Linear Streaming Bandit: Regret Minimization and Fixed-Budget Epsilon-Best Arm Identification

Yuming Shao[1,2], **Zhixuan Fang**[1,2*]

[1]IIIS, Tsinghua University, Beijing, China
[2]Shanghai Qi Zhi Institute, Shanghai, China
shaoym21@mails.tsinghua.edu.cn, zfang@mail.tsinghua.edu.cn

Abstract

Recently, there has been a focus on the streaming setting in a line of works on the Multi-Armed Bandit (MAB). In this scenario, a large number of arms arrive in a streaming manner, and the algorithm scans through the stream and stores some arms in its limited processing memory. We advance this line of research by introducing the Linear Streaming Bandit setup, where the arriving arms have profile vectors observable to the algorithm. The profile of an arm has a linear correlation with the expected reward. This setup is motivated by real-world applications, such as when a company or a crowdsourcing platform hires a worker from many sequentially arriving applicants with their resumes. We address two problems in this setup: Regret Minimization and Fixed-Budget ϵ-Best Arm Identification. For the former, we propose an algorithm whose regret is independent of the number of arms, thus it is able to handle arbitrarily long arm streams. For the latter, we present a multi-pass algorithm whose error probability is sub-linear w.r.t. the number of arms, and an algorithm identifying the exact best arm in only a single pass. We validate the effectiveness of all proposed algorithms through experiments on both synthetic and real-world datasets.

Introduction

The Multi-Armed Bandit (MAB) (Lattimore and Szepesvári 2020) is a simple but powerful model in online decision-making scenarios. Researchers have found a wide range of industrial applications for MAB, from online advertising (Schwartz, Bradlow, and Fader 2017; Avadhanula et al. 2021; Yang and Lu 2016) to clinical trials (Villar, Bowden, and Wason 2015; Aziz, Kaufmann, and Riviere 2021). In a MAB instance, there is a set of arms $\mathcal{K}$. The decision maker (also called the algorithm) continuously selects arms from the set, yielding a sequence of reward values that help the decision maker understand each arm's profitability. There are two problems that attract research interest in MAB: Regret Minimization and Best Arm Identification. In the former, the decision maker aims to generate as much cumulative reward as possible within a time constraint, while in the latter, the objective is to find the arm with the highest expected reward.

Recently, a rapidly developing line of work on MAB studies the streaming arm model (Maiti, Patil, and Khan 2020; Jin et al. 2021; Assadi and Wang 2020, 2022; Agarwal, Khanna, and Patil 2022; Li et al. 2023; Assadi and Wang 2023; Wang 2023), motivated by the explosive growth of data and the convenience of processing arms in a streaming manner. In this streaming scenario, a large number of arms arrive one by one, while the algorithm is limited in its storage capacity. The algorithm can only pull the arms temporarily stored in its memory. To accommodate future arms in the stream, it must discard some existing arms from memory. In the streaming bandit setting, both Regret Minimization and Best Arm Identification problems have been considered in the literature. However, existing streaming bandit algorithms unexceptionally face a serious problem: their theoretical upper bounds for either regret or sample complexity become loose very quickly as the number of arms K increases. For example, Agarwal, Khanna, and Patil (2022) introduce an algorithm with $\tilde{O}\big(T^{\alpha}\sqrt{K}\big)$ regret, where T is the time horizon and $\alpha > \frac{1}{2}$. If the magnitude of K is comparable to T, i.e., $K = \Omega(T)$, the algorithm loses theoretical guarantees since the upper bound scales at least linearly with T. Besides, Jin et al. (2021) propose Fixed-Confidence Best Arm Identification (to satisfy a given error probability requirement using as few samples as possible) algorithms whose sample complexity bounds grow linearly with K. In summary, existing streaming bandit algorithms are not good at processing a large amount of arms.

One crucial reason for this scalability issue is that every new arm in the stream must be carefully, or even equally, evaluated, as the previous arms provide no information about it. This phenomenon could lead to a waste of time and resources on sub-optimal arms, as each encounter must be handled anew, resulting in the unsatisfactory performance of existing algorithms. In fact, it is both reasonable and practical to assume that some useful information is revealed as soon as a new arm is encountered, which can be utilized to efficiently judge whether the arm is sub-optimal. For instance, consider a real-world recruitment scenario (Jin et al. 2021), where a company plans to hire a highly qualified employee from among many sequentially arriving applicants by interviewing them. Typically, applicants are required to submit their curricula vitae, which can provide insight into their true talents. It is also common practice for a company to screen out unsuitable applicants based on their CVs.

Motivated by the above observation, we propose the **Lin-**

*Corresponding author: Zhixuan Fang.

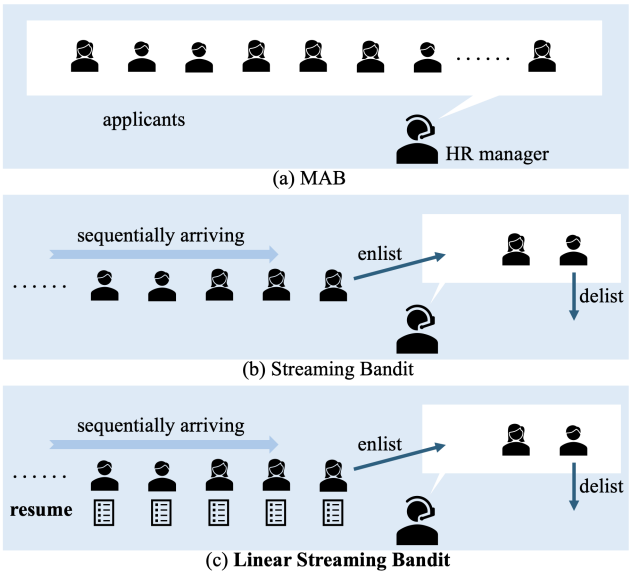

Figure 1: Three bandit models for recruitment. From top to bottom, these are MAB, Streaming Bandit, and Linear Streaming Bandit (**Our Model**), respectively. An HR manager (the algorithm) is interacting with many applicants (the arms). The white area represents the pool of applicants that the HR manager needs to maintain in each model.

ear Streaming Bandit model. We illustrate an example in Figure 1 to contrast our model with existing models. We assume that every arm in the stream has a d-dimensional profile vector that can be observed by the algorithm before it takes actions. The reward of pulling an arm is jointly determined by the arm's profile vector and an unknown d-dimensional parameter vector θ^*. Specifically, we assume a simple and common linear reward model. For example, consider the parameter vector as the return of each asset in a portfolio, and an arm as an investment manager with a certain investment policy (i.e., the arm profile vector) on each asset. The algorithm's profit is the inner product between the investment policy and the asset returns. In this paper, we consider two mainstream objectives in the proposed setting, namely Regret Minimization and Fixed-Budget Best Arm Identification.

