Multi-armed Bandits with Bounded Rewards: a Short Survey and Kullback-Leibler Maillard Sampling

Hao Qin

November 2023

1 Introduction

Problem: Multi-armed Bandits. Multi-armed Bandits (MAB) is a Machine Learning framework where a learning agent learns to make a series of decisions to maximize its reward. The learning agent is given a set of actions $\mathcal{A}$; each arm $a \in \mathcal{A}$ is associated with a reward distribution ν_a with expected mean μ_a and the optimal arm has expected reward $\mu_{\max} := \max_{a\in\mathcal{A}} \mu_a$. Initially, the learning agent does not know the arms' reward distributions $\mathcal{V} = (\nu_a)_{a\in\mathcal{A}}$. At each time step t, the learning agent takes an action a_t (also called pulling an arm a_t) and receives a reward r_{t,a_t} from the arm-associated distribution ν_{a_t}. The learning agent would like to maximize its cumulative reward in a time horizon of T by interacting and collecting information from the environment. To measure an algorithm's performance, we use the gap between the cumulative reward obtained by the algorithm to best the reward returned by executing the optimal arm in each round, called Regret. An algorithm's Pseudo-regret $\text{Regret}(T)$ is the expected difference by removing the random noise brought by the reward distribution. Precisely, the Pseudo-regret of an algorithm π on the Bandit instance $\mathcal{B}$ is

$$\text{Regret}_{\mathcal{B}}^{\pi}(T) = \sum_{t=1}^{T} \Delta_{a_t} = \sum_{t=1}^{T} \mu_{\max} - \mu_{a_t},$$

where $\Delta_{a_t} := \mu_{\max} - \mu_a$ is the suboptimal gap and $\mathcal{B} := (T, \mathcal{A}, \mathcal{V})$ represents the bandit instance specified by T, $\mathcal{A}$ and $\mathcal{V}$ (More details in Section2). Usually, we also use $\text{Regret}(T)$ to replace $\text{Regret}_{\mathcal{B}}^{\pi}(T)$ for short if we have set the bandit instance and the algorithm clearly and unchanged. The time horizon T corresponds to the total number of users exposed to the advertisement. Over a designated time T, at every moment t, the immediate reward r_{t,a_t} is 1 if the user purchases the promoted product. Otherwise, the reward is 0. We seek a strategy to minimize the cumulative regret. The challenge lies between exploring new arms or exploiting the current best arm. If we explore too many arms, we will suffer a suboptimality since the suboptimal arm has been selected considerably. If we always stick to the empirical best, we might lose a chance to identify the real optimal arm. This framework has been used in many applications. Below, we consider an example:

Example: Online Advertising Campaign. Imagine a company promoting a product on digital platforms, aiming to captivate customers as much as possible. There are plenty of website layouts. For example, a visually stunning website might fall short if it is not user-friendly or does not provide a seamless browsing experience. Different website layouts will change the functionality and navigability of each design. In the context of the MAB problem, we denote each website design layout as an 'arm'. Still, we assume that only one design generates the highest profit by attracting the most visitors, corresponding to that there is one optimal arm. The company wants to allocate the most effective layout design to visitors to increase the chances of engagement with the advertised product. However, the company needs to know which design yields the best results. The main challenge arises when balancing testing a new website design (exploration) and sticking with the website design believed to be the most effective (exploitation). The good thing is that the company can adjust its choice according to data from earlier interactions.

Problem: Offline Evaluation in Multi-Armed Bandits. Suppose we want to estimate the performance of a new policy. Directly deploying this policy online can raise concerns about losing rewards. Instead, we have the interaction history log generated by another algorithm with the environment available. A natural way is to use this data to evaluate the new policy. This is called **offlie evaluation**.

Next, we introduce more notation for the offline evaluation problem. We define history log up to time T as the collection of history decisions and return rewards, denoted by $\mathcal{H}_T := (a_t, r_{t,a_t})_{t=1}^T$. Suppose we have a collection of history log $\mathcal{H}_T$ within the time horizon T. Additionally, to estimate the expected performance of all arms, we need the **arm sampling probability distribution** at each time step t, denoted by $\mathbb{P}_t$. The agent pulls the arm at each time step following the arm sampling probability distribution. If $\mathbb{P}_t$ has been provided by the algorithm explicitly, we will use the augmented history log, which is the history log $\mathcal{H}_T$ added the arm sampling probability distribution $\mathbb{P}_t$ for all time step $0 \leq t \leq T$, $\mathcal{H}_T^+ := (a_t, r_{t,a_t}, \mathbb{P}_t)_{t=1}^T$. The augmented history log at each time step t, including the pulled arm a_t, the instantaneous reward r_{t,a_t}, and the arm sampling probability vector $\mathbb{P}_t$. In the k-armed bandit setting, $\mathbb{P}_t := (p_{t,a})_{a=1}^K$.

Consider the example of evaluating a policy that takes actions uniformly at random. We denote its expected performance by $\mu := \frac{1}{K}\sum_{k=1}^K \mu_k$. Next, we use the Inverse Probability Weighting (IPW) estimator [23] to estimate the μ. Given the augmented log data $\mathcal{H}_T^+$, we can construct the IPW estimator $\hat{r}_{t,a} := \frac{r_t \mathbf{1}\{a=a_t\}}{P_{t,a}}$ and we estimate μ by the following equation

$$\hat{\mu} = \frac{1}{KT}\sum_{t=1}^T \hat{r}_{t,a}.$$

One nice property of $\hat{\mu}$ is that it is an unbiased estimator to μ if it is generated by a well-behaved stochastic bandit algorithm (more precisely $\forall t \leq T$, $\min_a p_{t,a} > 0$). For those stochastic bandit algorithms whose arm sampling probability is not accessible, such as Thompson Sampling (more details can be found in section 3.2), if we construct the estimator like $\hat{\mu}$ but use an estimated arm sampling probability $\hat{\mathbb{P}}_t$ to replace $\mathbb{P}_t$ in place, it is hard to verify if the new estimator is unbiased.

In the following report, in Section 2, we give the formal definition of the Multi-arm bandit problem and several important regret measurements in the asymptotic and finite-time aspects. Section 3 includes typical bandit algorithm families from the literature and compares their regret results. Section 4 shows our proposed bandit algorithm, called *Kullback-Leibler Maillard Sampling* and its regret analysis. Section 5 contains two synthetic experiments related to the Kullback-Leibler Maillard Sampling. Section 6 summarizes the strength of Kullback-Leibler Maillard Sampling and gives some future research aims in the trajectory of our current work. For the summary of the notations used in this report, please see Appendix A.

2 Background

In this section, we present a formal framework for the MAB problem and the notations used throughout the report. We also define the interaction protocol and the performance measure for evaluating MAB algorithms.

- **K-armed bandit instance $\mathcal{B}$** A finite-time K-armed bandit instance $\mathcal{B}$ includes three important components, $\mathcal{B} = (T, \mathcal{A}, \mathcal{V})$. T represents the length of the time horizon. $\mathcal{A}$ is the arm set $\mathcal{A} = \{1, 2, \ldots, K\}$, with each number representing a different arm. $\mathcal{V} = (\nu_a)_{a=1}^K$ is the set of reward distributions ν_a, where ν_a is associated with the arm $a \in \mathcal{A}$ and is from a probability distribution family $\mathcal{F}$.

  There are many choices of $\mathcal{F}$, such as bounded distribution family over $[0,1]$, $\mathcal{F}_{[0,1]}$, Bernoulli distribution family, $\mathcal{F}_{\text{Bern}}$ and One-parameter exponential distribution (OPED) family, $\mathcal{F}_{\text{OPED},\eta,b}$. More formally, $[0,1]$-bounded distribution family is defined as:

$$\mathcal{F}_{[0,1]} := \left\{\nu : \int_0^1 \mathbb{P}_\nu(\mathrm{d}x) = 1\right\}.$$

For example, the reward distribution support set in [5] is bounded on $[0,1]$. Our KL-MS also works in the $[0,1]$-bounded reward setting, and reward distribution all come from $\mathcal{F}_{[0,1]}$.

Another particular case is the reward is $0,1$, like the Click-Through on an advertisement, click or no click represents 1 and 0, so the reward distribution is a Bernoulli distribution. The $(0,1)$-Bernoulli distribution family, which is defined as:

$$\mathcal{F}_{\text{Bern}} := \{\nu : \text{supp}(\nu) = \{0,1\}, \mathbb{P}_\nu(x=1) = \mu, \mu \in [0,1]\}.$$

One case is that the reward distribution has a light tail compared to the Gaussian distribution with variation σ^2, called σ^2-sub-Gaussian distribution family, $\mathcal{F}_{\sigma^2-\text{sub}-\text{G}}$. [1] Therefore, the distribution family set is

$$\mathcal{F}_{\sigma^2-\text{sub}-\text{G}} := \left\{\nu_\sigma : \nu \text{ is } \sigma^2\text{-subgaussian}\right\}.$$

If we want to take a broader view of the reward distribution to cover Bernoulli, Bounded, Possion and Gaussian distribution, the One-parameter exponential distribution (OPED) family will be the ideal distribution family to analyze. Formally, OPED family with some measure η and function $b : \Theta \to \mathbb{R}$ is defined as:

$$\mathcal{F}_{\text{OPED},\eta,b} := \left\{\nu_{\theta_\mu} : \frac{\mathrm{d}\nu_{\theta_\mu}}{\mathrm{d}\eta}(x) = \exp\left(x\theta_\mu - b(\theta_\mu)\right)\right\},$$

where θ_μ is the canonical parameter that maps mean parameter μ in $\mathbb{R}$. Many common distributions can be categorized in OPED with specific choices of canonical parameter θ_μ, the cumulant generating function $b(\theta_\mu)$, and the normalization $q(x)$. For instance, a Poisson distribution with pdf $p_\mu(x) := \frac{\mu^x e^{-\mu}}{xk!}$ is a OPED by setting $\theta_\mu = \ln(\mu)$, $b(\theta_\mu) = e^\theta + \ln(k!)$. And a Bernoulli ditribtion with pdf $p_\mu(x) := \mu^x (1-\mu)^{1-x}$ is a OPED by setting $\theta_\mu = \ln\left(\frac{\mu}{1-\mu}\right)$, $b(\theta_\mu) = \ln\left(e^\theta + 1\right)$.

- **Interaction Protocol**

  Under the finite time horizon setting, the learning agent interacts with the instance $\mathcal{B}$ within a total of T time steps. An interaction protocol of the learning agent is presented in the Protocol 1. The agent's action at the time step t can only depend on the history up to time step $t-1$, i.e. $\mathcal{H}_{t-1} = (a_i, r_{i,a_i})_{i=1}^{t-1}$. r_{t,a_t} is the instantaneous reward returned from the t-th step after pulling arm a_t. Given a time horizon T, the agent gets the cumulative reward $\text{Reward}(T) = \sum_{t=1}^T r_{t,a_t}$.

Protocol 1 Multi-armed Bandit Interaction Protocol

1: **Input:** $K \geq 2$
2: **for** $t = 1, 2, \cdots, T$ **do**
3: Pull an arm $a_t \in \mathcal{A}$
4: Observe reward $r_{t,a_t} \sim \nu_{a_t}$.
5: **end for**

The following notation has been used throughout the rest of this report: Based on $\mathcal{H}_t$, the number of arm a until time step t is denoted by $N_{t,a} := \sum_{s=1}^T \mathbf{1}\{a_s = a\}$. The empirical reward for arm a after time step t is denoted as $\hat{\mu}_{t,a} := \frac{\sum_{s=1}^t r_{s,a_s}\mathbf{1}\{a_s=a\}}{N_{t,a}}$. The suboptimality gap of an arm a is defined as $\Delta_a := \mu_{\max} - \mu_a$. The empirical estimation of Δ_a is $\hat{\Delta}_{t,a} := \hat{\mu}_{t,\max} - \hat{\mu}_{t,a}$, where $\hat{\mu}_{t,\max}$ is the empirical best mean reward which is set to $\max_{a\in\mathcal{A}} \hat{\mu}_{t,a}$.

We define the Kullback-Leibler divergence between two distributions ν and ρ as $\mathsf{KL}(\nu,\rho) = \mathbb{E}_{X\sim\nu}\left[\ln \frac{\mathrm{d}\nu}{\mathrm{d}\rho}(X)\right]$. Also when the reward distributions are Bernoulli, we define $\mathsf{kl}(\mu,\mu') := \mu \ln \frac{\mu}{\mu'} + (1-\mu)\ln\frac{1-\mu}{1-\mu'}$ which is

[1] sub-Gaussian property: we call a random variable X is σ-sub-Gaussian if and only if $\mathbb{P}(|X - \mathbb{E}[X]| \geq \lambda) \leq 2\exp\left(-\lambda^2/\sigma^2\right), \forall\lambda \geq 0$. Sub-gaussian distribution is a probability distribution with a tail decay lighter than the normal distribution. Intuitively, the tail of a sub-Gaussian distribution is dominated by a Gaussian distribution's tail.

the binary Kullback-Leibler divergence between two Bernoulli distributions with expected mean μ, μ' in $[0,1]$, respectively. Finally, We define the variance of a Bernoulli distribution ν with mean μ as $\dot{\mu} = \mu(1-\mu)$.

- **Performance measures**

  Besides regret, we have more fine-grained methods to measure the performance in the literature in an instance-independent or instance-dependent manner in an asymptotic or finite-time regime.

