Pure Exploration of Combinatorial Bandits

Shouyuan Chen

April 24, 2014

1 Preliminaries

1.1 Problems

Let n be the number of base arms. Let $\mathcal{M} \subseteq 2^{[n]}$ be the set of super arms.
In this note, we consider the following cases of $\mathcal{M}$.

Example 1 (Explore-m)**.** *$\mathcal{M}_{\mathsf{TOP}m}(n) = \{M \subseteq [n] \mid |M| = m\}$. This corresponds to finding the top m arms from $[n]$.*

Example 2 (Explore-m-bandits)**.** *Suppose $n = mk$. Then $\mathcal{M}_{\mathsf{BANDIT}m}(n)$ contains all subsets $M \subseteq [n]$ with size m, such that*

$$M \cap \{ik+1, \ldots, (i+1)k\} = 1, \quad \textit{for all } i \in \{0, \ldots, m-1\}.$$

This corresponds to finding the top arms from m bandits, where each bandit has k arms.

Example 3 (Perfect Matching)**.** *Let $G = (V, E)$ be a bipartite graph and $|E| = n$. For simplicity, let each edge $e \in E$ corresponds to a unique integer $i \in [n]$, and vice versa. Then $\mathcal{M}_{\mathsf{MATCH}}(n, G)$ contains all subsets $M \subseteq [n]$ such that M corresponds to a perfect matching in G.*

1.2 Diff-Sets

Definition 1 (Diff-set)**.** *An n-diff-set (or diff-set in short) is a pair of sets $c = (c_+, c_-)$, where $c_+ \subseteq [n]$, $c_- \subseteq [n]$ and $c_+ \cap c_- = \emptyset$.*

Definition 2 (Difference of sets)**.** *Given any $M_1 \subseteq [n], M_2 \subseteq [n]$. We define $M_1 \ominus M_2 \triangleq C$, where $C = (C_+, C_-)$ is a diff-set and $C_+ = M_1 \backslash M_2$ and $C_- = M_2 \backslash M_1$.*

Definition 3. *Denote $\mathsf{diff}[n]$ be the set of all possible n-diff-sets.*

Definition 4 (Set operations of diff-sets)**.** *Let $C = (C_+, C_-), D = (D_+, D_-)$ be two diff-sets. We define $C \cap D \triangleq (C_+ \cap D_+, C_- \cap D_-)$ and $C \backslash D \triangleq (C_+ \backslash D_+, C_- \backslash D_-)$.*
Further, for all $e \in [n]$, $e \in C \Leftrightarrow (e \in C_+) \vee (e \in C_-)$. And $|C| \triangleq |C_+| + |C_-|$.

Definition 5 (Valid diff-set)**.** *Given a set $M \subseteq [n]$ and a diff-set $C = (C_+, C_-)$, we call C a* valid diff-set *for M, iff $C_+ \cap M = \emptyset$ and $C_- \subseteq M$. In this case, we denote $C \prec M$.*

Definition 6 (Negative diff-set)**.** *Given a diff-set $A = (A_+, A_-)$, we define $\neg A = (A_-, A_+)$.*

1.2.1 diff-set operations

Definition 7 (Operators $\oplus$ and $\ominus$)**.** *Given any $M \subseteq [n]$ and $C \in \mathsf{diff}[n]$. If $C \prec M$, we define operator $\oplus$ such that $M \oplus C \triangleq M \backslash C_- \cup C_+$. On the other hand if $\neg C \prec M$, we define operator $\ominus$ such that $M \ominus C \triangleq M \oplus (\neg C) = M \backslash C_+ \cup C_-$.*

Definition 8. *Given two diff-sets $A = (A_+, A_-)$ and $B = (B_+, B_-)$. We denote $B \prec A$, if and only if $B_+ \cap A_+ = \emptyset$ and $A_+ \cap A_- = \emptyset$.*

Definition 9. *Given two diff-sets $A = (A_+, A_-)$ and $B = (B_+, B_-)$. If $B \prec A$, we define $A \oplus B = ((A_+ \cup B_+)\backslash(A_- \cup B_-), (A_- \cup B_-)\backslash(A_+ \cup B_+))$.*

Lemma 1. *Given two diff-sets $A = (A_+, A_-)$ and $B = (B_+, B_-)$. If $B \prec A$, then $A \oplus B$ is a diff-set.*

Proof. Let $C = A \oplus B$. By definition, we have $C_+ = (A_+ \cup B_+)\backslash(A_- \cup B_-)$ and $C_- = (A_- \cup B_-)\backslash(A_+ \cup B_+)$. We only need to show that $C_+ \cap C_- = \emptyset$.

$$\begin{aligned} C_+ \cap C_- &= \big((A_+ \cup B_+)\backslash(A_- \cup B_-)\big) \cap \big((A_- \cup B_-)\backslash(A_+ \cup B_+)\big) \\ &= (A_+ \cup B_+) \cap \big((A_- \cup B_-)\backslash(A_+ \cup B_+)\backslash(A_- \cup B_-)\big) \\ &= \emptyset. \end{aligned}$$

□

Lemma 2. *Given two diff-sets $A = (A_+, A_-)$ and $B = (B_+, B_-)$. If there exists $M \subseteq [n]$ such that $A \prec M$, and $B \prec (M \oplus A)$, then $B \prec A$ and $(M \oplus A \oplus B) \ominus M = A \oplus B$.*

Proof. We first show that $B \prec A$. Since $B \prec (M \oplus A)$, we know that $B_+ \cap (M\backslash A_- \cup A_+) = \emptyset$. Therefore, we have

$$\begin{aligned} \emptyset &= B_+ \cap (M\backslash A_- \cup A_+) \\ &= (B_+ \cap (M\backslash A_-)) \cup (B_+ \cap A_+) \end{aligned}$$

We see that $B_+ \cap A_+ = \emptyset$.
On the other hand, we have $B_- \subseteq (M\backslash A_- \cup A_+)$, therefore

$$\begin{aligned} B_- \cap A_- &\subseteq (M\backslash A_- \cup A_+) \cap A_- \\ &= (M\backslash A_- \cap A_-) \cup (A_+ \cap A_-) \\ &= \emptyset. \end{aligned}$$

Hence we proved that $B \prec A$.
Define $D = (M \oplus A \oplus B) \ominus M$ and write $D = (D_+, D_-)$. Then,

$$\begin{aligned} D_+ &= (M \oplus A \oplus B)\backslash M \\ &= (M\backslash A_- \cup A_+\backslash B_- \cup B_+)\backslash M \\ &= (A_+ \cup B_+)\backslash(A_- \cup B_-). \end{aligned}$$

Similarly, we have

$$\begin{aligned} D_- &= M\backslash(M \oplus A \oplus B) \\ &= M\backslash(M\backslash A_- \cup A_+\backslash B_- \cup B_+) \\ &= (A_- \cup B_-)\backslash(A_+ \cup B_+). \end{aligned}$$

□

1.2.2 Diff-set class

Definition 10 (Decomposition of diff-set)**.** *Given $\mathcal{B} \subseteq \mathsf{diff}[n]$ and $D \in \mathsf{diff}[n]$, a decomposition of D on $\mathcal{B}$ is a set $\{b_1, \ldots, b_k\} \subseteq \mathcal{B}$ satisfying the following*

1. *For all $i \in [k]$ and $j \in [k]$, we write $b_i = (b_i^+, b_i^-)$ and $b_j = (b_j^+, b_j^-)$. Then, the following holds $b_i^+ \cap b_j^+ = \emptyset$, $b_i^+ \cap b_j^- = \emptyset$, $b_i^- \cap b_j^+ = \emptyset$ and $b_i^- \cap b_j^- = \emptyset$.*

2. *$D = b_1 \oplus b_2 \oplus \ldots b_k$.*

Lemma 3. *Given $\mathcal{B} \subseteq \mathsf{diff}[n]$ and $D \in \mathsf{diff}[n]$. Let $\{b_1, \ldots, b_k\} \subseteq \mathcal{B}$ be a decomposition of D on $\mathcal{B}$. Then,*

1. *Let $D = (D_+, D_-)$ and for all $i \in [k]$, we write $b_i = (b_i^+, b_i^-)$. Then $D_+ = b_1^+ \cup \ldots \cup b_k^+$ and $D_- = b_1^- \cup \ldots \cup b_k^-$.*

2. *For all $M \subseteq [n]$, if $D \prec M$, then, for all $i \in [k]$, we have $b_i \prec M$.*

Proof. We prove (1) by induction. Let $D_i = b_1 \oplus \ldots \oplus b_i$ and write $D_i = (D_i^+, D_i^-)$. We show that $D_i^+ = \bigcup_{j=1}^i b_i^+$ and $D_{i-} = \bigcup_{j=1}^i b_i^-$ for all $i \in [k]$. For $i = 1$, this is trivially true. Then, assume that this is true for some $i > 1$. By definition $D_{i+1} = D_i \oplus b_{i+1}$, hence $D_{i+1}^+ = (D_i^+ \cup b_{i+1}^+) \backslash (D_i^- \cup b_{i+1}^-)$. Note that

$$\begin{aligned}(D_i^- \cup b_{i+1}^-) \cap (D_i^+ \cup b_{i+1}^+) &= (D_i^- \cap D_i^+) \cup (D_i^- \cap b_{i+1}^+) \cup (b_{i+1}^- \cap D_i^+) \cup (b_{i+1}^- \cap b_{i+1}^+) \\ &= (D_i^- \cap b_{i+1}^+) \cup (b_{i+1}^- \cap D_i^+) \\ &= \left(\left(\bigcup_{j=1}^{i} b_j^-\right) \cap b_{i+1}^+\right) \cup \left(\left(\bigcup_{j=1}^{i} b_j^+\right) \cap b_{i+1}^-\right) \\ &= \emptyset.\end{aligned}$$

Hence $D_{i+1}^+ = D_i^+ \cup b_{i+1}^+$. We can use the same method to show that $D_{i+1}^- = D_i^- \cup b_{i+1}^-$.
Next, we prove (2) using (1). To show that $b_i \prec M$, we only need to show that $b_i^+ \cap M = \emptyset$ and $b_i^- \subseteq M$. Since $D \prec M$, we know that $D_+ \cap M = \emptyset$ and $D_- \subseteq M$. By (1), we see that $b_i^+ \subseteq D_+$ and $b_i^- \subseteq D_-$. Therefore, we have $(b_i^+ \cap M) \subseteq (D_+ \cap M) = \emptyset$ and $b_i^- \subseteq D_- \subseteq M$.

□

Definition 11 (diff-set class)**.** *Given $\mathcal{M} \subseteq 2^{[n]}$. $\mathcal{B} \subseteq \mathsf{diff}[n]$ is a diff-set class for $\mathcal{M}$, if the following hold.*

1. *$(\emptyset, \emptyset) \notin \mathcal{B}$.*

2. *For all $M \in \mathcal{M}$ and for all $b \in \mathcal{B}$, if $b \prec M$, then $M \oplus b \in \mathcal{M}$.*

3. *For all $M_1 \in \mathcal{M}$ and $M_2 \in \mathcal{M}$, where $M_1 \neq M_2$. Let $D = M_1 \ominus M_2$. Then, there exists a decomposition of D on $\mathcal{B}$.*

Definition 12 (Rank of diff-set class)**.** *Let $\mathcal{B} \subseteq [n]$ be a diff-set class for some $\mathcal{M}$. We define*

$$\mathrm{width}(\mathcal{B}) \triangleq \max_{b \in \mathcal{B}} |b|.$$

Example 4 (diff-set class for Explore-m)**.** *One diff-set class $\mathcal{B}$ for $\mathcal{M}_{\mathsf{TOP}m}(n)$ is given by*

$$\mathcal{B} = \{(\{b_1\}, \{b_2\}) \mid b_1 \neq b_2, b_1 \in [n], b_2 \in [n]\}.$$

Proof omitted. Further, we see that $\mathrm{width}(\mathcal{B}) = 2$.

