MSMR: BANDIT WITH MINIMAL SWITCHING COST AND MINIMAL MARGINAL REGRET

Anonymous authors
Paper under double-blind review

ABSTRACT

Effectively balancing switching costs and regret remains a fundamental challenge in bandit learning, especially when the arms exhibit similar expected rewards. Traditional upper confidence bound (UCB) -based algorithms struggle with this trade-off by frequently switching during exploration, incurring high cumulative switching costs. Recent approaches attempt to reduce switching by introducing structured exploration or phase-based selection, yet they often do so at the expense of increased regret due to excessive exploitation of suboptimal arms. In this paper, we propose a new unified framework for bandit problems with switching costs, containing several classical algorithms, applicable to both Multi-Armed Bandits (MAB) and Combinatorial Multi-Armed Bandits (CMAB). Our approach is built on three key components: initial concentrated exploration, near-optimal exploitation, and predictive selection, which together achieve a principled balance between switching cost and regret. Based on this framework, we introduce the Minimal Switching Cost and Minimal Marginal Regret (MSMR) family of algorithms. Theoretically, we show that MSMR algorithms achieve a regret upper bound of $\mathcal{O}(\log n)$ over horizon n, incur only $\mathcal{O}((\log n)^{1-\varepsilon})$ switching cost, and its marginal loss has an upper bound of $\mathcal{O}(\lambda\sqrt{\log n})$ by setting $\varepsilon = 1/2$, where λ and $\varepsilon \in (0, 1)$ are hyper-parameters. Experiments show that MSMR algorithms reduce switching costs to 1.0% (MAB) and 1.3% (CMAB) of those incurred by standard baselines, while maintaining comparable regret, demonstrating their practical effectiveness.

1 INTRODUCTION

The stochastic multi-armed bandit (MAB) problem is a classical framework for sequential decision-making, where a learning agent repeatedly chooses from a set of arms with unknown reward distributions to minimize cumulative regret (16). Over the years, this framework has been extended to accommodate more complex scenarios. A notable generalization is the combinatorial multi-armed bandit (CMAB) problem, where the agent selects a subset of base arms—known as a super arm—in each round, and receives feedback from all the selected base arms. This formulation captures a wide range of real-world applications, including online advertising, network optimization, and healthcare systems (6; 25; 21; 19; 9).

To minimize regret, traditional bandit algorithms aim to balance exploration and exploitation. Since Thompson's early work on bandits for clinical trials (24), a rich body of theoretical and algorithmic developments has followed. Lai and Robbins (15) established the first lower bounds for regret, showing that it must grow at least logarithmically in the number of rounds. Auer et al. (3) proposed the UCB algorithm, achieving logarithmic regret. In the combinatorial setting, Chen et al. (6) introduced the CUCB algorithm, which was later extended to probabilistically triggered arms (8), both achieving $\mathcal{O}(\log n)$ regret bounds.

However, in many practical applications, regret is not the only performance measure, as switching between arms across rounds may incur explicit or implicit costs. For example, in co-branding recommendation systems,

Table 1: Comparison of Regret, Switching Cost, and Marginal Loss Upper Bound.

Setting	MAB Algorithm	Regret	Switching Cost	Marginal Loss
MAB	UCB(3)	$\mathcal{O}(\log n)$	$\mathcal{O}(\log n)$	$\mathcal{O}(\lambda \log(n))$
	Phased-UCB(16)	$\mathcal{O}(\log n)$	$\mathcal{O}(\log \log n)^{*}$	$\mathcal{O}(\log n + \lambda \log \log n)$
	Batched Tsallis-INF(1)	$\mathcal{O}(\log n)$	———	$\mathcal{O}(\lambda^{1/3} n^{2/3})$
	Batched Arm Elimination (13)	$\mathcal{O}(n^{1/B} \log n)^{**}$	$\mathcal{O}(B)$	$\mathcal{O}(n^{1/B} \log n + \lambda B)$
	MSMR-UCB	$\mathcal{O}((\log n)^{\varepsilon})$	$\mathcal{O}((\log n)^{1-\varepsilon})$	$\mathcal{O}(\lambda \sqrt{\log n})$
CMAB	CUCB(6)	$\mathcal{O}(\log n)$	$\mathcal{O}(\log n)$	$\mathcal{O}(\lambda \log n)$
	Phased-CUCB(16)	$\mathcal{O}(\log n)$	$\mathcal{O}(\log \log n)$	$\mathcal{O}(\log n + \lambda \log \log n)$
	B-FTRL (11)	$\mathcal{O}(n^{2/3})$	$\mathcal{O}(n^{2/3})$	$\mathcal{O}(\lambda n^{2/3})$
	MSMR-CUCB	$\mathcal{O}((\log n)^{\varepsilon})$	$\mathcal{O}((\log n)^{1-\varepsilon})$	$\mathcal{O}(\lambda \sqrt{\log n})$

* Appendix J theoretically analyzes why MSMR performs better than Phased methods.
** B is the number of batches and small B will cause large regret.

repeatedly switching recommended items adds fixed overhead beyond suboptimal choices. Similarly, in session-based recommendation scenarios (26), frequent product changes can fragment user attention and reduce click-through rates (CTR), creating additional operational costs. These overheads are commonly termed *switching costs*. To address this, several recent works propose switching-aware bandit algorithms. A common approach uses phased strategies that repeatedly select the same (super) arm within each phase, limiting the number of switches. This framework has been applied in both MAB settings, such as Batched Tsallis-INF (16; 14; 22; 1), and CMAB settings, such as B-FTRL (11), to reduce switching frequency.

Despite these advances, existing methods still suffer from two fundamental dilemmas: **(1) The dilemma between regret and switching cost:** To reduce switching, current algorithms often tolerate increased regret, both Batched Tsallis-INF and B-FTRL have polynomial-level regret, which may be unacceptable in regret-sensitive applications. **(2) The dilemma among arms with similar rewards:** When many (super) arms have similar expected rewards, existing methods like B-FTRL tend to oscillate between them, resulting in excessive switching while elimination-based methods (14) risk converging to suboptimal policy.

Regarding these challenges, we propose a novel framework called Bandit with Minimal Switching Cost and Minimal Marginal Regret (MSMR), which incorporates three key technical modules: *initial concentrated exploration*, *near-optimal exploitation*, and *predictive selection*. The initial concentrated exploration phase occurs at the beginning of the learning process and uses a single phase to gathering sufficient information for each arm. The near-optimal exploitation technique determines whether the currently selected arm should be pulled additional times within the current phase. The predictive selection technique anticipates whether the currently selected arm will need to be explored in the near future, allowing the algorithm to explore it proactively in advance. We prove the effectiveness of these techniques and theoretically demonstrate that MSMR achieves a switching cost of only $\mathcal{O}((\log n)^{\varepsilon})$, while maintaining asymptotically the same regret as standard bandit algorithms, which is $\mathcal{O}(\log n)$, where n is the time horizon and $0 < \varepsilon < 1$ is a hyper-parameter we can choose flexibly. The main contributions of this paper are as follows:

- We propose a novel unified framework to address the two dilemmas in bandit problems: the trade-off between regret and switching cost, and the instability caused by arms with similar expected rewards. This framework, which encompasses a range of classical algorithms, incorporates three core techniques and is highly flexible, allowing it to adapt to a wide range of bandit settings, including both MAB and CMAB.
- We provide rigorous theoretical guarantees for each core technique in the framework and prove that MSMR algorithms achieve a significantly improved trade-off between regret and switching cost. Unlike existing methods that typically reduce switching cost at the expense of marginal regret, through the carefully designed exploitation function in the *near-optimal exploitation module*, MSMR asymptotically achieves the same regret as standard algorithms, while incurring only minimal switching cost.

- We conduct extensive experiments and ablation studies on MSMR algorithms. The results show that MSMR achieves only 1.0% and 1.3% of the switching cost incurred by standard methods in MAB and CMAB settings respectively, while maintaining nearly the same level of regret. These results highlight the superior performance of our framework and the effectiveness of the key techniques.

2 Related Works

The Multi-Armed Bandit (MAB) problem serves as a foundational model in sequential decision-making, balancing exploration and exploitation to minimize regret (16). While MAB focuses on selecting a single arm, real-world applications like online advertising often require choosing combinations of arms, leading to the Combinatorial Multi-Armed Bandit (CMAB) framework, which generalizes MAB by allowing the selection of super arms, i.e., combinations of base arms, at each round (8; 25; 20).

Phased Bandits: Phased bandit algorithms partition the learning process into discrete phases, maintaining a fixed action within each phase to reduce computational overhead and accelerate exploration. This approach mitigates the frequent updates required in traditional bandit algorithms, offering efficiency gains. (16) provide a comprehensive overview of bandit algorithms, including phased strategies. In the batched setting, Perchet et al. (2016) analyze the trade-offs between batch size and regret, demonstrating that appropriately chosen batch sizes can yield near-optimal performance. In adversarial contexts, (11) introduce algorithms that adaptively determine phase lengths to balance exploration and exploitation effectively. Moreover, (4) discusses the benefits of structured exploration in adversarial environments. These phased approaches are particularly beneficial in scenarios where switching costs or computational constraints are significant concerns.

Switching Cost: Incorporating switching costs into bandit problems introduces additional complexity, as learners must balance the trade-off between exploration benefits and the incurred costs of changing actions. (11) analyze this scenario, establishing a regret lower bound of $\widetilde{\Theta}(n)$ for adversarial bandits with unit switching costs. (23) further explored the stochastic setting, revealing phase transitions in optimal regret rates as a function of the switching budget. Then (14; 1) gave the Batched Tsallis methods in MAB setting and (11) B-FTRL in CMAB settings. Phased strategies naturally align with the goal of minimizing switching costs by limiting action changes to phase boundaries. (2) extended this concept to settings with feedback graphs, proposing algorithms that consider both the structure of feedback and switching costs. These approaches demonstrate that structured exploration can effectively manage switching costs without significantly compromising regret.

While existing algorithms address either regret minimization or switching cost reduction, achieving an optimal balance between the two remains challenging. (11) highlights that minimizing switching costs often leads to increased regret, as infrequent action changes can hinder exploration. (23) demonstrates that strict switching constraints can cause abrupt changes in optimal strategies, complicating the learning process. Moreover, in environments with numerous similar arms, algorithms may oscillate between near-optimal actions, incurring unnecessary switching costs without substantial gains in reward. (2) addresses this by incorporating feedback graphs, yet challenges persist in balancing exploration and exploitation under switching constraints. The newest methods given by (14; 1; 11) still cause a large number of regret though reducing switching costs. These limitations highlight the need for a novel framework that significantly reduces switching costs while incurring only minimal marginal regret compared to standard methods.

3 Problem Setup

Regret. We denote $[\![K]\!]$ as the set $\{1, 2, \ldots, K\}$ for any $K \in \mathbb{N}^+$, and $\zeta(\cdot)$ as the Riemann Zeta Function, which is $\zeta(s) = \sum_{n=1}^{\infty} n^{-s}$. Let $[\![K]\!]$ denotes the set of arms. For each arm $i \in [\![K]\!]$, pulling it at round t yields a reward feedback $X_{i,t} \in [0, 1]$. The unknown reward vector is represented by $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$,

where $\mu_i = \mathbb{E}[X_{i,t}]$ denotes the expected reward for any arm i. The optimal arm is denoted as the arm i_* which maximizes the expected reward, i.e., $\mu_* = \max_{j \in [\![K]\!]} \mu_j$. At each round t, the agent selects an arm i_t. The objective of the MAB problem is to identify this optimal arm while minimizing regret in time horizon n, which is defined as:

$$Reg(n) = n\mu_* - \mathbb{E}[\sum_{t=1}^{n} X_{i_t,t}]. \tag{1}$$

Based on the definition above, in the CMAB scenario, the learning agent selects a combination of multiple base arms from $[\![K]\!]$ at each round, referred to this combination as a super arm S, which has m base arm in it. Let $\mathcal{S}$ as the set of all feasible super arms. At each round t, the agent selects a super arm $S_t \in \mathcal{S}$, and the outcomes $X_{i,t}$ for all base arms i in S_t are revealed. The reward for a selected super arm S_t at round t, denoted as $R(S_t)$, is a non-negative random variable that depends on the specific problem instance, the selected super arm S_t, and the rewards of the revealed base arms. In some scenarios, the reward can be simply as the sum of the rewards of the base arms in S_t: $R(S_t) = \sum_{i \in S} X_{i,t}$ (18), while in more general cases, the reward function can be more complex, such as nonlinear functions, non-symmetric functions of rewards from these base arms, etc. The expected reward of selecting a super arm is defined as $r_{\boldsymbol{\mu}}(S) = \mathbb{E}[R(S_t)]$. The optimal super arm is denoted as the super arm S_* that maximizes the expected reward, i.e., $r_{\boldsymbol{\mu}}(S_*) = \max_{S \in \mathcal{S}} r_{\boldsymbol{\mu}}(S)$. The goal of CMAB is to identify the optimal super arm while minimizing regret. For many reward functions, computing the exact S_* is NP-hard, even when $\boldsymbol{\mu}$ is known. To address this, CMAB literature (25; 27; 20; 9) often assumes access to an offline (α, β) -approximation oracle. This oracle, for given parameters $\alpha, \beta \leq 1$, takes an expectation vector $\boldsymbol{\mu}$ as input, and outputs a super arm $S \in \mathcal{S}$, such that $P[r_{\boldsymbol{\mu}}(S) \geq \alpha \cdot \mathrm{opt}_{\boldsymbol{\mu}}] \geq \beta$, where β is the success probability of the oracle, and $\mathrm{opt}_{\boldsymbol{\mu}} = r_{\boldsymbol{\mu}}(S_*)$ is the mean reward of the optimal super arm. The (α, β)-approximation regret of a CMAB algorithm after n rounds of play using such an oracle under the expectation vector $\boldsymbol{\mu}$ is formally defined as:

$$Reg_{\boldsymbol{\mu},\alpha,\beta}(n) = n \cdot \alpha \cdot \beta \cdot \mathrm{opt}_{\boldsymbol{\mu}} - \mathbb{E}\left[\sum_{t=1}^{n} r_{\boldsymbol{\mu}}(S_t)\right]. \tag{2}$$

Following (6; 8; 20), we make two mild assumptions about the expected reward $r_{\boldsymbol{\mu}}(S)$:

- **Monotonicity**. The expected reward of playing any super arm $S \in \mathcal{S}$ is monotonically non-decreasing with respect to the expectation vector. Specifically, if for all $i \in [\![K]\!]$, $\mu_i \leq \mu_i'$, then $r_{\boldsymbol{\mu}}(S) \leq r_{\boldsymbol{\mu}'}(S)$ for all $S \in \mathcal{S}$.
- **Bounded smoothness**. There exists a strictly increasing (and thus invertible) function $f(\cdot)$, called the bounded smoothness function, such that : (1) for any two super arm S and S', we have $0 \leq r_{\boldsymbol{\mu}}(S) - r_{\boldsymbol{\mu}}(S') \leq f(\Gamma_1)$ if $\min_{i \in S} \max_{j \in S'} |\mu_i - \mu_j| \leq \Gamma_1$. (2) for any two expectation vectors $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$, we have $|r_{\boldsymbol{\mu}}(S) - r_{\boldsymbol{\mu}'}(S)| \leq f(\Gamma_2)$ if $\max_{i \in S} |\mu_i - \mu_i'| \leq \Gamma_2$.

Switching Cost. We define the *switching cost* as the total number of times the agent changes its selected (super) arm between consecutive rounds. Formally, the switching cost $C(n)$ in MAB and CMAB settings are given by:

$$C(n) = \mathbb{E}[\sum_{t=1}^{n-1} \mathbb{I}(i_t \neq i_{t+1})], \quad \text{and} \quad C(n) = \mathbb{E}[\sum_{t=1}^{n-1} \mathbb{I}(S_t \neq S_{t+1})], \tag{3}$$

where $\mathbb{I}(\cdot)$ is the indicating function. This metric quantifies switching costs between selecting different (super) arms and is critical in applications where frequent changes incur penalties (11; 22).

Marginal Loss. When evaluating the trade-off between regret and switching cost, some studies(1; 22) have adopted a linear combination of the two as an integrated performance metric. Following a similar approach, this paper defines the marginal loss as the difference between such a combination and its theoretically optimal counterpart, formally given as:

$$R^A(\lambda, n) = (Reg^A(n) - Reg^{opt}(n)) + \lambda(C^A(n) - C^{opt}(n)), \tag{4}$$

where $Reg^{opt}(n)$ is the lowest regret that can be achieved up to now, $C^{opt}(n)$ is the lowest switching cost that can be achieved up to now. We choose standard UCB(CUCB) algorithm as $Reg^{opt}(n)$ and greedy algorithm as $C^{opt}(n)$(see Appendix G for details). In this setting, the metric reflects the capability of balancing regret and switching cost by comparing the existing measure with its theoretical optimum.

