An Explore-then-Commit Algorithm for Submodular Maximization Under Full-bandit Feedback

Abstract

We investigate the problem of combinatorial multi-armed bandits with stochastic submodular (in expectation) rewards and full-bandit feedback, where no extra information other than the reward of selected action at each time step \( t \) is observed. We propose a simple algorithm, Explore-Then-Commit Greedy (ETCG) and prove that it achieves a \((1 - 1/e)\)-regret upper bound of \( O(n^{1/3}k^{4/3}T^{2/3} \log(T)^{1/2}) \) for a horizon \( T \), number of base elements \( n \), and cardinality constraint \( k \). We also show in experiments with synthetic and real-world data that the ETCG empirically outperforms other full-bandit methods.

1 INTRODUCTION

The stochastic multi-armed bandit (MAB) problem was first introduced by Robbins \[1952\]. It formalizes challenging sequential decision problems faced by many organizations, including inventory selection, scheduling, work assignments and team formation, multi-market ad campaigns, product recommendation, crowd-sourcing, and investing. The decision maker selects an arm and observes reward that comes from an unknown distribution at each round. The goal of the decision maker is to maximize expected cumulative reward over all rounds. The solution to classical MAB problem demonstrates the trade-off between exploration and exploitation: should the agent try the arm that has not been tried many times so far (exploration) or should stick with the arm that performed well based on previous observations (exploitation)?

The combinatorial multi-armed bandit (CMAB) problem is an extension to MAB problem. In this setting, the decision maker selects a super arm composed of base arms at each round, and observes a reward corresponding to selected super arm. If the decision maker only learns the aggregated reward for the selected super arm, that feedback is referred to as full-bandit. Otherwise, if the decision maker learns additional information (e.g., individual rewards of the base arms), the feedback is referred to as semi-bandit. Furthermore, there are two common formalizations depending on the assumed nature of environments: the stochastic setting and the adversarial setting.

In the adversarial setting, the reward sequence is generated by an unrestricted adversary, potentially based on the history of decision maker’s actions \[Auer et al., 2003\]. In the stochastic environment, the reward of each arm is drawn independently from a fixed distribution \[Auer et al., 2002\]. For many bandit problems, the stochastic setting is a special case of the adversarial setting. For those problems, algorithms designed for the adversarial setting maintain the theoretical performance guarantees when applied to problems in the stochastic setting, though typically they empirically underperform algorithms specifically designed for the stochastic setting \[Lattimore and Szepesvári, 2020\]. Moreover, the strategies designed for the stochastic setting may have simpler designs and be computationally more efficient. Thus, developing efficient algorithms specializing in stochastic setting is important. Furthermore, as we will later describe, the stochastic setting we consider in this paper is not a special case of the adversarial settings that has been studied in the literature. Specifically, past research in the adversarial setting assume the reward function has extra properties that, when specialized to the stochastic setting, are overly restrictive.

When the reward depends non-linearly on the ground set, additional challenges have been added to develop efficient algorithms. For example, opening additional restaurants in a small market may result in diminishing returns due to market saturation. Such diminishing returns can be naturally modeled with the class of submodular set functions. A set function \( f : 2^\Omega \to \mathbb{R} \) defined on a finite ground set \( \Omega \) is said to be submodular if it satisfies diminishing return property: for all \( A \subseteq B \subseteq \Omega \), and \( x \in \Omega \setminus B \), it holds that \( f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B) \).
In this paper, we focus on the problem of combinatorial multi-armed bandits with stochastic submodular (in expectation) rewards and full-bandit feedback. We further assume that the reward function is monotone: a submodular set function \( f : 2^\Omega \rightarrow \mathbb{R} \) is called monotone if for any \( A \subseteq B \subseteq \Omega \) we have \( f(A) \leq f(B) \).

### 1.1 MOTIVATING EXAMPLES

**Influence Maximization**  Consider a case of social network where a company developed an application and wants to market it through the network. The best way to do this is selecting a set of highly influential users and hope they can love the application and recommend their friends to use it. Influence maximization is a problem of finding a small subset (seed set) in a network that can achieve maximum influence. This subset selection problem in social networks is commonly modeled as an offline submodular optimization problem [Domingos and Richardson 2001; Kempe et al. 2003; Chen et al. 2010]. Algorithms and heuristics for solving this problem often assume knowledge of the network and diffusion model. A recent line of research has generalized the problem as a multi-armed bandit problem (with extra feedback) where the knowledge of the network and diffusion model is not required [Lei et al. 2015; Wen et al. 2017; Yaswani et al. 2017; Li et al. 2020; Perrault et al. 2020].

**Recommender Systems**  When recommending bundles of items, such as movies, news articles, or consumer products, considering the estimated individual item rankings alone may be suboptimal. The system should recommend diversified items to maximize the coverage of information that users are interested in, in order to get as much positive feedback as possible. This is motivated by recommending items with redundant information leads to diminishing returns on utility. This problem of sequentially recommending sets of items to users has been studied through the framework of contextual submodular combinatorial bandits [Qin and Zhu 2013; Yue and Guestrin 2011; Takemori et al. 2020].

**Crowdsourcing and Crowdsensing**  Crowdsourcing involves batches of simple tasks being sequentially assigned to workers with unknown quality and speed. For example, workers may be recruited to manually label images in a database. Crowdsensing involves sequentially collecting data from large numbers of users in different locations. For instance, mobile phone accelerometer data can help identify potholes in city roads. Instances of these problems often involve sequential decision making of assigning/selecting subsets of workers/users with unknown qualities and under a budget. There is a line of research on this topic using the framework of combinatorial multi-armed bandits with submodular rewards [Zhang and van der Schaar 2012; Nushi et al. 2016; Song and Jin 2021].

### 1.2 OUR CONTRIBUTION

The main contribution of this paper can be summarized as follows:

- We propose Explore-then-Commit Greedy (ETCG), the first algorithm designed for stochastic CMAB problems with a submodular reward function (in expectation) and full-bandit feedback. It is procedurally simple and has low storage and per-round computational complexity.
- We prove that ETCG achieves \( O(n^{1/2}k^{1/2}T^{1/2} \log(T)^{1/2}) \) expected cumulative \( (1 - 1/e) \)-regret.
- We show ETCG outperforms other full-bandit methods on experiments with synthetic and real-world data.

### 1.3 RELATED WORK

We now briefly discuss related works from several research topics that overlap in multiple aspects with the problem we study. Table I lists related works and enumerates aspects of the problem setup including properties of the reward function, the feedback model, and regret type. We let \( n \) denote the number of base arms, \( k \) the maximum cardinality, and \( T \) the time horizon.

**Adversarial**  The closest related works are those for adversarial CMAB with submodular rewards, full-bandit feedback, and cumulative regret. In the adversarial setting, the environment chooses a sequence of monotone and submodular functions \( \{f_1, \ldots, f_T\} \). This is incompatible with our setting, since we only require the set function \( f_t \) to be monotone and submodular in expectation. Regret in the adversarial setting is also different—the decision-maker competes against a maximizing action over the sum of the sequence, \((1 - 1/e)\max_{a \in A} \sum_{t=1}^{T} f_t(a)\). We nonetheless consider the following regret bounds to be relevant benchmarks for the stochastic setting.

Streeter and Golovin [2008] proposed an algorithm that achieves \( O(k^2 (n \log n)^{1/3} T^{2/3} \log T)^2 \) \((1 - 1/e)\)-regret.

The method we propose, ETCG, will have a lower regret bound, by a factor of \( k^2 \) (ignoring log terms). Golovin et al. [2014] later proposed an algorithm that achieves \( O(k^{2/3} n^{2/3} (\log n)^{1/3} T^{2/3}) \) \((1 - 1/e)\)-regret. Recently, Niazadeh et al. [2021] proposed a new algorithm for the adversarial setting that achieves \( O(kn^{2/3} (\log n)^{1/3} T^{2/3}) \) \((1 - 1/e)\)-regret. The method we will propose, ETCG, will have a much lower regret bound than those two, by a factor of \( n^{1/3} \) for both (ignoring log terms), for problems where there are many base arms relative to the cardinality constraint (i.e. \( n \gg k \)), such as social influence maximization.