1. Asymptotic optimality Asymptotic optimality refers to the desirable property of a bandit algorithm becoming increasingly effective as the number of time steps approaches infinity. An asymptotically optimal algorithm is one that, over a long enough time horizon, approaches the best possible performance in terms of cumulative rewards or regret minimization. [28] and [13] proved that for any consistent algorithm, the lower bound exists. A bandit algorithm is consistent in an environment family $\mathcal{E} = \{\mathcal{B} : \mathcal{B} = (T, \mathcal{A}, \mathcal{V}), \forall a \in \mathcal{A}, \nu_a \in \mathcal{F}\}$ if the regret is sub-polynomial [2] for any bandit instances in that environment. Therefore, a lower bound of the asymptotic regret is shown in the Theorem 1.

Theorem 1 ([28, 13])**.** *For any bandit instance $\mathcal{B}$, in an environment family $\mathcal{E}$, given a consistent bandit algorithm over $\mathcal{E}$, the regret of such algorithm satisfies*

$$\liminf_{T\to\infty} \frac{\mathrm{Regret}_{\mathcal{B}}^{\pi}(T)}{\ln(T)} \geq \sum_{a:\Delta_a>0} \frac{\Delta_a}{\mathcal{K}_{\inf}^{\mathcal{F}}(\nu_a, \mu_{\max})}. \tag{1}$$

$\mathcal{K}_{\inf}^{\mathcal{F}}(\nu_a, \mu_{\max}) := \inf_{G\in\mathcal{F}} \{\mathsf{KL}(\nu_a, G) : \mathbb{E}_G(X) > \mu_{\max}\}$[11] denotes the minimum Kullback-Leibler divergence between two distributions, ν_a, which is the reward distribution associated with a suboptimal arm a and G, which is an arbitrary reward distribution from G whose expectation is greater than $\mu_{\max}$. For example, an algorithm is asymptotically optimal in the Bernoulli reward setting if for any Bernoulli bandit instance ($\mathcal{B} = (T, \mathcal{A}, \mathcal{V})$, $\forall a \in \mathcal{A}, \nu_a$ is a Bernoulli distribution).

$$\limsup_{T\to\infty} \frac{\mathrm{Regret}(T)}{\ln(T)} = \sum_{a:\Delta_a>0} \frac{\Delta_a}{\mathsf{kl}(\mu_a, \mu_{\max})}, \tag{2}$$

Recall that μ_a is the expectation of the Bernoulli distribution ν_a.

Remark 2. *When we discuss the asymptotic optimality, we need to clarify which reward distribution family we are considering. When focusing on the Bernoulli bandit scenario, the expected regret should satisfy the Eq.* (2) *if the algorithm is asymptotic optimality. If the reward distribution is in the σ^2-sub-Gaussian distribution family, $\mathcal{F}_{\sigma^2-sub\text{-}G}$, the asymptotic lower bound becomes Eq.* (3)

$$\liminf_{T\to\infty} \frac{\mathrm{Regret}(T)}{\ln(T)} \geq \sum_{a:\Delta_a>0} \frac{2\sigma^2}{\Delta_a}, \tag{3}$$

For any distribution that is supported on $[0,1]$, since it is also a $\frac{1}{4}$-sub-Gaussian distribution, we can regard the $[0,1]$-Bounded distribution or Bernoulli distribution as a subset of the sub-Gaussian distribution. However, the regret bound involving $\mathcal{K}_{\inf}^{\mathcal{F}}$ (like Eq. (1)*)) provides a superior regret than the Eq.* (3)*, which does not have the $\mathcal{K}_{\inf}^{\mathcal{F}}$ in the $[0,1]$-Bounded or the OPED reward setting. Considering the Pinsker's inequality, which establishes that $\mathsf{kl}(\mu_a, \mu_{\max}) \geq 2\Delta_a^2$, we can always show that*

$$\sum_{a:\Delta_a>0} \frac{\Delta_a}{\mathsf{kl}(\mu_a, \mu_{\max})} \leq \sum_{a:\Delta_a>0} \frac{1}{2\Delta_a}. \tag{4}$$

[2] Sub-polynomial: A bandit algorithm π is sub-polynomial over a class of bandit environment $\mathcal{E}$ if for any instance $\mathcal{B} \in \mathcal{E}$ and $p > 0$, it holds that $\lim_{n\to\infty} \frac{\mathrm{Regret}_{\mathcal{B}}^{\pi}}{n^p} = 0$.

Eq. (4) *shows that the bound without KL type is never more favorable than the KL type regret bound, even when the regret distribution is not Bernoulli, but the Bounded reward setting, such as* $\mathcal{F}_{[0,1]}$*, such superiority still exists because of the Eq.* (4)*.*

2. Minimax ratio In real-world scenarios, we often deal with finite time horizons and practical constraints. Therefore, the asymptotic perspective might not entirely reflect an algorithm's performance in these practical settings, which typically involve limited interactions. To evaluate a bandit algorithm's performance in the finite regime, we use the ratio between the worst-case regret bound and the minimax optimal regret, called the minimax ratio. Suppose the algorithm's worst regret is not significantly worse than the minimax optimal regret lower bound or the minimax ratio is not large. In that case, we can infer that the algorithm performs satisfactorily.

For any K-armed bandit instance $\mathcal{B}$ with bounded reward setting, [7] shows that the upper bound of the worst-case regret bound is $O\left(\sqrt{KT}\right)$, and [10] shows that lower bound is $O\left(\sqrt{KT}\right)$. Thus, the minimax optimal regret is $\Theta\left(\sqrt{KT}\right)$. Given a K-armed bandit problem with time horizon T, an algorithm has a minimax ratio of $f(T,K)$ if it has a worst-case regret bound of $O\left(\sqrt{KT}f(T,K)\right)$.

By minimizing the minimax ratio, an algorithm ensures that its performance remains competitive even in the presence of a worst-case reward distribution. This ratio captures the algorithm's robustness and adaptability across various bandit instances.

Remark 3. *The minimax ratio quantifies the performance of a bandit algorithm in a finite time horizon* T*, while the asymptotic optimality characterizes the order of* $\mathrm{Regret}(T)$ *of an algorithm as* $T \to \infty$*.*

3. The Sub-UCB criterion [29] shows that even if an algorithm satisfies the optimal minimax ratio and asymptotic optimality, it can still suffer a high regret in a finite time, especially compared with the widely popular Upper Confidence Bound (UCB) algorithm ([7, 14, 34, 8], see section 3.3 for details). This observation introduces another criterion that provides another fine-grained characterization of a bandit algorithm's regret guarantees: the sub-UCB criterion. The notion of the Sub-UCB criterion is initially defined in the context of sub-Gaussian bandits [30]: Given a bandit problem with K arms whose reward distributions are all σ^2-sub-Gaussian. An algorithm is said to satisfy the sub-UCB criterion if, for all σ^2-sub-Gaussian bandit instances, the following inequality is true

$$\mathrm{Regret}(T) \lesssim \sum_{a:\Delta_a>0} \Delta_a + \sum_{a:\Delta_a>0} \frac{\sigma^2}{\Delta_a} \ln T. \qquad \text{(sub-Gaussian case)}$$

$A \lesssim B$ represents that there is a constant $C \in \mathbb{R}^+$, $A \leq C \cdot B$. Specialized to the $[0,1]$-Bounded reward setting, as any distribution supported on $[0,1]$ is also $\frac{1}{4}$-sub-Gaussian, and all suboptimal arm gaps $\Delta_a \in (0,1]$ are such that $\Delta_a < \frac{1}{\Delta_a}$, we can simplify the sub-UCB criterion to: there exists some positive constant C, such that for all $[0,1]$-Bounded reward bandit instances, $\mathrm{Regret}(T) \lesssim \sum_{a:\Delta_a>0} \frac{\ln T}{\Delta_a}$.

4. Closed-form sampling probability distribution Closed-form sampling probability distribution has a nice property in the offline evaluation phase. We need to access the arm sampling probability distribution if we want to utilize the IPW estimator for offline evaluation. A bandit algorithm has a closed-form arm sampling probability distribution, which means that we can record the arm sampling probability distribution when executing the algorithm without introducing further steps to compute or approximate it.

In this report, we propose a new multi-armed bandit algorithm called Kullback-Leibler Maillard Sampling, abbreviated as KL-MS, in the $[0,1]$-Bounded reward setting. The proof details can be found in our recent

work '*Kullback-Leibler Maillard Sampling for Multi-armed Bandits with Bounded Rewards, Hao Qin, Kwang-Sung Jun, Chicheng Zhang, NeurIPS 2023*'[37]. KL-MS aims to achieve an adaptive worst-case regret in the MAB setting which has been reported in the literature. KL-MS is also a bandit algorithm with close-form arm sampling probability distribution, which enables the possibility of solving the efficient offline evaluation problem by combining it with the IPW estimator.

3 Prior Solutions

We list several categories of bandit algorithms in this section, but not all algorithms work in the $[0,1]$-Bounded reward setting; some set the reward distribution as sub-Gaussian or the general OPED. The Bernoulli distribution is a special case of OPED, and the $[0,1]$-Bounded reward setting is a special case of sub-Gaussian distribution. Although we are only interested in the $[0,1]$-Bounded reward setting, we still include the algorithm working for the sub-Gaussian distribution but fit them into the particular $[0,1]$-Bounded reward setting since $[0,1]$-Bounded distribution is $\frac{1}{4}$-sub-Gaussian. Also, for the distribution of rewards in the context of the OPED family, we are applying the regret result directly to the $[0,1]$-Bernoulli scenario regardless of its original setting. Furthermore, it is a known fact that bandit problems with rewards limited to the range $[0,1]$ can be transformed into Bernoulli bandit problems through a straightforward conversion method called **Binarization trick**. This involves observing a reward r_{t,a_t} from the range $[0,1]$ at each time step t, then simulating a Bernoulli trial $\tilde{r}_{t,a_t}$ based on r_{t,a_t}, and using this result in a Bernoulli bandit algorithm. However, this method fails to achieve asymptotic optimality in the context of $[0,1]$-Bounded reward setting (see Remark 2).

Generally speaking, two families of provably efficient algorithms are proposed in the literature for solving the MAB problem: deterministic and stochastic algorithms. In deterministic algorithms, the action taken at each time step t is deterministic, given the interaction history before that time. In stochastic algorithms, the decision-making follows an arm sampling distribution that depends on the interaction history.

Typical algorithms such as the Explore-Then-Commit (ETC) (section 3.1) and UCB-like algorithm (section 3.3) can be categorized as deterministic exploration algorithms. The stochastic algorithms include Thompson Sampling (section 3.2) and Boltzmann Exploration (section 3.4).

3.1 Explicit Explore and Exploitation Algorithms

3.1.1 ETC

One type of algorithm assigns some steps to explore by pulling all arms and acting greedily by selecting the arm having the best empirical reward in other steps. That algorithm is easy to implement in practice and can give a relatively good performance guarantee. A typical algorithm is the explore-then-commit algorithm (ETC). See Algorithm 1 for the exact definition of ETC. It consists of two main stages: exploration and exploitation.

Algorithm 1 Explore-then-commit

Input: $K \geq 2$, $\{\alpha_i, \beta_i\}_{i \in \mathcal{A}}$, m
for $t = 1, 2, \cdots, T$ **do**
  if $t \leq mk$ **then**
    Pull the arm $a_t = (t \mod k) + 1$.
  else
    Pull the arm $a_t = \text{argmax}_{1 \leq a \leq a} \hat{\mu}_{mk,a}$.
  end if
  Observe reward $r_{t,a_t} \sim \nu_{a_t}$.
  Update $\hat{\mu}_{t,a_t}$.
end for

- During the exploration phase, the algorithm allocates several trials to each arm to gather information about their potential rewards, with the goal of identifying which arm might yield the highest expected reward. By dedicating an initial portion of the interactions to exploration, the algorithm ensures that it has enough data to make informed decisions during the commit phase.
- During the exploitation phase, the algorithm selects the arm with the highest mean reward. The algorithm then 'commits' to exploiting the arm for the remainder of the interactions.

By tuning the length of the exploration phase mK, we can balance exploration and exploitation: If m is high, the policy spends excessive time exploring. Conversely, when m is low, the likelihood of the algorithm selecting a suboptimal arm in the exploitation phase increases. On the other hand, if the exploration phase is shorter, the arm with the highest estimated reward during the exploration phase is less likely to be the best arm, and the algorithm might suffer from suboptimal performance during the commit phase. The challenge lies in determining the optimal value for m.

In the $[0, 1]$-Bounded reward setting, we give a regret bound and a minimax ratio of ETC in Theorem 4 and Theorem 5, respectively.

Theorem 4 (ETC)**.** *[30, Ch 6] Given K arms in the $[0, 1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, With the appropriate choice of m, the regret of the ETC algorithm is upper-bounded by:*

$$\text{Regret}(T) \leq \sum_{a:\Delta_a>0} \frac{\Delta_a \ln(T)}{\min_{i\in K} \Delta_i^2} + \sum_{a:\Delta_a>0} \Delta_a. \tag{5}$$

A worst-case regret bound of ETC is summarized in Theorem 5.

Theorem 5 ([38], ETC)**.** *Given K arms in the $[0, 1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, the expected regret of the ETC algorithm is upper bounded as*

$$\text{Regret}(T) \lesssim T^{2/3} \left(K \ln(T)\right)^{1/3}$$

In general, T is much larger than K. Therefore, Theorem 5 indicates that the ETC algorithm suffers a high order regret at any time since the minimax optimal for the reward distribution in $\mathcal{F}_{[0,1]}$ is $O\left(\sqrt{TK}\right)$. Also, from Theorem 4 we cannot conclude whether the ETC satisfies the asymptotical optimality in the Bouned reward setting.

