) \leftarrow \min\{H+1-h, \tilde{V}_h(s_h), V_h(s_h)\}, \tag{6}$$

This step guarantees $V_h(s_h)$ to monotonically decrease at each step. This is helpful because in two-player zero-sum Markov games, all Nash equilibria share a unique value which we denote as $V^\star$. By design, we can prove that the V-values maintained in V-learning are high probability upper bounds of $V^\star$ (Lemma 18). This monotonic update allows our V-value estimates to always get closer to $V^\star$ after each update, which improves the accuracy of our V-value estimates.

**Markov output policy** For an arbitrary fixed $(s,h) \in \mathcal{S} \times [H]$, let $t_1$ be the last episode when the value $V_{1,h}(s)$ is updated (i.e., strictly decreases), and let $t_2$ be the last episode when the value $V_{2,h}(s)$ is updated. Then the output policy for this $(s,h)$ has the following form.

$$\tilde{\pi}_{1,h}(\cdot|s) := \sum_{i=1}^{t_2} \alpha_{t_2}^i \pi_{1,h}^{k^i}(\cdot|s), \quad \tilde{\pi}_{2,h}(\cdot|s) := \sum_{i=1}^{t_1} \alpha_{t_1}^i \pi_{2,h}^{k^i}(\cdot|s), \tag{7}$$

where $k^i$ denotes the index of episode when state $s$ is visited at step $h$ is visited for the $i^{\text{th}}$ time. Recall that $\pi^k_{j,h}(\cdot|s)$ is the policy maintained by the $j^{\text{th}}$ player at the beginning of the $k^{\text{th}}$ episode when she runs V-learning. That is, the new output policy is simply the weighted average of policies computed in the V-learning at each $(s,h)$ pair. Clearly, the policies $\tilde{\pi}_1$ and $\tilde{\pi}_2$ defined by (7) are Markov policies.

We remark that although the execution of $\tilde{\pi}_1$ and $\tilde{\pi}_2$ can be fully decentralized, in (7) the computation of $\tilde{\pi}_{1,h}(\cdot|s)$ depends on $t_2$ while the computation of $\tilde{\pi}_{2,h}(\cdot|s)$ depends on $t_1$. That is, two players need to communicate at the end the indexes of the most recent episodes when their $V$-values are updated. As a result, monotonic V-learning is not fully decentralized.

**Theorem 8.** *Monotonic V-learning with output policy $\tilde{\pi} = \tilde{\pi}_1 \times \tilde{\pi}_2$ as specified by* (7) *has the same theoretical guarantees as Theorem 5 with the same choices of hyperparamters.*

Theorem 8 asserts that V-learning can be modified to output Markov policies when finding Nash equilibria of two-player zero-sum Markov games. As a special case, the same technique and results directly apply to the single-agent setting.

# 7 Conclusion

In this paper, we develop the first *decentralized* algorithm that breaks the *curse of multiagents* for learning general Markov games. Behind this new result is a new class of single-agent RL algorithms—V-learning, which converts any adversarial bandit algorithm with suitable regret guarantees into a RL algorithm. A remarkable advantage of V-learning is its effortless extension to the multiagent setting while having much preferred theoretical guarantees over existing methods: by simply running V-learning independently for all agents, we find Nash equilibria (for two-player zero-sum games), CCE, and CE in a number of samples that scales with only $\max_{j\in[m]} A_j$, in contrast to existing algorithms whose number of samples scales with $\prod_{j\in[m]} A_j$.

# References


[1] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 263–272. JMLR. org, 2017.

[2] Yu Bai and Chi Jin. Provable self-play algorithms for competitive reinforcement learning. *arXiv preprint arXiv:2002.04017*, 2020.

[3] Yu Bai, Chi Jin, and Tiancheng Yu. Near-optimal reinforcement learning with self-play. In *Advances in Neural Information Processing Systems*, 2020.

[4] Bowen Baker, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch. Emergent tool use from multi-agent autocurricula. In *International Conference on Learning Representations*, 2020.

[5] Avrim Blum and Yishay Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8(6), 2007.

[6] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[7] Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

[8] Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, 365(6456): 885–890, 2019.

[9] Andrea Celli, Alberto Marchesi, Gabriele Farina, and Nicola Gatti. No-regret learning dynamics for extensive-form correlated equilibrium. *Advances in Neural Information Processing Systems*, 33, 2020.

[10] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[11] Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying pac and regret: Uniform pac bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 5713–5723, 2017.

[12] Constantinos Daskalakis. On the complexity of approximating a nash equilibrium. *ACM Transactions on Algorithms (TALG)*, 9(3):23, 2013.

[13] Constantinos Daskalakis, Dylan J Foster, and Noah Golowich. Independent policy gradient methods for competitive reinforcement learning. *arXiv preprint arXiv:2101.04233*, 2021.

[14] Andrew Gilpin and Tuomas Sandholm. Finding equilibria in large sequential games of imperfect information. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 160–169, 2006.

[15] Thomas Dueholm Hansen, Peter Bro Miltersen, and Uri Zwick. Strategy iteration is strongly polynomial for 2-player turn-based stochastic games with a constant discount factor. *Journal of the ACM (JACM)*, 60(1):1–16, 2013.

[16] Sergiu Hart and Andreu Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150, 2000.

[17] Junling Hu and Michael P Wellman. Nash q-learning for general-sum stochastic games. *Journal of machine learning research*, 4(Nov):1039–1069, 2003.

[18] Shariq Iqbal and Fei Sha. Actor-attention-critic for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 2961–2970. PMLR, 2019.

[19] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

[20] Zeyu Jia, Lin F Yang, and Mengdi Wang. Feature-based q-learning for two-player stochastic games. *arXiv preprint arXiv:1906.00423*, 2019.

[21] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017.

[22] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is Q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pages 4868–4878, 2018.

[23] 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, 2020.

[24] Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in Neural Information Processing Systems*, 2021.

[25] Chi Jin, Qinghua Liu, and Tiancheng Yu. The power of exploiter: Provable multi-agent rl in large state spaces. *arXiv preprint arXiv:2106.03352*, 2021.

[26] Daphne Koller and Nimrod Megiddo. The complexity of two-person zero-sum games in extensive form. *Games and economic behavior*, 4(4):528–552, 1992.

[27] Tor Lattimore and Csaba Szepesvári. Bandit algorithms. 2018.

[28] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier, 1994.

[29] Michael L Littman. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pages 322–328, 2001.

[30] Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model-based reinforcement learning with self-play. In *International Conference on Machine Learning*, pages 7001–7010. PMLR, 2021.

[31] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017.

[32] Weichao Mao and Tamer Başar. Provably efficient reinforcement learning in decentralized general-sum markov games. *arXiv preprint arXiv:2110.05682*, 2021.

[33] Gergely Neu. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 3168–3176, 2015.

[34] OpenAI. Openai five. `https://blog.openai.com/openai-five/`, 2018.

[35] Ian Osband and Benjamin Van Roy. On lower bounds for regret in reinforcement learning. *arXiv preprint arXiv:1608.02732*, 2016.

[36] Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. *arXiv preprint arXiv:1402.0635*, 2014.

[37] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[38] Goran Radanovic, Rati Devidze, David Parkes, and Adish Singla. Learning to collaborate in markov decision processes. In *International Conference on Machine Learning*, pages 5261–5270, 2019.

[39] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304. PMLR, 2018.

[40] Muhammed O Sayin, Kaiqing Zhang, David S Leslie, Tamer Basar, and Asuman Ozdaglar. Decentralized q-learning in zero-sum markov games. *arXiv preprint arXiv:2106.02748*, 2021.

[41] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

[42] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

[43] Aaron Sidford, Mengdi Wang, Lin Yang, and Yinyu Ye. Solving discounted stochastic two-player games with near-optimal time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pages 2992–3002. PMLR, 2020.

[44] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

[45] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

[46] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 5887–5896. PMLR, 2019.

[47] Ziang Song, Song Mei, and Yu Bai. When can we learn general-sum markov games with a large number of players sample-efficiently? *arXiv preprint arXiv:2110.04184*, 2021.

[48] Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. PAC model-free reinforcement learning. In *International Conference on Machine Learning*, pages 881–888, 2006.

[49] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*, 2017.

[50] Yi Tian, Yuanhao Wang, Tiancheng Yu, and Suvrit Sra. Online learning in unknown markov games. In *International Conference on Machine Learning*, pages 10279–10288. PMLR, 2021.

[51] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

[52] Chen-Yu Wei, Yi-Te Hong, and Chi-Jen Lu. Online reinforcement learning in stochastic games. In *Advances in Neural Information Processing Systems*, pages 4987–4997, 2017.

[53] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. Linear last-iterate convergence in constrained saddle-point optimization. *arXiv e-prints*, pages arXiv–2006, 2020.

[54] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. Last-iterate convergence of decentralized optimistic gradient descent/ascent in infinite-horizon competitive markov games. *arXiv preprint arXiv:2102.04540*, 2021.

[55] Qiaomin Xie, Yudong Chen, Zhaoran Wang, and Zhuoran Yang. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. *arXiv preprint arXiv:2002.07066*, 2020.

[56] Andrea Zanette, Alessandro Lazaric, Mykel Kochenderfer, and Emma Brunskill. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pages 10978–10989. PMLR, 2020.

[57] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pages 5872–5881. PMLR, 2018.

[58] Kaiqing Zhang, Sham M Kakade, Tamer Başar, and Lin F Yang. Model-based multi-agent rl in zero-sum markov games with near-optimal sample complexity. *arXiv preprint arXiv:2007.07461*, 2020.

[59] Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. *Advances in neural information processing systems*, 20:1729–1736, 2007.

**Algorithm 4** EXECUTING POLICY $\hat{\pi}_h^k$

1: **for** step $h' = h, h+1, \ldots, H$ **do**
2: observe $s_{h'}$, and set $t \leftarrow N_{h'}^k(s_{h'})$.
3: set $k \leftarrow k_{h'}^i(s_{h'})$, where $i \in [t]$ is sampled with probability $\alpha_t^i$.
4: take action $a_{h'} \sim \pi_{h'}^k(\cdot|s_{h'})$.

# A Notations and Basic Lemmas

## A.1 Notations

In this subsection, we introduce some notations that will be frequently used in appendixes. Recall that we use $V^k, N^k, \pi^k$ to denote the value, counter and policy maintained by V-learning algorithm at *the beginning* of the episode $k$.

We also introduce a new policy $\hat{\pi}_h^k$ for a single agent (defined by its execution in Algorithm 4), which can be viewed as a part of the output policy in Algorithm 3. The definition of $\hat{\pi}_h^k$ is very similar to $\hat{\pi}$ except two differences: (1) $\hat{\pi}_h^k$ is a policy for step $h, \ldots, H$ while $\hat{\pi}$ is a policy for step $1, \ldots, H$; (2) in $\hat{\pi}$ the initial value of $k$ is sampled uniformly at random from $[K]$ at the very beginning while in $\hat{\pi}_h^k$ the initial value of $k$ is given.