**Semi-bandit**  To our knowledge, all prior works on stochastic, combinatorial multi-armed bandits with submodular rewards assume semi-bandit feedback. In this setting,
Instead of evaluating algorithms in terms of cumulative regret, the decision maker may seek to only evaluate the regret of the action chosen at time $T$, allowing for more aggressive exploration, or to select an action within a pre-set level of confidence as quickly as possible. Several works have investigated this “pure exploration” setting with semi-bandit feedback [Chen et al., 2016; Mokhtari et al., 2018; Merlis and Mannor, 2019; Jourdan et al., 2021] and recently for full-bandit feedback [Du et al., 2021].

Non-submodular There are prior works for combinatorial MAB with stochastic rewards and full-bandit feedback, but the classes of the reward functions considered do not include submodular functions. In particular, there are works for linear reward functions [Dani et al., 2008; Rejwan and Mansour, 2020] and Lipschitz reward functions [Agarwal et al., 2021a,b]. For those classes of reward functions considered by [Rejwan and Mansour, 2020; Agarwal et al., 2021a,b], the optimal action (best set of $k$ arms) is to use the $k$ individually best arms; that property does not hold for submodular rewards.

### 2 PROBLEM STATEMENT

In this section, we will formally present the problem we will study. We consider sequential decision-making problems with a fixed time horizon $T$, where at each time step $t$, the learner selects a subset (action) $S_t \subseteq \Omega$ with cardinality at most $k$. Let $\Omega$ be the ground set of base arms, and let $n = |\Omega|$ denote the number of arms. We will use the terminologies subset and action interchangeably throughout the paper. Let $S = \{S \mid S \subseteq \Omega \text{ and } |S| \leq k\}$ denote the set of all allowed subsets at any time step. After the subset $S_t$ is selected, the learner receives reward $f_t(S_t)$. We assume the reward $f_t$ is stochastic, bounded in $[0, 1]$, and i.i.d. conditioned on a given subset. Define the expected reward function as $f(S) = \mathbb{E}[f_t(S)]$. We assume $f(S)$ to be submodular and monotonically non-decreasing. The goal of the learner is to maximize the cumulative reward $\sum_{t=1}^{T} f_t(S_t)$. To measure the performance of the algorithm, one common

### Continuous Submodular

There is an active area of research in (continuous) optimization for functions exhibiting diminishing returns properties analogous to (discrete) optimization of submodular set functions. Several methods have been proposed in the bandit setting, varying in the environment (adversarial/stochastic) and feedback model [Chen et al., 2018, 2020; Zhang et al., 2019; Hassani et al., 2017; Mokhtari et al., 2020; Hassani et al., 2020; Zhang et al., 2020]. Extensions of these methods to problems with discrete actions has been proposed, but require additional assumptions, semi-bandit feedback, or expensive sampling routines to estimate gradients.

#### Pure Exploration

Instead of evaluating algorithms in terms of cumulative regret, the decision maker may seek to only evaluate the regret of the action chosen at time $T$, allowing for more aggressive exploration, or to select an

<table>
<thead>
<tr>
<th>Reward</th>
<th>Feedback</th>
<th>Regret</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submodular</td>
<td>Stochastic</td>
<td>Full-Bandit</td>
</tr>
<tr>
<td>Streeter and Golovin [2008]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Golovin et al. [2014]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Niazadeh et al. [2021]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Agarwal et al. [2021b]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Agarwal et al. [2021a]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Chen et al. [2018]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Du et al. [2021]</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ETCG (ours)</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1: Table of select related works, enumerating which problem and performance aspects are shared with our proposed ETCG. The notation $\mathcal{O}(\cdot)$ drops log terms. [Chen et al., 2018] require additional smoothness properties of $f$ and the dependence on $k$ and $n$ is unknown.
metric is to compare the learner to an agent with access to a value oracle for \( f \). Let \( S^* = \arg \max_{S : |S| \leq k} f(S) \) denote the optimal solution. Maximizing a monotone submodular set function under a cardinality constraint is NP-hard even with a value oracle. The best achievable approximation ratio with a polynomial time algorithm is \( 1 - 1/e \) [Nemhauser et al., 1978]. Thus, we compare the learner’s cumulative reward to \( (1-1/e)Tf(S^*) \) and we denote the difference as the \((1-1/e)\)-regret \( R_{1/e,T} \):

\[
R_{1/e,T} := (1-1/e)Tf(S^*) - \sum_{t=1}^{T} f_t(S_t).
\]

Note that the \((1-1/e)\)-regret \( R_{1/e,T} \) is random, depending on the rewards and subsets chosen. In designing an algorithm, we will focus on minimizing the expected cumulative \((1-1/e)\)-regret

\[
\mathbb{E}[R_{1/e,T}] = (1-1/e)Tf(S^*) - \mathbb{E}\left[ \sum_{t=1}^{T} f_t(S_t) \right],
\]

where the expectation is over both the environment the sequence of actions. For ease of notation, we write \( R_T \) for \( R_{1/e,T} \) throughout this paper.

**Remark 2.1.** For the experiments in Section 5 we will not know \( S^* \) and so will not be able to compute the \((1-1/e)\) regret \( \mathbb{E}[R_{1/e,T}] \). We will instead compute an upper bound. We will compare ETCG and baselines against \( T \) times the expected value \( f(S_{\text{grd}}) \) of the solution \( S_{\text{grd}} \) returned from an offline (greedy) approximation algorithm [Nemhauser et al., 1978]. Since \( f(S_{\text{grd}}) \geq (1-1/e)f(S^*) \), the expected cumulative regret with respect to \( S_{\text{grd}} \) upper-bounds \( \mathbb{E}[R_{1/e,T}] \). When the inequality is strict, \( f(S_{\text{grd}}) > (1-1/e)f(S^*) \), it is possible the expected cumulative regret \( \mathbb{E}[R_{1/e,T}] \) is sub-linear in the horizon \( T \) while the expected cumulative regret with respect to \( S_{\text{grd}} \) is linear in the horizon \( T \).

## 3 ETCG Algorithm

In this section, we present our proposed algorithm, Explore-Then-Commit Greedy (ETCG). The pseudo code for ETCG is presented in Algorithm 1. Our algorithm adds base arms to a super arm (subset of base arms) over time greedily until the cardinality constraint is satisfied and then exploits that super arm. Let \( S^{(i)} \) denote the super arm when we have selected \( i < k \) base arms. Our procedure begins with the empty set, \( S^{(0)} = \emptyset \). After fixing a subset \( S^{(i-1)} \) with \( i-1 \) arms, our procedure explores base arms to add to \( S^{(i-1)} \) for an interval of time we refer to as phase \( i \). Our procedure repeats this process until the cardinality constraint \( k \) is satisfied.

Let \( T_i \) denote the time step when phase \( i \) finishes, for \( i \in \{1, \cdots, k\} \). For notational consistency, we also denote \( T_0 = 0 \) and \( T_{k+1} = T \). Let \( \bar{f}_t(S) \) denote the empirical mean reward of set \( S \) up to and including time \( t \). Let \( \bar{S}_i := \{S^{(i-1)} \cup \{a\} : a \in \Omega \setminus S^{(i-1)} \} \) denote the set of actions considered during phase \( i \). Each action consists of the super arm \( S^{(i-1)} \) decided during the last phase and an additional base arm. Each action \( S \in \bar{S}_i \) will be played the same number of times; let \( m \) denote that number. The choice of \( m \) will be optimized later to minimize regret. At the end of phase \( i \in \{1, \cdots, k\} \), ETCG will select the action that has the largest empirical mean,

\[
a_i = \arg \max_{a \in \Omega \setminus S^{(i-1)}} \bar{f}_{T_i}(S^{(i-1)} \cup \{a\}),
\]

and include it in the super arm \( S^{(i)} = S^{(i-1)} \cup \{a_i\} \). During the final phase, the algorithm exploits \( S^{(k)} \); it plays the same action \( S_i = S^{(k)} \) for \( t \in \{T_k + 1, \cdots, T\} \).

We note that for the special setting of deterministic rewards, the choice \( a_i = \arg \max_{a \in \Omega \setminus S^{(i-1)}} \bar{f}_{T_i}(S^{(i-1)} \cup \{a\}) \) corresponds to the classic offline greedy approximation algorithm proposed by [Nemhauser et al., 1978]. When the rewards are stochastic, the actions selected by ETCG may differ from those that the greedy algorithm [Nemhauser et al., 1978] would choose using a value oracle for the set function \( f \) of expected rewards.

