Impact of Decentralized Learning on Player Utilities in Stackelberg Games*

Kate Donahue[1], Nicole Immorlica[2], Meena Jagadeesan[3], Brendan Lucier[2], and Aleksandrs Slivkins[2]

[1]*Cornell*
[2]*Microsoft Research*
[3]*University of California, Berkeley*

Abstract

When deployed in the world, a learning agent such as a recommender system or a chatbot often repeatedly interacts with another learning agent (such as a user) over time. In many such two-agent systems, each agent learns separately and the rewards of the two agents are not perfectly aligned. To better understand such cases, we examine the learning dynamics of the two-agent system and the implications for each agent's objective. We model these systems as Stackelberg games with decentralized learning and show that standard regret benchmarks (such as Stackelberg equilibrium payoffs) result in worst-case linear regret for at least one player. To better capture these systems, we construct a relaxed regret benchmark that is tolerant to small learning errors by agents. We show that standard learning algorithms fail to provide sublinear regret, and we develop algorithms to achieve near-optimal $\mathcal{O}(T^{2/3})$ regret for both players with respect to these benchmarks. We further design relaxed environments under which faster learning ($\mathcal{O}(\sqrt{T})$) is possible. Altogether, our results take a step towards assessing how two-agent interactions in sequential and decentralized learning environments affect the utility of both agents.

1 Introduction

When learning agents such as recommender systems or chatbots are deployed into the world, these learning agents often repeatedly interact with other learning agents (such as humans). For example, a recommender system—through repeated interactions with a user—learns which content to suggest to the user, while the user simultaneously learns their own preferences over content (Example 2.2). As another example, a chatbot such as ChatGPT—during a chat session—can iteratively refine its generated content to the user's stylistic preferences, while the user (or prompt engineering agent) simultaneously learns how to best interact with the chatbot (Example 2.1).

These two-agent systems—among many others—share the following structural features. The environments are *decentralized* (both agents operate autonomously, without central coordination of their actions). Furthermore, the environments are *sequential* (one agent always chooses their action first[1]) and *misaligned* (the agents can obtain different utilities for the same pair of actions[2]).

*Authors in alphabetical order. Part of this work was conducted while KD and MJ were at Microsoft Research.

[1]E.g., a recommender system recommends a slate of items and the user picks among them; in a chatbot session, the user picks a prompt that the LLM responds to.

[2]Misalignment could arise from fundamental differences in agent preferences: the engagement metrics of recommender systems rarely align with user welfare (e.g., [Milli et al., 2021]); or a chatbot might be trained to optimize

Finally, the environments exhibit *learning* (both agents learn from repeated interactions about which actions to take). For such two-agent environments, core questions of interest include: *how quickly does this two-agent system learn, and what are the implications for each agent's objective?*

In the absence of learning, the interaction between misaligned agents taking sequential actions is formalized by Stackelberg games. In this setting, the leader chooses an action first and the follower chooses an action to respond. The two agents are allowed to have distinct utility functions over pairs of actions. The standard benchmark is the *Stackelberg equilibrium*, where the leader picks the best action they can, assuming that the follower will pick their own best response. However, this classical solution concept is tailored to the full-information setting where both players know their own utilities and the leader knows how the follower will best respond; in fact, the static Stackelberg game framework breaks down when agents instead must *learn* these utilities from noisy feedback.

Our focus is on a decentralized Stackelberg learning environment. In this setting, the leader and the follower repeatedly interact and each make decisions about which actions to take, where each agent only observes their own realized stochastic rewards. In this environment, it is natural to model each player's learning process as a multi-armed bandit algorithm[3] which learns over time which arms (actions) to pull. A unique feature of this sequential two-player learning environment is that agents must learn in two separate ways—first, both agents learn their own (fixed) preferences from stochastic observations, and secondly, the leader needs to learn and adapt to the follower's (evolving) responses to the leader's actions—which complicates the design of learning algorithms.

In this paper, we initiate the study of how this learning environment impacts *both* the leader and follower's utility, motivated by how both objectives are of societal interest in natural real-world settings (see Example 2.1 and Example 2.2). Rather than only focusing on the regret of the leader as is typical in learning in Stackelberg games, we thus examine the *maximum regret* of the two agents. Our main contributions are to design appropriate benchmarks for each agent and to construct algorithms which achieve near-optimal regret against these benchmarks. Our results apply to the most general setting which allows for *arbitrary relationships between the two player's utilities*.

- **Linear regret for Stackelberg benchmarks:** We first show that the player utilities in the Stackelberg equilibrium are fundamentally unachievable and necessarily lead to linear regret for at least one agent (Theorem 3.1).

- **Relaxed benchmarks:** The possibility of linear regret motivates us to design relaxed benchmarks which are more tolerant to the other agent being suboptimal. We thus define γ*-tolerant benchmarks* (Definition 4.1), which account for incomplete learning: the benchmark captures an agent's worst-case utility if the other agent is up to γ-suboptimal.[4]

- **Regret bounds:** Using the γ-tolerant benchmarks, we construct algorithms for the leader and follower that achieve $O(T^{2/3})$ regret (Theorem 4.5). Surprisingly, this dependence on T is unavoidable, and *any* pair of algorithms achieves $\Omega(T^{2/3})$ regret (Theorem 4.6). Nonetheless, under relaxed settings—either with a weaker benchmark or when players agree on which pairs of actions are meaningfully different[5]—we show that faster learning (i.e., $O(\sqrt{T})$ regret) is possible (Section 5).

societal preferences or cultural norms (e.g., avoiding violent language) which conflict with individual user preferences (e.g., [Bakker et al., 2022]). Misalignment could also arise from misspecification, if the metrics that the AI system optimizes do not perfectly capture user preferences (e.g., [Zhuang and Hadfield-Menell, 2020]).

[3]See Slivkins [2019], Lattimore and Szepesvári [2020] for textbook treatments of multi-armed bandits.

[4]Section 4 describes our benchmark and γ in greater detail, and Section 1.1 compares our benchmarks to prior work.

[5]We formalize this as a continuity requirement on the utilities (Section 5). This requirement allows players to be

From an algorithmic perspective, our results provide insight into which bandit algorithms for the leader allow for low regret for both players. Out-of-box stochastic algorithms do not provide this guarantee: for example, both agents choosing `ExploreThenCommit` can lead to linear regret even for the γ-tolerant benchmarks (Proposition 4.3). The intuition is that since the follower's actions can change between time steps, the leader is not operating in a stochastic environment; as a result, the follower's exploration phase can distort the leader's learning. This motivates us to design algorithms where *the leader waits for the follower to partially converge before starting to learn*: `ExploreThenCommitThrowOut` (Algorithm 2) and `ExploreThenUCB` (Algorithm 3). The more sophisticated of these two algorithms, `ExploreThenUCB` (Algorithm 3), guarantees a $T^{2/3}$ regret bound when the follower applies any algorithm with certain properties (i.e., high-probability instantaneous regret bounds) (Theorem 4.5). We then consider two relaxed environments where the leader no longer needs to worry about being overly distorted by the follower; in these environments, *the leader can start learning before the follower has partially converged*, which enables $O(\sqrt{T})$ regret bounds (Theorems 5.1 and 5.3).

More broadly, our work takes a step towards assessing the utility of *both learning agents* in decentralized, misaligned environments. Our model and results capture the general setting where the player utilities to be arbitrarily related, where players might not even agree upon which pairs of actions give similar or different rewards. This motivated us to design benchmarks which are tolerant to small errors in the other player. We hope that our benchmarks and algorithms serve as a starting point for assessing when two-agent learning systems in misaligned environments can ensure high utility for both agents.

1.1 Related Work

Most closely related is the work on learning in Stackelberg games (SGs) where both players incur stochastic rewards. Bai et al. [2021], Gan et al. [2023] focus on the centralized setting where the learner controls the actions and observes the rewards of both players; in contrast, we study a decentralized setting where each player controls their own actions and only observes their own rewards. Nonetheless, the benchmarks proposed in these papers are related to the γ-tolerant benchmarks that we consider, but with some key differences. For the leader's utility, their benchmark is equivalent to our γ-tolerant benchmark with a fixed value of ϵ (rather than an inf over $\epsilon \leq \gamma$ with a regularizer). For the follower's utility, their benchmark only ensures ϵ-optimality with respect to the leader's selected action; in contrast, we consider a different style of follower benchmark that is more conceptually similar to the benchmark for the leader. Also, we study regret, whereas they study the speed of convergence.

Several papers study *decentralized* online learning in SGs. Camara et al. [2020], Collina et al. [2023b] posit that the follower runs a no-counterfactual-internal-regret algorithm and design no-regret algorithms for the leader. However, they assume *strong alignment* between the players' rewards Camara et al. [2020] requires that a follower choosing an ϵ-suboptimal action only results in an $O(\epsilon)$ utility loss for the leader.[6] Collina et al. [2023b] partially relax this assumption, but still require the existence of *stable* actions for the leader. In contrast, we do not place any alignment conditions: in fact, misalignement is the driver of our linear regret result for the original Stackelberg benchmarks (Theorem 3.1). Other differences are that we focus on stochastic, rather than adversarial, rewards, and our benchmark is independent of the follower's choice of learning

misaligned (e.g. different preferences), but requires them to agree on which outcomes are *substantially different* from each other.

[6] See Assumption 2 in Camara et al. [2020]. Appendix D therein considers some relaxations, but they lead to $\Omega(T)$ worst-case regret.

algorithm.[7] Haghtalab et al. [2023] take a different perspective and consider the follower running a *calibrated* algorithm. They design a leader algorithm which waits for the follower to partially converge, and show that the Stackelberg value is obtained in the limit as $T \to \infty$. In contrast, we focus on *instance-independent regret bounds* for a fixed time horizon T, which requires us to relax the benchmark. Other differences are we focus on stochastic, rather than deterministic, rewards, we assume the follower observes the leader's action, and we consider the follower's utility in addition to the leader's utility.

The literature on learning in SGs is vast and includes many other variations. Many works (e.g., [Letchford et al., 2009, Balcan et al., 2015, Zhao et al., 2023]) consider the leader performing (offline or) online learning and followers myopically best-responding. Other model variants studied include the leader strategizing against a follower who is running a no-regret learning algorithm [Braverman et al., 2018, Deng et al., 2019, Guruganesh et al., 2024, Brown et al., 2023], the leader and follower both running gradient-based algorithms [Fiez et al., 2019], non-myopic followers who best-respond to a discounted utility over future time steps [Haghtalab et al., 2022, Hajiaghayi et al., 2023], repeated game formulations under complete information [Zuo and Tang, 2015, Collina et al., 2023a] the leader offering a menu of actions to the follower [Han et al., 2023], and both players having side information [Harris et al., 2024]. Other works have studied learning in structured SGs, including delegated choice (e.g., [Kleinberg and Kleinberg, 2018, Hajiaghayi et al., 2023]), strategic classification (e.g., [Dong et al., 2018, Chen et al., 2020, Zrnic et al., 2021, Ahmadi et al., 2021]), pricing under buyer and seller uncertainty (e.g., [Guo et al., 2023]), contract theory (e.g., [Zhu et al., 2023]), and aligned utilities (e.g., [Kao et al., 2022]).

Our work also connects to a broader literature on interacting learners. This literature examines interactions between *multiple bandit learners*, studying aspects such as the convergence of systems of no-regret learners to coarse correlated and correlated equilibrium (e.g. [Daskalakis et al., 2011, 2021, Anagnostides et al., 2022]), multiple bandit learners competing for market share (e.g., [Aridor et al., 2020, Jagadeesan et al., 2023]), and multiple autobidding algorithms competing in an auction (e.g., [Borgs et al., 2007, Balseiro and Gur, 2019, Lucier et al., 2023]). Most closely related to this paper is *corralling bandit algorithms* (e.g., [Agarwal et al., 2017, Pacchiano et al., 2020]), where a "master algorithm" dynamically chooses among several "base algorithms": our decentralized learning environment in the case of aligned player utilities is essentially an instance of corralling bandits, with the "base algorithms" corresponding to different leader actions. The interacting learner literature also examines *human-algorithm collaboration* studying aspects such as misalignment between engagement metrics and user welfare (e.g., [Ekstrand and Willemsen, 2016, Milli et al., 2021, Stray et al., 2021, Kleinberg et al., 2022]), impact of underspecification on human-AI misalignment (e.g., [Zhuang and Hadfield-Menell, 2020]), and "assistive" algorithmic tools (e.g. [Chan et al., 2019]). Most closely related to this paper is work on online learning in subset selection and conformal prediction Straitouri and Rodriguez [2023], Corvelo Benz and Rodriguez [2024], Straitouri et al. [2023], Wang et al. [2022], Donahue et al. [2024], Agarwal and Brown [2023], often with the goal of achieving complementarity [Bansal et al., 2021]. The related area of *human-AI interaction* (see [Preece et al., 1994, Kim, 2015, MacKenzie, 2024, Lazar et al., 2017] for textbook treatments) studies similar questions, often from a more behavioral angle. More broadly, the interacting learner literature also studies applied domains including *multi-agent reinforcement learning* (see Zhang et al. [2019] for a survey) and *federated learning* (see Yang et al. [2019] for a survey).

[7] However, our regret bounds assume that the follower's algorithm gracefully improves over time, see Section 2.3.

2 Model and assumptions

In this section we describe our formal model. We first define an instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ in our setup, which captures the setup of the underlying static Stackelberg game. Let $\mathcal{A}$ be the action set for the leader (Player 1) and let $\mathcal{B}$ be the action set for the follower (Player 2). Let $v_i(a, b) \in [0, 1]$ denote Player i's value (i.e., mean reward) for the leader choosing a and the follower choosing b. The Stackelberg equilibrium takes the following form. Let $b^*(a)$ be the best-response with respect to the follower's rewards:[8] $b^*(a) = \text{argmax}_{b \in \mathcal{B}} v_2(a, b)$. The Stackelberg equilibrium (a^*, b^*) is defined to be the best action the leader can take, assuming that the follower will exactly best-respond:

$$a^* = \text{argmax}_{a \in \mathcal{A}} v_1(a, b^*(a)) \text{ and } b^* = b^*(a^*).$$

In this paper, we move from the static Stackelberg Equilibrium environment to a repeated dynamic environment over T time steps where each player selects actions using a multi-armed bandit algorithm. The environment is *sequential*: at each time step t, the leader chooses an action $a_t \in \mathcal{A}$ and then the follower chooses an action $b_t \in \mathcal{B}$. This environment is also *decentralized*: each player i can observe their own stochastic rewards but not the stochastic rewards of the other player. We measure *regret* for each player i by their cumulative reward across all time steps relative to a benchmark.

2.1 Interaction between players

The interaction between the leader and follower proceeds as follows. The leader chooses an algorithm $\texttt{ALG}_1$ mapping their history (formalized below) of observed actions and rewards to a distributions over actions $\mathcal{A}$, and the follower similarly chooses an algorithm $\texttt{ALG}_2$ mapping the leader's action and their history to a distribution over actions $\mathcal{B}$. After the players select algorithms, the interaction between the leader and the follower proceeds as follows at each time step t:

1. The leader chooses action $a_t \sim \texttt{ALG}_1(H_{1,t})$ as a function of their history $H_{1,t}$ and reveals a_t to the follower.

2. After observing a_t, the follower chooses action $b_t \sim \texttt{ALG}_2(a_t, H_{2,t})$ as a function of their history $H_{2,t}$.

3. Players 1 and 2 incur stochastic rewards $r_{1,t}(a_t, b_t) \sim \mathcal{N}(v_1(a_t, b_t), 1)$ and $r_{2,t}(a_t, b_t) \sim \mathcal{N}(v_2(a_t, b_t), 1)$. The noise distribution is Gaussian with unit variance[9].

This interaction captures that the players are *dynamic* in their learning: in particular, this framework is sufficiently general to capture a wide range of learning strategies. However, we do not study the *meta-game* where agents strategically pick learning algorithms.[10] We believe that this model captures many real-world environments such as user-chatbot interactions and recommender system-user interactions, where agents learn about their incurred rewards from past interactions even if they do not actively optimize the higher-order manner in which they learn. See Section 2.4 (Example 2.1 and Example 2.2) for more details of how these real-world examples are captured by our model.

[8] Different variants (e.g. Strong Stackelberg Equilibrium and Weak Stackelberg Equilibrium) handle exact ties in slightly different ways. We take $b^*(a)$ to be the best-response with lowest utility for the leader.

[9] We assume the reward distributions are independent across time steps and players. We make the Gaussian assumption for simplicity, and we expect that our results would likely extend to subgaussian Bernoulli distributions.

[10] An example of such a meta-game is Kolumbus and Nisan [2022], where each player maximizes their worst-case reward against a class of learning algorithms that the other player could choose.

Information structures. Having described how the players interact, we next discuss the players' history, which further enforces decentralization. Each player i can only observe their own reward $r_{i,t}(a_t, b_t)$ (and cannot observe the reward of the other player). In the *strongly decentralized setting*, the follower can observe the leader's action a_t, but the leader cannot observe the follower's action b_t, whereas in the *weakly decentralized setting*, the leader can additionally observe the follower's action b_t. One motivation for the strongly decentralized setting is interpretability: the leader and follower may be taking actions in spaces that are not mutually understandable (e.g. a chatbot's representation of human preferences may not be interpretable). Notation for each player's histories are presented in Appendix B.

2.2 Measuring regret

As is typical in multi-armed bandits, we measure performance by the *regret* of each player with respect to a benchmark, where higher benchmarks make learning more challenging (further detail on benchmarks is in Sections 3 and 4). For each player $i \in \{1, 2\}$, given a benchmark β_i, we define the (expected) regret of Player i on instance $\mathcal{I}$ to be:

$$R_i(T; \mathcal{I}) = \beta_i \cdot T - \left(\sum_{t=1}^{T} \mathbb{E}[r_{i,t}(a_t, b_t)] \right)$$

where the expectation is over randomness in the algorithm and in the stochastic rewards. Given action sets $\mathcal{A}$ and $\mathcal{B}$, we let $R_i(T)$ denote the worst-case regret across all value functions v_1 and v_2 on instances of the form $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$.

Our goal is to obtain sublinear worst-case regret for *both* players: that is, we will assess $\max(R_1(T), R_2(T))$. We note that this challenging goal is a departure from previous work which has typically focused solely on sublinear regret for the leader (see Section 1.1). Our motivation for selecting this objective is that (a) a human could be either the leader or the follower, and (b) the societal welfare may demand that we care about the utility of both the leader and the follower (discussed further in Section 2.4).

2.3 Assumptions on the follower's algorithm $\texttt{ALG}_2$

When we analyze regret in Sections 4-5, many of our constructions do not require that the follower run a particular algorithm, but instead allow the follower to run any algorithm that has sufficiently good performance along certain fine-grained performance metrics that capture the extent to which an algorithm's performance gracefully improves over time.

These fine-grained performance metrics capture the follower's errors while learning. These errors are captured by the difference between $v_2(a_t, b_t)$ (the follower's realized mean reward) and the $\max_{b \in \mathcal{B}} v_2(a_t, b)$ (the best mean reward that the follower could achieve for the leader's action a_t). Note that this measure of suboptimality $\max_{b \in \mathcal{B}} v_2(a_t, b) - v_2(a_t, b_t)$ captures how well the follower is best-responding to the leader's action. This differs from our main notion of regret in Section 2.2, which captures the follower's level of discontent relative to a fixed benchmark.

For intuition, we first describe these performance metrics—*high-probability instantaneous regret* and *high-probability anytime regret*—for a typical single-bandit learner which acts in isolation. For the single learner setting, instances $\mathcal{I} = (\mathcal{C}, v)$ capture a single action set and a single value function. *High-probability instantaneous regret* measures the suboptimality of the arms that the algorithm pick arms at each time step. More formally, an algorithm acting over an action space $\mathcal{C}$ satisfies

high-probability instantaneous regret g if for any instance $\mathcal{I} = (\mathcal{C}, v)$:

$$\mathbb{P}\left[\forall t \in [T] \mid v(c_t) \geq \max_{c \in \mathcal{C}} v(c) - g(t, T, \mathcal{C})\right] \geq 1 - T^{-3},$$

where the probability captures randomness in the algorithm and in the stochastic rewards. A *high-probability anytime regret* bound guarantees that the regret bound for the algorithm holds with high probability at every time step t. More formally, an algorithm acting over an action space $\mathcal{C}$ satisfies high-probability anytime regret bound h if for any instance $\mathcal{I} = (\mathcal{C}, v)$, it holds that:

$$\mathbb{P}\left[\forall t \in [T] \mid \sum_{t' \leq t} v(c_{t'}) \geq \sum_{t' \leq t} \max_{c \in \mathcal{C}} v(c) - h(t, T, \mathcal{C})\right] \geq 1 - T^{-3},$$

where the randomness is over the algorithm.[11]

In our setting, we will require similar properties to hold for the follower's algorithm $\texttt{ALG}_2$, but we take account how the algorithm $\texttt{ALG}_2$ depends on the action a_t which is selected by the leader's algorithm $\texttt{ALG}_1$. Let $n_{t+1}(a)$ be the number of times that arm a has been pulled up prior to the $(t+1)$th time step. An algorithm $\texttt{ALG}_2$ satisfies a high-probability instantaneous regret bound of g if for any $\texttt{ALG}_1$ chosen by the leader and any $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$, it holds that:

$$\mathbb{P}\left[\forall t \in [T], a \in \mathcal{A} \mid v_2(a_t, b_t) \geq \max_{b \in \mathcal{B}} v_2(a_t, b) - g(n_{t+1}(a) + 1, T, \mathcal{B})\right] \geq 1 - |\mathcal{A}| \cdot T^{-3},$$

An algorithm $\texttt{ALG}_2$ satisfies a high-probability anytime regret bound of h if for any $\texttt{ALG}_1$ chosen by the leader and any instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$, it holds that:

$$\mathbb{P}\left[\forall t \in [T], a \in \mathcal{A} \mid \sum_{t' \leq t | a_{t'} = a} v_2(a, b_{t'}) \geq \sum_{t' \leq t | a_{t'} = a} \max_{b \in \mathcal{B}} v_2(a, b) - h(n_{t+1}(a), T, \mathcal{B})\right] \geq 1 - |\mathcal{A}| \cdot T^{-3},$$

We discuss these performance metrics in more detail (including their connection to each other) and analyze the performance of standard algorithms along these metrics in Section 7. As an example, if the follower runs a separate instantiation of `ActiveArmElimination` (Algorithm 7) on every arm $a \in \mathcal{A}$, this satisfies high-probability instantaneous regret $g(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot \log T / t})$ and high-probability anytime regret $h(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot t \cdot \log T})$ (Proposition 7.1).

2.4 Real-world examples

We describe two real-world examples which fit into our framework.

Example 2.1 (User-chatbot interaction)**.** Consider user-chatbot interactions where the user (e.g., a human or a prompt engineer) selects a prompt and the chatbot (e.g., an LLM-based application such as chatGPT) selects a response. We model the user as the leader and the chatbot as the follower: the user picks a (perhaps high-level) prompt or prompt engineering technique $a \in \mathcal{A}$, and the chatbot picks a response or style of response $b \in \mathcal{B}$. Repeated interactions may occur within a single chatbot session, such as with ChatGPT, where sessions can be resumed when the user logs in at a later time. An example of such an interaction is where the user repeatedly asks for help with similar queries (e.g. content generation or help with technical tasks) and learns better prompt

[11] Compared with standard definitions of instaneous and anytime regret, we require a high-probability bound (rather than expectation). For anytime regret, we also require the bound for all t for a given T (rather than for all T).

engineering techniques [Chen et al., 2023], while the chatbot learns how to best respond to this user by using the session history as its context [Hong et al., 2023, Pan et al., 2024]. The user and chatbot may have misaligned rewards for each prompt-output pair: this misalignment could arise from fundamental differences in preferences if the chatbot is trained to optimize societal preferences or cultural norms (e.g., avoiding violent language) which conflict with individual user preferences [Bakker et al., 2022]. Misalignment could also arise from unintentional misspecification if chatbot optimizes a metric which does not fully capture user preferences (e.g., if the user has an imperfect ability to communicate preferences [Zhuang and Hadfield-Menell, 2020]).

Example 2.2 (User-recommender system interaction)**.** Consider interactions between a recommender system and a user, where recommender system gives a slate (or subset) of items $a \in \mathcal{A}$ to the user, and the user picks an action $b \in \mathcal{B}$ from the slate. When the user returns to the same content recommendation system (e.g. a Netflix/Hulu user with a profile) many times, this becomes a repeated game where both the recommendation system and user learn about their preferences [Hajiaghayi et al., 2023]. Again, misalignment could occur from the engagement metric being misaligned with user welfare [Milli et al., 2021] or for unintentional reasons (e.g., misspecification due to discrete *thumbs up/thumbs down* user feedback, since true preferences are more nuanced).

Examples 2.1-2.2 motivate why our objective is to minimize regret for both the leader and the follower. First, we may inherently care about utility for the human, who could be either the leader (Example 2.1 in Section 2.4) or the follower (Example 2.2 in Section 2.4). Secondly, we may also care about utility for the algorithmic tools: for example, a recommendation system that fails to make money may go out of business, or in certain cases, the chatbot/recommender may better capture societal objective than certain humans.

3 Stackelberg value is unachievable

In this section, we show that the natural benchmark given by the players' utilities in the underlying static Stackelberg game is unachievable. More formally, given an instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$, let (a^*, b^*) be the Stackelberg equilibrium. We define the *Stackelberg benchmarks* to be each player's utility at (a^*, b^*), that is: $\beta_1^{\text{orig}} = v_1(a^*, b^*)$ and $\beta_2^{\text{orig}} = v_2(a^*, b^*)$ (where the superscript "orig" denotes that this is the benchmark for original offline Stackelberg games). The following result illustrates that it is not possible to simultaneously achieve sublinear regret with respect to both players' regret.[12]

Theorem 3.1. *Consider any algorithms* $\texttt{ALG}_1$ *and* $\texttt{ALG}_2$ *which operate in either the strongly decentralized setting or the weakly decentralized setting. There exists an instance* $\mathcal{I}^*$ *with* $|\mathcal{A}| = |\mathcal{B}| = 2$ *where at least one of the players incurs linear regret with respect to the Stackleberg benchmarks* β_1^{orig} *and* β_2^{orig}*, that is:* $\max(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(T)$.

Proof sketch of Theorem 3.1. It suffices to prove this lower bound in a *centralized* environment where a single learner can choose action pairs (a, b) and observes rewards for both players (Lemma C.1). We show that on the instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$ in Table 1 (with $\delta = O(1/\sqrt{T})$), at least one of the players incurs linear regret on at least one of the instances. The small value of δ means that the algorithm cannot distinguish between these instances with constant probability. Nonetheless, the benchmarks are very different: on instance $\mathcal{I}$, $(a^*, b^*) = (a_1, b_1)$, $\beta_1^{\text{orig}} = 0.6$ and $\beta_2^{\text{orig}} = \delta > 0$; on

[12]There exists a simple algorithm in the centralized environment that achieves sublinear regret for Player i: run a sublinear regret multi-armed bandit algorithm on the arms (a, b) using Player i's stochastic rewards (ignoring the rewards of the other player).

instance $\tilde{\mathcal{I}}$, $(a^*, b^*) = (a_2, b_1)$, $\beta_1^{\text{orig}} = 0.5$, and $\beta_2^{\text{orig}} = 0.6$. Intuitively, when the algorithm fails to distinguish between these instances, then it must choose the same distribution over $\mathcal{A} \times \mathcal{B}$ on both $\mathcal{I}$ and $\tilde{\mathcal{I}}$. However, any such distribution either incurs constant loss for the leader on $\mathcal{I}$ or constant loss for the follower on $\tilde{\mathcal{I}}$. We formalize this proof in Appendix C.4. □

The linear regret in Theorem 3.1 is driven by *misalignment* between the leader's utilities and the follower's utilities: small differences in the follower's utilities can lead to arbitrarily large differences in the leader's utilities. As a result, the suboptimal actions that the follower takes while learning are amplified in the leader's regret. This motivates the design of relaxed benchmarks that take into account the suboptimal actions players inevitably take while learning.

	b_1	b_2
a_1	$(0.6, \delta)$	$(0.2, \mathbf{0})$
a_2	$(0.5, 0.6)$	$(0.4, 0.4)$

(a) Mean rewards $(v_1(a,b), v_2(a,b))$ for $\mathcal{I}$

	b_1	b_2
a_1	$(0.6, \delta)$	$(0.2, \mathbf{2\delta})$
a_2	$(0.5, 0.6)$	$(0.4, 0.4)$

(b) Mean rewards $(\tilde{v}_1(a,b), \tilde{v}_2(a,b))$ for $\tilde{\mathcal{I}}$

Table 1: Two instances $\mathcal{I}$ (left) and $\tilde{\mathcal{I}}$ (right), which differ solely in the follower's reward for (a_1, b_2) (shown in **bold**). For δ sufficiently small, the instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$ are hard to distinguish and turn out to imply a $\Omega(T)$ lower bound on regret with respect to the original Stackelberg benchmarks (Theorem 3.1).