We remark that $\hat{\pi}_h^k$ is a non-Markov policy that does not depends on history before to the $h^{\text{th}}$ step. In symbol, we can express this class of policy as $\pi_j := \{\pi_{j,h'} : \Omega \times (\mathcal{S} \times \mathcal{A})^{h'-h} \times \mathcal{S} \to \mathcal{A}_j\}_{h'=h}^H$. We call this class of policy the *policy starting from the $h^{th}$ step*, and denote it as $\Pi_h$. Similar to Section 2, we can also define joint policy $\pi = \pi_1 \odot \ldots \odot \pi_m$ and product policy $\pi = \pi_1 \times \ldots \times \pi_m$ for policies in $\Pi_h$. We can also define value $V_h^\pi(s)$ for joint policy $\pi \in \Pi_h$ as

$$V_{i,h}^\pi(s) := \mathbb{E}_\pi \left[ \sum_{h'=h}^H r_{i,h'}(s_{h'}, \boldsymbol{a}_{h'}) \middle| s_h = s \right].$$

This allows us to define the corresponding best response of $\pi_{-i}$ as the maximizer of $\max_{\pi_i' \in \Pi_h} V_{i,h}^{\pi_i' \times \pi_{-i}}(s)$. We also denote this maximum value as $V_{i,h}^{\dagger,\pi_{-i}}(s)$. We define the strategy modification for policies starting from the $h^{\text{th}}$ step as $\phi_i := \{\phi_{i,h'} : (\mathcal{S} \times \mathcal{A})^{h'-h} \times \mathcal{S} \times \mathcal{A}_i \to \mathcal{A}_i\}_{h'=h}^H$, and denote the set of such strategy modification as $\Phi_h$.

Finally, for simplicity of notation, we define two operators $\mathbb{P}$ and $\mathbb{D}$ as follows:

$$\begin{cases} \mathbb{P}_h[V](s, \boldsymbol{a}) = \mathbb{E}_{s' \sim \mathbb{P}_h(\cdot|s,\boldsymbol{a})}[V(s')], \\ \mathbb{D}_\pi[Q](s) = \mathbb{E}_{\boldsymbol{a} \sim \pi(\cdot|s)}[Q(s, \boldsymbol{a})], \end{cases} \tag{8}$$

for any value function $V$, $Q$ and any one-step Markov policy $\pi$.

## A.2 Basic lemmas

We first present a proposition which clarify the relations among the three different kind of equilibria. In particular, we show that, similar to strategic games, we also have Nash $\subset$ CE $\subset$ CCE in Markov games.

**Proposition 9** (Nash $\subset$ CE $\subset$ CCE)**.** *In Markov games, any $\epsilon$-approximate Nash equilibrium is an $\epsilon$-approximate CE, and any $\epsilon$-approximate CE is an $\epsilon$-approximate CCE.*

*Proof.* We prove two claims separately.

For Nash $\subset$ CE, let $\pi = \pi_1 \times \pi_2 \times \cdots \pi_m$ be an $\epsilon$-approximate Nash equilibrium, then

$$\max_{\phi_i} V_{i,1}^{(\phi_i \diamond \pi_i) \times \pi_{-i}}(s_1) \overset{(a)}{=} \max_{\pi_i'} V_{i,1}^{\pi_i' \times \pi_{-i}}(s_1) \overset{(b)}{\leq} V_{i,1}^{\pi}(s_1) + \epsilon,$$

Step (a) is because that $\pi$ is a product policy, where the randomness of different agents are completely independent. In this case, maximizing over strategy modification $\phi_i$ is equivalent to maximizing over a new independent policy. Step (b) directly follows from $\pi$ being an $\epsilon$-approximate Nash equilibrium. By definition, this proves that $\pi$ is also an $\epsilon$-approximate CE.

For CE $\subset$ CCE, let $\pi = \pi_1 \odot \pi_2 \odot \cdots \pi_m$ be an $\epsilon$-approximate CE, then we have

$$\max_{\pi_i'} V_{i,1}^{\pi_i' \times \pi_{-i}}(s_1) \overset{(c)}{\leq} \max_{\phi_i} V_{i,1}^{(\phi_i \diamond \pi_i) \odot \pi_{-i}}(s_1) \overset{(d)}{\leq} V_{i,1}^{\pi}(s_1) + \epsilon,$$

Step (c) is because by definition of strategy modification $\phi_i := \{\phi_{i,h} : (\mathcal{S} \times \mathcal{A})^{h-1} \times \mathcal{S} \times \mathcal{A}_i \to \mathcal{A}_i\}$, we can consider a subset of strategy modification $\phi_i' := \{\phi_{i,h}' : (\mathcal{S} \times \mathcal{A})^{h-1} \times \mathcal{S} \to \mathcal{A}_i\}$ which modifies the policy ignoring whatever the action $\pi_i$ takes. It is not hard to see that maxmizing over the strategy modification in this subset is equivalent to maximizing over a new independent policy $\pi_i'$. Therefore, maximizing over all strategy modification is greater or equal to maximizing over $\pi_i'$. Finally, step (d) follows from $\pi$ being an $\epsilon$-approximate CE. By definition, this proves that $\pi$ is also an $\epsilon$-approximate CCE. □

Next, we present some basic lemmas that will be used in the proofs of different theorems. We start by introducing some useful properties of sequence $\{\alpha_t^i\}$ defined in (3).

**Lemma 10.** *([22, Lemma 4.1],[50, Lemma 2]) The following properties hold for $\alpha_t^i$:*

1. $\frac{1}{\sqrt{t}} \leq \sum_{i=1}^t \frac{\alpha_t^i}{\sqrt{i}} \leq \frac{2}{\sqrt{t}}$ *and* $\frac{1}{t} \leq \sum_{i=1}^t \frac{\alpha_t^i}{i} \leq \frac{2}{t}$ *for every* $t \geq 1$.
2. $\max_{i \in [t]} \alpha_t^i \leq \frac{2H}{t}$ *and* $\sum_{i=1}^t (\alpha_t^i)^2 \leq \frac{2H}{t}$ *for every* $t \geq 1$.
3. $\sum_{t=i}^\infty \alpha_t^i = 1 + \frac{1}{H}$ *for every* $i \geq 1$.

Finally, we have the following lemma which express the $\tilde{V}$ maintained in V-learning in the form of weighted sum of earlier updates.

**Lemma 11.** *Consider an arbitrary fixed $(s, h, k)$ tuple. Let $t = N_h^k(s)$ denote the number of times $s$ is visited at step $h$ at the beginning of episode $k$, and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step. Then the two V-values $\tilde{V}$ and $V$ in Algorithm 1 satisfy the following equation:*

$$\tilde{V}_{j,h}^k(s) = \alpha_t^0(H - h + 1) + \sum_{i=1}^t \alpha_t^i \left[ r_{j,h}(s, \boldsymbol{a}_h^{k^i}) + V_{j,h+1}^{k^i}(s_{h+1}^{k^i}) + \beta_{j,i} \right], \quad j \in [m]. \tag{9}$$

*Proof.* The proof follows directly from the update rule in Line 7 Algorithm 1. Note that $\alpha_t^0$ is equal to zero for any $t > 1$ and equal to one for $t = 0$. □

# B Proofs for Computing CCE in General-sum MGs

In this section, we give complete proof of Theorem 6. To avoid repeatedly state the condition of Theorem 6 in each lemma, we will use

- Condition of the adversarial bandit sub-procudure (Assumption 1) and
- Set the bonus $\{\beta_{j,t}\}_{t=1}^K$ of the $j^{\text{th}}$ player so that $\sum_{i=1}^t \alpha_t^i \beta_{j,i} = \Theta(H\xi(A_j, t, \iota) + \sqrt{H^3\iota/t})$ for any $t \in [K]$.

throughout the whole section.

The following Lemma is a direct consequence of Assumption 1, which will play an important role in our later analysis.

**Lemma 12.** *Under Assumption 1, the following event is true with probability at least $1-\delta$: for any $(s,h,k) \in \mathcal{S}\times[H]\times[K]$, let $t = N_h^k(s)$ and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step, then for all $j \in [m]$*

$$\max_{\mu} \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\mu\times\pi_{-j,h}^{k^i}} \left(r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}\right)(s) - \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left(r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}\right)(s) \le H\xi(A_j, t, \iota),$$

*where $\iota = \log(mHSAK/\delta)$.*

*Proof.* By Assumption 1 and the adversarial bandit update step in Algorithm 1, we have that with probability at least $1-\delta$, for any $(s,h,k) \in \mathcal{S}\times[H]\times[K]$,

$$\max_{\mu} \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left(\frac{H - r_{j,h} - \mathbb{P}_h V_{j,h+1}^{k^i}}{H}\right)(s) - \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\mu\times\pi_{-j,h}^{k^i}} \left(\frac{H - r_{j,h} - \mathbb{P}_h V_{j,h+1}^{k^i}}{H}\right)(s) \le \xi(A_j, t, \iota),$$

which implies the desired result by simple algebraic transformation. □

Then we show $V$ is actually an optimistic estimation of the value function of player $j$'th best response to the output policy.

**Lemma 13** (Optimism)**.** *For any $\delta \in (0,1]$, with probability at least $1-\delta$, for any $(s,h,k,j) \in \mathcal{S}\times[H]\times[K]\times[m]$, $V_{j,h}^k(s) \ge V_{j,h}^{\dagger,\hat{\pi}_{-j,h}^k}(s)$.*

*Proof of Lemma 13.* We prove by backward induction. The claim is trivially satisfied for $h = H+1$. Suppose it is true for $h+1$, consider a fixed state $s$. It suffices to show $\tilde{V}_{j,h}^k(s) \ge V_{j,h}^{\dagger,\hat{\pi}_{-j,h}^k}(s)$ because $V_{j,h}^k(s) = \min\{\tilde{V}_{j,h}^k(s), H-h+1\}$. Let $t = N_h^k(s)$ and suppose $s$ was previously visited at episodes

$k^1, \ldots, k^t < k$ at the $h$-th step. Then using Lemma 11,

$$\begin{aligned}
\tilde{V}^k_{j,h}(s) &= \alpha^0_t H + \sum_{i=1}^t \alpha^i_t \left[ r_{j,h}(s, \boldsymbol{a}^{k^i}_h) + V^{k^i}_{j,h+1}(s^{k^i}_{h+1}) + \beta_{j,i} \right] \\
&\overset{(i)}{\geq} \sum_{i=1}^t \alpha^i_t \mathbb{D}_{\pi^{k^i}_h} \left( r_{j,h} + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) + \sum_{i=1}^t \alpha^i_t \beta_{j,i} - \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) \\
&\overset{(ii)}{\geq} \max_{\mu} \sum_{i=1}^t \alpha^i_t \mathbb{D}_{\mu \times \pi^{k^i}_{-j,h}} \left( r_{j,h} + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) + \sum_{i=1}^t \alpha^i_t \beta_{j,i} - \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) - H\xi(A_j, t, \iota) \\
&\overset{(iii)}{\geq} \max_{\mu} \sum_{i=1}^t \alpha^i_t \mathbb{D}_{\mu \times \pi^{k^i}_{-j,h}} \left( r_{j,h} + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) \\
&\overset{(iv)}{\geq} \max_{\mu} \sum_{i=1}^t \alpha^i_t \mathbb{D}_{\mu \times \pi^{k^i}_{-j,h}} \left( r_{j,h} + \mathbb{P}_h V^{\dagger, \hat{\pi}^{k^i}_{-j,h+1}}_{j,h+1} \right)(s) \overset{(v)}{\geq} V^{\dagger, \hat{\pi}^k_{-j,h}}_{j,h}(s)
\end{aligned}$$

where $(i)$ is by martingale concentration and Lemma 2, $(ii)$ is by Lemma 12, $(iii)$ is by the definition of $\beta_{j,i}$, and $(iv)$ is by induction hypothesis.