Example 5 (diff-set class for Explore-m-badit)**.** *Let $n = mk$. One diff-set class $\mathcal{B}$ for $\mathcal{M}_{\mathsf{BANDIT}m}(n)$ is given by*

$$\mathcal{B} = \{(\{b_1\}, \{b_2\}) \mid b_1 \neq b_2, \exists i \in \{0, \ldots, k-1\}, b_1 \in \{ik+1, \ldots, (i+1)k\}, b_2 \in \{ik+1, \ldots, (i+1)k\}\}.$$

Proof omitted. Further, we see that $\mathrm{width}(\mathcal{B}) = 2$.

Example 6 (diff-set class for Perfect Matching)**.** *One diff-set class $\mathcal{B}$ for $\mathcal{M}_{\mathsf{MATCH}}(n, G)$ is the set of all augmenting cycles of G. More specifically,*

$$\mathcal{B} = \{(b_+, b_-) | b_+ \cup b_- \textit{ is a cycle of } G\}.$$

Note $\mathrm{width}(\mathcal{B}) \leq n$.

1.3 Weights and confidence bounds

Definition 13 (Weight functions)**.** *Define function $w : [n] \to \mathbb{R}^+$ which represents the weight of each base arm. Further, we slight abuse the notations, and extend the definition of w to diff-sets and sets as follows.*

1. *For all $M \subseteq [n]$, we denote $w(M) = \sum_{e \in M} w(e)$.*
2. *For all $b = (b_+, b_-) \in \mathsf{diff}[n]$, we denote $w(b) = \sum_{e \in b_+} w(e) - \sum_{e \in b_-} w(e)$.*

Lemma 4. *Let $c \in \mathsf{diff}[n], d \in \mathsf{diff}[n]$. Let w be a weight function. Then,*

$$w(c \cup d) = w(c) + w(d) - w(c \cap d). \tag{1}$$

Proof. Let $c = (c_+, c_-)$ and $d = (d_+, d_-)$. We have

$$\begin{aligned} w(c \cup d) &= w(c_+ \cup d_+) - w(c_- \cup d_-) & (2)\\ &= w(c_+) + w(d_+) - w(c_+ \cap d_+) - w(c_-) - w(d_-) + w(c_- \cap d_-) & (3)\\ &= w(c) + w(d) - (w(c_+ \cap d_+) - w(c_- \cap d_-)) & (4)\\ &= w(c) + w(d) - w(c \cap d). & (5) \end{aligned}$$

□

Definition 14 (Mean weight $\bar{w}_t$, sample size n_t)**.** *Given $t > 0$. Define $\bar{w}_t$ be a weight function such that, for all $e \in [n]$, $\bar{w}_t(e)$ equals to the empirical mean of e up to round t. Let $n_t : [n] \to \mathbb{N}$, such that $n_t(e)$ equals to number of plays of base arm e up to round t.*

Definition 15 (Confidence radius rad_t)**.** *Given n and $t > 0$. Define $\mathrm{rad}_t : [n] \to \mathbb{R}^+$ satisfying, for all $e \in [n]$,*

$$\mathrm{rad}_t(e) = c_{\mathrm{rad}} \log\left(\frac{c_\delta n t^2}{\delta}\right) \frac{1}{\sqrt{n_t(e)}}, \tag{6}$$

where $c_{\mathrm{rad}} > 0$ and $c_\delta > 0$ are some universal constants (specify later) and $\delta > 0$ is a parameter.
We extend the notation of rad_t to diff-sets and sets as follows.

1. *For all $M \subseteq [n]$, $\mathrm{rad}_t(M) \triangleq \sum_{e \in M} \mathrm{rad}_t(e)$.*
2. *For all $b = (b_+, b_-) \in \mathsf{diff}[n]$, $\mathrm{rad}_t(b) \triangleq \mathrm{rad}_t(b_+) + \mathrm{rad}_t(b_-)$.*

Definition 16 (UCB w_t^+)**.** *Define $w_t^+ : [n] \to \mathbb{R}^+$, s.t., for all $e \in [n]$,*

$$w_t^+(e) = \bar{w}_t(e) + \mathrm{rad}_t(e).$$

We extend the notation of w_t^+ to diff-sets and sets as follows.

1. *For all $M \subseteq [n]$, $w_t^+(M) \triangleq \bar{w}_t(M) + \mathrm{rad}_t(M)$.*
2. *For all $b = (b_+, b_-) \in \mathsf{diff}[n]$, $w_t^+(b) \triangleq \bar{w}_t(b) + \mathrm{rad}_t(b)$.*

Lemma 5. *Define random event*

$$\xi = \left\{ \forall e \in [n] \ \forall t > 0, |\bar{w}_t(e) - w(e)| \le \mathrm{rad}_t(e) \right\}.$$

Then, there exist constants c_{rad} *and* c_δ*,*

$$\Pr[\xi] \ge 1 - \delta.$$

Proof. Hoeffding inequality and union bound. □

Corollary 1.

$$\xi \implies \forall t, \forall e \in [n] \ w_t^+(e) \ge w(e).$$

$$\xi \implies \forall t, \forall M \subseteq [n], \ w_t^+(M) \ge w(M).$$

$$\xi \implies \forall t, \forall b \in \mathsf{diff}[n] \ w_t^+(b) \ge w(b).$$

1.4 Properties of rad_t

Lemma 6. *Let* $c \in \mathsf{diff}[n], d \in \mathsf{diff}[n]$*. Then*

$$\mathrm{rad}_t(c \backslash d) = \mathrm{rad}_t(c) - \mathrm{rad}_t(c \cap d). \tag{7}$$

Proof. Let $c = (c_+, c_-)$ and $d = (d_+, d_-)$. We have

$$\begin{aligned} \mathrm{rad}_t(c \backslash d) &= \mathrm{rad}_t(c_+ \backslash d_+) + \mathrm{rad}_t(c_- \backslash d_-) \\ &= \mathrm{rad}_t(c_+) - \mathrm{rad}_t(c_+ \cap d_+) + \mathrm{rad}_t(c_-) - \mathrm{rad}_t(c_- \cap d_-) \\ &= \mathrm{rad}_t(c) - \mathrm{rad}_t(c \cap d). \end{aligned}$$

□

Lemma 7. *Let* $C = (C_+, C_-)$ *and* $D = (D_+, D_-)$ *be two diff-sets. If* $D \prec C$*, then*

$$\mathrm{rad}_t(C \oplus D) = \mathrm{rad}_t(C) + \mathrm{rad}_t(D) - 2\,\mathrm{rad}_t(C_+ \cap D_-) - 2\,\mathrm{rad}_t(C_- \cap D_+).$$

In addition, if $\neg D \prec C$*, then*

$$\mathrm{rad}_t(C \ominus D) = \mathrm{rad}_t(C) + \mathrm{rad}_t(D) - 2\,\mathrm{rad}_t(C_+ \cap D_+) - 2\,\mathrm{rad}_t(C_- \cap D_-).$$

Proof. We prove the first part of the lemma. The second part follows from the first part and the definition of $\neg D$.
By definition, we have $C \oplus D = ((C_+ \cup D_+) \backslash (C_- \cup D_-), (C_- \cup D_-) \backslash (C_+ \cup D_+))$. Hence, we have

$$\begin{aligned} \mathrm{rad}_t((C_+ \cup D_+) \backslash (C_- \cup D_-)) &= \mathrm{rad}_t(C_+ \cup D_+) - \mathrm{rad}_t((C_+ \cup D_+) \cap (C_- \cup D_-)) \qquad (8) \\ &= \mathrm{rad}_t(C_+) + \mathrm{rad}_t(D_+) - \mathrm{rad}_t((C_+ \cup D_+) \cap (C_- \cup D_-)), \qquad (9) \end{aligned}$$

where the second equality holds due to $C_+ \cap D_+ = \emptyset$ by the definition of $D \prec C$.
Similarly, we have

$$\mathrm{rad}_t((C_- \cup D_-) \backslash (C_+ \cup D_+)) = \mathrm{rad}_t(C_-) + \mathrm{rad}_t(D_-) - \mathrm{rad}_t((C_+ \cup D_+) \cap (C_- \cup D_-)).$$

Combine both equalities, we have

$$\mathrm{rad}_t(C \oplus D) = \mathrm{rad}_t((C_+ \cup D_+) \backslash (C_- \cup D_{)}) + \mathrm{rad}_t((C_- \cup D_-) \backslash (C_- \cap D_+)) \tag{10}$$

$$= \mathrm{rad}_t(C_+) + \mathrm{rad}_t(D_+) + \mathrm{rad}_t(C_-) + \mathrm{rad}_t(D_-) - 2\,\mathrm{rad}_t((C_+ \cup D_+) \cap (C_- \cup D_-)) \quad (11)$$

$$= \mathrm{rad}_t(C) + \mathrm{rad}_t(D) - 2\,\mathrm{rad}_t((C_+ \cup D_+) \cap (C_- \cup D_-)). \quad (12)$$

□

2 Pure Exploration of Combinatorial Bandits

`ExpCMAB`: problem formulation. Suppose that the arms are numbered $1, 2, \ldots, n$. Each arm $e \in [n]$ is associated with a reward distribution φ_e. We assume that all reward distributions are b-subgaussian, i.e. (). Notice that all distributions that are supported on $[0, b]$ are b-subgaussian distributions []. Let $w(e)$ denote the expected reward of arm e, i.e. $w(e) = \mathbb{E}_{X\sim\varphi_e}[X]$. In addition, for any set of arms $M \subseteq [n]$, we define $w(M) = \sum_{e\in M} w(e)$ as the sum of expected rewards of arms that belong to M.

The learning problem of pure exploration combinatorial bandit can be formalized as a game between a learner and a stochastic environment. At the beginning of the game, the learner is given a collection of feasible sets $\mathcal{M} \subseteq 2^{[n]}$ which corresponds to some combinatorial problem. And the reward distributions $\{\varphi_e\}_{e\in[n]}$ are unknown to the learner. Then, the game is played for multiple rounds; on each round t, the learner pulls an arm $p_t \in [n]$ and observes a reward sampled from the associated reward distribution φ_{p_t}. The game continues until certain stopping condition is satisfied (specify later). After the game finishes, the learner is asked to output a set of arms $\mathsf{Out} \in \mathcal{M}$ which approximately maximizes the sum of expected weight, i.e. $w(M_*) - w(\mathsf{Out}) \le \epsilon$, where we denote $M_* = \arg\max_{M\in\mathcal{M}} w(M)$ to be the optimal set of arms. For the sake of simplicity, we shall assume that the optimal set M_* is unique throughout the paper. Notice that, if $\epsilon = 0$, then the learner is required to identify the optimal set, i.e. $\mathsf{Out} = M_*$.

Fixed confidence and fixed budget. We consider two different settings: *fixed confidence* and *fixed budget*. In the fixed confidence setting, the learner aims to achieve a fixed confidence about the optimality of the returned set using a small number of samples (pulls). Specifically, given a confidence parameter δ, the learner need to guarantee that $\Pr[w(M_*) - w(\mathsf{Out}) \le \epsilon] \ge 1 - \delta$ and the performance is evaluated by the number of pulls used by the learner. Notice that the learner can stop the game at any point in this setting. In the fixed budget setting, the learner tries to minimize the probability of error $\Pr[w(M_*) - w(\mathsf{Out}) > \epsilon]$ by using a fixed number of samples, i.e. the game stops after a fixed number of rounds. In this case, the learner's performance is measured by the probability of error.

Applications. Our formulation of the `ExpCMAB` problem covers many online learning tasks. We consider the following applications as running examples.