ETCG has low storage complexity and per-round time-complexity. During exploitation, for \( i \in \{T_k + 1, \cdots, T_{k+1}\} \), ETCG only needs to store the indices of the \( k \) base arms and does not need any computation. During exploration, for \( t \in \{1, \cdots, T_k\} \), ETCG just needs to update the empirical mean for the current action at time \( t \) and store the highest empirical mean so far in the current phase \( i \) and its associated base arm \( a \in \Omega \setminus S^{(i)} \). Thus, ETCG has \( \mathcal{O}(k) \) storage complexity and \( \mathcal{O}(1) \) per-round time complexity. For comparison, the algorithm proposed by [Streeter and

### Algorithm 1 Explore-then-Commit Greedy (ETCG)

**Input:** set of base arms \( \Omega \), horizon \( T \), cardinality constraint \( k \)

**Initialize** \( S^{(0)} \leftarrow \emptyset \), \( n \leftarrow |\Omega| \)

**Initialize** \( m \leftarrow \left\lceil \frac{T \log(T)}{n+2nk^2} \right\rceil \)

**for** phase \( i \in \{1, \cdots, k\} \) **do**

**for** arm \( a \in \Omega \setminus S^{(i-1)} \) **do**

Play \( S^{(i-1)} \cup \{a\} \) \( m \) times

Calculate the empirical mean \( \bar{f}(S^{(i-1)} \cup \{a\}) \)

end for

**for** remaining time **do**

Play action \( S^{(k)} \)

end for

---

[Streeter and] 4
Although the Eorem 4.1. For the sequential decision making problem defined in Section 2 with T ≥ n(k + 1), the expected cumulative (1 − 1/e)-regret of ETCG is at most O(n³k²T²/3 log(T)²).

The detailed proof is in Appendix A. We next briefly walk through the proof, highlighting some key steps.

Since for each phase i, we play each action S(i−1) \cup \{a\} ∈ S_i exactly m times, we consider the equal-sized confidence radii \( \text{rad} := \sqrt{2 \log(T)/m} \) for all the actions S(i−1) \cup \{a\} ∈ S_i at the end of phase i. Denote the event that the empirical means of actions played in phase i are concentrated about their statistical means as

\[
\mathcal{E}_i := \bigcap_{S \cup \{a\} \in S_i} \left\{ |f(S \cup \{a\}) - f(S \cup \{a\})| < \text{rad} \right\}. (4)
\]

Then we define the clean event \( \mathcal{E} \) to be the event that the empirical means of all actions played up to and including phase k are within rad of their corresponding statistical means:

\[
\mathcal{E} := \mathcal{E}_1 \cap \cdots \cap \mathcal{E}_k. (5)
\]

Although the \( \mathcal{E}_i \)'s are not independent, by conditioning on the sequence of selected subsets \{S(0), S(1), \ldots, S(k)\} and using the Hoeffding bound, we show \( \mathcal{E} \) happens with high probability. We then use the concentration of empirical means and properties of submodular set functions to show the following important lemma.

Lemma 4.2. Under the clean event \( \mathcal{E} \), for all \( i \in \{1, 2, \ldots, k\} \),

\[
f(S(i)) - f(S(i−1)) \geq \frac{1}{k} \left[ f(S^*) - f(S(i−1)) \right] - 2\text{rad.}
\]

This lemma (Lemma A.3 in the full proof) identifies a lower bound of the expected marginal gain \( f(S(i)) - f(S(i−1)) \) of the empirically best action \( S(i) \) at the end of phase \( i \). The sequence of subsets \{S(0), S(1), \ldots, S(k)\} that ETCG picks does not necessarily match the sequence chosen by the offline greedy approximation [Nemhauser et al., 1978] using a value oracle for the expected reward function \( f \). Even though ETCG may select a different sequence, Lemma 4.2 ensures the expected marginal gain is not too small. As a corollary of Lemma 4.2 using properties of submodular set functions and unraveling the recursion induced by Lemma 4.2, we can lower bound the expected value of ETCG’s chosen set \( S(k) \) of size \( k \), which is used for exploitation in phase \( k + 1 \):

\[
f(S(k)) \geq (1 - \frac{1}{e})f(S^*) - 2\text{rad}. (6)
\]

This corollary appears as Corollary A.4 in the full proof in Appendix A.1. Using Corollary 4.3, we can break up the \( \mathbb{E}[R(T)] \) into two parts, one part for the first \( k \) phases and one part for the exploitation phase,

\[
\mathbb{E}[R(T)|\mathcal{E}] = (1 - \frac{1}{e})Tf(S^*) - \sum_{i=1}^{T} \mathbb{E}[f_i(S_i)]
\]

\[
= \sum_{i=1}^{T} \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f_i(S_i)] \right]
\]

\[
= \sum_{i=1}^{k} \sum_{t=Ti−1+1}^{Ti} \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_i)] \right]
\]

\[
= \sum_{i=1}^{k} \sum_{t=Ti−1+1}^{Ti} \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_i)] \right]. (7)
\]

Recall that in phase \( i \), each of the \( n_i \) actions in \( S_i \) is played exactly \( m \) times, meaning \( T_i − Ti−1 = m(n_i + 1) \). For each action \( S_i \) played during phase \( i \), that is for \( t \in \{Ti−1+1, \cdots, Ti\} \), since \( S(i−1) \subset S_i \), by monotonicity of the expected reward function \( f \) we have \( f(S(i−1)) \leq f(S_i) \). Thus we can upper bound the expected regret \( \mathbb{E}[\mathcal{R}(T)|\mathcal{E}] \) incurred during the first \( k \) phases (first term of (7)) as

\[
\mathbb{E}[\mathcal{R}(T)|\mathcal{E}] \leq \sum_{i=1}^{k} \sum_{t=Ti−1+1}^{Ti} \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_i)] \right]
\]

\[
\leq \sum_{i=1}^{k} m(n_i + 1) \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S(i−1))] \right]
\]

\[
\leq mn \sum_{i=1}^{k} \left[ (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S(i−1))] \right]. (8)
\]
We next evaluate our proposed algorithm ETCG on both synthetic data and real world data.

We can further upper bound (8) as

$$\sum_{i=1}^{k} \left(1 - \frac{1}{e}\right) f(S^*) - \mathbb{E}[f(S^{(i-1)})] \leq \sum_{i=1}^{k} \left(f(S^*) - \mathbb{E}[f(S^{(i)})]\right) \leq k \sum_{i=1}^{k} \left(\mathbb{E}[f(S^{(i)})] - \mathbb{E}[f(S^{(i-1)})] + 2\text{rad}\right)$$

$$= k(\mathbb{E}[f(S^{(k)})] - \mathbb{E}[f(S^{(0)})] + 2\text{rad}) \leq k (1 + 2k\text{rad}),$$

where (7) follows by applying Lemma 4.2 and taking expectation, (10) follows by simplifying a telescoping sum, and (11) by $\mathbb{E}[f(S^{(k)})] \leq 1$ and $\mathbb{E}[f(S^{(0)})] = 0$.

We can upper bound the expected regret $\mathbb{E}[\mathcal{R}^\ast(T)|\mathcal{E}]$ incurred during the exploitation phase (phase $k+1$; second term of (7)) by applying Corollary 4.3 as

$$\sum_{t=\tau_k+1}^{T} \left(1 - \frac{1}{e}\right) f(S^*) - \mathbb{E}[f(S^{(k)})] \leq \sum_{t=\tau_k+1}^{T} 2\text{rad} \leq 2k\text{Trad}. \quad (12)$$

Combining the upper bounds (11) and (12) and then optimizing over the number of times $m$ each action is sampled during exploration, we get

$$\mathbb{E}[\mathcal{R}^\ast(T)|\mathcal{E}] \leq 4n^\frac{3}{2}k(\sqrt{T} \sqrt[4]{\log(T)})^\frac{1}{2} (1 + 2k \sqrt[4]{2 \log(T)})^\frac{1}{2} = O(n^\frac{3}{2}k^\frac{3}{4}T^\frac{1}{2} \log(T)^{\frac{1}{4}}). \quad (13)$$

We then show that because the clean event $\mathcal{E}$ happens with high probability, $\mathbb{E}[\mathcal{R}(T)]$ also satisfies (13), completing the proof.