Upper Confidence Bound. In bandit problems, Upper Confidence Bound (UCB)-based approaches are widely used to balance exploration and exploitation. These methods aim to exploit the best-known (super) arms while still exploring less-visited ones to avoid convergence to suboptimal solutions (16). Specifically, they maintain an upper confidence estimate for each (super) arm that combines its empirical mean with an exploration bonus, and select the (super) arm with the highest estimate at each round. The width of the confidence interval controls the level of exploration (19). Below, we outline the design of UCB in both the MAB and CMAB settings.

In the MAB setting, let $T_i(t)$ denote the number of times arm i has been pulled up to round t. The empirical mean reward of arm i is given by: $\hat{\mu}_{i,t} = (1/T_i(t)) \sum_{s=1}^{t} X_{i,s} \cdot \mathbb{I}(j_s = i)$. The corresponding upper confidence estimate is: $\bar{\mu}_{i,t} = \hat{\mu}_{i,t} + c_{i,T_i(t)}$, where the confidence bonus is defined as $c_{i,T_i(t)} = \sqrt{2 \ln t / T_i(t)}$. In the CMAB setting, $T_i(t)$ and $\hat{\mu}_{i,t}$ are computed in the same way as in the MAB case. The upper confidence estimate is also defined as $\bar{\mu}_{i,t} = \hat{\mu}_{i,t} + c_{i,T_i(t)}$, but with a slightly different confidence interval: $c_{i,T_i(t)} = \sqrt{3 \ln t / 2T_i(t)}$, since the super arm involves the combination of multiple base arms. For specific derivations, please refer to (6).

4 ALGORITHMS

Determining "*when to switch without sacrificing performance*" is a central challenge in bandit learning with switching costs. To address the trade-off between regret and switching cost, we propose the Bandit with Minimal Switching Cost and Minimal Marginal Regret (MSMR) framework for various bandit settings. MSMR begins with an "Initial Concentrated Exploration" phase, where all arms are explored in a single batch to collect sufficient statistics while avoiding repeated switches. In the subsequent "Near-optimal Exploitation" phase, switching costs are reduced by favoring arms with higher empirical rewards. Together, these phases substantially lower switching costs while incurring only minimal marginal regret. To further improve performance when (super) arms have similar rewards, we introduce a "Predictive Selection" technique that anticipates near-future selections to prevent unnecessary switches. Due to space constraints, we only present MSMR-CUCB in the main text and other variant is provided in Appendix C.

Phase 1: We adopt an initial concentrated exploration strategy. In the MAB setting, we continue exploring an arm i as long as $\hat{\mu}_{i,t} + \sqrt{2 \ln n / T_i(t)} \geq 1$. In the CMAB setting, a super arm S is explored while $\min_{j \in S} \hat{\mu}_{j,t} + \sqrt{3 \ln n / 2T_j(t)} \geq 1$, enabling efficient, compact exploration. Due to feature of UCB-based algorithms, where the frequency of exploring a suboptimal (super) arm depends on its reward gap from the optimal, even the worst-performing (super) arm is pulled at least $\mathcal{O}(\log n)$ times (15). Building on this, we concentrate these inevitable explorations into a single initial phase. Allocating sufficient exploration at the start naturally reduces switching costs and gathers information crucial for effective learning in subsequent phases.

Phase 2: If the selected (super) arm i or S has a relatively large empirical estimate, the agent pulls it $\gamma_i(t)$ or $\gamma_S(t) = \gamma_{\text{argmin}_{j \in S} T_j(t)}(t)$ times (we express it generally as $\gamma(t)$ for simplicity; see Section 5 for detailed expressions), where $\gamma(\cdot)$ is the exploitation function. For example, $\gamma(t) = 1$ in CUCB (6), while $\gamma(t) = \mathcal{O}(t^{1/2})$ in B-FTRL. A "relatively large empirical estimate" means $i_t = \hat{i}_{*,t}$ in the MAB setting and $r_{\hat{\boldsymbol{\mu}}}(S_t) \geq \alpha \cdot r_{\hat{\boldsymbol{\mu}}}(\hat{S}_{*,t})$ in the CMAB setting, where $i_t = \text{argmax}_{j \in [\![K]\!]} \bar{\mu}_{j,t}$, $\hat{i}_{*,t} = \text{argmax}_{j \in [\![K]\!]} \hat{\mu}_{j,t}$, $S_t = \text{argmax}_{S \in \mathcal{S}} r_{\bar{\boldsymbol{\mu}}}(S)$, and $\hat{S}_{*,t} = \text{argmax}_{S \in \mathcal{S}} r_{\hat{\boldsymbol{\mu}}}(S)$ in round t. Here, α is the approximation parameter in the (α, β)-approximation oracle.

Near-optimal exploitation is a strategy where the agent repeatedly exploits a selected (super) arm with a relatively large empirical estimate, assuming it is near-optimal. We argue that when the UCB-selected arm also attains the highest empirical mean, it becomes a more reliable candidate for "near-optimal exploitation," justifying more aggressive exploration. If the selected (super) arm is optimal, the agent exploits it without incurring any regret. If it is suboptimal, its empirical estimate is typically close to the optimal arm's. As noted in (15; 16), when expected rewards are similar, the theoretical lower bound on required explorations increases substantially. To reduce switching costs, these repeated explorations are grouped into a single phase. Importantly, exploring such arms in advance does not significantly affect overall regret, since these steps are inevitable under UCB-based algorithms and mainly occur during the under-sampled stage (see Appendix E for details).

Predictive Selection: In bandit problems, switching costs arise from two sources: (1) switching between suboptimal (super) arms and the optimal one, and (2) switching among suboptimal (super) arms. Once the agent selects a suboptimal (super) arm, at least one switching cost with the optimal arm is incurred, which is inevitable under UCB-based algorithms. Consider the MAB setting: if the optimal arm has the lowest upper confidence bound, the agent may keep exploring suboptimal arms until the optimal arm attains the highest bound, resulting in many switches among suboptimal arms. However, if we can predict that the selected arm i_t will be explored in the future before the optimal arm reaches the highest bound, we may explore i_t in advance even if its current estimate is not the highest. This strategy reduces switching among suboptimal arms and applies similarly to the CMAB setting, hence the term *predictive selection*.

In MAB setting, for any arm i, once the agent chooses an arm $i_{t_1} \neq \hat{i}_{*,t_1}$ to explore at round t_1, the algorithm will repeatedly select i_{t_1} in the subsequent phase regardless of the upper confidence estimation if the following inequality holds:

$$\sqrt{2\ln(t_2)/T_{i_{t_1}}(t_2)} + \hat{\mu}_{i_{t_1},t_2} \geq \sqrt{2\ln(t_3)/T_{\hat{i}_{*,t_2}}(t_2)} + \hat{\mu}_{\hat{i}_{*,t_2},t_2}, \tag{5}$$

where $t_2 = t_1 + \gamma_{i_{t_1}}(t_1)$ and $t_3 = t_2 + \sum_{j \neq i_{t_1}, \hat{i}_{*,t_2}} \gamma_j(t_2)$.

In CMAB setting , once the agent chooses a super arm $S_{t_1} \neq \hat{S}_{*,t_1}$ to explore at round t_1, the algorithm will repeatedly select S_{t_1} in the subsequent phase regardless of the upper confidence estimation if the following inequality holds:

$$r(\hat{\boldsymbol{\mu}}_{t_2}, \boldsymbol{c}_{t_2}, S_{t_1}) \geq r(\hat{\boldsymbol{\mu}}_{t_2}, \boldsymbol{c}_{t_3}, \hat{S}_{*,t_2}), \tag{6}$$

where $\boldsymbol{c}_t = (c_{i,t})_{i \in [\![K]\!]}$, $r(\hat{\boldsymbol{\mu}}_t, \boldsymbol{c}_t, \cdot) \triangleq r_{\hat{\boldsymbol{\mu}}_t + \boldsymbol{c}_t}(\cdot) = r_{\bar{\boldsymbol{\mu}}_t}(\cdot)$, $t_2 = t_1 + \gamma_{S_{t_1}}(t_1)$ and $t_3 = t_2 + \sum_{j \in \{[\![K]\!] \setminus (S_{t_1} \bigcup \hat{S}_{*,t_2})\}} \gamma_j(t_2)$. We use MSMR-P to represents the MSMR Algorithm with Predictive Selection 4 (See Appendix C for MSMR Algorithm with Predictive Selection).

5 Theoretical Analysis

In this section, we present some theoretical analyses of our proposed methods, including the regret bounds and switching cost of the MSMR algorithms. A comparison between the MSMR algorithms and other existing methods is shown in Table 1.

Lemma 5.1 *Initial concentrated exploration doesn't increase marginal regret with a probability larger than* $1 - Kn^{-4}$ *in MAB and* $1 - Kn^{-3}$ *in CMAB.*

Lemma 5.1 shows that the MSMR algorithm, when equipped with the *initial concentrated exploration* technique, incurs no higher regret than the standard MSMR algorithm without this technique. As demonstrated in Appendix D, the regret incurred during the initial phase is captured by the under-sampled stage of UCB-based algorithms. Therefore, the *initial concentrated exploration* technique effectively reduces switching costs without introducing marginal regret with a large probability.

Algorithm 1 MSMR-CUCB Algorithm

Input: Time horizon n, constant M, α, function $\gamma(\cdot)$

$t \leftarrow 1$, $\hat{\mu}_i \leftarrow 1$ for all i

while Exists i makes $T_i(t) = 0$ **do**

  $\{\hat{\mu}_{j_1}, \hat{\mu}_{j_2}, \ldots, \hat{\mu}_{j_K}\} \leftarrow$ Sort base arm by $\hat{\mu}_i$ in a decreasing way

  $S \leftarrow \{j_1, j_2, \ldots, j_m\}$

  Play S and observe $X_{i,t}$, for any $i \in S$, update $T_i(t) \leftarrow T_i(t) + 1$, $\hat{\mu}_{i,t} \leftarrow \frac{T_i(t-1)\cdot\hat{\mu}_{i,t-1}+X_{i,t}}{T_i(t)}$, $\bar{\mu}_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{3\ln t/2T_i(t)}$, $\bar{\mu}'_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{3\ln n/2T_i(t)}$ until $\min_{j\in S} \bar{\mu}'_{j,t} \leq 1$ and update t

end while

while $t \leq n$ **do**

  $S_t \leftarrow \text{argmax}_{S\in\mathcal{S}} r_{\bar{\boldsymbol{\mu}}}(S)$, $Z \leftarrow \gamma_{S_t}(t)$

  $\hat{S}_{*,t} \leftarrow \text{argmax}_{S\in\mathcal{S}} r_{\hat{\boldsymbol{\mu}}}(S)$,

  $Z \leftarrow M \cdot \gamma_{S_t}(t)$ when $r_{\hat{\boldsymbol{\mu}}}(S_t) \geq \alpha \cdot r_{\hat{\boldsymbol{\mu}}}(\hat{S}_{*,t})$

  Play super arm S_t $\min\{Z, n-t\}$times

  Update t and $T_i(t)$, $\hat{\mu}_{i,t}$, $\bar{\mu}_{i,t}$ for all base arms

end while

Theorem 5.2 *By setting $\gamma_i(t) = N(T_{i_t}(t))^\varepsilon$, $0 < \varepsilon < 1$ and N is a constant, with probability lager than $1 - Kn^{-4}$, the regret upper bound of MSMR-UCB is*

$$\sum_{i\neq i_*} \left(\frac{8\ln n}{\Delta_i} + MN(8\ln(n))^\varepsilon \Delta_i^{1-2\varepsilon} + 2MN\cdot\zeta(2-\varepsilon)\Delta_i \right), \tag{7}$$

where $\Delta_i = \mu_* - \mu_i$ for each arm i.

Before presenting the regret upper bound of MSMR-CUCB, we first define the gap between super arms in the CMAB setting. Under the (α, β)-approximation oracle, a super arm S is considered *sub-optimal* if $r_{\boldsymbol{\mu}}(S) < \alpha \cdot \text{opt}_{\boldsymbol{\mu}}$. Let $\mathcal{S}_{i,B}$ denote the set of all sub-optimal super arms that include base arm i. We sort the elements in $\mathcal{S}_{i,B}$ as $S^1_{i,\text{B}}, S^2_{i,\text{B}}, \ldots, S^{K_i}_{i,\text{B}}$ in increasing order of their expected rewards, where K_i is the number of such super arms. The regret gap for the j-th sub-optimal super arm is defined as $\Delta^{i,j} = \alpha\cdot\text{opt}_{\boldsymbol{\mu}} - r_{\boldsymbol{\mu}}(S^j_{i,\text{B}})$.

Theorem 5.3 *By setting $i_{S,t} = \text{argmin}_{j\in S} T_j(t)$ and $\gamma_S(t) = N(T_{i_{S_t,t}}(t))^\varepsilon$, with probability lager than $1 - Kn^{-3}$, the regret upper bound of MSMR-CUCB is*

$$\begin{aligned} &\sum_{i\in[\![K]\!], \Delta^i_{min}\geq 0} \left(\ell_n(\Delta^{i,K_i})\Delta^{i,K_i} + \int_{\Delta^{i,K_i}}^{\Delta^{i,1}} \ell_n(x)\,dx \right) + 2KMN\zeta(2-\varepsilon)\Delta_{\max} \\ &+ KMN \max_{k\in[\![K_i]\!], i\in[\![K]\!]} \{(\ell_n(\Delta^{i,k}))^\varepsilon \Delta^{i,k}\}. \end{aligned} \tag{8}$$

where $\ell_n(\Delta^{i,l}) = (6\ln n)/(f^{-1}(\Delta^{i,l}))^2$, $\Delta^i_{\min} = \Delta^{i,K_i}$ and $\Delta_{\max} = \max_{i\in[\![K]\!]} \Delta^{i,1}$,

Theorems 5.2 and 5.3 provide the regret bounds of MSMR algorithms, which dynamically depend on the hyperparameter ε. This parameter controls the trade-off between switching cost and marginal regret, affecting the length of the near-optimal exploitation phase. Its value can be chosen initially based on application requirements and known problem parameters, offering flexibility to adapt the algorithm to different scenarios. Let $Reg_1(n)$ and $Reg_2(n)$ denote the regret upper bounds of MSMR in MAB and CMAB settings, respectively, and $Reg_1^{opt}(n)$ and $Reg_2^{opt}(n)$ the bounds of classical algorithms established in prior work (3; 6). Then, the following relationship holds:

$$\lim_{n\to\infty}\frac{Reg_1(n)}{Reg_1^{opt}(n)}=\lim_{n\to\infty}\frac{Reg_2(n)}{Reg_2^{opt}(n)}=1. \tag{9}$$

This indicates that our algorithm is asymptotically consistent with the classical counterparts, incurring only minimal marginal regret regardless of the value of ε.

Theorem 5.4 *Setting $\gamma_i(t)=N(T_{i_t}(t))^{\varepsilon}$. If $\left(2^{\frac{\varepsilon}{1-\varepsilon}}/(N(1-\varepsilon))\right)\le\ln(n)$, the switching cost upper bound of MSMR-UCB is*

$$4KMN\cdot\zeta(2-\varepsilon)+2\sum_{i\neq i_*}\left(\frac{8\ln(n)}{\Delta_i^2}\right)^{1-\varepsilon}\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}+2K.$$

Theorem 5.5 *Setting $\gamma_S(t)=N(T_{i_{S_t,t}}(t))^{\varepsilon}$. If $\left(2^{\frac{\varepsilon}{1-\varepsilon}}/(N(1-\varepsilon))\right)\le\ln(n)$, the switching cost upper bound of MSMR-CUCB is*

$$4KMN\cdot\zeta(2-\varepsilon)+2\sum_{i=1}^{K}\left(\frac{6\ln(n)}{(f^{-1}(\Delta^{i,K_i}))^2}\right)^{1-\varepsilon}\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}+2\sum_{i=1}^{K}K_i.$$

Theorems 5.4 and 5.5 present the switching cost of the MSMR algorithms, which depend dynamically on the choice of the parameter ε, which plays a leading role. The value of ε can be still determined at the beginning based on specific application requirements and known problem parameters.