- **Multi**.
- **Match**.
- **Path**.

Useful notations.

3 Algorithm and Main Results

Our main contribution is an algorithm for solving the `ExpCMAB` problem. Our algorithm
Then, we analyze the sample complexity and the probability of error of our algorithm.

Maximization oracle. For most non-trivial combinatorial problems, the size of the collection of feasible sets $\mathcal{M}$ is exponential in n. Therefore, the learning algorithm needs a succinct representation of $\mathcal{M}$. In particular, we allow the learning algorithm to use a *maximization oracle* which can find the optimal set $M \in \mathcal{M}$ when the expected reward of each arm is known. Specifically, we assume that there exists an oracle which takes expected rewards $\{w(1), \ldots, w(n)\}$ as input and returns a set $\text{Oracle}(w) = \arg\max_{M \in \mathcal{M}} w(M)$. It is clear that a large class of combinatorial problems admit efficient maximization oracles.

Algorithm. Our algorithm works for both fixed confidence and fixed budget settings. In either settings, the behaviors of our algorithm only differ in the construction of confidence radius and the stopping condition. In the following, we describe the procedure of our algorithm. Our algorithm maintains empirical mean $\bar{w}_t(e)$ and confidence radius $\text{rad}_t(e)$ for each arm $e \in [n]$ and each round t. The construction of confidence radius ensures that $|w(e) - \bar{w}_t(e)| \le \text{rad}_t(e)$ holds with high probability for each arm $e \in [n]$ and each round $t > 0$. At each round t, our algorithm accesses the maximization oracle twice. The first access to the oracle computes the set $M_t = \arg\max_{M \in \mathcal{M}} \bar{w}_t(M)$. Notice that M_t is the "best" set according to the empirical means $\bar{w}_t$. Then, in order to explore possible refinements of M_t, the algorithm uses the confidence radius to compute an adjusted expectation vector $\tilde{w}_t$ in the following way: for each arm $e \in M_t$, $\tilde{w}_t(e)$ equals to the lower confidence bound $\tilde{w}_t(e) = \bar{w}_t(e) - \text{rad}_t(e)$; and for each arm $e \notin M_t$, $\tilde{w}_t(e)$ equals to the upper confidence bound $\tilde{w}_t(e) = \bar{w}_t(e) + \text{rad}_t(e)$. Intuitively, the adjusted expectation vector $\tilde{w}_t$ penalizes arms belonging to M_t and encourages exploring arms out of M_t. The algorithm then calls the oracle using the adjusted expectation vector $\tilde{w}_t$ as input, which returns another set $\tilde{M}_t = \arg\max_{M \in \mathcal{M}} \tilde{w}_t(M)$. The algorithm stops if $\tilde{w}_t(\tilde{M}_t) - \tilde{w}_t(M_t) \le \epsilon$ or the budget of samples is exhausted, i.e. $t = T$, in the fixed budget setting. In either cases, the algorithm outputs $\mathsf{Out} = M_t$ as result. Otherwise, the algorithm plays the arm belonging to the symmetric difference $(\tilde{M}_t \backslash M_t) \cup (M_t \backslash \tilde{M}_t)$ with the largest confidence radius in the end of round t. The pseudo-code of the algorithm is shown in Algorithm 1.

Algorithm 1 ExploreComb: Pure exploration algorithm for combinatorial bandits

```
Require: Confidence parameter: $\delta \in (0,1)$; Tolerance parameter: $\epsilon > 0$; Maximization oracle: Oracle($w$).
  Initialize: Play each arm $e \in [n]$ once. Initialize empirical means $\bar{w}_n(e)$ and set $T_n(e) \leftarrow 1$ for all $e$.
for $t = n, n+1, \ldots$ do
    $M_t \leftarrow \text{Oracle}(\bar{w}_t)$
    for $e \in [n]$ do
        if $e \in M_t$ then
            $\tilde{w}_t(e) \leftarrow \bar{w}_t(e) - \text{rad}_t(e)$
        else
            $\tilde{w}_t(e) \leftarrow \bar{w}_t(e) + \text{rad}_t(e)$
        end if
    end for
    $\tilde{M}_t \leftarrow \text{Oracle}(\tilde{w}_t)$
    if $\tilde{w}_t(\tilde{M}_t) - \tilde{w}_t(M_t) \le \epsilon$ then
        Out $\leftarrow M_t$
        return Out
    end if
    $p_t \leftarrow \arg\max_{(\tilde{M}_t \backslash M_t) \cup (M_t \backslash \tilde{M}_t)} \text{rad}_t(e)$
    Play $p_t$ and observe the reward
    Update empirical mean $\bar{w}_{t+1}(p_t)$
    Update $T_{t+1}(p_t) \leftarrow T_t(p_t) + 1$ and $T_{t+1}(e) \leftarrow T_t(e)$ for every $e \neq p_t$
end for
```

3.1 Analysis

In this part, we analyze the performance of Algorithm 1 for both fixed confidence and fixed budget settings.

Gap. We begin with defining a natural complexity measure of the `ExpCMAB` problem. For each arm $e \in [n]$, we define gap Δ_e as

$$\Delta_e = \begin{cases} w(M_*) - \max_{M \in \mathcal{M}: e \in M} w(M) & \text{if } e \notin M_*, \\ w(M_*) - \max_{M \in \mathcal{M}: e \notin M} w(M) & \text{if } e \in M_*. \end{cases} \tag{13}$$

By this definition of gap Δ_e, for each arm $e \notin M_*$, Δ_e represents the gap between the optimal set M_* and the best set that includes arm e; and, for each arm $e \in M_*$, Δ_e is the sub-optimality of the best set that does not include arm e. We notice that this definition resembles the definition of gaps for **Multi** proposed by ().

Exchange class. The analysis of our algorithm depends on certain exchange properties of combinatorial structures. To study these properties, we introduce notions of *exchange set* and *exchange class* as our tools for the analysis. In the following, we present their definitions.

We begin with the definition of exchange set. We define an exchange set b as an ordered pair of disjoint sets $b = (b_+, b_-)$. Then, we define operator $\oplus$ such that, for any set M and any exchange set $b = (b_+, b_-)$, we have $M \oplus b \triangleq M \backslash b_- \cup b_+$. We call a set of exchange sets $\mathcal{B}$ an *exchange class for* $\mathcal{M}$ if $\mathcal{B}$ satisfies the following property. Let M and M' be two elements of $\mathcal{M}$. Then, for any $e \in (M \backslash M')$, there exists an exchange set $(b_+, b_-) \in \mathcal{B}$ which satisfies $e \in b_-$, $b_+ \subseteq M'$, $b_- \subseteq M$ and $(M \oplus b) \in \mathcal{M}$. Finally, we define the *width* of exchange class $\mathcal{B}$ as follows

$$\operatorname{width}(\mathcal{B}) = \max_{(b_+,b_-)\in\mathcal{B}} |b_+| + |b_-|.$$

Intuitively, for any feasible sets M and M', there exists an "operation" $(b_+, b_-) \in \mathcal{B}$ which can transform M one step towards M'; this operation generates a new feasible set $M \oplus b$ by removing elements (including e) from M and adding elements which belongs to M'. One can chain these operations together such that, for any $M \neq M'$, there exists a sequence of exchange sets $b_1, \ldots, b_k$ of $\mathcal{B}$ such that $M' = M \oplus b_1 \ldots \oplus b_k$.

Next, we construct the exchange classes for our running examples. Our constructions are summarized in Lemma 8.

Lemma 8. *There exist exchange classes $\mathcal{B}_{\textbf{\textit{Multi}}}$, $\mathcal{B}_{\textbf{\textit{Matroid}}}$, $\mathcal{B}_{\textbf{\textit{Match}}}$ and $\mathcal{B}_{\textbf{\textit{Path}}}$ for $\mathcal{M}_{\textbf{\textit{Multi}}}$, $\mathcal{M}_{\textbf{\textit{Matroid}}}$, $\mathcal{M}_{\textbf{\textit{Match}}}$ and $\mathcal{M}_{\textbf{\textit{Path}}}$, respectively. These exchange classes can be constructed as follows*

1. $\mathcal{B}_{\textbf{\textit{Multi}}} = \big\{(\{i\}, \{j\}) \mid \forall i \in [n], j \in [n]\big\}$.
2. $\mathcal{B}_{\textbf{\textit{Matroid}}} = \big\{(\{i\}, \{j\}) \mid \forall i \in [n], j \in [n]\big\}$.
3. $\mathcal{B}_{\textbf{\textit{Match}}} = \big\{(C_+, C_-) \mid C_+ \cup C_-$ *is a cycle of* $G\big\}$.
4. $\mathcal{B}_{\textbf{\textit{Path}}} = \big\{(P_1, P_2) \mid P_1, P_2$ *are two disjoint paths of* G *with same endpoints*$\big\}$.

In addition, we have $\operatorname{width}(\mathcal{B}_{\textbf{\textit{Multi}}}) = 2$, $\operatorname{width}(\mathcal{B}_{\textbf{\textit{Matroid}}}) = 2$, $\operatorname{width}(\mathcal{B}_{\textbf{\textit{Match}}}) = |V|$ *and* $\operatorname{width}(\mathcal{B}_{\textbf{\textit{Path}}}) = |V|$.

The construction for **Multi** problem is straightforward. For **Matroid** problem, we leverage the basis exchange property of matroids (see Lemma 14 in the appendix). And for **Match** and **Path** problems, we use the graph-theoretical properties of matchings and paths. Figure Y illustrates these exchanges classes. A detailed proof of Lemma 8 is deferred to the supplementary material.

Main results. We show that, in the fixed confidence setting, our algorithm returns the optimal set with high probability and uses at most $\tilde{O}(\operatorname{width}(\mathcal{B})^2/\Delta_e^2)$ samples.

Theorem 1. *Given any $\delta \in (0, 1)$, any combinatorial problem $\mathcal{M} \subseteq 2^{[n]}$ and any expected rewards $\vec{w} \in \mathbb{R}^d$. Let $\mathcal{B}$ be an exchange class for $\mathcal{M}$ and $\{\Delta_e\}_{e\in[n]}$ be the gaps defined in Eq.* (13)*. Then, with probability at least $1 - \delta$, Algorithm 1 returns the optimal set and*

$$T \le O\left(\sum_{e\in[n]} \frac{\operatorname{width}(\mathcal{B})^2}{\Delta_e^2} \log\left(\frac{n}{\delta}\sum_{e\in[n]}\frac{\operatorname{width}(B)^2}{\Delta_e^2}\right)\right),$$

where T denotes the number of samples used by Algorithm 1.

Theorem 1 is highly general in the sense that it provides a sample complexity for any combinatorial problem $\mathcal{M}$. In addition, notice that $\mathrm{width}(\mathcal{B}_{\mathbf{Multi}}) = O(1)$. Therefore, our sample complexity bound for the **Multi** problem is $O(\sum_e \Delta_e^{-2} \log(n/\delta \sum_e \Delta_a^{-2}))$. This matches the previous best problem-dependent bounds for the **Multi** problem.

3.2 Lower bounds

In this part, we establish a problem dependent lower bound on the sample complexity of the `ExpCMAB` problem. To state our results, we first define the notion of *δ-correct algorithm* as follows. For any $\delta \in (0,1)$, we call an algorithm $\mathbb{A}$ a δ-correct algorithm if, for any expected reward $\vec{w}$, the probability of error of $\mathbb{A}$ is at most δ, i.e $\Pr[w(M_*) - w(\mathsf{Out}) \geq \epsilon] \leq \delta$, where Out is the output of algorithm $\mathbb{A}$.
Our next theorem shows shat, for any combinatorial problem $\mathcal{M}$, any expected rewards $\vec{w}$ and any δ-correct algorithm $\mathbb{A}$, algorithm $\mathbb{A}$ must use at least $\tilde{\Omega}\left(\sum_e \frac{1}{\Delta_e^2}\right)$ samples in expectation.

