A Unified Approach to Translate Classical Bandit Algorithms to the Structured Bandit Setting

Samarth Gupta, Shreyas Chaudhari, *Member, IEEE*, Subhojyoti Mukherjee, Gauri Joshi, and Osman Yağan, *Senior Member, IEEE*

***Abstract*—We consider a finite-armed structured bandit problem in which mean rewards of different arms are known functions of a common hidden parameter θ^*. Since we do not place any restrictions on these functions, the problem setting subsumes several previously studied frameworks that assume linear or invertible reward functions. We propose a novel approach to gradually estimate the hidden θ^* and use the estimate together with the mean reward functions to substantially reduce exploration of sub-optimal arms. This approach enables us to fundamentally generalize any classical bandit algorithm including UCB and Thompson Sampling to the structured bandit setting. We prove via regret analysis that our proposed UCB-C algorithm (structured bandit versions of UCB) pulls only a *subset* of the sub-optimal arms $O(\log T)$ times while the other sub-optimal arms (referred to as *non-competitive* arms) are pulled $O(1)$ times. As a result, in cases where all sub-optimal arms are non-competitive, which can happen in many practical scenarios, the proposed algorithm achieves bounded regret. We also conduct simulations on the MOVIELENS recommendations dataset to demonstrate the improvement of the proposed algorithms over existing structured bandit algorithms.**

***Index Terms*—Multi-armed bandits, sequential decision making, online learning, statistical learning, regret bounds.**

I. INTRODUCTION

A. Overview

THE MULTI-ARMED bandit problem [1] (MAB) falls under the umbrella of sequential decision-making problems. It has numerous applications such as clinical trials [2], system testing [3], scheduling in computing systems [4], and Web optimization [5], [6], to name a few. In the classical K-armed bandit formulation, a player is presented with K arms. At each time step $t = 1, 2, \ldots$, she decides to pull an arm $k \in \mathcal{K}$ and receives a random *reward* R_k with unknown mean μ_k. The goal of the player is to maximize their expected cumulative reward (or equivalently, minimize expected cumulative *regret*) over T time steps. In order to do so, the player must strike a balance between estimating the unknown rewards *accurately* by pulling all the arms (exploration) and always pulling the current best arm (exploitation). The seminal work of Lai and Robins (1985) proposed the Upper Confidence Bound (UCB) algorithm that balances the exploration-exploitation tradeoff in the MAB problem. Subsequently, several algorithms such as UCB1 [7], Thompson Sampling (TS) [8] and KL-UCB [9] were proposed and analyzed for the classical MAB setting.

In this article, we study a fundamental variant of classical multi-armed bandits called the *structured multi-armed bandit problem*, where mean rewards of the arms are functions of a *hidden* parameter θ. That is, the expected reward $\mathbb{E}[R_k|\theta] = \mu_k(\theta)$ of arm k is a *known* function of the parameter θ that lies in a (*known*) set Θ. However, the true value of θ, denoted as θ^*, is unknown. The dependence of mean rewards on the common parameter introduces a *structure* in the MAB problem. For example, rewards observed from an arm may provide partial information about the mean rewards of other arms, making it possible to significantly lower the resulting cumulative regret as compared to the classical MAB setting.

Structured bandit models arise in many applications and have been studied by several authors with motivating applications including dynamic pricing (described in [10]), cellular coverage optimization (by [11]), drug dosage optimization (discussed in [12]) and system diagnosis; see Section I-B for an illustrative application of the structured MAB framework. In this article, we consider a *general* version of the structured MAB framework that subsumes and generalizes several previously considered settings. More importantly, we propose a novel and unified approach that would allow extending any current or future MAB algorithm (UCB, TS, KL-UCB, etc.) to the structured setting; see Sections I-C and III for our main contributions and a comparison of our work with related literature.

B. An Illustrative Example

For illustration purposes, consider the example of movie recommendation, where a company would like to decide which movie(s) to recommend to each user with the goal

Manuscript received May 14, 2020; revised October 24, 2020; accepted November 17, 2020. Date of publication December 2, 2020; date of current version January 7, 2021. This work was supported in part by the NSF through under Grant CCF-1840860 and Grant CCF-2007834; in part by the Siebel Energy Institute; in part by the Carnegie Bosch Institute; in part by the Manufacturing Futures Initiative; and in part by the CyLab IoT Initiative. The work of Samarth Gupta was supported in part by the CyLab Presidential Fellowship and in part by the David H. Barakat and LaVerne Owen-Barakat CIT Dean's Fellowship. A short 4-page version of this paper is under review at IEEE ICASSP 2021. *(Corresponding author: Samarth Gupta.)*

Samarth Gupta, Shreyas Chaudhari, Gauri Joshi, and Osman Yağan are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: samarthg@andrew.cmu.edu; schaudh2@andrew.cmu.edu; gaurij@andrew.cmu.edu; oyagan@andrew.cmu.edu).

Subhojyoti Mukherjee is with the Department of Electrical and Computer Engineering, University of Wisconsin–Madison, Madison, WI 53706 USA (e-mail: smukherjee27@wisc.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/JSAIT.2020.3041246, provided by the authors.

Digital Object Identifier 10.1109/JSAIT.2020.3041246

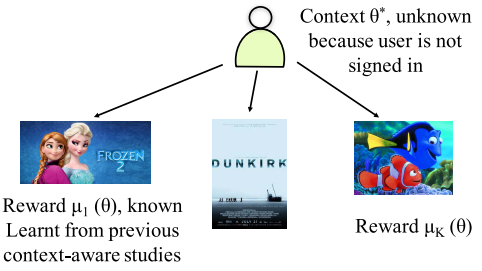

Fig. 1. Movie recommendation application of the structured bandit framework studied in this article. The context θ (for example, the age of the user) is unknown because the user is not signed in. But if a user gives a high rating the first movie (Frozen) one could infer that the age θ is small, which in turn implies that the user will give a high rating to the third movie (Finding Nemo).

of maximizing user engagement (e.g., in terms of click probability and time spent watching, etc.). In order to achieve this, the company needs to identify the most appealing movie for the user in an online manner and this is where multi-armed bandit algorithms can be helpful. However, classical MAB algorithms are typically based on the (implicit) assumption that rewards from different arms (i.e., different movies in this context) are independent of each other. This assumption is unlikely to hold in reality since the user choices corresponding to different movies are likely to be related to each other; e.g., the engagement corresponding to different movies may depend on the age/occupation/income/taste of the user.

To address this, *contextual* bandits [13], [14] have been proposed and studied widely for personalized recommendations. There, it is assumed that before making a choice (of which movie to recommend), a *context* feature of the user is observed; the context can include personal information of the user including age, occupation, income, or previous browsing information. Contextual bandit algorithms aim to learn the mapping from context information to the most appealing arm, and can prove useful in applications involving personalized recommendations (or, advertising). However in several use cases, observing contextual features leads to privacy concerns. In addition, the contextual features may not be visible for *new* users or users who are signed in anonymously to protect their identity.

The structured bandit setting considered in this article (and by many others [10], [11], [12]) can be viewed as the same problem setting with contextual bandits with the following difference. Unlike contextual bandits, the context of the users are *hidden* in the structured setting, but in return it is assumed that the mappings from the contexts to *mean* rewards of arms are known a priori. It is anticipated that the mean reward mappings can be learned from paid surveys in which users participate with their consent. The proposed structured bandit framework's goal is to use this information to provide the best recommendation to an anonymous user whose context vector θ is unknown; e.g., see Figure 1. Thus, our problem formulation is complementary to contextual bandits; in contextual bandits the context θ is known while the reward mappings $\mu_k(\theta)$ are unknown, whereas in our setting θ is unknown and the mean rewards $\mu_k(\theta)$ are known. A detailed problem formulation discussing the assumptions and extensions of this set-up is given in Section II.

C. Main Contributions and Organization

We summarize the key contributions of the paper below. The upcoming sections will develop each of these in detail.

1) *General Setting Subsuming Previous Works:* Structured bandits have been studied in the past [10], [12], [15], [16], [17], [18] but with certain restrictions (e.g., being linear, invertible, etc.) on the mean reward mappings $\mu_k(\theta)$. We consider a general setting that puts no restrictions on the mean reward mappings. In fact, our setting subsumes recently studied models such as Global Bandits [10], Regional Bandits [12] and structured bandits with linear rewards [15]; see Section III for a detailed comparison with previous works. There are a couple of recent works [19], [20] that do consider a general structured bandit setting similar to our work—see Section III for details. Our approach differs from these in its flexibility to extend any classical bandit algorithm (UCB, Thompson sampling, etc.) to the structured bandit setting. In particular, using Thompson sampling [21], [22] as the underlying bandit algorithm yields a robust and empirically superior way (see Section V-D) to minimize superfluous exploration. The UCB-S algorithm proposed in [19] extends the UCB algorithm to structured setting. However, the approach presented in [19] can not be extended to Thompson sampling or other classical bandit algorithms; in fact, this point was highlighted in [19] as an open question for future work. In [20], there are several assumptions in the model that are not imposed here, including the assumption that the conditional reward distributions are known and reward mappings are continuous. In addition, the main focus of [20] is the parameter regime where regret scales logarithmically with time T, while our approach demonstrates the possibility of achieving *bounded* regret.
2) *Extending any Classical Bandit Algorithm to the Structured Bandit Setting:* We propose a novel and unified approach that would allow extending any classical or future MAB algorithm (that is developed for the non-structured setting) to the structured bandit framework given in Figure 2. Put differently, we propose a *class* of structured bandit algorithms referred to as ALGORITHM-C, where "ALGORITHM" can be any classical bandit algorithm including UCB, TS, KL-UCB, etc. A detailed description of the resulting algorithms, e.g., UCB-C, TS-C, etc., are given in Section IV with their steps illustrated in Figure 3.
3) *Unified Regret Analysis:* A key benefit of our algorithms is that they pull a subset of the arms (referred to as the *non-competitive* arms) only O(1) times. Intuitively, an arm is non-competitive if it can be identified as sub-optimal with high probability using only the samples from the optimal arm k^*. This is in contrast to classical MAB algorithms where all sub-optimal arms are

pulled $\mathrm{O}(\log T)$ times, where T is the total number of rounds. This is shown by analyzing the expected regret $\mathbb{E}\big[Reg(T)\big]$, which is the difference between the expected cumulative reward obtained by using the proposed algorithm and the expected cumulative reward of a genie policy that always pulls the optimal arm k^*. In particular, we provide rigorous regret analysis for UCB-C as summarized in the theorem below, and describe how our proof technique can be extended to other classical MAB algorithms.

Theorem 1 (Expected Regret Scaling): The expected regret of the UCB-C algorithm has the following scaling with respect to the number of rounds T:

$$\mathbb{E}\big[Reg(T)\big] \leq \big(C\big(\theta^*\big) - 1\big) \cdot \mathrm{O}(\log T) + \big(K - C\big(\theta^*\big)\big)\mathrm{O}(1) \tag{1}$$

where $C(\theta^*)$ is the number of competitive arms (including the optimal arm k^*) and θ^* is the true value of the hidden parameter. The remaining $K - C(\theta^*)$ arms are called non-competitive. An arm is said to be non-competitive if there exists an $\epsilon > 0$ such that $\mu_k(\theta) < \mu_{k^*}(\theta)$ for all $\theta \in \Theta^{*(\epsilon)}$, where $\Theta^{*(\epsilon)} = \{\theta \in \Theta : |\mu_{k^*}(\theta) - \mu_{k^*}(\theta^*)| \leq \epsilon\}$ (more details in Section V). The exact regret upper bound with all the constants follows from Theorem 2 and Theorem 3 in Section V. Recall that for the standard MAB setting [1], the regret upper bound is $(K-1)\mathrm{O}(\log T)$, where K is the total number of arms. Theorem 1 reveals that with our algorithms only $C(\theta^*) - 1$ out of the $K-1$ sub-optimal arms are pulled $\mathrm{O}(\log T)$ times. The other arms are pulled only $\mathrm{O}(1)$ times.