Finally, we remark that $(v)$ is not directly from Bellman equation since $\hat{\pi}^k_{-j,h}$ is non-Markov policy, and the best reponse of a non-Markov policy is not necessary a Markov policy. We prove $(v)$ as follows. Recalls definitions for policies in $\Pi_h$ as in Appendix A, by the definition, we have

$$\begin{aligned}
V^{\dagger, \hat{\pi}^k_{-j,h}}_{j,h}(s) &= \max_{\mu \in \Pi_h} V^{\mu \times \hat{\pi}^k_{-j,h}}_{j,h} \\
&\overset{(a)}{=} \max_{\mu_h} \max_{\mu_{(h+1):H}} \sum_{i=1}^t \alpha^i_t \mathbb{E}_{\boldsymbol{a} \sim \mu_h \times \pi^{k^i}_{-j,h}} \left( r_{j,h}(s, \boldsymbol{a}) + \mathbb{E}_{s'} V^{\mu_{(h+1):H}, \hat{\pi}^{k^i}_{-j,h+1}}_{j,h+1}(s, \boldsymbol{a}, s') \right) \\
&\overset{(b)}{\leq} \max_{\mu_h} \sum_{i=1}^t \alpha^i_t \mathbb{E}_{\boldsymbol{a} \sim \mu_h \times \pi^{k^i}_{-j,h}} \left( r_{j,h}(s, \boldsymbol{a}) + \mathbb{E}_{s'} \max_{\mu_{(h+1):H}} V^{\mu_{(h+1):H}, \hat{\pi}^{k^i}_{-j,h+1}}_{j,h+1}(s, \boldsymbol{a}, s') \right) \\
&\overset{(c)}{=} \max_{\mu_h} \sum_{i=1}^t \alpha^i_t \mathbb{E}_{\boldsymbol{a} \sim \mu_h \times \pi^{k^i}_{-j,h}} \left( r_{j,h}(s, \boldsymbol{a}) + \mathbb{E}_{s'} V^{\dagger, \hat{\pi}^{k^i}_{-j,h+1}}_{j,h+1}(s') \right) \\
&= \max_{\mu} \sum_{i=1}^t \alpha^i_t \mathbb{D}_{\mu \times \pi^{k^i}_{-j,h}} \left( r_{j,h} + \mathbb{P}_h V^{\dagger, \hat{\pi}^{k^i}_{-j,h+1}}_{j,h+1} \right)(s)
\end{aligned}$$

where $V^{\pi}_{j,h+1}(s, \boldsymbol{a}, s')$ for policy $\pi \in \Pi_h$ is defined as:

$$V^{\pi}_{i,h+1}(s, \boldsymbol{a}, s') := \mathbb{E}_{\pi} \left[ \sum_{h'=h+1}^H r_{h'}(s_{h'}, \boldsymbol{a}_{h'}) \middle| s_h = s, \boldsymbol{a}_h = \boldsymbol{a}, s_{h+1} = s' \right].$$

Step (a) uses the relation between $\hat{\pi}^k_{-j,h}$ and $\{\hat{\pi}^{k^i}_{-j,h+1}\}_i$. Step (b) pushes $\max$ inside summation and expectation. Step (c) is because the Markov nature of Markov game and that $\{\hat{\pi}^{k^i}_{-j,h+1}\}_i$ are policies that does not depend on history at step $h$, we know the maximization over $\mu_{(h+1):H}$ is achieved at policies in $\Pi_{h+1}$. This finishes the proof. □

To proceed with the analysis, we need to introduce two pessimistic V-estimations $\underset{\sim}{V}$ and $\underline{V}$ that are defined similarly as $\tilde{V}$ and $V$. Formally, let $t = N^k_h(s)$ denote the number of times $s$ is visited at step $h$

at the beginning of episode $k$, and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step. Then

$$\underline{V}_{j,h}^k(s) = \sum_{i=1}^{t} \alpha_t^i \left[ r_{j,h}(s, \boldsymbol{a}_h^{k^i}) + \underline{V}_{j,h+1}^{k^i}(s_{h+1}^{k^i}) - \beta_{j,i} \right], \tag{10}$$

$$\underline{V}_{j,h}^k(s) = \max\{0, \underline{V}_{j,h}^k(s)\}, \tag{11}$$

for any player $j \in [m]$ and $k \in [K]$. We emphasize that $\underline{V}$ and $\underline{V}$ are defined only for the purpose of analysis. Neither do they influence the decision made by each agent, nor do the agents need to maintain these quantities when running V-learning.

Equipped with the lower estimations, we are ready to lower bound $V_{j,h}^{\hat{\pi}_h^k}$.

**Lemma 14** (Pessimism)**.** *For any $\delta \in (0, 1]$, with probability at least $1 - \delta$, the following holds for any $(s, h, k, j) \in \mathcal{S} \times [H] \times [K] \times [m]$ and any player $j$, $\underline{V}_{j,h}^k(s) \le V_{j,h}^{\hat{\pi}_h^k}(s)$.*

*Proof of Lemma 14.* We prove by backward induction. The claim is trivially satisfied for $h = H + 1$. Suppose it is true for $h+1$, consider a fixed state $s$. It suffices to show $\underline{V}_{j,h}^k(s) \le V_{j,h}^{\hat{\pi}_h^k}(s)$ because $\underline{V}_{j,h}^k(s) = \max\{\underline{V}_{j,h}^k(s), 0\}$. Let $t = N_h^k(s)$ and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step. Then by Equation 10,

$$\begin{aligned}
\underline{V}_{j,h}^k(s) &= \sum_{i=1}^{t} \alpha_t^i \left[ r_{j,h}(s, \boldsymbol{a}_h^{k^i}) + \underline{V}_{j,h+1}^{k^i}(s_{h+1}^{k^i}) - \beta_{j,i} \right] \\
&\overset{(i)}{\le} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h \underline{V}_{j,h+1}^{k^i} \right)(s) - \sum_{i=1}^{t} \alpha_t^i \beta_{j,i} + \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) \\
&\overset{(ii)}{\le} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h \underline{V}_{j,h+1}^{k^i} \right)(s) \\
&\overset{(iii)}{\le} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h V_{j,h+1}^{\hat{\pi}_h^{k^i}} \right)(s) \\
&= V_{j,h}^{\hat{\pi}_h^k}(s)
\end{aligned}$$

where $(i)$ is by martingale concentration, $(ii)$ is by the definition of $\beta_{j,i}$, and $(iii)$ is by induction hypothesis. □

To prove Theorem 6, it remains to bound the gap $\sum_{k=1}^{K} \max_j (V_{1,j}^k - \underline{V}_{1,j}^k)(s_1)$.

*Proof of Theorem 6.* Consider player $j$, we define $\delta_{j,h}^k := V_{j,h}^k(s_h^k) - \underline{V}_{j,h}^k(s_h^k) \ge 0$. The non-negativity here is a simple consequence of the update rule and induction. We want to bound $\delta_h^k := \max_j \delta_{j,h}^k$. Let $n_h^k = N_h^k\left(s_h^k\right)$ and suppose $s_h^k$ was previously visited at episodes $k^1, \ldots, k^{n_h^k} < k$ at the $h$-th step. Now by

the update rule of $V_{j,h}^k(s_h^k)$ and $\underline{V}_{j,h}^k(s_h^k)$,

$$\begin{aligned}\delta_{j,h}^k =& V_{j,h}^k(s_h^k) - \underline{V}_{j,h}^k(s_h^k)\\ \leq& \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \left[\left(V_{j,h+1}^{k^i} - \underline{V}_{j,h+1}^{k^i}\right)\left(s_{h+1}^{k^i}\right) + 2\beta_{j,i}\right]\\ =& \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{j,h+1}^{k^i} + \mathcal{O}(H\xi(A_j, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k})\end{aligned}$$

where in the last step we have used $\sum_{i=1}^t \alpha_t^i \beta_{j,i} = \Theta(H\xi(A_j, t, \iota) + \sqrt{H^3\iota/t})$.

Now by taking maximum w.r.t. $j$ on both sides and notice $\xi(B, t, \iota)$ is non-decreasing in $B$, we have

$$\delta_h^k \leq \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} + \mathcal{O}(H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k}).$$

Summing the first two terms w.r.t. $k$,

$$\sum_{k=1}^K \alpha_{n_h^k}^0 H = \sum_{k=1}^K H\mathbb{I}\left\{n_h^k = 0\right\} \leq SH,$$

$$\sum_{k=1}^K \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} \overset{(i)}{\leq} \sum_{k'=1}^K \delta_{h+1}^{k'} \sum_{i=n_h^{k'}+1}^{\infty} \alpha_i^{n_h^{k'}} \overset{(ii)}{\leq} \left(1 + \frac{1}{H}\right) \sum_{k=1}^K \delta_{h+1}^k.$$

where $(i)$ is by changing the order of summation and $(ii)$ is by Lemma 10. Putting them together,

$$\begin{aligned}\sum_{k=1}^K \delta_h^k =& \sum_{k=1}^K \alpha_{n_h^k}^0 H + \sum_{k=1}^K \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} + \sum_{k=1}^K \mathcal{O}(H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k})\\ \leq& HS + \left(1 + \frac{1}{H}\right) \sum_{k=1}^K \delta_{h+1}^k + \sum_{k=1}^K \mathcal{O}(H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k})\end{aligned}$$

Recursing this argument for $h \in [H]$ gives

$$\sum_{k=1}^K \delta_1^k \leq eSH^2 + e\sum_{h=1}^H \sum_{k=1}^K \mathcal{O}(H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k})$$

By pigeonhole argument,

$$\begin{aligned}\sum_{k=1}^K (H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k}) =& \mathcal{O}(1) \sum_s \sum_{n=1}^{N_h^K(s)} \left(H\xi(A, n, \iota) + \sqrt{\frac{H^3\iota}{n}}\right)\\ \leq& \mathcal{O}(1) \sum_s \left(H\Xi(A, N_h^K(s), \iota) + \sqrt{H^3 N_h^K(s)\iota}\right)\\ \leq& \mathcal{O}\left(HS\Xi(A, K/S, \iota) + \sqrt{H^3SK\iota}\right),\end{aligned}$$

where in the last step we have used concavity.

Finally take the sum w.r.t. $h \in [H]$ we have

$$\sum_{k=1}^{K} \max_j [V_{1,j}^k - \underline{V}_{1,j}^k](s_1) \leq \mathcal{O}\left(H^2 S \Xi(A, K/S, \iota) + \sqrt{H^5 S K \iota}\right),$$

which implies

$$\max_{j \in [m]} [V_{j,1}^{\dagger, \hat{\pi}_{-j}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \leq \mathcal{O}((H^2 S/K) \cdot \Xi(A, K/S, \iota) + \sqrt{H^5 S \iota / K}). \qquad \square$$

# C Proofs for Computing CE in General-sum MGs

In this section, we give complete proof of Theorem 7. To avoid repeatedly state the condition of Theorem 7 in each lemma, we will use

- Condition of the adversarial bandit sub-procudure (Assumption 2) and
- Set the bonus $\{\beta_{j,t}\}_{t=1}^K$ of the $j^{\text{th}}$ player so that $\sum_{i=1}^t \alpha_t^i \beta_{j,i} = \Theta(H \xi_{\text{sw}}(A_j, t, \iota) + \sqrt{H^3 \iota / t})$ for any $t \in [K]$.

throughout the whole section.