Theorem 2. *Fix any $\mathcal{M} \subseteq 2^{[n]}$ and any vector $\vec{w} \in \mathbb{R}^n$. Suppose that, for each arm $e \in [n]$, the reward distribution φ_e is given by $\varphi_e = \mathcal{N}(w(e), 1)$, where $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2. Then, for any $\delta \in (0, e^{-16}/4)$ and any δ-correct algorithm $\mathbb{A}$, we have*

$$\mathbb{E}[T] \geq \sum_e \frac{1}{16\Delta_e^2} \log(1/4\delta),$$

where T denote the number of total samples used by algorithm $\mathbb{A}$ and Δ_e is defined in Eq. (13).

Now, we compare the sample complexity of Algorithm 1 to the lower bound provided in Theorem 2 on our running examples **Multi**, **Matroid**, **Match** and **Path**. For clarity, we consider the case that $\epsilon = 0$ which corresponds to the learning problem of finding the optimal set. We see that Algorithm 1 uses at most $\tilde{O}(\sum_e \mathrm{width}(\mathcal{B})^2/\Delta_e^2)$ samples. Recall that, for **Multi** and **Matroid** problems, Lemma X shows that $\mathrm{width}(\mathcal{B}) = 2$. Hence, for these two problems, Algorithm 1 achieves optimal sample complexity within logarithmic factors.
On the other hand, for **Match**(V, E) and **Path**(V, E), Lemma X indicates that $\mathrm{width}(\mathcal{B}) = |V| \leq n$. This means that the gap between our algorithm and this lower bound is a factor of $|V|^2$. Notice this gap only depends on the underlying combinatorial structure of $\mathcal{M}$ and is independent of expected rewards $\vec{w}$. This means that the sample complexity of Algorithm 1 has an optimal dependency on $\{\Delta_e\}_{e\in[n]}$.
However, we still remain to investigate the necessity of the dependency on $\mathrm{width}(\mathcal{B})$ of Algorithm 1. To this end, we provide evidence showing that the sample complexity of any δ-correct algorithm should be related to size of exchange sets. In fact, we show that, for any optimal exchange set $b \in \mathcal{B}_{\mathsf{opt}}$ and any δ-correct algorithm, the algorithm must spend $\tilde{\Omega}\left(|b|^2/w(b)^2\right)$ samples on the arms belonging to b. This result is formalized in the following theorem.

Theorem 3. *Fix any $\mathcal{M} \subseteq 2^{[n]}$ and any vector $\vec{w} \in \mathbb{R}^n$. Suppose that, for each arm $e \in [n]$, the reward distribution φ_e is given by $\varphi_e = \mathcal{N}(w(e), 1)$, where $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2. Fix any $\delta \in (0, e^{-16}/4)$ and any δ-correct algorithm $\mathbb{A}$.*
Then, for any $b \in \mathcal{B}_{\mathsf{opt}}$, we have

$$\mathbb{E}[T_b] \geq \frac{|b|^2}{32w(b)^2} \log(1/4\delta),$$

where T_b denotes the number of samples of arms belonging to b used by algorithm $\mathbb{A}$.

Notice that

4 Proof of Main Results

Definition 17 (Optimal diff-sets)**.** *Given a diff-set class $\mathcal{B}$ and the optimal set M_*. We define $\mathcal{B}_{\mathsf{opt}}$ as a subset of $\mathcal{B}$, and for all $b \in \mathcal{B}$, $b \in \mathcal{B}_{\mathsf{opt}}$ if and only if, there exists $M \neq M_*$ and $M_* \ominus M$ can be decomposed as $b, b_1, \ldots, b_k$ on $\mathcal{B}$.*

Definition 18 (Hardness Δ_e of base arm e)**.** *For each $e \in [n]$, we define its hardness Δ_e as follows*

$$\Delta_e = \min_{b \in \mathcal{B}_{\mathsf{opt}}, e \in b} \frac{1}{\mathrm{width}(\mathcal{B})} w(b).$$

Definition 19 (Sufficient exploration)**.** *For all $t > 0$, we define $E_t^3 \subseteq [n]$, such that, for all $e \in [n]$ $e \in E_t^3$ if and only if $\mathrm{rad}_t(e) < \frac{1}{3}\Delta_e$.*

Corollary 2. *For all $t > 0$ and $e \in [n]$*

$$n_t(e) \geq O(\frac{1}{\Delta_e^2} \log(\Delta_e n/\delta)) \implies e \in E_t^3.$$

Theorem 4. *With probability at least $1 - \delta$, the algorithm returns M_*, and the number of samples used by the algorithm are at most*

$$\sum_{e \in [n]} \Delta_e^{-2} \log(\Delta_e n/\delta).$$

Theorem 5. *Given confidence parameter $\delta \in (0, 1)$, tolerance parameter $\epsilon \geq 0$, number of arms n and a combinatorial problem instance $\mathcal{M} \subseteq 2^{[n]}$. Let oracle $\mathrm{Oracle}(w)$ be a maximization oracle associated with $\mathcal{M}$ such that $\mathrm{Oracle}(w) = \arg\max_{M \in \mathcal{M}} w(M)$, where $w : 2^{[n]} \to R$ is a weight function.*
Then, with probability at least $1 - \delta$, the output Out of Algorithm 1 satisfies $w(M_*) - w(\mathsf{Out}) \leq \epsilon$, where $M_* = \arg\max_{M \in \mathcal{M}} w(M)$ is the optimal set. In addition, the number of samples T used by the algorithm satisfies

$$T \leq \mathbf{H}_\epsilon \log\left(\frac{n}{\delta}\mathbf{H}_\epsilon\right),$$

where

$$\mathbf{H}_\epsilon = \sum_{e \in [n]} \min\left\{\frac{\mathrm{width}(\mathcal{B})^2}{\Delta_e^2}, \frac{n^2}{\epsilon^2}\right\}.$$

Lemma 9. *For any arm $e \in [n]$ and any round $t > n$ after initialization, if $\mathrm{rad}_t(e) \leq \max\left\{\frac{\Delta_e}{3\,\mathrm{width}(\mathcal{B})}, \frac{\epsilon}{n}\right\}$, then arm e will not be played on round t, i.e. $p_t \neq e$.*

Proof. If $\mathrm{rad}_t(e) \leq \frac{\Delta_e}{3\,\mathrm{width}(\mathcal{B})}$, then we can apply Lemma 12 which immediately gives that $p_t \neq e$. Hence, we only need to prove the case that $\mathrm{rad}_t(e) \leq \frac{\epsilon}{n}$. By the definition of p_t, we know that for each $i \in D_t$, we have $\mathrm{rad}_t(i) \leq \mathrm{rad}_t(e) \leq \frac{\epsilon}{n}$. Summing up all $i \in D_t$, we obtain

$$\mathrm{rad}_t(D_t) \leq \epsilon. \tag{14}$$

Next, we notice that the definition of M_t gives that $\bar{w}_t(M_t) = \max_{M \in \mathcal{M}} \bar{w}_t(M) \geq \bar{w}_t(M_t \oplus D_t)$. This means that

$$\bar{w}_t(D_t) = \bar{w}_t(M_t \oplus D_t) - \bar{w}_t(M_t) \leq 0. \tag{15}$$

Using the above inequalities, we have.

$$\begin{aligned} w_t^+(D_t) &= \bar{w}_t(D_t) + \mathrm{rad}_t(D_t) && (16)\\ &\leq \bar{w}_t(D_t) + \epsilon && (17)\\ &\leq \epsilon, && (18) \end{aligned}$$

where Eq. (16) follows from the definition of $w_t^+(\cdot)$; Eq. (17) follows from Eq. (14); Eq. (18) holds since Eq. (15). □

Lemma 10. *If Algorithm 1 stops, then $w(M_*) - w(\mathsf{Out}) \leq \epsilon$.*

Proof. Suppose that $\mathsf{Out} \neq M_*$. Suppose that the algorithm stops on round T, we know that $\mathsf{Out} = M_T$. Consider the diff-set $D = M_* \ominus M_T$ and the diff-set D_T as defined in Step 15 of Algorithm 1. By Lemma Z, we see that

$$w_T^+(D_T) = \max_{C: C \prec M_T} w_T^+(C) \geq w_T^+(D). \tag{19}$$

On the other hand, the stopping condition of Algorithm 1 gives that

$$\begin{aligned}
\epsilon &\geq \tilde{w}_T(\tilde{M}_T) - \tilde{w}_T(M_T) \\
&= w_T^+(D_T) \geq w_T^+(D) \qquad (20)\\
&\geq w(D) = w(M_*) - w(M_T), \qquad (21)
\end{aligned}$$

where Eq. (20) follows from Eq. (19); Eq. (21) follows from the assumption that event ξ occurred.

□

Unless specified, we shall assume the random event ξ (defined in Lemma 5) holds in all the following proofs.

Lemma 11. *For any $t > 0$, if the algorithm terminates on round t, then $M_t = M_*$.*

Proof. Suppose $M_t \neq M_*$. Then $w(M_*) > w(M_t)$. Then, there exists $b \in \mathcal{B}$ such that $b \prec M_t$ and $w(b) > 0$. On the other hand, by Corollary 1, we have $w_t^+(b) > w(b)$. Hence $w_t^+(b) > 0$. This contradicts to the stopping condition of our algorithm. □

Lemma 12. *For any $t > 0$. If $e \in E_t^3$, then $p_t \neq e$.*

Proof. Suppose that $p_t = e$. Let $D = M_t^+ \ominus M_t$. Let $c, c_1, \ldots, c_k$ be decomposition of D on $\mathcal{B}$. And since $\mathcal{B}$ is a diff-set class, such decomposition exists. Assume, without loss of generality, that $e \in c$.
By Lemma Y, we know that

$$D_+ = c_+ \cup c_1^+ \cup \ldots \cup c_k^+ \quad \text{and} \quad D_- = c_- \cup c_1^- \cup \ldots c_k^-. \tag{22}$$

We also denote $K = \text{width}(\mathcal{B})$.
Case (1). Suppose that $c \in \mathcal{B}_{\mathsf{opt}}$. Then $w(c) > 0$. Since $e \in E_t^3$, we have $\text{rad}_t(e) \leq \frac{1}{3}\Delta_e \leq \frac{1}{3K}w(c)$. In addition, $\forall g \in c_t, g \neq e$, $\text{rad}_t(g) \leq \text{rad}_t(e) \leq \frac{1}{3K}w(c)$. Hence, $\text{rad}_t(c) = \sum_{g \in c_t} \text{rad}_t(g) \leq \frac{|c_t|}{3K}w(c) \leq \frac{1}{3}w(c)$. Hence, $\bar{w}_t(c) \geq w(c) - \text{rad}_t(c) \geq \frac{2}{3}w(c) > 0$. This means that $\bar{w}_t(M_t \oplus c) = \bar{w}_t(M_t) + \bar{w}_t(c) > \bar{w}_t(M_t)$. Therefore, $M_t \neq \max_{M \in \mathcal{M}} \bar{w}_t(M)$. This contradicts to the definition of M_t.
Case (2). Suppose that $c_t \notin \mathcal{B}_{\mathsf{opt}}$. Then, one of the following mutually exclusive cases must hold.
Case (2.1). $(e \in M_* \wedge e \in c_+)$ or $(e \notin M_* \wedge e \in c_-)$.
Let the decomposition of $M_* \ominus (M_t \oplus D \ominus c)$ on $\mathcal{B}$ be $b, b_1, \ldots, b_l$, which exists due to $\mathcal{B}$ is a diff-set class. Assume wlog that $e \in b$. We write $b = (b_+, b_-)$. It is easy to see that $b \in \mathcal{B}_{\mathsf{opt}}$.
Define $\tilde{D} = (M_t \oplus D \ominus c) \ominus M_t$ and $D' = (M_t \oplus \tilde{D} \oplus b) \ominus M_t$. By Lemma 2, we know that $\tilde{D} = D \ominus c$ and $D' = \tilde{D} \oplus b$. We also write $\tilde{D} = (\tilde{D}_+, \tilde{D}_-)$ and $D' = (D'_+, D'_-)$. By definition, we have

$$\begin{aligned}
\tilde{D}_+ &= (D_+ \cup c_-) \backslash (D_- \cup c_+) \\
&= (D_+ \cup c_- \backslash D_-) \cap (D_+ \cup c_- \backslash c_+) \\
&= D_+ \cap (D_+ \backslash c_-) \\
&= D_+ \backslash c_+.
\end{aligned}$$