Our main contributions are summarized as below:

- We propose the Linear Streaming Bandit setup, which is the first to establish a connection between streaming MAB and the well-investigated literature of linear bandit.
- We design a multi-pass streaming algorithm, named CR-MPS, for the regret minimization task. We show that it has a sub-linear regret upper bound, which is independent of the total number of arms.
- We design a multi-pass streaming algorithm, named G-MP-SE, which is able to identify an ϵ-best arm with high probability given constraints on sample budget and the times of scans. We show that its error probability has only a sub-linear dependency on the total number of arms. We also introduce a streaming algorithm, named SPC, for strict best arm identification, requiring only a single pass.
- We verify the effectiveness of each proposed algorithm by testing them on both synthetic and real-world datasets.

Related Work

Streaming Multi-Armed Bandit. A number of works on streaming bandits have emerged in recent years. For Regret Minimization, Wang (2023) argues that a simple uniform exploration algorithm achieves $\tilde{O}\big(K^{\frac{1}{3}}T^{\frac{2}{3}}\big)$ expected regret and further shows it is optimal by establishing a $\Omega\big(K^{\frac{1}{3}}T^{\frac{2}{3}}\big)$ regret lower bound for single-pass algorithms. As for the multi-pass scenario, Agarwal, Khanna, and Patil (2022) design an algorithm with $\tilde{O}\big(T^{\frac{1}{2}+\frac{1}{2^{B+2}-2}}\sqrt{KB}\big)$ high probability regret upper bound, where B is the maximum number of passes, and show that the bound is tight with respect to T. For Best Arm Identification (BAI), Maiti, Patil, and Khan (2020) propose a single-pass (ε,δ)-BAI algorithm with sample complexity $\tilde{O}\big(\frac{K}{\varepsilon^2}\ln\frac{1}{\delta}\big)$, where (ε,δ)-BAI means that, with probability at least $1-\delta$, the algorithm identifies an ε-best arm. Jin et al. (2021) introduce a single-pass (ε,δ,k)-KAI algorithm with sample complexity $\tilde{O}\big(\frac{K}{\varepsilon^2}\ln\frac{k}{\delta}\big)$, where (ε,δ,k)-KAI means that, with probability at least $1-\delta$, the algorithm identifies k arms whose reward means are lower than that of the k-th best arm by at most ε. Jin et al. (2021) also present a strict BAI algorithm with near optimal instance-dependent sample complexity, using $O(\ln\frac{1}{\Delta})$ passes in expectation, where Δ is the gap between the mean of the best arm and that of the second-best arm. Complementarily, Assadi and Wang (2023) show that any streaming algorithm with optimal sample complexity requires $\Omega\big(\frac{\ln(1/\Delta)}{\ln\ln(1/\Delta)}\big)$ passes. To the best of our knowledge, the fixed-budget BAI problem, where the goal is to minimize the error probability given a budget constraint, has not yet been explored in the streaming MAB setup.

Linear Bandit. For Regret Minimization, a well-known paper (Abbasi-yadkori, Pál, and Szepesvári 2011) proposes a near-optimal algorithm, OFUL. A recent paper (Yang et al. 2022) also considers the scenario of an extremely large arm set, as we do, but with a different purpose. The authors attempt to reduce the time complexity of scanning through the whole arm set but still assume arbitrary access to every arm in the set. The BAI problem has also been extensively studied in the linear bandit literature, including the fixed-confidence setting (Soare, Lazaric, and Munos 2014; Soare 2015; Fiez et al. 2019; Zaki, Mohan, and Gopalan 2020; Jedra and Proutiere 2020; Camilleri et al. 2021; Jourdan and Degenne 2022) and the fixed-budget setting (Alieva, Cutkosky, and Das 2021; Yang and Tan 2022). Most of the BAI algorithms proposed in the literature belong to the successive elimination framework. The procedure is divided into several rounds. In each round, the algorithms update their estimation for the underlying parameter vector and discard a fraction of arms with relatively low empirical means. To the best of our knowledge, no algorithms in the linear bandit literature are directly applicable to our Linear Streaming Bandit setting, as their operations heavily rely on simultaneous access to all arms in the arm set.

Problem Setup

Notation. Throughout the paper, we let $\langle \cdot, \cdot \rangle$ denote the inner product between two vectors, let $|| \cdot ||_p$ denote the ℓ^p-norm of a vector, let $||x||_V$, where V is a positive semi-definite matrix, denote $\sqrt{x^T V x}$, let $|\mathcal{S}|$ denote the cardinality of a set $\mathcal{S}$, let $[N]$ denote the set of the smallest N positive integers: $\{1, 2, ..., N\}$.

Linear Streaming Bandit

We introduce our linear streaming bandit problem setup. There is a set of arms $\mathcal{K} \subseteq \mathbb{R}^d$, where an arm $a \in \mathcal{K}$ is a vector of dimension d. The vector, also called a profile, can be interpreted as a list of characteristics of the arm that are potentially related to its reward distribution. Let $K = |\mathcal{K}|$ denote the total number of arms. At each time slot t, the algorithm pulls an arm a_t which yields a reward $r_t \in \mathbb{R}$. As in the conventional linear bandit setting, we assume that $\exists \theta^* \in \mathbb{R}^d$, such that the reward satisfies $r_t = \langle a_t, \theta^* \rangle + \eta_t$, where $\{\eta_t\}_{t \geq 1}$ is a sequence of σ-subGaussian random noise, i.e.,

$$\mathbb{E}\Big[\exp(\alpha \eta_t) \,\Big|\, \mathcal{F}_{t-1} \Big] \leq \exp\Big(\frac{\alpha^2 \sigma^2}{2}\Big), \quad \forall \alpha \in \mathbb{R}, t \geq 1,$$

with the filtration $\mathcal{F}_{t-1} = \sigma(a_1, r_1, ..., a_t)$. We call $\mu_a := \langle a, \theta^* \rangle$ the expected/mean reward of arm a, where the true parameter vector θ^* is unknown to the algorithm. Define the optimal arm $a^* = \arg\max_{a \in \mathcal{K}} \mu_a$. Define the reward gap of arm a to be $\Delta_a = \langle a^*, \theta^* \rangle - \langle a, \theta^* \rangle$. Let $\Delta_{[i]}$ denote the i-th smallest reward gap for any $i \in [K]$. It is obvious that $\Delta_{[1]} = 0$. Besides, let Δ denote $\Delta_{[2]}$ for notational simplicity. We assume that the arm set is uniformly upper bounded: $||a||_2 \leq L, \forall a \in \mathcal{K}$, where L is a positive constant. We also assume that the ℓ^2-norm of θ^* is upper bounded by a positive constant S: $||\theta^*||_2 \leq S$.