**Lower bounds:** For the setting we explore in this paper, with stochastic CMAB with submodular expected rewards and full-bandit feedback, it remains an open question if $O(T^{1/2})$ expected cumulative $(1 - 1/e)$-regret is possible (ignoring $n$ and $k$ dependence). For the special sub-class of linear reward functions, $\Omega(T^{1/2})$ is known [Dani et al., 2008].

## 5 EXPERIMENTS

We next evaluate our proposed algorithm ETCG on both synthetic data and real world data.

For the experiments, instead of $(1 - 1/e)$ regret Equation (11), which requires knowing $S^*$, we compute the cumulative rewards achieved by ETCG and baselines against $T f(S^{\text{grad}})$, where $S^{\text{grad}}$ is the solution returned by the offline $(1 - 1/e)$-approximation algorithm proposed by [Nemhauser et al., 1978]. Recall from Remark 2.1 that $T f(S^{\text{grad}}) \geq (1 - 1/e)T f(S^*)$, so $T f(S^{\text{grad}})$ is a more challenging reference value.

### 5.1 BASELINE METHODS

We use four algorithms designed for CMAB with full-bandit feedback as baselines.

- **Online Greedy with opaque feedback model (OGo)** [Streeter and Golovin, 2008] This algorithm is designed for the adversarial setting with submodular rewards. The adversary model is oblivious, meaning the sequence of monotone submodular reward functions is fixed in advance. OGo utilizes $k$ subroutines of randomized weighted majority algorithms [Littlestone and Warmuth, 1994] to select actions, where $k$ is the cardinality constraint. At each time step, the algorithm explores with probability $\gamma$ and exploits with probability $1 - \gamma$. During exploration, it randomly picks a randomized weighted majority subroutine to select a base arm to explore. OGo has a $O(T^{2/3})$ theoretical guarantee for the adversarial setting. We refer to our detailed implementation and parameter selection in Appendix C.

- **CMAB-SM** [Agarwal et al., 2021a] This algorithm assumes the expected reward functions are Lipschitz continuous functions of individual arm rewards. The algorithm divides all $n$ base arms into groups, sorts arms within each group, and then merges groups one by one to obtain the best $k$ arms. CMAB-SM has a $O(T^{2/3})$ theoretical guarantee.

- **DART** [Agarwal et al., 2021b] DART is a successive accept-reject style algorithm designed for Lipschitz reward functions that have an additional property related to the marginal gains of the base arms. DART has a $O(T^{1/2})$ theoretical guarantee.

### 5.2 EXPERIMENTS WITH SYNTHETIC DATA

We begin with experiments with deterministic reward functions from two special cases of submodular set functions—mean (linear) functions ($f(S) = \sum_{a \in S} f(\{a\})/k$) and max functions ($f(S) = \max_{a \in S} f(\{a\})$).

#### 5.2.1 Experiment Details

We use $n = 10$ base arms. The cardinality constraint is $k = 2$. We generate individual arm rewards $\{f(\{a\})\}_{a \in \Omega}$
we next run experiments for the application of social network influence maximization over a portion of the Facebook network graph. While there are prior works proposing algorithms for influence maximization bandit problems, the state of the art (e.g., [Wen et al., 2017]) presumes knowledge of the diffusion model (such as independent cascade) and, more importantly, extensive semi-bandit feedback on individual diffusions, such as which specific nodes became active or along which edges successful infections occurred, in order to estimate diffusion parameters. For social networks with user privacy, this information is not available.

5.3 EXPERIMENTS WITH REAL WORLD DATA

We next run experiments for the application of social network influence maximization over a portion of the Facebook network graph. While there are prior works proposing algorithms for influence maximization bandit problems, the state of the art (e.g., [Wen et al., 2017]) presumes knowledge of the diffusion model (such as independent cascade) and, more importantly, extensive semi-bandit feedback on individual diffusions, such as which specific nodes became active or along which edges successful infections occurred, in order to estimate diffusion parameters. For social networks with user privacy, this information is not available.

5.3.1 Data Set Description and Experiment Details

We next conduct experiments on an influence maximization problem using a portion of the Facebook network [Leskovec and Mcauley, 2012]. To facilitate running multiple experiments for different horizons, we used the community detection method proposed by Blondel et al. [2008] to detect a community with 534 nodes and 8158 edges. The diffusion process is simulated using the independent cascade model [Kempe et al., 2003], where in each discrete step, an active node (that was inactive at the previous time step) independently attempts to infect each of its inactive neighbors. We used uniform infection probabilities (0.1 for each edge). For each horizon $T \in \{2 \times 10^4, 4 \times 10^4, \ldots, 10^6\}$, we tested each method ten times.
ETCG significantly outperforms OG. Over the horizons tested, OG’s cumulative regret (averaged over ten runs) appears to grow linear with $T$. We saw in Section 5.2 that even for much simpler reward functions and with few arms $n$ and small cardinality $k$, OG performed poorly.

ETCG outperforms CMAB-SM for all time horizons and cardinalities, with significant gaps between ETCG and CMAB-SM for smaller $k$. From Figures 2a to 2c, CMAB-SM’s performance appears fairly stable across increasing cardinalities (though note limits of y-axes differ) while ETCG regret curve approaches that of CMAB appears to grow (relative to others). For a fixed horizon $T$, increasing $k$ means more phases, which (for this problem with large $n$) means more time exploring overall but less time in any one phase, so the arms selected may not be as good. In Figure 2d with $k = 4$, for instance, each of the four phases of ETCG’s exploration are distinct, and exploitation begins around $t = 20000$. In Figure 2f with $k = 16$, however, each of the sixteen phases of ETCG’s exploration are shorter and exploitation begins around $t = 35000$.

ETCG and DART have similar performance for small time horizons. However, DART’s cumulative regret curve has a steep jump which make the performance significantly worse. We attribute these jumps to the exponential episode lengths considered in DART with number of episodes $\lfloor \log_2(KT/N \log(NT)) \rfloor$. This creates a non-smooth behavior in the regret growth of the DART algorithm.

Figure 2d, Figure 2e and Figure 2f shows instantaneous rewards over a horizon $T = 10^5$ for corresponding cardinality constraints. Again curves for all methods are smoothed with a moving average with window size 100. Clearly we can see that ETCG has the fastest convergence over all methods. On the other hand, the set of size $k$ that is chosen by ETCG is worse that those of CMAB-SM and DART, since the latter two methods requires longer time to explore. We can also attribute the worse performance when $k$ gets larger to the larger $k$ term in the regret bound.
References


Baosheng Yu, Meng Fang, and Dacheng Tao. Linear submodular bandits with a knapsack constraint. In Thirtieth AAAI Conference on Artificial Intelligence, 2016.


A PROOFS

We will separate the proof of Theorem 4.1 into two cases. The first case is for when the clean event $\mathcal{E}$ [5] happens, which we will show in Lemma A.2 happens with high probability. Under the clean event, we will prove important preliminary results, namely Lemma A.3 and Corollary A.4. These will establish that even though ETCG, using random rewards, may pick a different sequence of subsets than an offline greedy algorithm [Nemhauser et al., 1978] using a value oracle for the expected reward function $f$, ETCG’s chosen set of size $k$ will nonetheless be near-optimal. The second case is when the complementary event happens, which occurs with low probability.

This proof structure is analogous to the standard MAB proof for explore-then-commit strategies (see for instance, Section 1.2 in (Slivkins, 2019)). However, unlike for standard MAB problems, ETCG makes sequences of decisions during exploration. Furthermore, the combinatorial action space and non-linear reward function make the problem challenging. Even in the special setting of deterministic rewards, the standard MAB problem becomes trivial (finding the largest of $n$ base arms) while maximizing a submodular function with a cardinality constraint is NP-hard [Nemhauser et al., 1978].

A.1 PRELIMINARY

We first introduce some new notations and lemmas that are useful in the analysis. Recall from Section 2 that for an action $S \in \mathcal{S}$, $f_t(S)$ denotes a (random) reward at time $t$, $f(S)$ denotes the expected value for action $S$, and $f_t(S)$ denotes the empirical mean of rewards received from playing action $S$ up to and including time $t$. In the following, we will drop the subscript $t$ from the empirical mean, writing $\bar{f}(S)$ when it is clear from context that action $S$ has been played $m$ times. Also recall that $S^{(i)}$ denotes the set of size $i \in \{1, \ldots, k\}$ chosen after finishing phase $i$, and by the greedy structure of Algorithm 1, $\emptyset = S^{(0)} \subset S^{(1)} \subset \cdots \subset S^{(k)}$. This sequence of subsets that ETCG picks does not necessarily match the sequence chosen by the offline greedy approximation [Nemhauser et al., 1978] using a value oracle for the expected reward function $f$. Even though ETCG may select a different sequence, we will later show in Lemma 4.2 that with high probability, ensures the expected marginal gain is not too small.