We begin with a swap regret version of Lemma 12.

**Lemma 15.** *The following event is true with probability at least $1 - \delta$: for any $(s, h, k) \in \mathcal{S} \times [H] \times [K]$, let $t = N_h^k(s)$ and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step, then for all $j \in [m]$*

$$\max_{\phi_j} \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\phi_j \diamond \pi_{j,h}^{k^i} \times \pi_{-j,h}^{k^i}} \left[r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}\right](s) - \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left(r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}\right)(s) \leq H \xi_{sw}(A_j, t, \iota),$$

*where $\iota = \log(KHS/\delta)$.*

*Proof.* By Assumption 2 and the adversarial bandit update step in Algorithm 1, we have that with probability at least $1 - \delta$, for any $(s, h, k) \in \mathcal{S} \times [H] \times [K]$,

$$\begin{aligned} \max_{\phi_j} \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\phi_j \diamond \pi_{j,h}^{k^i} \times \pi_{-j,h}^{k^i}} \left(\frac{H - r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}}{H}\right)(s) - \sum_{i=1}^t \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left(\frac{H - r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i}}{H}\right)(s) \\ \leq \xi_{\text{sw}}(A_j, t, \iota). \end{aligned}$$

$\square$

We begin with proving $V$ is actually an optimistic estimation of the value function under best response.

**Lemma 16** (Optimism)**.** *For any $\delta \in (0, 1)$, with probability at least $1 - \delta$, the following holds for any $(s, h, k, j) \in \mathcal{S} \times [H] \times [K] \times [m]$, $V_{j,h}^k(s) \geq \max_{\phi_j} V_{j,h}^{(\phi_j \diamond \hat{\pi}_{j,h}^k) \odot \hat{\pi}_{-j,h}^k}(s)$.*

*Proof of Lemma 16.* We prove by backward induction. The claim is trivially satisfied for $h = H + 1$. Suppose it is true for $h+1$, consider a fixed state $s$. It suffices to show $\tilde{V}^k_{j,h}(s) \geq \max_{\phi_j} V^{(\phi_j \diamond \hat{\pi}^k_{j,h}) \odot \hat{\pi}^k_{-j,h}}_{j,h}(s)$ because $V^k_{j,h}(s) = \min\{\tilde{V}^k_{j,h}(s), H - h + 1\}$. Let $t = N^k_h(s)$ and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the $h$-th step. Then using Lemma 11,

$$
\begin{aligned}
\tilde{V}^k_{j,h}(s) &= \alpha^0_t(H - h + 1) + \sum_{i=1}^{t} \alpha^i_t \left[ r_{j,h}(s, \boldsymbol{a}^{k^i}_h) + V^{k^i}_{j,h+1}(s^{k^i}_{h+1}) + \beta_{j,i} \right] \\
&\overset{(i)}{\geq} \sum_{i=1}^{t} \alpha^i_t \mathbb{D}_{\pi^{k^i}_h} \left( r_{j,h} + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) + \sum_{i=1}^{t} \alpha^i_t \beta_{j,i} - \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) \\
&\overset{(ii)}{\geq} \max_{\phi_j} \sum_{i=1}^{t} \alpha^i_t \mathbb{D}_{(\phi_j \diamond \pi^{k^i}_{j,h}) \times \pi^{k^i}_{-j,h}} \left( r_h + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) + \sum_{i=1}^{t} \alpha^i_t \beta_{j,i} - \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) - H\xi(A_j, t, \iota) \\
&\overset{(iii)}{\geq} \max_{\phi_j} \sum_{i=1}^{t} \alpha^i_t \mathbb{D}_{(\phi_j \diamond \pi^{k^i}_{j,h}) \times \pi^{k^i}_{-j,h}} \left( r_h + \mathbb{P}_h V^{k^i}_{j,h+1} \right)(s) \\
&\overset{(iv)}{\geq} \max_{\phi_j} \sum_{i=1}^{t} \alpha^i_t \mathbb{D}_{(\phi_j \diamond \pi^{k^i}_{j,h}) \times \pi^{k^i}_{-j,h}} \left( r_h + \mathbb{P}_h \max_{\phi'_j} V^{(\phi'_j \diamond \hat{\pi}^{k^i}_{j,h+1}) \odot \hat{\pi}^{k^i}_{-j,h+1}}_{j,h} \right)(s) \\
&\overset{(v)}{\geq} \max_{\phi_j} V^{(\phi_j \diamond \hat{\pi}^k_{j,h}) \odot \hat{\pi}^k_{-j,h}}_{j,h}(s)
\end{aligned}
$$

where $(i)$ is by martingale concentration and Lemma 2, $(ii)$ is by Lemma 15, $(iii)$ is by the definition of $\beta_{j,i}$, and $(iv)$ is by induction hypothesis. Finally, $(v)$ follows from a similar reasoning as in the proof of Lemma 13, which we omit here. □

We still need to lower bound $V^{\hat{\pi}^k_h}_{j,h}$. To do this, we estimate $\underline{V}$ and $\underline{\underline{V}}$ defined by Equation 10 and Equation 11. These quantities are indeed the lower bounds we need.

**Lemma 17** (Pessimism)**.** *For any $\delta \in (0,1)$, with probability at least $1 - \delta$, the following holds for any $(s, h, k, j) \in \mathcal{S} \times [H] \times [K] \times [m]$, $\underline{\underline{V}}^k_{j,h}(s) \leq V^{\hat{\pi}^k_h}_h(s)$.*

*Proof of Lemma 17.* We prove by backward induction. The claim is trivially satisfied for $h = H + 1$. Suppose it is true for $h+1$, consider a fixed state $s$. It suffices to show $\underline{V}^k_{j,h}(s) \leq V^{\hat{\pi}^k_h}_{j,h}(s)$ because $\underline{\underline{V}}^k_{j,h}(s) = \max\{\underline{V}^k_{j,h}(s), 0\}$. Let $t = N^k_h(s)$ and suppose $s$ was previously visited at episodes $k^1, \ldots, k^t < k$ at the

$h$-th step. Then by equation (10),

$$\begin{aligned}
\underline{V}_{j,h}^{k}(s) &= \sum_{i=1}^{t} \alpha_t^i \left[ r_{j,h}(s, \boldsymbol{a}_h^{k^i}) + \underline{V}_{j,h+1}^{k^i}(s_{h+1}^{k^i}) - \beta_{j,i} \right] \\
&\overset{(i)}{\leq} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h \underline{V}_{j,h+1}^{k^i} \right)(s) - \sum_{i=1}^{t} \alpha_t^i \beta_{j,i} + \mathcal{O}\left( \sqrt{\frac{H^3 \iota}{t}} \right) \\
&\overset{(ii)}{\leq} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h \underline{V}_{j,h+1}^{k^i} \right)(s) \\
&\overset{(iii)}{\leq} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_h + \mathbb{P}_h V_{j,h+1}^{\hat{\pi}_h^{k^i}} \right)(s) \\
&= V_{j,h}^{\hat{\pi}_h^k}(s)
\end{aligned}$$

where $(i)$ is by martingale concentration, $(ii)$ is by the definition of $\beta_{j,i}$, and $(iii)$ is by induction hypothesis. □

To prove Theorem 7, it remains to bound the gap $\sum_{k=1}^{K} \max_j (V_{1,j}^k - \underline{V}_{1,j}^k)(s_1)$.

*Proof of Theorem 7.* Consider player $j$, we define $\delta_{j,h}^k := V_{j,h}^k(s_h^k) - \underline{V}_{j,h}^k(s_h^k) \geq 0$. The non-negativity here is a simple consequence of the update rule and induction. We want to bound $\delta_h^k := \max_j \delta_{j,h}^k$. Let $n_h^k = N_h^k\left(s_h^k\right)$ and suppose $s_h^k$ was previously visited at episodes $k^1, \ldots, k^{n_h^k} < k$ at the $h$-th step. Now by the update rule of $V_{j,h}^k(s_h^k)$ and $\underline{V}_{j,h}^k(s_h^k)$,

$$\begin{aligned}
\delta_{j,h}^k =& V_{j,h}^k(s_h^k) - \underline{V}_{j,h}^k(s_h^k) \\
\leq& \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \left[ \left( V_{j,h+1}^{k^i} - \underline{V}_{j,h+1}^{k^i} \right)\left( s_{h+1}^{k^i} \right) + 2\beta_{j,i} \right] \\
=& \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{j,h+1}^{k^i} + \mathcal{O}(H \xi_{\text{sw}}(A_j, n_h^k, \iota) + \sqrt{H^3 \iota / n_h^k})
\end{aligned}$$

where in the last step we have used $\sum_{i=1}^{t} \alpha_t^i \beta_{j,i} = \Theta(H \xi_{\text{sw}}(A_j, t, \iota) + \sqrt{H^3 \iota / t})$.

Now by taking maximum w.r.t. $j$ on both sides and notice $\xi_{\text{sw}}(B, t, \iota)$ is non-decreasing in $B$, we have

$$\delta_h^k \leq \alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} + \mathcal{O}(H \xi_{\text{sw}}(A, n_h^k, \iota) + \sqrt{H^3 \iota / n_h^k}).$$

Summing the first two terms w.r.t. $k$,

$$\sum_{k=1}^{K} \alpha_{n_h^k}^0 H = \sum_{k=1}^{K} H \mathbb{I}\left\{ n_h^k = 0 \right\} \leq SH,$$

$$\sum_{k=1}^{K} \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} \overset{(i)}{\leq} \sum_{k'=1}^{K} \delta_{h+1}^{k'} \sum_{i=n_h^{k'}+1}^{\infty} \alpha_i^{n_h^{k'}} \overset{(ii)}{\leq} \left( 1 + \frac{1}{H} \right) \sum_{k=1}^{K} \delta_{h+1}^k.$$

where $(i)$ is by changing the order of summation and $(ii)$ is by Lemma 10. Putting them together,

$$\begin{aligned}\sum_{k=1}^{K}\delta_h^k =& \sum_{k=1}^{K}\alpha_{n_h^k}^0 H + \sum_{k=1}^{K}\sum_{i=1}^{n_h^k}\alpha_{n_h^k}^i \delta_{h+1}^{k^i} + \sum_{k=1}^{K}\mathcal{O}(H\xi_{\text{sw}}(A,n_h^k,\iota)+\sqrt{H^3\iota/n_h^k})\\ \leq& HS + \left(1+\frac{1}{H}\right)\sum_{k=1}^{K}\delta_{h+1}^k + \sum_{k=1}^{K}\mathcal{O}(H\xi_{\text{sw}}(A,n_h^k,\iota)+\sqrt{H^3\iota/n_h^k})\end{aligned}$$

Recursing this argument for $h \in [H]$ gives

$$\sum_{k=1}^{K}\delta_1^k \leq eSH^2 + e\sum_{h=1}^{H}\sum_{k=1}^{K}\mathcal{O}(H\xi_{\text{sw}}(A,n_h^k,\iota)+\sqrt{H^3\iota/n_h^k})$$

By pigeonhole argument,

$$\begin{aligned}\sum_{k=1}^{K}(H\xi_{\text{sw}}(A,n_h^k,\iota)+\sqrt{H^3\iota/n_h^k}) =& \mathcal{O}(1)\sum_{s}\sum_{n=1}^{N_h^K(s)}\left(H\xi_{\text{sw}}(A,n,\iota)+\sqrt{\frac{H^3\iota}{n}}\right)\\ \leq& \mathcal{O}(1)\sum_{s}\left(H\Xi_{\text{sw}}(A,N_h^K(s),\iota)+\sqrt{H^3N_h^K(s)\iota}\right)\\ \leq& \mathcal{O}\left(HS\Xi_{\text{sw}}(A,K/S,\iota)+\sqrt{H^3SK\iota}\right),\end{aligned}$$

where in the last step we have used concavity.