3.1.2 Pros and Cons

Explicit exploration and exploitation algorithms are straightforward to implement. It separates the exploration and exploitation phases into distinct steps, which are easy to understand and analyze. The exploration cost (in terms of suboptimal pulls) is known and fixed upfront, which can be advantageous in settings where a predictable number of exploratory trials is needed. When the differences in expected rewards between arms are large (large gap scenarios), ETC can quickly identify the best arm and perform well with less complex tuning compared to other algorithms.

However, the optimal setting of the exploration phase in ETC often depends on knowing the time horizon T, which may not be practical or possible in all applications. The performance of ETC is incomparable to other algorithms, such as Thompson Sampling (section 3.2) and UCB (section 3.3). Also, ETC does not have a closed-form arm sampling probability distribution at any time.

Next, we will introduce more algorithms that implicitly trade off explore/exploit steps.

3.2 Thompson Sampling

Thompson Sampling (TS) (see Algorithm 2 for the exact definition) uses a stochastic approach to balance exploring different arms with exploiting the arms that perform well. The key idea behind TS is to maintain

a general posterior distribution $\mathcal{P}$ of the expected reward of each arm and update the general posterior distribution based on the historical feedback accordingly. This general posterior distribution represents the algorithm's uncertainty about the expected reward of each arm.

When TS needs to select an arm, TS samples a value from the general posterior distributions $\mathcal{P}_{t,a}$ associated with each arm at time step t. Then, TS will select the arm that has the highest sampled value. Notably, the design of the general posterior probability should make arms with higher expected rewards more likely to be selected. However, there is still a need for suboptimal arms to be explored, achieving the balance between exploiting what is known to be the best and exploring new possibilities.

In TS, one typical way to maintain general posterior distribution for each arm is using a posterior distribution with conjugation pair (prior-posterior or proxy posterior)[4, 5, 27]. The prior distribution captures the agent's initial beliefs or assumptions about the expected reward of each arm. After observing the returned reward, TS updates the general posterior distribution to incorporate the new reward.

Algorithm 2 Generic Thompson Sampling

Input: T, arm set $[K]$, $\{\mathcal{P}_a\}_{a=1}^K$
for $t = 1, 2, \cdots, T$ **do**
  for $a = 1, 2, \ldots, K$ **do**
    Sample $\hat{\theta}_a$ from the general posterior distribution $\mathcal{P}_{t-1,a}$ associated with the arm a.
  end for
  Pull the arm $a_t := \arg\max_{1 \le a \le K} \hat{\theta}_a$.
  Observe reward $r_{t,a_t} \sim \nu_{a_t}$.
  Update $\hat{\mu}_{t,a_t}$ and $\{N_{t,a}\}_{a=1}^K$.
  Update general posterior distribution $\{\mathcal{P}_{t,a}\}_{a=1}^K$.
end for

3.2.1 Thompson Sampling using Beta priors (BernoulliTS)

Assuming that the reward distributions of all arms are Bernoulli, if we want to adapt the generic Thompson Sampling to the Bernoulli reward distribution on $\{0, 1\}$, we can utilize Bernoulli-Beta conjugation by initializing the general posterior distribution $\mathcal{P}_{0,k}$ to be $\mathsf{Beta}(\alpha_k, \beta_k)$ and update $\mathcal{P}_{t,k}, 1 \le t \le T$ following the rule of updating the posterior distribution. More specifically, once BernoulliTS pulls an arm a_t and receive a reward r_{t,a_t}, it updates posterior distribution by updating the parameter pair $(\alpha_{a_t}, \beta_{a_t}) = \left(\alpha_{a_{t-1}} + r_{t,a_t}, \beta_{a_{t-1}} + 1 - r_{t,a_t}\right)$. The regret guarantee has been analyzed in [5] and summarized as Theorem 6.

Theorem 6 ([27], BernoulliTS)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, for $\forall \varepsilon \in (0, 1)$, the number of the expected regret $\mathbb{E}[\text{Regret}(T)]$ of the BernoulliTS algorithm is upper bounded by:*

$$\mathbb{E}[\text{Regret}(T)] \le \frac{1+\varepsilon}{1-\varepsilon} \left(\sum_{a:\Delta_a > 0} \frac{\Delta_a \ln(T)}{\mathsf{kl}(\mu_a, \mu_{\max})} \right) + C, \tag{6}$$

where C is a constant that only relates to the Bandit instance $\mathcal{B}$ and ε.

Theorem 6 indicates that BernoulliTS satisfies the sub-UCB criterion because we can apply the Pinsker's inequality that lower bounds $\mathsf{kl}(\mu_a, \mu_{\max})$ by $2\Delta_a^2$. Also, [4, 5] show that BernoulliTS satisfies the asymptotic optimality and has minimax ratio $\sqrt{\ln(T)}$.

Theorem 7 ([4, 5], BernoulliTS)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$, BernoulliTS satisfies the asymptotical optimality*

$$\lim_{T \to \infty} \frac{\mathbb{E}[\text{Regret}(T)]}{\ln(T)} \le \sum_{a:\Delta_a > 0} \frac{\Delta_a}{\mathsf{kl}(\mu_a, \mu_{\max})}, \tag{7}$$

and has minimax ratio $\sqrt{\ln(T)}$, which satisfies the following equation

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(T)}. \tag{8}$$

Also, it is worth mentioning that when the reward distribution is in $\mathcal{F}_{[0,1]}$, TS with Gaussian prior[5] can achieve a lower order minimax ratio.

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(K)}. \tag{9}$$

Generally, the general posterior probability does not have to be the posterior distribution to make TS perform well. As pointed out by [1], once the general posterior distribution satisfies suitable properties of concentration or anti-concentration, we can guarantee the regret to satisfy asymptotical optimality and have a minimax ratio $\sqrt{K}$.

3.2.2 ExpTS

A most recent work aligned in this way is the ExpTS[24]. It works under the following assumptions of the reward distributions: Given a function $V : \Theta \to \mathbb{R}$, an OPED family $\mathcal{F}_{\text{OPED},\eta,b,V} := \mathcal{F}_{\text{OPED},\eta,b} \cap \{\nu_\theta : \text{variance of } \nu_\theta \text{ is } V(\theta) \text{ and less than } V_{\max}\}$. Here, we restrict the OPED family to the Bernoulli distribution family $\mathcal{F}_{\text{Bern}}$ and let $V_{\max}$ to be $\frac{1}{4}$. Therefore, exact PDF of the general posterior distribution $p_{t,a}$ of ExpTS has been defined to be $\forall t \in \mathbb{N}^+, 1 \le a \le K$,

$$\mathcal{P}_{t,a}(x) := \frac{(N_{t-1,a}-1)\left|x-\hat{\mu}_{t-1,a}\right|}{2V(x)}\exp\left(-(N_{t-1,a}-1)\mathsf{kl}\left(\hat{\mu}_{t-1,a},x\right)\right).$$

Theorem 8 ([24], ExpTS)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, the regret of the ExpTS algorithm is upper bounded by:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>\lambda}\frac{\ln\left(T\Delta_a^2\right)}{\Delta_a} + \sum_{a:\Delta_a\le\lambda}\Delta_a\sqrt{T},$$

where $\lambda \ge 8\sqrt{\frac{1}{T}}$.

In Theorem 8, by letting $\lambda = 8\sqrt{\frac{1}{T}}$ we can find the regret of ExpTS satisfies the sub-UCB criterion in $\mathcal{F}_{\text{Bern}}$. Also, ExpTS has been proven to satisfy the following regret metrics

Theorem 9 ([24], ExpTS)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, the ExpTS satisfies the asymptotic optimality*

$$\lim_{T\to\infty}\frac{\mathbb{E}\left[\text{Regret}(T)\right]}{\ln(T)} = \sum_{a:\Delta_a>0}\frac{\Delta_a}{\mathsf{kl}\left(\mu_a,\mu_{\max}\right)}. \tag{10}$$

And its regret has the minimax ratio $\sqrt{\ln(K)}$.

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(K)}. \tag{11}$$

The Theorem 8 and Theorem 9 also hold in the general OPED family. Details can be found in the paper [24].

3.2.3 ExpTS$^+$

ExpTS$^+$ adds a greedy step that selects the arm that has the best empirical average reward with probability $1-\frac{1}{K}$ or selects the arm that has the best ExpTS sample with probability $\frac{1}{K}$. More specifically, we denote the general posterior probability of ExpTS at time step t as $p_{t,a}^{\text{ExpTS}}$, the sampling step of ExpTS$^+$ becomes:

$$\begin{cases} \text{Samplg } \hat{\theta}_a \text{ from } \mathcal{P}_{t-1,a} \text{ and select the arm has the best sample,} & \text{with probability } \frac{1}{K}\ , \\ \text{Select arm } a_t := \arg\max_{a\in[K]} \hat{\mu}_{t-1,a}, & \text{with probability } 1-\frac{1}{K} \end{cases}$$

With the additional greedy step, ExpTS has been proven to satisfy the following regret metrics

Theorem 10 ([24], ExpTS$^+$)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, the ExpTS$^+$ satisfies the asymptotic optimality by satisfying the equation*

$$\lim_{T\to\infty} \frac{\mathbb{E}\left[\text{Regret}(T)\right]}{\ln(T)} = \sum_{a:\Delta_a>0} \frac{\Delta_a}{\mathsf{kl}\left(\mu_a, \mu_{\max}\right)}. \tag{12}$$

The minimax ratio of ExpTS$^+$ is 1 *and its regret satisfies the following inequality*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT}. \tag{13}$$

ExpTS achieves asymptotic optimality and the sub-UCB criterion, but the minimax ratio is $\sqrt{\ln K}$. Constract to ExpTS, ExpTS$^+$ allocates more probability of picking the empirical best, thus acting more greedily than ExpTS. ExpTS$^+$ can close the gap in the minimax ratio by removing the logarithmic term and achieves $\sqrt{KT}$ regret upper bound. However, ExpTS$^+$ does not satisfy the sub-UCB criterion.

3.2.4 Pros and Cons

Compared to other types of algorithms, TS-type algorithms have a good performance in many experiments[17, 16, 35]. The algorithm can be computationally simpler and faster, especially when the priors and likelihoods are conjugate pairs, making the posterior updates trivial. TS has theoretical solid guarantees under certain conditions, such as asymptotic optimality and small minimax ratio.

However, updating the general posterior distribution $\mathcal{P}$ can be computationally intensive for the distribution that do not have conjugate priors, requiring numerical methods or approximations. Also, [21] points out that in the Gaussian reward distribution with unknown mean and variance parameters, it is risky to choose prior, and the wrong choice could result in suboptimal performance.

Also, TS does not generally have a closed-form arm sampling probability distribution, and we need to use a simulation method to approximate the distribution of arm sampling probability to construct the IPW estimator in the offline evaluation. For example, using the Monte Carlo method to do sampling to estimate the arm sampling distribution $\mathbb{P}_t$. The computation complexity is higher than $O\left(K\right)$[6]. Hence, the approximation method is less efficient than closed-form arm sampling probability distribution methods.

3.3 Upper Confidence Bound Algorithm

The Upper Confidence Bound (UCB) algorithm (algorithm 3 follows the ideal of the optimism principle to solve the classic exploration-exploitation dilemma. UCB algorithm takes the arm that maximizes a surrogate function called UCB for the reward in each round. The construction of UCB for each arm is based on the empirical mean reward and a measure of uncertainty for that arm, which typically has a form like Eq (14)

$$U_a(t) := \hat{\mu}_{t-1,a} + B_{t,a}. \tag{14}$$

Usually $B_{t,a}$ is set as a decreasing function of $N_{t-1,a}$. The key here is that the uncertainty about the reward of each arm influences $B_{t,a}$. The basic idea is that if an arm has a high empirical reward $\hat{\mu}_{t,a}$ reflected

from the history log or is under-explored, making $B_{t,a}$ large, it will have a large UCB and thus is highly likely to be selected. The optimism principle comes into play when selecting which arm to pull next. The UCB algorithm always chooses the arm with the highest UCB $U_a(t)$. This optimistic approach assumes that the actual reward is close to the best-case scenario for each arm. As the algorithm progresses, it naturally balances exploration and exploitation. Arms that yield high rewards will gradually tighten their confidence intervals, reducing their upper bounds. Conversely, arms that have not been explored as much will continue to have higher UCB, encouraging exploration. Over time, the UCB algorithm's optimism in the face of uncertainty leads to both sufficient exploration of all actions and exploitation of the best actions. This results in the convergence towards the optimal action as the one with the highest expected reward becomes clearer.

Algorithm 3 Generic UCB algorithm

Input: arm set $[K]$
for $t = 1, 2, \cdots, T$ **do**
  if $t \leq K$ **then**
    Pull the arm $a_t = t$ and observe reward $r_{t,a_t} \sim \nu_{a_t}$.
  else
    Pull the arm $a_t := \arg\max_{1 \leq k \leq K} U_k(t)$.
    Observe reward $r_{t,a_t} \sim \nu_{a_t}$.
  end if
  Update confidence bound $U_{a_t}(t)$.
end for

For the $[0, 1]$-Bounded reward setting, there are many algorithms in the literature, such as UCB1[3, 9], MOSS[7], kl-UCB[14], kl-UCB++[34], UCB-V[8].

