Learning to Sparsify Stochastic Linear Bandits

Zhengmiao Wang[1,2] , **Ming Chi**[1,2] , **Zhi-Wei Liu**[1,2] , **Lintao Ye**[1,2,*] ,
Carla Fabiana Chiasserini[3]

[1]Key Laboratory of Image Processing and Intelligent Control, Ministry of Education
[2]School of Artificial Intelligence and Automation, Huazhong University of Science and Technology
[3]Department of Electronics and Telecommunications, Politecnico di Torino
{wangzhengmiao, chiming, zwliu, yelintao93}@hust.edu.cn, carla.chiasserini@polito.it

Abstract

This paper addresses the problem of learning to sparsify stochastic linear bandits, where a decision-maker sequentially selects actions from a high-dimensional space subject to a sparsity constraint on the number of nonzero elements in the action vector. The key challenge lies in minimizing cumulative regret while tackling the potential NP-hardness of finding optimal sparse actions due to the inherent combinatorial structure of the problem. We propose an adaptively phased exploration and exploitation algorithmic framework, utilizing ordinary least squares for parameter learning and specialized subroutines for sparse action selection. When the action set is a Euclidean ball, optimal sparse actions can be efficiently computed, enabling us to establish a $\tilde{\mathcal{O}}(d\sqrt{T})$ regret, where d is the dimension of the action vector and T is the time horizon length. For general convex and compact action sets where finding optimal sparse actions is intractable, we employ a greedy subroutine. For strongly convex action sets, we derive a $\tilde{\mathcal{O}}(d\sqrt{T})$ α-regret; for compact sets lacking strong convexity, we establish a $\tilde{\mathcal{O}}(dT^{2/3})$ α-regret, where α pertains to the approximation ratio of the greedy algorithm. We validate the performance of our algorithms using extensive experiments.

1 Introduction

Online decision-making under uncertainty, particularly the multi-armed bandit (MAB) framework, has been a central area of research [Auer *et al.*, 2002; Lattimore and Szepesvári, 2020]. Stochastic linear bandits model the reward at each decision step t as an inner product of an unknown parameter θ_* (i.e., context vector) and the action vector $\mathbf{x}_t$ plus noise [Abbasi-Yadkori *et al.*, 2011], offering a simple yet general approach to model sequential decision-making in recommendation systems and adaptive control [Li *et al.*, 2010; Chu *et al.*, 2011]. A key consideration in linear bandits for real-world settings is sparsity, i.e., the context vector is sparse and only a small subset of its features influences the rewards [Agrawal and Goyal, 2013; Bastani *et al.*, 2021]. Exploiting this sparsity, especially in high dimensions, can lead to more efficient algorithms and improved regret [Abbasi-Yadkori *et al.*, 2012; Russo *et al.*, 2018]. However, considering sparsity introduces an additional layer of complexity into the algorithm design and regret analysis [Oh *et al.*, 2021; Hao *et al.*, 2020; Dai *et al.*, 2023].

Linear Bandits with Sparse Context Vector. The earlier work [Abbasi-Yadkori *et al.*, 2012] established a $\tilde{\mathcal{O}}(\sqrt{HdT})$ regret bound for sparse linear bandits when the sparsity pattern of the unknown context vector is known, where H is the number of nonzero entries in $\theta_* \in \mathbb{R}^d$. Later, [Pacchiano *et al.*, 2022] studied sparse linear bandits with unknown sparsity pattern and [Jin *et al.*, 2024] proved a $\tilde{\mathcal{O}}(\sqrt{HdT})$ regret using a randomized model selection approach. The work of [Oh *et al.*, 2021; Ichikawa *et al.*, 2024] further studied the sparse contextual linear bandits setting. However, the aforementioned work assumed that the sparsity pattern of the context vector remains fixed and does not change over time.

Combinatorial Multi-Armed Bandits (CMAB). In the standard CMAB setting, a learner selects a subset of base arms (i.e., a super-arm) at each step, and the reward is given by a function of the individual reward of each arm [Chen *et al.*, 2013; Chen *et al.*, 2016]. For linear reward functions, an upper confidence bound method adapted from linear bandits has been proposed, which chooses a super-arm from the predefined set of base arms [Gai *et al.*, 2012]. For nonlinear reward functions, the recent work [Fourati *et al.*, 2024] proposes an online algorithm that achieves a $(1 - 1/e)$-regret bound of $\tilde{\mathcal{O}}(T^{2/3})$ for i.i.d. sampled monotone stochastic submodular reward functions, which leverages offline approximation algorithms as black-box oracles [Nie *et al.*, 2023]. A multi-agent grouped CMAB problem has been studied in [Vannella *et al.*, 2023a; Vannella *et al.*, 2023b], which uses a mean field approximation approach and characterizes its statistical and computational trade-off.

Online Combinatorial Optimization. A broader area of work studies cases when the decision-maker first chooses an action that is a subset of a ground set at each step, and then a set function of the chosen action is revealed to return the corresponding reward [Streeter and Golovin, 2008; Wan *et al.*, 2023]. Since the chosen action set often needs

*Corresponding author.

The full version of this paper that includes all the proofs can be found on arXiv as [Wang *et al.*, 2026].

to satisfy certain combinatorial constraints (e.g., cardinality constraints), the problem is also termed online combinatorial optimization [Golovin *et al.*, 2014]. A difficulty emerges when the offline version of the problem (i.e., when the set functions are known) is NP-hard, which necessitates the use of α-regret that compares the cumulative reward from an online algorithm against an approximately optimal reward with ratio α in hindsight [Streeter and Golovin, 2008]. The ratio $\alpha \in (0, 1)$ is set to be the approximation ratio of an offline algorithm [Das and Kempe, 2018].

Motivation. In the linear bandits with sparse context vector θ_*, the sparsity pattern of the context vector is a prescribed condition, i.e., when choosing the action vector $\mathbf{x}_t$, the learner cannot modify the sparsity of θ_*. Since the reward depends on the inner product between θ_* and $\mathbf{x}_t$, the sparsity of θ_* directly enforces the same sparsity pattern on $\mathbf{x}_t$. We now ask the following natural questions: *What if the learner must simultaneously determine the sparsity pattern of the action vector $\mathbf{x}_t$ and the non-zero entries of $\mathbf{x}_t$? Can we design algorithms for the learner to achieve meaningful regret?* We refer to the problem as *learning to sparsify stochastic linear bandits.* Such a problem arises in many applications, e.g., recommendation systems and stock marketing.

Contributions. Since finding an optimal sparse action is NP-hard in general, we are faced with the following challenge: *How to efficiently minimize cumulative regret when the action selection at each step constitutes a potentially NP-hard problem?* We tackle this challenge by introducing a novel algorithmic framework for learning to sparsify stochastic linear bandits. We propose an adaptively phased exploration and exploitation algorithm that *decouples* parameter learning via Ordinary Least Squares (OLS) from the combinatorial action selection, and employs subroutines for efficient selection of sparse actions. Our key contributions are:

- We show that an optimal sparse action is tractable under a Euclidean ball action set $\mathcal{X}$. We introduce a novel adaptive warm-up strategy with a data-driven stopping condition that does not rely on prior knowledge of the unknown parameters. Leveraging this adaptive strategy to recover the support of the optimal sparse action, we establish a $\tilde{\mathcal{O}}(d\sqrt{T})$ regret.
- When finding optimal sparse solutions becomes intractable for general action sets, we propose a combinatorial greedy algorithm as a subroutine to select a sparse action efficiently. Based on the geometry of the action set, we distinguish between two cases: i) For *strongly convex* action sets (e.g., an ellipsoid), we propose an algorithm that adaptively recovers the support of a greedy sparse action via the notion of greedy gap, and prove a $\tilde{\mathcal{O}}(d\sqrt{T})$ α-regret, relying on Lipschitz continuity of the optimal action map due to strongly convexity. ii) For *general compact* action sets (e.g., a hypercube), we show that a non-adaptive phased exploration and exploitation algorithm achieves a $\tilde{\mathcal{O}}(dT^{2/3})$ α-regret.
- Finally, we validate our theoretical results using comprehensive numerical experiments across action sets with different geometries, including a recommendation system instance.