Finally take the sum w.r.t. $h \in [H]$ we have

$$\sum_{k=1}^{K}\max_j[V_{1,j}^k - \underline{V}_{1,j}^k](s_1) \leq \mathcal{O}\left(H^2S\Xi_{\text{sw}}(A,K/S,\iota)+\sqrt{H^5SK\iota}\right),$$

which implies

$$\max_{j\in[m]}[V_{j,1}^{\dagger,\hat{\pi}_{-j}}(s_1) - V_{j,1}^{\hat{\pi}}(s_1)] \leq \mathcal{O}((H^2S/K)\cdot\Xi_{\text{sw}}(A,K/S,\iota)+\sqrt{H^5S\iota/K}).$$

□

# D Proofs for MDPs and Two-player Zero-sum MGs

In this section, we prove the main theorems for V-learning in the setting of single-agent (MDPs) and two-player zero-sum MGs.

*Proof of Theorem 5.* To begin with, we notice an equivalent definition of two-player zero-sum MGs is that the reward function satisfies $r_{1,h} = 1 - r_{2,h}$ for all $h \in [H]$. The reason we use this definition instead of the common version $r_{1,h} = -r_{2,h}$ is we want to make it consistent with our assumption that the reward function takes value in $[0,1]$ for any player. Although this definition does not satisfy the zero-sum condition, its Nash equilibria are the same as those of the zero-sum version because adding a constant to the reward function of player 2 per step will not change the dynamics of the game.

In order to show $\hat{\pi} = \hat{\pi}_1 \times \hat{\pi}_2$ is an approximate Nash policy, it suffices to control

$$\max_{\pi_1} V_{1,1}^{\pi_1,\hat{\pi}_2}(s_1) - \min_{\pi_2} V_{1,1}^{\hat{\pi}_1,\pi_2}(s_1).$$

Since $r_{1,h} = 1 - r_{2,h}$ for all $h \in [H]$, with probability at least $1-\delta$

$$\begin{aligned}
&\max_{\pi_1} V_{1,1}^{\pi_1,\hat{\pi}_2}(s_1) - \min_{\pi_2} V_{1,1}^{\hat{\pi}_1,\pi_2}(s_1) \\
=&\max_{\pi_1} V_{1,1}^{\pi_1,\hat{\pi}_2}(s_1) - \left(H - \max_{\pi_2} V_{2,1}^{\hat{\pi}_1,\pi_2}(s_1)\right) \\
=&\left(\max_{\pi_1} V_{1,1}^{\pi_1,\hat{\pi}_2}(s_1) - V_{1,1}^{\hat{\pi}_1\odot\hat{\pi}_2}(s_1)\right) + \left(\max_{\pi_2} V_{2,1}^{\hat{\pi}_1,\pi_2}(s_1) - V_{2,1}^{\hat{\pi}_1\odot\hat{\pi}_2}(s_1)\right) \\
\le&\mathcal{O}((H^2S/K)\cdot\Xi(A,K/S,\iota) + \sqrt{H^5S\iota/K}),
\end{aligned}$$

where the last inequality follows from Theorem 6. The reason we can use Theorem 6 here is the precondition of Theorem 5 is a special case of the precondition of Theorem 6. □

*Proof of Theorem 4.* Since MDPs is a subclass of two-player zero-sum MGs by simply choosing the action set of the second player to be a singleton, it suffices to only prove Theorem 5, from which the single-agent guarantee, Theorem 4 trivially follows. □

# E Proofs for Monotonic V-learning

In this section, we prove Theorem 8. The algorithm is V-learning with monotonic update, and the setting we consider is two-player zero-sum Markov games. As before, we assume $r_{1,h}(s,a) = 1 - r_{2,h}(s,a)$ for all $s, a, h$. The reason for assuming $r_{1,h}(s,a) = 1 - r_{2,h}(s,a)$ instead of $r_{1,h}(s,a) = -r_{2,h}(s,a)$ can be found in Appendix D.

For two player zero-sum MGs, we can define its minimax value function (Nash value function) by the following Bellman equations

$$\begin{cases} V_{j,h}^{\star}(s) = \max_{\pi_{j,h}} \min_{\pi_{-j,h}} \mathbb{D}_{\pi_{j,h}\times\pi_{-j,h}}[Q_{j,h}^{\star}](s), \\ Q_{j,h}^{\star}(s,\boldsymbol{a}) = r_{j,h}(s,\boldsymbol{a}) + \mathbb{P}_h[V_{j,h+1}^{\star}](s,\boldsymbol{a}), \\ V_{j,H+1}^{\star}(s) = Q_{j,H+1}^{\star}(s,\boldsymbol{a}) = 0. \end{cases} \tag{12}$$

**Lemma 18** (Optimism of V-estimates)**.** *With probability at least $1-\delta$, for any $(s,h,k,j) \in \mathcal{S}\times[H]\times[K]\times[2]$,*

$$\tilde{V}_{j,h}^{k}(s) \ge V_{j,h}^{k}(s) \ge V_{j,h}^{\dagger,\tilde{\pi}_{-j}}(s) \ge V_{j,h}^{\star}(s), \tag{13}$$

*where $V_{j,h}^{\star}$ is the minimax (Nash) value function defined above.*

*Proof of Lemma 18.* Note that $\tilde{V}_{j,h}^{k}(s) \ge V_{j,h}^{k}(s)$ is straightforward by the update rule of V-learning, and $V_{j,h}^{\dagger,\tilde{\pi}_{-j}}(s) \ge V_{j,h}^{\star}(s)$ directly follows from the definition of minimax value function. Therefore, we only need to prove the second inequality. We do this by backward induction.

The claim is true for $h = H+1$. Assume for any $s$ and $k$, $V_{j,h+1}^{k}(s) \ge V_{j,h+1}^{\dagger,\tilde{\pi}_{-j}}(s)$. For a fixed $(s,h,k) \in \mathcal{S}\times[H]\times[K]$, let $t = N_h^k(s)$ and suppose $s$ was previously visited in episode $k^1, \ldots, k^t < k$

at the $h$-th step. By Bellman equation,

$$\begin{aligned}
V_{j,h}^{\dagger,\tilde{\pi}_{-j}}(s) \leq & \alpha_t^0(H-h+1) + \max_{\mu} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\mu \times \pi_{-j,h}^{k^i}} \left( r_{j,h} + \mathbb{P}_h V_{j,h+1}^{\dagger,\tilde{\pi}_{-j}} \right)(s) \\
\leq & \alpha_t^0(H-h+1) + \max_{\mu} \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\mu \times \pi_{-j,h}^{k^i}} \left( r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i} \right)(s) \\
\leq & \alpha_t^0(H-h+1) + \sum_{i=1}^{t} \alpha_t^i \mathbb{D}_{\pi_h^{k^i}} \left( r_{j,h} + \mathbb{P}_h V_{j,h+1}^{k^i} \right)(s) + H\xi(A_j,t,\iota) \\
\leq & \alpha_t^0(H-h+1) + \sum_{i=1}^{t} \alpha_t^i \left[ r_{j,h}(s, \boldsymbol{a}_h^{k^i}) + V_{j,h+1}^{k^i}(s_{h+1}^{k^i}) \right] + \mathcal{O}\left( \sqrt{\frac{2H^3\iota}{t}} \right) + H\xi(A_j,t,\iota)
\end{aligned}$$

where the second inequality follows from our induction hypothesis and the monotonicity of $V^k$, the third inequality follows from Lemma 12, and the last one follows from martingale concentration as well as Lemma 10. By Lemma 11 and the precondition of Theorem 8, we know the RHS is no larger than $\tilde{V}_{j,h}^k(s)$. Note that $V^k$ can be equivalently defined as

$$V_{j,h}^k(s) = \min\{\min_{t \in [k]} \tilde{V}_{j,h}^t(s), H-h+1\},$$

we conclude $V_{j,h}^k(s) \geq V_{j,h}^{\dagger,\tilde{\pi}_{-j}}(s)$ for any $k \in [K]$. □

Now we are ready to prove Theorem 8.

*Proof of Theorem 8.* By the monotonicity of $V$ and Lemma 18

$$\begin{aligned}
& V_{1,1}^{\dagger,\tilde{\pi}_2}(s_1) - \min_{\pi_2} V_{1,1}^{\tilde{\pi}_1 \times \pi_2}(s_1) \\
= & V_{1,1}^{\dagger,\tilde{\pi}_2}(s_1) - \left( H - V_{2,1}^{\dagger,\tilde{\pi}_1}(s_1) \right) \\
\leq & V_{1,1}^K(s_1) + V_{2,1}^K(s_1) - H \\
\leq & \frac{1}{K} \sum_{k=1}^{K} \left( V_{1,1}^k(s_1) + V_{2,1}^k(s_1) - H \right) \\
\leq & \frac{1}{K} \sum_{k=1}^{K} \left( \tilde{V}_{1,1}^k(s_1) + \tilde{V}_{2,1}^k(s_1) - H \right),
\end{aligned}$$

where the first equality follows from the definition of two-player zero-sum game, i.e., $r_{1,h} = 1 - r_{2,h}$.

Now we can mimic the proof of Theorem 6. Define $\delta_h^k := \tilde{V}_{1,h}^k(s_h^k) + \tilde{V}_{2,h}^k(s_h^k) - (H-h+1)$. The non-negativity here follows from Lemma 18 as below

$$\tilde{V}_{1,h}^k(s_h^k) + \tilde{V}_{2,h}^k(s_h^k) - (H-h+1) \geq V_{1,h}^\star(s_h^k) + V_{2,h}^\star(s_h^k) - (H-h+1) = (H-h+1) - (H-h+1) = 0.$$

Let $n_h^k = N_h^k\left(s_h^k\right)$ and suppose $s_h^k$ was previously visited at episodes $k^1, \ldots, k^{n_h^k} < k$ at the $h$-th step. By

**Algorithm 5** FTRL for Weighted External Regret (FTRL)

**Initialize:** for any $b \in \mathcal{B}$, $\theta_1(b) \leftarrow 1/B$.
**for** episode $t = 1, \dots, K$ **do**
  Take action $b_t \sim \theta_t(\cdot)$, and observe loss $\tilde{l}_t(b_t)$.
  $\hat{l}_t(b) \leftarrow \tilde{l}_t(b_t)\mathbb{I}\{b_t = b\}/(\theta_t(b) + \gamma_t)$ for all $b \in \mathcal{B}$.
  $\theta_{t+1}(b) \propto \exp[-(\eta_t/w_t) \cdot \sum_{i=1}^{t} w_i \hat{l}_i(b)]$

Lemma 11 and the fact that $r_{1,h} = 1 - r_{2,h}$ for all $h$, we have

$$\begin{aligned}
\delta_h^k =& \tilde{V}_{1,h}^k(s_h^k) + \tilde{V}_{2,h}^k(s_h^k) - (H - h + 1) \\
=& 2\alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \left[ \left( \tilde{V}_{1,h+1}^{k^i} - \tilde{V}_{2,h+1}^{k^i} \right) \left( s_{h+1}^{k^i} \right) - (H - h) + \beta_{1,i} + \beta_{2,i} \right] \\
=& 2\alpha_{n_h^k}^0 H + \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \delta_{h+1}^{k^i} + \mathcal{O}(H\xi(A, n_h^k, \iota) + \sqrt{H^3\iota/n_h^k})
\end{aligned}$$

where in the last step we used $\sum_{i=1}^{t} \alpha_t^i \beta_{j,i} = \Theta(H\xi(A_j, t, \iota) + \sqrt{H^3\iota/t})$.