Lemma 5.6 *If $i_{t_1}\neq\hat{i}_{*,t_1}$ is selected at round t_1 and Eq.5 is hold at round $t_2=t_1+\gamma_{i_{t_1}}(t_1)$, at least one arm $i\neq\hat{i}_{*,t_2}$ will be pulled more than 1 phases before the round t' where $\hat{i}_{*,t_2}=i_{t'}$.*

Lemma 5.7 *If $S_{t_1}\neq\hat{S}_{*,t_1}$ is selected at round t_1 and Eq.6 is hold at round $t_2=t_1+\gamma_{S_{t_1}}(t_1)$, at least one base arm $i\notin\hat{S}_{*,t_2}$ will be pulled more than 1 phases before the round t' where $\hat{S}_{*,t_2}=S_{t'}$.*

Lemma 5.6 and 5.7 shows that the *predictive selection* technique predicts whether there exists any arm or base arm will be pulled more than one phases before we exploit the empirical optimal (super) arm. Taking this into consideration, We can directly explore the current selected (super) arm to reduce potential switching cost, which also have a large empirical estimation that creates less regret.

Theorem 5.8 *With the probability larger than $1-Kn^{-4}$ in MAB and $1-Kn^{-3}$ in CMAB, the marginal loss upper bound of MSMR algorithms is $\mathcal{O}((\log n)^{\varepsilon}+\lambda(\log n)^{1-\varepsilon})$.*

In most cases, λ is constant, and setting $\varepsilon=0.5$ yields the theoretical minimum marginal loss $\mathcal{O}(\lambda\sqrt{\log n})$. When λ depends on n, the optimal ε can be derived, e.g., for $\lambda=\sqrt{\log n}$, the optimal choice is $\varepsilon=2/3$. From a broader perspective, if the parameter $\varepsilon\in[0,1]$, our proposed MSMR framework encompasses two representative baseline algorithms as special cases. Taking the CMAB setting (Algorithm 1) as an example (ignoring the initial concentrated exploration phase, i.e., lines 2–4), we observe the following limiting cases: When $\varepsilon=0$, $M=1$, and $N=1$, the algorithm degenerates into standard CUCB, which incurs high switching cost. When $\varepsilon=1$, $M=1$, $N=\kappa$, and $\gamma_{S_t}(t)=\kappa t$, it reduces to Phased-CUCB, which typically suffers from higher regret. The MSMR framework achieves a better balance between regret and switching cost, lying between these extremes.

6 Numerical Simulations

In this section, we present experiments to assess the performance of our algorithms on both synthetic and real-world datasets. Each experiment was conducted over 20 independent trials to ensure reliability, with

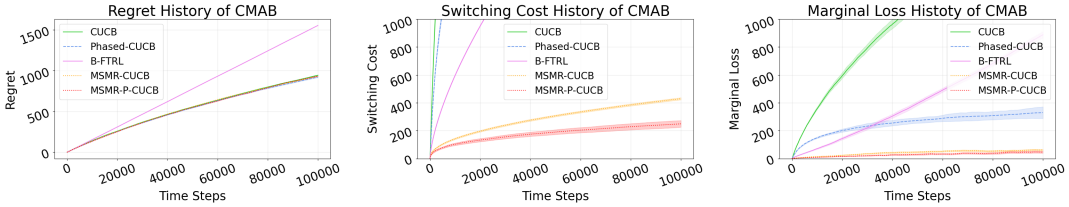
$n = 100000$, $N = 1$, $M = 5$ for all bandit settings and $\alpha = 0.95$, $\beta = 1$ for CMAB. The tests were performed on a macOS system equipped with an Apple M3 Pro processor and 18 GB of RAM. Here, we present only the experiments for the CMAB setting. For results on MAB settings, ablation studies, and real-world datasets, please refer to Appendix I.

6.1 Experiment Setup

Data Generation. We conduct experiments on cascading bandits, a specific instance of CMAB, comparing against algorithms CUCB (6), phased-CUCB(16), B-FTRL(11). The objective is to select $m = 5$items from a set of $K = 20$ to maximize the reward. We give a very similar reward distribution where $\mu_i = 0.3 + 0.002 \times i$. In each round t, a list $S_t = (a_{t,1}, \ldots, a_{t,m}) \subseteq [\![K]\!]$ is randomly selected. The outcome $X_{t,i}$ for each $i \in S_t$ is generated from a Bernoulli distribution with mean μ_i. Given the ranked list S_t, if stopping at the j_t-th item, the observed outcomes are: $(X_{t,a_1}, \ldots, X_{t,a_k}) = (0, \ldots, 0, 1, x, \ldots, x)$, where the first $j_t - 1$ items are 0, the j_t-th item is 1, and the rest are unobserved (x). If the list is exhausted, the observed outcomes are: $(X_{t,a_1}, \ldots, X_{t,a_k}) = (0, 0, \ldots, 0)$. The reward is 1 for stopping and 0 for exhausting the list. The reward function can be written as $r(S_t; \boldsymbol{\mu}) = 1 - \prod_{i \in S_t}(1 - \mu_i)$.

6.2 Experimental Results

Regret, Switching Cost and Marginal Loss. In Figure 1(a), we observe that the regret of MSMR and MSMR-P closely matches that of the standard baseline methods. In contrast, B-FTRL exhibits noticeably higher regret. Figure 1(b) further shows that standard and phased methods suffer from a substantial number of switches, often exceeding several thousand. B-FTRL also incurs a significant number of switches. In comparison, MSMR results in only 432 in the CMAB setting, amounting to merely 2.4% of the switches incurred by CUCB. Moreover, MSMR-P achieves even greater savings, reducing switching to just 1.3% of CUCB—representing a nearly 50% reduction in switching cost compared to MSMR. These results highlight the effectiveness of the *predictive selection* technique. In terms of marginal loss, figure 1(c) further shows that the MSMR framework achieves remarkably low loss compared to the best existing algorithm, significantly outperforming all other methods.



(a) Regret (b) Switching Cost (c) Marginal Loss

Figure 1: Synthetic Experiments on CMAB

7 Conclusion

This paper introduces a novel bandit framework that achieves minimal switching cost and minimal marginal regret, effectively addressing the trade-off between switching costs and regret in bandit algorithms. We develop general techniques—*initial concentrated exploration*, *near-optimal exploitation*, and *predictive selection*, which are broadly applicable to MAB and CMAB settings. Through rigorous theoretical analysis, we establish that these techniques guarantee only $\mathcal{O}((\log n)^{1-\varepsilon})$ switching cost while incurring negligible marginal regret, thereby achieving only $\mathcal{O}(\lambda(\sqrt{\log n})$ marginal loss. Empirical results further demonstrate that MSMR algorithms perform only a few hundred switches, merely 1.3% of those made by standard methods, highlighting the significant advantage of MSMR over existing algorithms. Besides, this paper only provides some theories for part of the bandit scenarios. We hope to extend this framework to more scenarios such as linear or constrained bandit in the future.

REFERENCES

[1] Idan Amir, Guy Azov, Tomer Koren, and Roi Livni. Better best of both worlds bounds for bandits with switching costs. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

[2] Raman Arora, Teodor V. Marinov, and Mehryar Mohri. *Bandits with feedback graphs and switching costs*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[3] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002.

[4] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

[5] Ivan Cantador, Peter Brusilovsky, and Tsvi Kuflik. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys '11, page 387–388, 2011.

[6] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159. PMLR, 2013.

[7] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: general framework, results and applications. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page I–151–I–159. JMLR.org, 2013.

[8] Wei Chen, Yajun Wang, Yang Yuan, and Qinshi Wang. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *J. Mach. Learn. Res.*, 17(1):1746–1778, January 2016.

[9] Xiangxiang Dai, Xutong Liu, Jinhang Zuo, Hong Xie, Carlee Joe-Wong, and John C. S. Lui. Variance-aware bandit framework for dynamic probabilistic maximum coverage problem with triggered or self-reliant arms. *IEEE Transactions on Networking*, pages 1–12, 2025.

[10] Xiangxiang Dai, Zhiyong Wang, Jize Xie, Tong Yu, and John CS Lui. Online learning and detecting corrupted users for conversational recommendation systems. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):8939–8953, 2024.

[11] Ofer Dekel, Jian Ding, Tomer Koren, and Yuval Peres. Bandits with switching costs: T2/3 regret. STOC '14, page 459–467, New York, NY, USA, 2014. Association for Computing Machinery.

[12] Devdatt Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, USA, 1st edition, 2009.

[13] Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:7340–7348, 05 2021.

[14] Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. *Batched multi-armed bandits problem*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[15] T.L Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Adv. Appl. Math.*, 6(1):4–22, March 1985.

[16] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[17] Zhuohua Li, Maoli Liu, Xiangxiang Dai, and John CS Lui. Towards efficient conversational recommendations: Expected value of information meets bandit learning. In *Proceedings of the ACM on Web Conference 2025*, pages 4226–4238, 2025.

[18] Xutong Liu, Xiangxiang Dai, Xuchuang Wang, Mohammad Hajiesmaili, and John Lui. Combinatorial logistic bandits. *arXiv preprint arXiv:2410.17075*, 2024.

[19] Xutong Liu, Jinhang Zuo, Xiaowei Chen, Wei Chen, and John CS Lui. Multi-layered network exploration via random walks: From offline optimization to online learning. In *International Conference on Machine Learning*, pages 7057–7066. PMLR, 2021.

[20] Xutong Liu, Jinhang Zuo, Siwei Wang, Carlee Joe-Wong, John Lui, and Wei Chen. Batch-size independent regret bounds for combinatorial semi-bandits with probabilistically triggered arms or independent arms. In *Advances in Neural Information Processing Systems*, 2022.

[21] Nadav Merlis and Shie Mannor. Batch-size independent regret bounds for the combinatorial multi-armed bandit problem. In *Conference on Learning Theory*, pages 2465–2489. PMLR, 2019.

[22] Chloé Rouyer, Yevgeny Seldin, and Nicolò Cesa-Bianchi. An algorithm for stochastic and adversarial bandits with switching costs. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9127–9135. PMLR, 18–24 Jul 2021.

[23] David Simchi-Levi and Yunzong Xu. *Phase transitions and cyclic phenomena in bandits with switching constraints*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[24] William R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25:285–294, 1933.

[25] Qinshi Wang and Wei Chen. Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications. In *Advances in Neural Information Processing Systems*, pages 1161–1171, 2017.

[26] Peiyan Zhang, Jiayan Guo, Chaozhuo Li, Yueqi Xie, Jae Boum Kim, Yan Zhang, Xing Xie, Haohan Wang, and Sunghun Kim. Efficiently leveraging multi-level user intent for session-based recommendation via atten-mixer network. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, WSDM '23, page 168–176, New York, NY, USA, 2023. Association for Computing Machinery.

[27] Jinhang Zuo and Carlee Joe-Wong. Combinatorial multi-armed bandits for resource allocation. In *2021 55th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–4. IEEE, 2021.

A NOTATION

Table 2: Notation.

Alphabet	Meanings
$c_{i,T_i(t)}$	confidence interval for arm i in round t
$C(n)$	switching cost
$f(\cdot)$	bounded smoothness function
F_t	oracle fails to produce an α approximate answer with respect to input vector $\bar{\boldsymbol{\mu}}_t$ in round t
i	arm in MAB or base arm in CMAB
$i_{S,t}$	arm belong to S with lowest pulled time up to round T
i_t	arm selected in round t
i_*	optimal arm
$\hat{i}_*$	arm with largest empirical mean reward
$\mathbb{I}(\cdot)$	indication function
K	number of arms
K_i	number of sub-optimal supers arm containing i
K_i'	number of sub-optimal super arms containing i indeed pulled in transition stage
$L(\cdot)$	$M \times \gamma(\cdot)$
$L_{i,j}$	length of j_{th} phase for arm i
$\ell(\cdot)$	function in CUCB(6)
M	hyperparameter in MSMR algorithms
n	time horizon
N	constant in $\gamma(\cdot)$
N_i	counter for base arm i
$N_{i,t}$	counter for base arm i after round t
$N_{i,t}^l$	increasing counter for base arm i due to $\mathcal{S}_{i,B}^l$ after round t
$N_{i,t}^{l,suf}$	increasing counter for base arm i due to $\mathcal{S}_{i,B}^l$ after round t in sufficiently sampled stage
$N_{i,t}^{l,und}$	increasing counter for base arm i due to $\mathcal{S}_{i,B}^l$ after round t in under-sampled stage
$\mathcal{N}_t$	process is nice at round t
$\text{opt}_{\boldsymbol{\mu}}$	expect reward for S_*
$r_{\boldsymbol{\mu}}(S)$	expect reward function for S
$R(S)$	reward function for S
$R^A(\lambda, n)$	marginal loss in time horizon n
$Reg(n)$	regret in time horizon n
$Reg_u(n)$	regret in under-sampled stage in time horizon n
$Reg_t(n)$	regret in transition stage in time horizon n
$Reg_s(n)$	regret in sufficiently sampled stage in time horizon n
S	super arm
S_t	super arm selected at round t
S_*	optimal super arm
$\mathcal{S}_{i,B}^j$	sub-optimal super arms containing arm i with j_{th} lowest expected reward
$\hat{S}_{*,t}$	super arm with largest reward under $r_{\hat{\boldsymbol{\mu}}}(\cdot)$ at round t
$\mathcal{S}$	set of super arms
$\mathcal{S}_{i,B}$	set of all sub-optimal super arms containing arm i
t	time round
$t_{u,i}$	number of phases for selecting arm i in under-sampled stage
$t_{t,i}$	number of phases for selecting arm i in transition stage

Continued on next page

Table 2 – continued from previous page

$t_{s,i}$	number of phases for selecting arm i in sufficiently sampled stage
t_s	number of phases for selecting any sub-optimal super arm in sufficiently sampled stage
$T_i(t)$	number of times arm i has been pulled up to round t
$W_{i,s}$	sum of $L_{i,1}$ to $L_{i,s}$
$X_{i,t}$	reward feedback for arm i in round t
$\boldsymbol{y}_i$	item vector for item i
α	parameter in (α, β) -approximation oracle
β	parameter in (α, β) -approximation oracle
$\delta_{i,k}$	regret gap of k_{th} sub-optimal super arm containing i indeed pulled in transition stage
Δ_i	regret gap for arm i
$\Delta^{i,j}$	regret gap for $S^j_{i,B}$
$\Delta^i_{\min}$	sub-optimal super arm with smallest regret gap containing i
$\Delta^i_{\max}$	sub-optimal super arm with largest regret gap containing i
$\Delta_{\max}$	sub-optimal super arm with largest regret gap
ε	index in $\gamma(\cdot)$
$\boldsymbol{\theta}_u$	user preference vector for user u
$\gamma(\cdot)$	exploitation function
$\Lambda_{i,t}$	confidence interval for base arm i in round t in CMAB
Λ_t	largest confidence interval among all base arm in round t in CMAB
$\Lambda^{i,l}$	$\sqrt{3\ln(n)/2\ell(\Delta^{i,l})}$
μ_i	expected reward for arm i
μ_*	expected reward for arm i_*
$\hat{\mu}_{i,t}$	empirical mean reward for arm i in round t
$\bar{\mu}_{i,t}$	upper confidence estimate for arm i in round
$\bar{\mu}'_{i,t}$	$\hat{\mu}_{i,t} + \sqrt{2\ln(n)/T_i(t)}$ (MAB) or $\hat{\mu}_{i,t} + \sqrt{3\ln(n)/2T_i(t)}$ (CMAB)
$\boldsymbol{\mu}$	expected reward vector
$\hat{\boldsymbol{\mu}}$	empirical mean reward vector
$\bar{\boldsymbol{\mu}}$	upper confidence estimate vector
τ_i	the start round of i_{th} phase in sufficiently sampled stage of CUCB
$\zeta(\cdot)$	Riemann Zeta Function

B Preliminary Knowledge and Notations

We use $[\![K]\!]$ to denote the set $\{1, 2, \ldots, K\}$ for $\forall K \in \mathbb{N}^+$.