Our regret results for Euclidean ball and strongly convex action sets readily match the regret lower bounds for linear bandits without sparsity [Lattimore and Szepesvári, 2020].

Notation. For a matrix $P \in \mathbb{R}^{d\times d}$, let $P^\top$, $\mathrm{tr}(P)$, $(P)_{ij}$ be its transpose, trace, and element in i-th row and j-th column, respectively. Let $\lambda_{\min}(A)$, $\lambda_{\max}(A)$ be the minimum and maximum eigenvalues of a positive definite matrix A, respectively. Let I_d be the d-dimensional identity matrix. Denote $[n] \triangleq \{1, \dots, n\}$ for $n \geq 1$. For any $\mathbf{x} \in \mathbb{R}^d$ and any $S \subseteq [d]$, let $\mathbf{x}(S) \in \mathbb{R}^d$ be such that $\mathrm{supp}(\mathbf{x}) = S$ and $(\mathbf{x}(S))_i = (\mathbf{x})_i$ for all $i \in [d]$, where $(\mathbf{x})_i$ denotes the i-th element of $\mathbf{x}$ and $\mathrm{supp}(\mathbf{x}) \triangleq \{i \in [d] : (\mathbf{x})_i \neq 0\}$. Denote by $a_{i:j}$ the sequence $a_i, a_{i+1}, \dots, a_j$. For $y \in \mathbb{R}$, let $|y|$ be its absolute value. For a finite set S, let $|S|$ be its cardinality.

2 Problem Formulation and Preliminaries

The stochastic linear bandit problem models a scenario of learning under uncertainty [Dani *et al.*, 2007; Abbasi-Yadkori *et al.*, 2011; Lattimore and Szepesvári, 2020]. Over T time steps, a decision-maker sequentially picks an action $\mathbf{x}_t$ from a feasible action space $\mathcal{X}_t \subset \mathbb{R}^d$ and receives a reward $Y_t \in \mathbb{R}$, which is a random variable with mean determined by the inner product of the action and an unknown parameter $\theta_* \in \mathbb{R}^d$, i.e., $\mathbb{E}[Y_t|\mathbf{x}_t] = \theta_*^\top \mathbf{x}_t$. The challenge is to devise a policy that minimizes the cumulative regret R_T, which quantifies the sum of performance gaps between the optimal action in hindsight and the chosen action at each step:

$$R_T = \sum_{t=1}^{T} (\max_{\mathbf{x}\in\mathcal{X}} \theta_*^\top \mathbf{x} - \theta_*^\top \mathbf{x}_t). \tag{1}$$

This basic model assumes that the reward Y_t depends on the current action $\mathbf{x}_t$; however, in many cases, the external environment has requirements for the number of non-zero components of the action vector $\mathbf{x}_t$. For instance, i) the number of non-zero components of $\mathbf{x}_t$ must be smaller than H for a deterministic $H \in [d]$ (H is known to the decision maker); ii) some components of the action vector $\mathbf{x}_t$ are not available (remaining 0 at time t). Specifically, letting $S_t = \mathrm{supp}(\mathbf{x}_t)$, at each time step t, the decision maker needs to select

$$\mathbf{x}_t(S_t) \in \mathcal{X},\ s.t.\ S_t \subseteq [d] \text{ and } |S_t| \leq H. \tag{2}$$

We make the following standard assumption on $\mathcal{X}$.

Assumption 1. *The set $\mathcal{X} \subset \mathbb{R}^d$ is closed and bounded.*

Remark 1. *The action set $\mathcal{X}_t$ can generally change over time as in the standard stochastic linear bandits setup, e.g. [Abbasi-Yadkori* et al.*, 2011]. More discussions on this is provided in [Wang* et al.*, 2026].*

The reward Y_t at any time t is then given by

$$Y_t = \theta_*^\top \mathbf{x}_t(S_t) + \eta_t, \tag{3}$$

where η_t is a random noise satisfying the following condition.

Assumption 2. *For any $t \in [T]$, η_t is a random noise drawn from a conditionally σ-sub-Gaussian distribution with unknown variance proxy σ, i.e., $\mathbb{E}[\eta_t \mid \mathbf{x}_{1:t}(S_{1:t}), \eta_{1:t-1}] = 0$, and $\mathbb{E}[\exp(\lambda\eta_t) \mid \mathbf{x}_{1:t}(S_{1:t}), \eta_{1:t-1}] \leq \exp(\frac{\lambda^2\sigma^2}{2})$ $\forall\lambda \in \mathbb{R}$.*

An optimal solution in hindsight (i.e., when θ_* is known) to the sparsifying linear bandits problem described above is given by solving

$$\mathbf{x}^\star(S^\star) \in \underset{\mathbf{x}(S)\in\mathcal{X}, S\subseteq[d], |S|\leq H}{\arg\max} \theta_*^\top \mathbf{x}(S). \tag{4}$$

Finding an optimal action $\mathbf{x}^\star(S^\star)$ involves solving an optimal subset selection problem, which is generally NP-hard [Das and Kempe, 2018; Ye *et al.*, 2025]. This computational barrier typically makes it intractable for any polynomial-time algorithm to compete with the true optimum, necessitating a relaxed performance benchmark known as α-regret [Kalai and Vempala, 2005; Cesa-Bianchi and Lugosi, 2006; Streeter and Golovin, 2008], which is given by

$$\alpha\text{-}R_T \triangleq \alpha \left(\sum_{t=1}^{T} \theta_*^\top \mathbf{x}^\star(S^\star) \right) - \sum_{t=1}^{T} \theta_*^\top \mathbf{x}_t(S_t). \quad (5)$$

When $\alpha = 1$, α-R_T in (5) reduces to R_T in (1). The α-regret defined in (5) compares against an α-approximation of the optimal action, where $\alpha \in (0, 1]$ is the approximation ratio of an algorithm for the offline version of problem (4).

A Relevant Problem Formulation. A sparse online linear optimization problem has been studied in [Ye *et al.*, 2025]. At each step t, the learner selects $\mathbf{x}_t \in \mathbb{R}^d$ and $S_t \subseteq [d]$ with $|S_t| \leq H$ and receives a reward given by $f_t(\mathbf{x}_t, S_t) = \theta_t^\top \mathbf{x}_t(S_t)$, where $\theta_t \in \mathbb{R}^d$ is potentially generated by an adversary. However, there is no regret guarantee for the bandit setting. The online linear optimization framework for a single continuous action (without sparsity consideration) has been studied in [Bubeck *et al.*, 2012], which achieves a $\tilde{\mathcal{O}}(\sqrt{dT})$-regret. In contrast, we study the stochastic linear bandits setting whose reward is a random variable given by (3).