4 γ-tolerant benchmark and regret bounds

Having shown that the Stackelberg equilibrium is unattainable, we next propose a novel benchmark and give tight sublinear regret bounds with respect to it.

4.1 γ-tolerant benchmark

Our benchmark is related to the Stackelberg Equilibrium, but adapted to account for the fact that both players are learning and cannot be counted on to exactly best respond. This benchmark is a function of the instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ but *independent* of the learning algorithms for either player. At a high-level, we construct a set of *approximate best responses* for each player and use these sets to construct more realistic benchmarks; within these sets, our benchmark will be *tolerant* to suboptimality with respect to the other player.

If the leader takes action a, then we define the follower's ϵ-best-response set $\mathcal{B}_\epsilon(a)$ as:

$$\mathcal{B}_\epsilon(a) := \{b \in \mathcal{B} \mid v_2(a,b) \geq \max_{b' \in \mathcal{B}} v_2(a,b') - \epsilon\}.$$

Defining the ϵ-best response set $\mathcal{A}_\epsilon$ for the leader is more subtle. Informally, we define this set to include any action $a \in \mathcal{A}$ which has "any chance" of doing at least as well as the the leader's best action if the follower is ϵ-best responding. Specifically, this includes actions a where *some* action $b \in \mathcal{B}_\epsilon(a)$ achieves utility close to $\max_{a' \in \mathcal{A}} \min_{b' \in \mathcal{B}_\epsilon(a)} v_1(a', b')$ for the leader:

$$\mathcal{A}_\epsilon = \{a \in \mathcal{A} \mid \max_{b \in \mathcal{B}_\epsilon(a)} v_1(a,b) \geq \max_{a' \in \mathcal{A}} \min_{b' \in \mathcal{B}_\epsilon(a')} v_1(a', b') - \epsilon\}$$

We use these ϵ-best-response sets to create the relaxed benchmarks for the leader and follower. In these benchmarks, we add an *ϵ-relaxed Stackelberg utility* term with a *ϵ-regularizer* term, and then

take an infimum over all possible values $\epsilon \leq \gamma$. In particular, the ϵ-relaxed Stackelberg utility takes a max over the player's actions and a min over the other player's ϵ-best response set; the regularizer adds a ϵ penalty for errors made by the other player.

Definition 4.1. Given a maximum tolerance $\gamma > 0$, we define the *γ-tolerant benchmarks* β_1^{tol} and β_2^{tol} to be:

$$\beta_1^{\text{tol}} = \inf_{\epsilon \leq \gamma} \Big(\underbrace{\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_\epsilon(a)} v_1(a,b)}_{\epsilon\text{-relaxed Stackelberg utility}} + \underbrace{\epsilon}_{\epsilon\text{-regularizer}} \Big)$$

$$\beta_2^{\text{tol}} = \inf_{\epsilon \leq \gamma} \Big(\underbrace{\min_{a \in \mathcal{A}_\epsilon} \max_{b \in \mathcal{B}} v_2(a,b)}_{\epsilon\text{-relaxed Stackelberg utility}} + \underbrace{\epsilon}_{\epsilon\text{-regularizer}} \Big).$$

We provide some high-level intuition for why our benchmark might be appropriate for learning environments. For small values of ϵ, the ϵ-best-response sets describe actions that (from the player's perspective) are similar and difficult to distinguish between while learning. The ϵ-relaxed Stackelberg utility takes a worst-case perspective and takes a minimum over the other player's ϵ-best-response set, since the player's choices within this set can be unpredictable while learning.[13] The ϵ-regularizer captures the player's intolerance of suboptimality of the other player (see Section 6 for a discussion of regularizers and γ).

We illustrate these benchmarks in the following example.

Example 4.2. Consider the example in Table 2 (with $0.4 > \gamma \geq 4\delta$). In this case, the leader's benchmark is equal to the Stackelberg utility ($\beta_1^{\text{orig}} = \beta_1^{\text{tol}} = 0.5+\delta$), while the *follower's* benchmark is weaker ($\beta_2^{\text{orig}} = 0.4 > 3\delta+\delta = \beta_2^{\text{tol}}$), where the second δ comes from the regularizer. The intuition is that the leader's ϵ-best-response set $\mathcal{A}_\delta = \{a_1, a_2\}$ contains both actions, even though a_2 is not a Stackelberg equilibrium, which noticeably lowers the follower's ϵ-relaxed Stackelberg utility. In Appendix A, we provide more detailed discussions of examples.

	b_1	b_2
a_1	$(0.5+\delta, 0.4)$	$(0.2, 0)$
a_2	$(0.5, 3\delta)$	$(0.4, 2\delta)$

Table 2: A single instance, illustrating the γ-tolerant benchmark (Example 4.2).

4.2 Linear regret for ExploreThenCommit

We first show that out-of-box stochastic bandit algorithms do not directly provide sublinear regret against the γ-tolerant benchmark, where the challenge is that the leader's learning gets distorted when both players simultaneously learn. To demonstrate this, we consider `ExploreThenCommit` (Algorithm 1) and show that if both the leader and follower are running this algorithm in the strongly decentralized setting, the regret could be linear for *both players*.

[13] For the leader, Bai et al. [2021] also takes a similar worst-case perspective over the ϵ-best-response set, but does not introduce a regularizer or take a minimum over ϵ.

`ExploreThenCommit`$(E, \mathcal{C})$ **(Algorithm 1).** The algorithm `ALG` = `ExploreThenCommit`$(E, \mathcal{C})$ operates in the strongly decentralized setting, taking as input $E \in [T]$ and a set of arms $\mathcal{C}$.[14] When `ALG` is applied to an instance, for the first $|\mathcal{C}| \cdot E$ rounds, the algorithm `ALG` pulls each arm in $\mathcal{C}$ a total of E times in a round-robin fashion. For the remaining $T - |\mathcal{C}| \cdot E$ rounds, the algorithm commits to the optimal empirical mean from the first $|\mathcal{C}| \cdot E$ rounds. This is a standard algorithm for multi-armed bandits [Slivkins, 2019, Lattimore and Szepesvári, 2020].

Algorithm 1: `ExploreThenCommit`$(E, \mathcal{C})$ applied to history H (see e.g., [Slivkins, 2019, Lattimore and Szepesvári, 2020])

Fix an arbitrary ordering $\mathcal{C} = \{c^1, \ldots, c^{|\mathcal{C}|}\}$.
Let $t = |H|$.
`/* Explore for the first` $E \cdot |\mathcal{C}|$ `rounds */`
if $t \leq E \cdot |\mathcal{C}|$ **then**
  Let $i = t \pmod{|\mathcal{C}|} + 1$ be the index of the action that should be pulled.
  return *point mass at* c^i
`/* Commit for the remaining rounds */`
if $t > E \cdot |\mathcal{C}|$ **then**
  `/* Discard history all but the first` $E \cdot |\mathcal{C}|$ `rounds. */`
  $H^* = \{(t', c_{t'}, r) \mid \exists (t', b_{t'}, r) \in H \text{ s.t. } t' \leq E \cdot |\mathcal{C}|\}$
  `/* Choose highest empirical mean. */`
  for $c \in \mathcal{C}$ **do**
    Set $S(c) := \{r \mid \exists (t', c_{t'}, r) \in H^* \text{ s.t. } c = c_{t'}\}$ `// observed rewards`
    $\hat{v}(c) \leftarrow (\sum_{r \in S(c)} r)/|S(c)|$ `// compute empirical mean`
  return *point mass at* $argmax_{c \in \mathcal{C}} \hat{v}(c)$

When both players run `ExploreThenCommit`, we show that if the leader's exploration phase ends before the follower's exploration phase, then both players can incur linear regret. This lower bound holds for any maximum tolerance $\gamma \leq 1$.

Proposition 4.3. *Suppose that the follower runs a separate instantiation of* `ExploreThenCommit`$(E, \mathcal{B})$ *for every* $a \in \mathcal{A}$. *Moreover, suppose that the leader runs* `ExploreThenCommit`$(E' \cdot |\mathcal{B}|, \mathcal{A})$ *for any* $E' \leq E$ *(i.e., the leader's exploration phase ends before the follower's exploration phase). Then, there exists an instance* $\mathcal{I}^*$ *such that both players incur linear regret with respect to the* γ*-tolerant benchmarks* β_1^{tol} *and* β_2^{tol}*: that is,* $\min(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(T)$.

Proof sketch of Proposition 4.3. The intuition is that in the leader's exploration phase, the follower alternates uniformly between all actions $\mathcal{B}$. This distorts the leader's learning during the leader's exploration phase, and as a result, the leader can choose a highly suboptimal arm during the commit phase. This can lead to linear regret for both players. We construct a single instance (Table 2, with $\delta = 0.1$) where both players incur linear regret. The full proof is deferred to Appendix D.1. □

4.3 Warmup: Simple Pair of Algorithms Achieving Sublinear Regret

The constant regret in Proposition 4.3 motivates the design of more sophisticated algorithms where the leader waits for the follower to partially converge before starting to learn. As a warmup, we show that a simple modification of the setup of Proposition 4.3 guarantees sublinear regret

[14] By setting $\mathcal{C} = \mathcal{A}$, this algorithm can be instantiated as $\texttt{ALG}_1$ for the leader; by setting $\mathcal{C} = \mathcal{B}$, this algorithm can be instantiated as $\texttt{ALG}_2$ for the follower.

(i.e., $O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3}\right)$ regret for both players. Both players run `ExploreThenCommit`-based algorithms, but the leader waits for the follower to finish exploring before starting to explore. More specifically, the leader runs `ExploreThenCommitThrowOut`, which acts similar to `ExploreThenCommit`, but with an extra exploration phase at the start, after which all rewards are thrown out. This initial phase is to allow the follower to partially converge.

`ExploreThenCommitThrowOut`$(E, E', \mathcal{C})$ **(Algorithm 2).** Taking as input $E, E' \in [T]$ and a set of arms $\mathcal{C}$, the algorithm $\texttt{ALG}_1 =$ `ExploreThenCommitThrowOut`$(E, E', \mathcal{C})$ operates in the strongly decentralized setting. It throws out the first $E' \cdot |\mathcal{C}|$ rounds and then runs `ExploreThenCommit`$(E, \mathcal{C})$.

Algorithm 2: `ExploreThenCommitThrowOut`$(E, E', \mathcal{C})$ applied to history H

Fix an arbitrary ordering $\mathcal{C} = \left\{c^1, \ldots, c^{|\mathcal{C}|}\right\}$. `/* Explore for first` $E' \cdot |\mathcal{C}|$ `rounds */`
if $t \leq E' \cdot |\mathcal{C}|$ **then**
  Let $i = t \pmod{|\mathcal{C}|} + 1$ be the index of the action that should be pulled.
  return *point mass at* c^i
`/* Run ETC for the remainder of time, throwing out first` $E' \cdot |\mathcal{C}|$ `rounds */`
if $t > E' \cdot |\mathcal{C}|$ **then**
  $H^* = \{(t' - E' \cdot |\mathcal{C}|, c_{t'}, r) \mid \exists (t', c_{t'}, r) \in H \text{ s.t. } t' > E' \cdot |\mathcal{C}|\}$ `// Throw out first` $E' \cdot |\mathcal{C}|$ `rounds of history`
  return *ExploreThenCommit*$(E, \mathcal{C})$ *applied to* H^*

We show that if the follower runs `ExploreThenCommit` and the leader runs `ExploreThenCommitThrowOut`, then both players achieve sublinear regret. For this result, we require that γ is not *too small*: $\gamma = \omega\left(T^{-1/3}\left(|\mathcal{A}| \cdot |\mathcal{B}|\right)^{1/3}\right)$ (see Section 6 for a discussion).

Theorem 4.4. *Let the follower run a separate instantiation of* *ExploreThenCommit*$(E_2, \mathcal{B})$ *for every* $a \in \mathcal{A}$*, and let the leader run* *ExploreThenCommitThrowOut*$(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$*. If* $E_2 = \Theta(|\mathcal{A}|^{-2/3}|\mathcal{B}|^{-2/3} \cdot (\log T)^{1/3}T^{2/3})$*, and* $E_1 = \Theta(|\mathcal{A}|^{-2/3} \cdot (\log T)^{1/3}T^{2/3})$*, then, the regret with respect to the* γ*-tolerant benchmarks is bounded as:*

$$\max(R_1(T), R_2(T)) = O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3}\right).$$

Proof sketch for Theorem 4.4. The "throw out" phase for $\texttt{ALG}_1$ enables $\texttt{ALG}_2$ to learn and commit to near-optimal actions. The meaningful exploration for $\texttt{ALG}_1$ thus begins *after* the follower has committed to actions. This enables $\texttt{ALG}_1$ to identify a near-optimal action given the arms that $\texttt{ALG}_2$ has already committed to after the first phase of exploration. Returning to our γ-tolerant benchmarks, for each player, we can upper bound regret by setting ϵ to be the suboptimality of the other player and achieve the desired regret bound. The full proof is deferred to Appendix D.2. □

One drawback of Theorem 4.4 is that requiring the follower to run a single algorithm is relatively restrictive. In the next subsection, we allow for a rich class of follower algorithms.

4.4 Main Result: Richer Algorithms Achieving Sublinear Regret

Our main result in this section is an adaptive algorithm for the leader (`ExploreThenUCB`, Algorithm 3) that achieves the same regret bounds while permitting greater flexibility for the follower (Theorem 4.5). Specifically, we only require that the follower converges to ϵ-optimal responses quickly,

which we formalize through high-probability instantaneous regret (Section 2.3). Since the leader's algorithm needs to be robust to a broader range of follower behaviors, we replace the commit phase of `ExploreThenCommit` with an adaptive algorithm. This motivates `ExploreThenUCB`, which explores in the first phase, and then runs a version of UCB on the arms $\mathcal{A}$. The initial exploration phase in `ExploreThenUCB`, similar to the initial exploration phase in `ExploreThenCommitThrowOut`, ensures that the leader waits for the follower to partially converge before starting to learn.

`ExploreThenUCB`(E) **(Algorithm 3).** The algorithm $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$ operates in the strongly decentralized setting, taking as input $E \in [T/|\mathcal{A}|]$. When $\texttt{ALG}_1$ applied to an instance, for the first $|\mathcal{A}| \cdot E$ rounds, the algorithm $\texttt{ALG}_1$ pulls each arm in $\mathcal{A}$ a total of E times, and then discards all history from these rounds. For the remaining $T - |\mathcal{A}| \cdot E$ rounds, the algorithm runs UCB, computing the upper confidence bound $v_1^{\mathrm{UCB}}(a) = \hat{v}_{1,t}(a) + \alpha_t(a)$ using confidence bound $\alpha_t(a) = \Theta\left(\sqrt{(\log T)/n_{E\cdot|\mathcal{A}|,t}(a)}\right)$, where $\hat{v}_{1,t}(a)$ is the empirical mean and $n_{E\cdot|\mathcal{A}|,t}(a)$ is the number of times that action a is chosen in the UCB phase after time step $E \cdot |\mathcal{A}|$ and prior to time step t. The algorithm then chooses the arm with maximum upper confidence bound.

Algorithm 3: `ExploreThenUCB`(E) applied to H

Fix an arbitrary ordering $\mathcal{A} = \{a^1, \ldots, a^{|\mathcal{A}|}\}$.
Let $t = |H|$.
`/* Explore for the first` $E \cdot |\mathcal{A}|$ `rounds */`
if $t \leq E \cdot |\mathcal{A}|$ **then**
  Let $i = \lceil \frac{t}{E} \rceil$ be the index of the action that should be pulled.
  return *point mass at* a^i
if $t > E \cdot |\mathcal{A}|$ **then**
  $H^* = \{(t' - E \cdot |\mathcal{A}|, a_{t'}, r) \mid \exists (t', a_{t'}, r) \in H \text{ s.t. } t' > E \cdot |\mathcal{A}|\}$ `// Throw out first` $E \cdot |\mathcal{A}|$ `rounds of history`
  Initialize $\hat{v}_1(a) = 1$ for $a \in \mathcal{A}$. `// Initialize empirical means.`
  Initialize $v_1^{\mathrm{UCB}}(a) = 1$ for $a \in \mathcal{A}$. `// Initialize UCB.`
  for $a \in \mathcal{A}$ **do**
    Set $S(a) := \{r \mid \exists (t', a_{t'}, r_{1,t'}(a_{t'}, b_{t'})) \in H^* \text{ s.t. } a = a_{t'}, r_{1,t'}(a_{t'}, b_{t'}) = r\}$ `// Observed rewards`
    if $S(a) \neq \emptyset$ **then**
      $\hat{v}_1(a) \leftarrow (\sum_{r \in S(a)} r)/|S(a)|$ `// Empirical mean`
      $\alpha(a) \leftarrow 10 \cdot \frac{\sqrt{\log T}}{\sqrt{|S(a)|}}$ `// confidence bound width`
      $v_1^{\mathrm{UCB}}(a) \leftarrow \min(1, \hat{v}_1(a) + \alpha(a))$
  Let $a^* = \mathrm{argmax}_{a \in \mathcal{A}} \left(v_1^{\mathrm{UCB}}(a)\right)$. `// arm with max upper confidence bound`
  return *point mass at* a^*

Even though the rewards observed by the leader are *not* stochastic (since the follower can pick different arms over time), we show if the leader runs `ExploreThenUCB` and the follower runs algorithms with sufficiently low high-probability instantaneous regret, then both players achieve $O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3}\right)$ regret. The assumptions on the follower's algorithm are satisfied by standard algorithms such as `ActiveArmElimination` (Algorithm 7; Proposition 7.1) and `ExploreThenCommit` (Algorithm 1; Proposition 7.2). For this result, we require that the maximum tolerance γ is not *too small*: $\gamma = \omega\left(T^{-1/3}\left(|\mathcal{A}| \cdot |\mathcal{B}|\right)^{1/3}\right)$ (see Section 6 for a discussion).

Theorem 4.5. *Let $E = \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}|\log T)^{1/3}T^{2/3})$. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret $g(t,T,\mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{-1/3}\right)$ for $t > E$ and $g(t,T,\mathcal{B}) = 1$ for $t \leq E$, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Then, it holds that the regret with respect to the γ-tolerant benchmarks β_1^{tol} and β_2^{tol} is bounded as:*

$$\max(R_1(T), R_2(T)) = O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3}\right).$$

Proof sketch of Theorem 4.5. The intuition is that the exploration phase of `ExploreThenUCB` ensures that all of the follower's actions have bounded suboptimality, and the UCB phase accounts for the follower changing which action they choose over time. In more detail, high-probability instantaneous regret guarantees that after the explore phase, all actions that the follower's chooses are within the ϵ-best-response set $\mathcal{B}_{\epsilon^*}(a)$ for $\epsilon^* = \Theta((|\mathcal{A}| \cdot |\mathcal{B}| \cdot \log T)^{1/3}T^{-1/3})$. For the UCB phase, the main lemma (Lemma D.4) is that if an arm $a \in \mathcal{A}$ is pulled, the empirical mean is at least $\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - \Theta\left(\sqrt{\log T / n_{E\cdot|\mathcal{A}|,t}(a)}\right)$ (the optimal utility for the leader when the follower worst-case ϵ-best-responds minus the confidence set size). Lemma D.4 enables us to analyze the leader's cumulative reward from each arm $a \in \mathcal{A}$ and thus bound the leader's regret. For the follower's regret, Lemma D.4 enables us to bound the number of times that arms outside of $\mathcal{A}_\epsilon$ are chosen, which enables us to bound the follower's regret. We defer the full proof to Appendix D.3. □

4.5 Regret lower bound

A natural question is whether the regret bound in Theorem 4.5 can be improved from $\tilde{O}(T^{2/3})$ to $\tilde{O}(\sqrt{T})$, given that such dependence is possible in single-player bandit problems. Interestingly, we show a *lower bound* of $T^{2/3}$ with respect to the γ-tolerant benchmarks, thus demonstrating that the dependence on T in Theorem 4.6 is near-optimal. This lower bound holds for any maximum tolerance $\gamma \leq 1$.

Theorem 4.6. *Let $\mathcal{A}$ and $\mathcal{B}$ be any action sets such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 2$. Consider any algorithms $\texttt{ALG}_1$ and $\texttt{ALG}_2$ which operate in either the strongly decentralized setting or the weakly decentralized setting. There exists an instance $\mathcal{I}^* = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ such that at least one of the players incurs $\Omega(T^{2/3} \cdot (|\mathcal{B}|)^{1/3})$ regret with respect to the γ-tolerant benchmarks β_1^{tol} and β_2^{tol}:*

$$\max(R_1(T;\mathcal{I}^*), R_2(T;\mathcal{I}^*)) = \Omega(T^{2/3} \cdot (|\mathcal{B}|)^{1/3}).$$

Proof sketch. In this sketch, we give intuition for a weaker bound of $\Omega(T^{2/3})$, deferring the strengthening to $\Omega(T^{2/3} \cdot (|\mathcal{B}|)^{1/3})$ to Appendix C.5. Like in the proof of Theorem 3.1, it suffices to consider a centralized environment (Lemma C.1). We show that on the $\mathcal{I}$ and $\tilde{\mathcal{I}}$ in Table 3 (with $\delta = \Theta(T^{-1/3})$), at least one player incurs $\Omega(T^{2/3})$ regret on at least one of these instances. The only way to distinguish the instances is to pull (a_1, b_2) at least $\Omega(T^{2/3})$ times, which gives low utility for both players. Intuitively, when the algorithm fails to distinguish $\mathcal{I}$ and $\tilde{\mathcal{I}}$, the algorithm must choose the same distribution over $\mathcal{A} \times \mathcal{B}$, but this gives $\Theta(T^{-1/3})$ loss for the leader on $\mathcal{I}$ or $\Theta(T^{-1/3})$ loss for the follower on $\tilde{\mathcal{I}}$. The full proof, which relies on a KL-divergence argument, is deferred to Appendix C.5. □

At a high-level, the $T^{2/3}$ regret bound in Theorem 4.6 is driven by the need to obtain precise estimates of *highly suboptimal action pairs* in order to learn to distinguish between two instances.

	b_1	b_2
a_1	$(0.5+\delta, \delta)$	$(0, \mathbf{0})$
a_2	$(0.5, 3\delta)$	$(0.5, 3\delta)$

(a) Mean rewards $(v_1(a,b), v_2(a,b))$ for $\mathcal{I}$

	b_1	b_2
a_1	$(0.5+\delta, \delta)$	$(0, \mathbf{2\delta})$
a_2	$(0.5, 3\delta)$	$(0.5, 3\delta)$

(b) Mean rewards $(\tilde{v}_1(a,b), \tilde{v}_2(a,b))$ for $\tilde{\mathcal{I}}$

Table 3: Two instances $\mathcal{I}$ (left) and $\tilde{\mathcal{I}}$ (right), which differ solely in the follower's reward for (a_1, b_2) (shown in **bold**). For δ sufficiently small, the instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$ are hard to distinguish and turn out to imply a $\Omega(T^{2/3})$ lower bound on regret with respect to the γ-tolerant benchmark (Theorem 4.6).

This is fundamentally different from single learner environments, where the learner only needs to obtain precise estimates of *near-optimal* arms. Our regret upper bound (Theorem 4.5) and lower bound (Theorem 4.6) have near-matching dependence on T and $|\mathcal{B}|$, but a gap in dependence on $|\mathcal{A}|$ (the upper bound scales with $|\mathcal{A}|^{1/3}$ while the lower bound is independent of $|\mathcal{A}|^{1/3}$). An interesting direction for future work is to close this gap.

5 Relaxed settings that permit faster learning

The lower bound in the previous section showed that $\Theta(T^{2/3})$ regret is optimal for the benchmarks β_1^{tol} and β_2^{tol} for general instances. Since a $T^{2/3}$ lower bound is atypical for K-armed bandits problems, we next consider relaxed environments under which faster learning—-i.e., $O(\sqrt{T})$ regret—is possible. In the first environment, we consider well-behaved instances (Section 5.1) and in the second environment, we weaken the benchmarks (Section 5.2). In both environments, we show that the learner does not need to worry about their learning being overly distorted by the follower; thus, the leader can start learning immediately, even before the follower's actions have partially converged, which leads to improved regret bounds. The algorithms that we design for the leader are variants of UCB.

5.1 Continuity condition on utilities

We first show that improved regret bounds are possible with a continuity condition on the player utilities. For intuition, the example in Table 1 gave a "hard" example resulting in linear regret in Theorem 3.1 and the related example in Table 3 resulted in $\Omega(T^{2/3})$ regret in Theorem 4.6. These examples relied on two outcomes with nearly identical utilities for the follower having significantly different utilities for leader, which could be viewed as a violation of continuity. This suggests that if arms that are *sufficiently different* for the leader were also *sufficiently different* for the follower, then it might be possible to beat the regret lower bound from Theorem 3.1 and Theorem 4.6.

We formalize continuity as follows: given an instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$, we define the Lipschitz constant L^* by[15]:

$$L^* = \sup_{i \neq j \in \{1,2\}} \sup_{(a,b) \neq (a',b')} \frac{|v_i(a,b) - v_i(a',b')|}{|v_j(a,b) - v_j(a',b')|}.$$

For example, when the two players have the same utilities (i.e., $v^1 = v^2$), then $L^* = 1$. More

[15]In the case of ties in rewards, if the numerator and denominator are both 0, we define $\frac{|v_i(a,b)-v_i(a',b')|}{|v_j(a,b)-v_j(a',b')|}$ to be 1 (because both players agree the elements are equivalent). If the denominator is 0 and numerator is nonzero, we define this fraction to be ∞ (because the items are indistinguishable to one player, while they give different rewards to the other).

generally, our continuity condition captures the extent to which players agree on which outcomes are different from each other (a more detailed discussion is given in Appendix E.1). Returning to the examples in Tables 1, 3, the "hard" instances yielding linear regret for the original Stackelberg benchmarks (Theorem 3.1; Table 1) have $L^* = \Theta(T^{-1/2})$ and the corresponding "hard" instances for $T^{2/3}$ regret for the γ-tolerant benchmarks (Theorem 4.6; Table 3) require that $L^* = \Theta(T^{-1/3})$; in contrast, we focus on utility functions where L^* is a constant.

When L^* is bounded, we show that it is possible for both players to achieve $\tilde{O}(\sqrt{T})$ regret even with respect to the *original Stackelberg benchmarks* in the strongly decentralized setting. The follower can run any algorithm $\texttt{ALG}_2$ with sufficiently low high-probability anytime regret (e.g., `ActiveArmElimination` as in Proposition 7.1 or UCB as in Proposition 7.3). We construct another UCB-based algorithm `LipschitzUCB` (Algorithm 4) for the leader, which expands the confidence sets based on the Lipschitz constant L^*.

`LipschitzUCB`(L, C) **(Algorithm 4).** The algorithm $\texttt{ALG}_1 =$ `LipschitzUCB`(L, C) operates in the strongly decentralized setting, taking as inputs parameters L and C. (The parameter L is intended to be an upper bound on the Lipschitz constant L^*, and the parameter C' is intended to be such that $\texttt{ALG}_2$ satisfies anytime regret bound $h(t, T, \mathcal{B}) = \sqrt{Ct \log T}$, where $C = C' \cdot \sqrt{|\mathcal{B}|}$ for a constant C'.) For each arm $a \in \mathcal{A}$, the algorithm computes UCB estimates $v_1^{\text{UCB}}(a)$ of the quantity $\max_{b \in \mathcal{B}} v_1(a, b)$ using the high-probability anytime regret bounds of $\texttt{ALG}_2$ as well as the upper bound on the Lipschitz constant. The algorithm then chooses the arm $a_t = \text{argmax}_{a \in \mathcal{A}} \max_{b \in \mathcal{B}'(a)} v_1^{\text{UCB}}(a)$.