Fact B.1 *(12) (Chernoff-Hoeffding bound). Let $X_1, \cdots, X_n$ be random variables with common support $[0, 1]$ and $\mathbb{E}[X_i] = \mu$. Let $S_n = X_1 + \cdots + X_n$. Then for all $t \geq 0$,*

$$P[S_n \geq n\mu + t] \leq e^{-2t^2/n}, P[S_n \leq n\mu - t] \leq e^{-2t^2/n} \tag{10}$$

C Supplementary Algorithm

Algorithm 2 MSMR-UCB Algorithm

Input: Time horizon n, constant M, α, function $\gamma(\cdot)$

1: $t \leftarrow 1$, $\hat{\mu}_i \leftarrow 1$ for all i
2: **for** all arm $i \in [\![K]\!]$ **do**
3: Play i and observe $X_{i,t}$, update $T_i(t) \leftarrow T_i(t) + 1$, $\hat{\mu}_{i,t} \leftarrow \frac{T_i(t-1)\cdot\hat{\mu}_{i,t-1}+X_{i,t}}{T_i(t)}$, $\bar{\mu}_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{2\ln t/T_i(t)}$, $\bar{\mu}'_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{2\ln n/T_i(t)}$ until $\bar{\mu}'_{i,t} \leq 1$ and update t
4: **end for**
5: **while** $t \leq n$ **do**
6: $i_t \leftarrow \mathrm{argmax}_{j\in[\![K]\!]}\bar{\mu}_j$
7: $Z \leftarrow \gamma_{i_t}(t)$
8: $\hat{i}_{*,t} \leftarrow \mathrm{argmax}_{j\in[\![K]\!]}\hat{\mu}_j$
9: $Z \leftarrow M \cdot \gamma_{i_t}(t)$ when $\hat{i}_{*,t} = i_t$
10: Play arm i_t $\min\{Z, n - t\}$times
11: Update t and $T_i(t)$, $\hat{\mu}_{i,t}$, $\bar{\mu}_{i,t}$ for all arms
12: **end while**

D Proof of Corollary 5.1

D.1 MAB Case

By Lemma B.1, if $T_i(t) > l_i$, we have $P(\hat{\mu}_{i,t} - \mu_i - \Delta_i/2 > 0) \leq exp(-2l_i(\Delta_i/2)^2) \leq n^{-4}$, and the following inequality:

Algorithm 3 MSMR-P-UCB Algorithm

Input: Time horizon n, constant M, α, function $\gamma(\cdot)$

1: $t \leftarrow 1$, $\hat{\mu}_i \leftarrow 1$ for all i
2: **for** all arm $i \in [\![K]\!]$ **do**
3: Play i and observe $X_{i,t}$, update $T_i(t) \leftarrow T_i(t)+1$, $\hat{\mu}_{i,t} \leftarrow \frac{T_i(t-1)\cdot\hat{\mu}_{i,t-1}+X_{i,t}}{T_i(t)}$, $\bar{\mu}_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{2\ln t/T_i(t)}$, $\bar{\mu}'_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{2\ln n/T_i(t)}$ until $\bar{\mu}'_{i,t} \leq 1$ and update t
4: **end for**
5: **while** $t \leq n$ **do**
6: $i_t \leftarrow \text{argmax}_{j\in[\![K]\!]}\bar{\mu}_j$
7: $Z \leftarrow \gamma_{i_t}(t)$
8: $\hat{i}_{*,t} \leftarrow \text{argmax}_{j\in[\![K]\!]}\hat{\mu}_j$
9: $Z \leftarrow M \cdot \gamma_{i_t}(t)$ when $\hat{i}_{*,t} = i_t$
10: Play arm i_t $\min\{Z, n-t\}$times
11: Update t and $T_i(t)$, $\hat{\mu}_{i,t}$, $\bar{\mu}_{i,t}$ for all arms
12: $\hat{i}_{*,t} \leftarrow \text{argmax}_{j\in[\![K]\!]}\hat{\mu}_j$
13: $t_3 \leftarrow t + \sum_{j\neq i_t,\hat{i}_{*,t}} \gamma_j(t)$
14: **if** $\sqrt{2\ln(t)/T_{i_t}(t)} + \hat{\mu}_{i_t,t} \geq \sqrt{2\ln(t_3)/T_{\hat{i}_{*,t}}(t)} + \hat{\mu}_{\hat{i}_{*,t},t}$ **then**
15: **goto** Line 10
16: **end if**
17: **end while**

$$P(\exists i : T_i(t) \geq l_i, \bar{\mu}'_{i,t} \geq 1) \leq \bigcup_{i\neq i_*} P(T_i(t) \geq l_i, \hat{\mu}_{i,t} + \sqrt{2\ln n/T_i(t)} \geq 1) \tag{11}$$

$$\leq \bigcup_{i\neq i_*} P(\hat{\mu}_{i,t} + \sqrt{2\ln n/l_i} \geq 1) \tag{12}$$

$$= \bigcup_{i\neq i_*} P(\hat{\mu}_{i,t} + \Delta_i/2 \geq 1) \tag{13}$$

$$= \bigcup_{i\neq i_*} P(\hat{\mu}_{i,t} - \mu_i - \Delta_i/2 \geq 1 - \mu_*) \tag{14}$$

$$\leq \bigcup_{i\neq i_*} P(\hat{\mu}_{i,t} - \mu_i - \Delta_i/2 \geq 0) \tag{15}$$

$$\leq Kn^{-4} \tag{16}$$

which means with a probability larger than $1 - Kn^{-4}$, initial concentrated exploration will occur in the under-sampled stage, which is calculated in the total regret.

D.2 CMAB CASE

By Lemma B.1, if $T_i(t) > \ell_n(\Delta^{i,1})$, we have $P(\hat{\mu}_{i,t} - \mu_i - f^{-1}(\Delta^{i,1})/2 > 0) \leq exp(-2\ell_n(\Delta^{i,1})(f^{-1}(\Delta^{i,1})/2)^2) \leq n^{-3}$, and the following inequality:

Algorithm 4 MSMR-CUCB Algorithm

Input: Time horizon n, constant M, α, function $\gamma(\cdot)$

1: $t \leftarrow 1$, $\hat{\mu}_i \leftarrow 1$ for all i
2: **while** Exists i makes $T_i(t) = 0$ **do**
3: $\{\hat{\mu}_{j_1}, \hat{\mu}_{j_2}, \ldots, \hat{\mu}_{j_K}\} \leftarrow$ Sort base arm by $\hat{\mu}_i$ in a decreasing way
4: $S \leftarrow \{j_1, j_2, \ldots, j_m\}$
5: Play S and observe $X_{i,t}$, for any $i \in S$, update $T_i(t) \leftarrow T_i(t) + 1$, $\hat{\mu}_{i,t} \leftarrow \frac{T_i(t-1)\cdot\hat{\mu}_{i,t-1}+X_{i,t}}{T_i(t)}$, $\bar{\mu}_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{3\ln t/2T_i(t)}$, $\bar{\mu}'_{i,t} \leftarrow \hat{\mu}_{i,t} + \sqrt{3\ln n/2T_i(t)}$ until $\min_{j\in S} \bar{\mu}'_{j,t} \leq 1$ and update t
6: **end while**
7: **while** $t \leq n$ **do**
8: $S_t \leftarrow \text{argmax}_{S\in\mathcal{S}} r_{\bar{\boldsymbol{\mu}}}(S)$, $Z \leftarrow \gamma_{S_t}(t)$
9: $\hat{S}_{*,t} \leftarrow \text{argmax}_{S\in\mathcal{S}} r_{\hat{\boldsymbol{\mu}}}(S)$, $Z \leftarrow M \cdot \gamma_{S_t}(t)$ when $r_{\hat{\boldsymbol{\mu}}}(S_t) \geq \alpha \cdot r_{\hat{\boldsymbol{\mu}}}(\hat{S}_{*,t})$
10: Play super arm S_t $\min\{Z, n-t\}$times
11: Update t and $T_i(t)$, $\hat{\mu}_{i,t}$, $\bar{\mu}_{i,t}$ for all base arms
12: $\hat{S}_{*,t} \leftarrow \text{argmax}_{S\in\mathcal{S}} r_{\hat{\boldsymbol{\mu}}}(S)$
13: $t_3 \leftarrow t + \sum_{j\in\{[\![K]\!]\setminus(S_t \bigcup \hat{S}_{*,t})\}} \gamma_j(t)$
14: **if** $r(\hat{\boldsymbol{\mu}}_t, \boldsymbol{c}_t, S_t) \geq r(\hat{\boldsymbol{\mu}}_t, \boldsymbol{c}_{t_3}, \hat{S}_{*,t})$ **and** $t < n$ **then**
15: **goto** Line 12
16: **end if**
17: **end while**

$$P(\exists i : T_i(t) \geq \ell_n(\Delta^{i,1}), \bar{\mu}'_{i,t} \geq 1) \leq \bigcup_{i\in[\![K]\!]} P(T_i(t) \geq \ell_n(\Delta^{i,1}), \hat{\mu}_{i,t} + \sqrt{3\ln n/2T_i(t)} \geq 1) \tag{17}$$

$$\leq \bigcup_{i\in[\![K]\!]} P(\hat{\mu}_{i,t} + \sqrt{3\ln n/2\ell_n(\Delta^{i,1})} \geq 1) \tag{18}$$

$$= \bigcup_{i\in[\![K]\!]} P(\hat{\mu}_{i,t} + f^{-1}(\Delta^{i,1})/2 \geq 1) \tag{19}$$

$$= \bigcup_{i\in[\![K]\!]} P(\hat{\mu}_{i,t} - \mu_i - f^{-1}(\Delta^{i,1})/2 \geq 1 - \mu_i - f^{-1}(\Delta^{i,1})) \tag{20}$$

$$\leq \bigcup_{i\in[\![K]\!]} P(\hat{\mu}_{i,t} - \mu_i - f^{-1}(\Delta^{i,1})/2 \geq 1 - \mu_{\text{argmax}_{j\in S_*}|\mu_j-\mu_i|}) \tag{21}$$

$$\leq \bigcup_{i\in[\![K]\!]} P(\hat{\mu}_{i,t} - \mu_i - f^{-1}(\Delta^{i,1})/2 \geq 0) \tag{22}$$

$$\leq \bigcup_{i\neq i_*} P(\hat{\mu}_{i,t} - \mu_i - \Delta_i/2 \geq 0) \tag{23}$$

$$\leq Kn^{-3} \tag{24}$$

which means with a probability larger than $1 - Kn^{-3}$, initial concentrated exploration will occur in the under-sampled stage, which is calculated in the total regret.

E PROOF OF REGRET

E.1 PROOF OF THEOREM 5.2

Let $[\![K]\!]$ denotes the set of arms. For each arm $i \in [\![K]\!]$, pulling it at round t yields a reward feedback $X_{i_t,t} \in [0,1]$. The unknown mean vector is represented by $\boldsymbol{\mu} = (\mu_1, ..., \mu_K)$, where $\mu_i = \mathbb{E}[X_{i,t}]$ denotes the expected reward for any base arm i. Besides, we define $\Delta_i = \mu_* - \mu_i$. The arm with the highest expected reward is called the optimal arm, and its mean reward is denoted by $\mu_* = \max_{i \in [\![K]\!]} \mu_i$. The objective of the MAB problem is to identify this optimal arm while minimizing regret, which is defined as:

$$Reg(n) = n\mu_* - \mathbb{E}[\sum_{t=1}^{n} X_{i_t,t}], \tag{25}$$

where n is the total time horizon, and $X_{i_t,t}$ represents the reward from the chosen arm at round t.

In MAB settings, we set that $\gamma_i(t) = N(T_{i_t}(t))^{\varepsilon}$ is the number of pulling times for the selected arm i_t, where N is a constant, $0 < \varepsilon < 1$ is a hyperparameter. If the selected arm $i_t = \mathrm{argmax}_{j \in [\![K]\!]} \bar{\mu}_{j,t}$ has the largest upper confidence estimation, which is $\hat{i}_* = \mathrm{argmax}_{j \in [\![K]\!]} \hat{\mu}_{j,t}$ and $i_t = \hat{i}_*$, agent assume that arm i_t has the relatively large empirical estimation. Under this condition, we execute the near-optimal exploitation and set $M \cdot \gamma_i(t) = MN(T_{i_t}(t))^{\varepsilon}$ as the exploitation times, where M is a constant. We decompose the regret of MSMR-UCB as three stages: (1) under-sampled stages; (2) transition stages; (3) sufficiently sampled stages.

Under-sampled stages. This stage is a fixed regret for a certain arm in expectation. According to (15), we know that each arm i must have some inevitable exploring regret in UCB-based methods, which is $\mathcal{O}(\log n)$. In MSMR-UCB, we set under-sampled regret number of arm i is $l_i = 8\ln(n)/\Delta_i^2$ and arm i is under-sampled or sufficiently sampled if $T_i(t) \leq l_i$ or $T_i(t) \geq l_i$. By our definition, the regret of under-sampled stage is at most:

$$Reg_u(n) = \sum_{i \neq i_*} l_i \Delta_i \leq \sum_{i \neq i_*} \frac{8 \ln n}{\Delta_i}. \tag{26}$$

Transition stages. When transiting from under-sampled stages to the sufficiently sampled cases, we may pull some arm i more times. By the definition of and near-optimal exploitation, we at most pull arm i at most $M\gamma_i(t)$ more times. The regret of transition stages is at most:

$$\begin{aligned} Reg_t(n) &\leq \sum_{i \neq i_*} M\gamma_i(t)\Delta_i & (27)\\ &\leq \sum_{i \neq i_*} MN(l_i)^{\varepsilon}\Delta_i & (28)\\ &\leq \sum_{i \neq i_*} MN(8\ln n)^{\varepsilon}\Delta_i^{1-2\varepsilon}. & (29) \end{aligned}$$

Sufficiently sampled stages. We define arm i as sufficiently sampled if $T_i(t) \geq l_i$. At sufficiently sampled stages, each suboptimal arm has been pulled enough times. In this situation, we have to get much information for any suboptimal arm i, it has a low probability to choose a suboptimal arm i. Specifically, denoting $\bar{\mu}_{i,T_i(t)}$

is the upper confidence estimate of arm t in round t, we have the following inequality:

$$\mathbb{E}[T_i(n)] = \sum_{t=l+1}^{n} P(\bar{\mu}_{i,T_i(t)} = \max_{j\in[\![K]\!]} \bar{\mu}_{j,T_j(t)}, T_i(t) \geq l)(\gamma_i(t) + (M-1)\gamma_i(t)P(i = \hat{i}_*)) \tag{30}$$

$$\leq \sum_{t=l+1}^{n} P(\bar{\mu}_{i,T_i(t)} = \max_{j\in[\![K]\!]} \bar{\mu}_{j,T_j(t)}, T_i(t) \geq l) \cdot M\gamma_i(t) \tag{31}$$

$$\leq \sum_{t=l+1}^{n} P(\bar{\mu}_{*,T_*(t)} \leq \bar{\mu}_{i,T_i(t)}, T_i(t) \geq l) \cdot M\gamma_i(t) \tag{32}$$

$$\leq \sum_{t=l+1}^{n} P(\min_{1<s<t} \bar{\mu}_{*,T_*(t)} \leq \max_{l\leq s_i\leq t} \bar{\mu}_{i,T_i(t)}) \cdot M\gamma_i(t) \tag{33}$$

$$\leq \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=l}^{t-1} P(\bar{\mu}_{*,s} \leq \bar{\mu}_{i,s_i}) \cdot M\gamma_i(t) \tag{34}$$

$$\leq \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=l}^{t-1} (P(\hat{\mu}_{*,s} \leq \mu_* - c_{*,s}) + P(\mu_i + c_{i,s_i} \leq \hat{\mu}_{i,s_i})) \cdot M\gamma_i(t) \tag{35}$$

$$\leq \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=l}^{t-1} \frac{2}{t^4} \cdot M\gamma_i(t) \tag{36}$$

$$\leq \sum_{t=1}^{\infty} \frac{2}{t^2} \cdot M\gamma_i(t) \tag{37}$$

$$\leq \sum_{t=1}^{\infty} \frac{2}{t^2} MNt^{\varepsilon} \tag{38}$$

$$= 2MN \cdot \zeta(2-\varepsilon) \tag{39}$$

$$\tag{40}$$

So, the regret of sufficiently sampled stage is:

$$Reg_s(n) = \sum_{i\neq i_*} 2MN \cdot \zeta(2-\varepsilon)\Delta_i \tag{41}$$

In summary, the regret upper bound is:

$$Reg(n) = Reg_u(n) + Reg_t(n) + Reg_s(n) \tag{42}$$

$$\leq \sum_{i\neq i_*} \left(\frac{8\ln n}{\Delta_i} + MN(8\ln(n))^{\varepsilon}\Delta_i^{1-2\varepsilon} + 2MN \cdot \zeta(2-\varepsilon)\Delta_i \right). \tag{43}$$

E.2 Proof of Theorem 5.3

In CMAB settings, we choose the super arm S_t who has the largest upper confidence estimation, which is $S_t = \text{argmax}_{S\in\mathcal{S}} r_{\bar{\boldsymbol{\mu}}}(S)$. We set that $\gamma_S(t) = N(T_{i_{S_t,t}}(t))^{\varepsilon}$ is the number of pulling times for the selected super arm S_t, where M is a constant, $0 < \varepsilon < 1$ is a hyperparameter and $i_{S,t} = \text{argmin}_{j\in S} T_j(t)$. Based

on (α, β)-approximation and defining $\hat{S}_* = \text{argmax}_{S \in \mathcal{S}} r_{\hat{\boldsymbol{\mu}}}(S)$, if $r_{\hat{\boldsymbol{\mu}}}(S_t) \geq \alpha \cdot r_{\hat{\boldsymbol{\mu}}}(\hat{S}_*)$, we think S_t has relatively large empirical estimation. Under this condition, we execute the near-optimal exploitation and set $M \cdot \gamma(t) = MN(T_{i_{S_t,t}}(t))^{\varepsilon}$ as the exploitation times, where N is a constant.