4) *Reduction in the Effective Number of Arms:* For any given set of reward functions $\mu_k(\theta)$, the number $C(\theta^*)$ of competitive arms depends on the *unknown* parameter θ^*; see Figure 4 in Section V for an illustration of this fact. We show that $C(\theta^*)$ can be much smaller than K in many practical cases. This is because, the reward functions (particularly that corresponding to the *optimal* arm) can provide enough information about the hidden θ^*, which in turn can help infer the sub-optimality of several other arms. More specifically, this happens when the reward functions are not flat around θ^*, that is, the pre-image set of $\{\theta \in \Theta{:}\mu_k(\theta) = \mu_k(\theta^*)\}$ is small. In the special case where the optimal arm k^* is *invertible* or has a unique maximum at $\mu_{k^*}(\theta^*)$, $C(\theta^*) = 1$ and our algorithms can achieve $\mathrm{O}(1)$ regret.
5) *Evaluation on Real-World Datasets:* In Section V, we present extensive simulations comparing the regret of the proposed algorithm with previous methods such as GLM-UCB [16] and UCB-S [19]. We also present simulation results for the case where the hidden parameter θ is a *vector*. In Section VII, we perform experiments on the MOVIELENS dataset to demonstrate the applicability of the UCB-C and TS-C algorithms. Our experimental results show that both UCB-C and TS-C lead to significant improvement over the performance of existing bandit strategies. In particular, TS-C is shown to consistently outperform all other algorithms across a wide range of settings.

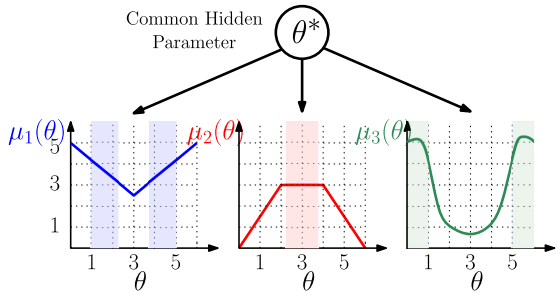

Fig. 2. Structured bandit setup: mean rewards of different arms share a common hidden parameter. This example illustrates a 3-armed bandit problem with shaded regions indicating the values of θ for which the particular arm is optimal.

II. Problem Formulation

Consider a multi-armed bandit setting with the set of arms $\mathcal{K} = \{1, 2, \ldots, K\}$. At each round t, the player pulls arm $k_t \in \mathcal{K}$ and observes a reward R_{k_t}. The reward R_{k_t} is a random variable with mean $\mu_{k_t}(\theta) = \mathbb{E}\big[R_{k_t}|\theta, k_t\big]$, where θ is a *fixed, but unknown parameter* which lies in a known set Θ; see Figure 2.

We denote the (unknown) true value of θ by θ^*. There are no restrictions on the set Θ. Although we focus on scalar θ in this article for brevity, the proposed algorithms and regret analysis can be generalized to the case where we have a hidden parameter *vector* $\vec{\theta} = [\theta_1, \theta_2, \ldots \theta_m]$. In Section V, we present simulation results for the case of a parameter vector θ. The mean reward functions $\mu_k(\theta) = \mathbb{E}[R_k|\theta]$ for $k \in \mathcal{K}$ can be arbitrary functions of θ with no linearity or continuity constraints imposed. While $\mu_k(\theta)$ are known to the player, the conditional distribution of rewards, i.e., $p(R_k|\theta)$ is not known.

We assume that the rewards R_k are sub-Gaussian with variance proxy σ^2, i.e., $\mathbb{E}\big[\exp(s(R_k - \mathbb{E}[R_k]))\big] \leq \exp(\frac{\sigma^2 s^2}{2})$ $\forall s \in \mathbb{R}$, and σ is known to the player. Both assumptions are common in the multi-armed bandit literature [10], [19], [23], [24]. In particular, the sub-Gaussianity of rewards enables us to apply Hoeffding's inequality, which is essential for the analysis of regret (defined below).

The objective of the player is to select arm k_t in round t so as to maximize her cumulative reward $\sum_{t=1}^{T} R_{k_t}$ after T rounds. If the player had known the hidden θ^*, then she would always pull arm $k^* = \arg\max_{k \in \mathcal{K}} \mu_k(\theta^*)$ that yields the highest mean reward at $\theta = \theta^*$. We refer to k^* as the *optimal* arm. Maximizing the cumulative reward is equivalent to minimizing the *cumulative regret*, which is defined as

$$Reg(T) \triangleq \sum_{t=1}^{T} \big(\mu_{k^*}\big(\theta^*\big) - \mu_{k_t}\big(\theta^*\big)\big) = \sum_{k \neq k^*} n_k(T)\Delta_k,$$

where $n_k(T)$ is the number of times arm k is pulled in T slots and $\Delta_k \triangleq \mu_{k^*}(\theta^*) - \mu_k(\theta^*)$ is the *sub-optimality gap* of arm k. Minimizing the cumulative regret is in turn equivalent to

minimizing $n_k(T)$, the number of times each sub-optimal arm $k \neq k^*$ is pulled.

Remark 1 (Connection to Classical Multi-Armed Bandits): The classical multi-armed bandit setting, which does not explicitly consider a *structure* among the mean rewards of different arms, is a special case of the proposed structured bandit framework. It corresponds to having a hidden parameter vector $\vec{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$ and the mean reward of each arm being $\mu_k = \theta_k$. In fact, our proposed algorithm described in Section IV reduces to standard UCB or Thompson sampling [1], [7] in this special case.

The proposed structured bandit subsumes several previously considered models where the rewards are assumed to be linear [15], [17], invertible and Hölder continuous [10], [12], etc. See Section III for a detailed comparison with these works.

III. Related Work

Since we do not make any assumptions on the mean reward functions $\mu_1(\theta), \mu_2(\theta), \ldots, \mu_K(\theta)$, our model subsumes several previously studied frameworks [10], [12], [15]. The similarities and differences between our model and existing works are discussed below.

Structured Bandits With Linear Rewards [15]: In [15], the authors consider a similar model with a common hidden parameter $\theta \in \mathbb{R}$, but the mean reward functions, $\mu_k(\theta)$ are linear in θ. Under this assumption, they design a greedy policy that achieves bounded regret. Our formulation does not make linearity assumptions on the reward functions. In the special case when $\mu_k(\theta)$ are linear, our proposed algorithm also achieves bounded regret.

Global and Regional Bandits: The papers [10], [12] generalize this to invertible and Hölder-continuous reward functions. Instead of scalar θ, [12] considers M common unknown parameters, that is, $\theta = (\theta_1, \theta_2, \ldots, \theta_M)$. Under these assumptions, [10], [12] demonstrate that it is possible to achieve bounded regret through a greedy policy. In contrast, our work makes no invertibility or continuity assumptions on the reward functions $\mu_k(\theta)$. In the special case when $\mu_k(\theta)$ are invertible, our proposed algorithm also achieves bounded regret.

Finite-Armed Generalized Linear Bandits: In the finite-armed linear bandit setting [17], the reward function of arm x_k is $\vec{\theta}^{\mathsf{T}} x_k$, which is subsumed by our formulation. For the case when $\mu_k(\theta) = g(\vec{\theta}^{\mathsf{T}} x_k)$, our setting becomes the generalized linear bandit setting [16], for some known function g. Here, θ is the shared unknown parameter. Due to the particular form of the mean reward functions, linear bandit algorithms construct confidence ellipsoid for θ^* to make decisions. This approach cannot be easily extended to non-linear settings. Although designed for the more general non-linear setting, our algorithms demonstrate comparable regret to the GLM-UCB [16], which is designed for linear bandits.

Minimal Exploration in Structured Bandits [20]: The problem formulation in [20] is very similar to this article. However, [20] assumes knowledge of the conditional reward distribution $p(R_k|\theta)$ in addition to knowing the mean reward functions $\mu_k(\theta)$. It also assumes that the mappings $\theta \to \mu_k(\theta)$ are continuous. As noted before, none of these assumptions are imposed in this article. Another major difference of [20] with our work is that they focus on obtaining asymptotically optimal results for the regimes where regret scales as $\log(T)$. When all arms are *non-competitive* (the case where our algorithms lead to O(1) regret), the solution to the optimization problem described in [20, Th. 1] becomes 0, causing the algorithm to get stuck in the exploitation phase. Put differently, the algorithm proposed in [20] is not applicable to cases where $C(\theta^*) = 1$. An important contribution of [20] is that it provides a lower bound on the regret of structured bandit algorithms. In fact, the lower bound presented in this article is directly based on the lower bound in [20].

Finite-Armed Structured Bandits [19]: The work closest to ours is [19]. They consider the same model that we consider and propose the UCB-S algorithm, which is a UCB-style algorithm for this setting. Our approach allows us to extend our UCB-style algorithm to other classical bandit algorithms such as Thompson sampling. In Section V and Section VII, we extensively compare our proposed algorithms (both qualitatively and empirically) with the UCB-S algorithm proposed in [19]. As observed in the simulations, UCB-S is susceptible to small changes in the mean reward functions and θ^*, whereas the UCB-C algorithm that we propose here is seen to be much more robust to such variations.

Connection to Information-Directed Sampling: Works such as [25], [26] consider a similar structured setting but assume that the conditional reward distributions $p(R_k|\theta)$ and the prior $p(\theta)$ are known, whereas we only consider that the *mean* reward functions $\mu_k(\theta) = \mathbb{E}[R_k|\theta]$ are known. The proposed algorithms are based on Thompson sampling from the posterior distribution of θ. Firstly, this approach will require a good prior over θ, and secondly, updating the posterior can be computationally expensive since it requires computing integrals over possibly high-dimensional spaces. The focus of [25] is on *worst-case* regret bounds (which are typically $\mathrm{O}(\sqrt{T})$), where the minimum gap between two arms can scale as $O(\log T/T)$, while [26] gives gap-dependent regret bounds in regimes where the regret scales as $\mathrm{O}(\log T)$. In addition to providing gap-dependent regret bounds, we also identify regimes where it is possible to achieve O(1) regret.

Best-Arm Identification: In several applications such as as hyper-parameter optimization [27] and crowd-sourced ranking [28], [29], [30], the objective is to maximize the probability of identifying the arm with the highest expected reward within a given time budget of T slots instead of maximizing the cumulative reward; that is, the focus is on exploration rather than exploitation. Best-arm identification started to be studied fairly recently [31], [32], [33]. A variant of the fixed-time budget setting is the fixed-confidence setting [24], [34], [35], [36], [37], where the aim is to minimize the number of slots required to reach a δ-error in identifying the best arm. Very few best-arm identification works consider structured rewards [6], [38], [39], [40], and they mostly assume *linear* rewards. The algorithm design and analysis tools are quite different in the best-armed bandit identification problem as compared to regret minimization. Thus, extending our approach to best-arm identification would be a non-trivial future research direction.