The remaining steps follow exactly the same as the proof of Theorem 6. As a result, we obtain

$$\begin{aligned}
V_{1,1}^{\dagger,\tilde{\pi}_2}(s_1) - \min_{\tilde{\pi}_2} V_{1,1}^{\tilde{\pi}_1 \times \tilde{\pi}_2}(s_1) &\le \frac{1}{K} \sum_{k=1}^{K} \left( \tilde{V}_{1,1}^k(s_1) + \tilde{V}_{2,1}^k(s_1) - H \right) \\
&\le \mathcal{O}\left( \frac{H^2 S}{K} \cdot \Xi(A, K/S, \iota) + \sqrt{\frac{H^5 S \iota}{K}} \right),
\end{aligned}$$

which completes the proof. □

# F Adversarial Bandit with Weighted External Regret

In this section, we present a Follow-the-Regularized-Leader (FTRL) style algorithm that achieves low weighted (external) regret for the adversarial bandit problem. Although FTRL is a classial algorithm in the adversarial bandit literature, we did not find a good reference of FTRL with changing step size, weighted regret and high probability bound. For completeness of this work, we provide detailed derivations here.

We present the FTRL algorithm in Algorithm 5. In Corollary 19, we prove that FTRL satisfies the Assumption 1 with good regret bounds. Recall that $B$ is the number of actions, and our normalization condition requires loss $\tilde{l}_t \in [0, 1]^B$ for any $t$.

**Corollary 19.** *By choosing hyperparameter $w_t = \alpha_t \left( \prod_{i=2}^{t} (1 - \alpha_i) \right)^{-1}$ and $\eta_t = \gamma_t = \sqrt{\frac{H \log B}{Bt}}$, FTRL (Algorithm 5) satisfies Assumption 1 with*

$$\xi(B, t, \log(1/\delta)) = 10\sqrt{HB \log(B/\delta)/t}, \qquad \Xi(B, t, \log(1/\delta)) = 20\sqrt{HBt \log(B/\delta)}$$

To prove Corollary 19, we show a more general weighted regret guarantee which works for any set of weights $\{w_i\}_{i=1}^{\infty}$ in addition to $\{\alpha_t^i\}_{i=1}^{t}$. In particular, a general weighted regret is defined as

$$\mathcal{R}(t) = \max_{\theta^\star} \sum_{i=1}^{t} w_i \langle \theta_i - \theta^\star, l_i \rangle \tag{14}$$

**Theorem 20.** *For any $t \le K$, following Algorithm 5, if $\eta_i \le 2\gamma_i$ and $\eta_i$ is non-increasing for all $i \le t$, let $\iota = \log(B/\delta)$ , then with probability $1 - 3\delta$, we have*

$$\mathcal{R}(t) \le \frac{w_t \log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t} \eta_i w_i + \frac{1}{2}\max_{i\le t} w_i \iota + B\sum_{i=1}^{t}\gamma_i w_i + \sqrt{2\iota \sum_{i=1}^{t} w_i^2} + \max_{i\le t} w_i \iota/\gamma_t.$$

We postpone the proof of theorem 20 to the end of this section. We first show how to obtain Corollary 19 from Theorem 20.

*Proof of Corollary 19.* The weights $\{w_t\}_{t=1}^K$ we choose satisfy a nice property: for any $t$ we have

$$\frac{w_i}{w_j} = \frac{\alpha_t^i}{\alpha_t^j}.$$

We prove this for $i \le j$ and the other case is similar. By definition,

$$\frac{w_i}{w_j} = \frac{\alpha_i}{\alpha_j}\prod_{k=i+1}^{j}(1-\alpha_k),$$

and

$$\frac{\alpha_t^i}{\alpha_t^j} = \frac{\alpha_i}{\alpha_j}\prod_{k=i+1}^{j}(1-\alpha_k).$$

We can easily verify that the RHS are the same.

Define $\tilde{\mathcal{R}}(t) := \max_{\theta\in\Delta_{\mathcal{B}}} \sum_{i=1}^{t} \alpha_t^i[\langle \theta_i, \ell_i\rangle - \langle \theta, \ell_i\rangle]$. By plugging $w_t = \alpha_t \left(\prod_{i=2}^{t}(1-\alpha_i)\right)^{-1}$ into Theorem 20, and using the property above, we have the regret guarantee

$$\tilde{\mathcal{R}}(t) \le \frac{\alpha_t \log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i \alpha_t^i + \frac{1}{2}\alpha_t\iota + B\sum_{i=1}^{t}\gamma_i\alpha_t^i + \sqrt{2\iota\sum_{i=1}^{t}\left(\alpha_t^i\right)^2} + \alpha_t\iota/\gamma_t.$$

By choosing $\eta_t = \gamma_t = \sqrt{\frac{H\log B}{Bt}}$ and using Lemma 10, we can further upper bound the regret by

$$\begin{aligned}
\tilde{\mathcal{R}}(t) \le& \frac{(H+1)\log B}{H+t}\sqrt{\frac{Bt}{H\log B}} + \frac{3}{2}\sqrt{HB\log B}\sum_{i=1}^{t}\frac{\alpha_t^i}{\sqrt{t}} \\
&+ \frac{(H+1)\,\iota}{2\,(H+t)} + \sqrt{2\iota\sum_{i=1}^{t}\left(\alpha_t^i\right)^2} + \frac{(H+1)\,\iota}{(H+t)}\sqrt{\frac{Bt}{H\log B}} \\
\le& 2\sqrt{\frac{HB\log B}{t}} + 3\sqrt{\frac{HB\log B}{t}} + \frac{H\iota}{t} + 2\sqrt{\frac{H\iota}{t}} + 2\sqrt{\frac{HB\log B}{t}} \\
\le& 9\sqrt{HB\iota/t} + H\iota/t.
\end{aligned}$$

To further simplify the above upper bound, consider two cases:

- If $H\iota/t \le 1$, $\sqrt{H\iota/t} \ge H\iota/t$ and thus $\tilde{\mathcal{R}}(t) \le 10\sqrt{HB\iota/t}$.

- If $H\iota/t \geq 1$, $\sqrt{HB\iota/t} \geq 1 \geq \tilde{\mathcal{R}}(t)$ where the last step is by the definition of $\tilde{\mathcal{R}}(t)$. Therefore we have $\tilde{\mathcal{R}}(t) \leq \sqrt{HB\iota/t}$.

Combining the two cases above gives $\tilde{\mathcal{R}}(t) \leq 10\sqrt{HB\iota/t}$.

Finally, we pick $\xi(B, t, \log(1/\delta)) := 10\sqrt{HB\iota/t}$, which is non-decreasing in $B$. Since $\sum_{t'=1}^{t} \xi(B, t, \log(1/\delta)) \leq 20\sqrt{HBt\iota}$, we choose $\Xi(B, t, \log(1/\delta)) = 20\sqrt{HBt\iota}$, which is concave in $t$. □

To prove Theorem 20, we first note that the weighted regret (14) can be decomposed into three terms

$$\begin{aligned}\sum_{i=1}^{t} w_i \left\langle \theta_i - \theta^\star, l_i \right\rangle =& \sum_{i=1}^{t} w_i \left\langle \theta_i - \theta^\star, l_i \right\rangle \\ =& \underbrace{\sum_{i=1}^{t} w_i \left\langle \theta_i - \theta^\star, \hat{l}_i \right\rangle}_{(A)} + \underbrace{\sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \hat{l}_i \right\rangle}_{(B)} + \underbrace{\sum_{i=1}^{t} w_i \left\langle \theta^\star, \hat{l}_i - l_i \right\rangle}_{(C)} \end{aligned} \tag{15}$$

The rest of this section is devoted to bounding three terms above. We begin with the following useful lemma adapted from Lemma 1 in [33], which is crucial in achieving high probability guarantees.

**Lemma 21.** *For any sequence of coefficients $c_1, c_2, \ldots, c_t$ s.t. $c_i \in [0, 2\gamma_i]^B$ is $\mathcal{F}_i$-measurable, we have with probability $1 - \delta$,*

$$\sum_{i=1}^{t} w_i \left\langle c_i, \hat{l}_i - l_i \right\rangle \leq \max_{i \leq t} w_i \iota.$$

*Proof.* Define $w = \max_{i \leq t} w_i$. By definition,

$$\begin{aligned} w_i \hat{l}_i(b) =& \frac{w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{\theta_i(b) + \gamma_i} \leq \frac{w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{\theta_i(b) + \frac{w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w} \gamma_i} \\ =& \frac{w}{2\gamma_i} \frac{\frac{2\gamma_i w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}}{1 + \frac{\gamma_i w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}} \overset{(i)}{\leq} \frac{w}{2\gamma_i} \log\left(1 + \frac{2\gamma_i w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}\right) \end{aligned}$$

where $(i)$ follows from $\frac{z}{1+z/2} \leq \log(1+z)$ for all $z \geq 0$.

Defining the sum

$$\hat{S}_i = \frac{w_i}{w} \left\langle c_i, \hat{l}_i \right\rangle, \quad S_i = \frac{w_i}{w} \left\langle c_i, l_i \right\rangle,$$

we have

$$\begin{aligned} \mathbb{E}_i\left[\exp\left(\hat{S}_i\right)\right] &\leq \mathbb{E}_i\left[\exp\left(\sum_b \frac{c_i(b)}{2\gamma_i} \log\left(1 + \frac{2\gamma_i w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}\right)\right)\right] \\ &\overset{(i)}{\leq} \mathbb{E}_i\left[\prod_b \left(1 + \frac{c_i(b) w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}\right)\right] \\ &= \mathbb{E}_i\left[1 + \sum_b \frac{c_i(b) w_i \tilde{l}_i(b) \mathbb{I}\{b_i = b\}}{w\theta_i(b)}\right] \\ &= 1 + S_i \leq \exp(S_i) \end{aligned}$$

where $(i)$ follows from $z_1 \log(1+z_2) \le \log(1+z_1 z_2)$ for any $0 \le z_1 \ge 1$ and $z_2 \ge -1$. Note that here we are using the condition $c_i(b) \le 2\gamma_i$ for all $b \in [B]$.

Equipped with the above bound, we can now prove the concentration result.

$$\begin{aligned}
\mathbb{P}\left[\sum_{i=1}^{t}\left(\hat{S}_i - S_i\right) \ge \iota\right] &= \mathbb{P}\left[\exp\left[\sum_{i=1}^{t}\left(\hat{S}_i - S_i\right)\right] \ge \frac{B}{\delta}\right] \\
&\le \frac{\delta}{B}\mathbb{E}_t\left[\exp\left[\sum_{i=1}^{t}\left(\hat{S}_i - S_i\right)\right]\right] \\
&\le \frac{\delta}{B}\mathbb{E}_{t-1}\left[\exp\left[\sum_{i=1}^{t-1}\left(\hat{S}_i - S_i\right)\right] E_t\left[\exp\left(\hat{S}_t - S_t\right)\right]\right] \\
&\le \frac{\delta}{B}\mathbb{E}_{t-1}\left[\exp\left[\sum_{i=1}^{t-1}\left(\hat{S}_i - S_i\right)\right]\right] \\
&\le \cdots \le \frac{\delta}{B}.
\end{aligned}$$

We conclude the proof by taking a union bound. □

With Lemma 21, we can bound the three terms $(A)$,$(B)$ and $(C)$ in (15) separately as below.