In the (α, β)-approximation oracle setting, a super arm S is considered *sub-optimal* if $r_{\boldsymbol{\mu}}(S) < \alpha \cdot \text{opt}_{\boldsymbol{\mu}}$. Let $\mathcal{S}_{i,B}$ be the set of all sub-optimal super arms containing base arm i. We sort all sub-optimal super arms in $\mathcal{S}_{i,B}$ as $S^1_{i,\text{B}}, S^2_{i,\text{B}}, \ldots, S^{K_i}_{i,\text{B}}$, in increasing order of their expected rewards, where K_i the number of sub-optimal super arms containing base arm i. Define the regret gap for the j-th sub-optimal super arm as: $\Delta^{i,j} = \alpha \cdot \text{opt}_{\boldsymbol{\mu}} - r_{\boldsymbol{\mu}}(S^{\boldsymbol{j}}_{\boldsymbol{i},\text{B}})$ and denote $\Delta^i_{\max} = \Delta^{i,1}$, $\Delta^i_{\min} = \Delta^{i,K_i}$ and $\Delta_{\max} = \max_{i \in [\![K]\!]} \Delta^i_{\max}$. And for each $S^l_{i,\text{B}}$, we define sufficient sampling of i with respect to $S^l_{i,\text{B}}$ as i being sampled $\ell_n(\Delta^{i,l}) = (6 \ln n)/(f^{-1}(\Delta^{i,l}))^2$times.

For the proof, we maintain counter N_i for each arm i. Let $N_{i,t}$ be the value of N_i after the t-th round and $N_{i,0} = 0$. Counters $\{N_i\}_{i=1}^K$ are updated in the following way: For a round $t > K$, let S_t be the super arm selected in round t by the MSMR-CUCB. Round t is a bad round if the oracle selects a super arm $S_t \in \mathcal{S}_{\text{B}}$, which is not an α-approximate super arm with respect to the true mean vector μ. If S_t is chosen and round t is bad, we increment $N_{i_{S_t,t}}$ by one, i.e., $N_{i_{S_t,t},t} = N_{i_{S_t,t},t-1} + 1$. In other words, we find the arm $i_{S_t,t}$ with the smallest counter in S_t and increase its counter. If $i_{S_t,t}$ is not unique, we pick an arbitrary arm with the smallest counters in S_t. By definition, we know $N_{i_{S_t,t},t} \leq T_i(t)$. Notice that in every bad round, exactly one counter in $\{N_i\}_{i=1}^K$ is increased. Each time N_i gets updated, one of the sub-optimal super arms containing i is played. We further divide counter N_i into more counters $\{N^l_i\}_{l=1}^{K_i}$, whose value at round n, $N^l_{i,n}$ is defined as follows:

$$\forall l \in [\![K_i]\!], N^l_{i,n} = \sum_{t=K+1}^{n} \mathbb{I}\{S_t = S^l_{i,\text{B}}, N_{i,t} > N_{i,t-1}\} \tag{44}$$

Notice that every arm in S_t must have been played at least $N_{i,t-1}$ times by round t, since in our updating rule we choose the smallest counter value among arms in S_t to update, and i is the chosen one. If $N_{i,t-1} > \ell_n(\Delta^{i,l})$, we say that the bad arm $S^l_{i,\text{B}}$ is sufficiently sampled. Otherwise, it is under-sampled. For our proof, we depart the $N^l_{i,n}$ into two parts: sufficiently sampled parts $N^{l,suf}_{i,n}$ and under-sampled parts $N^{l,Sund}_{i,n}$.

$$N^{l,suf}_{i,n} = \sum_{t=K+1}^{n} \mathbb{I}\{S_t = S^l_{i,\text{B}}, N_{i,t} > N_{i,t-1}, N_{i,t-1} > \ell_n(\Delta^{i,l})\} \tag{45}$$

$$N^{l,und}_{i,n} = \sum_{t=K+1}^{n} \mathbb{I}\{S_t = S^l_{i,\text{B}}, N_{i,t} > N_{i,t-1}, N_{i,t-1} \leq \ell_n(\Delta^{i,l})\} \tag{46}$$

Then we have $N_{i,n} = \sum_{l \in [K_i]} (N^{l,suf}_{i,n} + N^{l,und}_{i,n})$. Using this notation, the total reward at time horizon n is at least

$$n \cdot \alpha \cdot \text{opt}_{\boldsymbol{\mu}} - \mathbb{E}\left[\sum_{i \in [\![K]\!]} \left(\sum_{l \in [\![K_i]\!]} (N^{l,suf}_{i,n} + N^{l,und}_{i,n}) \cdot \Delta^{i,l} \right)\right], \tag{47}$$

where $\Delta^{i,1}$ is for the initialization. By the regret definition of (α, β) -approximation oracle, the regret of MSMR-CUCB can be written as:

$$(\beta - 1) \cdot n \cdot \alpha \cdot \text{opt}_{\boldsymbol{\mu}} + \mathbb{E}\left[\sum_{i \in [\![K]\!]} \left(\sum_{l \in [\![K_i]\!]} (N^{l,suf}_{i,n} + N^{l,und}_{i,n}) \cdot \Delta^{i,l} \right)\right]. \tag{48}$$

We decompose the regret of MSMR-CUCB as three stages: (1) under-sampled stages; (2) transition stages; (3) sufficiently sampled stages.

Under-sampled stages. In MSMR-CUCB, we set under-sampled regret number of super arm $S_{i,\mathrm{B}}^l$ is $\ell_n(\Delta^{i,l})$ and $S_{i,\mathrm{B}}^l$ is under-sampled if $N_{i,t-1} \leq \ell_n(\Delta^{i,l})$. For a specific arm i, its counter N_i will increase from 1 to $\ell_n(\Delta^{i,K_i})$. To simplify the notation, let $\ell_n(\Delta^{i,0}) = 0$. (Note that $N_{i,K} = 1$ for all i.) Before diving into the details, we briefly explain the essential idea behind Eq. (58). The range of the counter N_i is divided into discrete segments, i.e., $(\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]$ for $j \in [\![K_i]\!]$. Suppose that round t is bad and $N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]$ for some j. Note that we are only interested in cases where S_t is under-sampled. Specifically, this means $S_t = S_B^l$ for some $l > j$. (Otherwise, if S_t is sufficiently sampled, it is based on the counter value $N_{i,t-1}$, and no regret is incurred.)

Consequently, for counter $N_{i,t}$ in the range $(\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]$, the bad super arm will suffer a regret of $\Delta^{i,j}$ each time N_i is incremented, as indicated by Eq. (52). Therefore, the total regret incurred for these under-sampled arms within this interval is at most $(\ell_n(\Delta^{i,j}) - \ell_n(\Delta^{i,j-1})) \cdot \Delta^{i,j}$ (refer to Eq. (56)). We formalize this argument as follows. For any arm $i \in \{i \in [\![K]\!] \mid \Delta_{\min}^i > 0\}$, we have:

$$
\begin{aligned}
Reg_u(n) &= \sum_{l \in [\![K_i]\!]} N_{i,n}^{l,und} \cdot \Delta^{i,l} && (49)\\
&= \sum_{t=K+1} \sum_{l \in [\![K_i]\!]} \mathbb{I}\{S_t = S_{i,\mathrm{B}}^l, N_{i,t} > N_{i,t-1}, N_{i,t-1} \leq \ell_n(\Delta^{i,l})\} \cdot \Delta^{i,l} && (50)\\
&= \sum_{t=K+1}^{n} \sum_{l \in [\![K_i]\!]} \sum_{j=1}^{l} \mathbb{I}\{S_t = S_{i,\mathrm{B}}^l, N_{i,t} > N_{i,t-1}, N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]\} \cdot \Delta^{i,l} && (51)\\
&\leq \sum_{t=K+1}^{n} \sum_{l \in [\![K_i]\!]} \sum_{j=1}^{l} \mathbb{I}\{S_t = S_{i,\mathrm{B}}^l, N_{i,t} > N_{i,t-1}, N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]\} \cdot \Delta^{i,\boldsymbol{j}} && (52)\\
&\leq \sum_{t=K+1}^{n} \sum_{l \in [\![K_i]\!]} \sum_{j \in [\![K_i]\!]} \mathbb{I}\{S_t = S_{i,\mathrm{B}}^l, N_{i,t} > N_{i,t-1}, N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]\} \cdot \Delta^{i,j} && (53)\\
&= \sum_{t=K+1}^{n} \sum_{j \in [\![K_i]\!]} \mathbb{I}\{S_t \in \mathcal{S}_{i,\mathrm{B}}, N_{i,t} > N_{i,t-1}, N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]\} \cdot \Delta^{i,j} && (54)\\
&= \sum_{j \in [\![K_i]\!]} \sum_{t=K+1}^{n} \mathbb{I}\{S_t \in \mathcal{S}_{i,\mathrm{B}}, N_{i,t} > N_{i,t-1}, N_{i,t-1} \in (\ell_n(\Delta^{i,j-1}), \ell_n(\Delta^{i,j})]\} \cdot \Delta^{i,j} && (55)\\
&\leq \sum_{j \in [\![K_i]\!]} (\ell_n(\Delta^{i,j}) - \ell_n(\Delta^{i,j-1})) \cdot \Delta^{i,j} && (56)\\
&= \ell_n(\Delta^{i,K_i})\Delta^{i,K_i} + \sum_{j=1}^{K_i-1} \ell_n(\Delta^{i,j}) \cdot (\Delta^{i,j} - \Delta^{i,j+1}) && (57)\\
&\leq \ell_n(\Delta^{i,K_i})\Delta^{i,K_i} + \int_{\Delta^{i,K_i}}^{\Delta^{i,1}} \ell_n(x)\mathrm{d}x. && (58)
\end{aligned}
$$

Naturally, if we follow the design of MSMR-UCB, each super arm S is pulled an additional $N(T_S(t))^\varepsilon$ times in Algorithm 1, where $T_S(t)$ denotes the number of times S has been selected up to round t. Maintaining $T_S(t)$ for all possible super arms would require tracking up to 2^K different values, which leads to a combinatorial explosion in both computation and storage. To address this issue, we revise the definition of a *sufficiently sampled* super arm to facilitate the analysis of regret during different sampling stages. Specifically, we say that a sub-optimal super arm $\mathcal{S}_{i,\mathrm{B}}^l$ is sufficiently sampled if $T_{i,t-1} > \ell_n(\Delta^{i,l})$. This new definition, which depends only on the sampling counts of individual base arms, is adopted throughout the subsequent analysis. As a result, the algorithm only needs to maintain $T_i(t)$ for each base arm i, thereby avoiding the combinatorial overhead. We now discuss the implications of this modified definition. Under the original definition, the sufficiently sampled component $N_{i,n}^{l,\mathrm{suf}}$ and the under-sampled component $N_{i,n}^{l,\mathrm{und}}$ were disjoint. In contrast, with the new definition, these two components may partially overlap. However, this overlap does not affect the correctness of the regret decomposition: all sources of regret are still accounted for. Furthermore, the regret incurred in the overlapping region is negligible, as the primary contribution to the total regret originates from the under-sampled phase.

Transition stages. When entering the sufficiently sampled phase, a super arm S may be pulled additional times. Under the revised definition of a sufficiently sampled arm, we consider two possible criteria to determine the round marking the transition stage for a base arm i:

- The round t such that $N_{i,t} > \ell_n(\Delta^{i,l})$ and $N_{i,t-1} \leq \ell_n(\Delta^{i,l})$, indicating that the arm has just exited the under-sampled phase.

- The round t such that $T_i(t) > \ell_n(\Delta^{i,l})$ and $T_i(t-1) \leq \ell_n(\Delta^{i,l})$, indicating that the arm has just entered the sufficiently sampled phase.

We adopt the second criterion to define the transition stage. Given a fixed sub-optimal super arm $S_{i,\mathrm{B}}^l$ and its corresponding base arm i, $\ell_n(\Delta^{i,l})$ is fixed. Thus, each such super arm may only be "overpulled" once during the transition stage involving arm i. Prior to this transition, at least one base arm $j \in S$ must satisfy $T_j(t) < \ell_n(\Delta^{i,l})$. Since $T_i(t)$ monotonically increases over time, this guarantees progress toward the transition.

For a given base arm i, we define $\delta_{i,k}$ as the regret gap of the k-th sub-optimal super arm (among those that are actually pulled during the learning process), where $k \in [\![K_i']\!]$ and K_i' is a constant no greater than K_i, representing the number of such super arms that undergo transition stages involving arm i.

When the k-th sub-optimal super arm is pulled, it may incur up to $L(\ell_n(\delta_{i,k})) \triangleq MN(\ell_n(\delta_{i,k}))^\varepsilon$ additional pulls (referred to as "overpulls") during its transition stage. Each overpull contributes at most $\delta_{i,k}$ regret. However, the next regret gap $\delta_{i,k+1}$ has already been accounted for in the under-sampled phase of the $(k+1)$-th sub-optimal super arm. Therefore, the additional regret contribution from the transition stage is bounded by $(\delta_{i,k} - \delta_{i,k+1})L(\ell_n(\delta_{i,k}))$.

The total regret incurred during the transition stage involving base arm i is thus given by:

$$\delta_{i,K_i'} L(\ell_n(\delta_{i,K_i'})) + \sum_{k=1}^{K_i'-1} (\delta_{i,k} - \delta_{i,k+1}) L(\ell_n(\delta_{i,k})) \tag{59}$$

$$= \delta_{i,1} L(\ell_n(\delta_{i,1})) + \sum_{k=2}^{K_i'} \delta_{i,k} \left[L(\ell_n(\delta_{i,k})) - L(\ell_n(\delta_{i,k-1})) \right] \tag{60}$$

$$\leq \delta_{i,1} L(\ell_n(\delta_{i,1})) \tag{61}$$

$$\leq \max_{k \in [\![K_i]\!]} L(\ell_n(\Delta^{i,k})) \Delta^{i,k} \tag{62}$$

$$= \max_{k \in [\![K_i]\!]} MN(\ell_n(\Delta_{i,k}))^{\varepsilon} \Delta^{i,k}. \tag{63}$$

Then the regret upper bound for transition stage is:

$$Reg_t(n) = \sum_{i \in [\![K]\!]} \max_{k \in [\![K_i]\!]} \{MN(\ell_n(\Delta_{i,k}))^{\varepsilon} \Delta^{i,k}\} \tag{64}$$

$$= KMN \max_{k \in [\![K_i]\!], i \in [\![K]\!]} \{(\ell_n(\Delta^{i,k}))^{\varepsilon} \Delta^{i,k}\} \tag{65}$$

$$\tag{66}$$

To finish our proof, we first give some definitions and lemmas. The learning process is nice at time horizon t if:

$$\forall i \in [\![K]\!], \mid \hat{\mu}_{i,T_i(t-1)} - \mu_i \mid < \sqrt{\frac{3 \ln t}{2 T_i(t-1)}}. \tag{67}$$

Lemma E.1 *The probability that the process is nice at round t is at least $1 - 2Kt^{-2}$.*

By Chernoff-Hoeffding bound in Fact B.1, for all $i \in [\![K]\!]$, we have:

$$P \left[\mid \hat{\mu}_{i,T_i(t-1)} - \mu_i \mid \geq \sqrt{\frac{3 \ln t}{2 T_i(t-1)}} \right] \tag{68}$$

$$= \sum_{s=1}^{t-1} \Pr \left[\left\{ \mid \hat{\mu}_{i,s} - \mu_i \mid \geq \sqrt{\frac{3 \ln t}{2s}}, T_i(t-1) = s \right\} \right] \tag{69}$$

$$= \sum_{s=1}^{t-1} \Pr \left[\left\{ \mid \hat{\mu}_{i,s} - \mu_i \mid \geq \sqrt{\frac{3 \ln t}{2s}} \right\} \right] \tag{70}$$

$$= \leq t \cdot 2e^{-3 \ln t} = \frac{2}{t^2}. \tag{71}$$

Lemma E.2 *For any time horizon $n > t > K$*

$$\mathbb{E}\left[\sum_{i\in[\![K]\!],l\in[\![K_i]\!]} \mathbb{I}\{S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, T_i(t-1) > \ell_n(\Delta^{i,l})\}\right] \tag{72}$$

$$= \sum_{i\in[\![K]\!],l\in[\![K_i]\!]} \Pr\{S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_{i,B}^l, T_s(t-1) > \ell_n(\Delta^{i,l})\} \tag{73}$$

$$\leq (1-\beta) + 2Kt^{-2}. \tag{74}$$

Define $\Lambda_{i,t} = \sqrt{\frac{3\ln t}{2T_i(t-1)}}$ (a random variable since $T_i(t-1)$ is a random variable) and $\Lambda_t = \max\{\Lambda_{i,t} \mid i \in S_t\}$. Define $\Lambda^{i,l} = \sqrt{\frac{3\ln t}{2\ell_n(\Delta^{i,l})}}$ (not a random variable).