Algorithm 4: `LipschitzUCB`(L, C) applied to H

Initialize $\hat{v}_1(a) = 1$ for $a \in \mathcal{A}$. `// Initialize empirical means for` $\max_{b \in \mathcal{B}} v_1(a, b)$.
Initialize $v_1^{\text{UCB}}(a) = 1$ for $a \in \mathcal{A}$. `// Initialize UCB for` $\max_{b \in \mathcal{B}} v_1(a, b)$
for $a \in \mathcal{A}$ **do**
  Set $S(a) := \{r \mid \exists (t', a_{t'}, r) \in H \text{ s.t. } a = a_{t'}\}$ `// Observed rewards`
  if $S(a) \neq \emptyset$ **then**
    $\hat{v}_1(a) \leftarrow (\sum_{r \in S(a)} r)/|S(a)|$ `// Empirical mean`
    $\alpha(a) \leftarrow \frac{10\sqrt{\mathcal{B} \log T}}{\sqrt{|S(a)|}} + C \cdot L \cdot \frac{\sqrt{\log T}}{\sqrt{|S(a)|}}$ `// confidence bound width`
    $v_1^{\text{UCB}}(a) \leftarrow \min(1, \hat{v}_1(a) + \alpha(a))$
Let $a^* = \text{argmax}_{a \in \mathcal{A}} \left(v_1^{\text{UCB}}(a)\right)$. `// arm with max upper confidence bound`
return *point mass at* a^*.

We obtain the following regret bound with respect to the Stackelberg benchmark, our strongest benchmark.

Theorem 5.1. *Suppose that $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ has Lipschitz constant L^*. Let $\texttt{ALG}_2$ be any algorithm satisfying high-probability anytime regret $h(t, T, \mathcal{B}) = C'\sqrt{|\mathcal{B}| t \log T}$ where C' is a constant, and let $\texttt{ALG}_1 =$ `LipschitzUCB`$(L, C'\sqrt{|B|})$ for any $L \geq L^*$. Then both players achieve the following regret bounds with respect to the original Stackelberg benchmarks β_1^{orig} and β_2^{orig}: that is, $R_1(T; \mathcal{I}) = O\left(L\sqrt{T|\mathcal{A}||\mathcal{B}| \log T}\right)$ and $R_2(T; \mathcal{I}) = O\left(L^2\sqrt{T|\mathcal{A}| \cdot |\mathcal{B}| \log T}\right)$.*

Proof sketch for Theorem 5.1. The intuition is the continuity conditions imply that small errors by the follower translate into bounded suboptimality for the leader (and vice versa); moreover, the high-probability anytime regret requirements bound the follower's errors. Together, these properties guarantee that the leader's empirical mean $\hat{v}_1(a)$ for each arm $a \in \mathcal{A}$ is close to the mean reward

$v_1(a, b^*(a))$ that they would receive if the follower best-responded: in more detail, the main lemma (Lemma E.2) is that the empirical mean $\hat{v}_1(a)$ is at least $v_1(a, b^*(a)) - \Theta(L\sqrt{\log T}/\sqrt{n_t(a)})$, where $n_t(a)$ is the number of times that arm a has been pulled prior to time step t. Using Lemma E.2 to bound the suboptimality of the leader's choice of actions $a_t \in \mathcal{A}$ and using the anytime regret requirements to bound the follower's suboptimality, we can bound both the leader's regret and the follower's regret. We defer the full proof to Appendix E.2. □

Finally, we compare our continuity condition and results with those in other works. Our continuity condition bears resemblance to the restrictions on utilities in Camara et al. [2020], Collina et al. [2023b]: in fact, our conditions are conceptually stronger since we require Lipschitz continuity across *all* pairs of actions rather only for near-optimal actions. However, Theorem 5.1 is not directly comparable with the results in Camara et al. [2020], Collina et al. [2023b] since we consider a stronger benchmark (the original Stackelberg benchmark) and also restrict to stochastic rewards. An interesting direction for future work would be to relax the Lipschitz continuity assumptions in our work, perhaps borrowing intuition from the stable action requirement of Collina et al. [2023b].

5.2 Weaker benchmark

Finally, we will consider the case where utilities are allowed to be arbitrary (L^* can be unbounded), but where we compete with weakened benchmarks, which we call *self-γ-tolerant*. These benchmarks capture the case where the player is not only tolerant of suboptimality the other player, but also tolerant of their own suboptimality. We thus take a min over the ϵ-best-response sets of *both* players.

Definition 5.2. Given a maximum tolerance $\gamma > 0$, we define the *self-γ-tolerant benchmarks*, $\beta_1^{\text{self-tol}}$ and $\beta_2^{\text{self-tol}}$, to be:

$$\beta_1^{\text{self-tol}} = \inf_{\epsilon \le \gamma} \left(\min_{a \in \mathcal{A}_\epsilon} \min_{b \in \mathcal{B}_\epsilon(a)} v_1(a, b) + \epsilon \right)$$

$$\beta_2^{\text{self-tol}} = \inf_{\epsilon \le \gamma} \left(\min_{a \in \mathcal{A}_\epsilon} \min_{b \in \mathcal{B}_\epsilon(a)} v_2(a, b) + \epsilon \right).$$

The tolerance of a player to their own suboptimality is the key difference from the γ-tolerant benchmarks from Section 4. For the follower, the benchmarks behave similarly: for a given value of ϵ, moving from $\max_{b \in \mathcal{B}} v_2(a, b)$ to $\min_{b \in \mathcal{B}_\epsilon(a)} v_2(a, b)$ differs by only an additive value of ϵ. However for the leader, there is a conceptual difference: the value $\mathbf{min}_{a \in \mathcal{A}_\epsilon} \min_{b \in \mathcal{B}_\epsilon(a)} v_1(a, b) + \epsilon$ is *not* necessarily within ϵ of $\mathbf{max}_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_\epsilon(a)} v_1(a, b) + \epsilon$. This is because $\mathcal{A}_\epsilon$ includes any action a that achieves high reward for *some* (near-optimal) actions by the follower, even if the worst-case (near-optimal) action by the follower yields arbitrarily low reward for the leader. As an illustration, the "hard" instances specified in Table 3 with $\delta = \Theta(T^{-1/3})$ led to the $T^{2/3}$ regret bound. The self-tolerant benchmark $\beta_1^{\text{self-tol}}$ reduces to $0.2 + \delta$ (rather than 0.5), so choosing (a_1, b_2) no longer results in constant loss for the leader.

Example 4.2 [Continued]. Let's again consider $\mathcal{I}$ in Table 2, which we also used to illustrate the γ-tolerant benchmark. The minimum is again attained at $\epsilon = \delta$, but the benchmark values change to $\beta_1^{\text{self-tol}} = 0.4 + \delta$ and $\beta_2^{\text{self-tol}} = 2 \cdot \delta + \delta$. The intuition is that the self-γ-tolerance benchmark only requires each agent to compete with the *worst* element within the product set $\mathcal{A}_\delta \times \mathcal{B}_\delta(a)$. Note that the resulting benchmark differs from the γ-tolerant benchmark for the follower only by δ, but differs by 0.4 (a constant) for the leader. We provide a detailed derivation of this example along with diagrams illustrating richer examples in Appendix A.

For the self-tolerant benchmarks, we show it is possible to achieve $\tilde{O}(\sqrt{T|\mathcal{A}||\mathcal{B}|})$ regret for both players (Theorem 5.3), which outperforms the $T^{2/3}$ lower bound for the stronger benchmark shown in Theorem 4.6. To demonstrate this is feasible, we construct a specific pair of algorithms (in the weakly decentralized setting where the leader can observe the follower's actions), that achieve a $O(\sqrt{T})$ regret upper bound. For the follower, we take the algorithm $\texttt{ALG}_2$ to be `ActiveArmElimination` (Algorithm 7), which cycles through phases of exploration, after which all *sufficiently suboptimal arms* are eliminated. For the leader, we construct a UCB-based algorithm `PhasedUCB` (Algorithm 5) which constructs confidence bounds for every pair of actions (a, b).

`PhasedUCB`$(M_1, \ldots, M_P)$ (Algorithm 5). The algorithm $\texttt{ALG}_1 =$ `PhasedUCB`$(M_1, \ldots, M_P)$ operates in the weakly decentralized setting, taking as input the parameters $M_1, \ldots, M_P \geq 0$. (The parameter M_i is intended to capture the number of times that an arm is pulled in phase i by the instantiation of `ActiveArmElimination` specified by $\texttt{ALG}_2$.) The algorithm $\texttt{ALG}_1$ computes UCB estimates $v_1^{\text{UCB}}(a, b)$ for $v_1(a, b)$, computes the set of active arms $\mathcal{B}'(a)$ in the previous phase of $\texttt{ALG}_2$'s instantiation of `ActiveArmElimination` for each arm a (`ComputeActiveArms`, Algorithm 6), and chooses the arm with maximum UCB: $a_t = \text{argmax}_{a \in \mathcal{A}} \max_{b \in \mathcal{B}'(a)} v_1^{\text{UCB}}(a, b)$. `ComputeActiveArms` computes the active arms $\mathcal{B}'(a)$ by iterating through H and keeping track of whenever a new phase is entered using the parameters $M_1, \ldots, M_P$.

Algorithm 5: `PhasedUCB`$(M_1, \ldots, M_P)$ applied to H

Let $\hat{v}_1(a, b) = 0$ for $a \in \mathcal{A}$ and $b \in \mathcal{B}$. `// initialize empirical mean of` $v_1(a, b)$
Let $v_1^{\text{UCB}}(a, b) = 1$ for $a \in \mathcal{A}$ and $b \in \mathcal{B}$. `// initialize UCB for` $v_1(a, b)$
Let $\mathcal{B}'(a) =$ `ComputeActiveArms`$(M_1, \ldots, M_P, H)$. `// active arms in previous phase for` $\texttt{ALG}_2$
for $a \in \mathcal{A}$ **do**
  for $b \in \mathcal{B}$ **do**
    Set $S(a, b) := \{r \mid \exists (t', a_{t'}, b_{t'}, r) \in H \text{ s.t. } a = a_{t'}, b = b_{t'}\}$ `// observed rewards`
    if $S(a, b) \neq \emptyset$ **then**
      $\hat{v}_1(a, b) \leftarrow (\sum_{r \in S(a,b)} r)/|S(a, b)|$ `// compute empirical mean`
      $\alpha(a, b) := 10 \cdot \sqrt{\frac{\log T}{|S(a,b)|}}$ `// confidence bound width`
      $v_1^{\text{UCB}}(a, b) \leftarrow \min(1, \hat{v}_1(a, b) + \alpha(a, b))$ `// compute UCB`
Let $a^* = \text{argmax}_{a \in \mathcal{A}} \max_{b \in \mathcal{B}'(a)} \left(v_1^{\text{UCB}}(a, b)\right)$. `// arm with max upper confidence bound for any valid` b
return *point mass at* a^i

We show that both players achieve $O(\sqrt{T})$ regret. For this result, we require that the γ is not *too small*: $\gamma = \Omega(T^{-1/4}(|\mathcal{A}| \cdot |\mathcal{B}| \cdot \log T)^{1/2}))$ (see Section 6 for a discussion).

Theorem 5.3. *Suppose that for each $a \in \mathcal{A}$, the algorithm $\texttt{ALG}_2$ runs a separate instantiation of `ActiveArmElimination` with parameters $M_1, \ldots, M_P$ (where $M_i = \Theta(\log T \cdot 2^{2i})$ denotes the number of times that each arm is pulled in phase i). Let $\texttt{ALG}_1 =$ `PhasedUCB`$(M_1, \ldots, M_P)$. Then it holds that the regret with respect to the self-γ-tolerant benchmarks $\beta_1^{\textit{self-tol}}$ and $\beta_2^{\textit{self-tol}}$ is bounded as:*

$$\max(R_1(T), R_2(T)) = O\left(\sqrt{|\mathcal{A}| \cdot |\mathcal{B}| \cdot T \cdot \log T}\right).$$

Proof sketch for Theorem 5.3. The intuition is that the benchmark allows the leader to choose any $a \in \mathcal{A}_\epsilon$. The definition of $\mathcal{A}_\epsilon$ means that the leader can take an optimistic perspective on the

Algorithm 6: `ComputeActiveArms`$(M_1, \ldots, M_P, H)$

Initialize $s'(a) = 0$ for $a \in \mathcal{A}$. `// Index of the last completed phase for ALG`$_2$ `on arm` a`.`
Initialize $t'(a) = 1$ for $a \in \mathcal{A}$. `// Time step marking beginning of phase` $s'+1$ `for ALG`$_2$ `on arm` a`.`
Initialize $\mathcal{B}'(a) = \mathcal{B}$. `// Active arms in phase` s' `for ALG`$_2$ `on arm` a`.`
Initialize $newphase_a =$ False for $a \in \mathcal{A}$. `// Boolean for first time step in phase for ALG`$_2$ `on` a`.`
Let $t = |H|$.
for $t'' = 1$ **to** t **do**
  for $a \in \mathcal{A}$ **do**
    for $b \in \mathcal{B}$ **do**
      Let $n(a,b) := \left|\left\{(t'', a_{t''}, b_{t''}, r_{1,t''}(a_{t''}, b_{t''})) \in H \mid a_{t''} = a, b_{t''} = b, t'' \geq t'_a\right\}\right|$.
      if $n(a,b) > M_{s'_a+1}$ **then**
        $newphase_a =$ True.
    if $newphase_a = True$ **then**
      Update $\mathcal{B}'(a) \leftarrow$
       $\left\{b \in \mathcal{B} \mid \exists (t'', a_{t''}, b_{t''}, r_{1,t''}(a_{t''}, b_{t''})) \in H \text{ s.t. } t'_a \leq t'' < t, a_{t''} = a, b_{t''} = b\right\}$
      Update $s'(a) \leftarrow s'(a) + 1$.
      Update $t'(a) \leftarrow t$.
      $newphase_a =$ False.
return $\{\mathcal{B}'(a)\}_{a \in \mathcal{A}}$.

follower's choice of action $\mathcal{B}_\epsilon(a)$ (and not have to prepare for the worst-case action in $\mathcal{B}_\epsilon(a)$). This optimistic perspective surfaces in `PhasedUCB` in terms of how the leader evaluates an action a based on the maximum UCB $\max_{b \in \mathcal{B}'(a)} v_1^{\text{UCB}}(a,b)$ across all active arms $b \in \mathcal{B}'(a)$. To analyze this pair of algorithms, we show a bound ϵ_t for each time step t such that a_t is an ϵ_t-best-response for the leader and b_t is an ϵ_t-best-response for the follower: the main lemma (Lemma E.6) shows that we can set ϵ_t to be $\Theta(\sqrt{|\mathcal{B} \cdot \log T / n_t(a_t)})$ where $n_t(a_t)$ is the number of times that arm a_t has been pulled prior to time step t. The full proof is deferred to Appendix E.3. □

The regret bound in Theorem 5.3 is nearly optimal, as we show in the following $\Omega(\sqrt{T|\mathcal{A}| \cdot |\mathcal{B}|})$ lower bound for self-γ-tolerant benchmarks, which holds for any maximum tolerance $\gamma \leq 1$.

Proposition 5.4. *Let $\mathcal{A}$ and $\mathcal{B}$ be any action sets such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 2$. Consider any algorithms `ALG`$_1$ and `ALG`$_2$ which operate in either the strongly decentralized setting or weakly decentralized setting. There exists an instance $\mathcal{I}^* = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ such that at least one of the players incurs $\Omega(\sqrt{T \cdot (|\mathcal{A}| - 1) \cdot |\mathcal{B}|})$ regret with respect to the self-γ-tolerant benchmarks $\beta_1^{\text{self-tol}}$ and $\beta_2^{\text{self-tol}}$, that is: $\max(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(\sqrt{T \cdot (|\mathcal{A}| - 1) \cdot |\mathcal{B}|})$.*

Taken together, Theorem 5.3 and Proposition 5.4 demonstrate the self-γ-tolerant benchmarks lead to $\tilde{\Theta}(\sqrt{T|\mathcal{A}|\mathcal{B}|})$ regret bounds for each player.

We note that Theorem 5.3 requires the follower to run a specific algorithm: this contrasts with our results for the γ-tolerant benchmark (Theorem 4.5) and the Lipschitz benchmark (Theorem 5.1) which allowed for greater flexibility in the follower algorithm. An interesting direction for future work would be to design a leader algorithm for the self-γ-tolerant benchmark that permits a richer family of follower behaviors.

	b_1	b_2
a_1	(0.6, 0.05)	(0.2, 0.1)
a_2	(0.5, 0.2)	(0.4, 0.15)

Table 4: Taking γ to be too small makes the benchmark too easy: for $\gamma = 0$, we have $\beta_1^{\text{tol}} = 0.5, \beta_2^{\text{tol}} = 0.2$, but for $\gamma = 0.05$ we have $\beta_1^{\text{tol}} = 0.5$ and $\beta_2^{\text{tol}} = 0.15$ (see Section 6.1)

6 Discussion of Benchmark Parameters: The Maximum Tolerance and the Regularizer

Our relaxed benchmarks—the γ-tolerant benchmarks (Definition 4.1) and the self-γ-tolerant benchmarks (Definition 5.2)—depend on two parameters: (1) the maximum tolerance γ and (2) the ϵ-regularizer. In this section, we discuss the role of each parameter and describe extensions of our results to alternate settings of these parameters.

6.1 Maximum Tolerance γ

The value γ intuitively captures the players' maximum tolerance for suboptimality. Taking γ to be small makes our benchmarks more challenging, because it reduces the space of permissible suboptimality levels ϵ over which the infimum is taken. In contrast, taking γ to be large can make our benchmarks *too easy*: for example, consider Table 4, which shows a case where setting $\gamma = 0.05$ reduces the benchmark for the follower, but the instance has rewards that are sufficiently far apart that for large T the Stackelberg equilibrium should intuitively be learnable.

We briefly discuss how our results extend to different maximum tolerances γ. First, we prove our lower bounds (Theorem 4.6, Proposition 5.4) for the "hardest case" of $\gamma = 1$, which means that these lower bounds hold for *all* maximum tolerances γ.

On the other hand, our upper bounds require sufficiently large γ. For some intuition, all of our analyses require that $\gamma = \omega(1/\sqrt{T})$, since followers with high-probability instantaneous regret rates of $\Theta(\sqrt{|\mathcal{B}| \cdot \log(T)/t})$ require $\Omega(T)$ rounds to find a $O(1/\sqrt{T})$-optimal solution. As to what specific values of γ that each result requires, Theorems 4.4 and 4.5 hold for any $\gamma = \omega\left(T^{-1/3} |\mathcal{A}|^{1/3} |\mathcal{B}|^{1/3} \cdot (\log(T)^{1/3})\right)$, while Theorem 5.3 assumes that $\gamma = \Omega\left(T^{-1/4}\sqrt{|\mathcal{A}||\mathcal{B}| \cdot \log T}\right)$.

6.2 ϵ-Regularizer

Since the ϵ-regularizer adds an implicit penalty for increasing ϵ in the benchmark, a natural question is how our benchmark would change if we changed the regularizer from ϵ to other functional forms $f(\epsilon)$. To provide some preliminary intuition for this, we consider $f(\epsilon) = c \cdot \epsilon^d$ regularizer, which leads to the following generalized γ-tolerant benchmarks.

Definition 6.1 (Generalization of Definition 4.1)**.** Given a maximum tolerance $\gamma > 0$ and parameters $c > 0$, and $d > 0$, we define the *generalized* (c, d, γ)*-tolerant benchmarks* β_1^{tol} and β_2^{tol} to be:

$$\beta_1^{\text{tol}} = \inf_{\epsilon \le \gamma} \Big(\underbrace{\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_\epsilon(a)} v_1(a,b)}_{\epsilon\text{-relaxed Stackelberg utility}} + \underbrace{c \cdot \epsilon^d}_{\epsilon\text{-regularizer}} \Big)$$

$$\beta_2^{\text{tol}} = \inf_{\epsilon \le \gamma} \Big(\underbrace{\min_{a \in \mathcal{A}_\epsilon} \max_{b \in \mathcal{B}} v_2(a,b)}_{\epsilon\text{-relaxed Stackelberg utility}} + \underbrace{c \cdot \epsilon^d}_{\epsilon\text{-regularizer}} \Big).$$

At a conceptual level, different settings of c and d capture different levels of tolerance that a player has for sub-optimality in the other player. Higher values of c and smaller values of d capture greater intolerance, and thus lead to harsher penalties. The resulting changes in the benchmarks capture that if a player is less tolerant, we might expect them to experience a higher regret for a given suboptimality level of the other player.

We show how our two main upper bounds in Section 5 generalize to these new benchmarks, focusing on the case of $c \geq 1$ and $d \leq 1$ (where the benchmark becomes harder). We first show the following generalization of Theorem 4.4 by adjusting the explore phase length to depend on c and d.

Theorem 6.2. *Suppose that $c \geq 1$ and $d \leq 1$, and let $\eta := 2/(2+d)$. Let the follower run a separate instantiation of $\texttt{ExploreThenCommit}(E_2, \mathcal{B})$ for every $a \in \mathcal{A}$, and let the leader run $\texttt{ExploreThenCommitThrowOut}(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$. If $E_2 = \Theta(|\mathcal{A}|^{-\eta}|\mathcal{B}|^{-\eta} \cdot (\log T)^{1-\eta}(c \cdot T)^{\eta})$, and $E_1 = \Theta(|\mathcal{A}|^{-\eta} \cdot (\log T)^{1-\eta}(c \cdot T)^{\eta})$, then the leader and follower regret with respect to the generalized (c, d, γ)-tolerant benchmarks are both at most:*

$$\max(R_1(T), R_2(T)) = O\big((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta}\big).$$

We next show the following generalizations of Theorem 4.5 by again adjusting the explore phase length to depend on c and d. Like in Theorem 4.5, the assumptions on the follower's algorithm in this result are satisfied by standard algorithms such as `ActiveArmElimination` (Algorithm 7; Proposition 7.1) and `ExploreThenCommit` (Algorithm 1; Proposition 7.2).

Theorem 6.3. *Suppose that $c \geq 1$ and $d \leq 1$, and let $\eta := 2/(2+d)$. Let $E = \Theta(|\mathcal{A}|^{-\eta}(|\mathcal{B}| \log T)^{1-\eta}(c \cdot T)^{\eta})$. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret $g(t, T, \mathcal{B}) = O\left((|\mathcal{A}| \cdot |\mathcal{B}| \cdot \log T)^{\eta/2} \cdot (c \cdot T)^{-\eta/2}\right)$ for $t > E$ and $g(t, T, \mathcal{B}) = 1$ for $t \leq E$, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Then, then the leader and follower regret with respect to the generalized (c, d, γ)-tolerant benchmarks are both bounded as:*

$$\max(R_1(T), R_2(T)) = O((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta}).$$

The proofs of Theorem 6.2 and Theorem 6.3 follows from the same arguments as the proof of Theorem 4.4 and Theorem 4.5, respectively, but with the values of E_1, E_2 modified (full proofs are deferred to Appendix F). Note that as d decreases, the regret bound worsens: this aligns with the intuition that smaller values of d capture greater intolerance. Similarly, the regret increases with c.

We defer a more extensive treatment of these generalized benchmarks to future work. Moreover, another interesting for future work would be to extend our model and results to more general functions $f(\epsilon)$ and also allow the two players to have different regularizers.

7 Algorithms satisfying high-probability instantaneous regret and high-probability anytime regret bounds

Our algorithms for the leader placed assumptions on the fine-grained performance of the follower's algorithm. More specifically, the regret bound for `ExploreThenUCB` required an *high-probability instantaneous regret bound* for the follower (Theorem 4.5), and the regret bound for `LipschitzUCB` required an *high-probability anytime regret bound* for the follower (Theorem 5.1). In this section, we discuss these conditions in more detail and analyze which standard algorithms satisfy each of these conditions.

First, we observe that high probability instantaneous regret bound immediately translate to high-probability anytime regret bounds.

Observation 7.1. *Suppose that* $\texttt{ALG}_2$ *satisfies a high-probability instantaneous regret bound of* g*. Then it holds that* $\texttt{ALG}_2$ *satisfies an anytime regret bound of* h *defined as* $h(t,T) := \sum_{t'=1}^{t} g(t',T)$*.*

As a consequence, if $\texttt{ALG}_2$ satisfies the high-probability instantaneous regret bound in Theorem 4.5 (i.e., $g(t,T,\mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{-1/3}\right)$ for $t > E := \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}|\log T)^{1/3}T^{2/3})$ and $g(t,T,\mathcal{B}) = 1$ for $t \leq E$), then $\texttt{ALG}_2$ also satisfies an anytime regret bound of h defined for $t > E$ as:

$$h(t,T) := \sum_{t'=1}^{t} g(t',T) = E + \sum_{t'=E}^{t} O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{-1/3}\right) = O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{2/3}\right).$$

However, this naive high-probability anytime regret bound is not strong enough for Theorem 5.1: we can nonetheless achieve the desired regret bound with additional assumptions on $\texttt{ALG}_2$ as we describe below.

Algorithm 7: `ActiveArmElimination`$(M_1,\ldots,M_P)$ applied to (a,H) (adapted from [Even-Dar et al., 2002, Lattimore and Szepesvári, 2020])

Initialize $s' = 0$, $t' = 1$, $\mathcal{B}' = \mathcal{B}$ `// Index of the last completed phase, time step marking beginning of phase` $s'+1$`, active arms in phase` s'`.`
Initialize *newphase* = False. `// Boolean for first time step in phase.`
Let $t = |H|$.
for $t'' = 1$ **to** t **do**
  for $b \in \mathcal{B}'$ **do**
    Let $n(a,b) := |\{(t'', a_{t''}, b_{t''}, r) \in H \mid a_{t''} = a, b_{t''} = b, t'' \geq t'\}|$.
  if $n(a,b) = M_{s'+1} \forall b \in \mathcal{B}'$ **then**
    newphase = True.
  if *newphase* = *True* **then**
    for $b \in \mathcal{B}'$ **do**
      Set $S(a,b) := \{r \mid \exists (t'', a_{t''}, b_{t''}, r) \in H \text{ s.t. } a_{t''} = a, b = b_{t''}, t'' \geq t'\}$ `// observed rewards`
      $\hat{v}_2(a,b) \leftarrow (\sum_{r \in S(a,b)} r)/|S(a,b)|$ `// compute empirical mean`
    Update $\mathcal{B}' \leftarrow \{b \mid \hat{v}_2(a,b) + \frac{20\cdot\sqrt{\log T}}{\sqrt{M_{s'}}} \geq \max_{b \in \mathcal{B}'} \hat{v}_2(a,b)\}$.
    Update $s' \leftarrow s'+1$.
    Update $t' \leftarrow t$.
    newphase = False.
$i = ((t-t') \mod (|\mathcal{B}'|)) + 1$. `// Calculate next arm to be pulled`
return *point mass at* b_i.

Next, we show that `ActiveArmElimination` [Even-Dar et al., 2002] (Algorithm 7, see Lattimore and Szepesvári [2020] for a textbook treatment) satisfies both the high-probability instantaneous regret bound required for Theorem 4.5 and the high-probability anytime regret bound required for Theorem 5.1.

Proposition 7.1. *Suppose that for every* $a \in \mathcal{A}$*, the follower runs a separate instantiation of* `ActiveArmElimination`$(M_1,\ldots,M_P)$ *(Algorithm 7) with* $M_i = \Theta(\log T \cdot 2^{2i})$*. Then the follower satisfies high-probability instantaneous regret* $g(t,T,\mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot \log(T)/t}$*, which implies* $g(t,T,\mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{-1/3}\right)$ *for* $t \geq \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}|\log T)^{1/3}T^{2/3})$*. Moreover, the follower satisfies high-probability anytime regret* $h(t,T,\mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot \log(T) \cdot t})$*.*

Next, we show that `ExploreThenCommit` (Algorithm 1, see Slivkins [2019], Lattimore and Szepesvári [2020] for a textbook treatment) satisfies the high-probability instantaneous regret bound required for Theorem 4.5.

Proposition 7.2. *Suppose that the follower runs a separate instantiation of* `ExploreThenCommit`$(E, \mathcal{B})$ *(Algorithm 1) for every $a \in \mathcal{A}$. Then, the follower satisfies high-probability instantaneous regret $g(t, T, \mathcal{B}) = \mathcal{O}(\sqrt{\log T / E})$ for all time steps $t \geq E \cdot |\mathcal{B}|$. If $E = \Theta((|\mathcal{A} \cdot |\mathcal{B}|)^{-2/3} (\log T)^{1/3} T^{2/3})$, then $g(t, T, \mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}| \log T)^{1/3} T^{-1/3}\right)$ for $t \geq \Theta(|\mathcal{A}|^{-2/3} (|\mathcal{B}| \log T)^{1/3} T^{2/3})$.*

Note that `ExploreThenCommit` does *not* satisfy the high-probability anytime regret bound required for Theorem 5.1 due to the uniform exploration phase at the beginning of the algorithm means.