3.3.1 UCB1

[9] analyzes UCB1 over the $[0, 1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$ in the K-arm bandit problem. UCB for each arm a has been defined as

$$U_a(t) = \hat{\mu}_{t-1,a} + \sqrt{\frac{2\ln(T)}{N_{t-1,a}}} \tag{15}$$

The construction of UCB in UCB1 can be written as

$$\text{UCB1: } U_a(t) = \max\left\{\mu \in [0, 1], (\mu - \hat{\mu}_{t-1,a})^2 \leq \frac{2\ln(T)}{N_{t-1,a}}\right\} \tag{16}$$

Based on Eq. (15) and Eq. (16), we construct the confidence bound based on the Euclidean distance between the empirical reward $\hat{\mu}_{t,a}$ for arm a at time t and the possible true mean μ. The first term in Eq. (15) is the empirical mean estimator. The second component, corresponding to the element of 'optimism' in the face of uncertainty, is designed to shrink as the number of times arms a is played, denoted as $N_{t-1,a}$, increases. This implies that as we gather more data about the performance of arm a, our estimate of its true mean reward becomes more precise, allowing the confidence bound to tighten. This part of the equation treats arms chosen less frequently as potentially more advantageous, based on the idea that less information about them implies greater uncertainty and, therefore, the possibility that they might yield better rewards than more frequently chosen arms. This approach encourages exploring lesser-known arms in the decision-making process.

A regret upper bound of UCB1 is presented in Theorem 11:

Theorem 11 ([9], UCB1)**.** *Given K arms in the reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, the regret of the UCB1 algorithm is upper bounded by:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \leq 8 \sum_{a:\Delta_a>0} \frac{\ln(T)}{\Delta_a} + \sum_{a:\Delta_a>0} \Delta_a \tag{17}$$

Although UCB1 can give us a logarithmic regret w.r.t. T, the result shown in Theorem 11 is still not tight enough to guarantee asymptotic optimality. (See Remark 2 for more detailed discussion) The minimax ratio of UCB1 can be derived from Theorem 11, which is $\sqrt{\ln(T)}$.

Theorem 12 (UCB1)**.** *Given K arms in the reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, the regret of UCB1 has minimax ratio $\sqrt{\ln(T)}$:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(T)} \tag{18}$$

UCB1 has a loss bound in terms of Eq. (17) and the minimax ratio (18). We can find that $U_a(t)$ of all arms increases when time progresses, regardless of whether it has been pulled, resulting in favorable to the exploration instead of exploitation.

3.3.2 MOSS

[7] proposes a bandit algorithm called Minimax Optimal Strategy in the Stochastic case (MOSS), for the $[0,1]$-Bounded reward distribution $\mathcal{F}_{[0,1]}$ and the UCB is defined as

$$U_a(t) = \hat{\mu}_{t-1,a} + \sqrt{\frac{1}{N_{t-1,a}} \log_+\left(\frac{T}{KN_{t-1,a}}\right)}. \tag{19}$$

Also, the construction of the confidence bound of MOSS can be written as

$$\text{MOSS: } U_a(t) = \max\left\{\mu \in [0,1], (\mu - \hat{\mu}_{t-1,a})^2 \leq \frac{1}{N_{t-1,a}} \log_+\left(\frac{T}{KN_{t-1,a}}\right)\right\}. \tag{20}$$

The number of arms pulled w.r.t. the suboptimal arm a of MOSS is guaranteed by the following theorem

Theorem 13 ([7], MOSS)**.** *Given K arms in the reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, the regret of the MOSS algorithm is upper bounded by:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \frac{K\log\left(T\Delta_a^2/K\right)}{\Delta_a}. \tag{21}$$

And the minimax ratio of MOSS is 1*:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT}.$$

When the suboptimal gap Δ_a is large, the regret contributed by that arm will be much smaller than the UCB1. Thus, MOSS can achieve a lower minimax ratio. Although MOSS closes the gap between the upper bound and lower bound in terms of any time regret, it still does not satisfy the asymptotic optimality nor the sub-UCB criterion due to the additional K in the Eq. (21) before the logarithm factor.

The term $\frac{1}{N_{t-1,a}} \log_+\left(\frac{T}{KN_{t-1,a}}\right)$ serves as a variance term that accounts for the uncertainty due to limited observations. The $\log_+$ function ensures that this term is non-negative, taking into account the logarithmic growth of our confidence with respect to the total number of trials T, divided by the number of arms K and the number of times arm a has been played. Compare Eq. (15) and Eq. (19), we can find that the UCB of MOSS shrinks faster than UCB when the arm has been pulled. If the performance of an arm is not good,

that arm will be less explored in MOSS than UCB1; thus, MOSS is more favorable to exploitation rather than exploration.

However, the use of the squared Euclidean distance $(\mu - \hat{\mu}_{t,a})^2$ in UCB1 and MOSS makes the asymptotic regret bound to be bounded by $O\left(\sum_{a=1,\Delta_a<0}^{K} \frac{\ln(T)}{\Delta_a}\right)$, which is not exactly equal to $O\left(\sum_{a=1,\Delta_a<0}^{K} \frac{\Delta_a \ln(T)}{\mathsf{kl}(\mu_a,\mu_{\max})}\right)$ while the latter one can guarantee the asymptotic optimality in the Bernoulli reward setting, $\mathcal{F}_{\text{Bern}}$. Such difference reflects that we assume that the reward distributions are $\frac{1}{4}$-sub-Gaussian implicitly, which is a necessary condition to be a bounded distribution over $[0,1]$ but not sufficient. More discussion can be found in Remark 2.

To remedy this issue, another approach is to give the KL-divergence type confidence bound to measure the difference between two reward distributions to replace the Euclid distance.

3.3.3 kl-UCB

kl-UCB[14] uses KL-divergence to measure the difference between the confidence bound and the empirical estimation, and the UCB is defined as

$$U_a(t) = \max\left\{\mu \in [0,1], \mathsf{kl}\left(\hat{\mu}_{t-1,a}, \mu\right) \leq \frac{f(t)}{N_{t-1,a}}\right\},$$

where $f(t) = \log\left(1 + t\log^2(t)\right)$. The asymptotic optimality of kl-UCB and the minimax ratio has been summarized in Theorem 14

Theorem 14 ([14], kl-UCB)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, the kl-UCB satisfies the asymptotic optimality, which means the regret has the following equation*

$$\lim_{T\to\infty} \frac{\mathbb{E}\left[\text{Regret}(T)\right]}{\ln(T)} = \sum_{a:\Delta_a>0} \frac{\Delta_a \ln(T)}{\mathsf{kl}\left(\mu_a, \mu_{\max}\right)}. \tag{22}$$

Also, minimax ratio of kl-UCB is $\sqrt{\ln(T)}$

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(T)}. \tag{23}$$

3.3.4 kl-UCB++

[34] proposes another algorithm called kl-UCB++ which works in the one-parameter exponential family $\mathcal{F}_{\text{OPED},\eta,b}$, but we still only focus on the Bernoulli case. In kl-UCB++, suppose the support set of reward distribution is I. The UCB is defined as

$$U_a(t) = \max\left\{\mu \in I : \mathsf{kl}\left(\hat{\mu}_{t-1,a,},\mu\right) \leq \frac{f(N_{t-1,a})}{N_{t-1,a}}\right\},$$

where $f(t) = \log_+\left(\frac{T}{Kt}\left(\log_+^2\left(\frac{T}{Kt}\right) + 1\right)\right)$ and $\log_+(x) := \max\{\log(x), 0\}$.

Since we focus on the special case of Bernoulli, we give the asymptotic optimality of kl-UCB++ and the minimax ratio, which are summarized in the following theorems.

Theorem 15 ([34], kl-UCB++)**.** *Given K arms in the reward setting $\mathcal{F}_{\text{Bern}}$ and a finite time horizon T, the kl-UCB++ satisfies the asymptotic optimality. Equivalently,*

$$\lim_{T\to\infty} \frac{\mathbb{E}\left[\text{Regret}(T)\right]}{\ln(T)} = \sum_{a:\Delta_a>0} \frac{\Delta_a \ln(T)}{\mathsf{kl}\left(\mu_a, \mu_{\max}\right)}. \tag{24}$$

Also, its minimax ration is 1*:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT} \tag{25}$$

kl-UCB and kl-UCB++ satisfy the asymptotic optimality for Bernoulli reward setting because of the KL-type regret bound in constructing UCB. The distinct difference is that we can regard kl-UCB++ as using an upper bound assigning higher confidence on the observed best arm than kl-UCB. The UCB of arms in kl-UCB++ will not change until being pulled, but in kl-UCB, the UCB of arms not pulled will inflate with the time step increasing. From Theorem 14 and Theorem 15, we can find that kl-UCB++ achieves a lower worst regret guarantee than kl-UCB.

The minimax ratio of kl-UCB is $\sqrt{\ln(T)}$. With the refined design of UCB, kl-UCB++ closes the gap in the worst-case regret bound by making the minimax ratio 1.

3.3.5 UCB-V

UCB-V[8] assumes the $[0,1]$-Bounded reward setting. It is noticeable that UCB-V is a type of 'variance-aware' algorithm since the confidence bound is sensitive to the variance estimation w.r.t. that arm. In [9], the empirical performance of algorithms using estimated variance outperforms those not estimating variance. Therefore, UCB-V proposes utilizing a variance estimator in constructing confidence bounds and assigning a higher probability to the arm with large variation. At time t we have the variance estimator $\hat{V}_{t,a} = \frac{1}{N_{t,a}} \sum_{s=1}^{t} (r_{s,a_s} - \hat{\mu}_{t,a})^2 \mathbf{1}\{a_s = a\}$. The upper confidence bound to the suboptimal arm a is defined as

$$U_a(t) = \hat{\mu}_{t-1,a} + \sqrt{\frac{2\hat{V}_{t-1,a} g(N_{t-1,a}, t)}{N_{t-1,a}}} + \frac{3c}{N_{t-1,a}},$$

where $g(s,t)$ is an exploration function (Defined in [8]) and c is a constant. Based on the above setting, we obtain the regret guarantee to the UCB-V as

Theorem 16 ([8], UCB-V)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T. Denote V_a as the variance of the reward distribution associated with arm a. The regret of the UCB-V algorithm is upper bounded by:*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \left(\frac{V_a^2}{\Delta_a} + 1\right) \log T$$

Theorem 16 shows that UCB-V satisfies the sub-UCB criterion but can not guarantee the asymptotic optimality for the Bernoulli setting. A direct result from Theorem 16 would be the minimax ratio is $\sqrt{\ln(T)}$.

Theorem 17 ([37], UCB-V)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$ and a finite time horizon T, the expected regret of the UCB-V algorithm is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(T)}$$

3.3.6 Pros and Cons

The strength of the UCB type of algorithm consists of several aspects: The regret bound is easy to analyze theoretically. We can follow a certain streamline to decompose the event to conduct the regret analysis[30, Ch.7, 8, 9]. UCB is quite robust in various settings since the theoretical guarantee has been given to control each corner case clearly in the regret analysis. The UCB algorithms do not need to perform exploration and exploitation separately, as with the ETC method, because UCB inherently integrates these aspects into its framework.

Also, the UCB has several weaknesses: The construction of UCB in many algorithms depends on the time horizon T. Therefore the performance of UCB is often closely tied to the time horizon of the problem, and it requires knowledge or estimation of this horizon for optimal tuning. Other algorithms, such as Thompson Sampling, on the other hand, do not require knowledge of the time horizon and can adapt more fluidly. Also, most UCB-type algorithms are deterministic, and we can not obtain a reasonable close-form arm sampling probability distribution.

3.4 Boltzmann Exploration

Boltzmann exploration (BE)[39, 25] is a standard strategy in Reinforcement Learning, especially in sequential decision-making under uncertainty. From Algorithm 4, we can see the details of a generic BE algorithm, $f_a(\cdot)$, is an evaluation function to approximate the performance of arm k. It assigns exponential weight to the sampling probability of each arm, and each round samples an arm from the sampling probability distribution $\mathbb{P}$. Then, the sampling probability will be updated when receiving the reward.

Algorithm 4 Generic Boltzmann Exploration

1: **Input:** arm set $[K]$ and $\mathbb{P}_0$
2: **for** $t = 1, 2, \cdots, T$ **do**
3: Pull the arm

$$\forall 1 \leq a \leq K, p_{t,a} \propto \exp\left(f_a(\mathcal{H}_{t-1})\right)$$

4: Observe reward $r_{t,a_t} \sim \nu_{a_t}$.
5: Update the history record $\mathcal{H}_t$
6: Update the evaluation function $f_a(\mathcal{H}_t)$
7: Update sampling probability,

$$\forall 1 \leq a \leq K, p_{t,a} \propto \exp\left(f_a(\mathcal{H}_t)\right)$$

8: **end for**

3.4.1 BE with exponential learning rate

The basic version of BE chooses the evaluation function $f_a(\mathcal{H}_t^+)$ as $\eta_t \hat{\mu}_{t-1,a}$ (BE-exp-lr, exponential learning rate). η_t is the learning rate parameter changed with time step t. To minimize the regret, BE needs a carefully tuned series of learning rate parameters, $\eta_t > 0$. The literature has frequently highlighted the challenges in determining the correct schedule for η_t [31, 41]. One regret bound given by [15] is Theorem 18

Theorem 18 ([15], BE-exp-lr)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$, a finite time horizon T and a constant $\tau := \frac{16eK\ln(T)}{\Delta}$, with learning rate $\eta_t := \mathbf{1}\{t < \tau\} + \frac{\ln(t\Delta^2)}{\Delta}\mathbf{1}\{t \geq \tau\}$, the expected regret of BE-exp-lr is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \frac{K\ln(eT)}{\Delta^2}, \tag{26}$$

where Δ is the minimum non-negative reward gap among all arms in K.

If we compare the regret bound of BE-exp-lr in Eq (26) with previous methods such as UCB1(Eq. (17)) and MOSS(Eq. (21)), BE-exp-lr is much worse when Δ is small, regardless it has a higher order in the denominator.