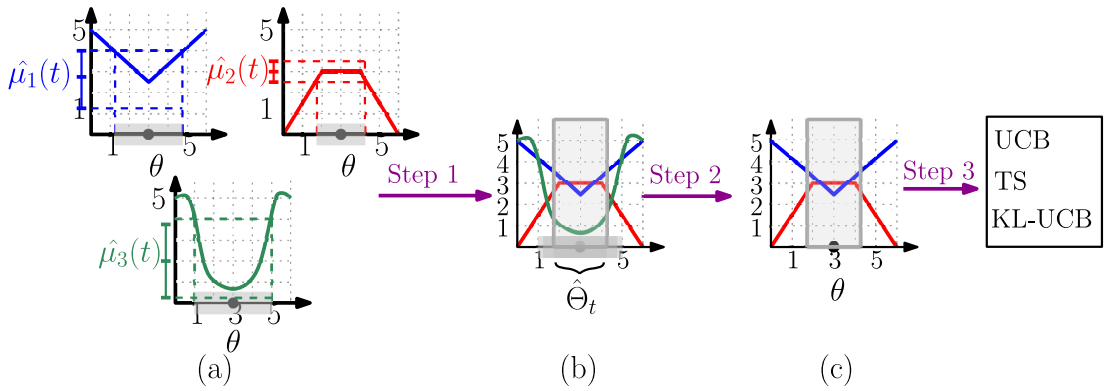

Fig. 3. Illustration of the steps of the proposed algorithm. In step 1, for each arm k we find the set of θ such that $|\mu_k(\theta) - \hat{\mu}_k(t)| < \sqrt{2\alpha\sigma^2 \log t / n_k(t)}$ (shaded in gray in part (a)). The intersection of these sets gives the confidence set $\hat{\Theta}_t$ shown in part (b). Next, we observe that the mean reward $\mu_3(\theta)$ of Arm 3 (shown in green) cannot be optimal if the unknown parameter θ^* lies in set $\hat{\Theta}_t$. Thus, it is declared as $\hat{\Theta}_t$-non-competitive and not considered in Step 3. In step 3, we pull one of the $\hat{\Theta}_t$-competitive arms (shown in (c)) using a classical bandit algorithm such as UCB, Thompson Sampling, KL-UCB, etc.

IV. Proposed Algorithm: Algorithm-C

For the problem formulation described in Section II, we propose the following three-step algorithm called Algorithm-Competitive, or, in short, Algorithm-C. Figure 3 illustrates the algorithm steps for the mean reward functions shown in Figure 2. Step 3 can employ any classical multi-armed bandit algorithm such as UCB or Thompson Sampling (TS), which we denote by Algorithm. Thus, we give a unified approach to translate any classical bandit algorithm to the structured bandit setting. The formal description of Algorithm-C with UCB and TS as final steps is given in Algorithm 1 and Algorithm 2, respectively.

At each round $t+1$, the algorithm performs the following steps.

Step 1 (Constructing a Confidence Set, $\hat{\Theta}_t$): From the samples observed till round t, we define the confidence set as follows:

$$\hat{\Theta}_t = \left\{ \theta : \forall k \in \mathcal{K}, \quad \left|\mu_k(\theta) - \hat{\mu}_k(t)\right| < \sqrt{\frac{2\alpha\sigma^2 \log t}{n_k(t)}} \right\}.$$

Here, $\hat{\mu}_k(t)$ is the empirical mean of rewards obtained from the $n_k(t)$ pulls of arm k. For each arm k, we construct a confidence set of θ such that the true mean $\mu_k(\theta)$ is within an interval of size $\sqrt{\frac{2\alpha\sigma^2 \log t}{n_k(t)}}$ from $\hat{\mu}_k(t)$. This is illustrated by the error bars along the y-axis in Figure 3(a), with the corresponding confidence sets shown in grey for each arm. Taking the intersection of these K confidence sets gives us $\hat{\Theta}_t$, wherein θ lies with high probability, as shown in Figure 3(b).

Step 2 (Finding the Set $\mathcal{C}_t$ of $\hat{\Theta}_t$-Competitive Arms): We let $\mathcal{C}_t$ denote the set of $\hat{\Theta}_t$-Competitive arms at round t, defined as follows.

Definition 1 ($\hat{\Theta}_t$-Competitive Arm): An arm k is said to be $\hat{\Theta}_t$-Competitive if its mean reward is the highest among all arms for some $\theta \in \hat{\Theta}_t$; i.e., $\exists \theta \in \hat{\Theta}_t$ such that $\mu_k(\theta) = \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$.

Definition 2 ($\hat{\Theta}_t$-Non-competitive Arm): An arm k is said to be $\hat{\Theta}_t$-Non-competitive if it is *not* $\hat{\Theta}_t$-Competitive; i.e., if $\mu_k(\theta) < \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$ for all $\theta \in \hat{\Theta}_t$.

If an arm is $\hat{\Theta}_t$-Non-competitive, then it cannot be optimal if the true parameter lies inside the confidence set $\hat{\Theta}_t$. These $\hat{\Theta}_t$-Non-competitive arms are not considered in Step 3 of the algorithm for round $t+1$. However, these arms can be $\hat{\Theta}_t$-Competitive in subsequent rounds; see also Remark 2. For example, in Figure 3(b), the mean reward of Arm 3 (shown in green) is strictly lower than the two other arms for all $\theta \in \hat{\Theta}_t$. Hence, this arm is declared as $\hat{\Theta}_t$-Non-competitive and only Arms 1 and 2 are included in the competitive set $\mathcal{C}_t$. In the rare case when $\hat{\Theta}_t$ is empty, we set $\mathcal{C}_t = \{1, \ldots, K\}$ and go directly to step 3 below.

Step 3 (Pull an Arm From the set $\mathcal{C}_t$ Using a Classical Bandit Algorithm): At round $t+1$, we choose one of the $\hat{\Theta}_t$-Competitive arms using any classical bandit Algorithm (for, e.g., UCB, Thompson sampling, KL-UCB, or any algorithm to be developed for the classical bandit framework which does not explicitly model a structure connecting the rewards of different arms). Formal descriptions for UCB-C and TS-C, i.e., the structured bandit versions on UCB [1], [7] and Thompson Sampling [21] algorithms, are presented in Algorithm 1 and Algorithm 2, respectively. The ability to employ any bandit algorithm in its last step is an important advantage of our algorithm. In particular, Thompson sampling has attracted a lot of attention [8], [21], [41], [42] due to its superior empirical performance. Extending it to the structured bandit setting results in significant regret improvement over previously proposed structured bandit algorithms.

Remark 2 (Connection to Successive Elimination Algorithms for Best-Arm Identification): Note that the empirically competitive set is updated at every round t. Thus, an arm that is empirically non-competitive at some round τ can be empirically competitive in subsequent rounds. Hence, the proposed algorithm is different from successive elimination methods used for best-arm identification [24], [31], [32], [33]. Unlike successive elimination methods, the proposed algorithm does not permanently eliminate empirically non-competitive arms but allows them to become competitive again in subsequent rounds.

Remark 3 (Comparison With UCB-S Proposed in [19]): The paper [19] proposes an algorithm called UCB-S for the

Algorithm 1 UCB-Competitive (UCB-C)

1: **Input:** Reward Functions $\{\mu_1, \mu_2 \dots \mu_K\}$
2: **Initialize:** $n_k = 0$ for all $k \in \{1, 2, \dots K\}$
3: **for** each round $t+1$ **do**
4: **Confidence set construction:**

$$\hat{\Theta}_t = \left\{\theta : \forall k \in \mathcal{K}, \mid \mu_k(\theta) - \hat{\mu}_k(t) \mid < \sqrt{\frac{2\alpha\sigma^2 \log t}{n_k(t)}}\right\}.$$

If $\hat{\Theta}_t$ is an empty set, then define $\mathcal{C}_t = \{1, \dots, K\}$ and go to step 6
5: **Define competitive set $\mathcal{C}_t$:**

$$\mathcal{C}_t = \left\{k : \mu_k(\theta) = \max_{\ell \in \mathcal{K}} \mu_\ell(\theta) \quad \text{for some } \theta \in \hat{\Theta}_t\right\}.$$

6: **UCB among competitive arms**

$$k_{t+1} = \arg\max_{k \in \mathcal{C}_t} \left(\hat{\mu}_k(t) + \sqrt{\frac{2\alpha\sigma^2 \log t}{n_k(t)}}\right).$$

7: Update empirical mean $\hat{\mu}_k(t+1)$ and $n_k(t+1)$ for arm k_{t+1}.
8: **end for**

Algorithm 2 Competitive Thompson Samp. (TS-C)

1: Steps 1 to 5 as in Algorithm 1
2: **Apply Thompson sampling on $\mathcal{C}_t$:**
for $k \in \mathcal{C}_t$ **do**
Sample $S_{k,t} \sim \mathcal{N}\left(\hat{\mu}_k(t), \frac{\beta\sigma^2}{n_k(t)}\right)$.
end for
$k_{t+1} = \arg\max_{k \in \mathcal{C}_t} S_{k,t}$
3: Update empirical mean, $\hat{\mu}_k$ and n_k for arm k_{t+1}.

same structured bandit framework considered in this work. UCB-S constructs the confidence set $\hat{\Theta}_t$ in the same way as Step 1 described above. It then pulls the arm $k = \arg\max_{k \in \mathcal{K}} \sup_{\theta \in \hat{\Theta}_t} \mu_k(\theta)$. Taking the supremum of $\mu_k(\theta)$ over θ makes UCB-S sensitive to small changes in $\mu_k(\theta)$ and to the confidence set $\hat{\Theta}_t$. Our approach of identifying competitive arms is more robust, as observed in Section V and Section VII. Moreover, the flexibility of using Thompson Sampling in Step 3 results in a significant reduction in regret over UCB-S. As noted in [19], the approach used to design UCB-S cannot be directly generalized to Thompson Sampling and other bandit algorithms.

Remark 4 (Computational Complexity of ALGORITHM-*C):* The computational complexity of ALGORITHM-C depends on the construction of $\hat{\Theta}_t$ and identifying $\hat{\Theta}_t$-competitive arms. The algorithm is easy to implement in cases where the set Θ is *small* or in situations where the pre-image of mean reward functions $\mu_k(\theta)$ can be easily computed. For our simulations and experiments, we discretize the set Θ wherever Θ is uncountable.

V. REGRET ANALYSIS AND INSIGHTS

In this section, we evaluate the performance of the UCB-C algorithm through a finite-time analysis of the expected cumulative regret defined as

$$\mathbb{E}\big[Reg(T)\big] = \sum_{k=1}^{K} \mathbb{E}[n_k(T)]\Delta_k, \tag{2}$$

where $\Delta_k = \mu_{k^*}(\theta^*) - \mu_k(\theta^*)$ and $n_k(T)$ is the number of times arm k is pulled in a total of T time steps. To analyze the expected regret, we need to determine $\mathbb{E}[n_k(T)]$ for each sub-optimal arm $k \neq k^*$. We derive $\mathbb{E}[n_k(T)]$ separately for competitive and non-competitive arms. Our proof presents a novel technique to show that each non-competitive arm is pulled only O(1) times; i.e., our algorithms stop pulling non-competitive arms after some finite time. To establish the fact that competitive arms are pulled O($\log T$) times each, we prove that the proposed algorithm effectively reduces a K-armed bandit problem to a $C(\theta^*)$-armed bandit problem, allowing us to extend the regret analysis of the underlying classical multi-armed bandit algorithm (UCB, Thompson Sampling, etc.).

A. Competitive and Non-Competitive Arms

In Section IV, we defined the notion of competitiveness of arms with respect to the confidence set $\hat{\Theta}_t$ at a fixed round t. For our regret analysis, we need asymptotic notions of competitiveness of arms, which are given below.

Definition 3 (Non-Competitive and Competitive Arms): For any $\epsilon > 0$, let

$$\Theta^{*(\epsilon)} = \big\{\theta : \big|\mu_{k^*}\big(\theta^*\big) - \mu_{k^*}(\theta)\big| < \epsilon\big\}.$$

An arm k is said to be non-competitive if there exists an $\epsilon > 0$ such that k is not the optimal arm for any $\theta \in \Theta^{*(\epsilon)}$; i.e., if $\mu_k(\theta) < \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$ for all $\theta \in \Theta^{*(\epsilon)}$. Otherwise, the arm is said to be competitive; i.e., if for all $\epsilon > 0$, $\exists \theta \in \Theta^{*(\epsilon)}$ such that $\mu_k(\theta) = \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$. The number of competitive arms is denoted by $C(\theta^*)$.