Finally, we show that UCB [Auer et al., 2002] (see Slivkins [2019], Lattimore and Szepesvári [2020] for a textbook treatment) satisfies the high-probability anytime regret bound required in Theorem 5.1.

Proposition 7.3. *Suppose that the follower runs a separate instantiation of UCB for every $a \in \mathcal{A}$. Then, the follower satisfies high-probability anytime regret bound $h(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot t \cdot \log(T)})$.*

We do not expect that UCB satisfies the high-probability instantaneous regret bound required for Theorem 4.5, using the intuition that UCB does not provide final-iterate convergence guarantees.

8 Discussion

In this paper, we studied two-agent environments where interactions are *sequential*, utilities are *misaligned*, and each agent *learns* their utilities over time. We modeled these environments as decentralized Stackelberg games where both agents are bandit learners who only observe their own utilities, and we investigated the implications for each agent's cumulative utility over time. Motivated by the offline Stackelberg equilibrium benchmarks being infeasible (Theorem 3.1), we designed γ-tolerant benchmarks which allow for approximate best responses by the other agent.

We proved that both players can achieve $\tilde{\Theta}(T^{2/3})$ regret with respect to the γ-tolerant benchmarks. To achieve this regret bound, we designed an algorithm (i.e., `ExploreThenUCB`; Algorithm 3) where the leader waits for the follower to partially converge before starting to learn; this algorithm achieves $\tilde{\Theta}(T^{2/3})$ regret for both players under a rich class of follower learning algorithms (Theorem 4.5). We further show that $\tilde{\Theta}(T^{2/3})$ regret is unavoidable for any pair of algorithms (Theorem 4.6). Furthermore, we showed that $O(\sqrt{T})$ regret is possible in two relaxed environments: i.e., under a relaxed benchmark that is (self-)tolerant of a player's own mistakes (Theorem 5.3) or when players agree on which pair of actions are different (Theorem 5.1)

Our results have broader implications for *designing* two-agent environments to achieve favorable utility for both agents. For example, given that our results illustrate that certain properties for the follower (such as high-probability instantaneous regret or high-probability anytime regret bounds) and certain properties for the leader (such as waiting for the follower to partially converge) are conducive to low regret, it may be helpful for a designer to engineer or encourage agents to follow these algorithmic principles. As another example, our continuity results in Section 5.1 illustrate the importance of reducing *near-ties* in utilities between different items, which could be achieved by allowing agents to express preferences between items in a nuanced fashion.

More broadly, our benchmarks and regret analysis suggest several interesting avenues for future work. For example, while Theorem 4.5 offered flexibility in the follower's choice of algorithm, we required that the leader follow a particular algorithm: it would be interesting to explore richer classes of leader algorithms which maintain low regret. Additionally, while our framework captures a range

of real-world applications including chatbots (Example 2.1 in Section 2.4) and recommender systems (Example 2.2 in Section 2.4), an interesting future direction would be to focus on a particular application and incorporate application-specific nuances (e.g., bidder learning rates in advertising auctions [Nekipelov et al., 2015, Noti and Syrgkanis, 2021, Nisan and Noti, 2017]). Finally, while we study the role of continuity requirements that reflect alignment (Section 5.1), it would be interesting to consider other structured bandit environments such as linear utility functions and generalize our benchmarks and results accordingly.

9 Acknowledgements

We thank Keegan Harris, Jason Hartline, Nika Haghtalab, Nick Wu, and Kunhe Yang for valuable comments and feedback. KD was partially supported by a Vannevar Bush Faculty Fellowship, a Simons Collaboration grant, and a grant from the MacArthur Foundation. MJ was partially supported by an Open Philanthropy AI Fellowship.

References

Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E. Schapire. Corralling a band of bandit algorithms. In *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, volume 65 of *Proceedings of Machine Learning Research*, pages 12–38. PMLR, 2017.

Arpit Agarwal and William Brown. Online recommendations for agents with discounted adaptive preferences. *CoRR*, abs/2302.06014, 2023.

Saba Ahmadi, Hedyeh Beyhaghi, Avrim Blum, and Keziah Naggita. The strategic perceptron. In *EC '21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*, pages 6–25. ACM, 2021.

Ioannis Anagnostides, Constantinos Daskalakis, Gabriele Farina, Maxwell Fishelson, Noah Golowich, and Tuomas Sandholm. Near-optimal no-regret learning for correlated equilibria in multi-player general-sum games. In *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 736–749. ACM, 2022.

Guy Aridor, Yishay Mansour, Aleksandrs Slivkins, and Zhiwei Steven Wu. Competing bandits: The perils of exploration under competition. *CoRR*, abs/2007.10144, 2020.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47(2-3):235–256, 2002.

Yu Bai, Chi Jin, Huan Wang, and Caiming Xiong. Sample-efficient learning of stackelberg equilibria in general-sum games. *Advances in Neural Information Processing Systems*, 34:25799–25811, 2021.

Michiel A. Bakker, Martin J. Chadwick, Hannah Sheahan, Michael Henry Tessler, Lucy Campbell-Gillingham, Jan Balaguer, Nat McAleese, Amelia Glaese, John Aslanides, Matt M. Botvinick, and Christopher Summerfield. Fine-tuning language models to find agreement among humans with diverse preferences. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Maria-Florina Balcan, Avrim Blum, Nika Haghtalab, and Ariel D Procaccia. Commitment without regrets: Online learning in stackelberg security games. In *Proceedings of the sixteenth ACM conference on economics and computation*, pages 61–78, 2015.

Santiago R. Balseiro and Yonatan Gur. Learning in repeated auctions with budgets: Regret minimization and equilibrium. *Manag. Sci.*, 65(9):3952–3968, 2019.

Gagan Bansal, Tongshuang Wu, Joyce Zhou, Raymond Fok, Besmira Nushi, Ece Kamar, Marco Túlio Ribeiro, and Daniel S. Weld. Does the whole exceed its parts? the effect of AI explanations on complementary team performance. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*, pages 81:1–81:16. ACM, 2021.

Christian Borgs, Jennifer T. Chayes, Nicole Immorlica, Kamal Jain, Omid Etesami, and Mohammad Mahdian. Dynamics of bid optimization in online advertisement auctions. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 531–540. ACM, 2007.

Mark Braverman, Jieming Mao, Jon Schneider, and S. Matthew Weinberg. Selling to a no-regret buyer. In *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 523–538. ACM, 2018.

William Brown, Jon Schneider, and Kiran Vodrahalli. Is learning in games good for the learners? *Advances in Neural Information Processing Systems*, 36, 2023.

Modibo K Camara, Jason D Hartline, and Aleck Johnsen. Mechanisms for a no-regret agent: Beyond the common prior. In *2020 ieee 61st annual symposium on foundations of computer science (focs)*, pages 259–270. IEEE, 2020.

Lawrence Chan, Dylan Hadfield-Menell, Siddhartha Srinivasa, and Anca Dragan. The assistive multi-armed bandit. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 354–363. IEEE, 2019.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. Unleashing the potential of prompt engineering in large language models: a comprehensive review. *CoRR*, abs/2310.14735, 2023.

Yiling Chen, Yang Liu, and Chara Podimata. Learning strategy-aware linear classifiers. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Natalie Collina, Eshwar Ram Arunachaleswaran, and Michael Kearns. Efficient stackelberg strategies for finitely repeated games. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023, London, United Kingdom, 29 May 2023 - 2 June 2023*, pages 643–651. ACM, 2023a.

Natalie Collina, Aaron Roth, and Han Shao. Efficient prior-free mechanisms for no-regret agents. *arXiv preprint arXiv:2311.07754*, 2023b.

Nina Corvelo Benz and Manuel Rodriguez. Human-aligned calibration for ai-assisted decision making. *Advances in Neural Information Processing Systems*, 36, 2024.

Constantinos Daskalakis, Alan Deckelbaum, and Anthony Kim. Near-optimal no-regret algorithms for zero-sum games. In Dana Randall, editor, *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 235–254. SIAM, 2011.

Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-optimal no-regret learning in general games. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27604–27616, 2021.

Yuan Deng, Jon Schneider, and Balasubramanian Sivan. Strategizing against no-regret learners. *Advances in neural information processing systems*, 32, 2019.

Kate Donahue, Kostas Kollias, and Sreenivas Gollapudi. When are two lists better than one?: Benefits and harms in joint decision-making. *AAAI '24'*, 2024.

Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 55–70. ACM, 2018.

Michael D. Ekstrand and Martijn C. Willemsen. Behaviorism is not enough: Better recommendations through listening to users. In Shilad Sen, Werner Geyer, Jill Freyne, and Pablo Castells, editors, *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15-19, 2016*, pages 221–224. ACM, 2016.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *COLT*, volume 2, pages 255–270. Springer, 2002.

Tanner Fiez, Benjamin Chasnov, and Lillian J Ratliff. Convergence of learning dynamics in stackelberg games. *arXiv preprint arXiv:1906.01217*, 2019.

Jiarui Gan, Minbiao Han, Jibang Wu, and Haifeng Xu. Robust stackelberg equilibria. *arXiv preprint arXiv:2304.14990*, 2023.

Wenshuo Guo, Nika Haghtalab, Kirthevasan Kandasamy, and Ellen Vitercik. Leveraging reviews: Learning to price with buyer and seller uncertainty. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*, 2023.

Guru Guruganesh, Yoav Kolumbus, Jon Schneider, Inbal Talgam-Cohen, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Joshua R. Wang, and S. Matthew Weinberg. Contracting with a learning agent. *CoRR*, abs/2401.16198, 2024.

Nika Haghtalab, Thodoris Lykouris, Sloan Nietert, and Alexander Wei. Learning in stackelberg games with non-myopic agents. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 917–918, 2022.

Nika Haghtalab, Chara Podimata, and Kunhe Yang. Calibrated stackelberg games: Learning optimal commitments against calibrated agents. *arXiv preprint arXiv:2306.02704*, 2023.

MohammadTaghi Hajiaghayi, Mohammad Mahdavi, Keivan Rezaei, and Suho Shin. Regret analysis of repeated delegated choice. *arXiv preprint arXiv:2310.04884*, 2023.

Minbiao Han, Michael Albert, and Haifeng Xu. Learning in online principal-agent interactions: The power of menus. *CoRR*, abs/2312.09869, 2023.

Keegan Harris, Zhiwei Steven Wu, and Maria-Florina Balcan. Regret minimization in stackelberg games with side information. *CoRR*, abs/2402.08576, 2024.

Joey Hong, Sergey Levine, and Anca D. Dragan. Zero-shot goal-directed dialogue via RL on imagined conversations. *CoRR*, abs/2311.05584, 2023.

Meena Jagadeesan, Michael I. Jordan, and Nika Haghtalab. Competition, alignment, and equilibria in digital marketplaces. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023*, pages 5689–5696, 2023.

Hsu Kao, Chen-Yu Wei, and Vijay G. Subramanian. Decentralized cooperative reinforcement learning with hierarchical information structure. In *International Conference on Algorithmic Learning Theory, 29 March - 1 April 2022, Paris, France*, volume 167 of *Proceedings of Machine Learning Research*, pages 573–605. PMLR, 2022.

Gerard Jounghyun Kim. *Human-computer interaction: fundamentals and practice*. CRC press, 2015.

Jon Kleinberg and Robert Kleinberg. Delegated search approximates efficient search. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 287–302, 2018.

Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. The challenge of understanding what users want: Inconsistent preferences and engagement optimization. *CoRR*, abs/2202.11776, 2022.

Yoav Kolumbus and Noam Nisan. How and why to manipulate your own agent: On the incentives of users of learning agents. *Advances in Neural Information Processing Systems*, 35:28080–28094, 2022.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. University of Cambridge ESOL Examinations, 2020. ISBN 9781108571401.

Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

Joshua Letchford, Vincent Conitzer, and Kamesh Munagala. Learning and approximating the optimal strategy to commit to. In *Algorithmic Game Theory, Second International Symposium, SAGT 2009, Paphos, Cyprus, October 18-20, 2009. Proceedings*, volume 5814 of *Lecture Notes in Computer Science*, pages 250–262. Springer, 2009.

Brendan Lucier, Sarath Pattathil, Aleksandrs Slivkins, and Mengxiao Zhang. Autobidders with budget and ROI constraints: Efficiency, regret, and pacing dynamics. *CoRR*, abs/2301.13306, 2023.

I. Scott MacKenzie. Human-computer interaction: An empirical research perspective. 2024.

Smitha Milli, Luca Belli, and Moritz Hardt. From optimizing engagement to measuring value. In *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 714–722. ACM, 2021.

Denis Nekipelov, Vasilis Syrgkanis, and Eva Tardos. Econometrics for learning agents. In *Proceedings of the sixteenth acm conference on economics and computation*, pages 1–18, 2015.

Noam Nisan and Gali Noti. An experimental evaluation of regret-based econometrics. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 73–81. International World Wide Web Conferences Steering Committee, 2017.

Gali Noti and Vasilis Syrgkanis. Bid prediction in repeated auctions with learning. In *Proceedings of the Web Conference 2021*, WWW '21, page 3953–3964, New York, NY, USA, 2021. Association for Computing Machinery.

Aldo Pacchiano, My Phan, Yasin Abbasi-Yadkori, Anup Rao, Julian Zimmert, Tor Lattimore, and Csaba Szepesvári. Model selection in contextual stochastic bandit problems. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Alexander Pan, Erik Jones, Meena Jagadeesan, and Jacob Steinhardt. Feedback loops with language models drive in-context reward hacking. *CoRR*, abs/2402.06627, 2024.

Jenny Preece, Yvonne Rogers, Helen Sharp, David Benyon, Simon Holland, and Tom Carey. *Human-computer interaction*. Addison-Wesley Longman Ltd., 1994.

Aleksandrs Slivkins. Introduction to multi-armed bandits. *Found. Trends Mach. Learn.*, 12(1-2): 1–286, 2019.

Eleni Straitouri and Manuel Gomez Rodriguez. Designing decision support systems using counterfactual prediction sets. *arXiv preprint arXiv:2306.03928*, 2023.

Eleni Straitouri, Lequn Wang, Nastaran Okati, and Manuel Gomez Rodriguez. Improving expert predictions with conformal prediction. In *International Conference on Machine Learning*, pages 32633–32653. PMLR, 2023.

Jonathan Stray, Ivan Vendrov, Jeremy Nixon, Steven Adler, and Dylan Hadfield-Menell. What are you optimizing for? aligning recommender systems with human values. *CoRR*, abs/2107.10939, 2021.

Lequn Wang, Thorsten Joachims, and Manuel Gomez Rodriguez. Improving screening processes via calibrated subset selection. In *International Conference on Machine Learning*, pages 22702–22726. PMLR, 2022.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.*, 10(2):12:1–12:19, 2019.

Kaiqing Zhang, Zhuoran Yang, and Tamer Basar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *CoRR*, abs/1911.10635, 2019.

Geng Zhao, Banghua Zhu, Jiantao Jiao, and Michael Jordan. Online learning in stackelberg games with an omniscient follower. In *International Conference on Machine Learning*, pages 42304–42316. PMLR, 2023.

Banghua Zhu, Stephen Bates, Zhuoran Yang, Yixin Wang, Jiantao Jiao, and Michael I. Jordan. The sample complexity of online contract design. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*, page 1188. ACM, 2023.

Simon Zhuang and Dylan Hadfield-Menell. Consequences of misaligned AI. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Tijana Zrnic, Eric Mazumdar, S. Shankar Sastry, and Michael I. Jordan. Who leads and who follows in strategic classification? In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 15257–15269, 2021.

Song Zuo and Pingzhong Tang. Optimal machine strategies to commit to in two-person repeated games. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 1071–1078. AAAI Press, 2015.

	b_1	b_2	b_3
a_1	$(1, 0.5 + 2\delta)$	$(0.7, 0.5 + \delta)$	$(1.1, 0)$
a_2	$(0.8, 3.5 \cdot \delta)$	$(1.2, 3 \cdot \delta)$	$(0.9, 4 \cdot \delta)$
a_3	$(0.5, 0.5)$	$(0.7, 0)$	$(2, 0.1)$

Table 5: Calculating the δ-tolerant benchmark: Note that (a_1, b_1) is the Stackelberg equilibrium, which by Theorem 3.1 cannot in general be learned with sublinear regret. For each row, cells shaded in blue if they are within the δ best response for the follower ($\mathcal{B}_\delta(a_i)$). Entry (a_2, b_1) (with purple text) gives the leader's δ-relaxed Stackelberg utility - the leader's best action, assuming the follower picks the worst item within the δ-response ball. Rows a_1, a_2 (shaded in red) are in $\mathcal{A}_\delta$, the set of actions where the leader has a chance of doing at least as well as the δ-relaxed Stackelberg utility ((a_2, b_1)). Finally, (a_2, b_3) (in green) gives the follower's best response, assuming the leader picks the worst action for it within $\mathcal{A}_\delta$.

	b_1	b_2	b_3
a_1	$(1, 0.5 + 2\delta)$	$(0.7, 0.5 + \delta)$	$(1.1, 0)$
a_2	$(0.8, 3.5 \cdot \delta)$	$(1.2, 3 \cdot \delta)$	$(0.9, 4 \cdot \delta)$
a_3	$(0.5, 0.5)$	$(0.7, 0)$	$(2, 0.1)$

Table 6: Calculating the **self-**δ-tolerant benchmark: Note that $\mathcal{B}_\delta, \mathcal{A}_\delta$ are defined the same as in the γ-tolerant benchmark in Table 5, so the only difference is the location of the δ-relaxed Stackelberg utility values for the leader and the follower, which are calculated by finding the *worst* expected reward for each within the $\mathcal{B}_\delta, \mathcal{A}_\delta$ sets. Here, they occur for the leader in (a_1, b_2) (in purple) and for the follower in (a_2, b_2) (in green).

A Relaxed benchmarks in example instances

A.1 Worked out version of Example 4.2 for γ-tolerant benchmark

We work out the γ-tolerant benchmark for Example 4.2 in more detail. Consider instance $\mathcal{I}$ (leftmost table) in Table 2 (with $0.4 > \gamma \geq 4\delta$), which we will use to illustrate our benchmark. We show that $\beta_1^{\text{tol}} = 0.5 + \delta$ and $\beta_2^{\text{tol}} = 4\delta$. To calculate the benchmarks, we compute the sum of the ϵ-relaxed Stackelberg value and ϵ-regularizer for different values of ϵ and then take a minimum. We will show that the minimum turns out to be achieved at $\epsilon = \delta$.

First, for $\epsilon = 0$ this benchmark is equal to the Stackelberg equilibrium, which gives values $0.5 + \delta, 0.4$ for the leader and follower respectively. For $\epsilon \in (0, \delta)$, the ϵ-relaxed Stackelberg value stays the same while the regularizer increases. For $\epsilon = \delta$, the behavior of the ϵ-Stackelberg utility becomes more complicated.

- **Follower ϵ-best-response set:** In this instance, $\mathcal{B}_\delta(a_1) = \{a_1\}$: for arm a_1, because $0.4 > \delta$, only $\{b_1\}$ is in the best-response set. However, $\mathcal{B}_\delta(a_2) = \{b_1, b_2\}$: both arms for the follower are within δ of optimal.

- **Leader ϵ-relaxed Stackelberg utility:** This term captures the best utility that the leader can achieve if the follower worst-case ϵ-best-responds according to $\text{argmin}_{b \in \mathcal{B}_\delta}(a)$. Since $\mathcal{B}_\delta(a_1) = \{b_1\}$, we see that $\min_{b \in \mathcal{B}_\delta}(a_1) = 0.5 + \delta$. However, for a_2, $\min_{b \in \mathcal{B}_\delta}(a) = v_1(a_2, b_2) = 0.4$. The leader's best action is to pick a_1, so the δ-relaxed Stackelberg utility is equal to $0.5 + \delta$.

- **Leader ϵ-best-response sets:** We construct the $\mathcal{A}_\delta$ set by considering all actions a where the *best-case* outcome within the $\mathcal{B}_\delta(a)$ gives reward at least within δ of our benchmark value of $0.5 + \delta$. We can see $\mathcal{A}_\delta = \{a_1, a_2\}$ because they both contain an item within δ of the benchmark value ((a_1, b_1) or (a_2, b_1) respectively).
- **Follower's ϵ-relaxed Stackelberg utility:** This term considers the worst-case action within $\mathcal{A}_\delta$ for the follower. If the leader picks a_1, the only response is b_1 which gives value 0.4, while if the leader picks a_2, the best response is b_2 which gives value $3 \cdot \delta$. The minimum of these, plus a regularizer term, gives a benchmark of $4 \cdot \delta$.

The above analysis shows that for $\epsilon = \delta$, the ϵ-relaxed Stackelberg utility plus the ϵ-regularizer are equal to $(0.5 + 2\delta, 4\delta)$ for the leader and follower, respectively. For $\epsilon \in (\delta, \gamma)$, the best response sets will not change, but the penalty for ϵ will increase, so these will not affect the infimum. Taking the minimum over the calculated benchmarks for $\epsilon \in \{0, \gamma\}$ gives $0.5 + \delta, 4\delta$ for the leader and follower respectively.

A.2 Worked out version of Example 4.2 for self-γ-tolerant benchmark

We work out the self-γ-tolerant benchmark for Example 4.2 in more detail. Again, consider $\mathcal{I}$ in Table 2, which we also used to illustrate the γ-tolerant benchmark in Example 4.2. Recall that for $\epsilon = 0$, we recover the Stackelberg equilibrium benchmark of $(0.5+\delta, 0.1)$ for the leader and follower, respectively. For $\epsilon \in (0, \delta)$ the $\mathcal{B}_\epsilon(a), \mathcal{A}_\epsilon$ sets don't change, but the penalty increases, so this is irrelevant for the infimum. Recall that from that analysis, we found that $\mathcal{B}_\delta(a_1) = \{b_1\}, \mathcal{B}_\delta(a_2) = \{a_1, a_2\}$, and $\mathcal{A}_\delta = \{a_1, a_2\}$. The self$-\gamma$-tolerance benchmark only requires each agent to compete with the *worst* element within the product set $\mathcal{A}_\delta \times \mathcal{B}_\delta(a)$ (if we consider the instance where $\epsilon = \delta$).

For the given instance, this gives the benchmarks for the leader and follower of $0.4 + \delta$ and $2 \cdot \delta + \delta$, where we have added a δ regularizer penalty to both. Finally, we note that for $\epsilon \in (\delta, 0.1)$, again the $\mathcal{B}_\epsilon(a), \mathcal{A}_\epsilon$ sets do not change but the penalty increases, so these are again irrelevant for the infimum. Taking the minimum of the benchmarks over $\epsilon \in \{0, \delta\}$ gives $0.4 + \delta, 3\delta$ for the leader and follower respectively. Note that this differs from the γ-tolerant benchmark for the follower only by δ, but differs by 0.1 (a constant) for the leader.

A.3 Additional worked out example for the benchmark

Tables 5 and 6 contain worked examples of how the benchmarks are calculated for more complex examples.

B Additional Notation and Auxiliary Lemmas

We introduce the following notation and auxiliary lemmas which will be convenient in our proofs.

Notation for Player Histories. First, we introduce the following notation for the player histories that will be convenient to use in algorithmic specifications and proofs.

In the *weakly decentralized setting*, let the leader's history up to time step t be the set of arms that were pulled, as well as the reward for the leader at each time step:

$$H_{1,t} := \left\{(t', a_{t'}, b_{t'}, r_{1,t'}(a_{t'}, b_{t'})) \mid 1 \le t' < t\right\}.$$

In the *strongly decentralized setting*, the leader cannot even observe the action chosen by the follower, but the follower's information remains unchanged. That is $H_{1,t} := \left\{(t', a_{t'}, r_{1,t'}(a_{t'}, b_{t'})) \mid 1 \le t' < t\right\}$.

Let the follower's history be

$$H_{2,t} := \left\{(t', a_{t'}, b_{t'}, r_{2,t'}(a_{t'}, b_{t'})) \mid 1 \le t' < t\right\}.$$

When the follower runs a separate algorithm on each choice of $a \in \mathcal{A}$ and does not share information across arms (e.g., in Proposition 4.3, Theorem 4.4, Proposition 7.3, and Proposition 7.1), then the follower's history for the arm $a \in \mathcal{A}$ is given by:

$$H_{2,t,a} := \left\{(n_{t'+1}(a), b_{t'}, r_{2,t'}(a_{t'}, b_{t'})) \mid 1 \le t' < t, a_{t'} = a\right\},$$

where $n_{t'+1}(a)$ is the number of times that arm a is pulled prior to the $(t'+1)$th time step.

Auxiliary lemma for regret analysis. Next, we introduce the following auxiliary lemma which will be useful in the regret analysis.

Lemma B.1. *Let $\mathcal{C}$ be a finite set of arms and let $T \ge 1$ be a time horizon. Let $(c_1, \ldots, c_T) \in \mathcal{C}^T$ denote any history of arm pulls. Let $n_t(c) = \sum_{t=1}^{t-1} \mathbb{1}[c_{t'} = c]$ denote the number of times that c is pulled prior to time step t. Then it holds that:*

$$\sum_{c \in \mathcal{C}} \frac{1}{\sqrt{n_t(c)}} \le O\left(\sqrt{T \cdot |\mathcal{C}|}\right)$$

Proof. We observe that

$$\sum_{c \in \mathcal{C}} \frac{1}{\sqrt{n_t(c)}} = \sum_{c \in \mathcal{C}} \sum_{n=1}^{n_t(c)} \frac{1}{\sqrt{n}} \le_{(A)} \sum_{c \in \mathcal{C}} O\left(\sqrt{n_t(c)+1}\right) \le_{(B)} O\left(\sqrt{T \cdot |\mathcal{C}|}\right) \le,$$

where (A) follows from an integral bound and (B) follows from Jensen's inequality. □

C Proofs of regret lower bounds

Our regret bounds analyze a centralized setting (Appendix C.1) and build on standard tools [Lattimore and Szepesvári, 2020] for regret lower bounds (Appendix C.2). We prove Proposition 5.4 in Appendix C.3, Theorem 3.1 in Appendix C.4, and Theorem 4.6 in Appendix C.5.

C.1 Centralized environment

When analyzing regret lower bounds, it is also convenient to consider a *centralized environment* where a single player controls the actions of both players and observes all past actions. While the centralized environment is not our primary focus, it can (informally speaking) be viewed as a limiting case of the decentralized setting with extremely sophisticated players who could communicate their strategies to each other. We define the history for the centralized environment to be:

$$H_t^{\mathrm{C}} = \left\{(t', a_{t'}, b_{t'}, r_{1,t'}(a_{t'}, b_{t'}), r_{2,t'}(a_{t'}, b_{t'})) \mid 1 \le t' \le t\right\}.$$

The centralized player chooses an algorithm `ALG` mapping a history to a joint distribution over pairs of actions.

We show that centralized algorithms are strictly more general than decentralized environments, in that any rewards realized in a decentralized environment can also be realized in a centralized environment.

Lemma C.1. *Fix an instance $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ and time horizon T. Let $\texttt{ALG}_1$, $\texttt{ALG}_2$ be any pair of decentralized algorithms operating in either the strongly decentralized setting or weakly decentralized setting. Then, there exists a centralized algorithm $\texttt{ALG}$ such that the leader rewards $(r_{1,1}(a_1, b_1), \ldots, r_{1,T}(a_T, b_T))$ are identically distributed for $\texttt{ALG}$ and $(\texttt{ALG}_1, \texttt{ALG}_2)$ and the follower rewards $(r_{2,1}(a_1, b_1), \ldots, r_{2,T}(a_T, b_T))$ are also identically distributed for $\texttt{ALG}$ and $(\texttt{ALG}_1, \texttt{ALG}_2)$.*

Lemma C.1 follows immediately from designing $\texttt{ALG}$ to "simulate" histories for the leader and the follower (by projecting away the information unavailable to each player) and then to choose arms by applying $\texttt{ALG}_1$ and $\texttt{ALG}_2$ on these histories.

C.2 Useful lemmas

Our regret bounds leverage the following standard tools [Lattimore and Szepesvári, 2020] which we restate for completeness. Like in Lattimore and Szepesvári [2020], we will use the Bretagnolle–Huber inequality.