Roadmap. Our algorithm design and regret analysis in the subsequent sections hinge on the geometry of the action set $\mathcal{X}$. We will split our approach into: (i) *Euclidean Ball Action Sets* (Section 3), where we show the optimal sparse action can be computed exactly and efficiently; (ii) *General Strongly Convex Action Sets* (Section 4), where we utilize a greedy approximation algorithm to select a sparse action; and (iii) *Extensions to General Compact Sets* (Section 5), where we address action sets lacking strong convexity.

A fundamental property required for our analysis is the variation of the optimal value function $h(S; \theta)$ with respect to the parameter θ. We present the following proposition to characterize this variation at two levels. First, for general compact action sets, the value function itself is Lipschitz continuous. Second, when the action set is further strongly convex, we obtain a stronger regularity: the gradient of the value function (i.e., the optimal action) also becomes Lipschitz continuous. The latter extends the Smooth Best Arm Response (SBAR) condition used in [Rusmevichientong and Tsitsiklis, 2010] (leveraging the duality properties of strongly convex sets [Polovinkin, 1996, Corollary 4]) to our sparse support framework.[1]

Proposition 1 (Lipschitz Continuity)**.** *Consider a compact set $\mathcal{X}$, and let $L_{\mathcal{X}} = \sup_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}\|_2$. For any fixed support set $S \subseteq [d]$, the value function $h(S; \theta)$ defined as $h(S; \theta) = \sup_{\mathbf{x}\in\mathcal{X}, \mathrm{supp}(\mathbf{x})\subseteq S} \theta^\top \mathbf{x}$ is convex and Lipschitz continuous with respect to θ, i.e., for any $\theta_1, \theta_2 \in \mathbb{R}^d$,*

$$|h(S; \theta_1) - h(S; \theta_2)| \leq L_{\mathcal{X}} \|\theta_1 - \theta_2\|_2 . \quad (6)$$

[1] According to [Polovinkin, 1996, Definition 1], a set $S \subseteq \mathbb{R}^d$ is said to be strongly convex set of radius $R > 0$ if we can represent it as an intersection of closed balls of radius R.

Furthermore, if $\mathcal{X}$ is strongly convex, the gradient of the value function, given by $\nabla_\theta h(S; \theta) = \mathbf{x}^\star(S; \theta)$ (the unique optimal action given $S \subseteq [d]$), is also Lipschitz continuous. That is, there exists a smoothness constant $L_{grad} > 0$ (depending on the curvature of $\mathcal{X}$) such that:

$$\|\mathbf{x}^\star(S; \theta_1) - \mathbf{x}^\star(S; \theta_2)\|_2 \leq L_{grad} \|\theta_1 - \theta_2\|_2 . \quad (7)$$

3 Euclidean Ball Action Sets

We first consider the scenario when $\mathcal{X}$ is an ℓ_2-ball (Euclidean ball) with radius $L_{\max}$, which permits the efficient and exact computation of the optimal sparse action. Furthermore, one can show that $L_{\mathcal{X}} = L_{grad} = L_{\max}$ in Proposition 1. The following proposition formalizes the closed-form solution for the optimal sparse action for ℓ_2-ball action set.

Proposition 2 (Optimal Sparse Action on ℓ_2-Ball)**.** *Let $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_2 \leq L_{max}\}$. For any parameter vector $\theta \in \mathbb{R}^d$, the optimal H-sparse action $\mathbf{x}^\star$ for $\mathbf{x}^\star \in \arg\max_{\mathbf{x}\in\mathcal{X}, \mathrm{supp}(\mathbf{x})\subseteq[d], |\mathrm{supp}(\mathbf{x})|\leq H} \theta^\top \mathbf{x}$ is given by $\mathbf{x}^\star = L_{max} \cdot \frac{\theta(S_H)}{\|\theta(S_H)\|_2}$, where S_H is the set of indices corresponding to the H largest absolute values of θ.*

Hence, finding the best H-sparse action $\mathbf{x}^\star(S^\star)$ for a given θ can be done in polynomial time by selecting the top H largest-magnitude components of θ. However, a critical challenge arises when the support set S_t changes dynamically, and the mapping from a parameter vector θ to its optimal sparse support $S_\theta^\star$ is discontinuous.

Key ideas. To establish the desired $\tilde{\mathcal{O}}(d\sqrt{T})$ regret, it is essential to ensure the proposed algorithm operates on the true optimal support $S_{\theta_*}^\star$ (which is simply denoted as $S^\star$ subsequently). To this end, we propose an *Adaptive Warm-up Strategy* that does not require any prior knowledge of the true signal gap (i.e., the difference between the H-th largest and the $(H+1)$-th largest in the absolute value of the elements in θ_*). Instead of fixing the exploration length a priori, our algorithm continuously monitors the *empirical gap* Δ_c' of the estimated parameter $\hat{\theta}_c$ (i.e., the aforementioned difference computed from $\hat{\theta}_c$). In Lemma 2, we will show that the above approach ensures Algorithm 1 transitions to the exploitation phase after identifying the true optimal support $S^\star$.

3.1 APSEE Algorithm

We now present the Adaptively Phased Sparse Exploration and Exploitation (**APSEE**) algorithm (Algorithm 1). The algorithm operates in cycles indexed by c. The cycle index c initializes as 1 and increments by 1 after each cycle, allowing the algorithm to dynamically adapt time steps during each cycle until the total horizon T is reached. Each cycle begins with an *Exploration Phase*, playing a spanning set of exploration arms. Subsequently, Algorithm 1 updates the OLS estimate $\hat{\theta}_c$ using accumulated data (line 8). To identify the true support $S^\star$, the algorithm performs a *Support Verification* step (lines 9-15): it computes an anytime error bound ε_c, sorts the absolute values based on the estimated $\hat{\theta}_c$ of the components $\{i_1', \ldots, i_d'\}$, and calculates the empirical gap Δ_c'. If the stopping condition $\Delta_c' > 2\varepsilon_c$ is satisfied, the support S_{est}

is locked (i.e., fixed). Finally, the algorithm enters the *Exploitation Phase* (lines 16-20), executing the best action on S_{est} for c steps. A key advantage of APSEE is to operate without requiring prior knowledge of the total time horizon T. To ensure that the OLS estimator utilized in the Exploration Phase is well-defined and possesses the necessary concentration properties for our theoretical analysis, we impose the following regularity condition on the exploration actions.

Assumption 3 (Exploration Actions)**.** *There exists a set of d actions $\{b_1(S_1), \dots, b_d(S_d)\} \subset \mathcal{X}$ with $|S_k| \leq H$ for all $k \in [d]$, such that $B \triangleq \sum_{k=1}^{d} b_k(S_k) b_k(S_k)^\top$ is invertible. Let $\lambda_0 = \lambda_{\min}(B) > 0$. For instance, the set of basis vectors for $\mathbb{R}^d$ satisfies this assumption.*

Remark 2. *The existence of a spanning set of sparse exploration actions in Assumption 3 is a mild and standard one in the linear bandit literature. For example, a similar assumption is made for the analysis of the phased exploration strategy in [Rusmevichientong and Tsitsiklis, 2010, Assumption 1(b)].Finding such a spanning set is straightforward; for instance, the standard basis vectors satisfy the condition. Later in Section 6, we empirically show the consistent sub-linear regret across varying choices of basis actions (e.g., standard, Gaussian-random, and uniform-random bases) as long as the invertibility condition of B holds.*

3.2 Adaptive Support Recovery

To start with, we establish the theoretical correctness of our adaptive support detection mechanism (lines 9-15 of Algorithm 1). Since this mechanism relies on the OLS estimator $\hat{\theta}_c$ computed from exploration actions, we provide a high-probability anytime upper bound on the estimation error.