Since the optimal arm k^* is competitive by definition, we have

$$1 \leq C\big(\theta^*\big) \leq K.$$

We can think of $\Theta^{*(\epsilon)}$ as a confidence set for θ obtained from the samples of the best arm k^*. To intuitively understand the meaning of non-competitiveness, recall that the observed rewards $\hat{\mu}_k(t)$ from the arms help infer that θ^* lies in the confidence set $\hat{\Theta}_t$ with high probability. The observed reward $\hat{\mu}_{k^*}(t)$ of arm k^* will dominate the construction of the confidence set $\hat{\Theta}_t$ because a good multi-armed bandit strategy pulls the optimal arm O(t) times, while other arms are pulled at most O($\log t$) times. Thus, for any $\epsilon > 0$, we expect the confidence set $\hat{\Theta}_t$ to converge to $\Theta^{*(\epsilon)}$ as the number $n_{k^*}(t)$ of pulls for the optimal arm gets larger. As a result, if a sub-optimal arm k is non-competitive as per the definition above, i.e., $\mu_k(\theta) < \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$ for all $\theta \in \Theta^{*(\epsilon)}$, then the proposed algorithm will identify k as $\hat{\Theta}_t$-non-competitive (and thus not pull it) with increasing probability at every round t. In fact, our regret analysis shows that the likelihood of a non-competitive arm being pulled at time t decays as $t^{-1-\gamma}$ for some $\gamma > 0$, leading to such arms being pulled only finitely many times.

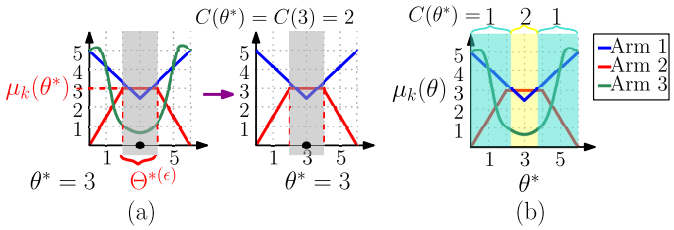

Fig. 4. (a) Illustration of how the number of competitive arms $C(\theta^*)$ depends on the value of θ^* and the mean reward functions, when $\theta^* = 3$. To identify the competitive arms, we first find the set $\Theta^*(\epsilon) = \{\theta : |\mu_{k^*}(\theta^*) - \mu_{k^*}(\theta)| < \epsilon\}$ for small $\epsilon > 0$. Since Arm 3 (shown in green) is sub-optimal for all $\theta \in \Theta^*(\epsilon)$ it is non-competitive. As a result, $C(\theta^*) = C(3) = 2$. (b) The number of competitive arms depend on the value of θ^*. The grey region illustrates range of θ^* where $C(\theta^*) = 1$ and the yellow region indicates the range of values for which $C(\theta^*) = 2$.

We note that the number of competitive $C(\theta^*)$ arms is a function of the unknown parameter θ^* and the mean reward functions $\mu_k(\theta)$. Figure 4(a) illustrates how $C(\theta^*)$ is determined for the set of reward functions in Figure 2 and when $\theta^* = 3$. If $\theta^* = 3$, arm 2 (shown in red) is optimal. The corresponding confidence set $\Theta^{*(\epsilon)} = [2 - \frac{2\epsilon}{3}, 4 + \frac{2\epsilon}{3}]$ is a slightly expanded version of the range of θ corresponding to the flat part of the reward function around θ^*. Arm 3 (shown in green) has sub-optimal mean reward $\mu_3(\theta)$ for all $\theta \in \Theta^{*(\epsilon)}$, and thus it is non-competitive. On the other hand, Arm 1 (shown in blue) is competitive. Therefore, the number of competitive arms $C(\theta^*) = 2$ when $\theta^* = 3$. Figure 4(b) shows how $C(\theta^*)$ changes with the value of θ^*. When θ^* is outside of $[2, 4]$, i.e., the flat portion of Arm 2, $\Theta^{*(\epsilon)}$ is a much smaller set and it is possible to show that both Arms 1 and 3 are non-competitive. Therefore, the number of competitive arms $C(\theta^*) = 1$ when θ^* is outside $[2, 4]$.

B. Upper Bounds on Regret

Definition 4 (Degree of Non-Competitiveness, ϵ_k): The degree of non-competitiveness ϵ_k of a non-competitive arm k is the largest ϵ for which $\mu_k(\theta) < \max_{\ell \in \mathcal{K}} \mu_\ell(\theta)$ for all $\theta \in \Theta^{*(\epsilon)}$, where $\Theta^{*(\epsilon)} = \{\theta : |\mu_{k^*}(\theta^*) - \mu_{k^*}(\theta)| < \epsilon\}$. In other words, ϵ_k is the largest ϵ for which arm k is $\Theta^{*(\epsilon)}$-non-competitive.

Our first result shows that the expected pulls for non-competitive arms are bounded with respect to time T. Arms with a larger degree of non-competitiveness ϵ_k are pulled fewer times.

Theorem 2 [Expected Pulls of Each of the $K - C(\theta^*)$ Non-Competitive Arms]: If arm k is non-competitive, then the number of times it is pulled by UCB-C is upper bounded as

$$\begin{aligned}\mathbb{E}[n_k(T)] &\leq Kt_0 + \sum_{t=1}^{T} 2Kt^{1-\alpha} + K^3 \sum_{t=Kt_0}^{T} 6\Big(\frac{t}{K}\Big)^{2-\alpha} \\ &= \mathrm{O}(1) \quad \text{for } \alpha > 3. \end{aligned} \tag{3}$$

Here,

$$t_0 = \inf\left\{\tau \geq 2 : \Delta_{\min}, \epsilon_k \geq 4\sqrt{\frac{K\alpha\sigma^2 \log \tau}{\tau}}\right\};$$
$$\Delta_{\min} = \min_{k \in \mathcal{K}} \Delta_k.$$

The O(1) constant depends on the degree of competitiveness ϵ_k through t_0. If ϵ_k is large, it means that t_0 is small and hence $\mathbb{E}[n_k(T)]$ is bounded above by a small constant. The second and third terms in (3) sum up to a constant for $\alpha > 3, \beta > 1$.

The next result shows that expected pulls for any competitive arm is $\mathrm{O}(\log T)$. This result holds for any sub-optimal arm, but for non-competitive arms we have a stronger upper bound (of O(1)) as given in Theorem 2. Regret analysis of UCB-C is presented in Appendix E. In Appendix D, we present a unified technique to prove results for any other ALGORITHM-C, going beyond UCB-C. We present the regret analysis of TS-C (with Beta prior and $K = 2$) in Appendix F.

Theorem 3 [Expected Pulls for Each of the $C(\theta^*) - 1$ Competitive Sub-Optimal Arms]: The expected number of times a competitive sub-optimal arm is pulled by UCB-C Algorithm is upper bounded as

$$\begin{aligned}\mathbb{E}[n_k(T)] &\leq 8\alpha\sigma^2 \frac{\log T}{\Delta_k^2} + \frac{2\alpha}{\alpha - 2} + \sum_{t=1}^{T} 2Kt^{1-\alpha} \\ &= \mathrm{O}(\log T) \quad \text{for } \alpha > 2, \end{aligned}$$

Plugging the results of Theorem 2 and Theorem 3 in (2) yields the bound on the expected regret in Theorem 1. Note that in this work we consider a finite-armed setting where the number of arms K is a fixed constant that does *not* scale with T – we focus on understanding how the cumulative regret scales with T while K remains constant.

C. Proof Sketch

We now present the proof sketch for Theorem 2. The detail proof is given in the Appendix. For UCB-C, the proof can be divided into three steps presented below. The analysis is unique to our paper and allows us to prove that the UCB-C algorithm pulls the non-competitive arms only O(1) times. The key strength of our approach is that the analysis can be easily extended to any ALGORITHM-C.

i) *The Probability of arm k^* Being $\hat{\Theta}_t$-Non-Competitive is Small:* Observe that $\theta^* \in \hat{\Theta}_t$ implies that k^* is $\hat{\Theta}_t$-competitive. Let $E_1(t)$ denote the event that the optimal arm k^* is $\hat{\Theta}_t$-non-competitive at round t. As we obtain more and more samples, the probability of θ^* lying outside $\hat{\Theta}_t$ decreases with t. Using this, we show that (viz. Lemma 3 in the Appendix)

$$\Pr(E_1(t)) \leq 2Kt^{1-\alpha}. \tag{4}$$

This enable us to bound the expected number of pulls of a competitive arm as follows.

$$\mathbb{E}[n_k(t)] \leq \sum_{t=1}^{T} \Pr(E_1(t)) + \sum_{t=0}^{T-1} \Pr(I_k(t) > I_{k^*}(t), k_{t+1} = k). \tag{5}$$

In view of (4), the first term in (5) sums up to a constant for $\alpha > 2$. The term $I_k(t)$ represents the UCB Index if the last step in the algorithm is UCB, i.e., $I_k(t) = \hat{\mu}_k(t) + \sqrt{\frac{2\alpha\sigma^2 \log t}{n_k(t)}}$. The analysis of second term is exactly same as that for the UCB algorithm [7]. Due to this, the upper bound on expected number of pulls of competitive arms using UCB-C has the same pre-log constants as UCB.

ii) *The Probability of a Non-Competitive Arm Being Pulled Jointly With the Event That $n_{k^*}(t) > t/K$ is Small:* Consider the joint event that a non-competitive arm with parameter ϵ_k is pulled at round $t+1$ and the number of pulls of optimal arm till round t is at least t/K. In Lemma 4 in Appendix C we show that this event is unlikely. Intuitively, this is because when arm k^* is pulled sufficiently many times, the confidence interval of mean of optimal arm is unlikely to contain any value outside $[\mu_{k^*}(\theta^*) - \epsilon_k, \mu_{k^*}(\theta^*) + \epsilon_k]$. Due to this, with high probability, arm k is eliminated for round $t+1$ in step 2 of the algorithm itself. This leads to the result of Lemma 4 in the Appendix,

$$\Pr\Big(k_{t+1} = k, n_{k^*}(t) > \frac{t}{K}\Big) \leq 2t^{1-\alpha} \quad \forall t > t_0 \tag{6}$$

iii) *The Probability That a Sub-Optimal Arm is Pulled More Than $\frac{t}{K}$ Times Till Round t is Small:* In Lemma 6 in the Appendix, we show that

$$\Pr\Big(n_k(t) > \frac{t}{K}\Big) \leq 6K^2\Big(\frac{t}{K}\Big)^{2-\alpha} \quad \forall t > Kt_0. \tag{7}$$

This result is specific to the last step used in ALGORITHM-C. To show (7) we first derive an intermediate result for UCB-C which states that

$$\Pr(k_{t+1} = k, n_k(t) \geq s) \leq (2K+4)t^{1-\alpha} \quad \text{for } s \geq \frac{t}{2K}.$$

Intuitively, if we have large number of samples of arm k, its UCB index is likely to be close to μ_k, which is unlikely to be larger than the UCB index of optimal arm k^* (which is around μ_{k^*} if n_{k^*} is also large, or even higher if n_{k^*} is *small* due to the exploration term added in UCB index).

The analysis of steps ii) and iii) are unique to our paper and help us obtain the O(1) regret for non-competitive arms. Using these results, we can write the expected number of pulls for a non-competitive arm as

$$\mathbb{E}[n_k(t)] \leq Kt_0 + \sum_{t=Kt_0}^{T-1} \Pr\Big(k_{t+1} = k, n_{k^*}(t) = \max_{k \in \mathcal{K}} n_k(t)\Big) + \sum_{t=Kt_0}^{T-1} \sum_{k \in \mathcal{K}, k \neq k^*} \Pr\Big(n_k(t) = \max_{k \in \mathcal{K}} n_k(t)\Big). \tag{8}$$

The second term in (8) is bounded through step ii) (viz. (6)) and the third term in (8) is bounded for each sub-optimal arm through step iii) (viz. (7)). Together, steps ii) and iii) imply that the expected number of pulls for a non-competitive arm is bounded.