Theorem C.2 (paraphrased from Theorem 14.2 in [Lattimore and Szepesvári, 2020])**.** *Let P and Q be probability measures on the same measurable space $(\Omega, \mathcal{F})$, and let $E \in \mathcal{F}$ be an arbitrary event. Then it holds that:*

$$P(G) + Q(G^c) \geq \frac{1}{2} e^{-KL(P,Q)}$$

where $G^c = \Omega \setminus G$ is the complement of G and $KL(P, Q)$ is the KL divergence between P and Q.

We similarly work with the canonical bandit model (Section 4.6 in Lattimore and Szepesvári [2020]) but with some modifications because there are two observed rewards (for the leader and the follower) in our setup. We call the analogous setup in our setting the *canonical centralized bandit model*. Note that the sample space of the probability space is now $(((\mathcal{A} \times \mathcal{B}) \times \mathbb{R} \times \mathbb{R})^T$ (instead of $([k] \times \mathbb{R})^T$, like in the typical canonical bandit model).

We show an analogous *divergence decomposition* (Lemma 15.1 in Lattimore and Szepesvári [2020]) applies to our setting. For this result, fix $\mathcal{A}$ and $\mathcal{B}$, and let v and $\tilde{v}$ be two different specifications of utilities. For $i \in \{1, 2\}$, let $r_i(a, b)$ denote the reward distribution $N(v_i(a, b), 1)$ and let $\tilde{r}_i(a, b)$ denote the reward distribution $N(\tilde{v}_i(a, b), 1)$.

Theorem C.3 (adapted from Lemma 15.1 in [Lattimore and Szepesvári, 2020])**.** *Fix an algorithm $\texttt{ALG}$ for the centralized environment. Let P (resp. $\tilde{P}$) denote the probability measure corresponding to the canonical centralized bandit model for $\texttt{ALG}$ applied to $(\mathcal{A}, \mathcal{B}, v)$ (resp. $(\mathcal{A}, \mathcal{B}, \tilde{v})$). Let $n_T(a, b) = \sum_{t=1}^{T} \mathbb{1}[a_t = a, b_t = b]$ denote the number of times that arm (a, b) is pulled. Then it holds that:*

$$D(P, \tilde{P}) = \sum_{(a,b) \in \mathcal{A} \times \mathcal{B}} \mathbb{E}_P[n_T(a, b)] \cdot (D(r_1(a, b), \tilde{r}_1(a, b)) + D(R_2(a, b), \tilde{r}_2(a, b)).$$

where $D(\cdot, \cdot)$ denotes the KL divergence, where $r_i(a, b)$ denotes the reward distribution $N(v_i(a, b), 1)$ and $\tilde{r}_i(a, b)$ denotes the reward distribution $N(\tilde{v}_i(a, b), 1)$ for $i = 1, 2$.

Proof. This follows from the exact same argument as the proof in Lattimore and Szepesvári [2020], where X_t is interpreted as the *pair* of rewards $(r_{1,t}(a_t, b_t), r_{2,t}(a_t, b_t))$ (or $(\tilde{r}_{1,t}(a_t, b_t), \tilde{r}_{2,t}(a_t, b_t))$ observed at time step t. Let $r(a, b)$ be the product distribution $r_1(a, b) \times r_2(a, b)$, and let $\tilde{r}(a, b)$ be the product distribution $\tilde{r}_1(a, b) \times \tilde{r}_2(a, b)$. This yields:

$$D(P, \tilde{P}) = \sum_{(a,b) \in \mathcal{A} \times \mathcal{B}} \mathbb{E}_P[n_T(a, b)] \cdot D(r(a, b), \tilde{r}(a, b)).$$

	b_1	$\dots$	b'	$\dots$
a_1	(δ, δ)	(δ, δ)	(δ, δ)	(δ, δ)
$\vdots$	$(0,0)$	$(0,0)$	$(0,0)$	$(0,0)$
a'	$(0,0)$	$(0,0)$	$*$	$(0,0)$
$\vdots$	$(0,0)$	$(0,0)$	$(0,0)$	$(0,0)$

Table 7: Hard instance for Proposition 5.4, where $*$ is equal to $(0,0)$ for instance $\mathcal{I}_{a_1,b_1}$, and $(2\delta, 2\delta)$ otherwise.

The result follows from applying the "chain rule" which implies that the KL divergence of a product distribution is the sum of KL divergences of the individual distributions:

$$D(r(a,b), \tilde{r}(a,b)) = D(r_1(a,b), \tilde{r}_1(a,b)) + D(r_2(a,b), \tilde{r}_2(a,b).$$

□

Recall that we assume Gaussian noise, which further simplifies Theorem C.3. By applying standard KL divergence bounds for univariate Gaussians, we obtain the following corollary of Theorem C.3.

Corollary C.4. *Fix an algorithm* `ALG` *for the centralized environment. Let P (resp. $\tilde{P}$) denote the probability measure corresponding to the canonical centralized bandit model for* `ALG` *applied to $(\mathcal{A}, \mathcal{B}, v)$ (resp. $(\mathcal{A}, \mathcal{B}, \tilde{v})$). Let $n_T(a,b) = \sum_{t=1}^{T} \mathbb{1}[a_t = a, b_t = b]$ denote the number of times that arm (a,b) is pulled. Then it holds that:*

$$D(P, \tilde{P}) = \sum_{(a,b) \in \mathcal{A} \times \mathcal{B}} \mathbb{E}_P[n_T(a,b)] \cdot \frac{(v_1(a,b) - \tilde{v}_1(a,b))^2 + (v_2(a,b) - \tilde{v}_2(a,b))^2}{2}.$$

C.3 Proof for Proposition 5.4

We prove Proposition 5.4, restated below.

Proposition 5.4. *Let $\mathcal{A}$ and $\mathcal{B}$ be any action sets such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 2$. Consider any algorithms* `ALG`$_1$ *and* `ALG`$_2$ *which operate in either the strongly decentralized setting or weakly decentralized setting. There exists an instance $\mathcal{I}^* = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ such that at least one of the players incurs $\Omega(\sqrt{T \cdot (|\mathcal{A}| - 1) \cdot |\mathcal{B}|})$ regret with respect to the self-γ-tolerant benchmarks $\beta_1^{\text{self-tol}}$ and $\beta_2^{\text{self-tol}}$, that is: $\max(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(\sqrt{T \cdot (|\mathcal{A}| - 1) \cdot |\mathcal{B}|})$.*

Proof of Proposition 5.4. Fix $\mathcal{A}$ and $\mathcal{B}$ such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 1$.

We define a family of instances in the centralized game and evaluate the self-tolerant benchmarks on this family of instances. Arbitrarily pick some $a_1 \in \mathcal{A}$ to be the "base" action. Let $\mathcal{F}_{\delta,\mathcal{A},\mathcal{B}}$ be the family of $(|\mathcal{A}| - 1) \cdot |\mathcal{B}| + 1$ instances of the form $(\mathcal{A}, \mathcal{B}, v_1, v_2)$ for varying settings of v_1 and v_2, where we index the instances by $(a', b') \in ((\mathcal{A} \setminus \{a_1\}) \times \mathcal{B}) \cup \{(a_1, b_1)\}$. The utility functions for the instance $\mathcal{I}_{(a',b')}$ are equal to the terms below (illustrated in Table 7):

$$v_1(a,b) = v_2(a,b) = \begin{cases} \delta & \text{if } a = a_1 \\ 0 & \text{if } (a',b') \neq (a,b), a \neq a_1 \\ 2\delta & \text{if } (a',b') = (a,b), a \neq a_1 \end{cases}$$

We claim that the $\beta_1^{\text{self-tol}} = \beta_2^{\text{self-tol}} = \delta$ for the instance $\mathcal{I}_{(a_1,b_1)}$ and $\beta_1^{\text{self-tol}} = \beta_2^{\text{self-tol}} = 2\delta$ for the instances $\mathcal{I}_{(a',b')}$ where $(a',b') \neq (a_1,b_1)$. To see this, observe that on the instance $\mathcal{I}_{(a_1,b_1)}$, it holds that $\mathcal{B}_\epsilon(a_1) = \mathcal{B}$ and $\mathcal{A}_\epsilon = \{a_1\}$ if $\epsilon < \delta$. Thus, it holds that $\min_{a\in\mathcal{A}_\epsilon} \min_{b\in\mathcal{B}_\epsilon(a)} v_1(a,b) + \epsilon \geq \delta$ for all ϵ, so the benchmark is equal to

$$\beta_1^{\text{self-tol}} = \beta_2^{\text{self-tol}} = \delta,$$

as desired. On instances $\mathcal{I}_{(a',b')}$ where $(a',b') \neq (a_1,b_1)$, it holds that $\mathcal{B}_\epsilon(a') = \{b'\}$ if $\epsilon < 2\delta$ and $\mathcal{A}_\epsilon = \{a'\}$ if $\epsilon < \delta$. If $\epsilon < \delta$ or $\epsilon \geq 2\delta$, then $\min_{a\in\mathcal{A}_\epsilon} \min_{b\in\mathcal{B}_\epsilon(a)} v_1(a,b) + \epsilon \geq 2\delta$. If $\delta \leq \epsilon < 2\delta$, then $\mathcal{A}_\epsilon = \{a', a_1\}$ and it also holds that $\min_{a\in\mathcal{A}_\epsilon} \min_{b\in\mathcal{B}_\epsilon(a)} v_1(a,b) + \epsilon \geq 2\delta$. This means that the self-tolerant benchmarks are equal to:

$$\beta_1^{\text{self-tol}} = \beta_2^{\text{self-tol}} = 2\delta,$$

as desired.

Because the utilities in $\mathcal{F}_{\delta,\mathcal{A},\mathcal{B}}$ and the benchmarks are the same for the leader and follower, we see that the regret is also the same for both players. Thus, for the remainder of the analysis, we do not need to distinguish between the regret of the leader and the regret of the follower. Let $R(T;\mathcal{I})$ denote the regret incurred on instance $\mathcal{I}$. Since the benchmarks are equal to the maximum reward across all pairs of arms, the expected regret is always nonnegative.

Fix any `ALG` for the centralized environment. For each $(a,b) \in ((\mathcal{A}\setminus\{a_1\})\times\mathcal{B}) \cup \{(a_1,b_1)\}$, let $P_{a,b}$ denote the probability measure over canonical centralized bandit model when `ALG` is applied to the instance $\mathcal{I}_{a,b}$ (see Appendix C.2). Let $n_T(a,b) = \sum_{t=1}^T \mathbb{1}[a_t = a, b_t = b]$ be the random variable denoting the number of times that (a,b) is pulled. We define:

$$(a_m, b_m) := \text{argmin}_{(a,b)\in\mathcal{A}\times\mathcal{B} | a\neq a_1} \mathbb{E}_{P_{(a_1,b_1)}}[n_T(a,b)]$$

to be the arm pulled the minimum number of times in expectation over $P_{(a_1,b_1)}$ (i.e., the expectation when `ALG` is applied to the instance $\mathcal{I}_{a_1,b_1}$). This means that

$$\mathbb{E}_{P_{(a_1,b_1)}}[n_T(a_m,b_m)] \leq \frac{T}{(|\mathcal{A}|-1)\cdot|\mathcal{B}|}.$$

We will construct δ such that the regret is high on at least one of the instances $\mathcal{I}_{(a_1,b_1)}$ and $\mathcal{I}_{(a_m,b_m)}$.

Now, let G denote the event that $\sum_{b\in\mathcal{B}} n_T(a_1,b) \leq T/2$ (i.e., the arm a_1 is pulled less than $T/2$ times). It is easy to see that the regret satisfies:

$$R(T;\mathcal{I}_{a_1,b_1}) \geq \frac{\delta\cdot T}{2}\cdot P_{a_1,b_1}[G]$$

$$R(T;\mathcal{I}_{a_m,b_m}) \geq \frac{\delta\cdot T}{2}\cdot P_{a_m,b_m}[G^c]$$

where G^c is the complement of G. We apply Theorem C.2 to see that:

$$\begin{aligned}
R(T;\mathcal{I}_{a_1,b_1}) + R(T;\mathcal{I}_{a_m,b_m}) &= \frac{\delta\cdot T}{2}\left(P_{a_1,b_1}[G] + P_{a_m,b_m}[G^c]\right) \\
&\geq_{(1)} \frac{\delta\cdot T}{2}\cdot\frac{1}{2}\exp\left(-KL(P_{a_1,b_1}, P_{a_m,b_m})\right) \\
&\geq_{(2)} \frac{\delta\cdot T}{2}\cdot\frac{1}{2}\exp\left(-\mathbb{E}_{P_{a_1,b_1}}[n_T(a_m,b_m)]\cdot(2\delta)^2\right) \\
&\geq_{(3)} \frac{\delta\cdot T}{4}\cdot\exp\left(-\frac{4\cdot\delta^2\cdot T}{(|\mathcal{A}-1)|\mathcal{B}|}\right).
\end{aligned}$$

where (1) applies Theorem C.2 and (2) applies Corollary C.4, and (3) applies the fact that $n_T(a_m, b_m) \leq \frac{T}{(|\mathcal{A}|-1)\cdot|\mathcal{B}|}$. If we set $\delta = \Theta(\sqrt{\frac{|\mathcal{A}-1)|\mathcal{B}|}{T}})$, then we obtain a bound of $\Theta(\sqrt{T \cdot (|\mathcal{A}|-1) \cdot |\mathcal{B}|})$. Since expected regret is nonnegative for these instances (see discussion above), this implies that either $R(T; \mathcal{I}_{a_1,b_1}) = \Omega(\sqrt{T \cdot (|\mathcal{A}|-1) \cdot |\mathcal{B}|})$ or $R(T; \mathcal{I}_{a_m,b_m}) = \Omega(\sqrt{T \cdot (|\mathcal{A}|-1) \cdot |\mathcal{B}|})$ as desired. □

C.4 Proof of Theorem 3.1

Theorem 3.1. *Consider any algorithms* $\texttt{ALG}_1$ *and* $\texttt{ALG}_2$ *which operate in either the strongly decentralized setting or the weakly decentralized setting. There exists an instance* $\mathcal{I}^*$ *with* $|\mathcal{A}| = |\mathcal{B}| = 2$ *where at least one of the players incurs linear regret with respect to the Stackleberg benchmarks* β_1^{orig} *and* β_2^{orig}*, that is:* $\max(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(T)$.

Proof. It suffices to prove this lower bound in a *centralized* environment where a single learner can choose action pairs (a, b) and observes rewards for both players (Lemma C.1). We construct a pair of instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$ such that at least one of the players incurs linear regret on at least one of the instances. In particular, we take $\mathcal{I}$ and $\tilde{\mathcal{I}}$ to be the instances depicted in Table 1 with $\delta = O(1/\sqrt{T})$ (reproduced here for convenience).

	b_1	b_2
a_1	$(0.6, \delta)$	$(0.2, \mathbf{0})$
a_2	$(0.5, 0.6)$	$(0.4, 0.4)$

(a) Mean rewards $(v_1(a,b), v_2(a,b))$ for $\mathcal{I}$

	b_1	b_2
a_1	$(0.6, \delta)$	$(0.2, \mathbf{2\delta})$
a_2	$(0.5, 0.6)$	$(0.4, 0.4)$

(b) Mean rewards $(\tilde{v}_1(a,b), \tilde{v}_2(a,b))$ for $\tilde{\mathcal{I}}$

We first compute the benchmarks on these two instances. On instance $\mathcal{I}$, it holds that $(a^*, b^*) = (a_1, b_1)$, $\beta_1^{\text{orig}} = 0.6$ and $\beta_2^{\text{orig}} = \delta \geq 0$. On the other hand, on instance $\tilde{\mathcal{I}}$, it holds that $(a^*, b^*) = (a_2, b_1)$, $\beta_1^{\text{orig}} = 0.5$, and $\beta_2^{\text{orig}} = 0.6$. It is easy to see that $R_1(T; \mathcal{I})$ and $R_2(T; \tilde{\mathcal{I}})$ are always *nonnegative*.

Fix any $\texttt{ALG}$ for the centralized environment. Let P (resp. $\tilde{P}$) denote the probability measure over canonical centralized bandit model when $\texttt{ALG}$ is applied to the instance $\mathcal{I}$ (resp. $\tilde{\mathcal{I}}$) (see Appendix C.2). We will show that the regret is high on at least one of the instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$.

Now let $n_T(a, b) = \sum_{t=1}^{T} \mathbb{1}[a_t = a, b_t = b]$ be the random variable denoting the number of times that (a, b) is pulled, and let G denote the event that $n_T(a_1, b_1) \leq T/2$ (i.e., the arm (a_1, b_1) is pulled less than $T/2$ times). It is easy to see that the regret satisfies:

$$R_1(T; \mathcal{I}) \geq \frac{0.1 \cdot T}{2} \cdot P[G]$$

$$R_2(T; \tilde{\mathcal{I}}) \geq \frac{(0.6 - \delta) \cdot T}{2} \cdot \tilde{P}[G^c]$$

	b_1	$\dots$	b'	$\dots$
a_1	$(0.5, 3\cdot\delta)$	$(0.5, 3\cdot\delta)$	$(0.5, 3\cdot\delta)$	$(0.5, 3\cdot\delta)$
$\vdots$	$(0.5+\delta, \delta)$	$(0,0)$	$*$	$(0,0)$
$\vdots$	$(0.5+\delta, \delta)$	$(0,0)$	$*$	$(0,0)$
$\vdots$	$(0.5+\delta, \delta)$	$(0,0)$	$*$	$(0,0)$

Table 9: Hard instance for Theorem 4.6, where $*$ is equal to $(0,0)$ for instance $\mathcal{I}_{a_1,b_1}$, and $(0, 2\delta)$ otherwise. Note that this example is structurally similar to the illustrative example in Table 3, but with $|\mathcal{A}|, |\mathcal{B}| \geq 2$.

where G^c is the complement of G. We apply Theorem C.2 to see that:

$$\begin{aligned}
R_1(T;\mathcal{I}) + R_2(T;\tilde{\mathcal{I}}) &= \frac{0.1\cdot T}{2}\cdot P[G] + \frac{(0.6-\delta)\cdot T}{2}\cdot \tilde{P}[G^c] \\
&\geq \frac{0.1\cdot T}{2}\cdot\left(P[G] + \tilde{P}[G^c]\right) \\
&\geq_{(1)} \frac{0.1\cdot T}{2}\cdot\frac{1}{2}\exp\left(-KL(P,\tilde{P})\right) \\
&\geq_{(2)} \frac{0.1\cdot T}{2}\cdot\frac{1}{2}\exp\left(-\mathbb{E}_P[n_T(a_1,b_2)]\cdot\frac{(2\cdot\delta)^2}{2}\right) \\
&\geq_{(3)} \frac{0.1\cdot T}{4}\cdot\exp\left(-2\cdot\delta^2\cdot T\right).
\end{aligned}$$

where (1) applies Theorem C.2 and (2) applies Corollary C.4, and (3) uses the fact that $n_T(a_1,b_2) \leq T$. If we take $\delta = O(T^{-1/2})$, then we obtain a bound of $\Omega(T)$. Since these expected regrets are always nonnegative (see discussion above), this implies that either $R_1(T;\mathcal{I}) = \Omega(T)$ or $R_2(T;\tilde{\mathcal{I}}) = \Omega(T)$ as desired. □

C.5 Proof of Theorem 4.6

Theorem 4.6. *Let $\mathcal{A}$ and $\mathcal{B}$ be any action sets such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 2$. Consider any algorithms $\texttt{ALG}_1$ and $\texttt{ALG}_2$ which operate in either the strongly decentralized setting or the weakly decentralized setting. There exists an instance $\mathcal{I}^* = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ such that at least one of the players incurs $\Omega(T^{2/3}\cdot(|\mathcal{B}|)^{1/3})$ regret with respect to the γ-tolerant benchmarks β_1^{tol} and β_2^{tol}:*

$$\max(R_1(T;\mathcal{I}^*), R_2(T;\mathcal{I}^*)) = \Omega(T^{2/3}\cdot(|\mathcal{B}|)^{1/3}).$$

Proof. Fix $\mathcal{A}$ and $\mathcal{B}$ such that $|\mathcal{A}| \geq 2$ and $|\mathcal{B}| \geq 2$.

We define a family of instances in the centralized game and evaluate the self-tolerant benchmarks on this family of instances. Arbitrarily pick some $(a_1, b_1) \in \mathcal{A}\times\mathcal{B}$ to be the "base" action. Let $\mathcal{F}_{\delta,\mathcal{A},\mathcal{B}}$ be the family of $|\mathcal{B}|$ instances of the form $(\mathcal{A},\mathcal{B},v_1,v_2)$ for varying settings of v, where we index the instances by $\mathcal{B}$. The utility functions for the instance $\mathcal{I}_{b'}$ are equal to terms below (illustrated in Table 9):

$$v_1(a,b) = \begin{cases} 0.5 & \text{if } a = a_1 \\ 0.5+\delta & \text{if } a \neq a_1, b = b_1 \\ 0 & \text{if } a \neq a_1, b \neq b_1. \end{cases}$$

$$v_2(a,b) = \begin{cases} 3\delta & \text{if } a = a_1 \\ \delta & \text{if } a \neq a_1, b = b_1 \\ 2\delta & \text{if } b = b', a \neq a_1, b \neq b_1 \\ 0 & \text{if } b \neq b', a \neq a_1, b \neq b_1 \end{cases}$$

We claim that the $\beta_1^{\text{tol}} = 0.5 + \delta$ and $\beta_2^{\text{tol}} = \delta$ for the instance $\mathcal{I}_{(a_1,b_1)}$ and $\beta_1^{\text{tol}} = 0.5$ and $\beta_2^{\text{tol}} = 3\delta$ for the instances $\mathcal{I}_{(a',b')}$ where $(a',b') \neq (a_1,b_1)$.

- *Instance $\mathcal{I}_{b_1}$:* If $\epsilon < \delta$, it holds that $\mathcal{B}_\epsilon(a) = \{b_1\}$ for $a \neq a_1$ and $\mathcal{A}_\epsilon = \mathcal{A} \setminus \{a_1\}$. If $\epsilon \geq \delta$, then it holds that $\mathcal{B}_\epsilon(a) = \mathcal{B}$ and $\mathcal{A}_\epsilon = \mathcal{A}$. Altogether, this means that $\beta_1^{\text{tol}} = 0.5 + \delta$ and $\beta_2^{\text{tol}} = \delta$.
- *Instances $\mathcal{I}_{b'}$ where $b' \neq b_1$:* If $\epsilon < \delta$, it holds that $\mathcal{B}_\epsilon(a) = \{b'\}$ for $a \neq a_1$ and $\mathcal{A}_\epsilon = \{a_1\}$. If $\delta \leq \epsilon < 2\delta$, then it holds that $\mathcal{B}_\epsilon(a_1) = \mathcal{B}$ and $\mathcal{B}_\epsilon(a) = \{b', b_1\}$ for $a \neq a_1$, and $\mathcal{A}_\epsilon = \mathcal{A}$. If $\epsilon \geq 2\delta$, then it holds that $\mathcal{B}_\epsilon(a) = \mathcal{B}$ and $\mathcal{A}_\epsilon = \mathcal{A}$. Altogether, this means that $\beta_1^{\text{tol}} = 0.5$ and $\beta_2^{\text{tol}} = 3\delta$.

It is easy to see that the regret $R_1(T; \mathcal{I}_{b_1})$ and the regret $R_2(T; \mathcal{I}_b)$ for $b \neq b_1$ are always nonnegative.

Fix an `ALG` be an algorithm for the centralized environment. For each $b \in \mathcal{B}$, let P_b denote the probability measure over canonical centralized bandit model when `ALG` is applied to the instance $\mathcal{I}_b$ (see Appendix C.2). Let $n_T(a,b) = \sum_{t=1}^T \mathbb{1}[a_t = a, b_t = b]$ be the random variable denoting the number of times that (a,b) is pulled. We define:

$$b_m := \text{argmin}_{b \in \mathcal{B} | b \neq b_1} \mathbb{E}_{P_{b_1}} \left[\sum_{a \neq a_1} n_T(a,b) \right]$$

to be the arm b such that the set of arms (a',b) for $a' \neq a_1$ is pulled the minimum number of times in expectation over P_{b_1} (i.e., the expectation when `ALG` is applied to the instance $\mathcal{I}_{b_1}$). This means that

$$\sum_{b \neq b_1} \sum_{a \neq a_1} \mathbb{E}_{P_{b_1}}[n_T(a,b)] \geq (|\mathcal{B}| - 1) \sum_{a \neq a_1} \mathbb{E}_{P_{b_1}}[n_T(a,b_m)].$$

We will construct δ such that the regret is high on at least one of the instances $\mathcal{I}_{b_1}$ and $\mathcal{I}_{b_m}$.

Now, let G denote the event that $\sum_{a \neq a_1} n_T(a,b_1) \leq T/2$ (i.e., arms of the form (a',b_1) for $a' \neq a$ are pulled less than $T/2$ times). It is easy to see that the regret satisfies:

$$R_1(T; \mathcal{I}_{b_1}) \geq \frac{\delta \cdot T}{2} \cdot P_{b_1}[E]$$

$$R_2(T; \mathcal{I}_{b_m}) \geq \frac{2 \cdot \delta \cdot T}{2} \cdot P_{b_m}[E^c]$$

$$R_1(T; \mathcal{I}_{b_1}) \geq (0.5 + \delta) \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b) \right] \geq 0.5 \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b) \right].$$

where G^c is the complement of G. We apply Theorem C.2 to see that:

$$
\begin{aligned}
&2 \cdot R_1(T; \mathcal{I}_{b_1}) + R_2(T; \mathcal{I}_{b_m}) \\
&= \frac{\delta \cdot T}{2} \cdot P_{b_1}[E] + \frac{2 \cdot \delta \cdot T}{2} \cdot P_{b_m}[E^c] + 0.5 \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b)\right] \\
&\geq \frac{\delta \cdot T}{2} \cdot (P_{b_1}[E] + P_{b_m}[E^c]) + 0.5 \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b)\right] \\
&\geq_{(1)} \frac{\delta \cdot T}{2} \exp\left(-KL(P_{b_1}, P_{b_m})\right) + +0.5 \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b)\right] \\
&\geq_{(2)} \frac{\delta \cdot T}{2} \cdot \frac{1}{2} \exp\left(-\mathbb{E}_{P_{b_1}}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \cdot \frac{(2\delta)^2}{2}\right) + 0.5 \cdot \mathbb{E}\left[\sum_{a \neq a_1, b \neq b_1} n_T(a,b)\right] \\
&\geq_{(3)} \frac{\delta \cdot T}{2} \cdot \frac{1}{2} \exp\left(-\mathbb{E}_{P_{b_1}}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \cdot \frac{(2\delta)^2}{2}\right) + 0.5(|\mathcal{B}|-1) \cdot \mathbb{E}\left[\sum_{b \neq b_1} n_T(a,b_m)\right]
\end{aligned}
$$

where (1) applies Theorem C.2 and (2) applies Corollary C.4 and where (3) uses the fact that $\sum_{b \neq b_1} \sum_{a \neq a_1} \mathbb{E}_P[n_T(a,b)] \geq (|\mathcal{B}|-1) \sum_{a \neq a_1} \mathbb{E}_P[n_T(a,b_m)]$.

We claim that the expression is $\Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$. We split into two cases based on the value of $\mathbb{E}\left[\sum_{a \neq a_1} n_T(a,b_m)\right]$: $\mathbb{E}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \geq \Theta(T^{2/3}(|\mathcal{B}|-1)^{-2/3})$ and $\mathbb{E}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \leq \Theta(T^{2/3}(|\mathcal{B}|-1)^{-2/3})$.

1. *Case 1:* $\mathbb{E}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \geq \Theta(T^{2/3}(|\mathcal{B}|-1)^{-2/3})$. In this case, we see that $0.5(|\mathcal{B}|-1) \cdot \mathbb{E}\left[\sum_{b \neq b_1} n_T(a,b_m)\right] = \Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$.

2. *Case 2:* $\mathbb{E}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \leq \Theta(T^{2/3}(|\mathcal{B}|-1)^{-2/3})$. In this case, we can write:

$$
\frac{\delta \cdot T}{2} \cdot \frac{1}{2} \exp\left(-\mathbb{E}_{P_{b_1}}\left[\sum_{a \neq a_1} n_T(a,b_m)\right] \cdot \frac{(2\delta)^2}{2}\right) \geq \frac{\delta \cdot T}{2} \cdot \frac{1}{2} \exp\left(-\Theta\left(T^{2/3}(|\mathcal{B}|-1)^{-2/3} \cdot \delta^2\right)\right).
$$

   In this case, we set $\delta = \Theta(T^{-1/3}(|\mathcal{B}|-1)^{1/3})$ and the expression becomes $\Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$.