**Lemma 22.** *For any $t \in [K]$, suppose $\eta_i \le 2\gamma_i$ for all $i \le t$. Then with probability at least $1-\delta$, for any $\theta^\star \in \Delta^B$,*

$$\sum_{i=1}^{t} w_i \left\langle \theta_i - \theta^\star, \hat{l}_i \right\rangle \le \frac{w_t \log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\le t} w_i \iota.$$

*Proof.* We use the standard analysis of FTRL with changing step size, see for example Exercise 28.13 in [27]. Notice the essential step size is $\eta_t / w_t$,

$$\begin{aligned}
\sum_{i=1}^{t} w_i \left\langle \theta_i - \theta^\star, \hat{l}_i \right\rangle &\le \frac{w_t \log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\eta_i w_i \left\langle \theta_i, \hat{l}_i^2 \right\rangle \\
&\le \frac{w_t \log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b\in\mathcal{B}}\eta_i w_i \hat{l}_i(b) \\
&\overset{(i)}{\le} \frac{w_t \log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b\in\mathcal{B}}\eta_i w_i l_i(b) + \frac{1}{2}\max_{i\le t} w_i \iota \\
&\le \frac{w_t \log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\le t} w_i \iota
\end{aligned}$$

where $(i)$ is by using Lemma 21 with $c_i(b) = \eta_i$ for any $b$. The any-time guarantee is justifed by taking union bound. □

**Lemma 23.** *For any $t \in [K]$, with probability $1-\delta$,*

$$\sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \hat{l}_i \right\rangle \le B\sum_{i=1}^{t}\gamma_i w_i + \sqrt{2\iota \sum_{i=1}^{t} w_i^2}.$$

*Proof.* We further decopose it into

$$\sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \hat{l}_i \right\rangle = \sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \mathbb{E}_i \hat{l}_i \right\rangle + \sum_{i=1}^{t} w_i \left\langle \theta_i, \mathbb{E}_i \hat{l}_i - \hat{l}_i \right\rangle.$$

The first term is bounded by

$$\begin{aligned}\sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \mathbb{E}_i \hat{l}_i \right\rangle &= \sum_{i=1}^{t} w_i \left\langle \theta_i, l_i - \frac{\theta_i}{\theta_i + \gamma_i} l_i \right\rangle \\ &= \sum_{i=1}^{t} w_i \left\langle \theta_i, \frac{\gamma_i}{\theta_i + \gamma_i} l_i \right\rangle \leq B \sum_{i=1}^{t} \gamma_i w_i.\end{aligned}$$

To bound the second term, notice

$$\left\langle \theta_i, \hat{l}_i \right\rangle \leq \sum_{b \in \mathcal{B}} \theta_i(b) \frac{\mathbb{I}\{b_t = b\}}{\theta_i(b) + \gamma_i} \leq \sum_{b \in \mathcal{B}} \mathbb{I}\{b_i = b\} = 1,$$

thus $\{w_i \left\langle \theta_i, \mathbb{E}_i \hat{l}_i - \hat{l}_i \right\rangle\}_{i=1}^{t}$ is a bounded martingale difference sequence w.r.t. the filtration $\{\mathcal{F}_i\}_{i=1}^{t}$. By Azuma-Hoeffding,

$$\sum_{i=1}^{t} \left\langle \theta_i, \mathbb{E}_i \hat{l}_i - \hat{l}_i \right\rangle \leq \sqrt{2\iota \sum_{i=1}^{t} w_i^2}.$$

□

**Lemma 24.** *For any $t \in [K]$, with probability $1 - \delta$, for any $\theta^\star \in \Delta^B$, if $\gamma_i$ is non-increasing in $i$,*

$$\sum_{i=1}^{t} w_i \left\langle \theta^\star, \hat{l}_i - l_i \right\rangle \leq \max_{i \leq t} w_i \iota / \gamma_t.$$

*Proof.* Define a basis $\{e_j\}_{j=1}^{B}$ of $\mathbb{R}^B$ by

$$e_j(b) = \begin{cases} 1 \text{ if } a = j \\ 0 \text{ otherwise} \end{cases}$$

Then for all the $j \in [B]$, we can apply Lemma 21 with $c_i = \gamma_t e_j$. Sincee $c_i(b) \leq \gamma_t \leq \gamma_i$, the condition in Lemma 21 is satisfied. As a result,

$$\sum_{i=1}^{t} w_i \left\langle e_j, \hat{l}_i - l_i \right\rangle \leq \max_{i \leq t} w_i \iota / \gamma_t.$$

Since any $\theta^\star$ is a convex combination of $\{e_j\}_{j=1}^{B}$, by taking the union bound over $j \in [B]$, we have

$$\sum_{i=1}^{t} w_i \left\langle \theta^\star, \hat{l}_i - l_i \right\rangle \leq \max_{i \leq t} w_i \iota / \gamma_t.$$

□

**Algorithm 6** FTRL for Weighted Swap Regret (FTRL_swap)

**Initialize:** for any $b \in \mathcal{B}$, $\theta_1(b) \leftarrow 1/B$.
**for** episode $t = 1, \ldots, K$ **do**
  Take action $b_t \sim \theta_t(\cdot)$, and observe loss $\tilde{l}_t(b_t)$.
  **for** each action $b \in \mathcal{B}$ **do**
    $\hat{l}_t(\cdot|b) \leftarrow \theta_t(b)\tilde{l}_t(b_t)\mathbb{I}\{b_t = \cdot\}/(\theta_t(\cdot) + \gamma_t)$.
    $\tilde{\theta}_{t+1}(\cdot|b) \propto \exp[-(\eta_t/w_t) \cdot \sum_{i=1}^{t} w_i \hat{l}_i(\cdot|b)]$
  Set $\theta_{t+1}$ such that $\theta_{t+1}(\cdot) = \sum_a \theta_{t+1}(b)\tilde{\theta}_{t+1}(\cdot|b)$.

Finally we are ready to prove Theorem 20.

*Proof of Theorem 20.* Note the conditions in Lemma 22 and Lemma 24 are satisfied by assumptions. Recall the regret decomposition (15). By bounding $(A)$ in Lemma 22, $(B)$ in Lemma 23 and $(C)$ in Lemma 24, with probability $1 - 3\delta$, we have that

$$\mathcal{R}(t) \leq \frac{w_t \log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\leq t} w_i\iota + B\sum_{i=1}^{t}\gamma_i w_i + \sqrt{2\iota\sum_{i=1}^{t} w_i^2} + \max_{i\leq t} w_i\iota/\gamma_t. \qquad \square$$

# G Adversarial Bandit with Weighted Swap Regret

In this section, we adapt Follow-the-Regularized-Leader (FTRL) algorithm that achieves low weighted swap regret for the adversarial bandit problem. We follow a similar technique presented in [5] which adapts external regret algorithms to swap regret algorithms for the unweighted case.

We present the FTRL_swap algorithm in Algorithm 6. Different from FTRL (Algorithm 5), FTRL_swap maintains an additional $B \times B$ matrix $\tilde{\theta}_t(\cdot|\cdot)$, and uses its eigenvector when taking actions. The matrix will be updated similarly to FTRL, with a subtle difference that the loss estimator here $\hat{\ell}_t(\cdot|b)$ is $\theta_t(b)$ times the loss estimator $\hat{\ell}_t(\cdot)$ in the FTRL algorithm (Line 4 in Algorithm 5).

In Corollary 25, we prove that FTRL_swap satisfies the Assumption 2 with good swap regret bounds. Recall that $B$ is the number of actions, and our normalization condition requires loss $\tilde{l}_t \in [0,1]^B$ for any $t$.

**Corollary 25.** *By choosing hyperparameter* $w_t = \alpha_t\left(\prod_{i=2}^{t}(1-\alpha_i)\right)^{-1}$ *and* $\eta_t = \gamma_t = \sqrt{\frac{H\log B}{t}}$, *FTRL_swap (Algorithm 6) satisfies Assumption 2 with*

$$\xi_{sw}(B,t,\log(1/\delta)) = 10B\sqrt{H\log(B^2/\delta)/t}, \quad \Xi_{sw}(B,t,\log(1/\delta)) = 20B\sqrt{Ht\log(B^2/\delta)}$$

Again, we prove Corollary 25 by showing a more general weighted swap regret guarantee which works for any set of weights $\{w_i\}_{i=1}^{\infty}$ in addition to $\{\alpha_t^i\}_{i=1}^t$. A general weighted swap regret is defined as

$$\mathcal{R}_{\text{swap}}(t) := \min_{\psi\in\Psi}\sum_{i=1}^{t} w_i[\langle\theta_i, l_i\rangle - \langle\psi\diamond\theta_i, l_i\rangle]. \tag{16}$$

**Theorem 26.** *For any $t \leq K$, following Algorithm 6, if $\eta_i \leq 2\gamma_i$ and $\eta_i$ is non-increasing for all $i \leq t$, let $\iota = \log(B^2/\delta)$, then with probability $1 - 3\delta$, we have*

$$\mathcal{R}_{swap}(t) \leq \frac{w_t B\log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\leq t} w_i\iota + B\sum_{i=1}^{t}\gamma_i w_i + B\sqrt{2\iota\sum_{i=1}^{t} w_i^2} + B\max_{i\leq t} w_i\iota/\gamma_t$$

We postpone the proof of Theorem 26 to the end of this section. We show first how Theorem 26 directly implies Corollary 25.

*Proof of Corollary 25.* As shown in the proof of Corollary 19, the weights $\{w_t\}_{t=1}^K$ we choose satisfies a nice property: for any $t$ we have

$$\frac{w_i}{w_j} = \frac{\alpha_t^i}{\alpha_t^j}.$$

Define $\tilde{\mathcal{R}}_{\text{swap}}(t) := \max_{\psi\in\Psi} \sum_{i=1}^t \alpha_t^i[\langle\theta_i, l_i\rangle - \langle\psi\diamond\theta_i, l_i\rangle]$. Plugging our choice of $w_i = \alpha_t \left(\prod_{i=2}^t (1-\alpha_i)\right)^{-1}$ into Theorem 26, we have

$$\begin{aligned}\tilde{\mathcal{R}}_{\text{swap}}(t) \leq& \frac{\alpha_t B\log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^t \eta_i\alpha_t^i + \frac{1}{2}\alpha_t\iota \\ &+ B\sum_{i=1}^t \gamma_i\alpha_t^i + B\sqrt{2\iota\sum_{i=1}^t \left(\alpha_t^i\right)^2} + B\alpha_t\iota/\gamma_t.\end{aligned}$$

By choosing $\eta_t = \gamma_t = \sqrt{\frac{H\log B}{t}}$ and using Lemma 10, we can further upper bound the swap regret by

$$\begin{aligned}\tilde{\mathcal{R}}_{\text{swap}}(t) \leq& \frac{(H+1)\,B\log B}{H+t}\sqrt{\frac{t}{H\log B}} + \frac{3B}{2}\sqrt{H\log B}\sum_{i=1}^t \frac{\alpha_t^i}{\sqrt{t}} \\ &+ \frac{(H+1)\,\iota}{2\,(H+t)} + B\sqrt{2\iota\sum_{i=1}^t \left(\alpha_t^i\right)^2} + B\frac{(H+1)\,\iota}{(H+t)}\sqrt{\frac{t}{H\log B}} \\ \leq& 2B\sqrt{H\frac{\log B}{t}} + 3B\sqrt{\frac{H\log B}{t}} + \frac{H\iota}{t} + 2B\sqrt{\frac{H\iota}{t}} + 2B\sqrt{\frac{H\log B}{t}} \\ \leq& 9B\sqrt{H\iota/t} + H\iota/t.\end{aligned}$$

To further simplify the above upper bound, consider two cases:

- If $H\iota/t \leq 1$, $\sqrt{H\iota/t} \geq H\iota/t$ and thus $\tilde{\mathcal{R}}_{\text{swap}}(t) \leq 10B\sqrt{H\iota/t}$.
- If $H\iota/t \geq 1$, $B\sqrt{H\iota/t} \geq 1 \geq \tilde{\mathcal{R}}_{\text{swap}}(t)$ where the last step is by the definition of $\tilde{\mathcal{R}}_{\text{swap}}(t)$. Therefore we have $\tilde{\mathcal{R}}_{\text{swap}}(t) \leq B\sqrt{H\iota/t}$.