By the same method, we are able to show that $\tilde{D}_- = D_- \backslash c_-$. Therefore we have

$$\tilde{D}_+ \subseteq D_+ \quad \text{and} \quad \tilde{D}_- \subseteq D_-. \tag{23}$$

First, we show that $\mathrm{rad}_t(c) \leq \frac{1}{3}w(b)$. Since $e \in E_t^3$, $e \in b$ and $b \in \mathcal{B}_{\mathsf{opt}}$, we have $\mathrm{rad}_t(e) \leq \frac{1}{3}\Delta_e \leq \frac{1}{3K}w(b)$. In addition, $\forall g \in c, g \neq e$, $\mathrm{rad}_t(g) \leq \mathrm{rad}_t(e) \leq \frac{1}{3K}w(b)$. Hence,

$$\begin{aligned} \mathrm{rad}_t(c) &= \sum_{g \in c} \mathrm{rad}_t(g) \\ &\leq \frac{|c|}{3K} w(b) \\ &\leq \frac{1}{3} w(b). \end{aligned} \tag{24}$$

Now, we show that $\mathrm{rad}_t(\tilde{D}_+ \cap b_-) + \mathrm{rad}_t(\tilde{D}_- \cap b_+) + \leq \frac{1}{3}w(b)$. Since Eq. (23), we have $\forall g \in (\tilde{D}_+ \cap b_-) \cup (\tilde{D}_- \cap b_+), g \neq e$, $\mathrm{rad}_t(g) \leq \mathrm{rad}_t(e) \leq \frac{1}{3K}w(b)$. Note that $|\tilde{D}_+ \cap b_-| + |\tilde{D}_- \cap b_+| \leq |b_+| + |b_-| \leq K$. Hence,

$$\begin{aligned} \mathrm{rad}_t(\tilde{D}_+ \cap b_-) + \mathrm{rad}_t(\tilde{D}_- \cap b_+) &= \sum_{g \in (\tilde{D}_+ \cap b_-) \cup (\tilde{D}_- \cap b_+)} \mathrm{rad}_t(g) \\ &\leq \frac{K}{3K} w(b) \\ &\leq \frac{1}{3} w(b). \end{aligned} \tag{25}$$

Then, we have

$$\begin{aligned} \mathrm{rad}_t(D') - \mathrm{rad}_t(D) &= \mathrm{rad}_t(\tilde{D} \oplus b) - \mathrm{rad}_t(D) && (26) \\ &= \mathrm{rad}_t(\tilde{D}) + \mathrm{rad}_t(b) - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) - \mathrm{rad}_t(D) && (27) \\ &= \mathrm{rad}_t(D \ominus c) + \mathrm{rad}_t(b) - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) - \mathrm{rad}_t(D) && (28) \\ &= \mathrm{rad}_t(D) + \mathrm{rad}_t(c) + \mathrm{rad}_t(b) - 2\,\mathrm{rad}_t(D_+ \cap c_+) - 2\,\mathrm{rad}_t(D_- \cap c_-) \\ &\quad - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) - \mathrm{rad}_t(D) && (29) \\ &= \mathrm{rad}_t(D) + \mathrm{rad}_t(c) + \mathrm{rad}_t(b) - 2\,\mathrm{rad}_t(c_+) - 2\,\mathrm{rad}_t(c_-) \\ &\quad - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) - \mathrm{rad}_t(D) && (30) \\ &= \mathrm{rad}_t(b) - \mathrm{rad}_t(c) - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+), && (31) \end{aligned}$$

where Eq. (27) and Eq. (29) follow from Lemma 7, and Eq. (30) follows from Eq. (22).
By the definition of D, we have that $w_t^+(D) \geq w_t^+(D')$. This means that

$$\begin{aligned} \bar{w}_t(D) + \mathrm{rad}_t(D) &\geq \bar{w}_t(D') + \mathrm{rad}_t(D') && (32) \\ &= \bar{w}_t(D) - \bar{w}_t(c) + \bar{w}_t(b) + \mathrm{rad}_t(D'). && (33) \end{aligned}$$

By regrouping the above inequality, we have

$$\begin{aligned} \bar{w}_t(c) &\geq \bar{w}_t(b) + \mathrm{rad}_t(D') - \mathrm{rad}_t(D) && (34) \\ &= \bar{w}_t(b) + \mathrm{rad}_t(b) - \mathrm{rad}_t(c) - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) && (35) \\ &\geq w(b) - \mathrm{rad}_t(c) - 2\,\mathrm{rad}_t(\tilde{D}_+ \cap b_-) - 2\,\mathrm{rad}_t(\tilde{D}_- \cap b_+) && (36) \\ &> w(b) - \frac{1}{3}w(b) - \frac{2}{3}w(b) && (37) \\ &= 0, && (38) \end{aligned}$$

where Eq. (37) follows from Eq. (24) and Eq. (25).
This contradicts to the definition of M_t.
Case (2.2). $(e \in M_* \wedge e \in c_-)$ or $(e \notin M_* \wedge e \in c_+)$.
Let the decomposition of $M_* \ominus (M_t \oplus D)$ on $\mathcal{B}$ be $b, b_1, \ldots, b_l$. Assume wlog that $e \in b$. We write that $b = (b_+, b_-)$. Note that $b \in \mathcal{B}_{\mathsf{opt}}$ and hence $w(b) > 0$.
Define $D' = (M_t \oplus D \oplus b) \ominus M_t$. By Lemma 2, we know that $D' = D \oplus b$.
First, we show that $|D \backslash D'| \leq |b|$. Let $C = D \backslash D'$ and write $C = (C_+, C_-)$. We can bound $|C_+|$ as follows.

$$\begin{aligned}
C_+ &= D_+ \backslash D'_+ \\
&= D_+ \backslash \left((D_+ \cup b_+) \backslash (D_- \cup b_-) \right) \\
&= (D_+ \cap (D_- \cup b_-)) \cup (D_+ \backslash (D_+ \cup b_+)) \\
&= D_+ \cap b_-.
\end{aligned}$$

Hence, we have $|C_+| \leq |b_-|$. Then, we move to bounding $|C_-|$

$$\begin{aligned}
C_- &= D_- \backslash D'_- \\
&= D_- \backslash \left((D_- \cup b_-) \backslash (D_+ \cup b_+) \right) \\
&= (D_- \cap (D_+ \cup b_+)) \cup (D_- \backslash (D_- \cup b_-)) \\
&= D_- \cap b_+.
\end{aligned}$$

Thus $|C_-| \leq |b_+|$ and we proved that $|D \backslash D'| \leq |b|$.
Next, we show that $\mathrm{rad}_t(D \backslash D') \leq \frac{1}{3} w(b)$. Since $e \in E_t^3$, $e \in b$ and $b \in \mathcal{B}_{\mathsf{opt}}$, we have $\mathrm{rad}_t(e) \leq \frac{1}{3}\Delta_e \leq \frac{1}{3K} w(b)$. In addition, $\forall g \in (D \backslash D'), g \neq e$, $\mathrm{rad}_t(g) \leq \mathrm{rad}_t(e) \leq \frac{1}{3K} w(b)$. Note that $|D \backslash D'| \leq |b| \leq K$. Hence, $\mathrm{rad}_t(D \backslash D') = \sum_{g \in (D \backslash D')} \mathrm{rad}_t(g) \leq \frac{K}{3K} w(b) \leq \frac{1}{3} w(b)$.
We also note that

$$\begin{aligned}
w(D' \backslash D) - w(D \backslash D') &= w(D' \backslash D) + w(D' \cap D) - w(D \cap D') - w(D \backslash D') && (39) \\
&= w(D') - w(D) && (40) \\
&= w(b), && (41)
\end{aligned}$$

where we have repeatedly applied Lemma 4.
Then, we show that $w_t^+(D') > w_t^+(D)$.

$$\begin{aligned}
w_t^+(D') - w_t^+(D) &= \bar{w}_t(D') - \bar{w}_t(D) + \mathrm{rad}_t(D') - \mathrm{rad}_t(D) && (42) \\
&= \bar{w}_t(D' \backslash D) - \bar{w}_t(D \backslash D') + \mathrm{rad}_t(D' \backslash D) - \mathrm{rad}_t(D \backslash D') && (43) \\
&\geq w(D' \backslash D) - w(D \backslash D') - 2\,\mathrm{rad}_t(D \backslash D') && (44) \\
&= w(b) - 2\,\mathrm{rad}_t(D \backslash D') && (45) \\
&> w(b) - \frac{2}{3} w(b) && (46) \\
&= \frac{1}{3} w(b) > 0, && (47)
\end{aligned}$$

where Eq. (43) follows from Lemma 6 and Eq. (44) follows from the fact that $\bar{w}_t(D' \backslash D) + \mathrm{rad}_t(D' \backslash D) \geq w(D' \backslash D)$ and that $\bar{w}_t(D \backslash D') + \mathrm{rad}_t(D \backslash D') \geq w(D \backslash D')$, under the random event ξ.
This contradicts to the fact that D is chosen on round t. □

5 Proof of Lower Bounds

Lemma 13.

$$\Delta_e = \min_{b: e \in b, b \in \mathcal{B}_{\mathsf{opt}}} w(b).$$

Proof. □

Proof. Fix $\delta > 0$, $\vec{w} = \{w(1), \ldots, w(n)\}$ and a δ-correct policy $\mathbb{A}$. For each $e \in [n]$, assume that the reward distribution is given by $\varphi_e = \mathcal{N}(w(e), 1)$. For any $e \in [n]$, let T_e denote the number of trials of arm e used by algorithm $\mathbb{A}$. In the rest of the proof, we will show that for any $e \in [n]$, the number of trials of arm e is lower-bounded by

$$\mathbb{E}[T_e] \geq \frac{1}{16\Delta_e^2} \log(1/4\delta). \tag{48}$$

Notice that the theorem follows immediately by summing up Eq. (48) for all $e \in [n]$.
Fix an arm $e \in [n]$. We now focus on proving Eq. (48). Consider two hypothesis H_0 and H_1. Under hypothesis H_0, all reward distributions are same with our assumption before

$$H_0 : \varphi_l = \mathcal{N}(w(l), 1) \quad \text{for all } l \in [n].$$

Under hypothesis H_1, we change the means of reward distributions such that

$$H_1 : \varphi_e = \begin{cases} \mathcal{N}(w(e) - 2\Delta_e, 1) & \text{if } e \in M_* \\ \mathcal{N}(w(e) + 2\Delta_e, 1) & \text{if } e \notin M_* \end{cases} \quad \text{and } \varphi_l = \mathcal{N}(w(l), 1) \quad \text{for all } l \neq e.$$

Define M_e be the "next-to-optimal" set as follows

$$M_e = \begin{cases} \arg\max_{M \in \mathcal{M}: e \in M} w(M) & \text{if } e \notin M_*, \\ \arg\max_{M \in \mathcal{M}: e \notin M} w(M) & \text{if } e \in M_*. \end{cases}$$