This proves that $2 \cdot R_1(T; \mathcal{I}_{b_1}) + R_2(T; \mathcal{I}_{b_m}) = \Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$.

Since expected regret is nonnegative for these instances (see discussion above), this implies that either $R_1(T; \mathcal{I}_{b_1}) = \Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$ or $R_2(T; \mathcal{I}_{b_m}) = \Omega(T^{2/3}(|\mathcal{B}|-1)^{1/3})$ as desired. □

D Proofs for Section 4

D.1 Proof of Proposition 4.3

We prove Proposition 4.3.

Proposition 4.3. *Suppose that the follower runs a separate instantiation of* `ExploreThenCommit`$(E, \mathcal{B})$ *for every* $a \in \mathcal{A}$*. Moreover, suppose that the leader runs* `ExploreThenCommit`$(E' \cdot |\mathcal{B}|, \mathcal{A})$ *for any* $E' \leq E$ *(i.e., the leader's exploration phase ends before the follower's exploration phase). Then, there exists an instance* $\mathcal{I}^*$ *such that both players incur linear regret with respect to the* γ*-tolerant benchmarks* β_1^{tol} *and* β_2^{tol}*: that is,* $\min(R_1(T; \mathcal{I}^*), R_2(T; \mathcal{I}^*)) = \Omega(T)$.

This proof holds for $\gamma < 0.1$ (the construction can be generalized to other constant γ by adjusting the values of the mean rewards; we present this construction which builds on Table 2).

	b_1	b_2
a_1	$(0.6, 0.4)$	$(0.2, 0)$
a_2	$(0.5, 0.3)$	$(0.4, 0.2)$

Table 10: A single instance, illustrating the γ-tolerant benchmark - variant of Table 2 with $\delta = 0.1$

Proof. We take $\mathcal{I}^*$ to be the instance $\mathcal{I}$ in Table 10 (equivalent to Table 2 with $\delta = 0.1$).

The fact that $E' < E$ means that the leader's exploration phase takes place entirely during the follower's exploration phase. Moreover, since the leader's exploration parameter $E' \cdot |\mathcal{B}|$ is divisible by $|\mathcal{B}|$, for every arm $a \in \mathcal{A}$, the follower pulls every arm $b \in \mathcal{B}$ an equal number of times. Given that follower explores evenly between the two arms b_1 and b_2, the leader's expected average reward $\mathbb{E}[\hat{v}_{a_1}^1]$ from a_1 during the first $E' \cdot |\mathcal{B}|$ rounds is given by $(0.6 + 0.2)/2 = 0.4$ and the leader's expected average reward $\mathbb{E}[\hat{v}_{a_2}^1]$ average reward from a_2 is given by $(0.5 + 0.4)/2 = 0.45$.

The proofs boils down to analyzing the relationship between the distributions $\hat{v}_{a_1}^1$ and $v_{a_2}^1$. Note that we allow E, E' to be arbitrary, so we cannot use standard concentration bounds. Instead, we leverage the symmetry of the distribution of the empirical mean $\hat{v}_1(a_1)$ (this follows from the fact that $\hat{v}_1(a_1) - \mathbb{E}[\hat{v}_1(a_1)]$ is distributed as a Gaussian). This means that:

$$\mathbb{P}[\hat{v}_1(a_1) > 0.4] = \mathbb{P}[\hat{v}_1(a_1) < 0.4] = 0.5.$$

(The probability $\mathbb{P}[\hat{v}_1(a_1) = \mathbb{E}[\hat{v}_1(a_1)] = 0.4]$ is equal to 0.) Similarly, we see that:

$$\mathbb{P}[\hat{v}_1(a_2) > 0.45] = \mathbb{P}[\hat{v}_1(a_2) < 0.45] = 0.5.$$

Because the stochastic rewards have independent randomness, we know that with probability at least 0.25 we have $\hat{v}_1(a_1) < 0.4$ and $\hat{v}_1(a_2) > 0.45$. When this occurs, the leader commits to pulling arm a_2.

Regardless of the follower's choice of action (b_1 or b_2) in the commit phase, this means that the follower obtains reward at most 0.3 and the leader obtains reward at most 0.5. However, recall that we found that the γ-tolerant benchmark (for $\gamma = 0.1$) are $\beta_1^{\text{tol}} = 0.6$ and $\beta_2^{\text{tol}} = 0.4$. This leads to linear regret (at least $0.25 \cdot 0.1 \cdot T$) for both players, even with respect to the γ-tolerant benchmark. □

D.2 Proof of Theorem 4.4

Theorem 4.4. *Let the follower run a separate instantiation of* `ExploreThenCommit`$(E_2, \mathcal{B})$ *for every* $a \in \mathcal{A}$*, and let the leader run* `ExploreThenCommitThrowOut`$(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$*. If* $E_2 = \Theta(|\mathcal{A}|^{-2/3}|\mathcal{B}|^{-2/3} \cdot (\log T)^{1/3} T^{2/3})$*, and* $E_1 = \Theta(|\mathcal{A}|^{-2/3} \cdot (\log T)^{1/3} T^{2/3})$*, then, the regret with respect to the* γ*-tolerant benchmarks is bounded as:*

$$\max(R_1(T), R_2(T)) = O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3} T^{2/3}\right).$$

In this theorem, we will assume $\gamma = \omega\left(T^{-1/3}\left|\mathcal{A}\right|^{1/3}\left|\mathcal{B}\right|^{1/3}\cdot(\log(T)^{1/3})\right)$ (see Section 6.1 for a discussion of γ) .

Notation. We will use the following notation in the proof. For $a \in \mathcal{A}$ and $b \in \mathcal{B}$, let $\hat{v}_2(a,b)$ denote the empirical mean of observations that the follower has seen for arm a during the first $E_2 \cdot |\mathcal{A}| \cdot |\mathcal{B}|$ time steps. For $a \in \mathcal{A}$, let $\hat{v}_1(a)$ denote the empirical mean of observations that the leader has seen for arm a during the first time steps $t \in [E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + 1, E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|]$. We denote by $\tilde{b}(a) = \text{argmax}_{b\in\mathcal{B}}\hat{v}_2(a,b)$ the arm that follower has committed to for rounds $t > E_2 \cdot |\mathcal{A}| \cdot |\mathcal{B}|$ onwards. We denote by $\tilde{a} = \text{argmax}_{a\in\mathcal{A}}\hat{v}_1(a)$ the arm that the leader has committed to for rounds $t > E_1 \cdot |\mathcal{A}|$.

Clean event. We define the clean event $G := G_L \cap G_F$ to be the intersection of a clean event G_L for the leader and a clean event G_F for the follower. Informally speaking, the clean event for the leader is the event that for all arms, the empirical mean reward $\hat{v}_1(a)$ is close to the true reward $v_1(a, \tilde{b}(a))$. The event G_L is formalized as follows:

$$\forall a \in \mathcal{A} : |\hat{v}_1(a) - v_1(a, \tilde{b}(a))| \leq \frac{10\sqrt{\log T}}{\sqrt{E_1}}.$$

Similarly, informally speaking, the clean event for the follower is the event that for all arms, the empirical mean reward $\hat{v}_2(a,b)$ is close to the true reward $v_2(a,b)$. The event G_F is formalized as follows:

$$\forall a \in \mathcal{A}, b \in \mathcal{B} : |\hat{v}_2(a,b) - v_2(a,b)| \leq \frac{10\sqrt{\log T}}{\sqrt{E_2}}.$$

We prove that the clean event occurs with high probability.

Lemma D.1. *Assume the notation above. Let the follower run a separate instantiation of* `ExploreThenCommit`$(E_2, \mathcal{B})$ *for every* $a \in \mathcal{A}$*, and let the leader run* `ExploreThenCommitThrowOut`$(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$*. Then the clean event occurs with probability* $\mathbb{P}[G] \geq 1 - (|\mathcal{A}| \cdot |\mathcal{B}| + |\mathcal{A}|)T^{-3}$.

Proof. First, we consider the follower's clean event G_F. For each $a \in \mathcal{A}, b \in \mathcal{B}$, the follower has seen E_2 samples, so by a Chernoff bound, we have that

$$P\left[|\hat{v}_2(a,b) - v_2(a,b)| \geq \frac{10\sqrt{\log T}}{\sqrt{E_2}}\right] \leq T^{-3}.$$

We union bound over $a \in \mathcal{A}$ and $b \in \mathcal{B}$.

Next, we consider the leader's clean event G_L. Note that $\hat{v}_1(a)$ estimate is derived from rewards only after the follower has committed to a best response, so it is drawn from a distribution centered at $v_1(a, \tilde{b}(a))$, with E_1 samples. Again by applying a Chernoff bound, we see that

$$P\left[|\hat{v}_1(a) - v_1(a, \tilde{b}(a))| \geq \frac{10\sqrt{\log T}}{\sqrt{E_1}}\right] \leq T^{-3}.$$

We union bound over $a \in \mathcal{A}$.

Finally, we apply another union bound which leads $\mathbb{P}[G] \geq 1 - (|\mathcal{A}| \cdot |\mathcal{B}| + |\mathcal{A}|) \cdot T^{-3})$. □

We also prove the following lower bounds on the leader's utility and follower's utility from the actions $\tilde{a}$ and $\tilde{b}(\tilde{a})$ that they commit to.

Lemma D.2. *Assume the notation above. Let the follower run a separate instantiation of* `ExploreThenCommit`$(E_2, \mathcal{B})$ *for every* $a \in \mathcal{A}$*, and let the leader run* `ExploreThenCommitThrowOut`$(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$*. Suppose that the clean event* G *holds. Then, for some* $\epsilon^* = \Theta\left(\max\left(\frac{\sqrt{\log T}}{\sqrt{E_1}}, \frac{\sqrt{\log T}}{\sqrt{E_2}}\right)\right)$*, it holds that:*

$$v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \epsilon^*$$

and that

$$v_2(\tilde{a}, \tilde{b}(\tilde{a})) \geq \min_{a \in \mathcal{A}_{\epsilon^*}} \max_{b \in \mathcal{B}} v_2(a, b) - \epsilon^*.$$

Proof of Lemma D.2. We assume that the clean event G holds. We take $\epsilon^* = \Theta\left((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1/3} \cdot T^{-1/3}\right)$ with sufficiently high implicit constant.

First, we show that the follower chooses a near-optimal action for every $a \in \mathcal{A}$: that is, $v_2(a, \tilde{b}(a)) \geq \max_{b \in \mathcal{B}} v_2(a, b) - \epsilon^*$. Since G_F holds, for every $a \in \mathcal{A}$ and $b \in \mathcal{B}$, we know that $|\hat{v}_2(a, b) - v_2(a, b)| \leq \frac{10\sqrt{\log T}}{\sqrt{E_2}}$. Based on our setting of E_2 and because $\tilde{b}(a) = \text{argmax}_{b \in \mathcal{B}} \hat{v}^2_{a,b}$, it holds that:

$$v_2(a, \tilde{b}(a)) \geq \left(\max_{b \in \mathcal{B}} v_2(a, b)\right) - \frac{20\sqrt{\log T}}{\sqrt{E_2}} \geq \left(\max_{b \in \mathcal{B}} v_2(a, b)\right) - \epsilon^*,$$

as desired.

Next, we show that the leader chooses a near-optimal action: that is, $v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a)) - \epsilon^*$. Since G_L holds, we know that $|\hat{v}_1(a) - v_1(a, \tilde{b}(a))| \leq \frac{10\sqrt{\log T}}{\sqrt{E_1}}$. Based on our setting of E_2 and because $\tilde{a} = \text{argmax}_{a \in \mathcal{A}} \hat{v}_1(a)$, it holds that:

$$v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \left(\max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a))\right) - \frac{20\sqrt{\log T}}{\sqrt{E_1}} \geq \left(\max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a))\right) - \epsilon^*.$$

as desired.

To bound the leader's utility, observe that $v_2(a, \tilde{b}(a)) \geq \max_{b \in \mathcal{B}} v_2(a, b) - \epsilon^*$ implies that $b \in \mathcal{B}_{\epsilon^*}(a)$. This, coupled with the other bound, means that:

$$v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a)) - \epsilon^* \geq \left(\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b)\right) - \epsilon^*.$$

To bound the follower's utility, observe that $v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a)) - \epsilon^*$ and $v_2(a, \tilde{b}(a)) \geq \max_{b \in \mathcal{B}} v_2(a, b) - \epsilon^*$ together imply that

$$\max_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(\tilde{a}, b) \geq v_1(\tilde{a}, \tilde{b}(\tilde{a})) \geq \max_{a \in \mathcal{A}} v_1(a, \tilde{b}(a)) - \epsilon^* \geq \left(\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b)\right) - \epsilon^*,$$

which implies that $a \in \mathcal{A}_{\epsilon^*}$. This means that

$$v_2(\tilde{a}, \tilde{b}(\tilde{a})) \geq \left(\max_{b \in \mathcal{B}} v_2(\tilde{a}, b)\right) - \epsilon^* \geq \min_{a \in \mathcal{A}_{\epsilon^*}} \max_{b \in \mathcal{B}} v_2(a, b) - \epsilon^*.$$

□

We now prove Theorem 4.4.

Proof of Theorem 4.4. Assume that the clean event G holds. This occurs with probability at least $1-(|\mathcal{A}|\cdot|\mathcal{B}|+|\mathcal{A}|)T^{-3}$ (Lemma D.1), so the clean event not occuring counts negligibly towards regret.

First, we consider the first $E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|$ time steps. Each time step results in $O(1)$ regret for both players. Based on the settings of E_1 and E_2, these phases contribute a regret of:

$$E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}| = O\left(|\mathcal{A}|^{1/3}\cdot|\mathcal{B}|^{1/3}\cdot(\log T)^{1/3}\cdot T^{2/3}\right).$$

We focus on $t > E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|$ for the remainder of the analysis. Our main ingredient is Lemma D.2. Note that $\epsilon^* = \Theta\left(\max\left(\frac{\sqrt{\log T}}{\sqrt{E_1}},\frac{\sqrt{\log T}}{\sqrt{E_2}}\right)\right) = \Theta\left((|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T))^{1/3}\cdot T^{-1/3}\right)$ based on the settings of E_1 and E_2. The regret of the leader can be bounded as:

$$\begin{aligned}
&\beta_1^{\text{tol}}\cdot(T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|) - \sum_{t>E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|} v_1(a_t,b_t)\\
&\le (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\cdot\left(\max_{a\in\mathcal{A}}\min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b)+\epsilon^*\right) - \sum_{t>E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|} v_1(\tilde{a},\tilde{b}(\tilde{a}))\\
&= (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\cdot\epsilon^* + (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\left(\max_{a\in\mathcal{A}}\min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(\tilde{a},\tilde{b}(\tilde{a}))\right)\\
&\le_{(A)} 2\cdot T\cdot\epsilon^*\\
&\le O\left(T^{2/3}(\log T)^{1/3}|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}\right).
\end{aligned}$$

where (A) follows from Lemma D.2. The regret of the follower can similarly be bounded as:

$$\begin{aligned}
&\beta_1^{\text{tol}}\cdot(T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|) - \sum_{t>E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|} v_2(a_t,b_t)\\
&\le (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\cdot\left(\min_{a\in\mathcal{A}_{\epsilon^*}}\max_{b\in\mathcal{B}} v_2(a,b)+\epsilon^*\right) - \sum_{t>E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|+E_1\cdot|\mathcal{A}|} v_2(\tilde{a},\tilde{b}(\tilde{a}))\\
&= (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\cdot\epsilon^* + (T-E_2\cdot|\mathcal{B}|\cdot|\mathcal{A}|-E_1\cdot|\mathcal{A}|)\left(\min_{a\in\mathcal{A}_{\epsilon^*}}\max_{b\in\mathcal{B}} v_2(a,b) - v_2(\tilde{a},\tilde{b}(\tilde{a}))\right)\\
&\le_{(B)} 2\cdot T\cdot\epsilon^*\\
&\le O\left(T^{2/3}(\log T)^{1/3}|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}\right).
\end{aligned}$$

where (B) follows from Lemma D.2. This proves the desired result.

□

D.3 Proof of Theorem 4.5

Theorem 4.5. *Let $E = \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}|\log T)^{1/3}T^{2/3})$. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret $g(t,T,\mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}|\log T)^{1/3}T^{-1/3}\right)$ for $t > E$ and $g(t,T,\mathcal{B}) = 1$ for $t \le E$, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Then, it holds that the regret with respect to the γ-tolerant benchmarks β_1^{tol} and β_2^{tol} is bounded as:*

$$\max(R_1(T),R_2(T)) = O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3}\right).$$

We assume $\gamma = \omega\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{-1/3}\right)$.

Notation. We will use the following notation in the proof. Let $\epsilon^* = \max_{t>E} g(t, T, \mathcal{B})$. Let $\tilde{a} = \text{argmax}_{a\in\mathcal{A}} \min_{\mathcal{B}_{\epsilon^*}(a)} v_1(a, b)$ be the optimal action for the leader if the follower can worst-case ϵ^*-best-respond to any action. Let $\hat{v}_{1,t}(a)$ be the empirical mean specified in `ExplorethenUCB` at the beginning of time step t: this is the empirical mean of all observations that the leader has seen for arm a prior to time step t during the UCB phase (i.e., after time step $E\cdot|\mathcal{A}|+1$ and prior to time step t). Moreover, for each arm $a \in \mathcal{A}$, let $S(a) = \{t > E \cdot |\mathcal{A}| \mid a_t = a\}$ be the set of time steps where arm a is pulled during the UCB phase, and let $n_{E\cdot|\mathcal{A}|,t}(a) = |\{E \cdot |\mathcal{A}| < t' < t \mid a_{t'} = a\}|$ be the number of times that a is pulled during the UCB phase prior to time step t'.

Clean event. We define the clean event $G := G_L \cap G_F$ to be the intersection of a clean event G_L for the leader and a clean event G_F for the follower. Informally speaking, the clean event for the leader is the event that for all arms $a \in \mathcal{A}$ and for all time steps t, the empirical mean $\hat{v}_{1,t}(a)$ is close to the true average of the mean rewards across actions taken by b when the leader has chosen action a. The event G_L is formalized as follows:

$$\forall a \in \mathcal{A}, t \le T : \left| \frac{1}{n_{E\cdot|\mathcal{A}|,t}(a)} \sum_{E\cdot|\mathcal{A}|<t'<t|a_{t'}=a} v_1(a_{t'}, b_{t'}) - \hat{v}_{1,t}(a) \right| \le \frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,t}(a)}}.$$

The clean event G_F for the follower is the event that the follower picks an item within the ϵ^* best response set: $\forall t > E \cdot |\mathcal{A}| : b_t \in \mathcal{B}_{\epsilon^*}(a_t)$.

We first prove that the clean event G occurs with high probability.

Lemma D.3. *Assume the notation above. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret g, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Then, the event G occurs with high probability: $\mathbb{P}[G] \ge 1 - T^{-3}(|\mathcal{A}| + 1)$.*

Proof. We first show that $\mathbb{P}[G_F] \ge 1 - |\mathcal{A}| \cdot T^{-3}$. A sufficient condition for this event to hold is that:

$$\forall t > E \cdot |\mathcal{A}| : v_2(a_t, b_t) \ge \max_{b\in\mathcal{B}} v_2(a_t, b) - \max_{t>E} g(t, T, \mathcal{B}).$$

Since the exploration phases pulls every arm $a \in \mathcal{A}$ a total of E times, the high-probability instantaneous regret assumption guarantees that this event holds with probability at least $1 - |\mathcal{A}| \cdot T^{-3}$, as desired.

We next show that $\mathbb{P}[G_L] \ge 1 - T^{-3}$. This follows from a a Chernoff bound (and using the analogue of one of the canonical bandit models in Lattimore and Szepesvári [2020]) combined with a union bound.

The lemma follows from another union bound over G_L and G_F. □

Our main lemma provides, an upper bound on $\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a')-1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_t, b_t)$, which is the average of the mean rewards obtained on a' across all time steps t where a' is pulled (except for the last round), for each arm $a' \in \mathcal{A}$. In particular, we upper bound this quantity by the worst-case optimal reward under ϵ-best-responses by the follower ($\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b)$) minus the twice the size of the confidence set of a.

Lemma D.4. *Assume the notation above. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret g, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Suppose that the clean event G holds. Then it holds that:*

$$\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_t, b_t) \ge \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}}.$$

Proof. We assume that the clean event $G = G_L \cap G_F$ holds. Note that $t^* = \max(S(a'))$ denotes the last time step during which a' is chosen. Recalling that $\tilde{a} = \operatorname{argmax}_{a \in \mathcal{A}} \min_{\mathcal{B}_{\epsilon^*}(a)} v_1(a, b)$, let $S = S(\tilde{a}) \cap [E \cdot |\mathcal{A}| + 1, t^* - 1]$ be the set of time steps during the UCB phase prior to time step t^* where arm $\tilde{a}$ is pulled. We see that at the beginning of time step t^*, it holds that:

$$\begin{aligned}
\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t \in S(a') \setminus \{t^*\}} v_1(a_t, b_t) &\geq_{(1)} \hat{v}_{1,t^*}(a') - \frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \\
&\geq v_{1,t^*}^{\mathrm{UCB}}(a') - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \\
&\geq v_{1,t^*}^{\mathrm{UCB}}(\tilde{a}) - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \\
&= \hat{v}_{1,t^*}(\tilde{a}) + \frac{10\sqrt{\log T}}{\sqrt{|S|}} - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \\
&\geq_{(2)} \frac{1}{|S|} \sum_{t \in S} v_1(a_t, b_t) - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \\
&\geq_{(3)} \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}}
\end{aligned}$$

where (1) and (2) use the clean event G_L. Step (3) uses the clean event G_F which guarantees that $b_t \in \mathcal{B}_{\epsilon^*}(a_t)$ for all t, which means that for any $t \in S$, it holds that:

$$v_1(a_t, b_t) = v_1(\tilde{a}, b_t) \geq \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(\tilde{a}, b) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b)$$

as desired. □

Now we are ready to prove Theorem 4.5.

Proof of Theorem 4.5. Assume that the clean event G holds. This occurs with probability at least $1 - (1 + |\mathcal{A}|)T^{-3}$ (Lemma D.3), so the clean event not occurring counts negligibly towards regret.

The regret in the explore phase is bounded by $O(1)$ in each round, the total regret from that phase is $O(T^{2/3}|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3})$ for either player.

The remainder of the analysis boils down to bounding the regret in the UCB phase. We separately analyze the regret of the leader and the follower. Observe that $\epsilon^* = \max_{t > E} g(t, T, \mathcal{B}) = O\left(|\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{-1/3}\right)$ based on the assumption on the follower's algorithm.

Regret for the leader. We bound the regret as:

$$
\begin{aligned}
&\beta_1^{\text{tol}} \cdot (T - E \cdot |\mathcal{A}|) - \sum_{t=E\cdot|\mathcal{A}|+1}^{T} v^1_{a_t,b_t} \\
&\leq \sum_{t=E\cdot|\mathcal{A}|+1}^{T} \left(\epsilon^* + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&= \sum_{a\in\mathcal{A}} \sum_{t\in T_a} \left(\epsilon^* + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&\leq |\mathcal{A}| + \sum_{a\in\mathcal{A}} \sum_{t\in n_{E\cdot|\mathcal{A}|,T}(a)\setminus\{\max(S(a))\}} \left(\epsilon^* + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&\leq |\mathcal{A}| + \underbrace{\epsilon^* \cdot T}_{(1)} \\
&+ \underbrace{\sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \left(\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - \frac{1}{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \sum_{t\in S(a)\setminus\{\max(S(a))\}} (v_1(a_t,b_t)) \right)}_{(2)}
\end{aligned}
$$

The term $|\mathcal{A}|$ computes negligibly, term (1) is equal to $\Theta(\mathcal{A}|^{1/3}|\mathcal{B}|^{1/3}(\log T)^{1/3}T^{2/3})$, and term (2) can be bounded by:

$$
\begin{aligned}
&\sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \left(\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - \frac{1}{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \sum_{t\in S(a)\setminus\{\max(S(a))\}} \left(v^1_{a_t,b_t}\right) \right) \\
&\leq \sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a)} - 1} \\
&\leq O\left(\sqrt{\log T} \cdot \sum_{a\in\mathcal{A}} \sqrt{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \right) \\
&\leq O\left(\sqrt{|\mathcal{A}| T \log T} \right),
\end{aligned}
$$

where the first inequality uses Lemma D.4 and the last inequality uses Jensen's inequality.

Regret for the follower. Note that $\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)$ denotes the set of time steps where an action in $\mathcal{A}_{\epsilon^*}$ is chosen. We bound the regret as:

$$
\begin{aligned}
&\beta_2^{\text{tol}} \cdot (T - E \cdot |\mathcal{A}|) - \sum_{t=E\cdot|\mathcal{A}|+1}^{T} v_2(a_t,b_t) \\
&\leq \underbrace{\left(\sum_{t=E\cdot|\mathcal{A}|+1}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] \right)}_{(1)} + \underbrace{\sum_{t\in\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)} \left(\min_{a\in\mathcal{A}_{\epsilon^*}} \max_{b\in\mathcal{B}} v_2(a,b) - v_2(a_t,b_t) \right)}_{(2)} + \underbrace{\epsilon^* \cdot |\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)|}_{(3)}
\end{aligned}
$$

We first bound term (1), which can be rewritten as $\sum_{t=E\cdot|\mathcal{A}|+1}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] = \sum_{a\notin\mathcal{A}_{\epsilon^*}} n_{E\cdot|\mathcal{A}|,T}(a)$. This counts the number of times that arms outside of $\mathcal{A}_{\epsilon^*}$ are pulled during the UCB phase. The key intuition is when an arm $a_t \notin \mathcal{A}_{\epsilon^*}$, it holds that:

$$v_1(a_t, b_t) \leq \max_{b\in\mathcal{B}_{\epsilon^*}(a')} v_1(a_t, b) < \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \epsilon^*,$$

where the first inequality uses the fact that $b_t \in \mathcal{B}_{\epsilon^*}(a_t)$ (which follows from the clean event G_F) and the second inequality uses the fact that $a_t \notin \mathcal{A}_{\epsilon^*}$. This implies that for any $a' \notin \mathcal{A}_{\epsilon^*}$, the average reward across all time steps (except for the last time step) where a' is pulled satisfies:

$$\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_{t'}, b_{t'}) < \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \epsilon^*.$$

However, by Lemma D.4, we can also lower bound the average reward across all time steps (except for the last time step) where a' is pulled in terms of $n_{E\cdot|\mathcal{A}|,T}(a')$ as follows:

$$\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_{t'}, b_{t'}) \geq \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}}.$$

Putting these two inequalities together, we see that:

$$\frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \geq \epsilon^*,$$

which bounds the number of times that a' is pulled during the UCB phase as follows:

$$n_{E\cdot|\mathcal{A}|,T}(a') \leq \Theta\left(\frac{\log T}{(\epsilon^*)^2}\right) = \Theta\left((\log T)^{1/3} T^{2/3} |\mathcal{A}|^{-2/3} |\mathcal{B}|^{-2/3}\right).$$

This means that:

$$\sum_{t=E\cdot|\mathcal{A}|+1}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] = \sum_{a\notin\mathcal{A}_{\epsilon}} n_{E\cdot|\mathcal{A}|+1,T}(a) \leq \Theta\left((\log T)^{1/3} T^{2/3} |\mathcal{A}|^{1/3} |\mathcal{B}|^{-2/3}\right)$$

Next, we bound term (2):

$$\begin{aligned}
\min_{a\in\mathcal{A}_{\epsilon^*}} \max_{b\in\mathcal{B}} v_2(a, b) - \mathbb{E}[v_2(a_t, b_t)] &\leq \sum_{t\in\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)} \left(\max_{b\in\mathcal{B}} v_2(a_t, b) - \mathbb{E}[v_2(a_t, b_t)]\right) \leq |\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)| \cdot \epsilon^* \\
&\leq T \cdot \epsilon^* \\
&= \Theta\left((\log T)^{1/3} T^{2/3} |\mathcal{A}|^{1/3} |\mathcal{B}|^{1/3}\right).
\end{aligned}$$

Finally, we bound term (3) as $\epsilon^* \cdot |S| \leq T \cdot \epsilon^* = \Theta\left((\log T)^{1/3} T^{2/3} |\mathcal{A}|^{1/3} |\mathcal{B}|^{1/3}\right)$.
Putting this all together yields the desired bound. □

	b_1	b_2
a_1	$(1, 0)$	$(1-x, y)$
a_2	$(1-2x, 2y)$	$(1-3x, 3y)$

Table 11: Set $x, y \in (0, 1/3)$ to obtain an example where both players have completely inverted ordered preferences over outcomes, but for $x, y > \mathcal{O}(1/T)$ have bounded continuity.