D. Discussion on Regret Bounds

Reduction in the Effective Number of Arms: The classical multi-armed bandit algorithms, which are agnostic to the structure of the problem, pull each of the $(K-1)$ sub-optimal arms $O(\log T)$ times. In contrast, our UCB-C algorithm pulls only a *subset* of the sub-optimal arms $O(\log T)$ times, with the rest (i.e., non-competitive arms) being pulled only O(1) times. More precisely, our algorithms pull each of the $C(\theta^*) - 1 \leq K - 1$ arms that are competitive but sub-optimal $O(\log T)$ times. It is important to note that the upper bound on the pulls of these competitive arms by UCB-C has the same pre-log constants with that of the UCB, as shown in Theorem 1. Consequently, the ability of UCB-C to reduce the pulls of non-competitive arms from $O(\log T)$ to O(1) results directly in it achieving a smaller cumulative regret than its non-structured counterpart.

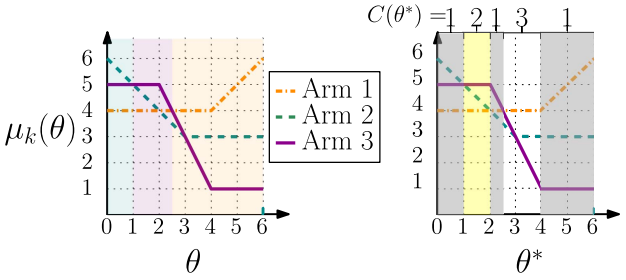

Fig. 5. (left) Arm 2 is optimal for $\theta^* \in [0,1]$, Arm 3 is optimal for $\theta^* \in [1, 2.5]$ and Arm 1 is optimal for $\theta^* \in [2.5, 6]$, (right) the number of competitive arms for different ranges of θ shaded in grey ($C(\theta) = 1$), yellow ($C(\theta) = 2$) and white ($C(\theta) = 3$).

The number of competitive arms, i.e., $C(\theta^*)$, depends on the functions $\mu_1(\theta), \ldots, \mu_K(\theta)$ *as well as* the hidden parameter θ^*. Depending on θ^*, it is possible to have $C(\theta^*) = 1$, or $C(\theta^*) = K$, or any number in between. When $C(\theta^*) = 1$, all sub-optimal arms are non-competitive due to which our proposed algorithms achieve O(1) regret. What makes our algorithms appealing is the fact that even though they do not explicitly try to predict the set (or, the number) of competitive arms, they *stop* pulling any non-competitive arm after finitely many steps.

Empirical Performance of ALGORITHM-C*:* In Figure 6 we compare the regret of ALGORITHM-C against the regret of ALGORITHM (UCB/TS/KL-UCB). We plot the cumulative regret attained under ALGORITHM-C vs. ALGORITHM of the example shown in Figure 5 for three different values of θ^*:0.5, 1.5 and 2.6. Refer to Figure 5 to see that $C = 1$, 2 and 3 for $\theta^* = 0.5$,1.5 and 2.6, respectively. Due to this, we see that ALGORITHM-C achieves bounded regret for $\theta^* = 0.5$, and reduced regret relative to ALGORITHM for $\theta^* = 1.5$ as only one arm is pulled $O(\log T)$ times. For $\theta^* = 2.6$, even though $C = 3$ (i.e., all arms are competitive), ALGORITHM-C achieves empirically smaller regret than ALGORITHM. We also see the advantage of using TS-C and KL-UCB-C over UCB-C in Figure 6 as Thompson Sampling and KL-UCB are known to outperform UCB empirically. For all the simulations, we set $\alpha = 3, \beta = 1$. Rewards are drawn from the distribution $\mathcal{N}(\mu_k(\theta^*), 4)$, i.e., $\sigma = 2$. We average the regret over 100 experiments. For a given experiment, all algorithms use the same reward realizations.

Performance Comparison With UCB-S: In the first row of Figure 6, we also plot the performance of the UCB-S algorithm proposed in [19], alongside UCB and UCB-C. The UCB-S algorithm constructs the confidence set $\hat{\Theta}_t$ just like UCB-C, and then in the next step selects the arm $k_{t+1} = \arg\max_{k \in \mathcal{K}} \sup_{\theta \in \hat{\Theta}_t} \mu_k(\theta)$. Informally, it finds the maximum possible mean reward $\mu_k(\theta)$ over $\theta \in \hat{\Theta}$ for each arm k. As a result, UCB-S tends to favor pulling arms that have the largest

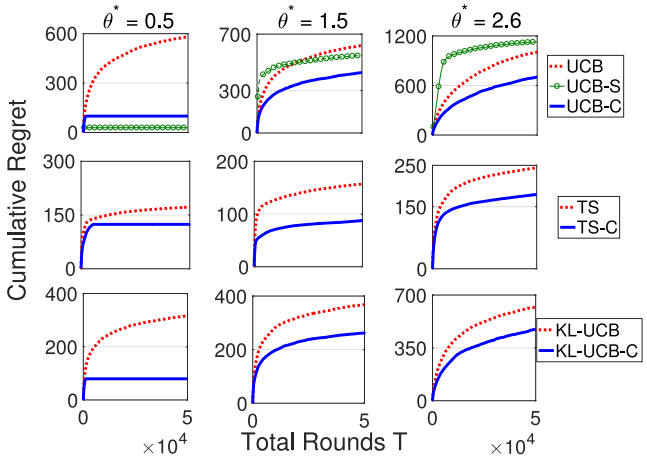

Fig. 6. Cumulative regret of ALGORITHM-C vs. ALGORITHM (UCB in row 1, TS in row 2 and KL-UCB in row 3) for the setting in Figure 5. The number of competitive arms is $C(\theta^*) = 1$ in the first column, $C(\theta^*) = 2$ in second column and $C(\theta^*) = 3$ in third column. Unlike UCB-S which only extends UCB, our approach generalizes any classical bandit algorithm such as UCB, TS, and KL-UCB to the structured bandit setting.

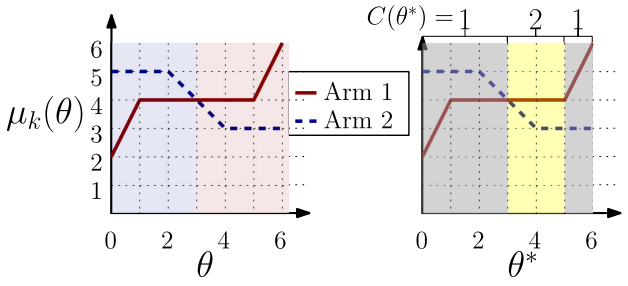

Fig. 7. Arm 2 is optimal for $\theta^* \in [0, 3]$ and Arm 1 is optimal for $\theta^* \in [3, 5]$. For $\theta \in [0, 3] \cup [5, 6]$, $C(\theta) = 1$ and $C(\theta) = 2$ for $\theta \in [3, 5]$.

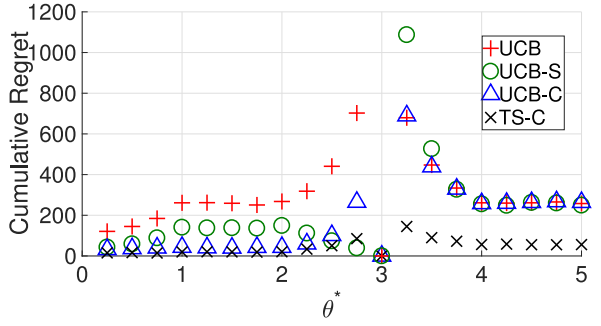

Fig. 8. Cumulative regret of UCB,UCB-S,UCB-C and TS-C versus θ^* for the example in Figure 7 over 50000 runs. UCB-S is sensitive to the value of θ^* and the reward functions as it is seen to achieve a small regret for $\theta^* = 2.75$, but obtains a worse regret than UCB for $\theta^* = 3.25$.

mean reward for $\theta \in \Theta^{*(\epsilon)}$. This bias renders the performance of UCB-S to depend heavily on θ^*. When $\theta^* = 0.5$, UCB-S has the smallest regret among the three algorithms compared in Figure 6, but when $\theta^* = 2.6$ it gives even worse regret than UCB. A similar observation can be made in another simulation setting described below.

Figure 8 compares UCB, UCB-S, UCB-C and TS-C for the functions shown in Figure 7. We plot the cumulative regret after 50000 rounds for different values of $\theta^* \in [0, 5]$ and

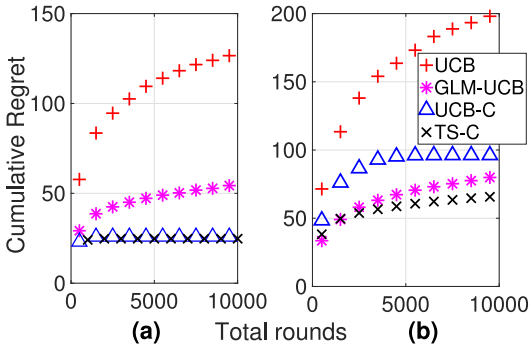

Fig. 9. Cumulative regret of UCB, GLM-UCB, UCB-C and TS-C in the linear bandit setting, with $x_1 = (2, 1)$, $x_2 = (1, 1.5)$ and $x_3 = (3, -1)$. Mean rewards are $(\vec{\theta}^*)^\intercal x_k$, with $\theta^* = (0.9, 0.9)$ in (a) and $\theta^* = (0.5, 0.5)$ in (b). While UCB-C and TS-C are designed for a much broader class of problems, they show competitive performance relative to GLM-UCB, which is a specialized algorithm for the linear bandit setting.

observe that TS-C performs the best for most θ^* values. As before, the performance of UCB-S varies significantly with θ^*. In particular, UCB-S has the smallest regret of all when $\theta^* = 2.75$, but achieves worse regret even compared to UCB when $\theta^* = 3.25$. On the other hand, our UCB-C performs better than or at least as good as UCB for all θ^*. While UCB-S also achieves the regret bound of Theorem 1, the ability to employ any ALGORITHM in the last step of ALGORITHM-C is a key advantage over UCB-S, as Thompson Sampling and KL-UCB can have significantly better empirical performance over UCB.

Comparison in Linear Bandit and Multi-Dimensional θ Settings: As highlighted in Section II, our problem formulation allows θ to be multi-dimensional as well. Figure 9 shows the performance of UCB-C and TS-C relative to GLM-UCB in a linear bandit setting. In a linear bandit setting, mean reward of arm k is $\mu_k(\theta^*) = (\theta^*)^\intercal x_k$. Here x_k is a vector associated with arm k, which is known to the player. The parameter θ^* is unknown to the player, and hence it fits in our structured bandit framework. It is important to see that while UCB-C and TS-C are designed for a much broader class of problems, they still show competitive performance relative to specialized algorithms (i.e., GLM-UCB) in the linear bandit setting (Figure 9). Figure 10 shows a setting in which θ is multi-dimensional, but the reward mappings are non-linear and hence the setting is not captured through a linear bandit framework. Our results in Figure 10 demonstrate that the UCB-C and TS-C algorithms work in such settings as well while providing significant improvements over UCB in certain cases.