3.4.2 Boltzmann-Gumbel Exploration

[15] demonstrates that typical learning rate schedules might fall short of almost optimal regret guarantees. Specifically, BE might overly rely on suboptimal arms even after accurately estimating all mean values or prematurely commit to a less-than-ideal arm and struggle to readjust later. Theorem 18 has a significant drawback: it depends on prior knowledge of problem parameters Δ, which are usually unknown at the outset of the learning process.

Given these findings, [15] concludes that the BE-exp-lr strategy does not offer a more effective approach to regret minimization than the simpler ϵ-greedy exploration method [40, 9]. [15] also offers a solution with a proposed learning rate schedule and calls the new algorithm Boltzmann–Gumbel exploration (BGE). In BGE for every step t, the arm has been pulled to maximize $\hat{\mu}_{t-1,a} + \sqrt{\frac{C^2}{N_{t-1,a}}} Z_{t-1,i}$ where C is a constant

and $Z_{t-1,i}$ follows the standard Gumbel distribution. With the new proposed method, [15] bounds the regret of BGE in Theorem 19. BGE works for σ^2-sub-Gaussian but we focus on the $[0,1]$-Bounded reward setting and let σ^2 to be $\frac{1}{4}$.

Theorem 19 ([15], BGE)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$, a finite time horizon T, the expected regret of BGE is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \frac{\ln\left(T\Delta_a^2\right)}{\Delta_a}.$$

and BGE's worst-case regret is bounded by Theorem 20

Theorem 20 ([15], BGE)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$, a finite time horizon T, the expected regret of BGE is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT}\ln(K).$$

Compared to the UCB1 and kl-UCB, although the logarithmic regret of BGE is worse than UCB1, we can find the minimax ratio of BGE has been improved from $\sqrt{\ln(T)}$ to $\ln(K)$ because of an additional Δ_a^2.

3.4.3 Minimum Empirical Divergence

Another type of BE method is minimum empirical divergence (MED)[19], followed by Maillard Sampling [32, 12]. MED is based on the assumption that the reward distribution has a finite support set, which utilizes all collected data points to construct the support set for each arm. Then, when constructing the arm sampling distribution, it finds the distribution with higher expectations than the optimal arm and has the minimum distance to the actual arm empirical distribution measured by the KL divergence. Specifically, at each time, t MED needs to solve the following question to find the evaluation function $f_a(\mathcal{H}_{t-1})$:

$$f_a(\mathcal{H}_{t-1}) := -N_{t-1,a}\mathsf{KL}\left(\hat{G}_{t-1,a}, \hat{\nu}_{\hat{\mu}_{t,\max}}\right) \tag{27}$$

where $\hat{G}_{t-1,a} := \arg\min_{G\in\mathcal{F},\mathbb{E}[G]\geq\mu_{\max}t} \mathsf{KL}\left(\hat{F}_{t-1,a}, G\right)$, $\hat{F}_{t-1,a}$ is the empirical distribution w.r.t. arm a at time $t-1$ and $\hat{\nu}_{\hat{\mu}_{t,\max}}$ is the best empirical distribution at time $t-1$. The idea of minimizing KL-diverge is to assign a more significant weight to the arm with a higher reward than those arms that have worse empirical performance.

Theorem 21 has guaranteed the regret of MED, and Theorem 22 shows MED satisfies the asymptotic optimality. We should notice that Theorem 21 and 22, but we only give the Bernoulli reward setting for simplicity.

Theorem 21 ([19], MED)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{\text{Bern}}$, a finite time horizon T, the expected regret of MED is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \frac{\Delta_a(1+\varepsilon)\ln(T)}{\mathsf{kl}\left(\mu_a,\mu_{\max}\right)} + O\left(\frac{1}{\Delta_a^3}\right) + O\left(K^2\right),$$

where ε is an arbitrary positive constant.

Theorem 22 ([19], MED)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{\text{Bern}}$, a finite time horizon T, MED is asymptotic optimality and satisfies the Eq.* (28)*:*

$$\lim_{T\to\infty} \frac{\mathbb{E}\left[\text{Regret}(T)\right]}{\ln(T)} \lesssim \sum_{a:\Delta_a>0} \frac{\Delta_a}{\mathsf{kl}\left(\mu_a,\mu_{\max}\right)}. \tag{28}$$

3.4.4 Maillard sampling

Another type of BE is the Maillard sampling (MS) [32, 12]. MS denotes the evaluation function $f_a(\mathcal{H}_t^+) := \exp\left(-\frac{1}{2\sigma^2} N_{t-1,a} \hat{\Delta}_{t-1,a}^2\right)$ and adding a burn-in phase where the agent pulls each arm once in the first K rounds. MS works for σ^2-sub-Gaussian but we focus on the $[0,1]$-Bounded reward setting and let σ^2 to be $\frac{1}{4}$. The regret bound of MS is summarized in Theorem 23

Theorem 23 ([32], MS)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$, a finite time horizon T, the expected regret of MS is upper bounded as*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \frac{\ln\left(T\Delta_a^2/\sigma^2\right)}{\Delta_a}.$$

and its worst-case regret has been bounded by Theorem 24

Theorem 24 ([32], MS)**.** *Given K arms in the $[0,1]$-Bounded reward setting $\mathcal{F}_{[0,1]}$, a finite time horizon T, MS satisfy the asymptotic optimality for sub-Gaussian reward setting,*

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a:\Delta_a>0} \frac{\ln(T)}{\Delta_a} + \sum_{a:\Delta_a>0} \Delta_a,$$

and the expected regret of MS is upper bounded as

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sqrt{KT\ln(T)},$$

MS does not satisfy the asymptotic optimality for the Bernoulli or the $[0,1]$-Bounded reward setting. The regret is always equal or greater than the theoretical lower bound, and this difference becomes particularly significant when $\mathsf{kl}\left(\mu_a, \mu_{\max}\right)$ becomes much larger than $2\Delta_a^2$. (More discussion can be found in Remark 2) An intriguing possibility lies in achieving an upper bound on regret that aligns perfectly with the lower bound, with the potential inclusion of the KL divergence as a crucial element in the algorithm.

3.4.5 Pros and Cons

By considering the exponential of action values, BE naturally incorporates the uncertainty in the value estimates, tending to explore more when uncertainty is high. Also, the arm sampling probability distribution is closed-form, which benefits the offline evaluation.

There are several shortcomings of BE. The efficiency of BE is highly sensitive to the choice of the evaluation function $f_a(\cdot)$. Choosing an appropriate evaluation function can be non-trivial and might require tuning or an adaptive strategy. In the BE with learning rate, the efficiency of the algorithm is highly sensitive to the choice of the learning rate. Choosing an appropriate learning rate can be non-trivial and might require tuning or an adaptive strategy. Computing the softmax probabilities can be computationally intensive, especially when there are many actions to choose from, as it requires normalization over all actions.

3.5 Algorithm comparison

We present a comparative analysis of various existing algorithms in table 1 to elucidate the motivation behind developing a novel algorithm. This comparative overview aims to underscore the shortcomings and limitations of the current bandit algorithms, thereby underscoring the necessity and rationale for creating a new algorithm.

Algorithm& Analysis	Asymptotic Optimality For Bernoulli Distribution	Finite-Time Regret Minimax Ratio	Sub-UCB	Closed-form Probability	Reference
TS	yes	$\sqrt{\ln K}$	yes	no	[4][5][27]
ExpTS	yes	$\sqrt{\ln K}$	yes	no	[24]
ExpTS$^+$	yes	1	no	no	[24]
UCB1	no	$\sqrt{\ln T}$	yes	N/A	[9]
MOSS	no	1	no	N/A	[7]
kl-UCB	yes	$\sqrt{\ln T}$	yes	N/A	[14]
kl-UCB++	yes	1	$-^{\star\star}$	N/A	[34]
UCB-V	no	$\sqrt{\ln T}$	yes	N/A	[8]
BE-exp-lr	no	$-$	no	yes	[15]
BGE	no	$\ln K$	yes	yes	[15]
MED	yes	$-$	$-$	no*	[19]
DMED	yes	$-$	$-$	N/A	[20]
IMED	yes	$-$	$-$	N/A	[22]
MS	no	$\sqrt{\ln T}$	yes	yes	[12]
KL-MS	yes	$\sqrt{\mu_{\max}(1-\mu_{\max})\ln K}$	yes	yes	this paper

Table 1: Comparison of regret bounds for $[0,1]$-Bounded reward distributions. '$-$' indicates that the corresponding analysis is not reported. 'N/A' indicates that the algorithm has a closed-form arm sampling probability distribution but is deterministic. '$\star$' indicates that its computational complexity for calculating the action probability is ln(1/precision). '$\star\star$'indicates that we conjecture that the algorithm is not sub-UCB. The asymptotic optimality means whether the algorithm satisfies such criterion under the $\mathcal{F}_{\text{Bern}}$ reward setting.

Based on the analysis presented in Table 1, it becomes evident that none of the algorithms under consideration simultaneously achieve asymptotic optimality, a minimax ratio of 1, and the sub-UCB property within the family of Bernoulli reward distributions, $\mathcal{F}_{\text{Bern}}$. Among the current set of algorithms, ExpTS and ExpTS$^+$ emerge as the major competitors. While ExpTS$^+$ achieves a minimax ratio of 1 by adopting a more aggressive arm-selection strategy than ExpTS, it does so at the expense of the sub-UCB property. Furthermore, as both ExpTS and ExpTS$^+$ are on the Thompson Sampling distribution framework, they confront challenges in deriving a closed-form arm sampling probability distribution, which may sometimes be unattainable. Consequently, our expectation of the KL-MS algorithm is threefold: firstly, to simultaneously accomplish asymptotic optimality and the sub-UCB property, and secondly, to attain a minimax ratio that is at least on par with ExpTS. Thirdly, to obtain the closed-form arm sampling probability distribution.

4 Kullback-Leibler Maillard Sampling

In this section, we will present the main findings of our study on Kullback-Leibler Maillard Sampling, abbreviated as KL-MS. KL-MS is a Bernoulli variant of the MS algorithm tailored to the $[0,1]$-Bounded reward setting, and we will conduct a comprehensive analysis using the measurement mentioned the section 2.

In MS, the evaluation function $f_a(\mathcal{H}_t) = -N_{t-1,a}\frac{\hat{\Delta}_{t-1,a}}{2\sigma^2}$ which we interpret as the Gaussian KL divergence between two Gaussian distributions where their means are $\hat{\mu}_{t-1,a}$ and $\hat{\mu}_{t-1,\max}$, respectively, and adjusted by the counting $N_{t-1,a}$. In KL-MS(alg. 5), we replace the Gaussian KL divergence $\frac{\hat{\Delta}_{t-1,a}}{2\sigma^2}$ by the Bernoulli KL divergence, $\mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right)$. For the remainder, we have denoted $\dot{\mu} := \mu(1-\mu)$, which is the variance of a Bernoulli distribution with mean μ. By making such adaption, we establish that KL-MS attains a finite-time regret bound (referred to as Theorem 25), which we can simultaneously transform into the following:

- (Asymptotically optimality in the Bernoulli setting) An upper bound on asymptotic regret (Theorem 27), which attains optimality in an asymptotic sense when applied to Bernoulli bandit scenarios.

- (sub-UCB) Through Theorem 26 we have proved KL-MS satisfies the Sub-UCB regret criterion. Many existing minimax optimal algorithms do not satisfy it, resulting in a suboptimal regret in a special bandit instance.
- (Finite-time regret improvement) We obtain an adaptive-worst case regret bound and make the minimax ratio to be $\sqrt{\dot{\mu}_{\max} \ln(K)}$. The regret bound is expressed as $O\left(\sqrt{\dot{\mu}_{\max} KT \ln K} + K \ln(T)\right)$ (referred to as Theorem 28). This regret bound demonstrates two noteworthy characteristics regarding the finite-time improvement. Firstly, in worst-case scenarios, it remains within a factor of at most $\sqrt{\ln K}$ compared to the minimax optimal regret of $\Theta(\sqrt{KT})$ as previously outlined in studies like [7, 10]. Secondly, the coefficient $\sqrt{\dot{\mu}_{\max}}$ in the bound adapts to the variance of the optimal arm's reward. This marks the first instance in the literature where such adaptability is observed in an algorithm with asymptotically optimal assurances.

  If the reader is interested in the proof procedure, please see our paper [37].