The primary distinction between our model and those in the linear bandit literature is that, instead of assuming the algorithm has simultaneous access to every arm $a \in \mathcal{K}$, we assume the arms are presented to the algorithm sequentially in a streaming manner. No restrictions are placed on the order of the arms; the order can be arbitrary or even adversarially chosen. The length of the stream K can be extremely large, while the algorithm typically has fixed and limited storage resources. Consequently, the algorithm cannot simultaneously record the information about all arms in its storage. We assume the algorithm can remember at most M arms in its memory at any given time, where M remains constant as K grows and typically $M \ll K$. It can also store intermediate data, provided the storage size required for this data remains constant with respect to K. When the algorithm encounters an arm in the stream, it observes the arm's profile and decides whether to store it in memory. Only arms currently stored in memory can be pulled. The algorithm can remove arms from memory to accommodate new arms from the remaining stream. However, discarded arms are forgotten and cannot be retrieved into memory unless the algorithm revisits the arm stream. Let $B \in \mathbb{N}^+$ denote the maximum number of passes the algorithm is allowed over the arm stream. The order of arms may differ completely between passes. The multi-pass scenario has been widely considered in the literature (Agarwal, Khanna, and Patil 2022; Assadi and Wang 2023). In certain cases, the algorithm can indeed scan the arm sequence multiple times. For instance, workers on crowdsourcing platforms may return to the platform repeatedly after completing their current tasks in search of additional work. In the remainder of this section, we introduce the two objectives considered in the aforementioned model.

Regret Minimization. In this problem, the algorithm aims to minimize cumulative regret. T is interpreted as the time horizon in this scenario. At each time slot $t \in [T]$, the algorithm pulls an arm a_t and receives a reward r_t. Formally, the pseudo-regret is defined as

$$R_T = \sum_{t=1}^{T} \langle \theta^*, a^* \rangle - \sum_{t=1}^{T} \langle \theta^*, a_t \rangle = \sum_{t=1}^{T} \langle \theta^*, a^* - a_t \rangle.$$

ϵ-Best Arm Identification. In Best Arm Identification, the algorithm's objective is to identify an arm in the arm set $\mathcal{K}$ that has the highest expected reward among all others. As in the Multi-Armed Bandit (MAB) literature, there are two complementary variants of best arm identification (Lattimore and Szepesvári 2020): (1) Fixed-Confidence setting. The algorithm is given a constant confidence parameter $\delta \in (0, 1)$ and outputs an arm prediction $\hat{a} \in \mathcal{K}$ such that $\mathbb{P}(\hat{a} = a^*) \geq 1 - \delta$, i.e., the prediction is correct with probability at least $1 - \delta$, while using as few samples as possible. (2) Fixed-Budget setting. The algorithm is given a sample budget constraint $T \in \mathbb{N}^+$. Its objective is to minimize the error probability $\mathbb{P}(\hat{a} \neq a^*)$ with at most T arm pulls.

In this paper, we consider a Fixed-Budget ϵ-Best Arm Identification problem in the proposed Linear Streaming Bandit setup. Here, T represents the sample budget, while B denotes the pass budget. That is, the algorithm is allowed to pull at most T arms and scan at most B passes over the arm stream. Once the algorithm has pulled T arms, it must output the prediction $\hat{a}$ and terminate. When the algorithm has already scanned B passes, it cannot initiate a new scan but may still pull arms stored in memory until the sample budget T is exhausted. Given a fixed approximation parameter $\epsilon > 0$, the algorithm's objective is to minimize $\mathbb{P}(\Delta_{\hat{a}} > \epsilon)$.

Regret Minimization

In this section, we introduce our algorithm for regret minimization in the linear streaming bandit setup. We begin by describing the main idea behind it. Suppose the algorithm aims to achieve a target precision $\epsilon > 0$ in estimating the underlying parameter vector θ^* by going through the arm stream. The most straightforward solution might be to scan the stream in a single pass and pull a sufficient number of arms until the precision target is met. However, in the streaming scenario, the algorithm must be overly conservative to avoid missing the optimal arm. This conservativeness can result in wasting the budget on poor arms, especially when the optimal arm appears very late in the stream. One of the key aspects of our algorithm design is utilizing the opportunity to access the arm stream multiple times to mitigate this issue. A feasible approach is to select another precision parameter $\epsilon' > \epsilon$. In the first pass, the algorithm

achieves the less stringent precision target ϵ' using significantly fewer samples. In the second pass, the algorithm aims to achieve the stricter precision target ϵ, leveraging the additional knowledge obtained during the first pass. Specifically, when an arm a is encountered, the algorithm reads the arm's profile without pulling it and immediately determines whether $\Delta_a \geq \Theta(\epsilon')$. By avoiding the need to pull highly sub-optimal arms, the cumulative regret can be effectively reduced.

We present the pseudo-code of our algorithm CR-MPS in Algorithm 1. As the pass index b increases, the algorithm selects smaller precision parameters ϵ_b. The algorithm stores at most two arms, $\hat{a}$ and $\tilde{a}$, simultaneously. $\hat{a}$ records the empirically optimal arm encountered in the current pass, while $\tilde{a}$ represents the current arm in this pass. In Line 6, the algorithm checks whether the current arm is highly sub-optimal. If not, it keeps pulling this arm until the precision in this arm's direction is lower than ϵ_b. Then the algorithm updates $\hat{a}$ if the arm $\tilde{a}$ is believed to be temporarily optimal in Line 13. When the algorithm has already scanned the arm stream B times, it enters the exploitation phase, where it repeats the empirically optimal arm $\hat{a}$ until time T. Theorem 1 gives a high probability pseudo-regret upper bound of CR-MPS.