E. When Do We Get Bounded Regret?

When $C(\theta^*) = 1$, all sub-optimal arms are pulled only O(1) times, leading to a bounded regret. Cases with $C(\theta^*) = 1$ can arise quite often in practical settings. For example, when functions are continuous or Θ is countable, this occurs when the optimal arm k^* is *invertible*, or has a unique maximum at $\mu_{k^*}(\theta^*)$, or any case where the set $\Theta^* = \{\theta : \mu_{k^*}(\theta) = \mu_{k^*}(\theta^*)\}$ is a *singleton*. These cases lead to all sub-optimal arms being non-competitive, whence UCB-C achieves bounded (i.e., $O(1)$) regret. There are more general scenarios where bounded

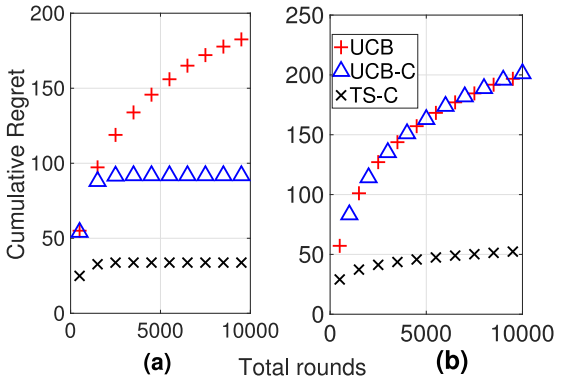

Fig. 10. Cumulative regret for UCB,UCB-C and TS-C for the case in which $\theta \in [-1,1] \times [-1,1]$. The reward functions are $\mu_1(\vec{\theta}) = \theta_1 + \theta_2$, $\mu_2(\vec{\theta}) = \theta_1 - \theta_2$, and $\mu_3(\vec{\theta}) = \max(|\theta_1|, |\theta_2|)$. The true parameter $\vec{\theta}^*$ is (0.9, 0.2) in (a) and $(-0.2, 0.1)$ in (b). The value of $C(\theta^*)$ is 1, 3 in (a) and (b) respectively.

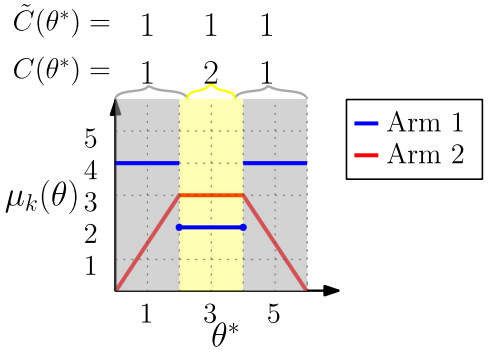

Fig. 11. For values of $\theta \in [2,4]$ Arm 2 is Θ^*-Non-Competitive but it is still Competitive. As for any set slightly bigger than Θ, i.e., $\Theta^{*(\epsilon)}$, it is $\Theta^{*(\epsilon)}$-Competitive. Hence this is one of the corner case situations where $C(\theta)$ and $\tilde{C}(\theta^*)$ are different.

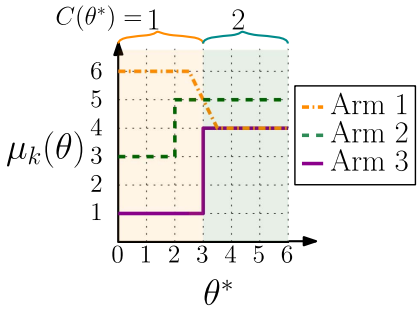

Fig. 12. In this example, Arm 3 has $\mu_3(\theta^*) = 1$ for $\theta^* < 3$ and $\mu_3(\theta^*) = 4$ for $\theta^* \geq 3$. See that Arm 3 is sub-optimal for all values of θ^*, and hence is non-competitive for all θ^*. However, a few pulls of Arm 3 can still be useful in getting some information on whether $\theta^* \geq 3$ or $\theta^* < 3$.

regret is possible. To formally present such cases, we utilize a lower bound obtained in [20].

Proposition 1 (Lower Bound): For any uniformly good algorithm [1], and for any $\theta \in \Theta$, we have:

$$\lim_{T\to\infty} \inf \frac{Reg(T)}{\log T} \geq L(\theta), \text{ where}$$

$$L(\theta) = \begin{cases} 0 & \text{if } \tilde{C}(\theta^*) = 1, \\ > 0 & \text{if } \tilde{C}(\theta^*) > 1. \end{cases}$$

An algorithm π is uniformly good if $Reg^{\pi}(T,\theta) = o(T^a)$ for all $a > 0$ and all $\theta \in \Theta$. Here $\tilde{C}(\theta^*)$ is the number of arms that are Θ^*-Competitive, with Θ^* being the set $\{\theta : \mu_{k^*}(\theta) = \mu_{k^*}(\theta^*)\}$.

This suggests that bounded regret is possible only when $\tilde{C}(\theta) = 1$ and logarithmic regret is unavoidable in all other cases. The proof of this proposition follows from a bound derived in [20] and it is given in Appendix B.

There is a subtle difference between $C(\theta^*)$ and $\tilde{C}(\theta^*)$. This arises in corner case situations when a Θ^*-Non-Competitive arm is competitive. Note that the set $\Theta^* = \{\theta : \mu_{k^*}(\theta^*) = \mu_{k^*}(\theta)\}$ can be interpreted as the confidence set obtained when we pull the optimal arm k^* infinitely many times. In practice, if we sample the optimal arm a *large* number of times, we can only obtain the confidence set $\Theta^{*(\epsilon)} = \{\theta : |\mu_{k^*}(\theta^*) - \mu_{k^*}(\theta)| < \epsilon\}$ for some $\epsilon > 0$. Due to this, there is a difference between $\tilde{C}(\theta^*)$ and $C(\theta^*)$. Consider the case shown in Figure 11 with $\theta^* = 3$. For $\theta^* = 3$, Arm 1 is optimal. In this case $\Theta^* = [2,4]$. For all values of $\theta \in \Theta^*$, $\mu_2(\theta) \leq \mu_1(\theta)$ and hence Arm 2 is Θ^*-Non-Competitive. However, for any $\epsilon > 0$, Arm 2 is $\Theta^{*(\epsilon)}$-competitive and hence Competitive. Due to this, we have $\tilde{C}(3) = 1$ and $C(3) = 2$ in this case.

If Θ is a countable set, a Θ^*-Non-Competitive arm is always $\Theta^{*(\epsilon)}$-Non-Competitive, that is, $\tilde{C}(\theta^*) = C(\theta^*)$. This occurs because one can always choose $\epsilon = \min_{\{\theta \in \Theta \setminus \Theta^*\}} \{|\mu_{k^*}(\theta^*) - \mu_{k^*}(\theta)|\}$ so that a Θ^*-Non-Competitive arm is also $\Theta^{*(\epsilon)}$-Non-Competitive. This shows that when Θ is a countable set (which is true for most practical situations where the hidden parameter θ is *discrete*), UCB-C achieves bounded regret *whenever possible*, that is, whenever $\tilde{C}(\theta^*) = 1$. While this property holds true for the case when Θ is a countable set, there can be more general cases where $C(\theta) = \tilde{C}(\theta)$. Our algorithms and regret analysis are valid regardless of Θ being countable or not.

VI. Additional Exploration of Non-Competitive but Informative Arms

The previous discussion shows that the UCB-C and TS-C algorithms enable substantial reductions in the effective number of arms and the expected cumulative regret. A strength of the proposed algorithms that can be a weakness in some cases is that they stop pulling non-competitive arms that are unlikely to be optimal after some finite number of steps. Although an arm may be non-competitive in terms of its reward yield, it can be useful in inferring the hidden parameter θ^*, which in turn may help reduce the regret incurred in subsequent steps. For instance, consider the example shown in Figure 12. Here, Arm 3 is sub-optimal for all values of $\theta^* \in [0,6]$ and is never pulled by UCB-C, but it can help identify whether $\theta^* \geq 3$ or $\theta^* < 3$. Motivated by this, we propose an add-on to Algorithm-C, named as the Informative Algorithm-C (Algorithm 3), that takes the *informativeness* of arms into account and performs additional exploration of the *most informative arm* with a probability that decreases over time.

A. Informativeness of an Arm

Intuitively, an arm is informative if it helps us to obtain information about the hidden parameter θ^*. At the end of

round t, we know a confidence interval $\hat{\Theta}_t$ for the hidden parameter θ^*. We aim to quantify the informativeness of an arm with respect to this confidence set $\hat{\Theta}_t$. For instance, if $\hat{\Theta}_t \in [2, 4]$ in Figure 12, we see that the reward function of Arm 3 $\mu_3(\theta)$ has high variance and it suggests that the samples of Arm 3 could be helpful in knowing about θ^*. On the other hand, samples of Arm 2 will not be useful in identifying θ^* if $\hat{\Theta}_t = [2, 4]$. There can be several ways of defining the informativeness $I_k(\hat{\Theta}_t)$ of an arm with respect to set $\hat{\Theta}_t$. We consider the following two metrics in this article.

KL-Divergence: Assuming that θ has a uniform distribution in $\hat{\Theta}_t$, we can define the informativeness $I_k(\hat{\Theta}_t)$ of an arm as the expected KL-Divergence between two samples of arm k, i.e., $I_k(\hat{\Theta}_t) = \mathbb{E}_{\theta_1,\theta_2}[D_{KL}(f_{R_k}(R_k|\theta_1), f_{R_k}(R_k|\theta_2))]$. Our intuition here is that larger expected KL-divergence for an arm indicates that samples from it have substantially different distributions under different θ^* values, which in turn indicates that those samples will be useful in inferring the true value of θ^*. Assuming that $\Pr(R_k|\theta)$ is a Gaussian distribution with mean $\mu_k(\theta)$ and variance σ^2, then the expected KL-Divergence can be simplified as

$$\begin{aligned}
&\mathbb{E}_{\theta_1,\theta_2}\left[D_{KL}\left(f_{R_k}(R_k|\theta_1), f_{R_k}(R_k|\theta_2)\right)\right] \\
&= \mathbb{E}_{\theta_1,\theta_2}\left[D_{KL}\left(\mathcal{N}\left(\mu_k(\theta_1), \sigma^2\right), \mathcal{N}\left(\mu_k(\theta_2), \sigma^2\right)\right)\right] \\
&= \mathbb{E}_{\theta_1,\theta_2}\left[\frac{1}{2}\left(\mu_k(\theta_1) - \mu_k(\theta_2)\right)^2\right] \\
&= \int_{\hat{\Theta}_t}\left(\mu_k(\theta) - \int_{\hat{\Theta}_t} \mu_k(\theta)U(\theta)\mathrm{d}\theta\right)^2 U(\theta)\mathrm{d}\theta = V_k\left(\hat{\Theta}_t\right),
\end{aligned}$$

where, $V_k(\hat{\Theta}_t)$ is the variance in the mean reward function $\mu_k(\theta)$, calculated when θ is uniformly distributed over the current confidence set $\hat{\Theta}_t$. Observe that the metric $I_k(\hat{\Theta}_t) = V_k(\hat{\Theta}_t)$ is easy to evaluate given the functions $\mu_k(\theta)$ and the confidence set obtained from Step 1.

Entropy: Alternatively, $\mu_k(\theta)$ can be viewed as a derived random variable of θ, where θ is uniformly distributed over the current confidence set $\hat{\Theta}_t$. The informativeness of arm k can then be defined as $I_k(\hat{\Theta}_t) = H(\mu_k(\theta))$. When $\mu_k(\theta)$ is discrete this will be the Shannon entropy $H(\mu_k(\theta)) = \sum_{\theta\in\hat{\Theta}_t} -\Pr(\mu_k(\theta))\log(\Pr(\mu_k(\theta)))$, while for continuous $\mu_k(\theta)$ it will be the *differential entropy* $H(\mu_k(\theta)) = \int_{\hat{\Theta}_t} -f_{\mu_k(\theta)}\log(f_{\mu_k(\theta)})d(\mu_k(\theta))$ where $f_{\mu_k(\theta)}$ is the probability density function of the derived random variable $\mu_k(\theta)$. Observe that differential entropy takes into account the shape as well as the range of $\mu_k(\theta)$. For example, if two reward functions are linear in θ, the one with a higher slope will have higher differential entropy, as we would desire from an informativeness metric. Evaluating the differential entropy in $\mu_k(\theta)$, i.e., can be computationally challenging.