Algorithm 5 KL Maillard Sampling (KL-MS)

1: **Input:** $K \geq 2$
2: **for** $t = 1, 2, \cdots, T$ **do**
3: **if** $t \leq K$ **then**
4: Pull the arm $a_t = t$ and observe reward $r_{a_t} \sim \nu_t$.
5: **else**
6: For every $a \in [K]$, compute

$$p_{t,a} = \frac{1}{M_t} \exp\left(-N_{t-1,a} \cdot \mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right)\right) \tag{29}$$

  where $M_t = \sum_{a=1}^{K} \exp(-N_{t-1,a} \mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right))$ is the normalizer.
7: Pull the arm $a_t \sim \mathbb{P}_t$ and observe reward $r_{t,a_t} \sim \nu_{a_t}$.
8: **end if**
9: **end for**

We build the KL-MS motivated by a Bayesian viewpoint for the problem. Although, like MS and MED, we do not follow the Bayesian law exactly. Consider the 2-arm bandit in the $[0, 1]$ bounded reward setting, where $K = 2$ and assume that arm 1 is the best. Then, $\mathsf{kl}\left(\hat{\mu}_{t,1}, \hat{\mu}_{t,\max}\right)$ should be 0 and $\exp\left(-N_{t,1}\mathsf{kl}\left(\hat{\mu}_{t,1}, \hat{\mu}_{t,\max}\right)\right) = 1$ after a few rounds of interactions. Therefore, the arm sampling probability $\mathbb{P}_t$ becomes roughly 1 and $\exp\left(-N_{t,2}\mathsf{kl}\left(\hat{\mu}_{t,2}, \hat{\mu}_{t,\max}\right)\right)$. Based on $\mathbb{P}_t$, the expected instantaneous regret at time t is $0 * 1 + \Delta_2 \exp\left(-N_{t,2}\mathsf{kl}\left(\hat{\mu}_{t,2}, \hat{\mu}_{t,\max}\right)\right) \simeq \Delta_2 \exp\left(-N_{t,2}\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right)$. Add the instantaneous regret from time 1 to time T, we can get

$$\begin{aligned}
\text{Regret}(T) \leq & \sum_{t=1}^{T} \Delta_2 \exp\left(-N_{t,2}\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right) \leq \sum_{t=1}^{\infty} \Delta_2 \exp\left(-t\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right) \\
\leq & \Delta_2 \cdot \frac{\exp\left(-t\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right)}{1 - \exp\left(-\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right)} \\
\leq & \Delta_2 \cdot \frac{1}{\exp\left(\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right)} \leq \frac{\Delta_2}{\exp\left(\mathsf{kl}\left(\mu_{t,2}, \mu_{\max}\right)\right)}
\end{aligned}$$

The last equality indicates KL-MS is asymptotically optimal in the 2-arm Bernoulli reward setting.

The KL Maillard Sampling Algorithm.(Algorithm 5)

- Initialization Step: At the outset, the algorithm ensures that each arm is pulled once (steps 3 to 4). This step guarantees that starting from time step $K + 1$, we have well-defined estimates for the arm sampling distribution.

- Empirical Suboptimality Measurement: Starting from time step $t = K + 1$, the algorithm computes the empirical mean, denoted as $\hat{\mu}_{t-1,a}$, for each arm a. Then, for each arm a, the algorithm calculates the binary KL divergence between $\hat{\mu}_{t-1,a}$ and $\hat{\mu}_{t-1,\max}$, denoted as $\mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right)$. This value quantifies the empirical suboptimality of each arm.
- Sampling Probability Computation: The sampling probability of arm a, $p_{t,a}$, is determined based on the exponential of the negative product of $N_{t-1,a}$ and $\mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right)$ (as described in Eq.(29) in step6). This arm sampling strategy effectively balances exploration and exploitation, favoring arms that have been pulled fewer times ($N_{t-1,a}$ is small) or appear to be closer to optimal empirically ($\mathsf{kl}\left(\hat{\mu}_{t-1,a}, \hat{\mu}_{t-1,\max}\right)$ is small).
- Arm Selection and Reward Observation: The algorithm then samples an arm, denoted as a_t, from the distribution $\mathbb{P}_t$, and observes the corresponding reward, denoted as r_{t,a_t}.

When the reward distributions ν_i are Bernoulli, KL-MS and the MED algorithm (as discussed in [19]) are equivalent. This equivalence arises because, in this scenario, all reward distributions exhibit binary support with values of 0 and 1.

Nevertheless, it is essential to highlight that KL-MS generally differs from the MED algorithm. The MED algorithm calculates the empirical distributions denoted as $\hat{F}_{t-1,a}$ which is a discrete probability distribution supported by a finite set of history, and selects actions based on probabilities defined as $p_{t,a} \propto \exp(-N_{t-1,a}D_{t-1,a})$. Here, the term $D_{t-1,a}$ represents the 'minimum empirical divergence' at time step $t-1$ between arm a and the arm with the highest empirical mean reward. This measure is distinct from the binary KL divergence of the mean rewards used in KL-MS, as explained in more detail in the remark 2.

4.1 Main Regret Theorem

We will show the main theorem serving those three goals to achieve the asymptotic optimality, adaptive worst-case regret bound, and satisfy the sub-UCB criterion simultaneously.

Theorem 25. *For any K-arm bandit problem with $[0,1]$ bounded reward distribution setting, $\mathcal{F}_{[0,1]}$, KL-MS has regret bounded as follows. For any $\Delta > 0$ and $c \in (0, \frac{1}{4}]$:*

$$\begin{aligned}\mathrm{Regret}(T) \leq T\Delta + \sum_{a:\Delta_a > \Delta} \frac{\Delta_a \ln(T\mathsf{kl}\left(\mu_a + c\Delta_a, \mu_{\max} - c\Delta_a\right) \vee e^2)}{\mathsf{kl}\left(\mu_a + c\Delta_a, \mu_{\max} - c\Delta_a\right)} \\ + O\left(\left(\sum_{a:\Delta_a > \Delta} \left(\frac{\dot{\mu}_{\max} + \Delta_a}{c^2\Delta_a}\right) \ln\left(\left(\frac{\dot{\mu}_{\max} + \Delta_a}{c^2\Delta_a^2} \wedge \frac{c^2 T\Delta_a^2}{\dot{\mu}_{\max} + \Delta_a}\right) \vee e^2\right)\right)\right) \end{aligned} \tag{30}$$

The regret bound presented in Theorem 25 comprises three distinct terms.

The first term, denoted as $T\Delta$, regulates the contribution of regret stemming from arms with a proximity to the optimal arm within a margin of Δ, as bounded by the threshold value Δ. For those arms that do not fall into the near-optimal category, the second and third terms govern their regret.

The second term exhibits an asymptotic behavior, approximately $(1 + o(1)) \sum_{a:\Delta_a > 0} \frac{\Delta_a}{\mathsf{kl}(\mu_a, \mu_{\max})} \ln(T)$, with an appropriately chosen constant term 'c'. This term grows logarithmically with the time horizon T.

The third term is simultaneously bounded by two inequalities, as expressed in equations (31) and (32):

- The right-hand side of equation (31) aids in establishing a stringent asymptotic upper bound on regret, as indicated by Theorem 27.

$$\text{third term in (30)} \leq \sum_{a:\Delta_a > 0} \left(\frac{\dot{\mu}_{\max} + \Delta_a}{c^2\Delta_a}\right) \ln\left(\left(\frac{\dot{\mu}_{\max} + \Delta_a}{c^2\Delta_a^2}\right) \vee e^2\right) \tag{31}$$

  Based on the inequality (31), we can see that KL-MS achieves the asymptotic optimality in the Bernoulli reward setting (Theorem 27 and satisfying the sub-UCB criterion (Theorem 26).

- The right-hand side of equation (32) scales on the order of $\ln(T\Delta_a^2)$ and aids in establishing a robust worst-case regret bound, as demonstrated in Theorem 28.

$$\text{third term in (30)} \leq \sum_{a:\Delta_a>0} \left(\frac{\dot{\mu}_{\max}+\Delta_a}{c^2\Delta_a}\right) \ln\left(\frac{c^2 T\Delta_a^2}{\dot{\mu}_{\max}+\Delta_a} \vee e^2\right) \tag{32}$$

  By introducing an analysis method borrowed from Jin et al. [24], we can give another inequality (32), thus we can improve the minimax ratio to $\sqrt{\dot{\mu}_{\max}\ln(K)}$ (Theorem 28).

To the best of our knowledge, existing regret analysis on Bernoulli bandits or bandits with bounded support have regret bounds of the form achieved by kl-UCB(section 3.3.3), kl-UCB++ (section 3.3.4) and ExpTS (section 3.2.2)

$$\mathbb{E}\left[\text{Regret}(T)\right] \leq \sum_{a:\Delta_a>0} \frac{\Delta_a \log(T)}{\mathsf{kl}(\mu_a,\mu_{\max})} + \sum_{a:\Delta_a>0} O\left(\frac{1}{\mu_{\max}}(\log(T))^{4/5}\log\log(T)\,\Delta_a\right) \tag{kl-UCB}$$

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a=1,\Delta_a>0}^{K} \frac{\Delta_a \log\left(\frac{T}{K}\left(1+\left(\log\left(\frac{T}{K}\right)\right)^2\right)\right)}{\mathsf{kl}(\mu_a+\Delta_a,\mu_{\max}-\Delta_a)} + \frac{K^2}{\Delta_a^2} + 1 \tag{kl-UCB++}$$

$$\mathbb{E}\left[\text{Regret}(T)\right] \lesssim \sum_{a=1,\Delta_a>0}^{K} \frac{\ln\left(T\Delta_a^2\right)}{\Delta_a} + \sqrt{T} \tag{ExpTS}$$

From the above two regret upper bounds, we find that the dominant term (the logarithmic term w.r.t. T) in the above regret bound do not have $\mathsf{kl}(,)$ in the logarithm. And its lower-order term does not have a KL type of bound, resulting in a looseness when $\mathsf{kl}(\mu_a,\mu_{\max})$ is largely deviated from $2\Delta_a^2$. Thus, they cannot derive the adaptive term $\dot{\mu}$ in the regret. As we will see shortly, our regret theorem yields a superior adaptive worst-case regret guarantee over previous works due to its tighter bounds.

Based on Theorem 25, we show that in the following corollaries, KL Maillard sampling achieves the sub-UCB, asymptotic optimality, and adaptive worst-case regret guarantee with a logarithmic factor $\sqrt{\ln(K)}$.

Corollary 26 (Sub-UCB)**.** *KL-MS's regret bound* (30) *is* $O\left(\left(\sum_{a:\Delta_a>0}\frac{\ln T}{\Delta_a}\right)\right)$ *and is therefore sub-UCB.*

Corollary 27 (Asymptotic Optimality in Bernoulli reward setting $\mathcal{F}_{\text{Bern}}$)**.** *For any K-arm bandit problem with reward distribution supported on* $[0,1]$*, KL-MS satisfies the following asymptotic regret upper bound:*

$$\limsup_{T\to\infty} \frac{\text{Regret}(T)}{\ln(T)} = \sum_{a\in[K]:\Delta_a>0} \frac{\Delta_a}{\mathsf{kl}(\mu_a,\mu_{\max})} \tag{33}$$

Although Corollay 27 cannot show the asymptotical optimality over the bounded reward setting, since the KL divergence $\mathsf{kl}(\mu_a,\mu_{\max})$ is superior to the $\mathcal{K}_{\inf}^{\mathcal{F}_{[0,1]}}(\nu_{\mu_a},\nu_{\mu_{\max}})$, where ν_{μ_a} and $\nu_{\mu_{\max}}$ are two distribution from $\mathcal{F}_{[0,1]}$ with mean μ_a and $\mu_{\max}$ respectively.

Corollary 28 (Worst-case regret)**.** *For any K-arm bandit problem with reward distribution supported on* $[0,1]$*, KL-MS has regret bounded as:* $\text{Regret}(T) \leq O\left(\left(\sqrt{\dot{\mu}_{\max}KT\ln K} + K\ln T\right)\right)$.

An immediate consequence of this is that KL Maillard sampling exhibits a regret of the order $O\left(\sqrt{KT\ln K}\right)$, which is a factor of $O\left(\sqrt{\ln K}\right)$ away from the minimax optimal regret of $\Theta\left(\sqrt{KT}\right)$, as previously established [34, 7]. This correspondence also aligns with the worst-case regret bound of $O\left(\sqrt{KT\ln(K)}\right)$ by ExpTS [24].

Another significant characteristic of this regret bound is its adaptivity to $\dot{\mu}_{\max}$, which stands for the variance of the reward associated with the optimal arm in the context of Bernoulli bandit or its upper bound

in the general bounded reward setting. Specifically, if the parameter $\mu_{\max}$ is situated close to either 0 or 1, leading to a very low value of $\dot{\mu}_{\max}$, the regret becomes substantially smaller than $O\left(\sqrt{KT\ln K}\right)$.

It is worth noting that UCB-V [8] and kl-UCB/kl-UCB++, while not explicitly mentioned, possess a worst-case regret bound of $O\left(\sqrt{\dot{\mu}_{\max}KT\ln T}\right)$, which is less favorable than our bound due to the difference in logarithmic factors. Among these, UCB-V does not attain asymptotic optimality in the Bernoulli case. Additionally, logistic linear bandits [2, 33] can be applied to Bernoulli K-armed bandits and achieve similar worst-case regret bounds that involve $\dot{\mu}_{\max}$, but their lower-order term can be notably worse.

5 Experiments

We will conduct two sets of synthetic experiments in this section. The first experiment compares the regrets among MS, KL-MS, and Thompson Sampling. The second experiment aims to show the superiority of KL-MS over the Thompson Sampling method in offline evaluation scenario.

5.1 Regret Comparison

We compare KL-MS and two other algorithms, BernoulliTS (Section 3.2.1) and MS (section 3.4.4).In the BernoulliTS approach, we choose a beta distribution as the prior (Beta$(0.5, 0.5)$). The reward environment is adapted from [26] and comprises two two-armed bandit scenarios. The mean reward environments are $\overrightarrow{\mu_1} := (0.20, 0.25)$ and $\overrightarrow{\mu_2} := (0.80, 0.90)$ and we run the simulation 2000 rounds and compare the empirical mean in both settings.

From Figure 1 and Figure 2, we observe that KL-MS outperforms MS by a noticeable margin, although it falls behind BernoulliTS. However, the next section will reveal that BernoulliTS produces somewhat unreliable logged data for offline evaluation.