Theorem 1. *Run CR-MPS algorithm in Algorithm 1 with confidence parameter $\delta \in (0,1)$, regularization parameter $\lambda > 0$, and precision parameters ϵ_0, ϵ_1 set as*

$$\epsilon_0 = \frac{LS}{3\sqrt{\beta_T}},$$

$$\epsilon_1 = \left(6T^{-1}Bd\ln\left(1+\frac{TL^2}{\lambda d}\right)\right)^{\frac{2^B-1}{2^{B+1}-1}} \epsilon_0^{\frac{2^B-1}{2^{B+1}-1}},$$

where $\sqrt{\beta_t} := \sqrt{\lambda}S + \sigma\sqrt{2\ln\frac{1}{\delta} + d\ln(1+\frac{tL^2}{\lambda d})}$ for any $t \in [T]$. We have that $\exists N_0 \in \mathbb{N}^+$ such that for any $T \geq N_0$, the pseudo-regret R_T is upper bounded by

$$4(LS/3)^{\frac{1}{2^{B+1}-1}}\left[6\beta_T Bd\ln\left(1+\frac{TL^2}{\lambda d}\right)\right]^{\frac{2^B-1}{2^{B+1}-1}} T^{\frac{2^B}{2^{B+1}-1}}$$

with probability at least $1-\delta$. Thus, with probability at least $1-\delta$, $R_T = \tilde{O}\big((d^2B)^{\frac{2^B-1}{2^{B+1}-1}} T^{\frac{2^B}{2^{B+1}-1}}\big)$.

Remark. It is important to note that the regret upper bound is independent of the total number of arms K. This implies that the algorithm is effective in handling arbitrarily long arm sequences, achieving sub-linear regret with respect to T, since $\alpha := \frac{2^B}{2^{B+1}-1} \leq \frac{2}{3}$ and $\alpha \downarrow \frac{1}{2}$ as $B \to \infty$. Consequently, the order of our regret upper bound for CR-MPS approaches $\Omega(\sqrt{T})$, a well-known regret lower bound for the classic Linear Bandit setting (Lattimore and Szepesvári 2020), when B is sufficiently large. Moreover, the CR-MPS algorithm only requires an arm memory of size $M = 2 \ll K$.

Proof of Theorem 1. For any $\delta > 0$, we define a good event

$$\mathcal{G} = \left\{\forall t \geq 0, x \in \mathbb{R}^d : |\langle x, \theta^* - \hat{\theta}_t\rangle| \leq ||x||_{V_t^{-1}}\sqrt{\beta_t}\right\},$$

Algorithm 1: Confidence Radius-directed Multi-Pass Sampling (CR-MPS)

Input: δ confidence parameter, λ regularization parameter, ϵ_0, ϵ_1 precision parameters

/* Initialization */

1 $t \leftarrow 1, V \leftarrow \lambda I, S \leftarrow 0, \hat{\mu}_{\text{last}} \leftarrow 0, \hat{\mu} \leftarrow 0$

2 **for** *pass* $b = 1, 2, ..., B$ **do**

/* We set $\epsilon_b = \epsilon_1^{\frac{2^b-1}{2^{b-1}}}\epsilon_0^{-\frac{2^{b-1}-1}{2^{b-1}}}$ */

3 $\hat{a} \leftarrow null$

/* Scan the b-th pass */

4 **for** $k = 1, 2, ..., K$ **do**

5 Read and store arm $a_{k,b}$ to $\tilde{a}$

6 **if** $b = 1$ *or* $b > 1$ *and* $\langle\tilde{a}, V^{-1}S\rangle \geq \hat{\mu}_{last} - 2\sqrt{\beta_T}\epsilon_1^{\frac{2^{b-1}-1}{2^{b-2}}}\epsilon_0^{-\frac{2^{b-2}-1}{2^{b-2}}}$ **then**

7 **while** $||\tilde{a}||_{V^{-1}} \geq \epsilon_1^{\frac{2^b-1}{2^{b-1}}}\epsilon_0^{-\frac{2^{b-1}-1}{2^{b-1}}}$ **do**

8 Pull arm $a_t \leftarrow \tilde{a}$ and obtain reward r_t

9 $V \leftarrow V + \tilde{a}\tilde{a}^T$

10 $S \leftarrow S + r_t\tilde{a}$

11 $t \leftarrow t + 1$

12 **end**

13 **if** $\hat{a} = null$ *or* $\langle\tilde{a}, V^{-1}S\rangle > \hat{\mu}$ **then**

14 $\hat{\mu} \leftarrow \langle\tilde{a}, V^{-1}S\rangle$

15 $\hat{a} \leftarrow \tilde{a}$

16 **end**

17 **end**

18 **end**

19 $\hat{\mu}_{\text{last}} \leftarrow \hat{\mu}$

20 **end**

/* Exploitation */

21 **while** $t \leq T$ **do**

22 Pull arm $a_t \leftarrow \hat{a}$

23 $t \leftarrow t + 1$

24 **end**

which means that the estimators $\hat{\theta}_t$ (computed as $V^{-1}S$ in the algorithm) for θ^* are accurate enough. It can be shown that this good event happens with high probability:

Lemma 1. $\mathbb{P}(\mathcal{G}) \geq 1 - \delta$.

We analyze the algorithm behavior under the event $\mathcal{G}$.

Step I. Utilizing the accuracy of $\hat{a}$

The algorithm stores a total of two arms, $\tilde{a}$ and $\hat{a}$. $\tilde{a}$ is the current arm that the algorithm is exploring, while $\hat{a}$ stores the empirically best arm observed so far. In the b-th pass, before each comparison between $\tilde{a}$ and $\hat{a}$, the algorithm repeatedly samples $\tilde{a}$ to ensure that both arms have confidence radii upper bounded by ϵ_b. Let $\hat{a}_b$ represent the content of $\hat{a}$ at the end of the b-th pass. We first show that $\hat{a}_b$ is, to some extent, a good predictor for a^*. Define $T_{k,b}$ to be the number of times arm $a_{k,b}$ is pulled in the b-th pass, and $T_b = \sum_{k=1}^K T_{k,b}$.