Lemma 1 (Anytime High-Probability Bound on OLS Error)**.** *Consider the OLS estimator $\hat{\theta}_c$ (expressed as (8) in Algorithm 1). Let $\delta \in (0,1)$ and $h_1 = \frac{2\sigma^2 \mathrm{Tr}(B)}{\lambda_0^2}$. With probability at least $1-\delta$, for **all** cycles $c \geq 1$ in Algorithm 1 simultaneously, the estimation error satisfies:*

$$\left\| \hat{\theta}_c - \theta_* \right\|_2^2 \leq \frac{h_1 \ln(2dc^2/\delta)}{c} \triangleq \varepsilon_c^2. \tag{9}$$

Building on this error bound of $\hat{\theta}_c$, we demonstrate that monitoring the empirical gap $\Delta'_c = |(\hat{\theta}_c)_{i'_H}| - |(\hat{\theta}_c)_{i'_{H+1}}|$ is sufficient to guarantee that the estimated support $S_c^\star$ (with $S_c^\star = S_{est}$ in line 14) matches the true support $S^\star$.

Lemma 2 (Support Consistency for Euclidean Ball Action Sets)**.** *Consider any $c \geq 1$. Suppose $\left\| \hat{\theta}_c - \theta_* \right\|_2 \leq \varepsilon_c$. Let the sorted indices of θ_* be $\{i_1^\star, \dots, i_d^\star\}$ such that $|(\theta_*)_{i_1^\star}| \geq \cdots \geq |(\theta_*)_{i_d^\star}|$, i.e., the true support is $S^\star = \{i_1^\star, \dots, i_H^\star\}$. Similarly, let the sorted indices of $\hat{\theta}_c$ be $\{i'_1, \dots, i'_d\}$ with estimated support $S_c^\star = \{i'_1, \dots, i'_H\}$. If the empirical gap satisfies $\Delta'_c = |(\hat{\theta}_c)_{i'_H}| - |(\hat{\theta}_c)_{i'_{H+1}}| > 2\varepsilon_c$, then $S_c^\star = S^\star$.*

This result provides the theoretical foundation for the warm-up (pure exploration) phase of our algorithm. Once the condition $\Delta'_c > 2\varepsilon_c$ is met, we can safely get into the next exploitation phase and incur low regret.

Algorithm 1 APSEE

Input: Dimension d, sparsity H, confidence parameter δ.
Requires: Exploration set $\{b_1(S_1), \dots, b_d(S_d)\}$ from Assumption 3.

1: **Initialize**: $B = \sum_{k=1}^{d} b_k(S_k) b_k(S_k)^\top$, $c = 1$, $t = 1$.
2: **Flag**: `SupportFound` = False, $S_{est} = \emptyset$.
3: **for** $c = 1, 2, \dots$ **do**
— Exploration Phase (d steps) —
4: **for** $k = 1, \dots, d$ **do**
5: Play action b_k and observe reward Y_t.
6: Set $Y_k(c) = Y_t$.
7: $t \leftarrow t + 1$.
8: Compute OLS estimate $\hat{\theta}_c$ using all past data:

$$\hat{\theta}_c = (cB)^{-1} \sum_{s=1}^{c} \sum_{k=1}^{d} b_k Y_k(s). \tag{8}$$

9: **if** `SupportFound` is False **then**
10: Compute error bound $\varepsilon_c = \sqrt{\frac{h_1 \ln(2dc^2/\delta)}{c}}$.
11: Sort indices $\{i'_1, \dots, i'_d\}$ such that $|(\hat{\theta}_c)_{i'_1}| \geq \cdots \geq |(\hat{\theta}_c)_{i'_d}|$.
12: Compute $\Delta'_c = |(\hat{\theta}_c)_{i'_H}| - |(\hat{\theta}_c)_{i'_{H+1}}|$.
13: **if** $\Delta'_c > 2\varepsilon_c$ **then**
14: Set $S_{est} = \{i'_1, \dots, i'_H\}$.
15: `SupportFound` $\leftarrow$ True.
— Exploitation Phase (c steps) —
16: **if** `SupportFound` is True **then**
17: Compute the best action on support S_{est}: $a_c = \arg\max_{\mathbf{x} \in \mathcal{X}, \mathrm{supp}(\mathbf{x}) \subseteq S_{est}} \hat{\theta}_c^\top \mathbf{x}$.
18: **for** $j = 1, \dots, c$ **do**
19: Play action a_c and observe reward Y_t.
20: $t \leftarrow t + 1$.
21: $c \leftarrow c + 1$.

3.3 Regret Analysis for Algorithm 1

Having established that Algorithm 1 correctly identifies the support $S^\star$ before entering the exploitation phase, we can now bound its cumulative regret defined in (5). Note that we have $\alpha = 1$ in (5), due to Proposition 2.

Theorem 1 (Regret Bounds for APSEE)**.** *Suppose Assumptions 1-3 hold. Assume the unknown parameter θ_* has a non-zero signal gap $\Delta_{\min} = |(\theta_*)_{i_H^*}| - |(\theta_*)_{i_{H+1}^*}| > 0$, where i_H^* and i_{H+1}^* are described in Lemma 2. Let $\delta \in (0,1)$. Then, with probability at least $1-\delta$:*

- *If $T > dC_0$, the cumulative regret satisfies*

$$R_T = \tilde{\mathcal{O}}\left(\left(\|\theta_*\|_2 + \frac{1}{\|\theta_*\|_2} \right) d\sqrt{T} \right). \tag{10}$$

- *If $T \leq dC_0$, the cumulative regret satisfies*

$$R_T = \tilde{\mathcal{O}}\left(\frac{dL_{\max} \|\theta_*\|_2 h_1}{\Delta_{\min}^2} \right). \tag{11}$$

Here, $\tilde{\mathcal{O}}(\cdot)$ hides $\log T$ factor and $C_0 = \mathcal{O}(h_1 \Delta_{\min}^{-2})$ is the duration of the warm-up phase in Algorithm 1.

Our analysis reveals that even with the additional constraint of sparse actions, we achieve a $\tilde{\mathcal{O}}(d\sqrt{T})$ regret matching standard linear bandits (without sparsity) [Abbasi-Yadkori *et al.*, 2011], which is known to be the best-achievable regret for standard linear bandits [Lattimore and Szepesvári, 2020]. Investigating the matching $\Omega(d\sqrt{T})$ regret lower bound in our sparse linear bandit setting is left as future work.

4 General Strongly Convex Action Sets

We now consider broader class of action sets $\mathcal{X}$ that are strongly convex, including ellipsoids and ℓ_p-balls for $p \in (1, 2]$, satisfying the strict positive curvature conditions [Polovinkin, 1996; Vial, 1982]. Unlike ℓ_2-ball, an explicit closed-form solution for the sparse optimization problem is generally unavailable for these sets [Natarajan, 1995]. Thus, obtaining the exact optimal sparse action typically requires enumerating all possible support sets, incurring high computational cost. To address this challenge, we adopt the greedy algorithm [Nemhauser *et al.*, 1978], an efficient heuristic for combinatorial optimization. While the greedy algorithm may not recover optimal solution, it guarantees an α-approximate one, where the approximation ratio α is determined by the *submodularity ratio* (defined later) of the objective function.