By definition of Δ_e, we know that $w(M_*) - w(M_e) = \Delta_e$.
Let $\vec{w}_0$ and $\vec{w}_1$ be expected reward vectors under H_0 and H_1 respectively. Notice that $w_0(M_*) - w_0(M_e) = \Delta_e > 0$. On the other hand, $w_1(M_*) - w_1(M_e) = -\Delta < 0$. This means that under H_1, M_* is not the optimal set. For $l \in \{0, 1\}$, we use $\mathbb{E}_l$ and $\Pr_l$ to denote the expectation and probability, respectively, under the hypothesis H_l.
Define $\theta = 4\delta$. Define

$$t_e^* = \frac{1}{16\Delta_e^2} \log\left(\frac{1}{\theta}\right). \tag{49}$$

Recall that T_e denotes the total number of samples of arm e. Define the event $\mathcal{A} = \{T_e \leq 4t_e^*\}$.
First, we show that $\Pr_0[\mathcal{A}] \geq 3/4$. This can be proved by Markov inequality as follows.

$$\begin{aligned} \Pr_0[T_e > 4t_e^*] &\leq \frac{\mathbb{E}_0[T_e]}{4t_e^*} \\ &= \frac{t_e^*}{4t_e^*} = \frac{1}{4}. \end{aligned}$$

Let $X_1, \ldots, X_{T_e}$ denote the sequence of reward outcomes of arm e. We define $K_t(e)$ as the sum of outcomes of arm e up to round t, i.e. $K_t(e) = \sum_{i \in [t]} X_i$. Next, we define the event

$$\mathcal{C} = \left\{ \max_{1 \leq t \leq 4t_e^*} |K_t(e) - t \cdot w(e)| < \sqrt{t_e^* \log(1/\theta)} \right\}.$$

We now show that $\Pr_0[\mathcal{C}] \geq 3/4$. First, notice that $K_t(e) - p_e t$ is a martingale under H_0. Then, by Kolmogorov's inequality, we have

$$\begin{aligned}
\Pr_0\left[\max_{1\leq t\leq 4t_e^*} |K_t(e) - t\cdot w(e)| \geq \sqrt{t_e^* \log(1/\theta)}\right] &\leq \frac{\mathbb{E}_0[(K_{4t_e^*}(e) - 4w(e)t_e^*)^2]}{t_e^* \log(1/\theta)} \\
&= \frac{4t_e^*}{t_e^* \log(1/\theta)} \\
&< \frac{1}{4},
\end{aligned}$$

where the second inequality follows from the fact that $\mathbb{E}_0[(K_{4t_e^*}(e) - 4w(e)t_e^*)^2] = 4t_e^*$; the last inequality follows since $\theta < e^{-16}$.
Then, we define the event $\mathcal{B}$ as the event that the algorithm eventually returns M_*, i.e.

$$\mathcal{B} = \{\mathsf{Out} = M_*\}.$$

Since the probability of error of the algorithm is smaller than $\delta < 1/4$, we have $\Pr_0[\mathcal{B}] \geq 3/4$. Define $\mathcal{S}$ be $\mathcal{S} = \mathcal{A}\cap\mathcal{B}\cap\mathcal{C}$. Then, by union bound, we have $\Pr_0[\mathcal{S}] \geq 1/4$.
Now, we show that if $\mathbb{E}_0[T_e] \leq t_e^*$, then $\Pr_1[\mathcal{B}] \geq \delta$. Let W be the history of the sampling process until the algorithm stops (including the sequence of arms chosen at each time and the sequence of observed outcomes). Define the likelihood function L_l as

$$L_l(w) = p_l(W = w),$$

where p_l is the probability density function under hypothesis H_l. Let K be the shorthand of $K_e(T_e)$.
Assume that the event $\mathcal{S}$ occurred. We will bound the likelihood ratio $L_1(W)/L_0(W)$ under this assumption. To do this, we divide our analysis into two different cases.
Case (1): $e \notin M_*$. In this case, the reward distribution of arm e under H_1 is a Gaussian distribution with mean $p_e + 2\Delta_e$ and variance 1. Recall that the probability density function of a Gaussian distribution with mean μ and variance σ^2 is given by $\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$. Hence, we have

$$\begin{aligned}
\frac{L_1(W)}{L_0(W)} &= \prod_{i=1}^{T_e} \exp\left(\frac{-(X_i - w(e) - 2\Delta_e)^2 + (X_i - w(e))^2}{2}\right) \\
&= \prod_{i=1}^{T_e} \exp\left(\Delta_e(2X_i - 2w(e)) - 2\Delta_e^2\right) \\
&= \exp\left(\Delta_e(2K - 2w(e)T_e) - 2\Delta_e^2 T_e\right) \\
&= \exp\left(\Delta_e(2K - 2w(e)T_e)\right)\exp(-2\Delta_e^2 T_e). \qquad (50)
\end{aligned}$$

Next, we bound each individual term on the right-hand side of Eq. (50). We begin with bounding the second term of Eq. (50)

$$\begin{aligned}
\exp(-2\Delta_e^2 T_e) &\geq \exp(-8\Delta_e^2 t_e^*) &(51)\\
&= \exp\left(-\frac{8}{16}\log(1/\theta)\right) &(52)\\
&= \theta^{1/2}, &(53)
\end{aligned}$$

where Eq. (79) follows from the assumption that event $\mathcal{S}$ occurred, which implies that event $\mathcal{A}$ occurred and therefore $T_e \leq 4t_e^*$; Eq. (80) follows from the definition of t_e^*.

Then, we bound the first term on the right-hand side of Eq. (50) as follows

$$\exp\left(\Delta_e(2K - 2w(e)T_e)\right) \geq \exp\left(-2\Delta_e\sqrt{t_e^* \log(1/\theta)}\right) \tag{54}$$

$$= \exp\left(-\frac{2}{\sqrt{4}}\log(1/\theta)\right) \tag{55}$$

$$= \theta^{1/2}, \tag{56}$$

where Eq. (82) follows from the assumption that event $\mathcal{S}$ occurred, which implies that event $\mathcal{C}$ and therefore $|2K - 2w(e)T_e| \leq \sqrt{t_e^* \log(1/\theta)}$; Eq. (83) follows from the definition of t_e^*.
Combining Eq. (81) and Eq. (84), we can bound $L_1(W)/L_0(W)$ for this case as follows

$$\frac{L_1(W)}{L_0(W)} \geq \theta. \tag{57}$$

(End of Case (1).)
Case (2): $e \in M_*$**.** In this case, we know that the mean reward of arm e under H_1 is $p_e - 2\Delta$. Therefore, the likelihood ratio $L_1(W)/L_0(W)$ is given by

$$\begin{aligned}\frac{L_1(W)}{L_0(W)} &= \prod_{i=1}^{T_e} \exp\left(\frac{-(X_i - w(e) + 2\Delta_e)^2 + (X_i - w(e))^2}{2}\right) \\ &= \prod_{i=1}^{T_e} \exp\left(\Delta_e(2w(e) - 2X_i) - 2\Delta_e^2\right) \\ &= \exp\left(\Delta_e(2w(e)T_e - 2K)\right)\exp(-2\Delta_e^2 T_e). \end{aligned} \tag{58}$$

Notice that the right-hand side of Eq. (58) differs from Eq. (50) only in its first term. Now, we bound the first term as follows

$$\exp\left(\Delta_e(2K - 2w(e)T_e)\right) \geq \exp\left(-2\Delta_e\sqrt{t_e^* \log(1/\theta)}\right) \tag{59}$$

$$= \exp\left(-\frac{2}{4}\log(1/\theta)\right) \tag{60}$$

$$= \theta^{1/2}, \tag{61}$$

where the inequalities hold due to reasons similar to Case (1): Eq. (59) follows from the assumption that event $\mathcal{S}$ occurred, which implies that event $\mathcal{C}$ and therefore $|2K - 2w(e)T_e| \leq \sqrt{t_e^* \log(1/\theta)}$; Eq. (60) follows from the definition of t_e^*.
Combining Eq. (81) and Eq. (84), we can obtain the same bound of $L_1(W)/L_0(W)$ as in Eq. (57), i.e. $L_1(W)/L_0(W) \geq \theta$.
(End of Case (2).)
At this point, we have proved that, if the event $\mathcal{S}$ occurred, then the bound of likelihood ratio Eq. (57) holds, i.e. $\frac{L_1(W)}{L_0(W)} \geq \theta$. Hence, we have

$$\begin{aligned}\frac{L_1(W)}{L_0(W)} &\geq \theta \\ &= 4\delta. \end{aligned} \tag{62}$$

Define 1_S as the indicator variable of event $\mathcal{S}$, i.e. $1_S = 1$ if and only if $\mathcal{S}$ occurs and otherwise $1_S = 0$.

Then, we have

$$\frac{L_1(W)}{L_0(W)} 1_S \geq 4\delta 1_S$$

holds regardless the occurrence of event $\mathcal{S}$. Therefore, we can obtain

$$\begin{aligned}
\Pr_1[\mathcal{B}] \geq \Pr_1[\mathcal{S}] &= \mathbb{E}_1[1_S] \\
&= \mathbb{E}_0\left[\frac{L_1(W)}{L_0(W)} 1_S\right] \\
&\geq 4\delta \mathbb{E}_0[1_S] \\
&= 4\delta \Pr_0[\mathcal{S}] > \delta.
\end{aligned}$$

Now we have proved that, if $\mathbb{E}_0[T_e] \leq t_e^*$, then $\Pr_1[\mathcal{B}] > \delta$. This means that, if $\mathbb{E}_0[T_e] \leq t_e^*$, algorithm $\mathbb{A}$ will choose M_* as the output with probability at least δ, under hypothesis H_1. However, under H_1, we have shown that M_* is not the optimal set since $w_1(M_e) > w_1(M_*)$. Therefore, algorithm $\mathbb{A}$ has a probability of error larger than δ under H_1. This contradicts to the assumption that algorithm $\mathbb{A}$ is a δ-correct algorithm. Hence, we must have $\mathbb{E}_0[T_e] > t_e^* = \frac{1}{16\Delta_e^2}\log(1/4\delta)$.

□

Proof. Fix $\delta > 0$, $\vec{w} \in \mathbb{R}^n$, diff-set $b = (b_+, b_-)$ and a δ-correct algorithm $\mathbb{A}$. Assume that $\varphi_e(e) = \mathcal{N}(w(e), 1)$ for all $e \in [n]$.

We define three hypotheses H_0, H_1 and H_2. Under hypothesis H_0, the reward distribution

$$H_0 : \varphi_l = \mathcal{N}(w(l), 1) \quad \text{for all } l \in [n].$$

Under hypothesis H_1, the mean reward of each arm is given by

$$H_1 : \varphi_e = \begin{cases} \mathcal{N}\left(w(e) + 2\frac{w(b)}{|b_-|}, 1\right) & \text{if } e \in b_-, \\ \mathcal{N}(w(e), 1) & \text{if } e \notin b_-. \end{cases}$$

And under hypothesis H_2, the mean reward of each arm is given by

$$H_2 : q_e = \begin{cases} \mathcal{N}\left(w(e) - 2\frac{w(b)}{|b_-|}, 1\right) & \text{if } e \in b_+, \\ \mathcal{N}(w(e), 1) & \text{if } e \notin b_+. \end{cases}$$

Since $b \in \mathcal{B}_{\mathsf{opt}}$, it is clear that $\neg b \prec M_*$. Hence we define $M = M_* \ominus b$. Let w_0, w_1 and w_2 be the expected reward vectors under H_0, H_1 and H_2 respectively. It is easy to check that $w_1(M_*) - w_1(M) = -w(b) < 0$ and $w_2(M_*) - w_2(M) = -w(b) < 0$. This means that under H_1 or H_2, M_* is not the optimal set. Further, for $l \in \{0, 1, 2\}$, we use $\mathbb{E}_l$ and $\Pr_l$ to denote the expectation and probability, respectively, under the hypothesis H_l. In addition, let W be the history of the sampling process until algorithm $\mathbb{A}$ stops. Define the likelihood function L_l as

$$L_l(w) = p_l(W = w),$$

where p_l is the probability density function under H_l.