Algorithm 2: ϵ-Grid Multi-Pass Successive Elimination (G-MP-SE)

Input: ϵ approximation parameter, $\tilde{T}$ sample budget parameter, B number of passes

Initialize the grid set $\mathcal{G}_\epsilon$ in the memory
Initialize $t \leftarrow 1$ and $C_b(g) \leftarrow 0$ for each $g \in \mathcal{G}_\epsilon, b \in [B-1]$
for *arm* a *in the first pass* **do**
  Find $g(a) \leftarrow \arg\min_{h \in \mathcal{G}_\epsilon} ||a - h||_2$
  $C_1(g(a)) \leftarrow C_1(g(a)) + 1$
end
for *pass* $b = 2, ..., B-1$ **do**
  Allocate the sample budget: $\boldsymbol{D}_b \leftarrow \text{SBA}(\mathcal{G}_\epsilon, \boldsymbol{C}_{b-1}, \boldsymbol{C}_1, \tilde{T})$
  Initialize $V_b \leftarrow 0, S_b \leftarrow 0$ and $\boldsymbol{c}$ as a counter with initial value 1 for each $g \in \mathcal{G}_\epsilon$
  for *arm* a *in pass* b **do**
    Find $g(a) \leftarrow \arg\min_{h \in \mathcal{G}_\epsilon} ||a - h||_2$
    for $s \in [D_b(g(a), c_{g(a)})]$ **do**
      Pull arm $a_t \leftarrow a$ and observe reward r_t
      $V_b \leftarrow V_b + a_t a_t^T$
      $S_b \leftarrow S_b + r_t a_t$
      $t \leftarrow t + 1$
    end
    $c_{g(a)} \leftarrow c_{g(a)} + 1$
  end
  $\hat{\theta}_b \leftarrow V_b^{-1} S_b, N_b \leftarrow \lceil K^{\frac{B-b-1}{B-2}} \rceil$
  $\boldsymbol{C}_b \leftarrow \text{C-Update}(\mathcal{G}_\epsilon, \boldsymbol{C}_{b-1}, \hat{\theta}_b, N_b)$
end
Scan the final pass, set $\hat{a}$ as the first a that is rounded to a grid point g with $C_{B-1}(g) > 0$

Output: $\hat{a}$

Lemma 2. *Under event $\mathcal{G}$, for any $1 \le b \le B$, $\hat{a}_b$ satisfies*

$$\mu_{\hat{a}_b} \ge \mu^* - 2\sqrt{\beta_T}\epsilon_b, \tag{1}$$

where $\mu^* := \mu_{a^*}$.

Thanks to the prediction accuracy of $\hat{a}_{b-1}$, the algorithm can effectively skip arms that are clearly sub-optimal compared to $\hat{a}_{b-1}$ in the b-th pass. For $b > 1$, if the algorithm visits arm $\tilde{a} = a_{k,b}$ in the b-th pass, we now have

$$\begin{aligned}
\Delta_{\tilde{a}} =& \langle a^* - \tilde{a}, \theta^* \rangle \le \langle \hat{a}_{b-1}, \theta^* \rangle + 2\sqrt{\beta_T}\epsilon_{b-1} - \langle \tilde{a}, \theta^* \rangle \\
\le& \langle \hat{a}_{b-1}, \theta^* \rangle + 2\sqrt{\beta_T}\epsilon_{b-1} \\
& - \langle \tilde{a}, \hat{\theta}_{t_{k-1}^{(b)}} \rangle + \sqrt{\beta_{t_{k-1}^{(b)}}} ||\tilde{a}||_{V^{-1}_{t_{k-1}^{(b)}}} \\
\le& \langle \hat{a}_{b-1}, \hat{\theta}_{t_{i'}^{(b-1)}} \rangle - \langle \tilde{a}, \hat{\theta}_{t_{k-1}^{(b)}} \rangle \\
& + 3\sqrt{\beta_T}\epsilon_{b-1} + \sqrt{\beta_{t_{i'}^{(b-1)}}} ||\hat{a}_{b-1}||_{V^{-1}_{t_{i'}^{(b-1)}}} \\
\le& 6\sqrt{\beta_T}\epsilon_{b-1},
\end{aligned}$$

where i' is the index of $\hat{a}_{b-1}$ in the $(b-1)$-th pass and $t_k^{(b)} := \sum_{s=1}^{k} T_{s,b} + \sum_{b'=1}^{b-1} T_{b'}$. The first inequality is by (1) and the last is due to the fact that arm $\tilde{a}$ satisfies the condition in Line 6. For the first pass, we only know a uniform upper bound $\Delta_{\tilde{a}} \le 2||\theta^*||_2 \max_{a \in \mathcal{A}} ||a||_2 \le 2LS =: \bar{\Delta}$. Now we start decomposing the pseudo-regret,

$$\begin{aligned}
R_T =& \sum_{t=1}^{T} \langle a^* - a_t, \theta^* \rangle = \sum_{b=1}^{B} \sum_{k=1}^{K} T_{k,b} \langle a^* - a_{k,b}, \theta^* \rangle \\
& + \Big(T - \sum_{b=1}^{B} T_b\Big) \langle a^* - \hat{a}_B, \theta^* \rangle \\
\le& T_1 \bar{\Delta} + \sum_{b=2}^{B} 6T_b \sqrt{\beta_T}\epsilon_{b-1} + T\langle a^* - \hat{a}_B, \theta^* \rangle \\
\le& T_1 \bar{\Delta} + \sum_{b=2}^{B} 6T_b \sqrt{\beta_T}\epsilon_{b-1} + 2T\sqrt{\beta_T}\epsilon_B.
\end{aligned}$$

Step II. Upper-bounding $T_b, \forall b \in [B]$

Algorithm 3: Sample Budget Assignment (SBA)

Input: $\mathcal{G}_\epsilon$ grid set, $\boldsymbol{C}$ counter of active arms on each grid point, $\boldsymbol{C}_1$ initial counter, $\tilde{T}$ sample budget parameter

Initialize the budget assignment $D(g, c)$ to be 0 for each $g \in \mathcal{G}_\epsilon, c \in [C_1(g)]$
Compute the design λ
Allocate the sample budget to each grid point: $N(g) \leftarrow \lceil \lambda_g \tilde{T} \rceil$ for each $g \in \mathcal{G}_\epsilon$
for $g \in \mathcal{G}_\epsilon$ **do**
  if $N(g) > 0$ **then**
    for $i \in [N(g)]$ **do**
      Sample $c \sim \text{Unif}([C_1(g)])$
      $D(g, c) \leftarrow D(g, c) + 1$
    end
  end
end

Output: $\boldsymbol{D}$ budget assignment

Algorithm 4: Active Arm Counter Update (C-Update)