4.1 Adaptive Support Recovery via Greedy Gap and APSEE-G Algorithm

We introduce the Adaptively Phased Sparse Exploration and Exploitation with Greedy (**APSEE-G**) algorithm (Algorithm 2). To start with, define the maximum expected reward achievable on a fixed support set $S \subseteq [d]$ with parameter θ as

$$h(S;\theta) \triangleq \max_{\mathbf{x}\in\mathcal{X}, \text{supp}(\mathbf{x})\subseteq S, |S|\le H, S\subseteq[d]} \theta^\top \mathbf{x}. \tag{12}$$

For a general strongly convex set $\mathcal{X}$, the function $h(S;\theta)$ is convex and Lipschitz continuous with respect to θ according to Proposition 1. Let S^G and S_c^G be the support sets obtained by running the greedy algorithm on the true parameter θ_* and the estimate $\hat{\theta}_c$ (obtained as (8) in Algorithm 1), respectively. Specifically, running on any given parameter θ, the greedy algorithm constructs the support set iteratively: starting with an empty set $G_0 \leftarrow \emptyset$, at each step $k = 1, \dots, H$, it adds an element $v \in [d] \setminus G_{k-1}$ to G_{k-1} that maximizes the marginal gain $\rho_k(v;\theta) = h(G_{k-1} \cup \{v\};\theta) - h(G_{k-1};\theta)$.

Greedy Gap and Lipschitz Continuity

We analyze the consistency of the greedy algorithm by defining the gap at each step $k \in \{1, \dots, H\}$. Let G_{k-1}^* be the set of size $k-1$ selected by the greedy algorithm given θ_*. Then, the marginal gain of an element v is $\rho_k(v;\theta_*) = h(G_{k-1}^* \cup \{v\};\theta_*) - h(G_{k-1}^*;\theta_*)$ for all $v \in [d] \setminus G_{k-1}^*$. Let g_k^* be the best element and v_k^{second} be the second-best element at step k, i.e., $g_k^* = \arg\max_{v\in[d]\setminus G_{k-1}^*} \rho_k(v;\theta_*)$ and $v_k^{\text{second}} = \arg\max_{v'\in[d]\setminus G_{k-1}^*, v'\neq g_k^*} \rho_k(v';\theta_*)$. We define: (i) **Step-wise Greedy Gap:** $\delta_k = \rho_k(g_k^*;\theta_*) - \rho_k(v_k^{\text{second}};\theta_*)$;

(ii) **Minimum Greedy Gap:** $\Delta_{\min} = \min_{k=1,\dots,H} \delta_k$. (13)

Algorithm 2 APSEE-G

Input: Dimension d, sparsity H, confidence parameter δ, action set $\mathcal{X}$.

Requires: Exploration set $\{b_1(S_1), \dots, b_d(S_d)\}$, Lipschitz constant $L_{\mathcal{X}}$.

1: **Initialize**: $B = \sum_{k=1}^{d} b_k(S_k) b_k(S_k)^\top$, $c = 1$, $t = 1$.
2: **Flag**: `SupportFound` = False, $S_{est} = \emptyset$.
3: **for** $c = 1, 2, \dots$ **do**
— Exploration Phase (d steps) —
4: Run lines 3-10 of Algorithm 1.
// Greedy Process (lines 5-13)
5: Let $G_0 = \emptyset$ and $\Delta_c^{Greedy} = \infty$.
6: **for** $k = 1, \dots, H$ **do**
7: Compute candidate marginal gains:
8: $\rho_k(v;\hat{\theta}_c) = h(G_{k-1} \cup \{v\};\hat{\theta}_c) - h(G_{k-1};\hat{\theta}_c)$, $\forall v \in [d] \setminus G_{k-1}$.
9: Find $g_k = \arg\max_v \rho_k(v;\hat{\theta}_c)$.
10: Find $v_{second} = \arg\max_{v\neq g_k} \rho_k(v;\hat{\theta}_c)$.
11: Compute $\hat{\delta}_k = \rho_k(g_k;\hat{\theta}_c) - \rho_k(v_{second};\hat{\theta}_c)$.
12: Update gap: $\Delta_c^{Greedy} = \min(\Delta_c^{Greedy}, \hat{\delta}_k)$.
13: Update set: $G_k = G_{k-1} \cup \{g_k\}$.
14: **if** $\Delta_c^{Greedy} > 2L_{\mathcal{X}}\varepsilon_c$ **then**
15: Set $S_{est} = G_H$.
16: `SupportFound` $\leftarrow$ True.
— Exploitation Phase (c steps) —
17: **if** `SupportFound` is True **then**
18: Compute optimal action on support S_{est}: $a_c = \arg\max_{\mathbf{x}\in\mathcal{X}, \text{supp}(\mathbf{x})\subseteq S_{est}} \hat{\theta}_c^\top \mathbf{x}$.
19: **for** $j = 1, \dots, c$ **do**
20: Play action a_c and observe reward Y_t.
21: $t \leftarrow t + 1$.
22: $c \leftarrow c + 1$.

Without loss of generality, we assume that $\Delta_{\min} > 0$; otherwise, one can consider the third best element v_k^{third} such that $v_k^{\text{third}} \neq v_k^{\text{second}}$ and $v_k^{\text{third}} \neq g_k^*$. To relate the estimation error to the true support recovery, we again use the Lipschitz property (Proposition 1). Given that $\left\|\hat{\theta}_c - \theta_*\right\|_2 \le \varepsilon_c$, we have $|h(S;\hat{\theta}_c) - h(S;\theta_*)| \le L_{\mathcal{X}}\varepsilon_c$, where $L_{\mathcal{X}} = \sup_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}\|_2$ in Proposition 1 and ε_c is defined in Lemma 1. Consequently, the error in estimating the marginal gain $\rho_k(v;\cdot)$ (which is the difference of two $h(\cdot)$ values) is bounded by:

$$\begin{aligned} |\rho_k(v;\hat{\theta}_c) - \rho_k(v;\theta_*)| &\le \Big| h(G \cup \{v\};\hat{\theta}_c) \\ &\quad - h(G \cup \{v\};\theta_*) \Big| + |h(G;\hat{\theta}_c) - h(G;\theta_*)| \\ &\le L_{\mathcal{X}}\varepsilon_c + L_{\mathcal{X}}\varepsilon_c = 2L_{\mathcal{X}}\varepsilon_c. \end{aligned} \tag{14}$$

Stopping Condition and Consistency Lemma

To guarantee $S_c^G = S^G$ for some $c \ge 1$, the idea is that the empirical gap Δ_c^{Greedy} (calculated in line 12 of Algorithm 2 using the estimate $\hat{\theta}_c$) must be large enough to overcome the estimation error of $\hat{\theta}_c$. Specifically, we need to distinguish

the best element g_k^* from the second-best element v_k^{second} despite the estimation error in $\hat{\theta}_c$. Similar to the discussions in Section 3.2, we prove the following result.

Lemma 3 (Support Consistency for Strongly Convex Sets)**.** *Suppose* $\left\|\hat{\theta}_c - \theta_*\right\|_2 \leq \varepsilon_c$ *holds. Let* Δ_c^{Greedy} *be the minimum empirical marginal gain gap observed during the greedy algorithm execution on* $\hat{\theta}_c$*. If the stopping condition in line 14 holds:* $\Delta_c^{Greedy} > 2L_{\mathcal{X}}\varepsilon_c$*, then the support set* S_c^G *obtained by running the greedy algorithm on* $\hat{\theta}_c$ *is identical to the support set* S^G *obtained by running the greedy algorithm on the true parameter* θ_**.*

4.2 Regret Analysis for Algorithm 2

We introduce the submodularity ratio for set functions, which plays a crucial role in proving the approximation ratio of greedy algorithms [Das and Kempe, 2018].