E Proofs for Section 5

E.1 Alignment and continuity discussion

We note that constant L^* still allows for a rich space of disagreement on values. We will formalize our discussion on the distinction between requiring that the leader and the follower have the same relative ordering on every pair of (a, b) outcomes (*ordered alignment*) and that they agree on which pairs of outcomes are *sufficiently different* (*continuity*). In particular, our Lipschitz condition requires continuity, but still allows for arbitrarily misordered alignment. As an example, Table 11 gives an example where the leader and the follower have completely inverted preferences over every outcome, but have utility that is $\max\left(\frac{x}{y}, \frac{y}{x}\right)$ Lipschitz continuous.

E.2 Proofs and examples for Section 5.1

In this section, we prove Theorem 5.1, restated below for convenience.

Theorem 5.1. *Suppose that $\mathcal{I} = (\mathcal{A}, \mathcal{B}, v_1, v_2)$ has Lipschitz constant L^*. Let $\texttt{ALG}_2$ be any algorithm satisfying high-probability anytime regret $h(t, T, \mathcal{B}) = C'\sqrt{|\mathcal{B}| t \log T}$ where C' is a constant, and let $\texttt{ALG}_1 = \texttt{LipschitzUCB}(L, C'\sqrt{|B|})$ for any $L \geq L^*$. Then both players achieve the following regret bounds with respect to the original Stackelberg benchmarks β_1^{orig} and β_2^{orig}: that is, $R_1(T; \mathcal{I}) = O\left(L\sqrt{T|\mathcal{A}||\mathcal{B}| \log T}\right)$ and $R_2(T; \mathcal{I}) = O\left(L^2\sqrt{T|\mathcal{A}| \cdot |\mathcal{B}| \log T}\right)$.*

Notation. Let $\hat{v}_{1,t}(a)$ be the empirical mean specified in `LipschitzUCB` at the beginning of time step t, which is the mean of the leader's stochastic rewards $\left\{r_{1,t'}(a_{t'}, b_{t'}) \mid a_{t'} = a, 1 \leq t' < t\right\}$. We also define $\hat{v}_{1,t}(a, b)$ to be the mean of the leader's stochastic rewards for the arm (a, b) up through time step $t-1$ (the set given by $\left\{r_{1,t'}(a_{t'}, b_{t'}) \mid a_{t'} = a, b_{t'} = b, 1 \leq t' < t\right\}$). Note that this quantity is not computable by the leader since the algorithm operates in the strongly decentralized setting, but we nonetheless find it convenient to consider in the analysis. Let $n_t(a) = |1 \leq t' < t \mid a_t = a|$ be the number of times that a has been chosen prior to time step t. Let $n_t(a, b) = |1 \leq t' < t \mid a_t = a, b_t = b|$ be the number of times that (a, b) has been chosen prior to time step t. For each arm $a \in \mathcal{A}$, let $b^*(a) = \text{argmax}_{b \in \mathcal{B}} v_2(a, b)$ be the follower's best response.

Clean event. We define the clean event $G = G_L \cap G_F$ to be the intersection of a clean event G_L for the leader and a clean event G_F for the follower. Informally speaking, the clean event for the leader is the event that for all pairs of arms, the empirical mean reward $\hat{v}_{1,t}(a, b)$ is close to the true reward $v_1(a, b)$. The event G_L is formalized as follows:

$$\forall a \in \mathcal{A}, t \leq T : |\hat{v}_{1,t}(a, b) - v_1(a, b)| \leq \frac{10\sqrt{\log T}}{\sqrt{n_t(a)}}.$$

Informally speaking, the clean event for the follower is the event that the follower satisfies high-probability anytime regret bounds. The event G_F is formalized as follows:

$$\forall a \in \mathcal{A}, t \leq T : \sum_{1 \leq t' < t | a_t = a} (v_2(a, b^*(a)) - v_2(a_t, b_t)) \leq C' \sqrt{n_t(a) \log T}$$

We first prove that the clean event G occurs with high probability.

Lemma E.1. *Assume the setup of Theorem 5.1 and the notation above. Then the clean event occurs with high probability:* $\mathbb{P}[G] \geq 1 - T^{-3}(|\mathcal{A}| + 1)$.

Proof. We union bound over G_L and G_F. The analysis for G_F follows from the high-probability anytime regret bound assumption. The analysis for G_L follows from a Chernoff bound (and using the analogue of one of the canonical bandit models in Lattimore and Szepesvári [2020]) combined with a union bound. □

The following lemma guarantees, for each arm $a \in \mathcal{A}$, that the empirical mean $\hat{v}_{1,t}(a)$ is close to the mean reward if the follower were to best-respond $\max_{b \in \mathcal{B}} v_1(a, b)$. Conceptually speaking, this lemma guarantees that the confidence sets for the leader are always "correct".

Lemma E.2. *Assume the setup of Theorem 5.1 and the notation above. Suppose that the clean event G holds. Then for any $t \leq T$ and $a \in \mathcal{A}$, it holds that:*

$$|\hat{v}_{1,t}(a) - v_1(a, b^*(a))| \leq \frac{10\sqrt{\mathcal{B} \log T}}{\sqrt{n_t(a)}} + C' \cdot L \cdot \frac{\sqrt{|\mathcal{B}| \log T}}{\sqrt{n_t(a)}}.$$

Proof. We observe that:

$$\begin{aligned}
|\hat{v}_{1,t}(a) - v_1(a, b^*(a))| &= \left| \left(\frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot \hat{v}_1(a, b) \right) - v_1(a, b^*(a)) \right| \\
&= \left| \left(\frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot \hat{v}_1(a, b) \right) - \frac{1}{n_t(a)} \left(\sum_{b \in \mathcal{B}} n_t(a, b) \cdot v_1(a, b^*(a)) \right) \right| \\
&\leq \frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot |\hat{v}_1(a, b) - v_1(a, b^*(a))| \\
&\leq \underbrace{\frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot |\hat{v}_1(a, b) - v_1(a, b)|}_{(A)} + \underbrace{\frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot |v_1(a, b) - v_1(a, b^*(a))|}_{(B)}.
\end{aligned}$$

First, we will bound term (A), which relates the error of the estimate of $v_1(a, b)$. We see that:

$$\begin{aligned}
\frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot |\hat{v}_1(a, b) - v_1(a, b)| &\leq_{(1)} \frac{1}{n_t(a)} \sum_{b \in \mathcal{B}} n_t(a, b) \cdot \frac{10\sqrt{\log T}}{\sqrt{n_t(a, b)}} \\
&= \frac{10\sqrt{\log T}}{n_t(a)} \sum_{\sum_{b \in \mathcal{B}}} \sqrt{n_t(a, b)} \\
&\leq_{(2)} \frac{10\sqrt{|\mathcal{B}| \log T}}{\sqrt{n_t(a)}}.
\end{aligned}$$

where (1) uses the clean event G_L and (2) uses Jensen's inequality.

Term (B) represents represents the difference in the leader's utility between the arm chosen by the follower and the follower's best-response. We can bound this as:

$$\begin{aligned}\frac{1}{n_t(a)}\sum_{b\in\mathcal{B}} n_t(a,b)\cdot|v_1(a,b)-v_1(a,b^*(a))| &\leq_{(1)} \frac{L^*}{n_t(a)}\sum_{b\in\mathcal{B}} n_t(a,b)\cdot|v_2(a,b)-v_2(a,b^*(a))| \\ &=_{(2)} \frac{L^*}{n_t(a)}\sum_{b\in\mathcal{B}} n_t(a,b)\cdot(v_2(a,b^*(a))-v_2(a,b))\,,\end{aligned}$$

where (1) uses the Lipschitz property and (2) uses the fact that $b^*(a)$ is the best arm for the follower, given that the leader pulls arm a. Using the clean event G_F and that $L \geq L^*$, we see that:

$$\frac{L^*}{n_t(a)}\sum_{b\in\mathcal{B}} n_t(a,b)\cdot(v_2(a,b^*(a))-v_2(a,b)) = \frac{L^*}{n_t(a)}\sum_{1\leq t'<t|a_t=a}(v_2(a,b^*(a))-v_2(a_t,b_t)) \leq C'\cdot L\frac{\sqrt{|\mathcal{B}|\log T}}{\sqrt{n_t(a)}}.$$

Taken together, these terms give the desired bound. □

It will also be convenient to bound the following two quantities which surface in our regret analysis. At a conceptual level, B_1 captures the sum of the sizes of the confidence sets of the arms pulled by the leader, and the term B_2 captures the cumulative suboptimality of the follower relative to the action a that they are provided in each time step.

Lemma E.3. *Assume the setup of Theorem 5.1 and the notation above. Suppose that the clean event G holds. Then it holds that:*

$$\begin{aligned}B_1 &:= \sum_{t=1}^{T}\left(\frac{10\sqrt{\mathcal{B}\log T}}{\sqrt{n_t(a)}} + C'\cdot L\cdot\frac{\sqrt{|\mathcal{B}|\log T}}{\sqrt{n_t(a)}}\right) \leq O\left(L\sqrt{T|\mathcal{A}||\mathcal{B}|\log T}\right) \\ B_2 &:= \sum_{t=1}^{T}(v_2(a_t,b^*(a_t))-v_2(a_t,b_t)) \leq O\left(\sqrt{T|\mathcal{A}||\mathcal{B}|\log T}\right)\end{aligned}$$

Proof. To bound B_2, we see that:

$$\begin{aligned}B_2 &= \sum_{t=1}^{T}(v_2(a_t,b^*(a_t))-v_2(a_t,b_t)) \\ &= \sum_{a\in\mathcal{A}}\sum_{t\in T|a_t=a}(v_1(a,b^*(a))-v_1(a,b_t)) \\ &\leq_{(A)} \sum_{a\in\mathcal{A}} C'\cdot\sqrt{|\mathcal{B}|\cdot n_t(a)\log T} \\ &= C'\cdot\sqrt{\mathcal{B}|\log T}\cdot\sum_{a\in\mathcal{A}}\sqrt{n_t(a)} \\ &\leq_{(B)} C'\cdot\sqrt{T|\mathcal{A}||\mathcal{B}|\log T},\end{aligned}$$

where (A) uses the event G_F and (B) uses Jensen's inequality.

To bound B_1, we note that we must upper bound this both with a) the gap of the confidence interval, as well as b) the error on the leader's estimates of their value for arm a. Taken together, this yields;

$$\begin{aligned}
B_1 &= \sum_{t=1}^{T} \left(\frac{10\sqrt{\mathcal{B}\log T}}{\sqrt{n_t(a)}} + C' \cdot L \cdot \frac{\sqrt{|\mathcal{B}|\log T}}{\sqrt{n_t(a)}} \right) \\
&= \sum_{t=1}^{T} \frac{10\sqrt{\mathcal{B}\log T}}{\sqrt{n_t(a)}} + \sum_{t=1}^{T} C' \cdot L \cdot \frac{\sqrt{|\mathcal{B}|\log T}}{\sqrt{n_t(a)}} \\
&\leq (10\sqrt{|\mathcal{B}\log T} + C' \cdot L\sqrt{|\mathcal{B}|\log T}) \sum_{t=1}^{T} \frac{1}{\sqrt{n_t(a)}} \\
&\leq_{(A)} (10\sqrt{|\mathcal{B}\log T} + C' \cdot L\sqrt{|\mathcal{B}|\log T}) \cdot (2 \cdot \sqrt{T|\mathcal{A}|} + |\mathcal{A}|) \\
&= O\left(L\sqrt{T|\mathcal{A}||\mathcal{B}|\log T} \right).
\end{aligned}$$

where (A) follows from Lemma B.1

□

We now prove Theorem 5.1.

Proof of Theorem 5.1. Assume that clean event G holds. This occurs with probability at least $1 - (|\mathcal{A}+1)T^{-3}$ (Lemma E.1), so the clean event not occurring counts negligibly towards regret.

Moreover, let $(a^*, b^*(a^*))$ be the Stackelberg equilibrium. Let $\alpha_t(a) = \frac{10\sqrt{\mathcal{B}\log T}}{\sqrt{n_t(a)}} + C \cdot L \cdot \frac{\sqrt{\log T}}{\sqrt{n_t(a)}}$ be the confidence bound size at time step t and let $v_{1,t}^{\mathrm{UCB}}(a) = \hat{v}_{1,t}(a) + \alpha_t(a)$ denote the UCB estimate in `LipschitzUCB`(L, C) computed during time step t prior to reward at time step t being observed.

We can bound the leader's regret as:

$$\begin{aligned}
R_1(T) &= \sum_{t=1}^{T} (v_1(a^*, b^*(a^*)) - v_1(a_t, b_t)) \\
&= \sum_{t=1}^{T} (v_1(a^*, b^*(a^*)) - v_1(a_t, b^*(a_t))) + \sum_{t=1}^{T} (v_1(a_t, b^*(a_t)) - v_1(a_t, b_t)) \\
&\leq_{(A)} \sum_{t=1}^{T} (\hat{v}_1(a^*) + \alpha_t(a^*) - \hat{v}_1(a_t) + \alpha_t(a_t)) + \sum_{t=1}^{T} |v_1(a_t, b^*(a_t)) - v_1(a_t, b_t)| \\
&\leq \sum_{t=1}^{T} \left(v_{1,t}^{\mathrm{UCB}}(a^*) - v_1^{\mathrm{UCB}}(a_t) + 2 \cdot \alpha_t(a_t) \right) + L \cdot \sum_{t=1}^{T} |v_2(a_t, b^*(a_t)) - v_2(a_t, b_t)| \\
&\leq 2 \cdot \sum_{t=1}^{T} \alpha_t(a_t) + L \cdot \sum_{t=1}^{T} (v_2(a_t, b^*(a_t)) - v_2(a_t, b_t)) \\
&= 2 \cdot \sum_{t=1}^{T} \left(\frac{10\sqrt{\mathcal{B}\log T}}{\sqrt{n_t(a)}} + C' \cdot L \cdot \frac{\sqrt{|\mathcal{B}|\log T}}{\sqrt{n_t(a)}} \right) + L \cdot B_2 \\
&= 2 \cdot B_1 + L \cdot B_2 \\
&\leq_{(B)} O\left(L\sqrt{T|\mathcal{A}||\mathcal{B}|\log T} \right)
\end{aligned}$$

where (A) uses Lemma E.2 and (B) uses Lemma E.3.

We also bound the follower's regret as:

$$\begin{aligned}
R_2(T) &= \sum_{t=1}^{T} \left(v_2(a^*, b^*(a^*)) - v_2(a_t, b_t)\right) \\
&= \sum_{t=1}^{T} \left(v_2(a^*, b^*(a^*)) - v_2(a^*, b^*(a_t))\right) + \sum_{t=1}^{T} \left(v_2(a^*, b^*(a_t)) - v_2(a_t, b_t)\right) \\
&= \sum_{t=1}^{T} L \cdot |v_1(a^*, b^*(a^*)) - v_1(a_t, b^*(a_t))| + B_2 \\
&=_{(A)} \sum_{t=1}^{T} L \cdot \left(v_1(a^*, b^*(a^*)) - v_1(a_t, b^*(a_t))\right) + B_2 \\
&\leq_{(B)} \sum_{t=1}^{T} L \cdot \left(\hat{v}_{1,t}(a^*) + \alpha_t(a^*) - \hat{v}_{1,t}(a_t) + \alpha_t(a^*)\right) + B_2 \\
&= \sum_{t=1}^{T} L \cdot \left(v_{1,t}^{\mathrm{UCB}}(a^*) - v_{1,t}^{\mathrm{UCB}}(a_t) + 2 \cdot \alpha_t(a_t)\right) + B_2 \\
&\leq \sum_{t=1}^{T} L \cdot \left(2 \cdot \alpha_t(a_t)\right) + B_2 \\
&= 2L \cdot \sum_{t=1}^{T} \left(\frac{10\sqrt{\mathcal{B} \log T}}{\sqrt{n_t(a)}} + C' \cdot L \cdot \frac{\sqrt{|\mathcal{B}| \log T}}{\sqrt{n_t(a)}}\right) + B_2 \\
&= 2L \cdot B_1 + B_2 \\
&\leq_{(C)} O\left(L^2 \sqrt{T|\mathcal{A}||\mathcal{B}| \log T}\right)
\end{aligned}$$

where (A) uses the fact that a^* is the action chosen by the leader at the Stackelberg equilibrium where (B) uses Lemma E.2 and (C) uses Lemma E.3.

□

E.3 Proof of Theorem 5.3

We prove Theorem 5.3, restated below.

Theorem 5.3. *Suppose that for each $a \in \mathcal{A}$, the algorithm $\texttt{ALG}_2$ runs a separate instantiation of $\texttt{ActiveArmElimination}$ with parameters $M_1, \ldots, M_P$ (where $M_i = \Theta(\log T \cdot 2^{2i})$ denotes the number of times that each arm is pulled in phase i). Let $\texttt{ALG}_1 = \texttt{PhasedUCB}(M_1, \ldots, M_P)$. Then it holds that the regret with respect to the self-γ-tolerant benchmarks $\beta_1^{\text{self-tol}}$ and $\beta_2^{\text{self-tol}}$ is bounded as:*

$$\max(R_1(T), R_2(T)) = O\left(\sqrt{|\mathcal{A}| \cdot |\mathcal{B}| \cdot T \cdot \log T}\right).$$

This theorem assumes that $\gamma = \Omega\left(T^{-1/4} \sqrt{|\mathcal{A}||\mathcal{B}| \cdot \log T}\right)$.

Notation. Let $\hat{v}_{1,t}(a, b)$ denote the empirical mean of the leader's observed rewards

$$\left\{r_{1,t'}(a, b) \mid 1 \leq t' < t, a_t = a, b_t = b\right\}$$

for (a, b) up to time step t. (The leader can observe this information because this algorithm operates in the weakly decentralized setting.) Let $v_{1,t}^{\mathrm{UCB}}(a, b)$ denote the UCB estimate in `PhasedUCB` during time step t. Let $n_t(a) = |\{1 \le t' < t \mid a_t = a\}|$ be the number of times that arm a is pulled before time step t. Let $n_t(a, b) = |\{1 \le t' < t \mid a_t = a, b_t = b\}|$ be the number of times that arms (a, b) are pulled before time step t. Let C be a constant such that `ActiveArmElimination` has high-probability instantaneous regret $g(t, T, \mathcal{B}) = C \cdot \sqrt{|\mathcal{B}| \log T/t)}$ (such a constant C exists by Proposition 7.1). Let $\mathcal{B}_t(a)$ be the computation of the active set at line 3 of `PhasedUCB` during time step t. Let $s_t(a)$ be the value of the variable $s'(a)$ at the end of the `ComputeActiveArms` algorithm, when it is called at the beginning of time step t in `PhasedUCB`. Let (a^*, b^*) be the Stackelberg equilibrium.

Clean event. We define the clean event $G := G_L \cap G_F \cap G_{L,F}$ to be the intersection of a clean event G_L for the leader, a clean event G_F for the follower, and a clean event $G_{L,F}$ for the follower (using the leader's assessment of the follower). Informally speaking, the clean event G_L for the leader is the event that the empirical mean $\hat{v}_1(a, b)$ is always sufficiently close to the true mean reward $v_1(a, b)$. We formalize the clean event G_L as follows:

$$\forall t \in T, a \in \mathcal{A}, b \in \mathcal{B} : |\hat{v}_{1,t}(a, b) - v_1(a, b)| \le \frac{10\sqrt{\log T}}{\sqrt{n_t(a, b)}}.$$

The clean event G_F for the follower is the event that the follower satisfies the high-probability instantaneous regret guarantee:

$$\forall t \le T : \left| v_2(a_t, b_t) - \max_{b \in \mathcal{B}} v_2(a_t, b) \right| \le C \cdot \frac{\sqrt{|\mathcal{B}| \log T}}{\sqrt{n_t(a)}}.$$

The final clean event $G_{L,F}$ is the event that the active arm set $\mathcal{B}_t(a^*)$ for the Stackelberg action always contains the follower's best-response:

$$\forall t \in T, b \in \mathcal{B} : \mathrm{argmax}_{b \in \mathcal{B}} v_2(a^*, b) \in \mathcal{B}_t(a^*).$$

Lemma E.4. *Assume the setup of Theorem 5.3 and notation above. Then the clean event G occurs with high probability: $\mathbb{P}[G] \ge 1 - (2 \cdot |\mathcal{A}| + 1) \cdot T^{-3}$.*

Proof. We union bound for G_F, G_L, and $G_{L,F}$. The analysis for G_L follows from a Chernoff bound (and using the analogue of one of the canonical bandit models in Lattimore and Szepesvári [2020]) combined with a union bound. The analysis for G_F follows from Proposition 7.1. The analysis for $G_{L,F}$ follows from standard properties of `ActiveArmElimination` (e.g., see Lattimore and Szepesvári [2020]) combined with a union bound over $\mathcal{A}$. □

The first lemma shows that if the follower runs ActiveArmElimination, for every $a \in \mathcal{A}$ and $b \in \mathcal{B}_t(a)$, we can upper and lower bound the number of pulls $n_t(a, b)$ in terms of the last phase that the follower has completed (as assessed by the leader).

Lemma E.5. *Assume the setup of Theorem 5.3 and notation above. Then for every time step t, and every $a \in \mathcal{A}$ and $b \in \mathcal{B}_t(a)$, it holds that:*

$$n_t(a, b) \in \left[\sum_{i=1}^{s_t(a)} M_i, \sum_{i=1}^{s_t(a)+1} M_i + 1 \right]$$

Proof. This follows from the implementation of `ComputeActiveArms` combined with the specification of `ActiveArmElimination`, which guarantees that the follower has finished phase $s_t(a)$ by the end of round $t-2$ and is at most one step into phase $s_t(a)+2$. □

The next lemma guarantees that at every time step t, the chosen pair of actions (a_t, b_t) are in the ϵ_t-best-response sets for each player, where ϵ_t depends on the number of times $n_t(a_t)$ that arm a_t has been chosen so far.

Lemma E.6. *Assume the setup of Theorem 5.3 and notation above. Suppose that the clean event G holds. Then for every time step t, it holds that for*

$$v_1(a_t, b_t) \geq \min_{a \in \mathcal{A}^1_{\epsilon_t}} \min_{b \in \mathcal{B}_{\epsilon_t}(a)} v_1(a, b)$$

$$v_2(a_t, b_t) \geq \min_{a \in \mathcal{A}^1_{\epsilon_t}} \min_{b \in \mathcal{B}_{\epsilon_t}(a)} v_2(a, b).$$

for $\epsilon_t = \Theta(\sqrt{|\mathcal{B}| \cdot \log T / n_t(a_t)})$.

Proof. It suffices to show that $a_t \in \mathcal{A}_{\epsilon_t}$ and $b_t \in \mathcal{B}_{\epsilon_t}(a_t)$.

By the clean event G_F, it immediately follows that $b_t \in \mathcal{B}_{\epsilon_t}(a_t)$.

To show that $a_t \in \mathcal{A}_{\epsilon_t}$, it suffices to show that $\max_{b \in \mathcal{B}_{\epsilon_t}(a_t)} v_1(a_t, b) \geq \max_{a' \in \mathcal{A}} \min_{b' \in \mathcal{B}_{\epsilon}(a')} v_1(a', b') - \epsilon_t$, which can be written as $\max_{a' \in \mathcal{A}} \min_{b' \in \mathcal{B}_{\epsilon_t}(a')} v_1(a', b') \leq \max_{b \in \mathcal{B}_{\epsilon_t}(a_t)} v_1(a, b) + \epsilon_t$. To see this, observe that:

$$\begin{aligned}
\max_{a' \in \mathcal{A}} \min_{b' \in \mathcal{B}_{\epsilon_t}(a')} v_1(a', b') &\leq v_1(a^*, b^*) \\
&\leq_{(A)} \max_{b \in \mathcal{B}'(a^*)} v_{1,t}^{\mathrm{UCB}}(a^*, b) \\
&\leq \max_{b \in \mathcal{B}'_t(a_t)} v_{1,t}^{\mathrm{UCB}}(a_t, b) \\
&\leq_{(B)} \max_{b \in \mathcal{B}'_t(a_t)} \left(v_1(a_t, b) + 20 \cdot \sqrt{\frac{\log T}{n_t(a_t, b)}} \right) \\
&\leq_{(C)} \max_{b \in \mathcal{B}'_t(a_t)} \left(v_1(a_t, b) + 20 \cdot \sqrt{\frac{\log T}{\sum_{i=1}^{s_t(a)} M_i}} \right) \\
&\leq_{(D)} \max_{b \in \mathcal{B}'_t(a_t)} (v_1(a_t, b)) + \Theta\left(\sqrt{\frac{|\mathcal{B}| \log T}{n_t(a_t)}} \right) \\
&\leq \max_{b \in \mathcal{B}'_t(a_t)} v_1(a_t, b) + \epsilon_t \\
&\leq_{(E)} \max_{b \in \mathcal{B}_{\epsilon_t}(a_t)} v_{1,t}^{\mathrm{UCB}}(a_t, b) + \epsilon_t.
\end{aligned}$$

where (A) uses the event $G_{L,F}$, (B) uses the event G_L, (C) applies the lower bound in Lemma E.5, (D) uses the upper bound in Lemma E.5 to see that:

$$n_t(a_t) \leq \sum_{b \in \mathcal{B}} n_t(a_t, b) \leq \sum_{b \in \mathcal{B}} \left(\left(\sum_{i=1}^{s_t(a)+1} M_i \right) + 1 \right) \leq \Theta\left(|\mathcal{B}| \cdot \sum_{i=1}^{s_t(a)} M_i \right)$$

since every arm is pulled and (E) uses the clean event G_F. □

Now, we prove Theorem 5.3.

Proof of Theorem 5.3. Assume that the clean event G occurs. This occurs with probability at least $1-(2\cdot|\mathcal{A}|+1)\cdot T^{-3}$ (Lemma E.4), so the clean event not occurring counts negligibly towards regret.

We apply Lemma E.6 to see that at time step t, it holds that for $\epsilon_t = \Theta(\sqrt{|\mathcal{B}|\cdot\log T/n_t(a_t)})$, it holds that

$$v_1(a_t,b_t) \geq \min_{a\in\mathcal{A}_{\epsilon_t}} \min_{b\in\mathcal{B}_{\epsilon_t}(a)} v_1(a,b)$$

$$v_2(a_t,b_t) \geq \min_{a\in\mathcal{A}_{\epsilon_t}} \min_{b\in\mathcal{B}_{\epsilon_t}(a)} v_2(a,b).$$

For the leader, this implies that:

$$\begin{aligned}
R_1(T) &= \beta_1^{\text{self-tol}}\cdot T - \sum_{t=1}^{T} v_1(a_t,b_t) \\
&\leq \sum_{t=1}^{T}\left(\epsilon_t + \min_{a\in\mathcal{A}_{\epsilon_t}}\min_{b\in\mathcal{B}_{\epsilon_t}(a)} v_1(a,b) - \sum_{t=1}^{T} v_1(a_t,b_t)\right) + \sum_{t=1}^{T}\mathbb{1}[\epsilon_t>\gamma] \\
&\leq \left(\sum_{t=1}^{T}\epsilon_t\right) + \sum_{t=1}^{T}\mathbb{1}[\epsilon_t>\gamma].
\end{aligned}$$

For the follower, this similarly implies that:

$$\begin{aligned}
R_2(T) &= \beta_2^{\text{self-tol}}\cdot T - \sum_{t=1}^{T} v_2(a_t,b_t) \\
&\leq \sum_{t=1}^{T}\left(\epsilon_t + \min_{a\in\mathcal{A}_{\epsilon_t}}\min_{b\in\mathcal{B}_{\epsilon_t}(a)} v_2(a,b) - \sum_{t=1}^{T} v_2(a_t,b_t)\right) \\
&\leq \left(\sum_{t=1}^{T}\epsilon_t\right) + \sum_{t=1}^{T}\mathbb{1}[\epsilon_t>\gamma].
\end{aligned}$$

To bound $\sum_{t=1}^{T}\epsilon_t$, we observe that:

$$\begin{aligned}
\sum_{t=1}^{T}\epsilon_t &= \sum_{t=1}^{T}\Theta\left(\sqrt{\frac{|\mathcal{B}|\cdot\log T}{n_t(a_t)}}\right) \\
&= \Theta\left(\sqrt{|\mathcal{B}|\cdot\log T}\cdot\sum_{t=1}^{T}\frac{1}{\sqrt{n_t(a_t)}}\right) \\
&\leq_{(A)} O\left(\sqrt{|\mathcal{B}|\cdot\log T}\cdot\sqrt{|\mathcal{A}|\cdot T}\right)
\end{aligned}$$

where (A) follows from Lemma B.1. This gives the desired upper bound.