Input: $\mathcal{G}_\epsilon$ grid set, $\boldsymbol{C}_0$ counter of active arms on each grid point, $\hat{\theta}$ estimator for the true parameter, N number of remaining arms

Initialize $C(g) \leftarrow 0$ for each $g \in \mathcal{G}_\epsilon$
Sort $\boldsymbol{g}'$ as an ordered list of $g \in \mathcal{G}_\epsilon$ with decreasing values of $\langle g, \hat{\theta} \rangle$
for $g \in \boldsymbol{g}'$ **do**
  $C(g) \leftarrow \min\{C_0(g), N\}$
  $N \leftarrow \max\{N - C_0(g), 0\}$
  if $N = 0$ **then**
    break
  end
end

Output: $\boldsymbol{C}$ updated active arm counter

The algorithm sets

$$\epsilon_b = \epsilon_1^{\frac{2^b-1}{2^{b-1}}} \epsilon_0^{-\frac{2^{b-1}-1}{2^{b-1}}} \quad (\forall b \geq 1)$$

and it can be verified that $\exists N_0 \in \mathbb{N}^+$ such that $\forall T \geq N_0$, $\epsilon_B^2 \leq ... \leq \epsilon_1^2 \leq \frac{5}{2}$. For any $t \in [T]$, we have

$$\det(V_t) = \det(V_{t-1} + a_t a_t^T) = \det(V_{t-1})(1 + ||a_t||^2_{V_{t-1}^{-1}}).$$

Since the algorithm pulls each arm a_t until its corresponding confidence radius is below ϵ_b in the b-th pass, we make an important observation that $||a_t||_{V_{t-1}^{-1}} \geq \epsilon_b, \forall t : \sum_{b'=1}^{b-1} T_{b'} < t \leq \sum_{b'=1}^{b} T_{b'}$. Thus for any $b \in [B]$, if $T \geq N_0$,

$$\begin{aligned}
&\det\left(V_{\sum_{b'=1}^{b} T_{b'}}\right) \\
=&\det\left(V_{\sum_{b'=1}^{b-1} T_{b'}}\right) \prod_{s=1+\sum_{b'=1}^{b-1} T_{b'}}^{\sum_{b'=1}^{b} T_{b'}} (1 + ||a_s||^2_{V_{s-1}^{-1}}) \\
\geq&\det(\lambda I)(1+\epsilon_b^2)^{T_b} \\
\geq&\det(\lambda I)\exp\Big(\frac{T_b \epsilon_b^2}{2}\Big),
\end{aligned}$$

where the last inequality is due to the fact that $e^{\frac{x}{2}} \leq 1 + x, \forall x \in [0, \frac{5}{2}]$ and $\forall T \geq N_0, \epsilon_B^2 \leq ... \leq \epsilon_1^2 \leq \frac{5}{2}$. Thus

$$\begin{aligned}
T_b \leq& \frac{2}{\epsilon_b^2} \ln \frac{\det\left(V_{\sum_{b'=1}^{b} T_{b'}}\right)}{\det(\lambda I)} \leq \frac{2}{\epsilon_b^2} \ln \frac{(\lambda + TL^2/d)^d}{\lambda^d} \\
=& \frac{2d}{\epsilon_b^2} \ln\Big(1 + \frac{TL^2}{\lambda d}\Big),
\end{aligned}$$

where the last inequality is by Lemma 10 in Abbasi-yadkori, Pál, and Szepesvári (2011).

Step III. Wrapping up the proof

By our previous discussion, under the good event $\mathcal{G}$,

$$\begin{aligned}
R_T \leq& \sum_{b=1}^{B} 6T_b\sqrt{\beta_T}\epsilon_{b-1} + 2T\sqrt{\beta_T}\epsilon_B \\
\leq& 2\sqrt{\beta_T}\left[6d\ln\Big(1+\frac{TL^2}{\lambda d}\Big)\sum_{b=1}^{B}\frac{\epsilon_{b-1}}{\epsilon_b^2} + \epsilon_B T\right] \\
=& 2\sqrt{\beta_T}\left[6Bd\ln\Big(1+\frac{TL^2}{\lambda d}\Big)\frac{\epsilon_0}{\epsilon_1^2} + \epsilon_B T\right].
\end{aligned}$$

By substituting the values of ϵ_0, ϵ_1, and ϵ_B into the RHS, we complete the proof. □

Fixed-Budget ϵ-Best Arm Identification

A Multi-Pass Algorithm

In this section, we propose a streaming algorithm that is good at handling an extremely large arm set, given a budget constraint T and a pass constraint B. The key idea is to discretize the arm space with a grid set $\mathcal{G}_\epsilon$ given a fixed approximation parameter $\epsilon > 0$. The grid set $\mathcal{G}_\epsilon$ satisfies that, for every arm $a \in \mathcal{K}$, there exists a grid point $g \in \mathcal{G}_\epsilon$ such that the ℓ^2 distance $||a - g||_2 \leq \epsilon$. For example, $\mathcal{G}_\epsilon$ can be $\big\{\frac{\epsilon}{\sqrt{d}}\boldsymbol{x} \mid \boldsymbol{x} \in \mathbb{Z}^d, \big|\big|\frac{\epsilon}{\sqrt{d}}\boldsymbol{x}\big|\big|_2 \leq L + \epsilon\big\}$. Specifically, arm a is rounded to a grid point $g(a) := \arg\min_{h \in \mathcal{G}_\epsilon} ||a - h||_2$. In this way, the algorithm only needs to maintain a counter $\boldsymbol{C}$ associated with $\mathcal{G}_\epsilon$, recording the number of arms rounded to each grid point, whose size is independent of K.