Let $\mathcal{N}_t$ indicate the event that the process is nice at round t. Let F_t be the event that the oracle fails to produce an α-approximate answer with respect to input vector $\bar{\mu}_t$ in round t. We have $\Pr[F_t] = \mathbb{E}[\mathbb{I}\{F_t\}] \leq 1-\beta$.

Notice that $\bar{\mu}_{i,t} = \hat{\mu}_{i,t} + \sqrt{\frac{3\ln t}{2T_i(t-1)}}$. We have the following properties.

$$\mathcal{N}_t \Rightarrow \forall i \in [\![K]\!], \bar{\mu}_{i,t} - \mu_i > 0 \tag{75}$$

$$\mathcal{N}_t \Rightarrow \forall i \in S_t, \bar{\mu}_{i,t} - \mu_i < 2\Lambda_t, \tag{76}$$

$$\forall i \in [\![K]\!], \forall l \in [\![K_i]\!], \{S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\} \Rightarrow \Lambda^{i,l} > \Lambda_t. \tag{77}$$

For any particular $i \in [\![K]\!]$ and $l \in [\![K_i]\!]$, if $\{\mathcal{N}_t, \neg F_t, S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\}$ holds at round t, we have the following properties:

$$\begin{aligned}
r_{\boldsymbol{\mu}}(S_t) + f(2\Lambda^{i,l}) &> r_{\boldsymbol{\mu}}(S_t) + f(2\Lambda_t) && \text{strict monotonicity of } f(\cdot) \text{ and Eq. 77}\\
&\geq r_{\bar{\boldsymbol{\mu}}_t}(S_t) && \text{bounded smoothness property and Eq.76}\\
&\geq \alpha \cdot \text{opt}_{\bar{\boldsymbol{\mu}}_t} && \neg F_t \Rightarrow S_t \text{ is an } \alpha \text{ approximation w.r.t } \bar{\boldsymbol{\mu}}_t\\
&\geq \alpha \cdot r_{\bar{\boldsymbol{\mu}}_t}(S_*) && \text{definition of opt}_{\bar{\boldsymbol{\mu}}_t}\\
&\geq \alpha \cdot r_{\boldsymbol{\mu}}(S_*) = \alpha \cdot \text{opt}_{\boldsymbol{\mu}}. && \text{monotonicity of } r_{\boldsymbol{\mu}}(S) \text{ and Eq.75}
\end{aligned}$$

So we have

$$r_\mu(S_{i,B}^l) + f(2\Lambda^{i,l}) > \alpha \cdot \text{opt}_\mu. \tag{78}$$

Since $\ell_n(\Delta^{i,l}) = \frac{6\ln n}{(f^{-1}(\Delta^{i,l}))^2}$, we have $2\Lambda^{i,l} = f^{-1}(\Delta^{i,l}) \cdot \sqrt{\frac{\ln t}{\ln n}}$ which implies $f(2\Lambda^{i,l}) \leq \Delta^{i,l}$ by the monotonicity of $f(\cdot)$ and $t \leq n$. Therefore, Eq.(23) contradicts the definition of $\Delta^{i,l}$ in Eq.(14). In other words,

$$\forall i \in [\![K]\!], \forall l \in [\![K_i]\!], \Pr\{\{\mathcal{N}_t, \neg F_t, S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\} = 0 \tag{79}$$

$$\Rightarrow \Pr\{\mathcal{N}_t, \neg F_t, \exists i \in [\![K]\!], \exists l \in [\![K_i]\!], S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\} = 0 \tag{80}$$

$$\Rightarrow \Pr\{\exists i \in [\![K]\!], \exists l \in [\![K_i]\!], S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\} \tag{81}$$

$$\leq \Pr[F_t \vee \neg\mathcal{N}_t] \leq (1-\beta) + 2Kt^{-2} \tag{82}$$

$$\Rightarrow \sum_{i\in[\![K]\!],l\in[\![K_i]\!]} \Pr\{S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_n(\Delta^{i,l})\} \leq (1-\beta) + 2Kt^{-2}. \tag{83}$$

The second inequality in Eq.82 uses the facts that $\Pr\{F_t\} = (1-\beta)$ and $\Pr\{\neg\mathcal{N}_t\} \leq 2Kt^{-2}$ (Lemma E.1). The left side of Eq.83 equals the left side of Eq.82, because the events $\{S_t = S_{i,B}^l, N_{i,t} > N_{i,t-1}, \forall s \in S_t, T_s(t-1) > \ell_t(\Delta^{i,l})\}$ for all $i \in [\![K]\!]$ and $l \in [\![K_i]\!]$ are mutually exclusive, which in turn is because in each round when S_t is sub-optimal, only one arm $i \in S_t$ gets to increment its counter N_i and thus $N_{i,t} > N_{i,t-1}$, and within arm i, only one index l satisfies $S_t = S_{i,B}^l$.

Sufficiently sampled stage. In this stage, if we choose a sub-optimal super arms to pull and pull it more times from τ_i to τ_{i+1}, it in fact occupies all the exploring chances from τ_i to τ_{i+1}, so the regret upper bound of this stage is:

$$\mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} P\Bigg(\sum_{i\in[\![K]\!],l\in[\![K_i]\!]} \mathbb{I}\{S_t = \mathcal{S}_{i,\mathrm{B}}^l, N_{i,t} > N_{i,t-1}, T_i(t-1), \ell_n(\Delta^{i,l})\}\Bigg)\times \tag{84}$$

$$(\gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C))\Bigg]\Delta_{max} \tag{85}$$

where $\tau_{i+1} = \tau_i + \gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C)$ and $\mathbb{I}(C) = 1$ means we exploit this sub-optimal super arm more times via near-optimal exploitation. $\tau_{i+1} - \tau_i$ is the round span of 'overpull' this sub-optimal super arm.

Then we can derive the regret of sufficiently sampled cases:

$$Reg_s(n) \tag{86}$$

$$\leq \mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} ((1-\beta) + 2Kt^{-2})(\gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C))\Bigg]\Delta_{max} \tag{87}$$

$$= \mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} (1-\beta)(\gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C))\Bigg]\Delta_{max} + \tag{88}$$

$$\mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} 2Kt^{-2}(\gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C))\Bigg]\Delta_{max} \tag{89}$$

$$\leq \mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} (1-\beta)(\gamma_{S_t}(t) + (M-1)\gamma_{S_t}(t)\mathbb{I}(C))\Bigg]\Delta_{max} + \tag{90}$$

$$\mathbb{E}\Bigg[\sum_{t=\tau_1,\tau_2,\dots} 2Kt^{-2}(M\gamma_{S_t}(t))\Bigg]\Delta_{max} \tag{91}$$

$$\leq (1-\beta)n\Delta_{max} + \sum_{t\geq K} 2KMt^{-2}\gamma_{S_t}(t)\Delta_{max} \tag{92}$$

$$\leq (1-\beta)n\Delta_{max} + 2KMN\zeta(2-\varepsilon)\Delta_{\max} \tag{93}$$

Finally, we can calculate the regret upper bound:

$$Reg(n) \leq n \cdot \alpha \cdot \beta \cdot \mathrm{opt}_{\boldsymbol{\mu}} - \left(\alpha \cdot n \cdot \mathrm{opt}_{\boldsymbol{\mu}} - Reg_u(n) - Reg_t(n) - Reg_s(n) \right) \tag{94}$$

$$= (\beta - 1) \cdot n \cdot \alpha \cdot \mathrm{opt}_{\boldsymbol{\mu}} + Reg_u(n) + Reg_t(n) + Reg_s(n) \tag{95}$$

$$\leq \sum_{i \in [\![K]\!], \Delta^i_{min} \geq 0} \left(\ell_n(\Delta^{i,K_i}) \Delta^{i,K_i} + \int_{\Delta^{i,K_i}}^{\Delta^{i,1}} \ell_n(x) dx \right) + 2KMN\zeta(2-\varepsilon)\Delta_{\max} + \tag{96}$$

$$KMN \max_{k \in [\![K_i]\!], i \in [\![K]\!]} \{ (\ell_n(\Delta^{i,k}))^{\varepsilon} \Delta^{i,k} \}. \tag{97}$$

F Proof of Switching Cost

We first give the following two lemmas.

Lemma F.1 *Let $\{a_n\}_{n \geq 0}$ be a sequence defined recursively by*

$$a_n = a_{n-1} + N a_{n-1}^{\varepsilon}, \quad \textit{for } n \geq 1,$$

with $a_0 > 0$, $N > 0$, and $0 < \varepsilon < 1$. Then there exists a constant $C > 0$ such that

$$a_n \geq C n^{\frac{1}{1-\varepsilon}}, \quad \forall n \geq n_0,$$

for some integer $n_0 \geq 1$.

Let $\gamma = \frac{1}{1-\varepsilon}$ and define $b_n = Cn^{\gamma}$. We will show by induction that $a_n \geq b_n$ for sufficiently large n.

Assume $a_n \geq b_n$ for some $n \geq n_0$, then

$$a_{n+1} = a_n + N a_n^{\varepsilon} \geq b_n + N b_n^{\varepsilon} = Cn^{\gamma} + NC^{\varepsilon} n^{\varepsilon\gamma}.$$

Note that

$$b_{n+1} = C(n+1)^{\gamma} = Cn^{\gamma} \left(1 + \frac{1}{n} \right)^{\gamma}.$$

By the mean value theorem, for some $\xi \in (n, n+1)$,

$$(n+1)^{\gamma} - n^{\gamma} = \gamma \xi^{\gamma - 1} \leq \gamma (n+1)^{\gamma - 1}.$$

Thus,

$$b_{n+1} - b_n \leq C\gamma (n+1)^{\gamma - 1}.$$

So, to ensure $a_{n+1} \geq b_{n+1}$, it suffices that

$$NC^{\varepsilon} n^{\varepsilon\gamma} \geq C\gamma (n+1)^{\gamma - 1}.$$

Since $\varepsilon\gamma = \gamma - 1$, this reduces to:

$$NC^{\varepsilon - 1} \geq \gamma \left(1 + \frac{1}{n} \right)^{\gamma - 1}.$$

Observe that $\left(1 + \frac{1}{n}\right)^{\gamma - 1} \leq 2^{\gamma - 1}$ for all $n \geq 1$, so a sufficient condition is:

$$NC^{\varepsilon - 1} \geq \gamma \cdot 2^{\gamma - 1}.$$

This holds if we set

$$C = \left(\frac{\gamma \cdot 2^{\gamma - 1}}{N} \right)^{\frac{1}{\varepsilon - 1}} = \left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)} \right)^{\frac{1}{\varepsilon - 1}},$$

which is positive since $\varepsilon - 1 < 0$ and the base is positive. By choosing n_0 large enough so that $a_{n_0} \geq b_{n_0}$, the induction is complete.

Lemma F.2 *By defining t' as the start round of different phases, we have the following conclusion:*

$$C(n) \leq 2\mathbb{E}[\sum_{t'} \mathbb{I}(i_{t'} \neq i^*)]. \tag{98}$$

The above conclusion can be deduced as follows:

$$\begin{aligned}
C(n) &= \mathbb{E}[\sum_{t=1}^{n-1} \mathbb{I}(i_t \neq i_{t+1})] && (99)\\
&= \mathbb{E}[\sum_{t' \neq 1} \mathbb{I}(i_{t'-1} \neq i_{t'})] && (100)\\
&\leq \mathbb{E}[\sum_{t' \neq 1} 1 - \mathbb{I}(i_{t'-1} = i_{t'})] && (101)\\
&= \mathbb{E}[\sum_{t' \neq 1} 1 - \sum_{j=1}^{K} \mathbb{I}(i_{t'-1} = i_{t'} = j)] && (102)\\
&\leq \mathbb{E}[\sum_{t' \neq 1} 1 - \mathbb{I}(i_{t'-1} = i_{t'} = i_*)] && (103)\\
&\leq \mathbb{E}[\sum_{t' \neq 1} \mathbb{I}(i_{t'-1} \neq i_*) + \mathbb{I}(i_{t'} \neq i_*)] && (104)\\
&\leq 2\mathbb{E}[\sum_{t'} \mathbb{I}(i_{t'} \neq i_*)]. && (105)
\end{aligned}$$

During most of the time, UCB-based methods pull the best arm i_*. So we make some relaxation to the switching cost $C(n)$ to simplify our proof.

Lemma F.3 *Initial concentrated exploration pull each (base) arm at least $\ln(n)$ times.*

If $T_i(t) \leq \ln(n)$, both $\sqrt{2\ln(n)/T_i(t)}$ and $\sqrt{3\ln(n)/2T_i(t)}$ are larger than 1, which contradicts with $\bar{\mu}'_{i,t} \leq 1$.

F.1 Proof of Theorem 5.4

For MSMR-UCB, when arm i is selected for the j-th time, we define the length of the corresponding phase as $L_{i,j}$. Similar to the regret analysis, we decompose the total switching cost into three components. Specifically, let $t_{u,i}$, $t_{t,i}$, and $t_{s,i}$ denote the number of phases in which the sub-optimal arm i is selected during the under-sampled stage, the transition stage, and the sufficiently sampled stage, respectively. The total switching cost is then bounded as:

$$C(n) \leq 2 \sum_{i \neq i_*} (t_{u,i} + t_{t,i} + t_{s,i}). \tag{106}$$

In the under-sampled stage, define $W_{i,s} = \sum_{j=1}^{s} L_{i,j}$ as the cumulative length of the first s phases for arm i. According to the definition of near-optimal exploitation, we have:

$$W_{i,s} \geq W_{i,s-1} + N(W_{i,s-1})^{\varepsilon}. \tag{107}$$

Let l_i denote the total number of pulls of arm i during the under-sampled stage. By construction, $t_{u,i}$ satisfies:

$$W_{i,t_{u,i}+1} > l_i > W_{i,t_{u,i}}. \tag{108}$$

Applying Lemma F.3, if

$$\left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}} < \ln(n), \tag{109}$$

we obtain:

$$L_{i,1} \geq \left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}}. \tag{110}$$

Using Lemma F.1, we derive:

$$W_{i,t_{u,i}} \geq l_i \tag{111}$$

$$\Leftarrow \quad \left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}} t_{u,i}^{\frac{1}{1-\varepsilon}} \geq \frac{8\ln(n)}{\Delta_i^2} \tag{112}$$

$$\Leftarrow \quad t_{u,i} \geq \left(\frac{8\ln(n)}{\Delta_i^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}. \tag{113}$$

This contradicts $l_i < W_{i,t_{u,i}}$, so we conclude that:

$$t_{u,i} \leq \left(\frac{8\ln(n)}{\Delta_i^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}. \tag{114}$$

In the transition stage, by definition, $t_{t,i} \leq 1$ for any arm i.

In the sufficiently sampled stage, the number of phases $t_{s,i}$ for each arm i is upper bounded by $2MN \cdot \zeta(2-\varepsilon)$, which is a constant.