Definition 1. *The submodularity ratio of a set function* $g : 2^{[n]} \to \mathbb{R}_{\geq 0}$ *is the largest* $\gamma \in \mathbb{R}$ *such that for all* $S, \Omega \subseteq [n]$*,*

$$\sum_{\omega \in \Omega \setminus S} \left(g(S \cup \{\omega\}) - g(S)\right) \geq \gamma \left(g(S \cup \Omega) - g(S)\right).$$

Recalling the α-regret defined in (5), we will let $\alpha = 1 - e^{-\gamma}$ and derive the corresponding α-regret upper bound, where γ is the submodularity ratio of the set function $h(S;\theta)$ (with respect to $S \subseteq [d]$) defined in (12). We now show the following essential result that $\gamma > 0$, which implies that $\alpha > 0$, i.e., the α-regret is not vacuous.

Proposition 3 (Submodularity Ratio for General (Strongly) Convex Sets)**.** *Let* $\mathcal{X}$ *be a convex compact set containing the origin. For any fixed parameter* θ*, consider the set function* $h(S;\theta) = \max_{\mathbf{x} \in \mathcal{X}, \text{supp}(\mathbf{x}) \subseteq S} \theta^\top \mathbf{x}$*. The submodularity ratio* γ *of* $h(\cdot;\theta)$ *is lower bounded by:*

$$\gamma \geq \min_{S, \Omega \subseteq [d]} \frac{\sum_{\omega \in \Omega \setminus S} \max_{\mathbf{x} \in \mathcal{X}, \text{supp}(\mathbf{x}) \subseteq \{\omega\}} \theta^\top \mathbf{x}}{h(S \cup \Omega; \theta) - h(S; \theta)}. \quad (15)$$

Moreover, $\gamma > 0$ *when* θ *satisfies* $\exists \mathbf{x} \in \mathcal{X},\ s.t.\ \theta^\top \mathbf{x} \neq 0$*.*

It is worth mentioning that when $\mathcal{X}$ is an ellipsoid $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{x}^\top A \mathbf{x} \leq 1\}$, where A is positive definite and assumed without loss of generality that $\lambda_{\max}(A) \leq 1$, the submodularity ratio γ of the set function $h(\cdot)$ in Proposition 3 satisfies that $\gamma > \lambda_{\min}(A)$. More discussions on the ellipsoid case are deferred to [Wang *et al.*, 2026]. Under the above guarantees and the condition that the greedy supports match (i.e., $S_c^G = S^G$), we derive the regret upper bound for Algorithm 2.

Theorem 2 (Regret bounds for APSEE-G)**.** *Suppose Assumptions 1-3 hold. Let* $L_{max} = \max_{\mathbf{x} \in \mathcal{X}} \|\mathbf{x}\|_2$*. Assume the true parameter* θ_* *satisfies the Minimum Greedy Gap* (13) *condition with* $\Delta_{\min} > 0$*. Let* $\delta \in (0,1)$*. Then, with probability at least* $1 - \delta$*:*

• *If* $T > dC_0$*, the cumulative* α*-regret satisfies*

$$\alpha\text{-}R_T = \tilde{\mathcal{O}}\left(d\sqrt{T}\left(\|\theta_*\|_2 + \frac{1}{\|\theta_*\|_2}\right)\right). \quad (16)$$

• *If* $T \leq dC_0$*, the cumulative* α*-regret satisfies*

$$\alpha\text{-}R_T = \tilde{\mathcal{O}}\left(\frac{dL_{\max}\|\theta_*\|_2 h_1}{\Delta_{\min}^2}\right). \quad (17)$$

Here, $\tilde{\mathcal{O}}(\cdot)$ *hides* $\log T$ *factor;* $C_0 = \mathcal{O}(h_1 \Delta_{\min}^{-2})$ *is the warm-up phase length in Algorithm 2;* $\alpha = 1 - e^{-\gamma}$*, and* γ *is the submodularity ratio of* $h(\cdot;\theta_*)$ *defined as* (12)*.*

5 General Compact Action Sets

When the action set $\mathcal{X}$ is a general compact set (e.g., a hypercube or a polytope), it does not necessarily possess the strong convexity property (curvature) as the Euclidean balls or ellipsoids. As discussed in Section 4, we use the submodularity ratio γ of the set function $h(S;\theta) = \max_{\mathbf{x} \in \mathcal{X}, \text{supp}(\mathbf{x}) \subseteq S} \theta^\top \mathbf{x}$ (with $S \subseteq [d]$) to parameterize the α-regret as $\alpha = 1 - e^{-\gamma}$. To ensure $\gamma > 0$ for general compact set $\mathcal{X}$, we make the following assumption.

Assumption 4 (Convex Hull Regularity)**.** *Let* $\mathcal{X}^\star = \{\mathbf{x}^\star(S) : S \subseteq [d], |S| \leq H\}$*, where* $\mathbf{x}^\star(S) \in \arg\max_{\mathbf{x} \in \mathcal{X}, \text{supp}(\mathbf{x}) \subseteq S} \theta_*^\top \mathbf{x}$*. Assume that* $\text{conv}(\mathcal{X}^\star) \subseteq \mathcal{X}$*.*[2]

One can show that Assumption 4 directly holds for convex $\mathcal{X}$. Under Assumption 4, we show in [Wang *et al.*, 2026] that $h(\cdot;\theta_*)$ possesses a strictly positive submodularity ratio $\gamma > 0$. We then prove the following regret bounds for a modified version of Algorithm 2.

Theorem 3 (Regret Bounds for General Compact Sets)**.** *Suppose Assumptions 1-4 hold. Run a modified version of Algorithm 2, which skips the support verification step (lines 14-17) and sets the exploitation length (line 19) to be* $\lfloor \sqrt{c} \rfloor$ *for every cycle* $c = 1, 2, \ldots$*. Let* $\delta \in (0,1)$*. Then, with probability at least* $1 - \delta$*, the cumulative* α*-regret satisfies* $\alpha\text{-}R_T = \tilde{\mathcal{O}}\left(dT^{2/3}\right)$*, where* $\tilde{\mathcal{O}}(\cdot)$ *hides* $\log T$ *factor,* $\alpha = 1 - e^{-\gamma}$*, and* γ *is the submodularity ratio of* $h(\cdot;\theta_*)$ *defined as* (12)*.*

6 Numerical Experiments

We empirically validate the theoretical performance and core mechanisms of our proposed algorithms. We focus our primary evaluation on the ellipsoid action set, which belongs to the strongly convex setting discussed in Section 4. We refer the readers to [Wang *et al.*, 2026] for additional experiments on Euclidean balls, general compact sets, and a recommendation system application.