Algorithm 5: Single-Pass Comparison (SPC)

Input: η regularization parameter

/* Initialization */

$t \leftarrow 1, V \leftarrow \eta I, S \leftarrow 0, \hat{a} \leftarrow null$

$\epsilon \leftarrow \sqrt{\frac{2d}{T}\ln\Big(1 + \frac{TL^2}{\eta d}\Big)}$

for *each arm a in the stream* **do**

  Read and store arm a to $\tilde{a}$

  while $||a||_{V^{-1}} \geq \epsilon$ **do**

    Pull arm $a_t \leftarrow \tilde{a}$ and obtain reward r_t

    $V \leftarrow V + \tilde{a}\tilde{a}^T$

    $S \leftarrow S + r_t\tilde{a}$

    $t \leftarrow t + 1$

  end

  if $\hat{a} = null$ *or* $\langle V^{-1}S, \hat{a}\rangle < \langle V^{-1}S, \tilde{a}\rangle$ **then**

    $\hat{a} \leftarrow \tilde{a}$

  end

end

Output: $\hat{a}$

We present the pseudo-code of our algorithm G-MP-SE in Algorithm 2. The entire process can be divided into three phases: the first pass, passes $b = 2$ to $b = B - 1$, and the last pass. In the first pass, the algorithm constructs the counter $\boldsymbol{C}_1$ without pulling any arms. In each pass b of the second phase, a budget plan is assigned to each grid point. Algorithm 3 employs a randomized yet non-unique implementation of this plan. Alternative correct implementations do not affect our analysis. When the scanning of pass b is finished, the algorithm computes a new estimator $\hat{\theta}_b$ for θ^* and uses it to calculate the empirical means of each active grid point (a grid point is active if and only if the counter on it is still positive). The algorithm then eliminates the grid points with relatively low empirical means by setting their counters to zero. As the third phase begins, there remains a unique grid point g whose counter is still positive. An arm corresponding to this g is output as the final prediction. The algorithm requires only an arm memory of size $M = 1$ and needs to store some additional data, the amount of which is independent of K. An asymptotic error probability upper bound for G-MP-SE is derived in Theorem 2.

Theorem 2. *Run G-MP-SE algorithm with input* $(\epsilon, \tilde{T} = \frac{T}{B-2} - \frac{d(d+1)}{2}, B)$ *and design* $\lambda^{(b)} = \lambda_{b-1}^*, b = 2, ..., B-1$, *where*

$$\lambda_b^* := \arg\min_{\lambda \in \mathcal{P}(\mathcal{K}_b(\epsilon))} \max_{g' \in \mathcal{K}_b(\epsilon)} ||g'||^2_{\left(\sum_{g \in \mathcal{K}_b(\epsilon)} \lambda_g g g^T\right)^{-1}},$$

$$\mathcal{K}_b(\epsilon) := \{g \in \mathcal{G}_\epsilon \mid C_b(g) > 0\},$$

and $\mathcal{P}(\Omega)$ *is the set of all probability measures over* Ω. *If*

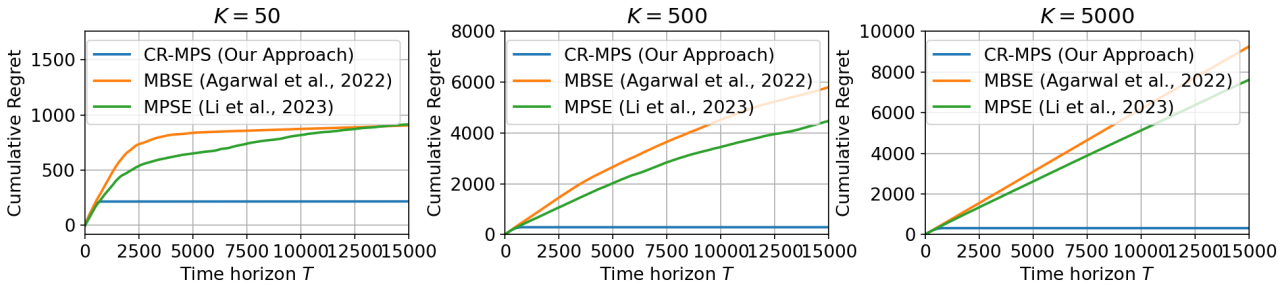

Figure 2: Experiment results for the regret minimization task on a synthetic dataset with various numbers of arms.

$T > d(d+1)(B-2)/2$, *we have that*

$$\lim_{\epsilon \downarrow 0} \mathbb{P}\big(\Delta_{\hat{a}} > 2||\theta^*||_2 \epsilon\big) \leq (B-2)\big(2K^{\frac{1}{B-2}} + 1\big) \times \exp\Big(-\frac{\Delta^2}{4\sigma^2 d}\big(\frac{T}{B-2} - \frac{d(d+1)}{2}\big)\Big).$$

Remark. The advantages of G-MP-SE are twofold: Firstly, state-of-the-art algorithms in the linear bandit scenario are unable to handle a large number of streaming arms given limited memory. For example, in each of its arm elimination phases, OD-LinBAI in Yang and Tan (2022) not only computes the G-optimal design for all the remaining arms but also sorts all of them according to their empirical reward means. These operations heavily rely on complete access to the entire arm set, which is infeasible in the streaming arm setting. Other algorithms, such as PELEG in Zaki, Mohan, and Gopalan (2020), RAGE in Fiez et al. (2019), and $\mathcal{XY}$-Adaptive in Soare, Lazaric, and Munos (2014), also have this issue, though they are specialized for the fixed-confidence setting. Secondly, the error probability upper bound of OD-LinBAI grows linearly with the number of arms K, whereas the error probability upper bound of our algorithm scales with $K^{\frac{1}{B-2}}$, which is a much weaker dependency on K when B is large.

A Single-Pass Algorithm

We note that the proposed G-MP-SE algorithm is only applicable in the multi-pass scenario (requiring $B > 2$). Here, we also design an alternative algorithm that is feasible even when $B = 1$ or 2. The pseudo-code for this single-pass strict Best Arm Identification algorithm is provided in Algorithm 5. This algorithm sets a precision parameter ϵ and scans a single pass through the arm stream, searching for the optimal arm. This choice of ϵ minimizes the error probability while ensuring that the sample budget T is never exceeded. This algorithm requires only an arm memory of size $M = 2$. An error probability upper bound is given in Theorem 3.

Theorem 3. *Run SPC algorithm in Algorithm 5 with regularization parameter $\eta > 0$. If the sample budget T is sufficiently large such that $2d \ln\left(1 + \frac{TL^2}{\eta d}\right) \leq \frac{5}{2}T$, then the probability that the algorithm makes an incorrect prediction*

$$\mathbb{P}(\hat{a} \neq a^*) \leq (K-1)\exp\Big(-\frac{T\Delta^2}{8\sigma^2 d \ln\left(1 + \frac{TL^2}{\eta d}\right)}\Big).$$

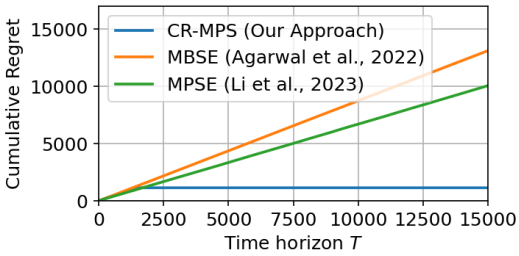

Figure 3: Experiment results for the regret minimization task on a real-world dataset.