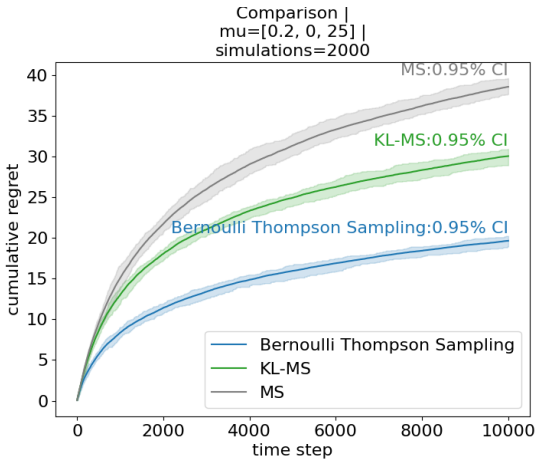

Figure 1: $\overrightarrow{\mu_1} := (0.20, 0.25), T = 10^4$

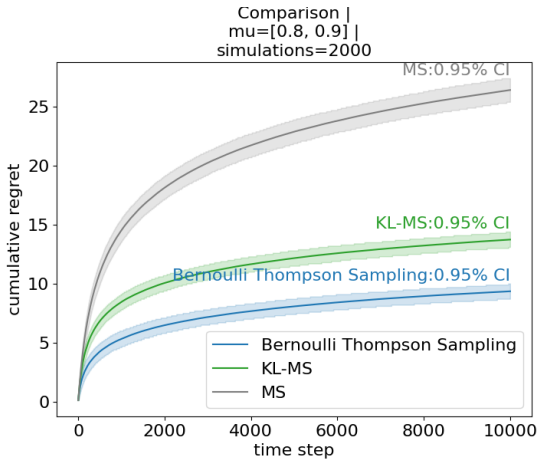

Figure 2: $\overrightarrow{\mu_2} := (0.80, 0.90), T = 10^4$

5.2 Offline Evaluation

In this section, we present the results of our simulations focusing on offline evaluation using logged data. The logged data is generated by two bandit algorithms: KL-MS and BernoulliTS. We aim to estimate the expected reward of a policy that selects actions uniformly at random from the arm set $[K]$. Hence the expected reward is equal to $\bar{\mu} = \frac{1}{K}\sum_{i=1}^{K}\mu_i$.

The format of the logged data is as follows: $\mathcal{H}_T^+ := (a_t, r_{t,a_t}, \mathbb{P}_t)_{t=1}^T$. Reminding that a_t represents the chosen action, and r_{t,a_t} signifies the received reward, and $\mathbb{P}_t = (p_{t,a})_{a\in\mathcal{A}}$ denotes the arm sampling probability

distribution (exact or approximate), all at time step t. We employ the Inverse Probability Weighting (IPW) estimator [23] to estimate μ, defined as:

$$\hat{\mu} = \frac{1}{T}\sum_{t=1}^{T}\frac{r_{t,a_t}}{\mathbb{P}_t(i=a_t)K}.$$

We consider two different time horizons for the interaction log, setting T to be 10^3 or 10^4. For BernoulliTS, we employ a Monte Carlo (MC) method to estimate the action probabilities, and we vary the number of MC samples M from the set $10^3, 10^4, 10^5$. Note that MC estimation of action probabilities comes with a significant computational cost. In our simulations, for $T = 10^3$, KL-MS takes 0.43 seconds to generate its logged data, whereas BernoulliTS with $M = 10^3$ requires 15.21 seconds for the same task. Setting $M = 10^4$ or $M = 10^5$ might be impractical in real-world applications.

Figures 3 to 14 are based on the average result over 2×10^3 independent trials. Thus, we can depict histograms of the IPW estimates of the average reward derived from logged data generated by KL-MS and BernoulliTS with MC estimation of action probabilities.

Specifically, we use two 2-armed bandit problems with mean rewards of $\overrightarrow{\mu_1} = (0.20, 0.25)$ and $\overrightarrow{\mu_2} = (0.8, 0.9)$.

Tables 2 to 9 provide details on the Mean Squared Error (MSE) and the bias estimate of each estimator. We can observe from the figures and tables that:

The logged data generated by KL-MS consistently yield more accurate estimates of μ compared to the data produced by BernoulliTS with MC estimation of action probabilities. The performance of offline evaluation using BernoulliTS's logged data is sensitive to the number of MC samples M. While setting $M = 10^4$ or $M = 10^5$ aligns the performance with that of KL-MS, the estimation error for the more practical $M = 10^3$ setting is notably higher (Figure 9, 12) As the time step T increases, from Figure 3 and 6 to Figure 9 and 12, the error between the IPW estimator using BernoulliTS's logged data and the actual performance becomes larger At the same time, KL-MS maintains a consistent level of error, which is smaller than that of BernoulliTS.

Therefore, we conclude that if the Monte Carlo method is used to approximate arm sampling distribution without enough precision during the offline policy evaluation, the IPW estimator can be biased, and this discrepancy will increase with time. If we want to increase the precision of the Monte Carlo method, we will suffer a high computation cost. This problem does not exist when using the history record generated by KL-MS, an algorithm with a closed-form arm sampling distribution.

$$\overrightarrow{\mu_1} := (0.20, 0.25)\,, T = 10^3$$

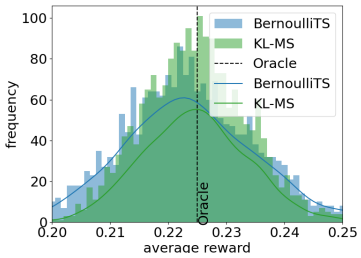

Figure 3: $M = 10^3$

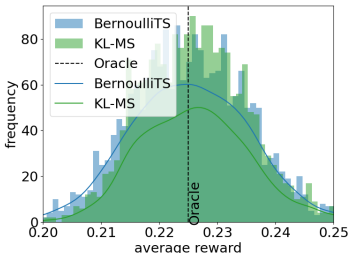

Figure 4: $M = 10^4$

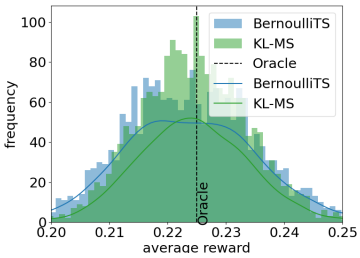

Figure 5: $M = 10^5$

$\overrightarrow{\mu_2} := (0.80, 0.90)\,, T = 10^3$

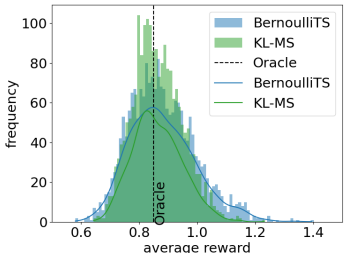

Figure 6: $M = 10^3$

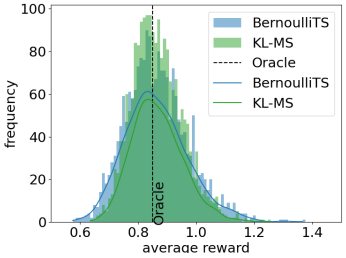

Figure 7: $M = 10^4$

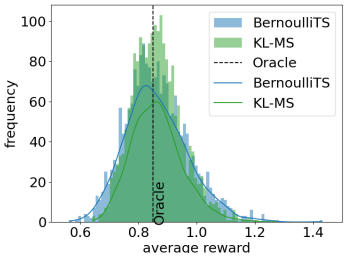

Figure 8: $M = 10^5$

$\overrightarrow{\mu_1} := (0.20, 0.25)\,, T = 10^4$

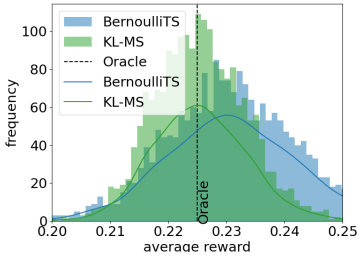

Figure 9: $M = 10^3$

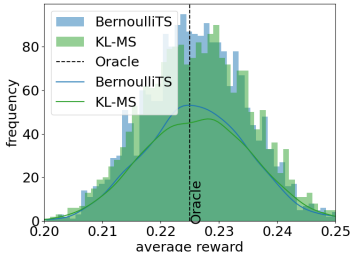

Figure 10: $M = 10^4$

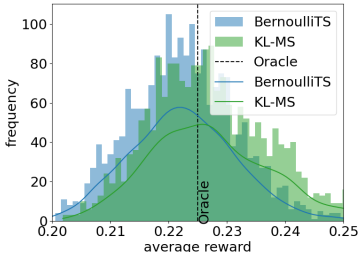

Figure 11: $M = 10^5$

$\overrightarrow{\mu_2} := (0.80, 0.90)\,, T = 10^4$

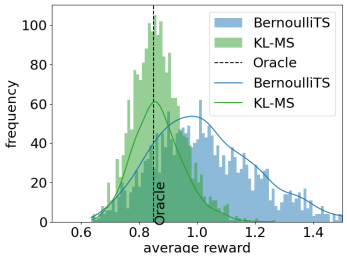

Figure 12: $M = 10^3$

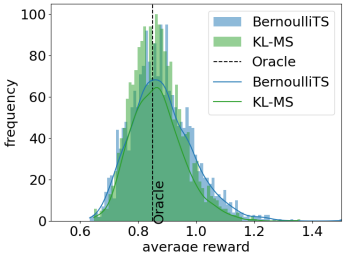

Figure 13: $M = 10^4$

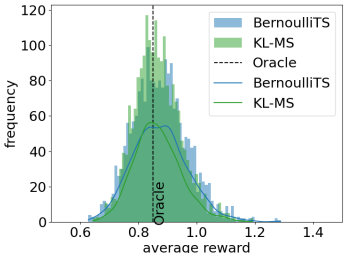

Figure 14: $M = 10^5$

Table 2: MSEs for $\overrightarrow{\mu_1} := (0.20, 0.25)$, $T = 10^3$

	M		
	10^3	10^4	10^5
BernoulliTS	0.00014	0.00012	0.00014
KL-MS	0.00001	0.00001	0.00001

Table 3: Bias for $\overrightarrow{\mu_1} := (0.20, 0.25)$, $T = 10^3$

	M		
	10^3	10^4	10^5
BernoulliTS	-0.00059	0.00106	-0.00068
KL-MS	-0.00096	0.00118	0.00011

Table 4: MSEs for $\overrightarrow{\mu_2} := (0.80, 0.90)$, $T = 10^3$

	M		
	10^3	10^4	10^5
BernoulliTS	0.01464	0.01143	0.01228
KL-MS	0.00733	0.00782	0.00749

Table 5: Bias for $\overrightarrow{\mu_2} := (0.80, 0.90)$, $T = 10^3$

	M		
	10^3	10^4	10^5
BernoulliTS	0.02911	0.01741	0.01636
KL-MS	0.01304	0.01412	0.01355

Table 6: MSEs for $\overrightarrow{\mu_1} := (0.20, 0.25)$, $T = 10^4$

	M		
	10^3	10^4	10^5
BernoulliTS	0.00017	0.00010	0.00009
KL-MS	0.00007	0.00006	0.00011

Table 7: Bias for $\overrightarrow{\mu_1} := (0.20, 0.25)$, $T = 10^4$

	M		
	10^3	10^4	10^5
BernoulliTS	0.00637	0.00142	-0.00240
KL-MS	0.00052	0.00066	0.00220

Table 8: MSEs for $\overrightarrow{\mu_2} := (0.80, 0.90)$, $T = 10^4$

	M		
	10^3	10^4	10^5
BernoulliTS	0.06842	0.01276	0.01220
KL-MS	0.00898	0.00804	0.00929

Table 9: Bias for $\overrightarrow{\mu_2} := (0.80, 0.90)$, $T = 10^4$

	M		
	10^3	10^4	10^5
BernoulliTS	0.17947	0.03401	0.04313
KL-MS	0.02046	0.01731	0.01123

6 Conclusion and Future Work

We have conducted a literature review over several existing families of bandit algorithms, including explicitly exploration and exploitation algorithms, the Upper Confidence Bound family, the Thompson Sampling family, and the Boltzmann Exploration family, and proposed KL-MS, a KL version of Maillard sampling for stochastic multi-armed bandits in the $[0, 1]$-bounded reward setting, with asymptotic optimality for the Bernoulli reward setting, an adaptive minimax ratio $\sqrt{\dot{\mu}_{\max}}$, the sub-UCB criterion, and a closed-form arm sampling probability, which is highly amenable to off-policy evaluation. One immediate advantage of KL-MS is that it only requires constant time complexity concerning the target numerical precision in computing the arm probability.

We have many possible revenues to extend KL-MS:

A more general reward distribution setting Since many Thompson Sampling algorithms work for a general setting of the reward distribution, such as ExpTS and kl-UCB work for the one-parameter exponential family and the family of MED work over semi-bounded reward distribution, we would like to extend the KL-MS to a more general reward setting and compare it with existing methods. For instance, generalizing the KL-MS algorithm from a bounded reward setting to the one-parameter exponential distribution family would be promising. Such a generalization should preserve key attributes, such as asymptotic optimality and the sub-UCB criterion, and minimax ratio comparable to or surpassing that of the UCB standards $\sqrt{ln(K)}$.

Our speculation is to replace the current arm sampling probability distribution $\mathbb{P}_t$ by the following setting,

$$p_{t,a} \propto \exp\left(-N_{t-1,a} \cdot \mathsf{KL}\left(\hat{\nu}_a, \hat{\nu}_{\max}\right)\right), \tag{34}$$

Recall that $N_{t-1,a}$ is the number of arm a has been pulled up to time $t-1$(inclusive), $\mathsf{KL}(,)$ is the KL divergence in a given OPED family $\mathcal{F}_{\text{OPED}}$, $\hat{\nu}_a$ is the distribution in $\mathcal{F}_{\text{OPED}}$ with empirical mean $\hat{\mu}_a$ and $\hat{\nu}_{\max}$ is the distribution in $\mathcal{F}_{\text{OPED}}$ with the best empirical mean $\hat{\mu}_{\max}$. As for the minimax ratio, we make a conjecture that its minimax ratio is $\sqrt{V_{\mu_{\max}} \ln(K)}$, where $V_{\mu_{\max}}$ is the variance from the optimal arm.