Define $\theta = 4\delta$. Let T_{b_-} and T_{b_+} denote the number of trials of arms belonging to b_- and b_+, respectively. In the rest of the proof, we will bound $\mathbb{E}_0[T_{b_-}]$ and $\mathbb{E}_0[T_{b_+}]$ individually.

Part (1): Lower bound of $\mathbb{E}_0[T_{b_-}]$. In this part, we will show that $\mathbb{E}_0[T_{b_-}] \geq t_{b_-}^*$, where we define $t_{b_-}^* = \frac{|b_-|^2}{16w(b)^2}\log(1/\theta)$.

Consider the complete sequence of sampling process by algorithm $\mathbb{A}$. Formally, let $W = \{(\tilde{I}_1, \tilde{X}_1), \ldots, (\tilde{I}_T, \tilde{X}_T)\}$

be the sequence of all trials by algorithm $\mathbb{A}$, where $\tilde{I}_i$ denotes the arm played in i-th trial and $\tilde{X}_i$ be the reward outcome of i-th trial. Then, consider the subsequence W_1 of W which consists all the trials of arms in b_-. Specifically, we write $W = \{(I_1, X_1), \ldots, (I_{T_{b_-}}, X_{T_{b_-}})\}$ such that W_1 is a subsequence of W and $I_i \in b_-$ for all i.

Next, we define several random events in a way similar to the proof of Theorem 2. Define event $\mathcal{A}_1 = \{T_{b_-} \leq 4t^*_{b_-}\}$. Define event

$$\mathcal{C}_1 = \left\{ \max_{1 \leq t \leq 4t^*_{b_-}} \left| \sum_{i=1}^{t} X_i - \sum_{i=1}^{t} w(I_i) \right| < \sqrt{t^*_{b_-} \log(1/\theta)} \right\}.$$

Define event

$$\mathcal{B} = \{\mathsf{Out} = M_*\}. \tag{63}$$

Define event $\mathcal{S}_1 = \mathcal{A}_1 \cap \mathcal{B} \cap \mathcal{C}_1$. Then, we bound the probability of events $\mathcal{A}_1$, $\mathcal{B}$, $\mathcal{C}_1$ and $\mathcal{S}_1$ under H_0 using methods similar to Theorem 2. First, we show that $\Pr_0[\mathcal{A}_1] \geq 3/4$. This can be proved by Markov inequality as follows.

$$\begin{aligned} \Pr_0[T_{b_-} > 4t^*_{b_-}] &\leq \frac{\mathbb{E}_0[T_{b_-}]}{4t^*_{b_-}} \\ &= \frac{t^*_{b_-}}{4t^*_{b_-}} = \frac{1}{4}. \end{aligned}$$

Next, we show that $\Pr_0[\mathcal{C}_1] \geq 3/4$. Notice that the sequence $\left\{ \sum_{i=1}^{t} X_i - \sum_{i=1}^{t} p_{I_i} \right\}_{t \in [4t^*_{b_-}]}$ is a martingale. Hence, by Kolmogorov's inequality, we have

$$\begin{aligned} \Pr_0 \left[\max_{1 \leq t \leq 4t^*_{b_-}} \left| \sum_{i=1}^{t} X_i - \sum_{i=1}^{t} w(I_i) \right| \geq \sqrt{t^*_e \log(1/\theta)} \right] &\leq \frac{\mathbb{E}_0 \left[\left(\sum_{i=1}^{4t^*_{b_-}} X_i - \sum_{i=1}^{4t^*_{b_-}} w(I_i) \right)^2 \right]}{t^*_e \log(1/\theta)} \\ &= \frac{4t^*_{b_-}}{t^*_{b_-} \log(1/\theta)} \\ &< \frac{1}{4}, \end{aligned}$$

where the second inequality follows from the fact that all reward distributions have unit variance and hence $\mathbb{E}_0 \left[\left(\sum_{i=1}^{4t^*_{b_-}} X_i - \sum_{i=1}^{4t^*_{b_-}} p_{I_i} \right)^2 \right] = 4t^*_{b_-}$; the last inequality follows since $\theta < e^{-16}$. Last, since algorithm $\mathbb{A}$ is a δ-correct algorithm with $\delta < 1/4$. Therefore, it is easy to see that $\Pr_0[\mathcal{B}] \geq 3/4$. And by union bound, we have

$$\Pr_0[\mathcal{S}_1] \geq 1/4.$$

Now, we show that if $\mathbb{E}_0[T_{b_-}] \leq t^*_{b_-}$, then $\Pr_1[\mathcal{B}] \geq \delta$. Assume that the event $\mathcal{S}_1$ occurred. We bound the likelihood ratio $L_1(W)/L_0(W)$ under this assumption as follows

$$\begin{aligned} \frac{L_1(W)}{L_0(W)} &= \prod_{i=1}^{T_{b_-}} \exp \left(\frac{-\left(X_i - w(I_i) - \frac{2w(b)}{|b_-|} \right)^2 + (X_i - w(I_i)^2}{2} \right) \\ &= \prod_{i=1}^{T_{b_-}} \exp \left(\frac{w(b)}{|b_-|}(2X_i - 2w(I_i)) - \frac{2w(b)^2}{|b_-|^2} \right) \end{aligned}$$

$$
\begin{aligned}
&= \exp\left(\frac{w(b)}{|b_-|}\left(\sum_{i=1}^{T_{b_-}} 2X_i - 2w(I_i)\right) - \frac{2w(b)^2}{|b_-|^2}T_{b_-}\right) \\
&= \exp\left(\frac{w(b)}{|b_-|}\left(\sum_{i=1}^{T_{b_-}} 2X_i - 2w(I_i)\right)\right) \exp\left(-\frac{2w(b)^2}{|b_-|^2}T_{b_-}\right). \qquad (64)
\end{aligned}
$$

Then, we bound each term on the right-hand side of Eq. (64). First, we bound the second term of Eq. (64).

$$
\begin{aligned}
\exp\left(-\frac{2w(b)^2}{|b_-|^2}T_{b_-}\right) &\geq \exp\left(-\frac{2w(b)^2}{|b_-|^2}4t_b^*\right) &(65)\\
&= \exp\left(-\frac{8}{16}\log(1/\theta)\right) &(66)\\
&= \theta^{1/2}, &(67)
\end{aligned}
$$

where Eq. (65) follows from the assumption that events $\mathcal{S}_1$ and $\mathcal{A}_1$ occurred and therefore $T_{b_-} \leq 4t_{b_-}^*$; Eq. (66) follows from the definition of $t_{b_-}^*$. Next, we bound the first term of Eq. (64) as follows

$$
\begin{aligned}
\exp\left(\frac{w(b)}{|b_-|}\left(\sum_{i=1}^{T_{b_-}} 2X_i - 2w(I_i)\right)\right) &\geq \exp\left(-\frac{2w(b)}{|b_-|}\sqrt{t_b^* \log(1/\theta)}\right) &(68)\\
&= \exp\left(-\frac{2}{4}\log(1/\theta)\right) &(69)\\
&= \theta^{1/2}, &(70)
\end{aligned}
$$

where Eq. (68) follows since event $\mathcal{S}_1$ and $\mathcal{C}_1$ occurred and therefore $|2K - 2p_eT_e| \leq \sqrt{t_e^* \log(1/\theta)}$; Eq. (69) follows from the definition of $t_{b_-}^*$.

Hence, if event $\mathcal{S}_1$ occurred, we can bound the likelihood ratio as follows

$$
\frac{L_1(W)}{L_0(W)} \geq \theta = 4\delta. \qquad (71)
$$

Let 1_{S_1} denote the indicator variable of event $\mathcal{S}_1$. Then, we have $\frac{L_1(W)}{L_0(W)}1_{S_1} \geq 4\delta 1_{S_1}$. Therefore, we can bound $\Pr_1[\mathcal{B}]$ as follows

$$
\begin{aligned}
\Pr{}_1[\mathcal{B}] \geq \Pr{}_1[\mathcal{S}_1] &= \mathbb{E}_1[1_{S_1}] \\
&= \mathbb{E}_0\left[\frac{L_1(W)}{L_0(W)}1_{S_1}\right] \\
&\geq 4\delta\mathbb{E}_0[1_{S_1}] \\
&= 4\delta \Pr{}_0[\mathcal{S}_1] > \delta. \qquad (72)
\end{aligned}
$$

This means that, if $\mathbb{E}_0[T_{b_-}] \leq t_{b_-}^*$, then, under H_1, the probability of algorithm $\mathbb{A}$ returning M_* as output is at least δ. But M_* is not the optimal set under H_1. Hence this contradicts to the assumption that $\mathbb{A}$ is a δ-correct algorithm. Hence we have proved that

$$
\mathbb{E}_0[T_{b_-}] \geq t_{b_-}^* = \frac{|b_-|^2}{16w(b)^2}\log(1/4\delta). \qquad (73)
$$

(End of Part (1).)

Part (2): Lower bound of $\mathbb{E}_0[T_{b_+}]$. In this part, we will show that $\mathbb{E}_0[T_{b_+}] \geq t_{b_+}^*$, where we define

$t^*_{b_+} = \frac{|b_+|^2}{16w(b)^2}\log(1/\theta)$. The arguments used in this part are similar to that of Part (1). Hence, we will omit the redundant parts and highlight the differences.
Recall that we have defined that W to be the history of all trials by algorithm $\mathbb{A}$. We define W be the subsequence of $\tilde{S}$ which contains the trials of arms belonging to b_+. We write $S_2 = \{(J_1, Y_1), \ldots, (J_{T_{b_+}}, Y_{T_{b_+}})\}$, where J_i is i-th played arm in sequence S_2 and Y_i is the associated reward outcome.
We define the random events $\mathcal{A}_2$ and $\mathcal{C}_2$ similar to Part (1). Specifically, we define

$$\mathcal{A}_2 = \{T_{b_+} \le 4t^*_{b_+}\} \quad \text{and} \quad \mathcal{C}_2 = \left\{ \max_{1\le t\le 4t^*_{b_+}} \left| \sum_{i=1}^{t} Y_i - \sum_{i=1}^{t} w(J_i) \right| < \sqrt{t^*_{b_+}\log(1/\theta)} \right\}.$$