Combining all three stages, the total switching cost for MSMR-UCB satisfies:

$$C(n) \leq 2\sum_{i=1}^{K}(t_{u,i} + t_{t,i} + t_{s,i}) \tag{115}$$

$$\leq 2K + 4KMN \cdot \zeta(2-\varepsilon) + 2\sum_{i\neq i_*}\left(\frac{8\ln(n)}{\Delta_i^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}. \tag{116}$$

F.2 Proof of Theorem 5.5

For MSMR-CUCB, when a super arm S is selected and base arm $i = i_{S,t} \triangleq \arg\min_{j\in S} T_j(t)$ is chosen for the j-th time, we define the length of the corresponding phase as $L_{i,j}$. As in the regret analysis, we decompose the switching cost into three parts. Let $t_{u,i}$ and $t_{t,i}$ denote the number of phases in which a super arm containing base arm i is selected during the under-sampled and transition stages, respectively. Let t_s denote the number of phases in which any sub-optimal super arm is selected in the sufficiently sampled stage. The total switching cost is then bounded by:

$$C(n) \leq 2\left(t_s + \sum_{i=1}^{K}(t_{u,i} + t_{t,i})\right). \tag{117}$$

In the under-sampled stage, define $W_{i,s} = \sum_{j=1}^{s} L_{i,j}$ as the cumulative length of the first s phases for base arm i. By the definition of near-optimal exploitation, we have:

$$W_{i,s} \geq W_{i,s-1} + N(W_{i,s-1})^{\varepsilon}. \tag{118}$$

Let l_i denote the total number of pulls of base arm i in the under-sampled stage. Then the total number of under-sampled phases $t_{u,i}$ satisfies:

$$W_{i,t_{u,i}+1} > l_i > W_{i,t_{u,i}}. \tag{119}$$

Applying Lemma F.3, if

$$\left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}} < \ln(n), \tag{120}$$

then we obtain:

$$L_{i,1} \geq \left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}}. \tag{121}$$

Using Lemma F.1, we derive:

$$W_{i,t_{u,i}} \geq \ell_n(\Delta^{i,K_i}) \tag{122}$$

$$\Leftarrow \quad \left(\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\right)^{\frac{1}{\varepsilon-1}} t_{u,i}^{\frac{1}{1-\varepsilon}} \geq \frac{6\ln(n)}{(f^{-1}(\Delta^{i,K_i}))^2} \tag{123}$$

$$\Leftarrow \quad t_{u,i} \geq \left(\frac{6\ln(n)}{(f^{-1}(\Delta^{i,K_i}))^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}. \tag{124}$$

This contradicts $l_i < W_{i,t_{u,i}}$, so we conclude that:

$$t_{u,i} \leq \left(\frac{6\ln(n)}{(f^{-1}(\Delta^{i,K_i}))^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}. \tag{125}$$

In the transition stage, by definition, $t_{t,i} \leq K_i$ for each base arm i.

In the sufficiently sampled stage, the number of phases t_s involving any sub-optimal super arm is at most $2KMN \cdot \zeta(2-\varepsilon)$, which is a small constant.

Therefore, the total switching cost of MSMR-CUCB is bounded by:

$$C(n) = 2\left(t_s + \sum_{i=1}^{K}(t_{u,i} + t_{t,i})\right) \tag{126}$$

$$\leq 4KMN \cdot \zeta(2-\varepsilon) + 2\sum_{i=1}^{K}\left(\frac{6\ln(n)}{(f^{-1}(\Delta^{i,K_i}))^2}\right)^{1-\varepsilon} \cdot \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)} + 2\sum_{i=1}^{K} K_i. \tag{127}$$

G Proof of Marginal Loss

In MAB setting, According to (3), the regret upper bound of UCB is

$$Reg^{opt}(n) = \sum_{i \neq i_*}\left(\frac{8\ln n}{\Delta_i} + (1+\frac{\pi^2}{3})\Delta_i\right). \tag{128}$$

In CMAB setting, according to (7), the regret upper bound of CUCB is

$$\sum_{i \in [\![K]\!], \Delta^i_{min} \geq 0}\left(\ell_n(\Delta^{i,K_i})\Delta^{i,K_i} + \int_{\Delta^{i,K_i}}^{\Delta^{i,1}} \ell_n(x)\,dx\right) + (1+\frac{\pi^2}{3}) \cdot m \cdot \Delta_{\max}. \tag{129}$$

In the greedy algorithm, it directly pulls each base(super) arm fixed times and then chooses the one with the largest empirical reward. In MAB setting, we have K base arms, so $C^{opt}(n) = K$. In CMAB setting, the super arms may just contain 1 base arm, so the greedy algorithm at least to pull K super arms to get the evaluation of all base arms, resulting in $C^{opt}(n) = K$. According to theorem 5.2 5.4 5.3 5.5 and the definition of marginal loss 4, we can get that:

$$R^{MSMR}(\lambda, n) = \mathcal{O}((\log n)^{\varepsilon} + \lambda(\log n)^{1-\varepsilon}) \tag{130}$$

For any other bandit algorithm X mentioned in table 1, it is easy to find that $Reg^X(n) - Reg^{opt}(n) = \mathcal{O}(Reg^X(n))$, so the marginal loss of algorithm X is $\mathcal{O}(Reg^X(n) + \lambda C^X(n))$. In most cases, λ is constant, and setting $\varepsilon = 0.5$ yields the theoretical minimum marginal loss $\mathcal{O}(\lambda\sqrt{\log n})$

H Proof about Predictive Selection

In MAB settings, for any arm i, once the agent chooses an arm $i_{t_1} \neq \hat{i}_{*,t_1}$ to explore at round t_1, the algorithm will repeatedly select i_{t_1} in the subsequent phase regardless of the upper confidence estimation if the following inequality holds:

H.1 Proof of Lemma 5.6

In round t_1, we choose arm i_{t_1} for exploration. After this phase, we define round $t_2 = t_1 + \gamma_{i_{t_1}}(t_1)$ and a subsequent round $t_3 = t_2 + \sum_{j \neq i_{t_1}, \hat{i}_{*,t_2}} \gamma_j(t_2)$.

If neither $\hat{i}_{*,t_2}$ nor i_{t_1} is selected in any round $t \in [t_2, t_3]$, and Eq. 5 holds, we obtain the following sequence of inequalities:

$$\begin{aligned}
\bar{\mu}_{\hat{i}_{*,t_2},t} &= \sqrt{\frac{2\ln(t)}{T_{\hat{i}_{*,t_2}}(t)}} + \hat{\mu}_{\hat{i}_{*,t_2},t} && (131)\\
&= \sqrt{\frac{2\ln(t)}{T_{\hat{i}_{*,t_2}}(t_2)}} + \hat{\mu}_{\hat{i}_{*,t_2},t_2} && (132)\\
&\leq \sqrt{\frac{2\ln(t_3)}{T_{\hat{i}_{*,t_2}}(t_2)}} + \hat{\mu}_{\hat{i}_{*,t_2},t_2} && (133)\\
&\leq \sqrt{\frac{2\ln(t_2)}{T_{i_{t_1}}(t_2)}} + \hat{\mu}_{i_{t_1},t_2} && (134)\\
&\leq \bar{\mu}_{i_{t_1},t}, && (135)
\end{aligned}$$

which implies that the upper confidence bound of arm i_{t_1} is always greater than or equal to that of $\hat{i}_{*,t_2}$ throughout the interval $[t_2, t_3]$. Consequently, if $\hat{i}_{*,t_2}$ is selected at any round $t \in [t_2, t_3]$, then arm i_{t_1} must have already been selected at least once prior to t.

We now consider three exhaustive cases:

Case 1: Arm $\hat{i}_{*,t_2}$ is selected in some round $t \in [t_2, t_3]$. From Eq. 135, we conclude that arm i_{t_1} must have been selected at least twice before arm $\hat{i}_{*,t_2}$ is selected. Therefore, Lemma 5.6 holds in this case.

Case 2.1: Arm $\hat{i}_{*,t_2}$ is not selected in $[t_2, t_3]$, but arm i_{t_1} is. In this case, arm i_{t_1} is selected at least twice before round t_3, while arm $\hat{i}_{*,t_2}$ is never selected. Thus, Lemma 5.6 also holds here.

Case 2.2: Neither arm $\hat{i}_{*,t_2}$ nor arm i_{t_1} is selected in $[t_2, t_3]$. We further divide this into two subcases:

Subcase 1: If there exists an arm $i' \notin \{\hat{i}_{*,t_2}, i_{t_1}\}$ that is selected twice in the interval $[t_2, t_3]$, then this arm is selected at least twice before round t_3, and Lemma 5.6 is satisfied.

Subcase 2: If no such arm $i' \notin \{\hat{i}_{*,t_2}, i_{t_1}\}$ is selected more than once in $[t_2, t_3]$, then every arm in this set is selected at most once. However, since

$$t_3 = t_2 + \sum_{j \notin \{i_{t_1}, \hat{i}_{*,t_2}\}} \gamma_j(t_2),$$

it follows that each arm $j \notin \{i_{t_1}, \hat{i}_{*,t_2}\}$ is selected exactly once between t_2 and t_3. By Eq. 135, arm i_{t_1} still maintains a higher UCB than $\hat{i}_{*,t_2}$ at round t_3, and hence the algorithm must select an arm $i' \neq \hat{i}_{*,t_2}$. Since every such arm has already been selected once from t_1 to t_3, this implies that one of them is selected twice. Thus, Lemma 5.6 is proved in this case as well.

H.2 Proof of Lemma 5.7

In round t_1, we choose a super arm S_{t_1} for exploration. After this phase, we set $t_2 = t_1 + \gamma_{S_{t_1}}(t_1)$ and define a future round $t_3 = t_2 + \sum_{j \in [\![K]\!] \setminus (S_{t_1} \cup \hat{S}_{*,t_2})} \gamma_j(t_2)$.

We decompose the base arm set $[\![K]\!]$ into four disjoint subsets:

$$\begin{aligned} U_1 &= S_{t_1} \cap \hat{S}_{*,t_2}, \\ U_2 &= S_{t_1} \setminus U_1, \\ U_3 &= \hat{S}_{*,t_2} \setminus U_1, \\ U_4 &= [\![K]\!] \setminus (S_{t_1} \cup \hat{S}_{*,t_2}). \end{aligned}$$

By the monotonicity assumption, if Eq. (6) holds, then there exists $k_1 \in U_2$ and $k_2 \in U_3$ such that

$$\hat{\mu}_{k_1,t_2} + \sqrt{\frac{3\ln(t_2)}{2T_{k_1}(t_2)}} \geq \hat{\mu}_{k_2,t_2} + \sqrt{\frac{3\ln(t_3)}{2T_{k_1}(t_2)}}.$$

Otherwise, for any $k_1 \in U_2$, $k_2 \in U_3$, we have

$$\hat{\mu}_{k_1,t_2} + \sqrt{\frac{3\ln(t_2)}{2T_{k_1}(t_2)}} < \hat{\mu}_{k_2,t_2} + \sqrt{\frac{3\ln(t_3)}{2T_{k_1}(t_2)}},$$

and for any $k_3 \in U_1$,

$$\hat{\mu}_{k_3,t_2} + \sqrt{\frac{3\ln(t_2)}{2T_{k_3}(t_2)}} < \hat{\mu}_{k_3,t_2} + \sqrt{\frac{3\ln(t_3)}{2T_{k_3}(t_2)}}.$$

Now consider the ordered arrangements of base arms in S_{t_1} and $\hat{S}_{*,t_1}$:

$$\begin{aligned} A &= \{U_{1,1}, \ldots, U_{1,|U_1|}, U_{2,1}, \ldots, U_{2,|U_2|}\}, \\ B &= \{U_{1,1}, \ldots, U_{1,|U_1|}, U_{3,1}, \ldots, U_{3,|U_3|}\}, \end{aligned}$$

where $|\cdot|$ denotes the cardinality of the set and $U_{x,j}$ denotes the j-th smallest element of U_x. Then for any $j \in \{1, 2, \ldots, m\}$, we have $\bar{\mu}_{A_j,t_2} < \bar{\mu}_{B_j,t_3}$, which implies by the monotonicity assumption that

$$r(\hat{\boldsymbol{\mu}}_{t_2}, \boldsymbol{c}_{t_2}, S_{t_1}) < r(\hat{\boldsymbol{\mu}}_{t_2}, \boldsymbol{c}_{t_3}, \hat{S}_{*,t_2}),$$

contradicting Eq. (6).

If none of the base arms in $U_2 \cup \{k_2\}$ are selected during $t \in [t_2, t_3]$ and Eq. (6) holds, we derive the following:

$$\begin{aligned}
\bar{\mu}_{k_2,t} &= \hat{\mu}_{k_2,t} + \sqrt{\frac{3\ln(t)}{2T_{k_2}(t)}} \\
&= \hat{\mu}_{k_2,t_2} + \sqrt{\frac{3\ln(t)}{2T_{k_2}(t_2)}} \\
&\leq \hat{\mu}_{k_2,t_2} + \sqrt{\frac{3\ln(t_3)}{2T_{k_2}(t_2)}} \\
&\leq \hat{\mu}_{k_1,t_2} + \sqrt{\frac{3\ln(t_2)}{2T_{k_1}(t_2)}} \\
&\leq \bar{\mu}_{k_1,t}. \qquad (136)
\end{aligned}$$

This indicates that for any super arm S containing base arm k_2, there exists an alternative super arm $S' = S \setminus \{k_2\} \cup \{k_1\}$ such that

$$r_{\bar{\boldsymbol{\mu}}_t}(S) \leq r_{\bar{\boldsymbol{\mu}}_t}(S').$$

Since $k_2 \in \hat{S}_{*,t_2}$, this implies that there exists another super arm better than $\hat{S}_{*,t_2}$ in round $t \in [t_2, t_3]$ if no arm from $U_2 \cup \{k_2\}$ is selected.

Case 1: A super arm S containing base arm k_2 is selected during $t \in [t_2, t_3]$.
By Eq. (136), it must be that a super arm containing k_1 has been selected before S is selected. Since $k_1 \notin \hat{S}_{*,t_2}$, it is pulled at least twice before t, completing the proof of Lemma 5.7.

Case 2.1: No super arm containing k_2 is selected during $[t_2, t_3]$, but a super arm containing $i \in U_2$ is selected. Then i is pulled twice before t_3, and $\hat{S}_{*,t_2}$ is never selected before t_3, completing the proof of Lemma 5.7.

Case 2.2: No super arm containing any element of $\{k_2\} \cup U_2$ is selected during $[t_2, t_3]$.
Let $S_1, \ldots, S_{K-|S_{t_1} \cup \hat{S}_{*,t_2}|}$ be the next super arms selected during $[t_2, t_3]$, and define the recursive update:

$$x_j = \gamma_{S_j}\left(t_2 + \sum_{k=1}^{j-1} x_k\right), \quad t_3' = t_2 + \sum_{j=1}^{K-|S_{t_1} \cup \hat{S}_{*,t_2}|} x_j.$$

Each $S_j \subseteq U_3 \cup U_4 \setminus \{k_2\}$, so each contains at least one element of U_4.

Subcase 1: If any base arm $i \in U_4$ is selected twice in $[t_2, t_3]$, the proof is complete.

Subcase 2: If no base arm $i \in U_4$ is selected more than once, then for each j,

$$x_j = \gamma_{S_j}(t_2 + \sum_{k=1}^{j-1} x_k) \leq \gamma_{\{k_2\} \cup S_j \setminus \hat{S}_{*,t_2}}(t_2).$$

Therefore, $t_3 \geq t_3'$. Since every $i \in U_4$ is selected once during $[t_2, t_3]$ and

$$\{k_1\} \cup \hat{S}_{*,t_2} \setminus \{k_2\}$$

has a higher estimated reward than $\hat{S}_{*,t_2}$ in round t_3, the algorithm selects another super arm. As all base arms not in $U_3 \setminus \{k_2\}$ are selected once starting from t_1, there exists at least one base arm $i' \notin \hat{S}_{*,t_2}$ that is pulled twice before t_3, completing the proof of Lemma 5.7.

I MORE EXPERIMENTS

Here we present the complete set of experiments.

I.1 EXPERIMENT SETUP

Each experiment was conducted over 20 independent trials to ensure reliability, with $n = 100000$, $N = 1$, $M = 5$ for all bandit settings and $\alpha = 0.95$, $\beta = 1$ for CMAB. The tests were performed on a macOS system equipped with an Apple M3 Pro processor and 18 GB of RAM.

Baselines. The algorithms used for comparison include UCB (3), phase-UCB(16), Batched Tsallis-INF(1), Batched Arm Elimination (13)in MAB settings and CUCB (6), phased-CUCB(16), B-FTRL(11) in CMAB setting.