Regret Performance on Ellipsoid Action Sets. We fix the action dimension to $d = 20$ and the sparsity constraint to $H = 5$. The true parameter θ_* is drawn uniformly from the hypercube $[-1, 1]^d$. We simulate the environment for 2,000 time steps, where the noise η_t in the reward is i.i.d. Gaussian noise drawn from $\mathcal{N}(0, 0.5^2)$. The results are based on 20 independent trials. We construct an ellipsoid action set $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{x}^\top A \mathbf{x} \leq 1\}$, where A is a randomly generated positive definite matrix with $\lambda_{\max}(A) \leq 1$. We employ the **APSEE-G** algorithm (Algorithm 2) and validate the results established in Theorem 2. Figure 1 (Left) illustrates the cumulative α-regret of Algorithm 2. The scatter cloud highlights the variance across different runs, while the mean trajectory demonstrates a sub-linear growth, indicating

[2]The convex hull of a set S, denoted $\text{conv}(S)$, is the intersection of all convex sets containing S [Bertsekas, 2015].

that the algorithm successfully learns the optimal sparse action. To verify the regret bounds provided in Theorem 2, Figure 1 (Right) plots the α-regret normalized by $\sqrt{T}$. The curve of the mean normalized regret stabilizes and exhibits a slow, logarithmic growth trend, confirming the $\tilde{\mathcal{O}}(\sqrt{T})$ bound established in Theorem 2.

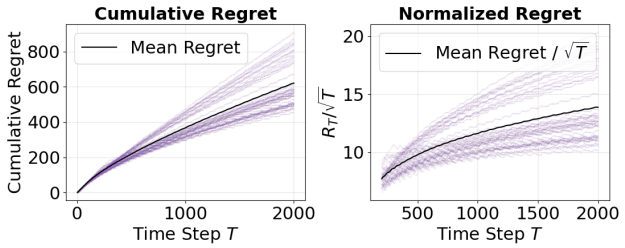

Figure 1: Numerical results for the APSEE-G algorithm.

Impact of Basis Actions on Exploration. We examine the sensitivity of the APSEE-G algorithm (Algorithm 2) to the choice of basis actions utilized during the exploration phase. We conduct experiments using the following three distinct sets of basis actions under the same ellipsoid action set: (i) the standard basis vectors in $\mathbb{R}^d$ (consistent with the main evaluation), (ii) a random orthogonal basis generated from a standard Gaussian distribution, and (iii) a random orthogonal basis sampled from a uniform distribution. All these sets of basis actions satisfy the invertibility condition in Assumption 3. As illustrated in Figure 2, the cumulative α-regret trajectories of APSEE-G under these different basis sets are nearly indistinguishable, all exhibiting matching sub-linear growth rates. This empirical evidence corroborates the theoretical claims derived from Lemmas 2 and 3: provided that the chosen basis actions span the necessary space to satisfy Assumption 3, the exact distribution of the basis actions in the exploration phase has a negligible impact on the overall regret of the algorithm.

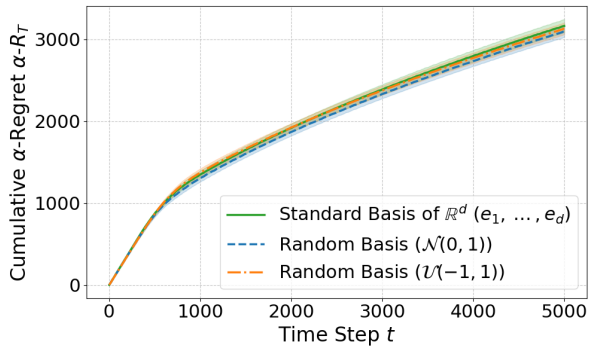

Figure 2: Cumulative α-regret of the APSEE-G algorithm under different choices of basis actions in the exploration phase.

Empirical Verification of Support Recovery. While the recovery of the true support $S^\star$ is theoretically justified in Lemmas 2 and 3, we show that this guarantee holds in practical evaluations. To this end, we conduct additional experiments to track the real-time support recovery process of the APSEE algorithm (Algorithm 1) under two distinct configurations of the ground-truth parameter vector $\theta_* \in \mathbb{R}^{20}$: a "standard" θ_* (where the non-zero elements have more separated absolute values, making them easily identifiable) and an "adversarial" θ_* (where the absolute values of the non-zero elements are nearly identical, making them highly difficult to distinguish). Both parameter vectors are constructed to share the exact same signal gap $\Delta_{\min}$ (defined as the absolute difference between the H-th and $(H+1)$-th largest elements of θ_*). According to our theoretical proofs, sharing an identical $\Delta_{\min}$ ensures that the theoretical worst-case stopping time for the exploration phase, $t^* = \mathcal{O}(1/\Delta_{\min}^2)$, is exactly the same for both cases. Figure 3 plots the evolution of the support recovery over time. For these two distinct cases, we conduct 50 trials each and plot scatter plots as well as average curves. The results clearly demonstrate that despite the differing difficulty landscapes of the standard and adversarial parameter vectors, the APSEE algorithm successfully identifies the true support $S^\star$ within the theoretical stopping time t^*.

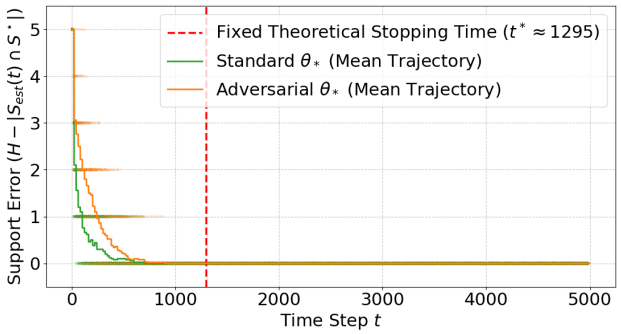

Figure 3: Support recovery performance of the APSEE algorithm over time under standard and adversarial θ_* vectors with the same signal gap $\Delta_{\min}$.

7 Conclusion

We formulated and analyzed the problem of learning to sparsify stochastic linear bandits with sparse action constraints. We proposed a novel algorithmic framework that effectively decouples parameter learning via OLS from combinatorial action selection. By leveraging an adaptive warm-up mechanism, we achieved $\tilde{\mathcal{O}}(d\sqrt{T})$ regret for Euclidean and strongly convex action sets, and $\tilde{\mathcal{O}}(dT^{2/3})$ regret for compact sets. The regret upper bound $\tilde{\mathcal{O}}(d\sqrt{T})$ readily matches the minimax lower bound for standard linear stochastic bandits. Our results bridge the gap between high-dimensional linear bandits and online combinatorial optimization, opening future avenues for bandits or general RL problems with sparsity constraints on the chosen actions or policies.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62525603, U25A20469 and U24A20268; in part by the Key Research and Development Program of Hubei Province under Grants 2025BEB009. Please contact Lintao Ye (yelintao93@hust.edu.cn) for correspondence.

References

[Abbasi-Yadkori *et al.*, 2011] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Proc. of Conference on Neural Information Processing Systems*, pages 2312–2320, 2011.

[Abbasi-Yadkori *et al.*, 2012] Yasin Abbasi-Yadkori, David Pal, and Csaba Szepesvari. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Proc. of Artificial Intelligence and Statistics*, pages 1–9, 2012.

[Adibi *et al.*, 2022] Arman Adibi, Aryan Mokhtari, and Hamed Hassani. Minimax optimization: The case of convex-submodular. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 3556–3580, 2022.

[Agrawal and Goyal, 2013] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proc. of International conference on machine learning*, pages 127–135, 2013.

[Auer *et al.*, 2002] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.

[Bastani *et al.*, 2021] Hamsa Bastani, Mohsen Bayati, and Khashayar Khosravi. Mostly exploration-free algorithms for contextual bandits. *Management Science*, 67(3):1329–1349, 2021.

[Bertsekas, 2015] Dimitri Bertsekas. *Convex optimization algorithms*. Athena Scientific, 2015.