Experimental Results

We implement our algorithms on both synthetic and real-world datasets. Due to space limitations, we only present the results for the CR-MPS algorithm here and defer the rest to the appendix.

Synthetic Datasets. We run CR-MPS against two baseline streaming bandit algorithms: MBSE in Agarwal, Khanna, and Patil (2022) and MPSE in Li et al. (2023). Their regret curves, averaged over $N = 20$ repetitions, are shown in Figure 2. In each independent repetition, we uniformly randomly sample $\mathcal{K}, \theta^*$ and run these algorithms. In our experiments, we set K to 50, 500, and 5000, respectively. As K increases, the baseline curves gradually become non-convergent, whereas our algorithm continues to perform well.

Real-World Dataset. The Kaggle dataset (Chaudhari 2023)[1] contains information on more than 17k anonymous workers, including their resume details and performance scores. We select $K = 10$k of them and run these algorithms again on this dataset. The regret curves are shown in Figure 3. The poor performance of the baseline algorithms is due to K being too large for them to handle.

Conclusion

In this paper, we introduce the Linear Streaming Bandit and propose solutions for regret minimization and fixed-budget ϵ-best arm identification tasks in this setup. The effectiveness of our algorithms is supported by both theoretical analysis and experimental results. Future research directions include developing matching lower bounds for both tasks.

[1] https://www.kaggle.com/datasets/sanjanchaudhari/employees-performance-for-hr-analytics/data

Acknowledgments

This work is supported by Tsinghua University Dushi Program and Shanghai Qi Zhi Institute Innovation Program SQZ202312.

References

Abbasi-yadkori, Y.; Pál, D.; and Szepesvári, C. 2011. Improved Algorithms for Linear Stochastic Bandits. In Shawe-Taylor, J.; Zemel, R.; Bartlett, P.; Pereira, F.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc.

Agarwal, A.; Khanna, S.; and Patil, P. 2022. A Sharp Memory-Regret Trade-off for Multi-Pass Streaming Bandits. In Loh, P.-L.; and Raginsky, M., eds., *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, 1423–1462. PMLR.

Alieva, A.; Cutkosky, A.; and Das, A. 2021. Robust pure exploration in linear bandits with limited budget. In *International Conference on Machine Learning*, 187–195. PMLR.

Assadi, S.; and Wang, C. 2020. Exploration with limited memory: streaming algorithms for coin tossing, noisy comparisons, and multi-armed bandits. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on theory of computing*, 1237–1250.

Assadi, S.; and Wang, C. 2022. Single-pass streaming lower bounds for multi-armed bandits exploration with instance-sensitive sample complexity. *Advances in Neural Information Processing Systems*, 35: 33066–33079.

Assadi, S.; and Wang, C. 2023. The best arm evades: Near-optimal multi-pass streaming lower bounds for pure exploration in multi-armed bandits. *arXiv preprint arXiv:2309.03145*.

Avadhanula, V.; Colini Baldeschi, R.; Leonardi, S.; Sankararaman, K. A.; and Schrijvers, O. 2021. Stochastic bandits for multi-platform budget optimization in online advertising. In *Proceedings of the Web Conference 2021*, 2805–2817.

Aziz, M.; Kaufmann, E.; and Riviere, M.-K. 2021. On multi-armed bandit designs for dose-finding trials. *Journal of Machine Learning Research*, 22(14): 1–38.

Camilleri, R.; Xiong, Z.; Fazel, M.; Jain, L.; and Jamieson, K. G. 2021. Selective sampling for online best-arm identification. *Advances in Neural Information Processing Systems*, 34: 11071–11082.

Chaudhari, S. 2023. Employee's Performance for HR Analytics. https://www.kaggle.com/ds/3537629. Accessed: 2025-01-03.

Fiez, T.; Jain, L.; Jamieson, K. G.; and Ratliff, L. 2019. Sequential Experimental Design for Transductive Linear Bandits. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Jedra, Y.; and Proutiere, A. 2020. Optimal Best-arm Identification in Linear Bandits. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 10007–10017. Curran Associates, Inc.

Jin, T.; Huang, K.; Tang, J.; and Xiao, X. 2021. Optimal Streaming Algorithms for Multi-Armed Bandits. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 5045–5054. PMLR.

Jourdan, M.; and Degenne, R. 2022. Choosing Answers in Epsilon-Best-Answer Identification for Linear Bandits. In *International Conference on Machine Learning*, 10384–10430. PMLR.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit algorithms*. Cambridge University Press.

Li, S.; Zhang, L.; Wang, J.; and Li, X.-Y. 2023. Tight memory-regret lower bounds for streaming bandits. *arXiv preprint arXiv:2306.07903*.

Maiti, A.; Patil, V.; and Khan, A. 2020. Streaming algorithms for stochastic multi-armed bandits. *arXiv preprint arXiv:2012.05142*.

Schwartz, E. M.; Bradlow, E. T.; and Fader, P. S. 2017. Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Science*, 36(4): 500–522.

Soare, M. 2015. *Sequential Resource Allocation in Linear Stochastic Bandits*. Theses, Université Lille 1 - Sciences et Technologies.

Soare, M.; Lazaric, A.; and Munos, R. 2014. Best-arm identification in linear bandits. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, 828–836. Cambridge, MA, USA: MIT Press.

Villar, S. S.; Bowden, J.; and Wason, J. 2015. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2): 199.

Wang, C. 2023. Tight regret bounds for single-pass streaming multi-armed bandits. In *International Conference on Machine Learning*, 35525–35547. PMLR.

Yang, H.; and Lu, Q. 2016. Dynamic contextual multi arm bandits in display advertisement. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 1305–1310. IEEE.

Yang, J.; and Tan, V. 2022. Minimax Optimal Fixed-Budget Best Arm Identification in Linear Bandits. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 12253–12266. Curran Associates, Inc.

Yang, S.; Ren, T.; Shakkottai, S.; Price, E.; Dhillon, I. S.; and Sanghavi, S. 2022. Linear bandit algorithms with sublinear time complexity. In *International Conference on Machine Learning*, 25241–25260. PMLR.

Zaki, M.; Mohan, A.; and Gopalan, A. 2020. Explicit best arm identification in linear bandits using no-regret learners. *arXiv preprint arXiv:2006.07562*.