Regret, Switching Cost and Marginal Loss. We set $\varepsilon = 0.5$ in MSMR and MSMR-P. We set $\gamma(t) = 0.01t$ and $\gamma(t) = 0.001t$ for Phase-UCB and Phase-CUCB. For B-FTRL, we set $a = 0.5$ and $b = 1$. For marginal loss, we set $R^A(\lambda, n) = \max(Reg^A(n) - Reg^{opt}(n), 0) + \lambda \max(C^A(n) - C^{opt}(n), 0)$and set $\lambda = 0.1$.

Data Generation. For simulation, in MAB settings, we set $K = 10$ and give a similar reward distribution which $\mu_i = 0.3 + 0.002 \times i$. In each round t, an arm i_t is selected and the agent observes the outcome $X_{i_t,t}$. We conduct experiments on cascading bandits of the CMAB as an instance, where the objective is to select $m = 5$ items from a set of $K = 20$ to maximize the reward. We still give a very similar reward distribution which is $\mu_i = 0.3 + 0.002 \times i$. In each round t, a list $S_t = (a_{t,1}, \ldots, a_{t,m}) \subseteq [\![K]\!]$ is randomly selected. The outcome $X_{t,i}$ for each $i \in S_t$ is generated from a Bernoulli distribution with mean μ_i. Given the ranked list S_t, if stopping at the j_t-th item, the observed outcomes are: $(X_{t,a_1}, \ldots, X_{t,a_k}) = (0, \ldots, 0, 1, x, \ldots, x)$, where the first $j_t - 1$ items are 0, the j_t-th item is 1, and the rest are unobserved (x). If the list is exhausted, the observed outcomes are: $(X_{t,a_1}, \ldots, X_{t,a_k}) = (0, 0, \ldots, 0)$. The reward is 1 for stopping and 0 for exhausting the list. The reward function can be written as $r(S_t; \boldsymbol{\mu}) = 1 - \prod_{i \in S_t}(1 - \mu_i)$.

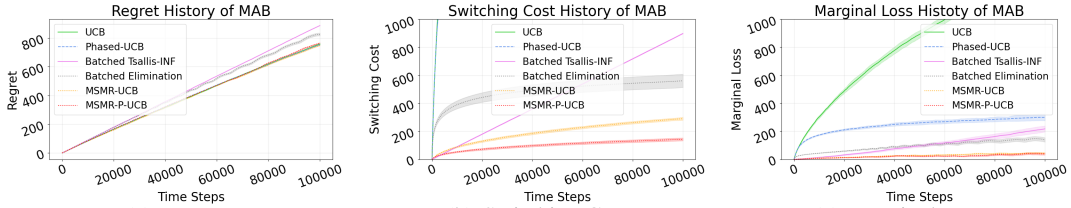
For the real-world setting, we conduct experiments using the real-world Last.fm dataset (5), sourced from the Last.fm online music platform[1]. This dataset comprises 186,479 tag assignments, connecting 1,892 users to 17,632 artists. To model user preferences, following the existing works (10; 17), we derive feedback from ratings: if an item's rating exceeds 3, we assign a feedback value of 1; otherwise, it is 0. This binary feedback approach simplifies the representation of user preferences by distinguishing positive interactions from negative or neutral ones. Based on this feedback, we calculate the expected reward μ for each item in the Last.fm dataset. In MAB settings, we set $K = 10$ and randomly choose 10 similar distributions which $\mu_i \in [0.3, 0.32]$ is randomly sampled. In CMAB settings, We conduct experiments on cascading bandits as an instance, where the objective is to select $m = 5$ items from a set of $K = 20$ to maximize the reward. We still give 20 similar reward distributions which $\mu_i \in [0.3, 0.34]$ is randomly sampled.

I.2 EXPERIMENTAL RESULTS

Synthetic Data. In Figures 2(a) and 3(a), we observe that the regret of MSMR and MSMR-P closely matches that of the standard baseline methods. In contrast, Batched Tsallis-INF and B-FTRL exhibit noticeably higher regret. Figures 2(b) and 3(b) further show that standard and phased methods suffer from a substantial number of switches, often exceeding several thousand. Batched Tsallis-INF and B-FTRL also incur a significant number of switches. In comparison, MSMR results in only 289 switches in the MAB setting and 432 in the CMAB setting, amounting to merely 2.1% of the switches incurred by UCB and 2.4% of those by CUCB. Moreover, MSMR-P achieves even greater savings, reducing switching to just 1.0% of UCB and 1.3% of CUCB—representing a nearly 50% reduction in switching cost compared to MSMR. These results highlight

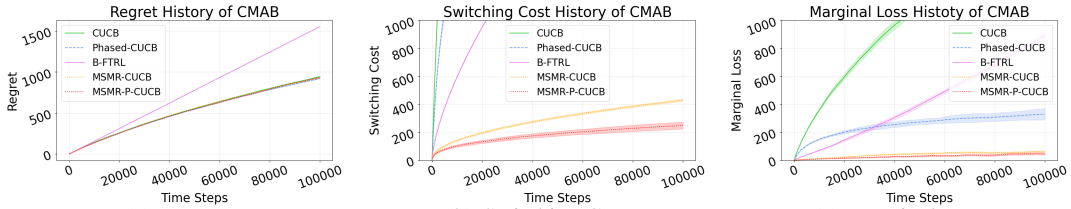
[1] `https://www.last.fm`

the effectiveness of the *predictive selection* technique. In terms of marginal loss, figures 2(c) and 3(c) further shows that the MSMR framework achieves remarkably low loss compared to the best existing algorithm, significantly outperforming all other methods.



(a) Regret (b) Switching Cost (c) Marginal Loss

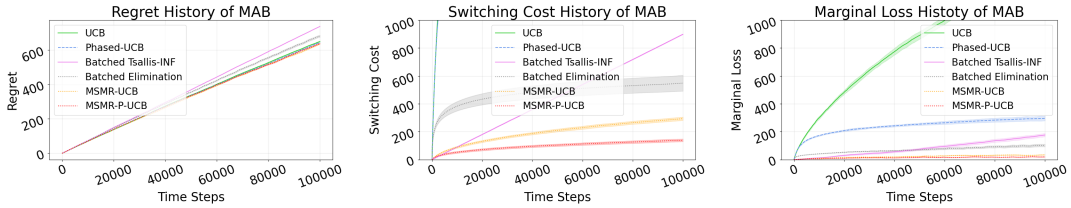
Figure 2: Synthetic Experiments on MAB



(a) Regret (b) Switching Cost (c) Marginal Loss

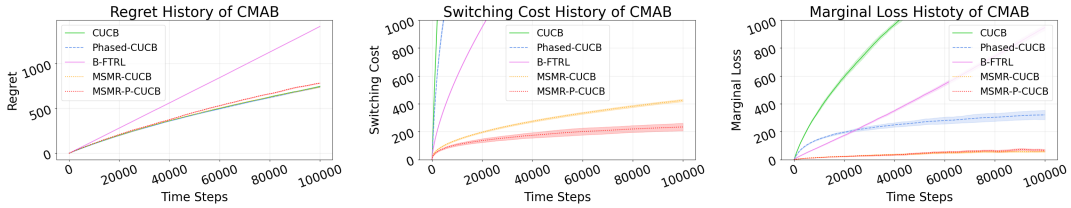
Figure 3: Synthetic Experiments on CMAB

Real-world Data. On real-world datasets, we find trends consistent with the simulation results. As shown in Figures 4(a) and 5(a), MSMR and MSMR-P achieve regret comparable to standard baselines, while Batched Tsallis-INF and B-FTRL incur noticeably higher regret. Figures 4(b) and 5(b) demonstrate that MSMR substantially reduces switching, with MSMR-P achieving the lowest cost overall. Finally, Figures 4(c) and 5(c) confirm that the MSMR framework maintains consistently low marginal loss, outperforming all existing methods.



(a) Regret (b) Switching Cost (c) Marginal Loss

Figure 4: Real-world Experiments on MAB



(a) Regret (b) Switching Cost (c) Marginal Loss

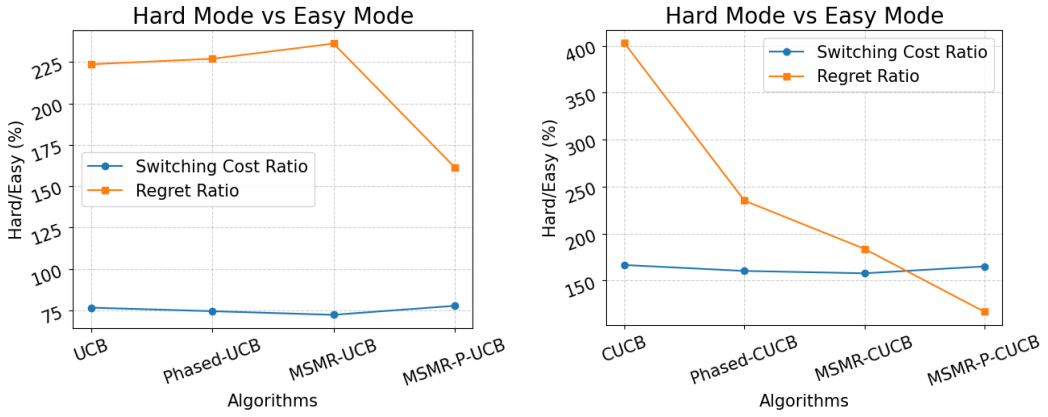
Figure 5: Real-world Experiments on CMAB

I.3 Ablation Studies

In this section, we investigate the impact of various settings on the MSMR algorithm, including different arm gaps and different values of ε.

Different Arm Settings. For arm sets with larger reward gaps, identifying the optimal arm becomes relatively easier, leading to greater gains. Our primary interest, however, lies in scenarios with small gaps, where we aim to examine whether the algorithm can still secure competitive rewards while incurring only minimal switching cost. Accordingly, we design experiments with varying arm gaps for comparison. For MAB setting, we set $K = 10, \mu_i = 0.3 + 0.002(i-1)$ in hard mode and $K = 5, \mu_i = 0.3 + 0.01(i-1)$ in easy mode. For CMAB setting, we set $K = 20, m = 5, \mu_i = 0.3 + 0.002(i-1)$ in hard mode and $K = 10, m = 3, \mu_i = 0.3 + 0.01(i-1)$ in easy mode. The vertical axis *ratio* represents the percentage of regret and switching cost in the hard mode compared to the easy mode.

We observe that across different arm settings, the percentage change in regret from Easy to Hard mode remains consistent across algorithms. However, in terms of switching cost, Figure 6(a) shows that in the MAB setting, MSMR-UCB maintains stable performance, while MSMR-P-UCB experiences a much smaller increase than other algorithms, highlighting its advantage in handling arms with similar rewards. The performance of MSMR is further explained by the fact that, in this experiment, it switches only about 100 times, nearly reaching the theoretical lower bound. In the CMAB setting, as shown in Figure 6(b), regret across algorithms also remains relatively stable. Yet under the Hard mode, the increase in switching frequency for MSMR-based algorithms is minimal, with MSMR-P remaining almost unchanged.



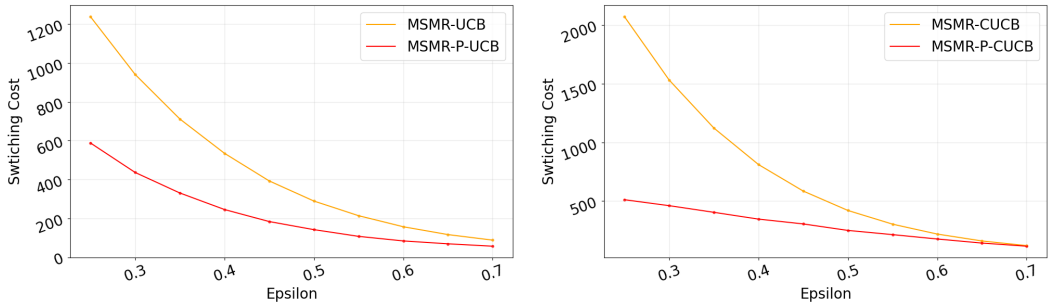
(a) MAB Case (b) CMAB Case

Figure 6: Different Arm Settings

Different ε. To further study the impact of the function $\gamma(t)$ on switching cost, we conduct a series of experiments by varying the parameter ε. We set $\varepsilon = 0.2 + 0.05 \times (j+1)$ for $j = 1, 2, \ldots, 10$. As shown in Figures 7(a) and 7(b), both MSMR and MSMR-P experience reduced switching cost as ε increases. When $\varepsilon \geq 0.3$, the switching cost remains consistently low for both algorithms. Notably, across all tested values of ε, MSMR-P consistently outperforms MSMR in terms of switching cost due to the design of the predictive selection technique, especially when ε is small .

J Phased Methods vs MSMR

In fact, the switching cost upper bound of Phased-UCB and Phased-CUCB is $\mathcal{O}(\log \log n)$. We take the MAB setting as an example to illustrate why the $\mathcal{O}(\sqrt{\log n})$ bound of our MSMR framework is preferable to



(a) MAB Case (b) CMAB Case

Figure 7: Different ε

$\mathcal{O}(\log\log n)$. The regret upper bound of Phased-UCB is $\mathcal{O}((1+\kappa)\log n)$, and to control regret one typically selects a small κ. In our experiments, we set $\kappa = 0.01$.

Since Phased-UCB is a specific instance of the MSMR framework, we can similarly derive the upper bound on its switching cost. which is

$$C^{Phased}(n) = \frac{2}{\kappa}\sum_{i\neq i_*}(\ln(\frac{1}{\kappa}) + \max(2\ln(\kappa) + \ln\left(\frac{8\ln(n)}{\Delta_i^2}\right), 0) + 8K \tag{137}$$

According to 5.4,the switching cost upper bound of MSMR-UCB is:

$$C^{MSMR-UCB}(n) = 4KMN\cdot\zeta(2-\varepsilon) + 2\sum_{i\neq i_*}\left(\frac{8\ln(n)}{\Delta_i^2}\right)^{1-\varepsilon}\frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)} + 2K. \tag{138}$$

Let $C^{Phased}(n) = C^{MSMR-UCB}(n)$, we can evaluate n by the following expression:

$$n = \exp(\frac{\Delta_{\min}^2}{8}\exp(-\frac{B'}{C'} - \frac{1}{1-\varepsilon}W_{-1}(-\frac{D'(1-\varepsilon)}{C'}\exp(\frac{B'(1-\varepsilon)}{C'})))), \tag{139}$$

where $W_{-1}(\cdot)$ is the lower branch of Lambert W Function,

$$B' = (K-1)(\frac{\ln(\kappa)}{\kappa} - 2MN\zeta(2-\varepsilon) + 3) - \frac{2}{\kappa}\sum_{i=1}^{K-1}\ln(\frac{\Delta_i}{\Delta_{\min}}), \tag{140}$$

$$C' = \frac{K-1}{\kappa}, \quad D' = \frac{2^{\frac{\varepsilon}{1-\varepsilon}}}{N(1-\varepsilon)}\sum_{i=1}^{K-1}(\frac{\Delta_{\min}}{\Delta_i})^{2-2\varepsilon}, \quad \Delta_{\min} = \min_{i\neq i_*}\Delta_i \tag{141}$$

It is worth noting that n is a very large and complex expression. For example, in the main text experiment with $K = 10$, $\kappa = 0.01$, $\Delta_{\min} = 0.002$, $\varepsilon = 0.5$, $\mu_i = 0.3 + 0.002i$, $M = 5$, and $N = 1$, even such a small $\Delta_{\min}$ produces an extremely large n, with $n \geq 10^{500}$. Accordingly, MSMR maintains a theoretical advantage over Phased-UCB as long as $n \leq 10^{500}$ in our experiment, which already covers any conceivable practical application. Therefore, in realistic scenarios, MSMR can be regarded as superior to Phased-UCB, not only in terms of regret but also in terms of switching cost.

It is worth noting that n is a very large and complex expression. For example, in the main text experiment with $K = 10$, $\kappa = 0.01$, $\Delta_{\min} = 0.002$, $\varepsilon = 0.5$, $\mu_i = 0.3 + 0.002i$, $M = 5$, and $N = 1$, even such a small

$\Delta_{\min}$ yields an extremely large n, with $n \geq 10^{500}$. Hence, MSMR maintains a theoretical advantage over Phased-UCB whenever $n \leq 10^{500}$ in our experiment, which covers all practical applications. Therefore, in realistic settings, MSMR can be regarded as superior to Phased-UCB, both in terms of regret and switching cost.