Finite time analysis of MED/IMED/DMED In the context of MAB algorithms, finite-time analysis for algorithms like MED, IMED, and DMED involves a detailed investigation of their performance within finite-time horizons. This analysis often includes the pursuit of exact minimax ratios and the examination of the sub-UCB criterion. Here's a deeper exploration of these aspects:

- Upon closely examining Lemma 9 in [19], specifically equation (20), we observe a significant bound applied to each suboptimal arm a. The authors have limited the expected number of times a is chosen, denoted as $\mathbb{E}[N_{T,a}]$, by a term not smaller than $\sum_{t=1}^{T} K(t+1)^{|\text{supp}(\nu_1)|} \exp\left(-tC(\mu_1, \mu_1 - \varepsilon)\right)$, where $C(\mu, \mu') := \frac{(\mu-\mu')^2}{2\mu'(1+\mu)}$ and $\varepsilon \leq \Delta_a$. This bound translates to $\Omega\left(\frac{1}{\Delta_a^{2|\text{supp}(\nu_1)|}}\right)$ when μ_1 remains bounded away from both 0 and 1. Since $|\text{supp}(\nu_1)| \geq 2$, the lower order term becomes $\Omega\left(\frac{1}{\Delta_a^3}\right)$, resulting in a higher order worst-case regret bound.

- (Exact Minimax Ratio) The exact minimax ratio of MED, IMED, and DMED aims to quantify their performance relative to the minimax optimal strategy within finite time horizons. Since the regret bound is hidden behind $O(\cdot)$ in their original regret analysis, we can revisit their analysis and try to give a bound with the exact expression as we did in the KL-MS.

- (Examination of the sub-UCB criterion) The sub-UCB criterion is one of the key elements of finite-time analysis. It shows that the target MAB algorithm exhibits performance comparable to the UCB algorithm, indicating that their regret bound always has the same order as the UCB if the criterion has been satisfied by such an algorithm. This analysis also does not appear in the original paper. Since we regard KL-MS as a specialized case of MED in the Bernoulli reward setting, we believe MED can also achieve sub-UCB in a broader setting, such as the OPED reward setting.

Moving from stateless to a stateful environment. Building upon the extension of Thompson Sampling from MAB problems to the more complex Markov Decision Process (MDP) setting, as explored in [18, 36, 42], we aim to extend Maillard Sampling similarly. Therefore, Maillard Sampling could offer a novel approach to MDPs by providing a closed-form transition probability distribution.

7 Acknowledgement

Thanks to the comprehensive exam committee, Chicheng Zhang, Kwang-Sung Jun, Xueying Tang, and Marek Rychlik, for your invaluable participation and guidance during my comprehensive exam. Your insights and expertise have contributed significantly to my academic journey. I thank Professor Kwang-Sung Jun and Professor Chicheng Zhang for their dedication and assistance in my research, which have been instrumental in shaping my work. I am also immensely thankful to Professor Chicheng Zhang for the guidance in writing my report. Your constructive feedback and expert advice have been crucial in refining my work and enhancing its quality.

References

[1] M. Abeille and A. Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*, pages 176–184. PMLR, 2017.

[2] M. Abeille, L. Faury, and C. Calauzenes. Instance-Wise Minimax-Optimal Algorithms for Logistic Bandits. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3691–3699, 2021.

[3] R. Agrawal. Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):1054–1078, 1995.

[4] S. Agrawal and N. Goyal. Further optimal regret bounds for thompson sampling. In *Artificial intelligence and statistics*, pages 99–107, 2013.

[5] S. Agrawal and N. Goyal. Near-optimal regret bounds for thompson sampling. *Journal of the ACM (JACM)*, 64(5):1–24, 2017.

[6] D. F. Anderson, D. J. Higham, and Y. Sun. Computational complexity analysis for monte carlo approximations of classically scaled population processes. *Multiscale Modeling & Simulation*, 16(3):1206–1226, 2018.

[7] J.-Y. Audibert, S. Bubeck, et al. Minimax policies for adversarial and stochastic bandits. In *COLT*, volume 7, pages 1–122, 2009.

[8] J.-Y. Audibert, R. Munos, and C. Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

[9] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

[10] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The Nonstochastic Multiarmed Bandit Problem. *SIAM J. Comput.*, 32(1):48–77, jan 2003. ISSN 0097-5397. doi: 10.1137/S0097539701398375.

[11] D. Baudry, K. Suzuki, and J. Honda. A general recipe for the analysis of randomized multi-armed bandit algorithms. *arXiv preprint arXiv:2303.06058*, 2023.

[12] J. Bian and K.-S. Jun. Maillard Sampling: Boltzmann Exploration Done Optimally. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 54–72, 2022.

[13] A. N. Burnetas and M. N. Katehakis. Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*, 17(2):122–142, 1996.

[14] O. Cappé, A. Garivier, O.-A. Maillard, R. Munos, and G. Stoltz. Kullback-leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, pages 1516–1541, 2013.

[15] N. Cesa-Bianchi, C. Gentile, G. Lugosi, and G. Neu. Boltzmann exploration done right. *Advances in neural information processing systems*, 30, 2017.

[16] O. Chapelle and L. Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24, 2011.

[17] A. Garivier, H. Hadiji, P. Menard, and G. Stoltz. Kl-ucb-switch: optimal regret bounds for stochastic bandits from both a distribution-dependent and a distribution-free viewpoints. *The Journal of Machine Learning Research*, 23(1):8049–8114, 2022.

[18] A. Gopalan and S. Mannor. Thompson sampling for learning parameterized markov decision processes. In *Conference on Learning Theory*, pages 861–898. PMLR, 2015.

[19] J. Honda and A. Takemura. An asymptotically optimal policy for finite support models in the multi-armed bandit problem. *Machine Learning*, 85(3), 2011.

[20] J. Honda and A. Takemura. Finite-time regret bound of a bandit algorithm for the semi-bounded support model. *arXiv preprint arXiv:1202.2277*, 2012.

[21] J. Honda and A. Takemura. Optimality of thompson sampling for gaussian bandits depends on priors. In *Artificial Intelligence and Statistics*, pages 375–383. PMLR, 2014.

[22] J. Honda and A. Takemura. Non-asymptotic analysis of a new bandit algorithm for semi-bounded rewards. *J. Mach. Learn. Res.*, 16:3721–3756, 2015.

[23] D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association*, 47(260):663–685, 1952.

[24] T. Jin, P. Xu, X. Xiao, and A. Anandkumar. Finite-time regret of thompson sampling algorithms for exponential family multi-armed bandits. In *Advances in Neural Information Processing Systems*, 2022.

[25] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

[26] E. Kaufmann, N. Korda, and R. Munos. Thompson sampling: An asymptotically optimal finite-time analysis. In *Proceedings of the international conference on Algorithmic Learning Theory (ALT)*, pages 199–213, 2012.

[27] N. Korda, E. Kaufmann, and R. Munos. Thompson sampling for 1-dimensional exponential family bandits. *Advances in neural information processing systems*, 26, 2013.

[28] T. L. Lai, H. Robbins, et al. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[29] T. Lattimore. Refining the Confidence Level for Optimistic Bandit Strategies. *Journal of Machine Learning Research*, 19(20):1–32, 2018. URL `http://jmlr.org/papers/v19/17-513.html`.

[30] T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[31] M. Littman and A. Moore. Reinforcement learning: A survey, journal of artificial intelligence research 4, 1996.

[32] O.-A. Maillard. *APPRENTISSAGE SÉQUENTIEL: Bandits, Statistique et Renforcement.* PhD thesis, Université des Sciences et Technologie de Lille-Lille I, 2011.

[33] B. Mason, K.-S. Jun, and L. Jain. An experimental design approach for regret minimization in logistic bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7736–7743, 2022.

[34] P. Ménard and A. Garivier. A minimax and asymptotically optimal algorithm for stochastic bandits. In *International Conference on Algorithmic Learning Theory*, pages 223–237. PMLR, 2017.

[35] M.-h. Oh and G. Iyengar. Thompson sampling for multinomial logit contextual bandits. In *Advances in Neural Information Processing Systems*, pages 3151–3161, 2019.

[36] Y. Ouyang, M. Gagrani, A. Nayyar, and R. Jain. Learning unknown markov decision processes: A thompson sampling approach. *Advances in neural information processing systems*, 30, 2017.

[37] H. Qin, K.-S. Jun, and C. Zhang. Kullback-leibler maillard sampling for multi-armed bandits with bounded rewards, 2023.

[38] A. Slivkins et al. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.

[39] R. S. Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 216–224. Elsevier, 1990.

[40] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

[41] J. Vermorel and M. Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.

[42] W. Zeng and Y. Liu. Markov decision process modeled with bandits for sequential decision making in linear-flow. *arXiv preprint arXiv:2107.00204*, 2021.

A Table of Notations

$\mathcal{A}$: The set of all arms.

T : Total time length.

$\mathcal{V}$: Collection of reward distribution associated with an instance. In K-arm bandit, $\mathcal{V} = (\nu_a)_{a=1}^{K}$,

K : The number of arms.

$\mathcal{B} := (T, \mathcal{A}, \mathcal{V})$, Multi-arm bandit instance.

a_t : Arm pulled at time t.

μ_a : Mean of reward distribution associated with arm a.

$\mu_{\max}$: The best-expected reward returned from the optimal arm

$\Delta_a := \mu_{\max} - \mu_a$The expected reward gap from the optimal arm to the arm a

r_{t,a_t} : Reward returned at time t by pulling arm a_t.

π : a MAB algorithm

$\text{Regret}_{\mathcal{B}}^{\pi}$: The expected regret induced by the MAB algorithm π on the instance $\mathcal{B}$.

$\text{Regret}(T)$: The expected regret w.r.t. time length T

$\mathbb{P}_t$: Probability distribution of arm pulling at the time step t. In K-arm bandit, $\mathbb{P}_t = (p_{t,a})_{a=1}^K$,

$p_{t,a}$: The probability of pulling arm a at the time step t.

$\hat{\mathbb{P}}_t$: The empirical estimation to the probability distribution of arm pulling at the time step t.

In K-arm bandit, $\hat{\mathbb{P}}_t = \left(\hat{P}_{t,a}\right)_{a=1}^K$.

$\hat{p}_{t,a}$: The empirical estimation to the probability of pulling arm a at the time step t.

$\mathcal{P}_{t,a}$: General posterior distribution for arm a at the time step t in TS.

$\mathcal{H}_T := (a_t, r_{t,a_t})_{t=1}^T$, History log among T time steps.

$\mathcal{H}_T^+ := (a_t, r_{t,a}, \mathbb{P}_t)_{t=1}^T$, Enhanced history log.

$\mathcal{E} := \{\mathcal{B} : \mathcal{B} = (T, \mathcal{A}, \mathcal{V}), \forall a \in \mathcal{A}, \nu_a \in \mathcal{F}\}$ Environment family

$\mathsf{KL}(\nu_1, \nu_2)$: KL divergence between two distributions, ν_1 and ν_2

$\mathsf{kl}(\mu_1, \mu_2)$: KL divergence between two Bernoulli distributions specified by mean μ_1 and μ_2

$\mathcal{K}_{\inf}^{\mathcal{F}} := \mathcal{K}_{\inf}^{\mathcal{F}}(F_i, \mu_{\max}) := \inf_{G \in \mathcal{F}} \{\mathsf{KL}(F_i, G) : \mathbb{E}_G(X) > \mu_{\max}\}$

$\mathcal{F}$: Reward distribution family.

$\mathcal{F}_{[0,1]} := \left\{\nu : \int_0^1 \mathbb{P}_\nu \, \mathrm{d}x = 1\right\}$, Bounded reward distribution family.

$\mathcal{F}_{\text{Bern}} := \{\nu : \text{supp}(\nu) = \{0,1\}, \mathbb{P}(x=1) = \mu, \mu \in [0,1]\}$, Bernoulli reward distribution family.

$\mathcal{F}_{\text{OPED},\eta,b} := \{\nu_\theta : \mathbb{P}_\theta(x) = \exp(x\theta - b(\theta) + c(x))\}$, One-parameter exponential reward distribution family.

$\mathcal{F}_{\sigma^2-\text{sub}-\text{G}} := \mathcal{F}_{\text{sub}-\text{G}} := \{\nu_\sigma : \nu \text{ is } \sigma\text{-subgaussian}\}$, σ^2-Sub-Gaussian reward distribution family.

$\hat{\mu}_{t,a} := \frac{\sum_{i=1}^t \mathbf{1}\{a_i = a\} r_{t,a_i}}{\sum_{i=1}^t \mathbf{1}\{a_i = a\}}$, Empirical mean estimation w.r.t. arm a until time t(inclusive).

$N_{t,a} := \sum_{i=1}^t \mathbf{1}\{a_i = a\}$, Number of arm pulls w.r.t. arm a until time t(inclusive).

$\hat{\mu}_{t,\max} := \max_{a \in \mathcal{A}}$, The best empirical mean reward up to time step t

$\hat{\Delta}_{t,a} := \hat{\mu}_{t,\max} - \hat{\mu}_{t,a}$, The gap between the empirical best mean reward and the empirical mean reward of arm a

$\dot{\mu} := \mu(1-\mu)$

V_a : Variance of ν_a

$\hat{V}_{t,a}$: The empirical estimation to V_a at time step t

$U_a(t)$: The upper confidence bound to arm a at time t