Now we define events that are important in our analysis. Recall that $\bar{f}(S^{(i-1)} \cup \{a\})$ is the empirical mean of the $m$ rewards from playing action $S^{(i-1)} \cup \{a\}$ in phase $i$. For each subset $S^{(i-1)} \cup \{a\}$, the $m$ rewards are i.i.d. with mean $f(S^{(i-1)} \cup \{a\})$ and bounded in $[0, 1]$. Thus, we can bound the deviation of the (unbiased) empirical mean $\bar{f}(S^{(i-1)} \cup \{a\})$ from the expected value $f(S^{(i-1)} \cup \{a\})$ for each action in $\mathcal{S}_i$. Specifically, we can use a two-sided Hoeffding bound for bounded variables.

**Lemma A.1** (Hoeffding’s inequality). Let $X_1, \cdots, X_n$ be independent random variables bounded in the interval $[0, 1]$, and let $\bar{X}$ denote their empirical mean. Then we have for any $\epsilon > 0$,\[ \mathbb{P}\left( |\bar{X} - \mathbb{E}[\bar{X}]| \geq \epsilon \right) \leq 2 \exp\left( -2\epsilon^2 \right). \]

We will use Hoeffding’s inequality to bound the probabilities of the empirical means $\bar{f}(S^{(i-1)} \cup \{a\})$ for all actions $S^{(i-1)} \cup \{a\} \in \mathcal{S}_i$ played in phase $i$. By Algorithm 1, each action will be played the same number of times, denoted by $m$, so we consider bounding the probabilities of equal-sized confidence radii $\text{rad} := \sqrt{2 \log(T)/m}$ for all the actions $S^{(i-1)} \cup \{a\} \in \mathcal{S}_i$ played in phase $i$.

We consider the event that the empirical means of all actions played in phase $i$ are concentrated around their statistical means within a radius rad. Denote this event as $\mathcal{E}_i$,

$$\mathcal{E}_i := \bigcap_{S \cup \{a\} \in \mathcal{S}_i} \left\{ |\bar{f}(S \cup \{a\}) - f(S \cup \{a\})| < \text{rad} \right\}. \quad (15)$$

Define the **clean event** $\mathcal{E}$ to be the event that the empirical means of all actions played up to and including phase $k$ are within rad of their corresponding statistical means:

$$\mathcal{E} := \mathcal{E}_1 \cap \cdots \cap \mathcal{E}_k. \quad (16)$$

**Lemma A.2.** The probability of the clean event $\mathcal{E}$ [16] satisfies:

$$\mathbb{P}(\mathcal{E}) \geq 1 - \frac{2nk}{T^4}. \quad (17)$$
Proof. We begin by breaking up the probability of the clean event \( \mathcal{E} \) into conditional probabilities for the events \( \{ \mathcal{E}_i \}_{i=1}^k \) for each phase,

\[
\mathbb{P}(\mathcal{E}) = \mathbb{P}(\mathcal{E}_1 \cap \cdots \cap \mathcal{E}_k) = \prod_{i=1}^k \mathbb{P}(\mathcal{E}_i | \mathcal{E}_1, \ldots, \mathcal{E}_{i-1}).
\]  

(17)

Recall that \( \mathcal{E}_i \), defined in (15), is the event where the empirical means of all actions played in phase \( i \) were concentrated around their statistical means. Which actions are available in phase \( i \), namely \( \{ S^{(i-1)} \cup \{ a \} \}_{a \in \Omega \setminus S^{(i-1)}} \), depends on the action \( S^{(i-1)} \) from the previous phase that had the highest empirical mean, which in turn is related to \( \mathcal{E}_{i-1} \). Although we cannot directly evaluate (17), by conditioning on \( S^{(i-1)} \) we will be able to obtain a bound on (17).

\[
\mathbb{P}(\mathcal{E}_i | \mathcal{E}_1, \ldots, \mathcal{E}_{i-1}) = \sum_{S \in \{ S \mid S \subseteq \Omega, |S|=i-1 \}} \mathbb{P}(S^{(i-1)} = S, \mathcal{E}_i | \mathcal{E}_1, \ldots, \mathcal{E}_{i-1})
\]

(law of total probability)

\[
= \sum_{S \in \{ S \mid S \subseteq \Omega, |S|=i-1 \}} \mathbb{P}(S^{(i-1)} = S | \mathcal{E}_1, \ldots, \mathcal{E}_{i-1}) \times \mathbb{P}(\mathcal{E}_i | S^{(i-1)} = S, \mathcal{E}_1, \ldots, \mathcal{E}_{i-1})
\]

\[
= \sum_{S \in \{ S \mid S \subseteq \Omega, |S|=i-1 \}} \mathbb{P}(S^{(i-1)} = S | \mathcal{E}_1, \ldots, \mathcal{E}_{i-1}) \times \mathbb{P}(\mathcal{E}_i | S^{(i-1)} = S),
\]  

(18)

where (18) follows from rewards in phase \( i \) being conditionally independent of rewards from other phases, given the corresponding actions played during phase \( i \).

We now focus on bounding \( \mathbb{P}(\mathcal{E}_i | S^{(i-1)} = S) \). By conditioning on the set chosen in the previous phase, \( S^{(i-1)} = S \), we know all the actions that will be played in the current phase \( i \), \( \{ S^{(i-1)} \cup \{ a \} \}_{a \in \Omega \setminus S^{(i-1)}} \). The rewards of all the actions are bounded in \([0, 1]\) and are conditionally independent (given the corresponding action).

Apply Lemma A.1 to the empirical mean \( \bar{f}(S^{(i-1)} \cup \{ a \}) \) of \( m \) rewards for action \( S^{(i-1)} \cup \{ a \} \) and choosing \( \epsilon = \text{rad} = \sqrt{2 \log(T)/m} \) gives

\[
\mathbb{P} \left[ |\bar{f}(S^{(i-1)} \cup \{ a \}) - f(S^{(i-1)} \cup \{ a \})| \geq \text{rad} \right] \leq 2 \exp (-2m\text{rad}^2) \\
\leq 2 \exp (-2m(2\log(T)/m)) \\
= 2 \exp (-4 \log(T)) \\
= \frac{2}{T^4}.
\]

Thus, for any individual action \( S^{(i-1)} \cup \{ a \} \in S_i \), we can bound the probability that its sample mean \( \bar{f}(S^{(i-1)} \cup \{ a \}) \) is within a specified confidence radius (complementary of the event above) as

\[
\mathbb{P} \left[ |\bar{f}(S^{(i-1)} \cup \{ a \}) - f(S^{(i-1)} \cup \{ a \})| < \text{rad} \right] = 1 - \mathbb{P} \left[ |\bar{f}(S^{(i-1)} \cup \{ a \}) - f(S^{(i-1)} \cup \{ a \})| \geq \text{rad} \right] \\
\geq 1 - \frac{2}{T^4}.
\]  

(19)

We can then use (19) to bound \( \mathbb{P}(\mathcal{E}_i | S^{(i-1)} = S) \) for any set \( S \subseteq \Omega \) of \( i - 1 \) arms.
\[ P(E_i | S^{(i-1)} = S) = P \left[ \bigcap_{a \in \Omega \setminus S^{(i-1)}} \left\{ \left| \hat{f}(S^{(i-1)} \cup \{a\}) - f(S^{(i-1)} \cup \{a\}) \right| < \text{rad} \right\} | S^{(i-1)} = S \right] \quad \text{(definition of } E_i) \]

\[ = \prod_{a \in \Omega \setminus S^{(i-1)}} P \left[ \left\{ \left| \hat{f}(S^{(i-1)} \cup \{a\}) - f(S^{(i-1)} \cup \{a\}) \right| < \text{rad} \right\} | S^{(i-1)} = S \right] \]