To bound $\sum_{t=1}^{T}\mathbb{1}[\epsilon_t>\gamma]$, based on the setting of ϵ_t, we observe that $\epsilon_t\leq\gamma$ when $n_{a_t} = O\left(\frac{|\mathcal{B}|\cdot(\log T)}{\epsilon_t^2}\right)$. This means that $\mathbb{1}[\epsilon_t>\gamma]$ occurs in at most $\Theta\left(\frac{|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T)}{\gamma^2}\right)$ time steps. As long as $\gamma = \Omega\left(T^{-1/4}\sqrt{|\mathcal{A}||\mathcal{B}|\cdot\log T}\right)$, this term contributes $O\left(\sqrt{|\mathcal{B}|\cdot\log T}\cdot\sqrt{|\mathcal{A}|\cdot T}\right)$ to regret.

□

F Proofs for Section 6

F.1 Proof of Theorem 6.2

Theorem 6.2. *Suppose that $c \geq 1$ and $d \leq 1$, and let $\eta := 2/(2+d)$. Let the follower run a separate instantiation of* `ExploreThenCommit`$(E_2, \mathcal{B})$ *for every $a \in \mathcal{A}$, and let the leader run* `ExploreThenCommitThrowOut`$(E_1, E_2 \cdot |\mathcal{B}|, \mathcal{A})$. *If $E_2 = \Theta(|\mathcal{A}|^{-\eta}|\mathcal{B}|^{-\eta} \cdot (\log T)^{1-\eta}(c \cdot T)^{\eta})$, and $E_1 = \Theta(|\mathcal{A}|^{-\eta} \cdot (\log T)^{1-\eta}(c \cdot T)^{\eta})$, then the leader and follower regret with respect to the generalized (c, d, γ)-tolerant benchmarks are both at most:*

$$\max(R_1(T), R_2(T)) = O\big((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta}\big).$$

The proof follows a similar argument to the proof of Theorem 4.4 and borrows some lemmas from Appendix D.2

Proof of Theorem 6.2. Assume that the clean event G holds. This occurs with probability at least $1 - (|\mathcal{A}| \cdot |\mathcal{B}| + |\mathcal{A}|)T^{-3}$ (Lemma D.1), so the clean event not occuring counts negligibly towards regret.

First, we consider the first $E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|$ time steps. Each time step results in $O(1)$ regret for both players. Based on the settings of E_1 and E_2, these phases contribute a regret of:

$$E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}| = O\left(|\mathcal{A}|^{1-\eta} \cdot |\mathcal{B}|^{1-\eta} \cdot (\log T)^{1-\eta}(c \cdot T)^{\eta}\right).$$

We focus on $t > E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|$ for the remainder of the analysis. Our main ingredient is Lemma D.2. Note that $\epsilon^* = \Theta\left(\max\left(\frac{\sqrt{\log T}}{\sqrt{E_1}}, \frac{\sqrt{\log T}}{\sqrt{E_2}}\right)\right) = \Theta\left((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{\eta/2} \cdot (c \cdot T)^{-\eta/2}\right)$ based on the settings of E_1 and E_2. The regret of the leader can be bounded as:

$$\begin{aligned}
&\beta_1^{\text{tol}} \cdot (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) - \sum_{t > E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|} v_1(a_t, b_t) \\
&\leq (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \cdot \left(\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b) + c \cdot (\epsilon^*)^d\right) - \sum_{t > E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|} v_1(\tilde{a}, \tilde{b}(\tilde{a})) \\
&= (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \cdot c \cdot (\epsilon^*)^d + (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \left(\max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - v_1(\tilde{a}, \tilde{b}(\tilde{a}))\right) \\
&\leq_{(A)} T \cdot c \cdot (\epsilon^*)^d + T \cdot \epsilon^* \\
&\leq_{(B)} T \cdot c \cdot (\epsilon^*)^d \\
&\leq \Theta\left((c \cdot T)^{1-(\eta \cdot d/2)} \cdot (|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{\eta \cdot d/2}\right) \\
&= \Theta\left((c \cdot T)^{\eta} \cdot (|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta}\right).
\end{aligned}$$

where (A) follows from Lemma D.2 and (B) uses the fact that $c \geq 1$ and $d \leq 1$. The regret of the

follower can similarly be bounded as:

$$\begin{aligned}
&\beta_1^{\text{tol}} \cdot (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) - \sum_{t > E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|} v_2(a_t, b_t) \\
&\leq (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \cdot \left(\min_{a \in \mathcal{A}_{\epsilon^*}} \max_{b \in \mathcal{B}} v_2(a, b) + c \cdot (\epsilon^*)^d \right) - \sum_{t > E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| + E_1 \cdot |\mathcal{A}|} v_2(\tilde{a}, \tilde{b}(\tilde{a})) \\
&= (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \cdot c \cdot (\epsilon^*)^d + (T - E_2 \cdot |\mathcal{B}| \cdot |\mathcal{A}| - E_1 \cdot |\mathcal{A}|) \left(\min_{a \in \mathcal{A}_{\epsilon^*}} \max_{b \in \mathcal{B}} v_2(a, b) - v_2(\tilde{a}, \tilde{b}(\tilde{a})) \right) \\
&\leq_{(B)} T \cdot c \cdot (\epsilon^*)^d + T \cdot \epsilon^* \\
&\leq O\left(T^{2/3} (\log T)^{1/3} |\mathcal{A}|^{1/3} |\mathcal{B}|^{1/3} \right).
\end{aligned}$$

where (B) follows from Lemma D.2. This proves the desired result.

□

F.2 Proof of Theorem 6.3

Theorem 6.3. *Suppose that $c \geq 1$ and $d \leq 1$, and let $\eta := 2/(2+d)$. Let $E = \Theta(|\mathcal{A}|^{-\eta} (|\mathcal{B}| \log T)^{1-\eta} (c \cdot T)^{\eta})$. Let $\texttt{ALG}_2$ be any algorithm with high-probability instantaneous regret $g(t, T, \mathcal{B}) = O\left((|\mathcal{A}| \cdot |\mathcal{B}| \cdot \log T)^{\eta/2} \cdot (c \cdot T)^{-\eta/2} \right)$ for $t > E$ and $g(t, T, \mathcal{B}) = 1$ for $t \leq E$, and let $\texttt{ALG}_1 = \texttt{ExploreThenUCB}(E)$. Then, then the leader and follower regret with respect to the generalized (c, d, γ)-tolerant benchmarks are both bounded as:*

$$\max(R_1(T), R_2(T)) = O((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta}).$$

The proof follows a similar argument to the proof of Theorem 4.4 and borrows some lemmas from Appendix D.3

Proof of Theorem 6.3. Assume that the clean event G holds. This occurs with probability at least $1 - (1 + |\mathcal{A}|)T^{-3}$ (Lemma D.3), so the clean event not occurring counts negligibly towards regret.

The regret in the explore phase is bounded by $O(1)$ in each round, the total regret from that phase is $E \cdot |\mathcal{A}| = O((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta})$ for either player.

The remainder of the analysis boils down to bounding the regret in the UCB phase. We separately analyze the regret of the leader and the follower. Observe that $\epsilon^* = \max_{t > E} g(t, T, \mathcal{B}) = O\left((|\mathcal{A}| \cdot |\mathcal{B}| \log T)^{\eta/2} \cdot (c \cdot T)^{-\eta/2} \right)$ based on the assumption on the follower's algorithm.

Regret for the leader. We bound the regret as:

$$
\begin{aligned}
&\beta_1^{\mathrm{tol}} \cdot (T - E \cdot |\mathcal{A}|) - \sum_{t=E\cdot|\mathcal{A}|}^{T} v^1_{a_t,b_t} \\
&\leq \sum_{t=E\cdot|\mathcal{A}|+1}^{T} \left(c \cdot (\epsilon^*)^d + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&= \sum_{a\in\mathcal{A}} \sum_{t\in T_a} \left(c \cdot (\epsilon^*)^d + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&\leq |\mathcal{A}| + \sum_{a\in\mathcal{A}} \sum_{t\in n_{E\cdot|\mathcal{A}|,T}(a)\setminus\{\max(S(a))\}} \left(c \cdot (\epsilon^*)^d + \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - v_1(a_t,b_t) \right) \\
&\leq |\mathcal{A}| + \underbrace{c \cdot (\epsilon^*)^d \cdot T}_{(1)} \\
&+ \underbrace{\sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \left(\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - \frac{1}{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \sum_{t\in S(a)\setminus\{\max(S(a))\}} (v_1(a_t,b_t)) \right)}_{(2)}
\end{aligned}
$$

The term $|\mathcal{A}|$ computes negligibly and term (1) is equal to $O((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{\eta\cdot d/2} \cdot (c \cdot T)^{1-\eta\cdot d/2}) = O((|\mathcal{A}| \cdot |\mathcal{B}| \cdot (\log T))^{1-\eta} \cdot (c \cdot T)^{\eta})$. Term (2) can be bounded by by the same argument as Theorem 4.5, which we repeat for completeness:

$$
\begin{aligned}
&\sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \left(\max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a,b) - \frac{1}{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \sum_{t\in S(a)\setminus\{\max(S(a))\}} \left(v^1_{a_t,b_t}\right) \right) \\
&\leq \sum_{a\in\mathcal{A}} (n_{E\cdot|\mathcal{A}|,T}(a) - 1) \cdot \frac{20\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a)} - 1} \\
&\leq O\left(\sqrt{\log T} \cdot \sum_{a\in\mathcal{A}} \sqrt{n_{E\cdot|\mathcal{A}|,T}(a) - 1} \right) \\
&\leq O\left(\sqrt{|\mathcal{A}| T \log T} \right),
\end{aligned}
$$

where the first inequality uses Lemma D.4 and the last inequality uses Jensen's inequality.

Regret for the follower. Note that $\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)$ denotes the set of time steps where an action in $\mathcal{A}_{\epsilon^*}$ is chosen. We bound the regret as:

$$
\begin{aligned}
&\beta_2^{\mathrm{tol}} \cdot (T - E \cdot |\mathcal{A}|) - \sum_{t=E\cdot|\mathcal{A}|}^{T} v_2(a_t,b_t) \\
&\leq \underbrace{\left(\sum_{t=E\cdot|\mathcal{A}|}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] \right)}_{(1)} + \underbrace{\sum_{t\in\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)} \left(\min_{a\in\mathcal{A}_{\epsilon^*}} \max_{b\in\mathcal{B}} v_2(a,b) - v_2(a_t,b_t) \right)}_{(2)} + \underbrace{c \cdot (\epsilon^*)^d \cdot |\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)|}_{(3)}
\end{aligned}
$$

Term (1) can be bounded by a similar argument to Theorem 4.5, which we repeat for completeness. This term can be rewritten as $\sum_{t=E\cdot|\mathcal{A}|}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] = \sum_{a\notin\mathcal{A}_{\epsilon^*}} n_{E\cdot|\mathcal{A}|,T}(a)$. This counts the number of times that arms outside of $\mathcal{A}_{\epsilon^*}$ are pulled during the UCB phase. The key intuition is when an arm $a_t \notin \mathcal{A}_{\epsilon^*}$, it holds that:

$$v_1(a_t, b_t) \leq \max_{b\in\mathcal{B}_{\epsilon^*}(a')} v_1(a_t, b) < \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \epsilon^*,$$

where the first inequality uses the fact that $b_t \in \mathcal{B}_{\epsilon^*}(a_t)$ (which follows from the clean event G_F) and the second inequality uses the fact that $a_t \notin \mathcal{A}_{\epsilon^*}$. This implies that for any $a' \notin \mathcal{A}_{\epsilon^*}$, the average reward across all time steps (except for the last time step) where a' is pulled satisfies:

$$\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_{t'}, b_{t'}) < \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \epsilon^*.$$

However, by Lemma D.4, we can also lower bound the average reward across all time steps (except for the last time step) where a' is pulled in terms of $n_{E\cdot|\mathcal{A}|,T}(a')$ as follows:

$$\frac{1}{n_{E\cdot|\mathcal{A}|,T}(a') - 1} \sum_{t\in S(a')\setminus\{\max(S(a'))\}} v_1(a_{t'}, b_{t'}) \geq \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}_{\epsilon^*}(a)} v_1(a, b) - \frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}}.$$

Putting these two inequalities together, we see that:

$$\frac{10\sqrt{\log T}}{\sqrt{n_{E\cdot|\mathcal{A}|,T}(a') - 1}} \geq \epsilon^*,$$

which bounds the number of times that a' is pulled during the UCB phase as follows:

$$n_{E\cdot|\mathcal{A}|,T}(a') \leq \Theta\left(\frac{\log T}{(\epsilon^*)^2}\right) = \Theta\left((|\mathcal{A}|\cdot|\mathcal{B}|)^{-\eta}\cdot(\log T)^{1-\eta}\cdot(c\cdot T)^{\eta}\right).$$

This means that:

$$\sum_{t=E\cdot|\mathcal{A}|}^{T} \mathbb{1}[t \notin \cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)] = \sum_{a\notin\mathcal{A}_{\epsilon}} n_{E\cdot|\mathcal{A}|,T}(a) \leq \Theta\left((|\mathcal{A}|\cdot\log T)^{1-\eta}\cdot(|\mathcal{B}|)^{-\eta}\cdot(c\cdot T)^{\eta}\right)$$

Next, we bound term (2):

$$\begin{aligned}
\min_{a\in\mathcal{A}_{\epsilon^*}} \max_{b\in\mathcal{B}} v_2(a, b) - \mathbb{E}[v_2(a_t, b_t)] &\leq \sum_{t\in\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)} \left(\max_{b\in\mathcal{B}} v_2(a_t, b) - \mathbb{E}[v_2(a_t, b_t)]\right) \leq |\cup_{a\in\mathcal{A}_{\epsilon^*}} S(a)|\cdot\epsilon^* \\
&\leq T\cdot\epsilon^* \\
&\leq T\cdot c\cdot(\epsilon^*)^d \\
&= O((|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T))^{\eta\cdot d/2}\cdot(c\cdot T)^{1-\eta\cdot d/2}) \\
&\leq O((|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T))^{1-\eta}\cdot(c\cdot T)^{\eta}).
\end{aligned}$$

Finally, we bound term (3) as

$$\epsilon^*\cdot|S| \leq T\cdot\epsilon^* \leq T\cdot c\cdot(\epsilon^*)^d \leq O((|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T))^{\eta\cdot d/2}\cdot(c\cdot T)^{1-\eta\cdot d/2}) = O((|\mathcal{A}|\cdot|\mathcal{B}|\cdot(\log T))^{1-\eta}\cdot(c\cdot T)^{\eta}).$$

□

G Proofs for Section 7

The follower algorithms $\texttt{ALG}_2$ that we analyze in this section run a separate instantation of a standard bandit algorithm for every $a \in \mathcal{A}$. We show that if $\texttt{ALG}$ satisfies a high-probability instantaneous (resp. anytime) regret bound, the same high-probability instantaneous (resp. anytime) regret bound is inherited for $\texttt{ALG}_2$ (recall that in Section 2.3 we defined high-probability instantaneous regret and high-probability anytime regret for both single-bandit learners which act in isolation and follower algorithms).

Lemma G.1. *Suppose that the follower algorithm $\texttt{ALG}_2$ runs a separate instantation, for every $a \in \mathcal{A}$, of an single-bandit learning algorithm $\texttt{ALG}$ operating on the arms $\mathcal{B}$. If $\texttt{ALG}$ satisfies high-probability instantaneous regret g, then $\texttt{ALG}_2$ satisfies high-probability instantaneous regret g. Similarly, if $\texttt{ALG}$ also satisfies high-probability anytime regret h, then $\texttt{ALG}_2$ also satisfies high-probability anytime regret h.*

Proof. We use the following notation in the proof. Let $n_t(a)$ be the number of times that arm a has been pulled up prior to time step t. Following Appendix B, the follower's history can be represented as:

$$H_{2,t} := \left\{(t', a_{t'}, b_{t'}, r_{2,t'}(a_{t'}, b_{t'})) \mid 1 \leq t' < t, a_{t'} = a\right\},$$

and the follower's history on the arm $a \in \mathcal{A}$ can be represented as:

$$H_{2,t,a} := \left\{(n_{t'+1}(a), b_{t'}, r_{2,t'}(a_{t'}, b_{t'})) \mid 1 \leq t' < t, a_{t'} = a\right\}.$$

Using this notation and by the definition of $\texttt{ALG}_2$, we see that $\texttt{ALG}_2(a_t, H_{2,t}) = \texttt{ALG}(H_{2,t,a_t})$. We use this relationship to analyze the high-probability instantaneous regret and high-probability anytime regret of $\texttt{ALG}_2$.

High-probability instantaneous regret. Let the time horizon be T, and suppose that $\texttt{ALG}$ satisfies high-probability instantaneous regret $g(t, T, \mathcal{B})$ for every $1 \leq t \leq T$. Using this combined with the fact that $\texttt{ALG}_2(a_t, H_{2,t}) = \texttt{ALG}(H_{2,t,a_t})$, we see that for each $a \in \mathcal{A}$:

$$\mathbb{P}\left[\forall t \in [T] \mid v_2(a_t, b_t) \geq \max_{b \in \mathcal{B}} v_2(a_t, b) - g(n_{t+1}(a), T)\right] \geq 1 - T^{-3}.$$

Taking a union bound over $a \in \mathcal{A}$ demonstrates that:

$$\mathbb{P}\left[\forall t \in [T], a \in \mathcal{A} \mid v_2(a_t, b_t) \geq \max_{b \in \mathcal{B}} v_2(a_t, b) - g(n_t(a) + 1, T)\right] \geq 1 - |\mathcal{A}| \cdot T^{-3},$$

so $\texttt{ALG}_2$ satisfies high-probability instantaneous regret g.

High-probability anytime regret. Let the time horizon be T, and suppose that $\texttt{ALG}$ satisfies high-probability anytime regret $h(t, T, \mathcal{B})$ for every $1 \leq t \leq T$. Using this combined with the fact that $\texttt{ALG}_2(a_t, H_{2,t}) = \texttt{ALG}(H_{2,t,a_t})$, we see that for each $a \in \mathcal{A}$:

$$\mathbb{P}\left[\forall t \in [T] \mid \sum_{t' \leq t \mid a_{t'} = a} \max_{b \in \mathcal{B}} v_2(a, b) - \sum_{t' \leq t \mid a_{t'} = a} v_2(a, b_{t'}) \leq h(n_{t+1}(a), T)\right] \geq 1 - T^{-3}.$$

Taking a union bound over $a \in \mathcal{A}$ demonstrates that:

$$\mathbb{P}\left[\forall t \in [T], a \in \mathcal{A} \mid \sum_{t' \le t | a_{t'} = a} \max_{b \in \mathcal{B}} v_2(a,b) - \sum_{t' \le t | a_{t'} = a} v_2(a, b_{t'}) \le h(n_{t+1}(a), T)\right] \ge 1 - |\mathcal{A}| \cdot T^{-3},$$

so $\texttt{ALG}_2$ satisfies high-probability anytime regret h. □

Using Lemma G.1, it suffices to analyze the high-probability instantaneous regret and high-probability anytime regret of the following standard bandit algorithms as single-bandit learners with arms $\mathcal{B}$, mean rewards $v_2(b)$, and stochastic rewards $r_{2,t}(b)$. In the proofs, we let $n_t(b)$ denote the number of times that arm b has been pulled prior to time step t.

Proposition 7.1. *Suppose that for every $a \in \mathcal{A}$, the follower runs a separate instantiation of* `ActiveArmElimination`$(M_1, \ldots, M_P)$ *(Algorithm 7) with $M_i = \Theta(\log T \cdot 2^{2i})$. Then the follower satisfies high-probability instantaneous regret $g(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot \log(T)/t}$, which implies $g(t, T, \mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}| \log T)^{1/3} T^{-1/3}\right)$ for $t \ge \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}| \log T)^{1/3} T^{2/3})$. Moreover, the follower satisfies high-probability anytime regret $h(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot \log(T) \cdot t})$.*

Proof of Proposition 7.1. We first show the high-probability instantaneous regret bound and then deduce the high-probability anytime regret bound.

High-probability instantaneous regret bound. By Lemma G.1, it suffices to show the bound for `ActiveArmElimination` (using phase lengths $M_i = \Theta(\log(T) \cdot 2^{2i})$) as a single-bandit learner with arms $\mathcal{B}$, mean rewards $v_2(b)$, and stochastic rewards $r_{2,t}(b)$. We let $n_t(b)$ denote the number of times that arm b has been pulled prior to time step t in the current phase. Let $\hat{v}_{2,t}(b)$ denote the empirical mean reward for arm b over the rewards observed prior to time step t in the *previous* (last completed) phase. Let $\mathcal{B}'_{t,\text{curr}}$ be the set of arms active in the *current* phase, and let $\mathcal{B}'_{t,\text{prev}}$ be the set of arms active in the *previous* (last completed) phase. For each time step t, let s'_t denote the index of the *previous* (last completed) phase at time step t.

Let the clean event G denote the event that at every time step t, it holds that:

$$\forall t \in [T], b \in \mathcal{B}'_{t,\text{prev}} : |v_2(b) - \hat{v}_{2,t}(b)| \le \frac{10\sqrt{\log T}}{\sqrt{M_{s'_t}}}.$$

Applying a Chernoff bound and a union bound, it holds that $P[G] \ge 1 - T^{-3}$.

We condition on the clean event G for the remainder of the analysis. Let $b^* = \text{argmax}_{b \in \mathcal{B}} v_2(b)$. Using the elimination rule, we can bound the suboptimality of each arm $b \in \mathcal{B}'_{t,\text{curr}}$:

$$\begin{aligned}
&|v_2(b^*) - v_2(b)| \\
&\le |\hat{v}_{2,t}(b^*) - v_2(b^*)| + |\hat{v}_{2,t}(b) - v_2(b)| + |\hat{v}_{2,t}(b^*) - \hat{v}_{2,t}(b)| \\
&\le 40\frac{\sqrt{\log(T)}}{\sqrt{M_{s'_t}}} \\
&\le \Theta(2^{-s'_t}).
\end{aligned}$$

It suffices to lower bound $2^{-2 \cdot s'_t}$. We observe that:

$$t \le |\mathcal{B}| \left(M_{s'_t+1} + \sum_{s=1}^{s'_t} M_s\right) \le \Theta(|\mathcal{B}| \cdot \log(T) \cdot 2^{2 \cdot s'_t}),$$

where the last expression uses the geometric rate of increase of $M_i = \Theta(\log(T) \cdot 2^{2i})$. This implies that

$$2^{-s'_t} = O(\sqrt{|\mathcal{B}| \cdot \log T/t}).$$

Altogether, this implies that:

$$v_2(b_t) \geq \max_{b \in \mathcal{B}} v_2(b) - O(\sqrt{|\mathcal{B}| \cdot \log T/t}),$$

as desired.

High-probability anytime regret bound. Using Observation 7.1, it holds that the high-probability anytime regret can be bounded as:

$$\sum_{t'=1}^{t} O\left(\sqrt{\frac{\log(T) \cdot |\mathcal{B}|}{t'}}\right) = \sqrt{\log(T) \cdot |\mathcal{B}|} \cdot O\left(\sum_{t'=1}^{t} \frac{1}{\sqrt{i}}\right) \leq_{(A)} \Theta(\sqrt{\log(T) \cdot t \cdot |\mathcal{B}|})$$

where (A) follows from an integral bound and Jensen's inequality. This proves the desired bound. □

Proposition 7.2. *Suppose that the follower runs a separate instantiation of* `ExploreThenCommit`$(E, \mathcal{B})$ *(Algorithm 1) for every $a \in \mathcal{A}$. Then, the follower satisfies high-probability instantaneous regret $g(t, T, \mathcal{B}) = \mathcal{O}(\sqrt{\log T/E})$ for all time steps $t \geq E \cdot |\mathcal{B}|$. If $E = \Theta((|\mathcal{A} \cdot |\mathcal{B}|)^{-2/3}(\log T)^{1/3}T^{2/3})$, then $g(t, T, \mathcal{B}) = O\left((|\mathcal{A}||\mathcal{B}| \log T)^{1/3}T^{-1/3}\right)$ for $t \geq \Theta(|\mathcal{A}|^{-2/3}(|\mathcal{B}| \log T)^{1/3}T^{2/3})$.*

Proof of Proposition 7.2. By Lemma G.1, it suffices to show the instantaneous regret bound for `ExploreThenCommit` as a single-bandit learner with arms $\mathcal{B}$, mean rewards $v_2(b)$, and stochastic rewards $r_{2,t}(b)$. We let $n_t(b)$ denote the number of times that arm b has been pulled prior to time step t. Let $\hat{v}_{2,t}(b)$ denote the empirical mean reward for arm b over the rewards observed prior to time step t.

Let the clean event G capture the event that the empirical mean of every arm is close to the true mean whenever $t > E \cdot |\mathcal{B}|$ time steps, that is:

$$\forall b \in \mathcal{B}, t > E \cdot |\mathcal{B}| : |\hat{v}_{2,t}(b) - v_2(b)| \leq 10 \cdot \frac{\sqrt{\log(T)}}{\sqrt{E}}$$

Applying a Chernoff bound (and using the analogue of one of the canonical bandit models in Lattimore and Szepesvári [2020]), it holds that $P[G] \geq 1 - T^{-3}$.

Now, conditioning on the clean event G, we see that after time step $t > E \cdot |\mathcal{B}|$, it holds that:

$$|\hat{v}_{2,t}(b) - v_2(b)| \leq 10\frac{\sqrt{\log(T)}}{\sqrt{E}}.$$

Since the algorithm chooses the arm with highest empirical mean from the first $E \cdot |\mathcal{B}|$ time steps is selected, this means that:

$$\max_{b \in \mathcal{B}} v_2(b) - v_2(b) \leq 20 \cdot \frac{\sqrt{\log(T)}}{\sqrt{E}}$$

for any $t > E \cdot |\mathcal{B}|$. □

Proposition 7.3. *Suppose that the follower runs a separate instantiation of UCB for every $a \in \mathcal{A}$. Then, the follower satisfies high-probability anytime regret bound $h(t, T, \mathcal{B}) = O(\sqrt{|\mathcal{B}| \cdot t \cdot \log(T)})$.*

Proof of Proposition 7.3. By Lemma G.1, it suffices to show the anytime regret bound for UCB as a single-bandit learner with arms $\mathcal{B}$, mean rewards $v_2(b)$, and stochastic rewards $r_{2,t}(b)$. We let $n_t(b)$ denote the number of times that arm b has been pulled prior to time step t. Let $\hat{v}_{2,t}(b)$ denote the empirical mean reward for arm b over the rewards observed prior to time step t.

We define the *clean event* G as the true mean being contained within the upper and lower confidence bounds for each arm a, that is:

$$\forall b \in \mathcal{B}, t \leq T : |\hat{v}_{2,t}(b) - v_2(b)| \leq 10 \cdot \sqrt{\frac{\log(T)}{n_t(b)}}.$$

By a Chernoff bound (and using the analogue of one of the canonical bandit models in Lattimore and Szepesvári [2020]) followed by a union bound, we have that $P[G] \geq 1 - T^{-3}$.

We condition on G for the remainder of the analysis. Since the arm with highest upper confidence bound is always chosen and since G holds, the selected arm b_t's true mean $v_2(b_t)$ falls within the $2 \cdot \sqrt{\frac{\log(T)}{n_t(b_t)}}$ bound. By Lemma B.1, this means that the regret at any time step t for any arm $a \in \mathcal{A}$ is upper bounded by:

$$10 \cdot \sum_{t'=1}^{t} \sqrt{\frac{\log(T)}{n_{t'}(b_{t'})}} \leq 10 \cdot \sqrt{\log(T) \cdot |\mathcal{B}| \cdot t}$$

as desired. □