Using the similar arguments, we can show that $\Pr_0[\mathcal{A}_2] \ge 3/4$ and $\Pr_0[\mathcal{C}_2] \ge 3/4$. Define event $\mathcal{S}_2 = \mathcal{A}_2 \cap \mathcal{B} \cap \mathcal{C}_2$, where $\mathcal{B}$ is defined in Eq. (63). By union bound, we see that

$$\Pr_0[\mathcal{S}_2] \ge 1/4.$$

Then, we show that if $\mathbb{E}_0[T_{b_+}] \le t^*_{b_+}$, then $\Pr_2[\mathcal{B}] \ge \delta$. We bound likelihood ratio $L_2(W)/L_0(W)$ under the assumption that $\mathcal{S}_2$ occurred as follows

$$\begin{aligned}
\frac{L_2(W)}{L_0(W)} &= \prod_{i=1}^{T_{b_+}} \exp\left(\frac{-\left(Y_i - w(J_i)) + \frac{2w(b)}{|b_-|}\right)^2 + (Y_i - w(J_i))^2}{2} \right) \\
&= \prod_{i=1}^{T_{b_+}} \exp\left(\frac{w(b)}{|b_+|}(2w(J_i) - 2Y_i) - \frac{2w(b)^2}{|b_+|^2} \right) \\
&= \exp\left(\frac{w(b)}{|b_+|} \left(\sum_{i=1}^{T_{b_+}} 2w(J_i) - 2Y_i \right) - \frac{2w(b)^2}{|b_+|^2} T_{b_+} \right) \\
&= \exp\left(\frac{w(b)}{|b_+|} \left(\sum_{i=1}^{T_{b_+}} 2w(J_i) - 2Y_i \right) \right) \exp\left(-\frac{2w(b)^2}{|b_+|^2} T_{b_+} \right) \\
&\ge \theta \qquad (74) \\
&= 4\delta,
\end{aligned}$$

where Eq. (74) can be obtained using same method as in Part (1) as well as the assumption that $\mathcal{S}_2$ occurred.
Next, similar to the derivation in Eq. (72), we see that

$$\Pr_2[\mathcal{B}] \ge \Pr_2[\mathcal{S}_2] = \mathbb{E}_2[1_{S_2}] = \mathbb{E}_0\left[\frac{L_2(W)}{L_0(W)} 1_{S_2}\right] \ge 4\delta\mathbb{E}_0[1_{S_2}] > \delta,$$

where 1_{S_2} is the indicator variable of event $\mathcal{S}_2$. Therefore, we see that if $\mathbb{E}_0[T_{b_+}] \le t^*_{b_+}$, then, under H_2, the probability of algorithm $\mathbb{A}$ returning M_* as output is at least δ, which is not the optimal set under H_2. This contradicts to the assumption that algorithm $\mathbb{A}$ is a δ-correct algorithm. In sum, we have proved that

$$\mathbb{E}_0[T_{b_+}] \ge t^*_{b_+} = \frac{|b_+|^2}{16w(b)^2}\log(1/4\delta). \qquad (75)$$

(End of Part (2))
Finally, we combine the results from both parts, i.e. Eq. (73) and Eq. (75). We obtain

$$\mathbb{E}_0[T_b] = \mathbb{E}_0[T_{b_-}] + \mathbb{E}_0[T_{b_+}]$$

$$\geq \frac{|b_+|^2 + |b_-|^2}{16 w(b)^2} \log(1/4\delta)$$
$$\geq \frac{|b|^2}{32 w(b)^2} \log(1/4\delta).$$

□

Now we prove a lower bound on the probability of error in the fixed budget setting. In particular, we show that for any expected rewards $\{w(1), \ldots, w(n)\}$ and any fixed budget algorithm $\mathbb{A}$ for pure exploration combinatorial bandit problem with feasible sets $\mathcal{M}$. We show that one can slightly modify the vector $\{w(e)\}_{e \in [n]}$ to construct another vector $\{\tilde{w}(1), \ldots, \tilde{w}(n)\}$ such that the probability of error of the fixed algorithm on a `ExpCMAB` problem with expected rewards given by $\tilde{w}$ is at least $\Omega(\exp(n/\mathbf{H}(w)))$.

Theorem 6. *Given a vector $\{w(1), \ldots, w(n)\}$, a budget $T > 0$ and a collection of feasible sets $\mathcal{M} \subseteq 2^{[n]}$. Let $\mathbb{A}$ be an arbitrary algorithm for $\mathcal{M}$-`ExpCMAB` problem which uses at most T samples. There exists a vector $\{\tilde{w}(1), \ldots, \tilde{w}(n)\}$ such that $\mathbf{H}(\tilde{w}) \leq 2\mathbf{H}(w)$ and satisfies the following property. Consider the bandit problem with reward distributions defined by $\varphi_e = \mathcal{N}(\tilde{w}(e), 1)$ for all $e \in [n]$, where $\mathcal{N}(\mu, \sigma^2)$ denotes Gaussian distribution with mean μ and variance σ^2. The probability of error of $\mathbb{A}$ on this bandit problem satisfies*

$$\Pr[\mathsf{Out} \neq M_*] \geq \exp\left(-\frac{T}{\mathbf{H}(w)}\right),$$

where Out is the output of $\mathbb{A}$ and $M_* = \arg\max_{M \in \mathcal{M}} w(M)$ is the optimal set. In addition, vector $\{\tilde{w}(1), \ldots, \tilde{w}(n)\}$ differs from vector $\{w(1), \ldots, w(n)\}$ on exactly one index.

Proof. Fix $\mathcal{M} \subseteq 2^{[n]}$, $w(e)$ for all $e \in [n]$ and a fixed budget algorithm $\mathbb{A}$ for $\mathcal{M}$-`ExpCMAB` problem. Let $\sigma(1), \ldots, \sigma(n)$ be a permutation of $1, 2, \ldots, n$ such that $\Delta_{\sigma(1)} \leq \Delta_{\sigma(2)} \ldots \leq \Delta_{\sigma(n)}$. Define $L' = \arg\max_{i \in [n]} i/\Delta^2_{\sigma(i)}$ and $L = \sigma(L')$.
Then, we construct hypothesis H_0 as follows

$$H_0 : \varphi_e = \mathcal{N}(w(e), 1) \quad \text{for all } e \in [n].$$

We define random event $\mathcal{C}$ as follows.
We show that $\Pr_0[\mathcal{C}] \geq 1/2$.
We define random variables X, Y, Z as follows

$$X = \arg\min_{i \in [L] \setminus \mathsf{Out}} T_i, \quad Y = \arg\min_{i \in [L] \cap \mathsf{Out}} T_i \quad \text{and } Z = \arg\min_{i \in [L]} T_i,$$

where, for convenience, if $[L] \setminus \mathsf{Out} = \emptyset$, we set $X = 0$; and if $[L] \cap \mathsf{Out} = \emptyset$, we set $Y = 0$. By definition, we see that $X \neq Y$, $Z \in \{X, Y\}$ and $Z \neq 0$. Now, by summing up all possible values of X, Y and Z, we have

$$1/2 < \Pr_0[\mathcal{C}] = \sum_{\substack{x \in \{0, \ldots, L\} \\ y \in \{0, \ldots, L\} \\ x \neq y, z \in \{x, y\}}} \Pr_0[\mathcal{C} \cap \{X = x, Y = y, Z = z\}].$$

Since a maximal is larger than an average, we see that there exists x, y, z such that $x \neq y, z \in \{x, y\}$ and

$$\Pr_0[\mathcal{C} \cap \{X = x, Y = y, Z = z\}] \geq \frac{1}{4L(L+1)}. \tag{76}$$

We point out that x, y and z are deterministic and only depends $\mathbb{A}$, w and $\mathcal{M}$. Now, depending on the value of x, y and z, we divide our analysis into two cases.

Case (1): $(z = x \wedge x \in M_*)$ **or** $(z = y \wedge y \notin M_*)$**.** Eq. (76) implies that

$$\begin{aligned}\Pr_0[\{X = x, Y = y, Z = z\}] &\geq \Pr_0[\mathcal{C} \cap \{X = x, Y = y, Z = z\}] \\ &\geq \frac{1}{4L(L+1)} \geq G.\end{aligned}$$

First, let us assume that $z = x$ and $x \in M_*$. By definition, we have $X \notin \mathsf{Out}$. Notice that x belong M_*. Therefore the event that $X = x$ and the assumption that $x \in M_*$ imply that $\mathsf{Out} \neq M_*$. This means that, if $z = x$ and $x \in M_*$, then $\Pr_0[\mathsf{Out} \neq M_*] \geq \Pr_0[X = x] \geq G$.
Next, we assume that $z = y$ and $y \notin M_*$. Notice that $Y \in \mathsf{Out}$ and $y \notin M_*$. Hence, the event $Y = y$ and the assumption that $y \notin M_*$ imply that $\mathsf{Out} \neq M_*$. Therefore, if $z = y$ and $y \notin M_*$, then $\Pr_0[\mathsf{Out} \neq M_*] \geq \Pr_0[Y = y] \geq G$.
Therefore, we proved that, in Case (1), the probability of error of algorithm $\mathbb{A}$ is larger than G under H_0.
Case (2): $(z = x \wedge x \notin M_*)$ **or** $(z = y \wedge y \in M_*)$**.** By definition, Z is the arm with smallest number of samples among arms in $[L]$ and algorithm $\mathbb{A}$ uses at most T samples. Therefore, we have

$$T_Z \leq \frac{T}{L}. \tag{77}$$

Then, we consider two cases separately.
Case (2.1): $(z = x \wedge x \notin M_*)$**.** We construct hypothesis H_1 as follows

$$H_1 : \varphi_x = \mathcal{N}(w(x) + \Delta_L + \varepsilon, 1) \quad \text{and} \quad \varphi_e(e) = \mathcal{N}(w(e), 1) \quad \text{for all } e \neq y.$$

Notice that, by the choice of L, we have $\Delta_x \geq \Delta_L$. Hence we see that $w_1(M_x) = w_0(M_x) + \Delta_L \geq w_0(M_x) + \Delta_x = w_0(M_*) = w_1(M_*)$. Therefore, under H_1, M_* is not the optimal set. Now we bound the likelihood ratio $L_1(W)/L_0(W)$ as follows

$$\begin{aligned}\frac{L_1(W)}{L_0(W)} &= \prod_{i=1}^{T_Z} \exp\left(\frac{-(X_i - w(x) - \Delta_L)^2 + (X_i - w(x))^2}{2}\right) \\ &= \prod_{i=1}^{T_Z} \exp\left(\Delta_L(X_i - w(x)) - \Delta_L^2\right) \\ &= \exp\left(\Delta_L(K - T_Z w(x)) - \Delta_L^2 T_Z\right) \\ &= \exp\left(\Delta_L(K - T_Z w(x))\right)\exp(-\Delta_L^2 T_Z). \end{aligned} \tag{78}$$

Then we analyze the right-hand side of Eq. (78) as follows

$$\begin{aligned}\exp(-\Delta_Z^2 T_Z) &\geq \exp(-\Delta_Z^2 T/L) && (79)\\ &= \exp\left(-\frac{8}{16}\log(1/\theta)\right) && (80)\\ &= \theta^{1/2}, && (81)\end{aligned}$$

where Eq. (79) follows from the assumption that event $\mathcal{S}$ occurred, which implies that event $\mathcal{A}$ occurred and therefore $T_e \leq 4t_e^*$; Eq. (80) follows from the definition of t_e^*.
Then, we bound the first term on the right-hand side of Eq. (50) as follows

$$\begin{aligned}\exp\left(\Delta_e(2K - 2p_e T_e)\right) &\geq \exp\left(-2\Delta_e\sqrt{t_e^* \log(1/\theta)}\right) && (82)\\ &= \exp\left(-\frac{2}{\sqrt{4}}\log(1/\theta)\right) && (83)\end{aligned}$$

$$= \theta^{1/2}, \tag{84}$$

where Eq. (82) follows from the assumption that event $\mathcal{S}$ occurred, which implies that event $\mathcal{C}$ and therefore $|2K - 2p_e T_e| \le \sqrt{t_e^* \log(1/\theta)}$; Eq. (83) follows from the definition of t_e^*.

Case (2.2): $y \notin M_*$**.** By definition of y, we see that the event $Y = y$ implies that $y \notin \mathsf{Out}$ and therefore $\mathsf{Out} \neq M_*$. On the other hand, using Eq. (76), we have

$$\Pr_0[Y = y] \ge \Pr_0[\mathcal{C} \cap \{Y = y, Z = 0\}] = \frac{1}{2(L+1)^2}.$$

This gives that $\Pr_0[\mathsf{Out} \neq M_*] \ge \frac{1}{2(L+1)^2} \ge A$.

□

6 Technical Lemmas

Lemma 14 (Basis exchange property)**.** *AA*

References