(rewards are independent conditioned on actions)

\[ \geq \left( 1 - \frac{2}{T^4} \right)^{|S^{(i-1)} \setminus \Omega|} \quad \text{(using (19))} \]

\[ = (1 - \frac{2}{T^4})^{n-i+1} \]

\[ \geq (1 - \frac{2}{T^4})^n. \quad \text{(20)} \]

Using (18) and (20), we are now ready to lower bound the probability of a clean event.

\[ P(E) = P(E_1 \cap \cdots \cap E_k) \]

\[ = \prod_{i=1}^{k} P(E_i | E_1, \ldots, E_{i-1}) \]

\[ = \prod_{i=1}^{k} \sum_{S \in \{S' | S' \subseteq \Omega, |S'|=i-1\}} P(S^{(i-1)} = S | E_1, \ldots, E_{i-1}) \times P(E_i | S^{(i-1)} = S) \quad \text{(using (18))} \]

\[ \geq \prod_{i=1}^{k} \sum_{S \in \{S' | S' \subseteq \Omega, |S'|=i-1\}} P(S^{(i-1)} = S | E_1, \ldots, E_{i-1}) \times \left( 1 - \frac{2}{T^4} \right)^n \quad \text{(using (20))} \]

\[ = \prod_{i=1}^{k} \left( 1 - \frac{2}{T^4} \right)^n \sum_{S \in \{S' | S' \subseteq \Omega, |S'|=i-1\}} P(S^{(i-1)} = S | E_1, \ldots, E_{i-1}) \]

\[ = \prod_{i=1}^{k} \left( 1 - \frac{2}{T^4} \right)^n \]

\[ = \left( 1 - \frac{2}{T^4} \right)^{nk} \]

\[ \geq 1 - \frac{2nk}{T^4}. \quad \text{(Bernoulli’s inequality)} \]

This concludes the proof for Lemma A.2.

In Lemma A.2, we showed that the clean event \( E \) will happen with high probability. Next, we present a lemma showing that the marginal gain of the action selected at the end of any exploitation phase is large under the condition that the clean event \( E \) happens. Specifically, we next show that under event \( E \), we can bound the gap of the expected rewards with the expected reward of the optimal set, even though the empirically best actions at the end of each phase might not match those chosen by the offline greedy algorithm (which we know is near-optimal and which ETCG would pick if the rewards were deterministic).

**Lemma A.3 (Lemma 4.2 in Section 4).** Under the clean event \( E \), for all \( i \in \{1, \cdots, k\} \),

\[ f(S^{(i)}) - f(S^{(i-1)}) \geq \frac{1}{k} \left[ f(S^*) - f(S^{(i-1)}) \right] - 2\text{rad}. \quad \text{(21)} \]
We next lower bound the expected reward

where (24) follows from a well known bound for submodular functions (see Appendix B.2).

We apply (22) to two specific arms, the empirically best arm

and the statistically best arm. We get

Subtracting \( f(S^{(i)}) \) on both side we have

Recall from Section 2 that \( S^* = \arg \max_{|S| \leq k} f(S) \) denotes the optimal solution in the offline problem. We will next show that the improvements in expectation of the chosen actions from one phase to the next are lower bounded by the gap between the optimal set \( S^* \) of cardinality \( k \) and the set \( S^{(i)} \) chosen in the previous round.

where (24) follows from a well known bound for submodular functions (see Appendix B.2).

Lemma A.3 identifies a lower bound of the expected marginal gain \( f(S^{(i)}) - f(S^{(i-1)}) \) of the empirically best action \( S^{(i)} \) at the end of phase \( i \). The sequence of subsets \( \{S^{(0)}, S^{(1)}, \ldots, S^{(k)}\} \) that ETCG picks does not necessarily match the sequence chosen by the offline greedy approximation [Nemhauser et al., 1978] using a value oracle for the expected reward function \( f \). Even though ETCG may select a different sequence, Lemma A.2 ensures the expected marginal gain is not too small. As a corollary of Lemma A.3 using properties of submodular set functions and unraveling the recursion induced by Lemma A.3 we can lower bound the expected value of ETCG’s chosen set \( S^{(k)} \) of size \( k \), which is used for exploitation in phase \( k+1 \).

**Corollary A.4** (Corollary 4.3 in Section 4). Under the clean event \( \mathcal{E} \),

\[
f(S^{(k)}) \geq (1 - \frac{1}{e}) f(S^*) - 2krad.
\]
Proof. We begin by unraveling the recursion induced by Lemma A.3 and using properties of submodular set functions,

\[ f(S^{(i)}) - f(S^{(i-1)}) \geq \frac{1}{k} \left[ f(S^*) - f(S^{(i-1)}) \right] - 2 \text{rad.} \]  
(copied from (21))

\[ \iff f(S^{(i)}) \geq \frac{1}{k} f(S^*) + (1 - \frac{1}{k}) f(S^{(i-1)}) - 2 \text{rad} \]  
(rearranging)

\[ = \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] + (1 - \frac{1}{k}) f(S^{(i-1)}). \]  
(26)

Applying (26) recursively for \( i = k \),

\[ f(S^{(k)}) \geq \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] + (1 - \frac{1}{k}) f(S^{(k-1)}) \]  
(using (26) for \( i = k \))

\[ \geq \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] + (1 - \frac{1}{k}) \left( \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] + (1 - \frac{1}{k}) f(S^{(k-2)}) \right) \]  
(using (26) for \( i = k - 1 \))

\[ = \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] \sum_{\ell=0}^{k-1} (1 - \frac{1}{k})^\ell + (1 - \frac{1}{k})^2 f(S^{(k-2)}) \]  
(rearranging)

\[ \vdots \]  
(continue recursing until we get to \( S^{(0)} = \emptyset; f(\emptyset) = 0 \))

\[ \geq \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] \sum_{\ell=0}^{k-1} (1 - \frac{1}{k})^\ell \]  
(27)

Simplifying the geometric summation,

\[ \sum_{\ell=0}^{k-1} (1 - \frac{1}{k})^\ell = \frac{1 - (1 - \frac{1}{k})^k}{1 - (1 - \frac{1}{k})} = k \left( 1 - (1 - \frac{1}{k})^k \right). \]

Continuing with (27),

\[ f(S^{(k)}) \geq \left[ \frac{1}{k} f(S^*) - 2 \text{rad} \right] k \left( 1 - (1 - \frac{1}{k})^k \right) \]

\[ = \left( 1 - \left( 1 - \frac{1}{k} \right)^k \right) f(S^*) - 2k \left( 1 - (1 - \frac{1}{k})^k \right) \text{rad} \]

\[ \geq \left( 1 - \left( 1 - \frac{1}{k} \right)^k \right) f(S^*) - 2k \text{rad.} \]  
(simplifying with \( (1 - \frac{1}{k})^k \leq 1 \))

Using the lower bound \( \left( 1 - (1 - \frac{1}{k})^k \right) \geq 1 - \frac{1}{e} \) (see Appendix B.1), we get

\[ f(S^{(k)}) \geq (1 - \frac{1}{e}) f(S^*) - 2k \text{rad.} \]

Rearranging terms we have

\[ (1 - \frac{1}{e}) f(S^*) - f(S^{(k)}) \leq 2k \text{rad.} \]

\[ \square \]
A.2 THEOREM 4.1 PROOF

Now we are ready to prove the main theorem, Theorem 4.1.

Case 1: clean event $\mathcal{E}$ happens

In the first case we analyse the expected regret under the condition that the clean event $\mathcal{E}$ happens. In this section, all expectations will be conditioned on $\mathcal{E}$, but to simplify notation we will write $\mathbb{E}[\cdot]$ instead of $\mathbb{E}[\cdot|\mathcal{E}]$.