Other than the two metrics described above, there might be alternative (and potentially more complicated) ways of quantifying the informativeness of an arm. Another candidate would be the *information gain* metric proposed in [25], which defines informativeness in terms of identifying the best arm, rather than inferring θ^*. However, as already mentioned in [25] by the authors, information gain is computationally challenging to implement in practice outside of certain specific class of problems where prior distribution of θ is Beta or Gaussian.

Algorithm 3 Informative UCB-C

1: Steps 1 to 5 as in Algorithm 1

2: **Identify $k_{\hat{\Theta}_t}$, i.e., the most informative arm for set $\hat{\Theta}_t$:** $k_{\hat{\Theta}_t} = \arg\max_{k\in\mathcal{K}} I_k(\hat{\Theta}_t)$

3: **Play informative arm with probability $\frac{\gamma}{t^d}$, play UCB-C otherwise:**

$$k_{t+1} = \begin{cases} k_{\hat{\Theta}_t} & \text{w.p. } \frac{\gamma}{t^d}, \\ \arg\max_{k\in\mathcal{C}_t}\left(\hat{\mu}_k(t) + \sqrt{\frac{2\alpha\sigma^2\log t}{n_k(t)}}\right) & \text{w.p. } 1 - \frac{\gamma}{t^d} \end{cases}$$

4: Update empirical mean, $\hat{\mu}_k$ and n_k for arm k_{t+1}.

B. Proposed Informative Algorithm-C and Its Expected Regret

Given an informativeness metric $I_k(\hat{\Theta}_t)$, we define the most informative arm for the confidence set $\hat{\Theta}_t$ as $k_{\hat{\Theta}_t} = \arg\max_{k\in\mathcal{K}} I_k(\hat{\Theta}_t)$. At round t, Informative Algorithm-C (described in Algorithm 3) picks the most informative arm $k_{\hat{\Theta}_t}$ with probability $\frac{\gamma}{t^d}$ where $d > 1$, and otherwise uses UCB-C or TS-C to pull one of the competitive arms. Here, γ and d are hyperparameters of the Informative UCB-C algorithm. Larger γ or small d results in more exploration during the initial rounds. Setting the probability of pulling the most informative arm as $\frac{\gamma}{t^d}$ ensures that the algorithm pulls the informative arms more frequently at the beginning. This helps shrink $\hat{\Theta}_t$ faster. Setting $d > 1$ ensures that informative but non-competitive arms are only pulled $\sum_{t=1}^{\infty}\frac{\gamma}{t^d} = \mathrm{O}(1)$ times in expectation. Thus, asymptotically the algorithm will behave exactly as the underlying Algorithm-C and the regret of Informative-Algorithm-C is at most an O(1) constant worse than the Algorithm-C algorithm.

C. Simulation Results

We implement two versions of Informative-Algorithm-C, namely ALGORITHM-C-KLdiv and ALGORITHM-C-Entropy, which use the KL-divergence and Entropy metrics respectively to identify the most informative arm $k_{\hat{\Theta}_t}$ at round t. ALGORITHM-C-KLdiv picks the arm with highest variance in $\hat{\Theta}_t$, i.e., $I_k(\hat{\Theta}_t) = \arg\max_k V_k(\hat{\Theta}_t)$. ALGORITHM-C-Entropy picks an arm whose mean reward function, $\mu_k(\theta)$, has largest shannon entropy for $\theta \in \hat{\Theta}_t$ (assuming θ to be a uniform random variable in $\hat{\Theta}_t$). As a baseline for assessing the effectiveness of the informativeness metrics, we also implement ALGORITHM-C-Random which selects $k_{\hat{\Theta}_t}$ by sampling one of the arms uniformly at random from the set of all arms $\mathcal{K}$ at round t.

Figure 13 shows the cumulative regret of the aforementioned algorithms for the reward functions shown in Figure 12, where the hidden parameter $\theta^* = 3.1$. Among UCB-C, UCB-C-KLdiv, UCB-C-Entropy and UCB-C-Random, UCB-C-KLdiv

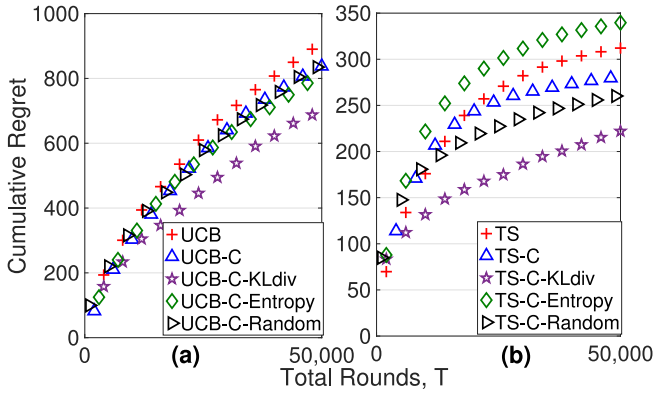

Fig. 13. Performance comparison of ALGORITHM, ALGORITHM-C and Informative ALGORITHM-C algorithms (with parameter $\gamma = 30$, $d = 1.1$) for the example shown in Figure 12 with $\theta^* = 3.1$. UCB-C, TS-C do not pull Arm 3 at all, but UCB-C-KLdiv, TS-C-KLdiv pull it in the initial rounds to determine whether $\theta^* > 3$ or not. As a result, UCB-C-KLdiv and TS-C-KLdiv shrink $\hat{\Theta}_t$ faster initially and have a better empirical performance than UCB-C and TS-C, while retaining similar regret guarantees of UCB-C and TS-C respectively.

has the smallest cumulative regret. This is because UCB-C-KLdiv identifies Arm 3 as the most informative arm, samples of which are helpful in identifying whether $\theta^* > 3$ or $\theta^* < 3$. Hence, occasional pulls of Arm 3 lead to fast shrinkage of the set $\hat{\Theta}_t$. In contrast to UCB-C-KLdiv, UCB-C-Entropy identifies Arm 2 as the most informative arm for $\hat{\Theta} = [0, 6]$, due to which UCB-C-Entropy samples Arm 2 more often in the initial stages of the algorithm. As the information obtained from Arm 2 is relatively less useful in deciding whether $\theta^* > 3$ or not, we see that UCB-C-Entropy/TS-C-Entropy does not perform as well as UCB-C-KLdiv/TS-C-KLdiv in this scenario. UCB-C-Random picks the most informative arm by selecting an arm uniformly at random from the available set of arms. The additional exploration through random sampling is helpful, but the cumulative regret is larger than UCB-C-KLdiv as UCB-C-Random pulls Arm 3 fewer times relative to UCB-C-KLdiv. For this particular example, cumulative regret of UCB-S was 2500, whereas other UCB style algorithms achieve cumulative regret of 600-800 as shown in the Figure 13(a). This is due to the preference of UCB-S to pick Arm 1 in this example. We see similar trends among TS-C, TS-C-KLdiv, TS-C-Entropy and TS-C-Random. The cumulative regret is smaller for Thompson sampling variants as Thompson sampling is known to outperform UCB empirically.

We would like to highlight that the additional exploration by Informative-Algorithm-C is helpful only in cases where non-competitive arms help significantly shrink the confidence set $\hat{\Theta}_t$. For the experimental setup presented in Section VII below, the reward functions are mostly flat as seen in Appendix G and thus, Informative Algorithm-C does not give a significant improvement over the corresponding Algorithm-C. Therefore for clarity of the plots, we do not present experiments results for Informative-C in other settings of this article.

VII. EXPERIMENTS WITH MOVIELENS DATA

We now show the performance of UCB-C and TS-C on a real-world dataset. We use the MOVIELENS dataset [43] to demonstrate how UCB-C and TS-C can be deployed in practice and demonstrate their superiority over classical UCB and TS. Since movie recommendations is one of many applications of structured bandits, we do not compare with methods such as collaborative filtering that are specific to recommendation systems. Also, we do not compare with contextual bandits since the structured bandit setting has a different goal of making recommendations *without accessing a user's contextual features*.

The MOVIELENS dataset contains a total of 1M ratings made by 6040 users for 3883 movies. There are 106 different user *types* (based on having distinct age and occupation features) and 18 different genres of movies. The users have given ratings to the movies on a scale of 1 to 5. Each movie is associated with one (and in some cases, multiple) genres. For the experiments, of the possibly multiple genres for each movie, we choose one uniformly at random. The set of users that belong to a given type is referred to as a *meta-user*; thus there are 106 different meta-users. These 106 different meta-users correspond to the different values that the hidden parameter θ can take in our setting. For example, one of the meta-users in the data-set represents college students whose age is between 18 and 24, and this corresponds to the case $\theta^* = 25$. We split the dataset into two equal parts, training and test. This split is done at random, while ensuring that the training dataset has samples from all 106 meta-users.

For a particular meta-user whose features are unknown (i.e., the true value of θ is hidden), we need to sequentially choose one of the genres (i.e., one of the arms) and recommend a movie from that genre to the user. In doing so, our goal is to maximize the *total* rating given by this user to the movies we recommended. We use the training dataset (50% of the whole data) to learn the mean reward mappings from meta-users (θ) to different genres (arms); these mappings are shown in Appendix G. The learned mappings indicate that the mean-reward mappings of meta-users for different genres are related to one another. For example, on average 56+ year old retired users may like documentaries more than children's movies. In our experiments, these dependencies are learned during the training. In practical settings of recommendations or advertising, these mappings can be learned from pilot surveys in which users participate with their consent.

We test the algorithm for three different meta-users, i.e., for three different values of θ^*. The movie rating samples for these meta-users are obtained from the test dataset, (the remaining 50% of the data). Figure 14 shows that UCB-C and TS-C achieve significantly lower regret than UCB, TS as only a few arms are pulled $O(\log T)$ times. This is because only $C(\theta^*) - 1$ of the sub-optimal arms are pulled $O(\log T)$ times by our UCB-C and TS-C algorithms. For our experimental setting, the value of C depends on θ^* (which is unknown to the algorithm). Figure 15 shows how $C(\theta^*)$ varies with θ^*, where it is seen that $C(\theta^*)$ is significantly smaller than K for all θ^*. As a result, the performance improvements observed in Figure 14 for our UCB-C and TS-C algorithms will apply to other θ^* values as well. There are θ^* values for which UCB-C is better than UCB-S, and vice versa. But, TS-C always outperforms UCB-C and UCB-S in our experiments. We tried

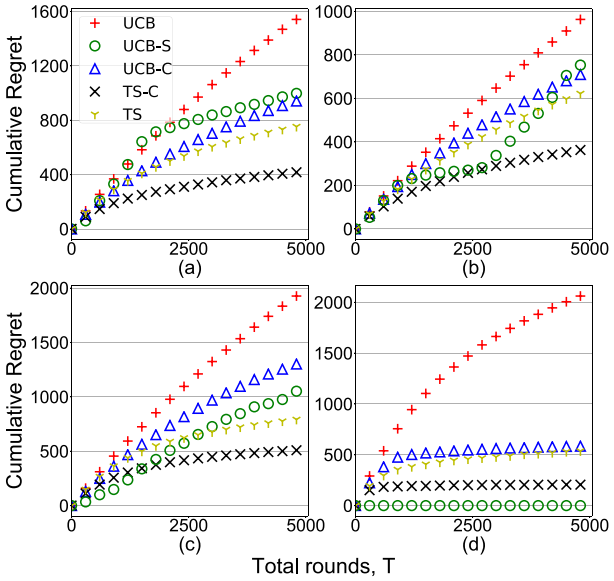

Fig. 14. Regret plots for UCB, UCB-S, UCB-C, TS and TS-C for (a) $\theta^* = 67$ (35-44 year old grad/college students), (b) $\theta^* = 87$ (45-49 year old clerical/admin), (c) $\theta^* = 25$ (18-24 year old college students) and (d) $\theta^* = 93$ (56+ Sales and Marketing employees). The value of $C(\theta^*)$ is 6, 6, 3 and 1 for (a), (b), (c) and (d) respectively – in all cases $C(\theta^*)$ is much smaller than $K = 18$.