[Bian *et al.*, 2017] Andrew An Bian, Joachim M Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *Proc. of International Conference on Machine Learning*, pages 498–507, 2017.

[Bubeck *et al.*, 2012] Sébastien Bubeck, Nicolo Cesa-Bianchi, and Sham M Kakade. Towards minimax policies for online linear optimization with bandit feedback. In *Proc. of Conference on Learning Theory*, pages 41–1, 2012.

[Bunton and Tabuada, 2022] Jonathan Bunton and Paulo Tabuada. Joint continuous and discrete model selection via submodularity. *Journal of Machine Learning Research*, 23(329):1–42, 2022.

[Carpentier and Munos, 2012] Alexandra Carpentier and Rémi Munos. Bandit theory meets compressed sensing for high dimensional stochastic linear bandit. In *Proc. of Artificial Intelligence and Statistics*, pages 190–198, 2012.

[Cesa-Bianchi and Lugosi, 2006] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

[Chen *et al.*, 2013] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *Proc. of International Conference on Machine learning*, pages 151–159, 2013.

[Chen *et al.*, 2016] Wei Chen, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu. Combinatorial multi-armed bandit with general reward functions. In *Proc. of Conference on Neural Information Processing Systems*, pages 1659–1667, 2016.

[Chu *et al.*, 2011] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[Dai *et al.*, 2023] Yan Dai, Ruosong Wang, and Simon Shaolei Du. Variance-aware sparse linear bandits. In *Proc. of International Conference on Learning Representations*, pages 13657–13692, 2023.

[Dani *et al.*, 2007] Varsha Dani, Thomas P Hayes, and Sham M Kakade. The price of bandit information for online optimization. In *Proc. of Conference on Neural Information Processing Systems*, pages 345–352, 2007.

[Danskin, 2012] John M Danskin. *The theory of max-min and its application to weapons allocation problems*, volume 5. Springer Science & Business Media, 2012.

[Das and Kempe, 2011] Abhimanyu Das and David Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proc. of International Conference on Machine Learning*, pages 1057–1064, 2011.

[Das and Kempe, 2018] Abhimanyu Das and David Kempe. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *Journal of Machine Learning Research*, 19(3):1–34, 2018.

[Fourati *et al.*, 2024] Fares Fourati, Christopher John Quinn, Mohamed-Slim Alouini, and Vaneet Aggarwal. Combinatorial stochastic-greedy bandit. In *Proc. of the AAAI Conference on Artificial Intelligence*, pages 12052–12060, 2024.

[Gai *et al.*, 2012] Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking*, 20(5):1466–1478, 2012.

[Glasgow and Rakhlin, 2023] Margalit Glasgow and Alexander Rakhlin. Tight bounds for γ-regret via the decision-estimation coefficient. *arXiv preprint arXiv:2303.03327*, 2023.

[Golovin *et al.*, 2014] Daniel Golovin, Andreas Krause, and Matthew Streeter. Online submodular maximization under a matroid constraint with application to learning assignments. *arXiv preprint arXiv:1407.1082*, 2014.

[Hao *et al.*, 2020] Botao Hao, Tor Lattimore, and Mengdi Wang. High-dimensional sparse linear bandits. In *Proc. of Conference on Neural Information Processing Systems*, pages 10753–10763, 2020.

[Horn and Johnson, 2012] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge University Press, 2012.

[Ichikawa *et al.*, 2024] Koji Ichikawa, Shinji Ito, Daisuke Hatano, Hanna Sumita, Takuro Fukunaga, Naonori Kakimura, and Ken-ichi Kawarabayashi. New classes of the greedy-applicable arm feature distributions in the sparse linear bandit problem. In *Proc. of the AAAI Conference on Artificial Intelligence*, pages 12708–12716, 2024.

[Jin *et al.*, 2024] Tianyuan Jin, Kyoungseok Jang, and Nicolò Cesa-Bianchi. Sparsity-agnostic linear bandits with adaptive adversaries. In *Proc. of Conference on Neural Information Processing Systems*, pages 42015–42047, 2024.

[Kalai and Vempala, 2005] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

[Kim and Paik, 2019] Gi-Soo Kim and Myunghee Cho Paik. Doubly-robust lasso bandit. In *Proc. of Conference on Neural Information Processing Systems*, pages 5877–5887, 2019.

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

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proc. of International Conference on World Wide Web*, pages 661–670, 2010.

[Liu and Song, 2025] Jingyu Liu and Yanglei Song. Minimax rate-optimal algorithms for high-dimensional stochastic linear bandits. *arXiv preprint arXiv:2505.17400*, 2025.

[Natarajan, 1995] Balas Kausik Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995.

[Nemhauser *et al.*, 1978] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978.

[Nie *et al.*, 2023] Guanyu Nie, Yididiya Y Nadew, Yanhui Zhu, Vaneet Aggarwal, and Christopher John Quinn. A framework for adapting offline algorithms to solve combinatorial multi-armed bandit problems with bandit feedback. In *Proc. of International Conference on Machine Learning*, pages 26166–26198, 2023.

[Oh *et al.*, 2021] Min-hwan Oh, Garud Iyengar, and Assaf Zeevi. Sparsity-agnostic lasso bandit. In *Proc. of International Conference on Machine Learning*, pages 8271–8280, 2021.

[Pacchiano *et al.*, 2022] Aldo Pacchiano, Christoph Dann, and Claudio Gentile. Best of both worlds model selection. In *Proc. of Conference on Neural Information Processing Systems*, pages 1883–1895, 2022.

[Polovinkin, 1996] Evgenii Sergeevich Polovinkin. Strongly convex analysis. *Sbornik: Mathematics*, 187(2):259, 1996.

[Rusmevichientong and Tsitsiklis, 2010] Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

[Russo *et al.*, 2018] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.

[Streeter and Golovin, 2008] Matthew Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *Proc. of Conference on Neural Information Processing Systems*, pages 1577–1584, 2008.

[Vannella *et al.*, 2023a] Filippo Vannella, Alexandre Proutiere, and Jaeseong Jeong. Statistical and computational trade-off in multi-agent multi-armed bandits. In *Proc. of Conference on Neural Information Processing Systems*, pages 69021–69031, 2023.

[Vannella *et al.*, 2023b] Filippo Vannella, Alexandre Proutiere, and Jaeseong Jeong. Best arm identification in multi-agent multi-armed bandits. In *Proc. of International Conference on Machine Learning*, pages 34875–34907, 2023.

[Vial, 1982] Jean-Philippe Vial. Strong convexity of sets and functions. *Journal of Mathematical Economics*, 9(1-2):187–205, 1982.

[Wainwright, 2019] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

[Wan *et al.*, 2023] Zongqi Wan, Jialin Zhang, Wei Chen, Xiaoming Sun, and Zhijie Zhang. Bandit multi-linear dr-submodular maximization and its applications on adversarial submodular bandits. In *Proc. of International Conference on Machine Learning*, pages 35491–35524, 2023.

[Wang *et al.*, 2026] Zhengmiao Wang, Ming Chi, Zhi-Wei Liu, Lintao Ye, and Carla Fabiana Chiasserini. Learning to sparsify stochastic linear bandits. *arXiv preprint arXiv:2605.10151*, 2026.

[Ye *et al.*, 2025] Lintao Ye, Ming Chi, Zhi-Wei Liu, Xiaoling Wang, and Vijay Gupta. Online mixed discrete and continuous optimization: Algorithms, regret analysis and applications. *Automatica*, 175:112189, 2025.