First we can break up the expected $(1 - \frac{1}{e})$-regret \[\text{(2)}\] conditioned on $\mathcal{E}$ into two parts, one for the first $k$ phases, and the second for the exploitation phase. Also recall that $f_t(S_t)$ is the random reward for taking action $S_t$, which itself is random, depending on empirical means of actions in earlier phases.

\[
\mathbb{E}[\mathcal{R}(T)] = (1 - \frac{1}{e})Tf(S^*) - \sum_{t=1}^{T} \mathbb{E}[f_t(S_t)]
\]

\[
= (1 - \frac{1}{e})Tf(S^*) - \sum_{t=1}^{T} \mathbb{E}[f_t(S_t)|S_t]
\]

\[
= (1 - \frac{1}{e})Tf(S^*) - \sum_{t=1}^{T} \mathbb{E}[f(S_t)]
\]

\[
= \sum_{t=1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) \quad \text{(using the definition (2))}
\]

\[
= \sum_{t=1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) \quad \text{(law of total expectation)}
\]

\[
= \sum_{i=1}^{k} \sum_{t=T_i-1+1}^{T_i} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) + \sum_{t=T_k+1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) \quad \text{(rearranging)}
\]

\[
= \sum_{i=1}^{k} \sum_{t=T_i-1+1}^{T_i} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) + \sum_{t=T_k+1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(k)})] \right). \tag{28}
\]

Recall that in phase $i$, each of the $n - i + 1$ actions in $\mathcal{S}_t$ is played exactly $m$ times, meaning $T_i - T_{i-1} = m(n - i + 1)$. Since all actions played in phase $i$ include the set $S^{(i-1)}$ (the empirically best set played in phase $i - 1$), in notation $S^{(i-1)} \subseteq S_t$ for $t \in \{T_{i-1} + 1, \cdots, T_i\}$, by monotonicity of the expected reward function $f$, we have $f(S^{(i-1)}) \leq f(S_i)$, for $t \in \{T_{i-1} + 1, \cdots, T_i\}$. Thus, we can simplify the inner summation in the first term of \[\text{(28)}\] as

\[
\sum_{t=T_i-1+1}^{T_i} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S_t)] \right) \leq \sum_{t=T_i-1+1}^{T_i} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(i-1)})] \right)
\]

\[
\quad \text{(monotonicity: } f(S^{(i-1)}) \leq f(S_t))
\]

\[
= m(n - i + 1) \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(i-1)})] \right). \tag{29}
\]

Plugging \[\text{(29)}\] back into \[\text{(28)}\],

\[
\mathbb{E}[\mathcal{R}(T)] \leq \sum_{i=1}^{k} m(n - i + 1) \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(i-1)})] \right) + \sum_{t=T_k+1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(k)})] \right)
\]

\[
\leq mn \sum_{i=1}^{k} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(i-1)})] \right) + \sum_{t=T_k+1}^{T} \left( (1 - \frac{1}{e})f(S^*) - \mathbb{E}[f(S^{(k)})] \right). \tag{30}
\]

Now we upper bound the two terms above using Corollary A.4.
We next check the second derivative,

We want to optimize \( m \) yielding

Setting \( p = 1 \), then

formula, we have

Plugging in the definition of \( \mu = \sqrt{2 \log(T)/m} \), yielding

\[ (1 - \frac{1}{e})f(S^*) - E[f(S^{(k)})] \leq 2k \mu. \] (33)

Apply (32) and (33) to the first and second terms in (30) respectively yields

\[
E[R(T)] \leq mnk \sum_{i=1}^{k} \left( (1 - \frac{1}{e})f(S^*) - E[f(S^{(i-1)})] \right) + \sum_{t=T_k+1}^{T} \left( (1 - \frac{1}{e})f(S^*) - E[f(S^{(k)})] \right) \] (copying (30))

\[ \leq mnk \sum_{i=1}^{k} \left( E[f(S^{(i)})] - E[f(S^{(i-1)})] + 2\mu \right) + \sum_{t=T_k+1}^{T} (2\mu) \] (using 1 - \frac{1}{e} \leq 1 in first sum)

\[ \leq mnk \left( E[f(S^{(k)})] - E[f(S^{(0)})] + 2\mu \right) + \sum_{t=T_k+1}^{T} (2\mu) \] (telescoping sum)

\[ \leq mnk \left( E[f(S^{(k)})] + 2\mu \right) + 2k \mu T \] (rewards are bounded in \([0, 1]\))

Plugging in the definition of \( \mu = \sqrt{2 \log(T)/m} \) and using the bound \( \sqrt{2 \log(T)/m} < \sqrt{2 \log(T)} \) to simplify the formula, we have

\[ E[R(T)] \leq mnk \left( 1 + 2k \sqrt{2 \log(T)/m} \right) + 2kT \sqrt{2 \log(T)/m} \]

\[ \leq mnk \left( 1 + 2k \sqrt{2 \log(T)} \right) + 2kT \sqrt{2 \log(T)/m}. \] (34)

We want to optimize \( m \), the number of times actions are played. Denoting the regret bound (34) as a function of \( m \)

\[ g(m) = mnk \left( 1 + 2k \sqrt{2 \log(T)} \right) + 2kT \sqrt{2 \log(T)/m}, \] (35)

then

\[ g'(m) = nk \left( 1 + 2k \sqrt{2 \log(T)} \right) - kT \sqrt{2 \log(T)m^{-3/2}}. \] (36)

Setting \( g'(m) = 0 \) and solving for \( m \),

\[ m^* = \left( \frac{T \sqrt{2 \log(T)}}{n + 2nk \sqrt{2 \log(T)}} \right)^{2/3}. \] (37)

We next check the second derivative,

\[ g''(m) = \frac{3}{2} kT \sqrt{2 \log(T)}m^{-5/2}. \] (38)
For positive values of \( m \), \( g''(m) > 0 \), thus \( g(m) \) reaches a minima at \( m^\dagger \).

Since \( m \) is the number of times actions are played, we (trivially) need \( m \geq 1 \) and \( m \) to be an integer. We choose

\[
m^\dagger = \left( \frac{T \sqrt{2 \log(T)}}{n + 2nk \sqrt{2 \log(T)}} \right)^{2/3}.
\]

(39)

Since from (38) we have that \( g''(m) > 0 \) for positive \( m \), \( g(m^\dagger) \leq g(m^\dagger) \).

For \( T \geq n(k + 1) \), we have

\[
m^\dagger = \left( \frac{T \sqrt{2 \log(T)}}{n + 2nk \sqrt{2 \log(T)}} \right)^{2/3} \geq \left( \frac{n(k + 1)}{\sqrt{2 \log(n(k+1))} + 2nk} \right)^{2/3} \geq \left( \frac{k + 1}{2k + 1} \right)^{2/3} \geq \left( \frac{1}{2} \right)^{2/3}.
\]

(40)

Plugging (39) back in to (34),

\[
E[R(T)] \leq m^\dagger nk \left( 1 + 2k \sqrt{2 \log(T)} \right) + 2kT \sqrt{2 \log(T)/m^\dagger}
\]

(34) with \( m^\dagger \) samples for each action

\[
= \left[ m^\dagger \right] nk \left( 1 + 2k \sqrt{2 \log(T)} \right) + 2kT \sqrt{2 \log(T)/m^\dagger}
\]

(Since \( \left[ m^\dagger \right] \geq m^\dagger \))

\[
\leq m^\dagger nk \left( 1 + 2k \sqrt{2 \log(T)} \right) + 2kT \sqrt{2 \log(T)/m^\dagger}
\]

(Since \( m^\dagger \geq 1/2, \ [m^\dagger] \leq 2m^\dagger \))

\[
= 2 \left( \frac{T \sqrt{2 \log(T)}}{n + 2nk \sqrt{2 \log(T)}} \right)^{2/3} nk(1 + 2k \sqrt{2 \log(T)}) + 2kT \sqrt{2 \log(T)} \left( \frac{n + 2nk \sqrt{2 \log(T)}}{T \sqrt{2 \log(T)}} \right)^{1/3}
\]

(using (37))

\[
= \frac{2(T \sqrt{2 \log(T)})^{2/3}}{n^{2/3}(1 + 2k \sqrt{2 \log(T)})^{2/3}} nk(1 + 2k \sqrt{2 \log(T)}) + 2kT \sqrt{2 \log(T)} \left( \frac{n^{1/3}(1 + 2k \sqrt{2 \log(T)})^{1/3}}{(T \sqrt{2 \log(T)})^{1/3}} \right)
\]

(rearranging)

\[
= 2(T \sqrt{2 \log(T)})^{2/3} n^{1/3} k \left( 1 + 2k \sqrt{2 \log(T)} \right)^{1/3} + 2k(T \sqrt{2 \log(T)})^{2/3} n^{1/3} (1 + 2k \sqrt{2 \log(T)})^{1/3}
\]

(cancelling common terms)

\[
= 4n^{1/3} k \sqrt{T} \left( 1 + 2k \sqrt{2 \log(T)} \right)^{1/3}
\]