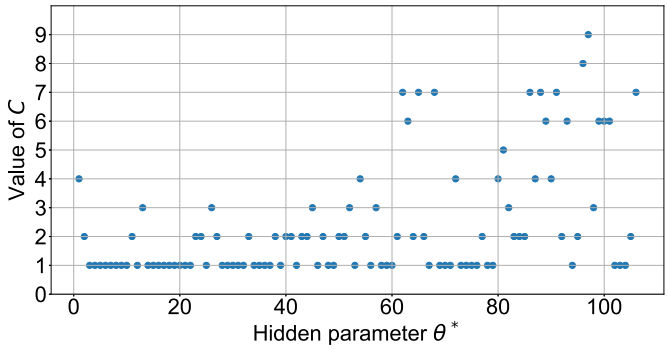

Fig. 15. The value of $C(\theta^*)$ varies with the unknown hidden parameter θ^* (i.e., the age and occupation of the anonymous user). We see that for all θ^*, $C(\theta^*) < K$. While the total number of arms, $K = 18$, the value of $C(\theta^*)$ ranges between 1 and 9. This suggests that the ALGORITHM-C approach can lead to significant performance improvement for this problem.

Informative UCB-C in this setting as well, but the results were similar to that of UCB-C because the arms in this setting are not too informative.

VIII. CONCLUDING REMARKS

In this work, we studied a structured bandit problem in which the mean rewards of different arms are related through a common hidden parameter. Our problem setting makes no assumptions on mean reward functions, due to which it subsumes several previously studied frameworks [10], [12], [15]. We developed an approach that allows us to extend a classical bandit ALGORITHM to the structured bandit setting, which we refer to as ALGORITHM-C. We provide a regret analysis of UCB-C (structured bandit versions of UCB). A key insight from this analysis is that ALGORITHM-C pulls only $C(\theta^*) - 1$ of the $K - 1$ sub-optimal arms $O(\log T)$ times and all other arms, termed as *non-competitive* arms, are pulled only O(1) times. Through experiments on the MOVIELENS dataset, we demonstrated that UCB-C and TS-C give significant improvements in regret as compared to previously proposed approaches. Thus, the main implication of this article is that it provides a unified approach to exploit the structured rewards to drastically reduce exploration in a principled manner.

For cases where non-competitive arms can provide information about θ that can shrink the confidence set $\hat{\Theta}_t$, we propose a variant of ALGORITHM-C called informative-ALGORITHM-C that takes the informativeness of arms into account without increasing unnecessary exploration. Linear bandit algorithms [16], [17], [44] shrink the confidence set $\hat{\Theta}_t$ in a better manner by taking advantage of the linearity of the mean reward functions to estimate θ^* as the solution to least squares problem [44]. Moreover, linearity helps them to use self-normalized concentration bound for vector valued martingale, ([44, Th. 1]) to construct the confidence intervals. Extending this approach to the general structured bandit setting is a non-trivial open question due to the absence of constraints on the nature of mean reward functions $\mu_k(\theta)$. The paper [20] proposes a statistical hypothesis testing method for the case of known conditional reward distributions. Generalizing it to the setting considered in this article is an open future direction. While we state our results for a scenario where mean reward functions are known, our algorithmic approach, analysis and results can also be extended to a setting where only lower and upper bounds on the mean reward function $\mu_k(\theta)$ are known. This setting is discussed in Appendix A. Another open direction in this field is to study the problem of structured best-arm identification where the goal is to conduct pure exploration and identify the best arm in the fewest number of rounds.

REFERENCES

[1] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.
[2] S. S. Villar, J. Bowden, and J. Wason, "Multi-armed bandit models for the optimal design of clinical trials: Benefits and challenges," *Stat. Sci. Rev. J. Inst. Math. Stat.*, vol. 30, no. 2, p. 199, 2015.
[3] C. Tekin and E. Turgay, "Multi-objective contextual multi-armed bandit problem with a dominant objective," 2017. [Online]. Available: arXiv:1708.05655.
[4] J. Niño-Mora, "Stochastic scheduling," in *Encyclopedia of Optimization*, vol. 5, 2nd ed. New York, NY, USA: Springer, 2009, pp. 367–372.
[5] J. White, *Bandit Algorithms for Website Optimization*. Sebastopol, CA, USA: O'Reilly Media, 2012.
[6] R. Sen, K. Shanmugam, A. G. Dimakis, and S. Shakkottai, "Identifying best interventions through online importance sampling," in *Proc. PMLR*, 2017, pp. 3057–3066.
[7] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed Bandit problem," *Mach. Learn.*, vol. 47, nos. 2–3, pp. 235–256, 2002.
[8] S. Agrawal and N. Goyal, "Analysis of thompson sampling for the multi-armed bandit problem," in *Proc. Conf. Learn. Theory*, 2012, pp. 39–41.
[9] A. Garivier and O. Cappé, "The KL-UCB algorithm for bounded stochastic bandits and beyond," in *Proc. 24th Annu. Conf. Learn. Theory*, 2011, pp. 359–376.
[10] O. Atan, C. Tekin, and M. van der Schaar, "Global multi-armed bandits with Hölder continuity," in *Proc. AISTATS*, 2015, pp. 28–36.

[11] C. Shen, R. Zhou, C. Tekin, and M. van der Schaar, "Generalized global bandit and its application in cellular coverage optimization," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 218–232, Feb. 2018.

[12] Z. Wang, R. Zhou, and C. Shen, "Regional multi-armed bandits," in *Proc. AISTATS*, 2018, pp. 510–518.

[13] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. ACM 19th Int. Conf. World Wide Web*, 2010, pp. 661–670.

[14] R. Sen, K. Shanmugam, and S. Shakkottai, "Contextual bandits with stochastic experts," in *Proc. 21st Int. Conf. Artif. Intell. Stat.*, vol. 84, 2018, pp. 852–861. [Online]. Available: http://proceedings.mlr.press/v84/sen18a.html

[15] A. J. Mersereau, P. Rusmevichientong, and J. N. Tsitsiklis, "A structured multi-armed bandit problem and the greedy policy," *IEEE Trans. Autom. Control*, vol. 54, no. 12, pp. 2787–2802, Dec. 2009.

[16] S. Filippi, O. Cappe, A. Garivier, and C. Szepesvári, "Parametric bandits: The generalized linear case," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 586–594.

[17] T. Lattimore and C. Szepesvari, "The end of optimism? An asymptotic analysis of finite-armed linear bandits," 2016. [Online]. Available: arXiv:1610.04491.

[18] T. L. Graves and T. L. Lai, "Asymptotically efficient adaptive choice of control laws incontrolled markov chains," *SIAM J. Control Optim.*, vol. 35, no. 3, pp. 715–743, 1997.

[19] T. Lattimore and R. Munos, "Bounded regret for finite-armed structured bandits," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 550–558.

[20] R. Combes, S. Magureanu, and A. Proutière, "Minimal exploration in structured stochastic bandits," in *Proc. NIPS*, 2017, pp. 1763–1771.

[21] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, nos. 3–4, pp. 285–294, Dec. 1933.

[22] S. Agrawal and N. Goyal, "Thompson sampling for contextual bandits with linear payoffs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 127–135. [Online]. Available: http://dl.acm.org/citation.cfm?id=3042817.3043073

[23] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Found. Trends Mach. Learn.*, vol. 5, no. 1, pp. 1–122, 2012.

[24] K. Jamieson and R. Nowak, "Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting," in *Proc. Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2014, pp. 1–6.

[25] D. Russo and B. Van Roy, "Learning to optimize via posterior sampling," *Math. Oper. Res.*, vol. 39, no. 4, pp. 1221–1243, 2014.

[26] A. Gopalan, S. Mannor, and Y. Mansour, "Thompson sampling for complex online problems," in *Proc. 31st Int. Conf. Mach. Learn.*, vol. 32. Bejing, China, Jun. 2014, pp. 100–108. [Online]. Available: http://proceedings.mlr.press/v32/gopalan14.html

[27] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "HyperBand: A novel bandit-based approach to hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6765–6816, Jan. 2017. [Online]. Available: http://dl.acm.org/citation.cfm?id=3122009.3242042

[28] E. Tánczos, R. Nowak, and B. Mankoff, "A KL-LUCB algorithm for large-scale crowdsourcing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5894–5903.

[29] K. G. Jamieson, L. Jain, C. Fernandez, N. J. Glattard, and R. Nowak, "NEXT: A system for real-world development, evaluation, and application of active learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2656–2664.

[30] R. Heckel, N. B. Shah, K. Ramchandran, and M. J. Wainwright. (2016). *Active Ranking From Pairwise Comparisons and the Futility of Parametric Assumptions*. [Online]. Available: http://arxiv.org/abs/1606.08842

[31] S. Bubeck, R. Munos, and G. Stoltz, "Pure exploration in multi-armed bandits problems," in *Proc. Int. Conf. Algorithmic Learn. Theory*, Oct. 2009, pp. 23–37.

[32] J.-Y. Audibert, S. Bubeck, and R. Munosr, "Best arm identification in multi-armed bandits," in *Proc. Annu. Conf. Learn. Theory (COLT)*, 2010, pp. 359–376.

[33] K. Jamieson, M. Malloy, R. Nowak, and S. Bubeck, "LIL'UCB: An optimal exploration algorithm for multi-armed bandits," in *Proc. Conf. Learn. Theory*, 2014, pp. 423–439.

[34] E. Kaufmann, O. Cappé, and A. Garivier, "On the complexity of best-arm identification in multi-armed bandit models," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1–42, 2016.

[35] S. Mannor and J. N. Tsitsiklis, "The sample complexity of exploration in the multi-armed bandit problem," *J. Mach. Learn. Res.*, vol. 5, pp. 623–648, Jun. 2004.

[36] A. Garivier and E. Kaufmann, "Optimal best arm identification with fixed confidence," in *Proc. Annu. Conf. Learn. Theory (COLT)*, vol. 49, Jun. 2016, pp. 998–1027. [Online]. Available: http://proceedings.mlr.press/v49/garivier16a.html

[37] V. Gabillon, M. Ghavamzadeh, and A. Lazaric, "Best arm identification: A unified approach to fixed budget and fixed confidence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 3212–3220.

[38] M. Soare, A. Lazaric, and R. Munos, "Best-arm identification in linear bandits," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 828–836. [Online]. Available: http://papers.nips.cc/paper/5460-best-arm-identification-in-linear-bandits.pdf

[39] R. Huang, M. M. Ajallooeian, C. Szepesvári, and M. Müller, "Structured best arm identification with fixed confidence," in *Proc. Int. Conf. Algorithm. Learn. Theory (ALT)*, vol. 76, Oct. 2017, pp. 593–616. [Online]. Available: http://proceedings.mlr.press/v76/huang17a.html

[40] C. Tao, S. Blanco, and Y. Zhou, "Best arm identification in linear bandits with linear dimension dependency," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 80, Jul. 2018, pp. 4877–4886. [Online]. Available: http://proceedings.mlr.press/v80/tao18a.html

[41] O. Chapelle and L. Li, "An empirical evaluation of thompson sampling," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2249–2257.

[42] D. Russo, B. V. Roy, A. Kazerouni, and I. Osband. (2017). *A Tutorial on Thompson Sampling*. [Online]. Available: http://arxiv.org/abs/1707.02038

[43] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interactive Intell. Syst.*, vol. 5, no. 4, p. 19, 2015.

[44] Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári, "Improved algorithms for linear stochastic bandits," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2312–2320.