Combine the above two cases, $\tilde{\mathcal{R}}_{\text{swap}}(t) \leq 10B\sqrt{H\iota/t}$.

Finally, we pick $\xi_{\text{sw}}(B, t, \log(1/\delta)) := 10B\sqrt{H\iota/t}$, which is non-decreasing in $B$. On the other hand, since $\sum_{t'=1}^t \xi_{\text{sw}}(B, t, \log(1/\delta)) \leq 20B\sqrt{Ht\iota}$, we choose $\Xi_{\text{sw}}(B, t, \log(1/\delta)) = 20B\sqrt{Ht\iota}$, which is concave in $t$. □

To prove Theorem 26, we again first decompose the swap regret. We first note that by Line 7 of Algorithm 6, we have:

$$w_i\langle\theta_i, l_i\rangle = \sum_{b\in\mathcal{B}} w_i\langle\tilde{\theta}_i(\cdot|b), \theta_i(b)l_i(\cdot)\rangle.$$

On the other hand, by the definition of strategy modification $\Psi$, we have

$$\min_{\psi\in\Psi}\sum_{i=1}^{t} w_i\langle \psi\diamond\theta_i, l_i\rangle = \sum_{b\in\mathcal{B}}\min_{\theta^\star(\cdot|b)}\sum_{i=1}^{t} w_i\theta_i(b)\cdot\langle\theta^\star(\cdot|b), l_i(\cdot)\rangle.$$

Therefore, we have the following decomposition of the swap regret

$$\begin{aligned}
\mathcal{R}_{\text{swap}}(t) :=& \min_{\psi\in\Psi}\sum_{i=1}^{t} w_i[\langle\theta_i, l_i\rangle - \langle\psi\diamond\theta_i, l_i\rangle] = \sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i[\langle\tilde{\theta}_i(\cdot|b) - \theta^\star(\cdot|b), \theta_i(b)l_i\rangle] \\
=& \underbrace{\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\left\langle\tilde{\theta}_i(\cdot|b) - \theta^\star(\cdot|b), \hat{l}_i(\cdot|b)\right\rangle}_{(A)} + \underbrace{\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\left\langle\tilde{\theta}_i(\cdot|b), \theta_i(b)l_i(\cdot) - \hat{l}_i(\cdot|b)\right\rangle}_{(B)} \\
&+ \underbrace{\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\left\langle\theta^\star(\cdot|b), \hat{l}_i(\cdot|b) - \theta_i(b)l_i(\cdot)\right\rangle}_{(C)}
\end{aligned} \tag{17}$$

For the remaining proof, we bound term $(A), (B), (C)$ separately in Lemma 27, Lemma 28, Lemma 29.

**Lemma 27.** *For any $t\in[K]$, suppose $\eta_i\le 2\gamma_i$ for all $i\le t$. The with probability $1-\delta$, for any $\theta^\star$,*

$$\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\left\langle\tilde{\theta}_i(\cdot|b) - \theta^\star(\cdot|b), \hat{l}_i(\cdot|b)\right\rangle \le \frac{w_t B\log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\le t} w_i\iota.$$

*Proof.* Similar to Lemma 22, we have,

$$\begin{aligned}
\sum_{i=1}^{t} w_i\left\langle\tilde{\theta}_i(\cdot|b) - \theta^\star(\cdot|b), \hat{l}_i(\cdot|b)\right\rangle &\le \frac{w_t\log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\eta_i w_i\left\langle\tilde{\theta}_i(\cdot|b), \hat{l}_i^2(\cdot|b)\right\rangle \\
&= \frac{w_t\log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b'\in\mathcal{B}}\eta_i w_i\tilde{\theta}_i(b'|b)\frac{\theta_i^2(b)\tilde{l}_i^2(b_i)\mathbb{I}\{b_i=b'\}}{(\theta_i(b')+\gamma_i)^2} \\
&\le \frac{w_t\log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b'\in\mathcal{B}}\eta_i w_i\frac{\tilde{\theta}_i(b'|b)\theta_i(b)}{\theta_i(b')}\frac{\hat{l}_i(b'|b)}{\theta_i(b)}
\end{aligned}$$

Summing over $b$ and using the fact that $\sum_{b\in\mathcal{B}}\tilde{\theta}_i(b'|b)\theta_i(b) = \theta_i(b')$,

$$\begin{aligned}
\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\left\langle\tilde{\theta}_i(\cdot|b) - \theta^\star(\cdot|b), \hat{l}_i(\cdot|b)\right\rangle &\le \frac{w_t B\log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b'\in\mathcal{B}}\eta_i w_i\frac{\hat{l}_i(b'|b)}{\theta_i(b)} \\
&\overset{(i)}{\le} \frac{w_t B\log B}{\eta_t} + \frac{1}{2}\sum_{i=1}^{t}\sum_{b'\in\mathcal{B}}\eta_i w_i l_i\left(b'\right) + \frac{1}{2}\max_{i\le t} w_i\iota \\
&\le \frac{w_t B\log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\le t} w_i\iota
\end{aligned}$$

where $(i)$ is by using Lemma 21 with $c_i(b) = \eta_i$. Notice the quantity $\frac{\hat{l}_i(b'|b)}{\theta_i(b)}$ actually doesn't depend on $b$, so it is well-defined even after we take the summation with respect to $b$. The any-time guarantee is justified by taking union bound. □

**Lemma 28.** *For any $t \in [K]$, with probability $1-\delta$ ,*

$$\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \theta_i(b) l_i(\cdot) - \hat{l}_i(\cdot|b) \right\rangle \leq B\sum_{i=1}^{t}\gamma_i w_i + B\sqrt{2\iota\sum_{i=1}^{t} w_i^2}.$$

*Proof.* We further decompose it into

$$\sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \theta_i(b) l_i(\cdot) - \hat{l}_i(\cdot|b) \right\rangle = \sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \theta_i(b) l_i(\cdot) - \mathbb{E}_i\hat{l}_i(\cdot|b) \right\rangle + \sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \mathbb{E}_i\hat{l}_i(\cdot|b) - \hat{l}_i(\cdot|b) \right\rangle.$$

The first term is bounded by

$$\begin{aligned}\sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \theta_i(b) l_i(\cdot) - \mathbb{E}_i\hat{l}_i(\cdot|b) \right\rangle &= \sum_{i=1}^{t} w_i\theta_i(b) \left\langle \tilde{\theta}_i(\cdot|b), (1 - \frac{\theta_i(\cdot)}{\theta_i(\cdot)+\gamma_i}) l_i(\cdot) \right\rangle \\ &= \sum_{i=1}^{t} w_i\theta_i(b) \left\langle \tilde{\theta}_i(\cdot|b), \frac{\gamma_i}{\theta_i(\cdot)+\gamma_i} l_i \right\rangle.\end{aligned}$$

So by taking the sum with respect to $b$, we have

$$\begin{aligned}\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \theta_i(b) l_i(\cdot) - \mathbb{E}_i\hat{l}_i(\cdot|b) \right\rangle &\leq \sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i\theta_i(b) \left\langle \tilde{\theta}_i(\cdot|b), \frac{\gamma_i}{\theta_i(\cdot)+\gamma_i} l_i \right\rangle \\ &\leq \sum_{b'\in\mathcal{B}}\sum_{i=1}^{t} w_i\gamma_i l_i(b') \\ &\leq B\sum_{i=1}^{t}\gamma_i w_i.\end{aligned}$$

To bound the second term, notice $\tilde{\theta}_i(b'|b)\theta_i\,(b) \leq \theta_i(b')$ for any $b, b' \in \mathcal{B}$,

$$\left\langle \tilde{\theta}_i(\cdot|b), \hat{l}_i(\cdot|b) \right\rangle \leq \sum_{b'\in\mathcal{B}} \tilde{\theta}_i(b'|b)\theta_i\,(b) \frac{\mathbb{I}\left\{b_t = b'\right\}}{\theta_i(b') + \gamma_i} \leq \sum_{b'\in\mathcal{B}} \mathbb{I}\left\{b_i = b'\right\} = 1,$$

thus $\{w_i \left\langle \tilde{\theta}_i(\cdot|b), \mathbb{E}_i\hat{l}_i(\cdot|b) - \hat{l}_i(\cdot|b) \right\rangle\}_{i=1}^{t}$ is a bounded martingale difference sequence w.r.t. the filtration $\{\mathcal{F}_i\}_{i=1}^{t}$. By Azuma-Hoeffding,

$$\sum_{i=1}^{t} w_i \left\langle \tilde{\theta}_i(\cdot|b), \mathbb{E}_i\hat{l}_i(\cdot|b) - \hat{l}_i(\cdot|b) \right\rangle \leq \sqrt{2\iota\sum_{i=1}^{t} w_i^2}.$$

The proof is completed by taking the summation with respect to $b$ and a union bound. □

**Lemma 29.** *For any $t \in [K]$, suppose $\gamma_i$ is non-increasing in $i$, then with probability $1-\delta$, and any $\theta^\star$,*

$$\sum_{b\in\mathcal{B}}\sum_{i=1}^{t} w_i \left\langle \theta^\star(\cdot|b), \hat{l}_i(\cdot|b) - \theta_i(b) l_i(\cdot) \right\rangle \le B \max_{i\le t} w_i\iota/\gamma_t.$$

*Proof.* The proof follows from Lemma 24 and taking the summation with respect to $b$. □

Finally, we are ready to prove Theorem 26.

*Proof of Theorem 26.* Recall the decomposition of swap regret (17). We bound $(A)$ in Lemma 27, $(B)$ in Lemma 28 and $(C)$ in Lemma 29. Putting everything together, we have

$$\mathcal{R}_{\text{swap}}(t) \le \frac{w_t B\log B}{\eta_t} + \frac{B}{2}\sum_{i=1}^{t}\eta_i w_i + \frac{1}{2}\max_{i\le t} w_i\iota + B\sum_{i=1}^{t}\gamma_i w_i + B\sqrt{2\iota\sum_{i=1}^{t} w_i^2} + B\max_{i\le t} w_i\iota/\gamma_t.$$

□