(41)

\[= \mathcal{O} \left( n^{1/3} k \sqrt{T} \right). \]

where (41) follows by factoring. In conclusion, the expected \( (1 - 1/e) \) regret (2) is upper bounded by (41) if the clean event \( \mathcal{E} \) happens.
Case 2: clean event \( E \) does not happen

We next derive an upper bound for the expected \((1 - 1/e)\) regret \((2)\) for case that the event \( E \) does not happen. By Lemma A.2

\[
P(\bar{E}) = 1 - P(E) \leq \frac{2nk}{T^4}.
\]

Since the reward function \( f_t(\cdot) \) is upper bounded by 1, the expected \((1 - 1/e)\) regret \((2)\) incurred under \( \bar{E} \) for a horizon of \( T \) is at most \( T \).

\[
E[\mathcal{R}(T)|\bar{E}] \leq T.
\] (42)

Putting it all together

Combining Cases 1 and 2 we have,

\[
E[\mathcal{R}(T)] = E[\mathcal{R}(T)|E] \cdot P(E) + E[\mathcal{R}(T)|\bar{E}] \cdot P(\bar{E}) \quad \text{(Law of total expectation)}
\]

\[
\leq \left[ 4n \frac{k}{T} (\sqrt{2\log(T)})^\frac{3}{2} (1 + 2k\sqrt{2\log(T)})^{\frac{3}{2}} \right] \cdot 1 + T \cdot 2nkT^{-4} \quad \text{(using (41), Lemma A.2 and (42))}
\]

\[
= O(n \frac{k}{T} T^{\frac{3}{2}} \log(T)).
\]

This concludes the proof of Theorem 4.1.

B BASIC FACTS ABOUT SUBMODULAR FUNCTIONS

B.1

For completeness, we show the (well-known) lower bound

\[
1 - \left(1 - \frac{1}{k}\right)^k \geq 1 - \frac{1}{e}
\] (43)

for all \( k \geq 1 \). The right hand side is the limit of the left hand side as \( k \to \infty \).

Proof. First, we consider the limit of \( 1 - \left(1 - \frac{1}{k}\right)^k \). Rearranging,

\[
1 - \left(1 - \frac{1}{k}\right)^k = 1 - \left(\frac{k - 1}{k}\right)^k = 1 - \left(\frac{k - 1}{k}ight)^k
\]

\[
= \frac{k^k - (k - 1)^k}{k^k} = \frac{k^k}{(k-1)^k} - 1
\]

\[
= \frac{(1 + \frac{1}{k-1})^k - 1}{(1 + \frac{1}{k-1})^k} = \frac{(1 + \frac{1}{k-1})^{k-1} (1 + \frac{1}{k-1}) - 1}{(1 + \frac{1}{k-1})^{k-1} (1 + \frac{1}{k-1})}
\]

Since \( \lim_{k \to \infty} (1 + \frac{1}{k})^k = e \), in the limit as \( k \to \infty \),

\[
\lim_{k \to \infty} \left(1 - \frac{1}{k}\right)^k = \frac{e \cdot 1 - 1}{e \cdot 1} = 1 - \frac{1}{e}.
\]

Second, we show that \( 1 - \left(1 - \frac{1}{x}\right)^x \) is decreasing in \( k \), and thus its limit is a lower bound for all \( k \). Consider the continuous function \( 1 - \left(1 - \frac{1}{x}\right)^x \) for \( x > 1 \). Its derivative is

\[
\frac{d}{dx} \left(1 - \frac{1}{x}\right)^x = 0 - \left(1 - \frac{1}{x}\right)^x \log x.
\] (44)
For \( x > 1, 0 < \frac{1}{x} < 1, \) hence \( (1 - \frac{1}{x}) > 0, \) \( \log x > 0 \) for all \( x > 1, \) thus the derivative is negative for all \( x > 1 \) and consequently \( 1 - (1 - \frac{1}{x})^x \) is monotone decreasing. \( \square \)

**B.2**

For completeness, we reproduce the following result from [Nemhauser et al., 1978].

**Lemma B.1.** For a monotonically non-decreasing submodular function \( f \) defined on the set of subsets of \( \Omega, \) we have for arbitrary \( A, B \subseteq \Omega, \)

\[
f(B) - f(A) \leq \sum_{j \in B \setminus A} [f(A \cup \{j\}) - f(A)].
\]

**Proof.** Enumerate the elements that are in set \( B \) but not in \( A \) as \( B \setminus A = \{j_1, \cdots, j_q\}. \) We have

\[
f(A \cup B) - f(A) = \sum_{\ell=1}^q [f(A \cup \{j_1, \cdots, j_\ell\}) - f(A \cup \{j_1, \cdots, j_{\ell-1}\})] \quad \text{(telescoping sum)}
\]

\[
\leq \sum_{\ell=1}^q [f(A \cup \{j_\ell\}) - f(A)],
\]

with (45) following from the definition of submodularity in Section 1.

Using the monotonicity of \( f, \) we have

\[
f(B) \leq f(A \cup B)
\]

\[
\iff -f(A \cup B) + f(B) \leq 0.
\]

(46)

Adding (46) and (45), we get

\[
f(B) - f(A) \leq \sum_{\ell=1}^q [f(A \cup \{j_\ell\}) - f(A)].
\]

\( \square \)
C ALGORITHM OG* 

In this section we describe implementation details and parameter selection for OG* algorithm [Streeter and Golovin 2008]. The choice of exploration probability is given by the original paper: \( \gamma = n^{1/3}k \left( \frac{\log(n)}{T} \right)^{1/3} \). \( \epsilon \) is the learning rate for Randomized Weighted Majority (WMR) expert algorithm [Arora et al. 2012]. It is chosen by setting the derivative of regret upper bound to zero, which is \( \epsilon = \sqrt{\frac{k \log(n)}{T} \gamma} \). In experiments, there are many cases the chosen \( \gamma \) is large or even larger than 1, so we cap the probability of exploring \( \gamma \) by 1/2 to avoid exploring too much. The following is the pseudo code for implementation details of this algorithm.

**Algorithm 2** Online Greedy for Opaque Feedback Model (OG*)

**Input:** set of base arms \( \Omega \), horizon \( T \), cardinality constraint \( k \)

**Initialize** \( n \leftarrow |\Omega| \), \( \gamma \leftarrow n^{1/3}k \left( \frac{\log(n)}{T} \right)^{1/3} \), \( \epsilon \leftarrow \sqrt{\frac{k \log(n)}{T} \gamma} \)

**Initialize** \( \omega_1 \leftarrow \text{ones}(k, n) \)

**for** \( t \in [1, \cdots, T] \) **do**

**S** \( _t \leftarrow \emptyset \)

\( l \leftarrow \text{zeros}(k, n) \)  

Randomly sample a value \( \xi \sim \text{Uniform}([0, 1]) \)

if \( \xi \leq \gamma \) **then**

\( e \sim \text{Uniform}\{1, \cdots, k\} \)

**for** \( i \in [1, \cdots, e-1] \) **do**

Select an arm \( a \) with probability \( \frac{\omega_t[i,a]}{\sum_{j=1}^{n} \omega_t[i,j]} \), re-sample if \( a \in S_t \)

\( S_t \leftarrow S_t \cup \{a\} \)

end for

\( a \sim \text{Uniform}\{1, \cdots, n\} \setminus S_t \)

\( S_t \leftarrow S_t \cup \{a\} \)

Play action \( S_t \), observe \( f_t(S_t) \)

Update \( l[i, j] \leftarrow f_t(S_t) \) for all \( i = e \) and \( j \neq a \)  

Feed back \( f_t(S_t) \) to expert \( e \) associated with action \( a \)

Update \( \omega_{t+1}[i, j] \leftarrow \omega_t[i, j] \exp(-\epsilon l[i, j]) \) for all pairs of \( i \) and \( j \)

else

**for** \( i \in [1, \cdots, k] \) **do**

Select an arm \( a \) with probability \( \frac{\omega_t[i, a]}{\sum_{j=1}^{n} \omega_t[i, j]} \), re-sample if \( a \in S_t \)

\( S_t \leftarrow S_t \cup \{a\} \)

end for

Play action \( S_t \), observe \( f_t(S_t) \)

\( \omega_{t+1}[i, j] \leftarrow \omega_t[i, j] \)  

Since feeding back 0 to all expert-action payoffs, loss is 0, no update

end if

end for